PKD SLC
93/MOS Ref.

A KNOWLEDGE BASED DATABASE SYSTEM FOR JET IMPINGEMENT HEAT TRANSFER CORRELATIONS.

MICHAEL ANDREW MOSS

A thesis submitted in partial fulfilment of
the requirements of The Nottingham Trent University
for the degree of Doctor of Philosophy

August 1993

The Nottingham Trent University
Nottingham

For Cathy and Andrew naturally
and for our memory of Anthony

Abstract

A KNOWLEDGE BASED DATABASE SYSTEM FOR JET IMPINGEMENT HEAT TRANSFER CORRELATIONS

Michael Andrew Moss

A database containing correlations and graphical data for the heat transfer due to single and multiple impinging air jets has been assembled. A knowledge based database system has been constructed which assists the formulation of a simple description of geometry and flow and, using this description, retrieves and evaluates only the relevant and valid information. Where the information in the database does not satisfy this specification the system has the capability either to select the most appropriate information for extrapolation or to modify the database query to select alternative information.

Available selection techniques, both in database systems and in knowledge based systems, are not sufficiently specific for a correlation information system. Consequently a network of descriptive keywords and associated information, based on a detailed review of the jet impingement heat transfer literature, is used to control the selection process. Additional information, associated with each correlation, is used in determining whether the correlation is appropriate to the geometry and flow described.

The program described not only enables new correlations to be added as they appear but also enables the knowledge in the network to be extended to accommodate the addition of correlations for new geometries and flow conditions. Because of the integration of the knowledge base and the database new information must satisfy additional constraints, which are identified, to enable it to be interpreted correctly.

The information in the system developed is confined to the heat transfer due to impinging jets but it is believed that the methods developed are more widely applicable.

A correlation for a single orthogonally impinging air jet has been developed from experimental data extracted from the literature. Heat transfer coefficients may be derived for jet Reynolds numbers in the range 5000 to 124000 and nozzle-to-plate spacings of 1.2, 5 and 10. The correlation provides a more appropriate variation of the exponent of Reynolds number between the values appropriate to a laminar boundary layer and a turbulent boundary layer than existing correlations given in the literature.

Acknowledgement

The author is grateful for all the assistance he has received during the course of this work. My thanks are due to the staff at The Nottingham Trent University, especially to Doreen Corlett who has maintained the large collection of papers on jet impingement on which this work is based, to Jim Corlett and the library staff for their helpfulness in obtaining all manner of information and to Iris Foweather who helped with early experiments with an Oracle database. I am also grateful to my fellow researchers Shirley Ashforth-Frost, Shabir Kapasi and Chun Chin Lin for their friendliness and for useful discussions on jet impingement heat transfer and similarly to Stuart Hartle on the subject of knowledge based systems. I also wish to express my appreciation to my supervisors, to 'Nath' Jambunathan, my director of studies, to Eugene Lai, particularly for his useful comments and encouragement after reading the first draft of this thesis, and to Bryan Button who's long-standing interest in jet impingement heat transfer and belief in the potential of expert systems brought about the initiation of this work. Amongst my new colleagues at Rolls-Royce my thanks go particularly to David Campbell and Robin Monico for their constructive criticism of some of the prototypes.

This work was funded by the Polytechnic and Colleges Funding Council under their special research initiative. My thanks are also due to Rolls-Royce plc and to Martin Hamblett who arranged for Rolls-Royce to support this work in its final stages.

Finally to Cathy, who bravely tackled two drafts of this thesis, my thanks which hardly seem enough.

CONTENTS

FIGURES

NOMENCLATURE

a	area (m^2)
af	$\frac{\text{sum nozzle areas}}{\text{area heat transfer surface}}$
ap	area heat transfer surface (m^2)
b	equivalent slot width (m)
Cd	discharge coefficient
cp	specific heat (j kg^{-1} K^{-1})
D	nozzle exit dia.(m)
D_{AB}	diffusion coefficient (m^2/s)
G	mass flow per unit area (kg $m^{-2}s^{-1}$)
Gr	Grashof number
h	heat transfer coefficient (W $m^{-2}K^{-1}$)
l	nozzle length (m)
lx	jet-to-jet spacing parallel to a cross-flow (m)
ly	jet-to-jet spacing perpendicular to a cross-flow (m)
mc	cross-flow (kg/s)
mi	initial cross-flow (kg/s)
n	number of jets
nx	number of jets on the X axis
ny	number of jets on the Y axis
Nu	Nusselt Number (h D/λ)
P	Pressure (Pa)
p	Static pressure (Pa)
Pr	Prandtl Number (cp μ /λ)
Pu	Potential core length (m)
Q	heat flux (W/m^2)
R	gas constant for air
R	recovery factor
r	radius (m)
Re	Reynolds Number (D U_n/ν)

Sc	Schmidt Number ($\mu\ \rho^{-1} D_{AB}^{-1}$)
Sh	Sherwood Number ($k\ D/D_{AB}$)
T	temperature (K)
t	static temperature (K)
U	velocity (m/s)
v	velocity (m/s)
x	distance from stagnation point (m)
y	distance from stagnation point (m)
z	axial distance nozzle to plate (m)
α	impingement angle
η	effectiveness
λ	conductivity air (w $m^{-1}K^{-1}$)
μ	viscosity (N s/m^2)
ν	kinematic viscosity (m^2/s)
ρ	density (kg/m^3)
ϕ	angular position relative to the stagnation point ($^\circ$)
θ	angular displacement ($^\circ$)

suffixes

amb	ambient
aw	adiabatic wall
b	equivalent slot width
c	cross-flow
f	value for a free jet at the corresponding z/D
m	centre line value
n	nozzle exit
r	mean value for a row of jets
rec	recovery
s	stagnation point
w	wall
1	nozzle inlet
2	nozzle exit
1/2	half width

GLOSSARY AND ABBREVIATIONS

AI	Artificial Intelligence
Clause	A single fact or rule asserted into the Prolog database.
DBMS	Database Management System.
EDS	Expert Database System.
KBS	Knowledge Based System - A computer program which applies AI techniques to a narrow domain of knowledge.
KBDS	Knowledge Based Database System.
Predicate	A logical assertion which may be composed of one or more clauses.
Tuple	A relational database record, a Prolog clause has this form.

CHAPTER 1
INTRODUCTION

In the modelling or simulation of many types of systems knowledge of the heat transfer rates are vital. Heat transfer rates are required in predicting the temperatures, thermal stresses and material properties of many engineering components and processes, examples include gas turbine blades, heat exchangers and micro-electronic components. In all but a few simple situations the analytical or numerical prediction of heat transfer rates is not feasible, either because the process is too complex to model analytically or because a numerical solution is prohibitively expensive, and so engineers are forced to rely on empirical data. However there is a vast quantity of empirical information available in the literature, even in a specialised area such as jet impingement heat transfer there are over 1000 references available (Button and Wilcock 1978 and Button and Jambunathan 1989). Whilst the vast quantity of data suggests that there may well be a correlation suited to a particular flow geometry it also gives rise to a problem; an engineer cannot be expected to become familiar with more than a small proportion of this data even in a specialised domain such as jet impingement. Hence a need exists for a system which will make the correlation information easily available.

It will be seen that a conventional database cannot adequately satisfy the requirements for a correlation information system. An alternative approach, that of using symbolic Artificial Intelligence or Expert Systems techniques to create an Expert Database Systems or Knowledge Based Database System (KBDS) was suggested by a review of the literature. The selection of the most appropriate data should be performed entirely by the KBDS based on only a simple skeleton description of the heat transfer problem. The usefulness of such a system would be considerably enhanced if it were to derive the appropriate values using the information selected. This thesis considers the form that a description of a heat transfer problem (in particular that of impinging air jets) should take, methods to assist in the formulation of this description and the structure of a database to hold the necessary information. The design and implementation of a system which will support the retrieval and evaluation of empirical correlations from the database and also enable new correlations for a wide variety of heat transfer flow geometries to be added to the database are then detailed.

1.1 The limitations of conventional databases in obtaining correlation information

Heat transfer data for design or analysis is naturally dispersed over a vast number of sources. As has already been mentioned the subject of jet impingement heat transfer alone has generated more than 1000 English language publications. Over recent years the abstracts and publication details from these papers have been entered into a conventional computer database at The Nottingham Trent University. The database contains the title, abstract, authors' names, journal name and year of publication for each of the documents. A system for selecting documents from this collection was also developed at Nottingham Trent University by Dr C.C. Lin. A search of any combination of the fields listed above can be performed, the system searches the selected fields for the keywords (character strings) specified by the user. Document selections resulting from Boolean combinations of these keywords can be obtained. Provided the user is familiar with the process of jet impingement heat transfer and the terminology used to describe key aspects of impinging jet systems the database makes possible the rapid acquisition of documents relevant to a particular problem. The system was found to be useful for research, enabling a collection of potentially useful documents to be rapidly acquired; though the system only retrieved a proportion of the total number of the useful documents in the database. The cause of this problem being the difficulty in specifying combinations of keywords which are sufficiently general to select the relevant documents but sufficiently specific not to select a large number of irrelevant documents. Typically only a small proportion of the documents selected would contain heat transfer correlations, which it is expected would be a serious limitation to the systems utility in a design environment.

A selection process which is based on the presence of specified words in the title and abstract suffers from a number of limitations. There is a tendency for authors to address the abstract to contemporary research issues rather than describe the data presented, as a consequence publications containing relevant data may not be selected. In addition the range of conditions over which data is valid is generally not described in the abstract and so papers which are not relevant may also be selected. Finally the database does not help in the extraction and application of relevant data from these documents so that once the selection has been made the engineer must still distinguish between appropriate and inappropriate data, extract the data and derive values for heat transfer rates. The latter process may be time consuming because of errors in the data presented, a lack of uniformity in data presentation and the distribution of the relevant data amongst a number of different papers.

Even databases where the entire text is available, usually referred to as full text databases, are not satisfactory. Recall rates (the proportion of the relevant documents in the database actually retrieved) of only 51 per cent on average were found by Fenichel (1981) for searches based on boolean combinations of words. A full text database also does not address the problem of extracting the relevant data from the text and applying this data.

1.2 Desirable capabilities of a correlation information system

The aim for a correlation information system must be a method for retrieving (and preferably evaluating) all the relevant correlations from the database. The selected correlations should be confined to those that are relevant. It should be possible to obtain these correlations with only a minimal prior knowledge of the specialised terminology used to describe different jet impingement heat transfer processes.

New information on all types of heat transfer, including jet impingement heat transfer, is regularly published. If the information system is not to become rapidly out-of-date it must be made possible to add new data without the need to rewrite the system. This is a capability more usually associated with databases than with knowledge based systems.

It is likely that for a significant number of inquiries to the system no information which strictly applies will be present. Under these circumstances it is desirable that the search of the database should be modified to retrieve and evaluate correlations which are expected to give the best estimate of the heat transfer. This can only be achieved through a knowledge of the heat transfer process. An understanding of how and to what degree changes in geometry and fluid flow rates affect the heat transfer coefficients is required. This information should be represented in a knowledge base and be used by the KBDS in automatically modifying the database search when required.

The knowledge based database system created has used jet impingement heat transfer as an example of a fairly complex heat transfer process. However the methods employed in the KBDS have been designed to be general so that it should be possible to encompass other types of heat transfer data and information from other domains, such as fluid flow pressure losses, in a system of the same design.

1.3 An overview of the thesis contents

Chapter 2

An overview of AI and expert systems research with an emphasis on engineering applications is given. Work on expert database systems is reviewed; this review considers the different architectures that have been proposed but concentrates particularly on work aimed at developing methods to provide assistance in retrieving information.

Chapter 3

Published work on single and multiple air jet impingement heat transfer is reviewed. The information obtained from this review is essential in the construction of the KBDS. In addition to the correlations incorporated into the database the review also provides the information used by the KBDS to control the search of the correlation database.

Chapter 4

Based on the review of the jet impingement literature an outline of the required content of the database is presented. The need for both a qualitative and a quantitative description of a heat transfer geometry and flow regime to enable a sufficiently precise selection from a database is established. An outline is given of how knowledge of jet impingement heat transfer may be used to generalise this description and so improve the results of the search.

Chapter 5

The development of the knowledge based database for jet impingement correlations is described. The correlation database consists of a program to assist in the selection of correlations from the database and a program to add new correlations to the database. The knowledge based methods used and developed during the course of implementing these programs are detailed.

Chapter 6

The suitability of code written in a procedural language, rather than in the declarative language Prolog, for some of the processes performed by the KBDS is identified. The justification for the choice of interface between the KBDS and the database files is set out. The consequences of the decision to store the correlation statements in relational records are given.

Chapter 7

The degree to which the knowledge based system developed improves the availability of correlations is discussed. The symbolic AI and

Expert system techniques used in building the KBS, particularly techniques used to assist in obtaining information from databases are also discussed. The discussion emphasizes the developments of these techniques required in order to assist in obtaining heat transfer coefficients from a database. The additional functions which the KBDS must perform in order to extend its capabilities from the addition of new information to the alteration of the existing information are also discussed.

Chapter 8

It is concluded that the feasibility of a knowledge based database system which satisfies the original aims of providing access to the most appropriate correlations based on a simple description of a heat transfer process has been demonstrated. A statement of the knowledge based techniques required in implementing the jet impingement database system is made; it is concluded that a number of developments of existing techniques were required in creating the knowledge based database system.

Chapter 9

Four potential lines for further investigation are considered. These are:

(i) The design of the interface for selection of correlation data from the database.

(ii) The provision of assistance in the acquisition of new rules for the knowledge base which automatically modifies the database search.

(iii) The automatic creation of an optimal design for a system of impinging jets.

(iv) The use of neural networks to correlate heat transfer coefficient distributions which it is not feasible to represent explicitly.

CHAPTER 2

A REVIEW OF RELATED ARTIFICIAL INTELLIGENCE, EXPERT SYSTEM AND KNOWLEDGE BASED SYSTEM APPLICATIONS AND METHODOLOGY

2.1 Applications to Engineering Design and Analysis

Many types of software techniques have been applied to aid in the design and analysis of engineering systems. These include databases (of material properties for example), mathematical modeling (eg finite element systems for stress analysis) and drawing systems. Expert System or Knowledge Based System (KBS) techniques are being applied with the aim of automating decision making in the design or analysis process or to tackle problems which hitherto were unsuitable for existing techniques because of the nature of the information, however there is no clear distinction between conventional and KBS software.

The early work on Artificial Intelligence suggested that useful applications would only be achievable if the system were confined to a limited domain (Winston 1992). Examples of these applications include a system for the diagnosis of diseases of the blood (Davis, Buchanan and Shortliffe 1977). A second application was designed to match geological data from a location to the best of a number of models and then proceeded to predict the mineral deposits to be found in that location; this application successfully predicted the existence of a previously undiscovered mineral deposit (Duda and Reboh 1984). The knowledge in both these domains is factual, and the authors of these systems were able to represent the information in the form of If-Then rules. In contrast the solution to an engineering problem generally requires the combination of factual and procedural information, the procedural information commonly taking the form of a mathematical analysis. For example heat transfer by conduction in a solid requires the derivation and solution of the relevant partial differential equations but the selection of a suitable mathematical model could be represented by factual rules.

In the field of engineering, systems have been built to perform diagnosis, process control and design tasks including analysis of behaviour, automatic design and design support. Analysis systems predict the behaviour of a system where the configuration is known, design systems generate and optimise design configurations. Systems of these types tend to be confined to a specialised domain. An alternative approach is to allow a human designer to direct the design process, the software acts as a design assistant relieving the designer of tasks where possible. These systems are often referred to as design support systems. The inclusion of a designer allows creative design and enables the system to be used on a wider range of problems.

Analysis

Stallman and Sussman (1977) described a system for electronic circuit analysis. The system works by applying basic rules about components to a database of facts about the circuit. These rules are applied in a forward chaining manner, a rule is fired whenever the facts in the database satisfy the rule conditions, the conclusions to the rule are further facts about the circuit which are then added to the database along with the conditions on which the conclusion depends. The system is able to make assumptions when needed, if a later rule indicates that the assumption was incorrect then the system removes only those conclusions which depend on the assumption rather than all conclusions inferred since the assumption was made. This technique is referred to as 'Dependency-directed Backtracking' and has the effect of reducing the search space.

The automatic use of physical laws was also described by Brown (1988), in this case to perform an analysis of the dynamic behaviour of a mechanical system.

Adeli and Hung (1990) outlined a system for making available data on the effect of earthquakes on buildings. A large quantity of observed data is stored, either as production rules or in a relational database. The use of a relational database rather than rules simplifies the addition of further earthquake data. Graphics are used as visual aids to help explain the contents of the knowledge base.

Autonomous design

The design process has been divided into categories by numerous authors, these range from creative design to routine design. In routine design both the design process and the form or forms of possible solutions are known. In creative design or invention only the required functions of the final product are known. A number of strategies for autonomous design have been suggested or implemented.

Starting from a list of specified major components 'R1' (McDermott 1982) derives a valid configuration for a computer system. This program advances a step at a time towards a completed configuration by adding components to the current partially completed configuration. This method is often referred to as the propagation of constraints because at each step the current partially completed configuration imposes restrictions on the addition of the next component. Provided these constraints are satisfied in selecting and adding a component then the newly developed configuration will be a valid step towards a completed configuration. This strategy is carried out by forward chaining rules; backtracking is not required to produce an acceptable solution and no optimisation of the design is attempted.

A system for the selection of roller bearings (Sim and Chan 1991) bases an initial selection of bearing type (eg thrust or radial) on the functional requirements (eg whether the loading is axial, radial or a combination of axial and radial). This selection is performed using a knowledge base of empirical rules. Once a type of bearing has been identified a procedure is initiated to calculate values for the various parameters on which the selection of a specific bearing of the identified type is based. The particular bearing is selected from a database on which the technical details from manufacturers' catalogues have been captured.

A contrasting strategy is a variation on 'generate and test'. Maher and Fenvers (1985) described a system, Hi-Rise, for the preliminary design of buildings. A selection from a finite number of design configurations is made, the selection must satisfy imposed constraints (for example it must perform the intended function and satisfy the building regulations). While all the possible configurations or forms of the design solution are known at the start of the process not enough is known about this solution to make a specific selection. The authors identify three stages in the design process: Synthesis, Analysis and Evaluation. During synthesis component types which are appropriate to the form of the design being considered and which satisfy constraints are selected. Analysis is performed to evaluate loads and to size components to provide the required load capacity, the result is a feasible design. An evaluation is then performed which assigns a score to the design. The system then repeats the process for other design configurations. A selection of design configuration can then be made either automatically, based on the score, or by the user.

A system to design cam-followers (Wang and Lin 1989) employed a strategy similar to generate and test. A combination of rules, used to estimate appropriate parameter combinations, and algorithms, used for precise calculations and optimisation of components, enabled a design to be build-up. In formulating a design it is not always possible to know precisely all the constraints affecting the choice of a design parameter at the point when a value must be specified; under these circumstances an assumption must be made. If a partially complete design is found to violate a constraint then the system will backtrack and attempt to construct a feasible design by selecting a new combination of parameters. Once the feasible designs have been constructed a number of rules are followed by the system which enable it to select the optimum from the feasible designs.

It is not always possible to specify rules which definitively determine the design, nor is it always feasible to use backtracking to examine all the possible combinations. Under these circumstances rules to propose revisions to the design when a constraint is violated are required. Marcus *et al* (1988) describe a KBS developed to perform the design of elevators. In this description the problems of constructing networks of rules to propose design parameter values and to revise partially completed designs is considered.

The use of Knowledge Based methods to perform design optimisation requires explicit knowledge of how the design variables should be changed to improve the design at every point in the design space. Numerical optimisation methods and Genetic Algorithms do not require this explicit knowledge.

Numerical methods for optimisation of designs are well established, (see for example Fox 1971), these methods are appropriate if the design variables can be related mathematically or numerically to yield a continuous function, the evaluation function. The optimal values for the design variables are found by iteratively choosing values for the design variables and deriving the corresponding evaluation function. Conventionally the optimum design gives the minimum value for the evaluation function. The application of these methods can involve a number of problems, these include the finding of a local optimum rather than the global optimum, applying constraints to the design variables and the number of iterations required to arrive at a solution. The number of iterations required is particularly significant where the derivation of the evaluation function requires the running of a numerical model.

Genetic Algorithms (GA's) provide a more reliable means of finding the global optimum when a number of local optima exist. However these methods generally require a larger number of derivations of the evaluation function to find a solution. Powell *et al* (1991) describe a hybrid technique using Knowledge Based Systems, numerical optimisation and GA's which, by using the most appropriate of these techniques at each stage in the design optimisation, either avoids or minimises the drawbacks of each technique when used separately. The authors give as an example the aerodynamic design of a two stage turbine for General Electric. The hybrid optimisation achieved a higher predicted efficiency than was achieved by manual optimisation.

Design Support

An early design support system, 'Sketchpad', is described by Sutherland (1963). In addition to producing drawings this system was able to model the movements in systems of mechanical linkages. This was achieved by maintaining the topology or connectivity of each part in a drawing, by specifying geometric constraints (eg linking points on different parts) and by satisfying these constraints as parts were moved. In addition Sketchpad was able to maintain the connections between the components in a circuit diagram as the components were moved and to perform a deflection and force analysis of frame structures.

Some proposed systems require interaction with the user, as do the analysis systems, but can still be viewed as design systems. A system for electronic design (Ketonen *et al* 1988) divides the design to make it manageable. Each sub-task is carried out either using heuristics or by numerical optimisation depending on the particular task. User intervention is required to ensure correct interaction between sub-tasks.

A later report (Popplestone *et al* 1986) set out a number of features required in an ambitious 'design support system'. A key feature was an 'encyclopaedia of engineering knowledge'. This encyclopaedia should include heuristics to guide the design process and a taxonomic database of design information, the database should be organised into classes (eg a spur gear is in the class gear). It was anticipated that multiple inheritance would be required as a component may fit into, and thus obtain values from, more than one class.

Over recent [illegible] progress has been made towards a 'design support system'. A theoretical model of the design process has been developed by Smithers *et al* (1992). In a simplified description of this model an initial design requirement, which may be incomplete or inconsistent, is progressively refined to give a complete definition of the design. This process may involve creating potential designs, evaluating these designs and then modifying these designs. The exploration of the possible designs is directed by the designer but assisted by the system which provides specific knowledge and methods. A system embodying these ideas has been developed by Smithers *et al* to assist in the molecular design of drugs. The system required the representation of two types of engineering design knowledge: the first requirement was to store and present information which may be useful to the designer (for example data from text books and example designs), the second requirement was for knowledge of procedures or methods to be followed to achieve elements of the overall task. The system must be capable of identifying when these tasks can be performed, what assumptions are required and of performing the tasks.

A different emphasis is given by Clarke and MacRandal (1991) who described a system which co-ordinates a number of separate programs in performing a single design task, in this case building design. A conflict exists between the requirement for programs to be powerful and comprehensive whilst also being simple and intuitive to use. The authors propose a solution to the problem, an 'intelligent front end', where the interface presented to the user is specific to the users technical objectives and experience. The system uses terminology, offers default values and suggests methods appropriate to the user's objectives, in addition the system provides help and guidance appropriate to the designers experience.

In the view of Fischer and Nakakoji (1992) the provision of information to a designer is crucial. As part of a design support system Fischer and Nakakoji report the implementation of a hypermedia system which organises and displays figures and text. This system displays entries from a catalogue of illustrative designs. To assist the designer the system only selects designs from this catalogue which are feasible modifications or additions to the current design status, ie only those entries that are relevant.

2.2 Expert Database Systems

Expert Database Systems (EDS) were described by Kerschberg (1990) as database management systems (DBMS) which use knowledge or expertise to direct the provision of data from the database. Knowledge about the data may be used to improve the database performance (for example in helping the user to form a query and in query optimisation). Alteratively the data may be interpreted by a knowledge based system to provide an 'intelligent' answer to a query.

The literature provides a number of alternative architectures for EDS, these include:

(i) A KBS loosely-coupled with a conventional database system.

(ii) A DBMS enhanced with a query language which supports inference.

(iii) Enhancements of KBS to incorporate database access and maintenance functions.

(iv) A fully integrated data and knowledge management system.

In the first of these architectures the two processes are separate and communicate through an interface such as SQL. The remaining architectures enable knowledge about the structure of one component to be available to the other, these systems are referred to as tightly-coupled.

The use of the relational data model is well established and its use in EDS is being investigated. An alternative model, the object oriented database, is a more recent development. The object oriented database can provide a more flexible description of structures made up of many components, a computer system for example (Chen and Tseng 1989). A representation of the design can be built up into a hierarchy of objects where each object in the hierarchy forms a part of its parent. Each object may in turn be separated into its components, each of these components can be represented as child objects in the hierarchy. Databases of this type have been proposed for representing information in autonomous design systems. Preiss (1980) describes the concept of a design system in which the design is represented by a hierarchy of frames or objects. The design is built-up by providing values for the slots which describe different aspects of these objects, the design is complete once all these slots have been given values.

Expert Database System Architectures

Loose-coupling between the knowledge based program and the database has the advantage of allowing the use of existing Expert system shells and DBMS. This is a common approach, examples include a system described by Van der Spec (1990) which advises Do-It-Yourself customers on the selection of paints. Based on the customers description of an intended job the KBS decides on the required paint characteristics. The external database, which holds details of paint characteristics, container sizes and prices is then searched and the customer is advised of a suitable purchase. The roller bearing selection system of Sim and Chen (1991) also uses a database to contain the details of the alternative solutions, in this case the properties of specific bearings.

A prototype interface resulting in loose-coupling between multiple KBS's and DBMS's was described by Howard and Rehak (1989). This system decides to which database or databases a query should be addressed and translates the query into a suitable form. The system contains knowledge of each of the databases to enable it to perform the query translation and to perform some inference to provide an 'intelligent' answer to the query. An example quoted was for finding a structural component, a beam, with the minimum cross-sectional area which also satisfied a given set of constraints which were expressed as rules.

A number of proposals have been made which extend the capabilities of existing DBMS to inference. An example is POSTGRES (Stonebraker and Rowe 1986), this is an extension to the relational DBMS Ingres which will include forward and backward chaining rules and virtual relations or fields where the values are inferred at the time of an appropriate query.

Enhancements which give KBS a database capability have been proposed. Delcambre (1988) suggested a production rule language which provides access to a relational database. Production rules would be in the form

if < conditions > then < conclusions >,

the conditions could involve queries to a database and the conclusions would be able to assign values to fields in records.

KBS are commonly written in the logic programming language Prolog. Extensions to Prolog and other logic programming languages to include database management functions have been proposed. Systems which use logic for managing databases are often referred to as deductive databases. The connection between the relational database model and logic programming has been investigated. Lloyd (1983) showed how logic programming can be used to express queries and views and to derive new 'virtual' or 'implicit' relations from existing 'explicit' relations using rules. Logic is a single uniform language which can be used to express data definitions, queries and complex integrity constraints, and in addition to define programs. 'Indeed, when both data bases and programs are formulated in logic the distinction between them disappears' (Kowalski 1988). These benefits were recognised by Brodie and Jarke (1986) in a discussion of Prolog as a database language. Amongst the limitations of logic as a knowledge representation they included:

(i) The lack of commonly accepted DBMS facilities such as efficient access of large databases, support for multiple users, concurrency control, query optimisation, data recovery and security.

(ii) The representation of procedural knowledge.

(iii) On backtracking from a failed attempt to prove a goal Prolog does not retract any asserted clauses. If these clauses were asserted into a database this could cause unwanted or incorrect data to be added. The usual DBMS practice is to cache updates and to add them to the database only at the end of a successful transaction.

SQUIRREL was proposed as a language for a deductive database system based on Prolog by Waugh *et al* (1990). This language is an extension of SQL (a standard language for DBMS), the extended language includes rules and allows virtual relations or fields to be inferred. Attention is given to making the access of data by the application program independent of the format ot the data in the database.

The implementation of efficient large scale deductive databases based on Prolog appears to be a problem however. Torsun and Ng (1986) developed a system closely coupling Prolog to a DBMS, this work demonstrated that processing of the repeated searches involved in a recursive query remained a problem (although recursive queries also pose an efficiency problem for existing DBMS).

Lucas (1988) described a KBS developed in a logic programming language which was tightly coupled to functions written to provide access to database files. Tight-coupling to form a single process is desirable because evaluation of a query in a logic program occurs one record at a time, and so access to the database information is made for each record rather than once to retrieve a set of records as in a conventional relational DBMS. This technique provided a useful basis for the development of a system despite the lack of useful database management facilities.

McLean and Weise (1991) described a system which uses an extended relational database language (the relational calculus is a subset of first order logic) to perform deductive reasoning through backward chaining and recursion. The benefits of the facilities offered by DBMS are retained. This contrasts with the use of a logic language such as Prolog as a database front-end or an extension to a logic language to add some database management functions. However the system described by McLean and Weise is non-procedural and so is not effective for writing programs, does not backtrack (this is not required as the search returns a set of all the successful queries rather than return one at a time) and has no arithmetic capabilities.

In a discussion of coupling knowledge base systems and database systems Kerschberg (1990) proposed that the integration of the KBS and the database is required where both the database and the knowledge base may be updated together. This will occur where the KBS has knowledge of the contents of the database for example where the integrity constraints are expressed as rules in the knowledge base. In such a system the addition of new data can only be performed with reference to these constraints.

Enhanced Front-ends for Database Applications

Knowledge of the contents of the database can be used to aid in the specification of indexing keywords for new records (Humphrey 1987), in the front end to aid the user in formulating a query (Smith *et al* 1987 and Parsaye *et al* 1989), and to automatically broaden or narrow a search (Smith *et al* 1989).

Humphrey (1987) described a knowledge based system which assists in the creation of a frame to represent and index an entry to a medical database. The user is prompted for keywords which are entered into pre-defined slots and so represent explicitly values for given attributes, for example the entry ‘club foot’ might be made for the slot ‘disease’. A hierarchy ~~of valid terms~~ is used to prevent some un-acceptable combinations of keywords. The hierarchy is also used to specify that the disease club foot, for example, implies that the value ‘foot’ should be entered for the ‘body part’ slot, and so some slots can be filled automatically by the system. In addition the system will also substitute some keywords with preferred synonyms.

The retrieval of information from text databases has been considered by Parsaye *et al* (1989), who proposed a method which they called ‘concept [illegible]g’. This method makes use of a group of keywords which have already been selected by human experts to describe a document. The keywords are organised into hierarchies (again by human experts) linking each keyword to other keywords which are either more general or more specific. Each document is represented or indexed by a frame, the value of each slot in the frame is filled by a word taken from the corresponding keyword hierarchy. A document is retrieved by specifying the contents of each of these slots.

Smith *et al* (1989) also use a frame in which the slots and associated values describe a document in the database, the possible slot values being taken from a 'tangled' hierarchy. A query is formulated by specifying values for the various slots. A study of the use of a database by students revealed the following methods of changing a query as the most common:

(i) Broadening a query by deleting a slot value.

(ii) Narrowing a query by adding a value to a previously empty slot.

(iii) Changing a slot filler to a more or less general term, respectively broadening and narrowing the query.

(iv) Adding alternative fillers to an existing slot so broadening the query (in effect an OR query).

(v) The addition of synonyms to a slot making the search more thorough.

The system developed automatically tries the available alternatives and informs the user of how these alterations to the query would affect the number of documents selected.

Using these principles Smith *et al* built an expert system specifically to provide access to documents on environmental pollution. A more general expert search intermediary was described by Gauch (1990), in addition to the techniques outlined by Smith *et al* three other techniques were incorporated:

(i) A search will retrieve more documents by searching for the stem of a specified word rather than the entire word. For example if the word 'searching' is specified then a search using its stem, 'search', would retrieve documents containing the words 'search', 'searches', 'searched' etc in addition to the word 'searching'.

(ii) The context in which groups of words must appear can be altered. If a database query which includes a pair of words initially required to appear adjacent to each other is altered to allow the two words to appear anywhere in the same sentence then more documents will be selected.

(iii) A query will retrieve more documents if occurrences of the Boolean operator 'AND' are changed to 'OR'.

Gauch showed a significant improvement in the time taken for users to formulate queries and retrieve data.

An further approach based on knowledge of the database subject matter was given by Lucarella and Morara (1991). Lucarella and Morara suggested that both a query and a description of the content of a document will be imperfect and imprecise. To overcome this problem they proposed the use of fuzzy inference. This method makes use of a knowledge base which consisted of frames representing documents, queries and concepts connected by weighted links. Document summaries were analysed and linked to the appropriate concepts in the network. Queries, expressed in natural language, were also analysed. The most appropriate documents were identified through an examination of the strength of the links between the query and each document.

Arniaiz *et al* (1991) considered the interpretation of the natural language text in technical documents using syntactic and semantic analysis. The system was designed to provide information and sections of text to an expert system. The difficulties of natural language interpretation were eased by the restricted meaning of words in technical documents and the comparative ease of interpretation of the expert system requests.

Database interface methods which enable a reduction in the constraints on the queries that a user is permitted to enter may be expected to offer benefits. Systems employing such methods include those described by Gauch (1990), which allows an unconstrained choice of keywords but requires queries in the form of Boolean expressions, and Lucarella and Morara (1991), which analyses a natural language query. A further benefit is that the addition of new documents to these databases will be simplified, ie the user is not required to associate a description with each new document. This is possible provided that the knowledge base includes the terminology employed in the new document and that all the information relevent to the query is given in the text of the new document.

An example of a system to provide an intelligent answer to a query is provided by Cuppens and Demolombe (1989). The system attempts to recognise why the user presents a given query, if a reason for the query is established then the system will provide information in addition to that explicitly requested. The system performs this service, called ‘co-operative answering’ by its authors, by transforming the users query. The example given is the response to a request for flight departure times over a given short route. Recognising the route in the query the system transforms the query into a request for train departure times in addition to flight departure times. A condition in the transformation rule is that the train journey should not be significantly longer than the flight, which is often true over short journeys.

A system to optimise the search of a deductive database using a heuristic method is described by Yoon and Henschen (1989). The method developed makes use of the rules defining the implicit relations and the integrity constraints. The optimisation is divided into two phases. Prior to processing the query the technique manipulates the rules in order to avoid repeat searches of the same relations. The searches are also ordered in an attempt to minimise backtracking . To speed query processing the number of rules forming integrity constraints is minimised by combining constraints where possible and choosing the most restrictive integrity constraints.

Linking hypertext to databases

The use of hypertext to enhance the user interface to a database has been proposed (Parsaye *et al* 1989). Words and fragments of text can be associated with each other by links. A network can be built up from these links and the nodes formed by the text, this structure forms the basis of a hypertext document. By choosing and following links from one related piece of text to another a user has the freedom to browse a document. The user is able to select those parts of the document which interest him and is not constrained to follow a pre-defined linear path as in a printed document.

A history of the development of hypertext systems, both academic and commercial, was given by Parsaye *et al*. They then considered the possible development of hypertext to aid in the accessing of databases. The most straightforward application is where the terms which can be used in a query would form the nodes in a hypertext document. The hierarchy of terms is followed to reach the node corresponding to the desired record. If a Boolean query is required where more than one term must be associated with the selected record (eg Term 1 AND Term 2 AND Term 3) this method will not suffice. Parsaye *et al* suggested that this type of query might be formed by selecting terms during the course of browsing the hypertext, these terms could be collected to form a query. It was suggested that hypertext could be used to link terms to definitions, synonyms and to related terms.

A full text scientific database was described by Marshall (1991) which used links to associate items of data. In this application the links are followed from one piece of information to another as in hypertext but the links are dynamically controlled, the active links depend on such criteria as who the user is and what data he has already retrieved. For example in response to the term 'ram' an aerodynamicist would be provided with information on engine intakes and a computer scientist with information on computer memory. In addition links are not only made to the words in a piece of text but also depend on the meaning of the text.

A hypertext system for the management of electronic circuit designs was described by Brauer and Stuchly (1991). The Electronic Design Interchange Format (EDIF) is used to describe circuits; complex circuits can be built up from more simple circuits forming a hierarchy of data. For example an XOR gate is built up from interconnected AND, OR and NOT gates. The hypertext system uses the EDIF description of a design, considering the components as nodes and the relations used in the hierarchy and the interconnections as links. Uses of the system include browsing a design and the input and manipulation of design data.

Chapter 3
Jet Impingement Heat Transfer

The desire to provide easy access to the information in the heat transfer literature provides the motivation for work on a KBDS. The design of the KBDS is based on information obtained from a detailed review of the literature. This review was undertaken to establish the significant geometric and flow features which should be considered when selecting a correlation, this information is fundamental in providing a front end to the database which will assist an engineer in obtaining suitable correlations. In addition the review was required to identify the forms in which the correlation information is presented, the correlations must be represented in the database and suitable evaluation methods must be developed. The review, which was aimed at developing a summary of the current understanding of the physical processes affecting heat transfer in impinging jets, was necessary because significant results contributing to the understanding of the heat transfer processes involved in impingement have been published since the most recent detailed review (Yeh and Stepka 1984).

Jet impingement heat transfer has many industrial applications where high rates of heat transfer are required; applications include the cooling of gas turbine blades, protection of aircraft from the build-up of ice by heating external surfaces, cooling of computer components and the heat treatment of metals. Jet impingement heat transfer occurs when one or more fluid jets are directed onto a surface at a temperature different to that of the jet .

The heat transfer coefficients derived by the KBDS are confined to heat transfer involving air, though these coefficients may be obtained from correlations derived from experiments performed using heat transfer in other fluids or from mass transfer. Because the process of mass transfer is analogous to that of heat transfer, heat transfer coefficients can be derived from mass transfer coefficients. As a consequence much of the experimental work on understanding jet impingement heat transfer has been performed using mass transfer. For the sake of brevity mass transfer is generally not mentioned explicitly in the following text even though the database contains examples of mass transfer measurements.

3.1 Presentation of the Data

The literature on jet impingement addresses four main issues:

(i) Empirical measurements of heat transfer.

(ii) Other empirical measurements aimed at improving the understanding of the flow and heat transfer processes.

(iii) Numerical and analytical solutions of these processes.

(iv) Applications of jet impingement heat transfer.

Individual publications may consider one or more of these issues.

The value of heat transfer due to jet impingement is generally obtained, if desired for the design of an artifact, by consulting empirical correlations published in the literature. The publications giving empirical heat transfer data typically give experimental details such as the number and geometry of the jet nozzles, the geometry of the heat transfer surface, relevant flow rates and details of the heat or mass transfer measurement techniques employed. The publications also summarise the results of the heat or mass transfer measurements and any other related measurements. Many methods of presenting these results have been used, experimental data has been presented as heat transfer coefficients, as Nusselt numbers, as mass transfer coefficients and as Sherwood numbers; these parameters may be given as distributions of a local value or as an area mean value, they may also be given graphically or as equations or in the form of tables or in any combination of these. An appreciation of the jet impingement flow and heat transfer processes is needed in selecting and applying a correlation from the literature.

3.2 Impingement of a Single Circular Jet

A review of the literature, Jambunathan *et al* (1992), for the simplest configuration, that of a single circular jet impinging orthogonally on a flat plate (Figure 1) is given in Appendix C. The determination of the heat transfer coefficient was seen to be complex; the most significant parameters being Nusselt number (Nu), Prandtl number (Pr), Reynolds number (Re), the non-dimensional nozzle-to-plate

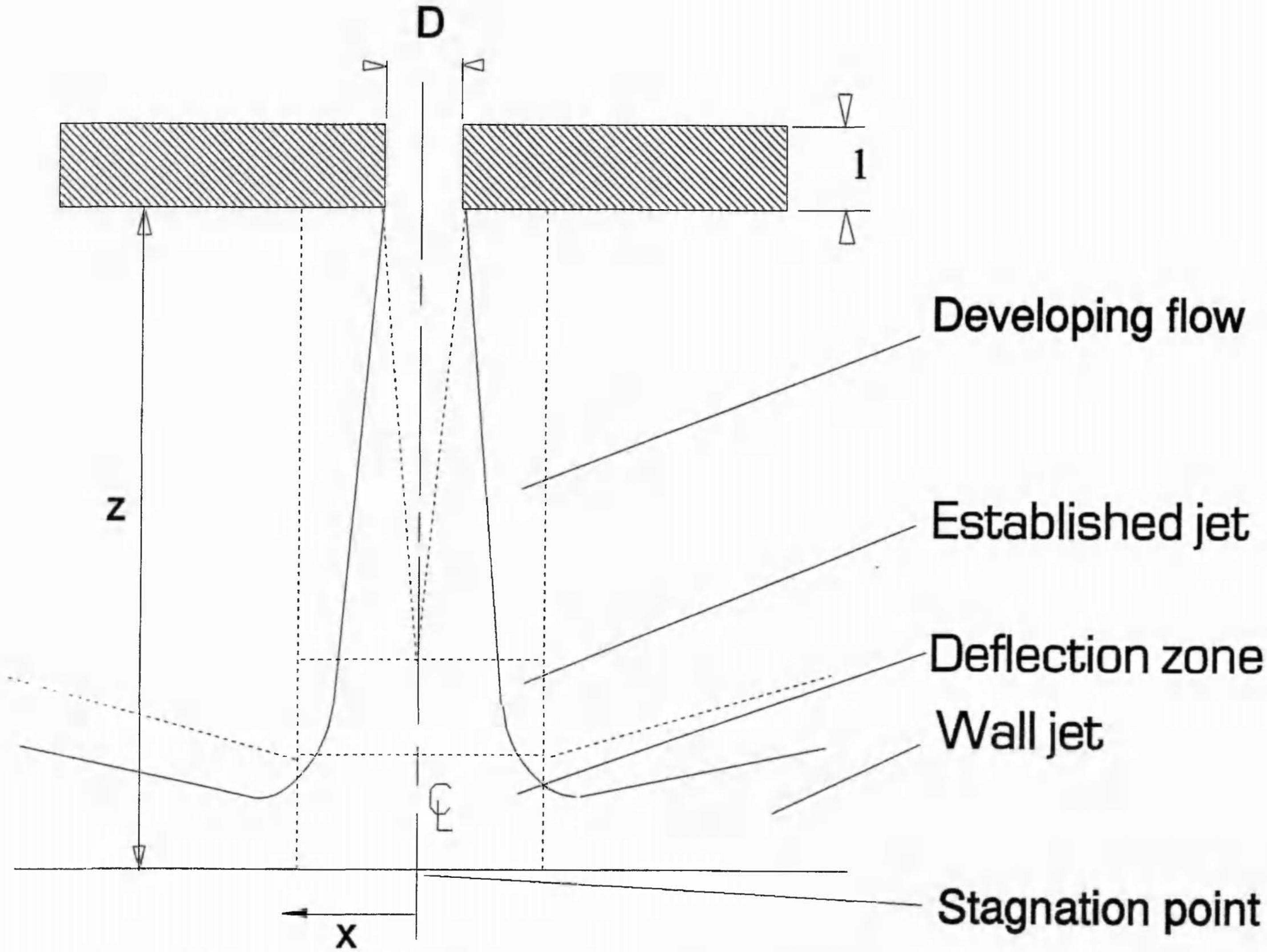


Figure 1. A single circular jet impinging on a flat plate.

spacing (z/D) and the non-dimensional displacement from the stagnation point (x/D):

$$Nu = f(Re, Pr, z/D, x/D)$$

Results have also been published demonstrating that the effects of nozzle geometry and the extent to which the air is confined after impingement are also significant. For example Obot *et al* (1979) performed a number of experimental studies which compared the heat transfer coefficients obtained from jets issuing from nozzles of different geometry. At present there is insufficient data to quantitatively predict the effects of these less significant parameters with a high degree of confidence.

Experimental results are usually presented as plots of heat transfer coefficient (often non-dimensionalised) versus displacement from the stagnation point (x/D), each plot applying to a particular value of the Reynolds number and of the nozzle-to-plate spacing (z/D), see Gardon and Carbonpue (1962) for example. Experimental results are also summarised as correlations applicable over a narrow range of parameters. The majority of correlations yield only area mean heat transfer coefficients; Goldstein and Franchett (1988) present one of the few correlations available for local Nusselt number

$$\frac{Nu}{Re^{0.7}} = A \; e^{-(B + C\cos\phi)(r/D)^{0.75}}$$

where r and ϕ are the polar coordinates of a point on the heat transfer surface relative to the stagnation point, $A = f_1(\alpha, z/D)$, B =

$f_2(\alpha)$ and $C = f_3(\alpha)$. This correlation is applicable to $10000 \leq Re \leq 30000$. Different values for the correlation coefficients A, B and C are given to cover three values of the nozzle-to-plate spacing, (z/D equal to 4, 6 and 10) and four values of the impingement angle (α equal to 90°, 60°, 45° and 30°).

An investigation into providing a single correlation covering a wide range of parameters was made, this work is detailed in Appendix C, Jambunathan *et al* (1992). Based on existing empirical data a set of graphical correlations were obtained for the local heat transfer due to a single impinging jet in the form:

$$\frac{Nu}{Re^{f(x/D,z/D)}} = f_1(x/D, z/D)$$

where

$$f(x/D,z/D) = 0.82 - \frac{0.32}{\left(1 + A\left(\frac{x}{D}\right)^k + B\left(\frac{x}{D}\right)^l\right)\left(1 + C\left(\frac{z}{D}\right)^m + D\left(\frac{z}{D}\right)^n\right)}$$

and A = -1.95, B = 2.23, C = -0.21, D = 0.21, k = 1.8, l = 2, m = 1.25 and n = 1.5. The correlation coefficient, $f_1(x/D,z/D)$, is presented graphically as Figure 2.

This correlation method can be used to calculate the effect of changes in the nozzle exit Reynolds number on the local heat transfer coefficient. This relationship was shown to provide a reasonable agreement with experimental data over a wide range of Reynolds numbers ($5000 \leq Re_n \leq 124\,000$). However only a few experiments which yielded data over a sufficiently wide range of Reynolds numbers were available

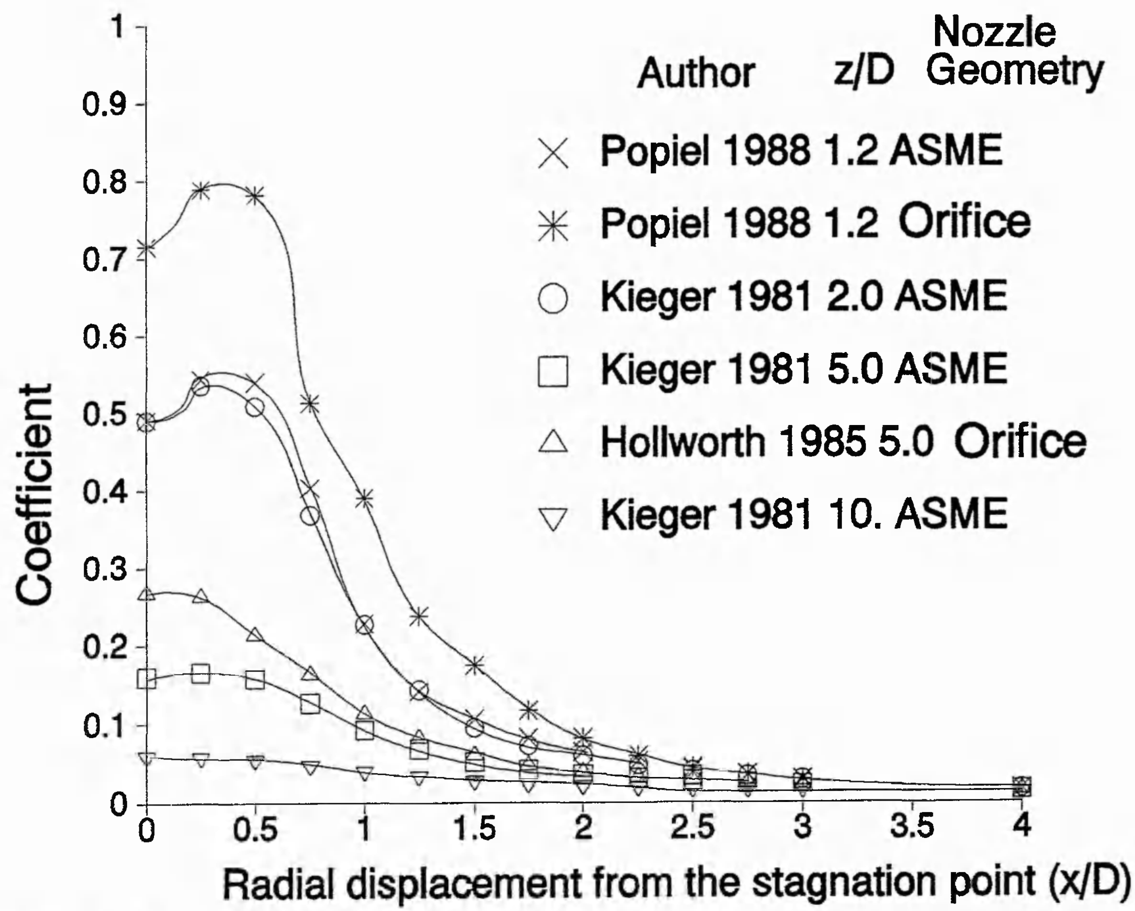


Figure 2. Heat transfer due to a single circular jet impinging on a flat plate

and so the satisfactory behaviour of the correlation has only been demonstrated at nozzle-to-plate spacings of 1.2, 5 and 10 nozzle diameters. Significant differences in the nozzle geometry, in the constraints on the flow in the impingement region and in experimental technique exist between the experiments performed by different researchers. These differences have a significant effect on the heat transfer coefficients and so prevent the techniques used in Appendix C being used to confirm the applicability of the method at other nozzle-to-plate spacings.

To derive the heat transfer rate where the ambient and jet temperatures are not equal, the adiabatic wall temperature on the heat transfer surface must be evaluated and the total heat transfer calculated from the integral of

$$q = \frac{Nu\lambda}{D}(T_w - T_{aw}).$$

Goldstein and Behbahani (1982) defined a parameter, effectiveness, which can be used to derive the adiabatic wall temperature under a non-ambient jet.

$$\text{Effectiveness } \eta = (T_{aw} - T_{rec})/(T_n - T_{amb})$$

where

$$T_{rec} = t_n + R(T_n - t_n),$$

$$R = (T_{aw} - t_n)/(U_n^2/2Cp),$$

T_n and t_n are the total and static temperatures at the nozzle exit, T_{rec} is the local adiabatic wall temperature measured when the jet is ambient ($T_n = T_{amb}$) and R is the recovery factor. The effectiveness reduces from a maximum at the stagnation point with radial displacement from this point. Because of the radial variation in effectiveness the adiabatic wall temperature is also non-uniform and so the local values of heat transfer coefficient, effectiveness and recovery factor are required in-order to calculate the total heat transferred. The jet effectiveness and recovery factor can be obtained using the experimental data presented in Butler (1984), Hollworth and Wilson (1984) and Goldstein *et al* (1990). No theoretical model capable of calculating the local distribution of heat flux due to a non-ambient jet was found in the literature. An approximate empirical model, based on the work of these researchers has been developed, this model derives heat flux from three sets of graphical data:

Nusselt number $Nu = f(Re, Pr, x/D, z/D)$,

Jet effectiveness $\eta = f(x/D, z/D)$,

Temperature recovery factor $R = f(x/D, z/D)$.

After interpolation or extrapolation of the Nusselt number, effectiveness and recovery factor to the required radial displacement (x/D) and nozzle-to-plate spacing (z/D) local flux is derived as follows.

$$T_{rec} = T_{amb} + (R - 1)*(T_n - t_n)$$

$$T_{aw} = T_{rec} + \eta(t_n - T_{amb})$$

$$q = Nu\lambda/D(T_w - T_{aw}).$$

The total flux is derived from a simple integration using the trapezium rule. This empirical model is used in the KBDS to derive heat transfer coefficients and fluxes for single impinging jets. The model can be applied to any distribution of local Nusselt number given in the form Nu = f(x/D), which is the most common method of presenting experimental results for single impinging jet heat transfer. Typically each publication will give a number of these distributions, each specific to a particular value for Re and z/D. In general an interpolation is required to obtain a distribution for the Nusselt number at the required Re and z/D; an algorithm to select the most appropriate curves for this interpolation is described in Lai *et al* (1992). This paper is reproduced as Appendix D.

The Effect of Cross-flow

In the presence of a cross-flow the heat transfer rate is given by the function

$$Nu = f(Re_n, Pr, z/D, x/D, G_c/G_n).$$

Cross-flow (a flow parallel to the impingement plate) causes a deflection of the point of maximum heat transfer. Hollworth and Bowley (1975) gave experimental results showing that the jet nozzle-to-impingement plate spacing (z/D) and the cross-flow ratio have the most significant effect on the deflection. They gave a correlation for the deflection as a function of the ratio of cross-flow and momentum flux of the jet $(\rho_c v_c^2)/(\rho_n v_n^2)$ and of z/D.

Flow visualisation (Bouchez and Goldstein 1975) demonstrated that at high cross-flow rates the jet will not impinge on the target. This occurs when the flow ratio $(\rho_n v_n)/(\rho_c v_c)$ is between 3 and 7 at z/D = 6. The flow ratio at which impingement ceases depends on z/D. Sparrow *et al* (1976) discovered that the maximum Nusselt number occurs when $5 \le z/D \le 6$ at low cross-flow rates where $(\rho_n v_n)/(\rho_c v_c) \ge 8$. At higher cross-flow rates the maximum heat transfer occurs at smaller nozzle to plate spacings. Further measurements by Goldstein and Behbahani (1982) revealed that at z/D = 6 a small cross-flow, $(\rho_n v_n)/(\rho_c v_c) \ge 9$, can increase the heat transfer rate, larger rates of cross-flow reduce the impingement heat transfer.

3.3 The Impingement of a Row of Jets

A common application of a single row of impinging jets (Figure 3) has been inside the leading edge of gas turbine blades. For this reason much of the available experimental work concentrates on this configuration. A review by Yeh and Stepka (1984) considered data from seven papers. These early papers revealed some of the most significant parameters in determining the heat transfer coefficients. Chupp *et al* (1969) showed that the area mean Nusselt number, $\overline{Nu}$, for a semi-cylinder is given by

$$\overline{Nu} = f(Re, ly/D, r/D, z/D)$$

and that the mean Nusselt number on a line parallel to the line through the stagnation points is also a function of the displacement

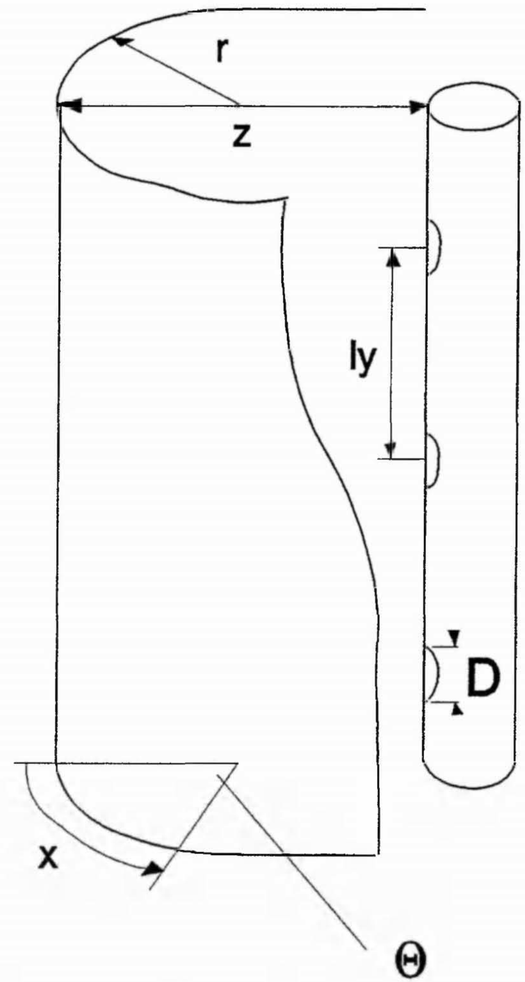


Figure 3. A single row of jets impinging on a concave surface.

from this line. This result was confirmed by Justonis (1970) who obtained the correlation

$$\overline{Nu}_b = 0.03\, f(\theta)\ (z/b)^{-0.4} Re_b^{0.7} Pr^{0.3}$$

where

$$f(\theta) = 0.087*2^{(1+\theta/20)}+0.826,$$

θ is the angular displacement from the line through the stagnation points in the range

$$0^\circ \le \theta \le 20^\circ,$$

and b is the equivalent slot width

$$b = \frac{\pi}{4}\frac{D^2}{ly}.$$

The effect of the nozzle diameter, jet-to-jet spacing and the radius of curvature of the heat transfer surface were not investigated.

Metzger *et al* (1969) and (1972) also used the equivalent slot width, as the characteristic length rather than the nozzle diameter. However measurements taken over a range of values for jet-to-jet spacing and nozzle diameter shows that the use of equivalent slot width does not allow the elimination of a further term for ly/D from the correlation.

The work of Yeh and Stepka shows a considerable variation in the values for heat transfer coefficients obtained in different experiments. Metzger *et al* (1972) identified the effect of the shape of the concave cavity as a significant parameter. The causes of this scatter might also be expected to include those identified for single jet impingement, namely the geometry of the jet nozzle and flow confinement.

An investigation into the effect of the sharpness of the concave cavity was carried out by Bunker and Metzger (1988). Results were presented as contour plots of local Nusselt number

$$Nu = f(x/D, y/D)$$

and as distributions of the line mean Nusselt number versus the displacement along the arc from the apex of the cavity, x (Figure 3). These results confirmed that the sharpness of the cavity ($r^* = x/r$), the nozzle-to-plate spacing (z/D) and the jet-to-jet spacing (ly/D) are significant in determining the heat transfer rate. At a given mass flow and total flow area reducing the jet nozzle diameter and the jet-to-jet spacing by a factor of two resulted in higher stagnation point heat transfer coefficients but a more rapid reduction in coefficients as the distance from the cavity apex was increased. For the range of Reynolds numbers and nozzle-to-plate spacings tested the overall, area mean, heat transfer coefficients were found to be approximately the same for both rows of nozzles (ly/D = 3.33 and 4.67). A slot jet (ly/D = 0) was found to give significantly lower values of heat transfer.

Rows of jets are also used to the rear of the leading edge in gas turbine blades. In this situation the row of jets is likely to be subject to a cross-flow of air produced by rows of jets towards the leading edge of the blade (Figure 4). Investigations by Metzger and Korstad (1972) were aimed at quantifying the effect of cross-flow. The cross-flow was presented as a ratio of the cross-flow mass flow per jet to the jet mass flow (m_c/m_n). As for the single jet, the cross-flow was found to deflect the stagnation point, and so the point of maximum heat transfer, in the downstream direction and to reduce the area mean heat transfer coefficients. Except at low cross-flow rates these effects were found to be greatest at large nozzle-to-plate spacings. The effect of cross-flow is reduced as the jet-to-jet spacing (ly/D) is increased, presumably because for a given total cross-flow the cross-flow velocity is reduced. This suggests that the cross-flow per unit area ($\rho_c U_c$) or the cross-flow velocity (U_c) should be used in non-dimensional groups to quantify the effect of cross-flow.

Measurements of local heat transfer were made by Koopman and Sparrow (1976) using a naphthalene sublimation technique. Their results indicated that the distribution of mass transfer in the region around the stagnation point of each jet was similar to that seen by Gardon and Carbonpue (1962) for a single impinging jet. At large nozzle-to-plate spacings a simple peak was seen at the stagnation point, at nozzle-to-plate spacings of less than four diameters a complex radial distribution around each stagnation point was seen. In addition, at all nozzle-to-plate spacings, Koopman and Sparrow saw a peak in the mass transfer coefficient mid-way between the jets running along the line where the wall jets, produced by the impinging jets,

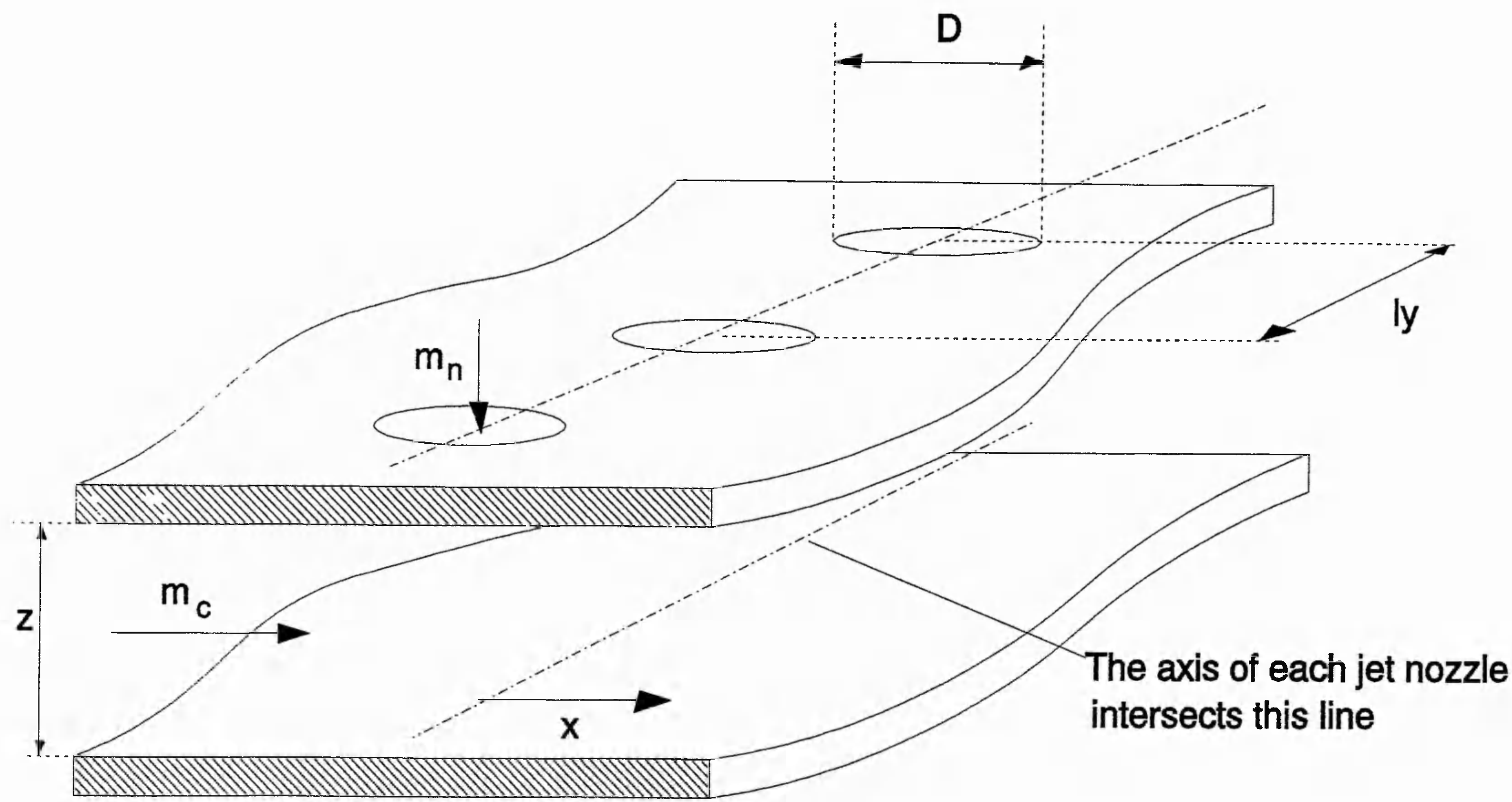


Figure 4. A single row of impinging jets in a cross-flow.

would collide. This peak was reduced by an increase in the jet-to-jet spacing from 4 to 6.67 nozzle diameters.

When the temperature of the ambient air or cross-flow differs from the jet temperature then mixing of the two flows causes the jet temperature to change. An effectiveness, η, and a recovery factor have been used to quantify this effect (Seol 1987) in a similar manner to the single jet. Over the range of Reynolds numbers tested the effectiveness and the recovery factor were found to be dependent on the geometry only. The effectiveness was found to be a maximum at the stagnation point, the effectiveness reducing with increasing nozzle-to-plate spacing.

3.4 Heat Transfer Under Arrays Of Impinging Jets

Measurements of local heat transfer rates on a flat surface under an array of jets (Figure 5) by Gardon and Carbonpue (1962) were presented as contours, these show a maximum heat transfer rate opposite the nozzles. A smaller secondary maximum lies mid-way between the nozzles, Gardon and Corbonpue attribute this second maximum to turbulence generated by the collision of wall jets resulting from the impingement of adjacent jets. A correlation was given based on these results (see Figure 5 for nomenclature).

$$\overline{Nu} = 0.286\ Re^{0.625},$$

where

$$Re = U_{mf} lx\ \rho/\mu.$$

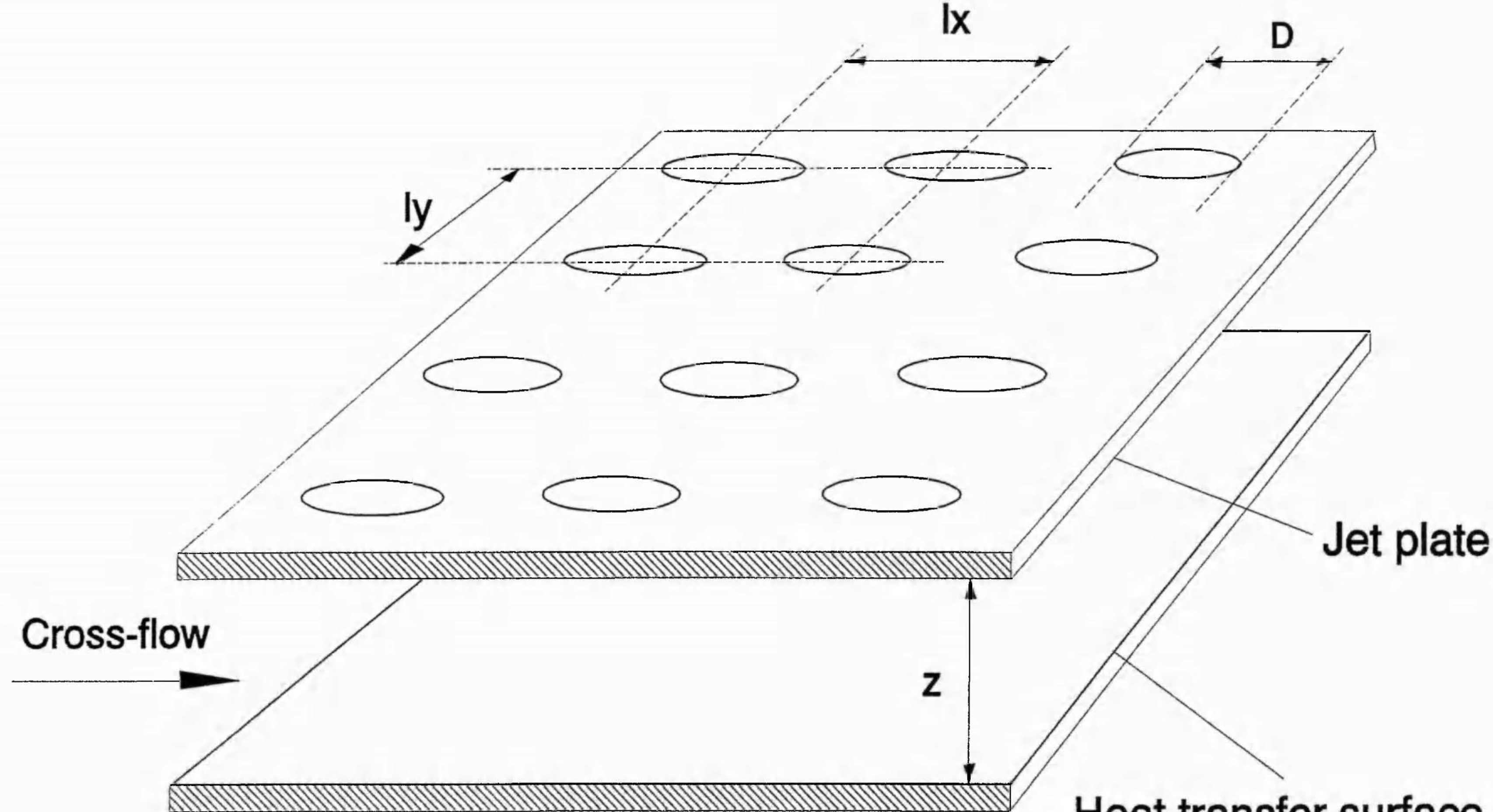


Figure 5. An array of impinging jets.

This indicated that the area mean Nusselt number ($\overline{Nu}$) is dependent on the 'arrival' velocity of the jet, ie the axial velocity of the free jet at the appropriate nozzle-to-plate spacing, and the jet-to-jet spacing.

A dimensional analysis (Kercher and Tabakoff 1970) gave the mean Nusselt number under a symmetrical array to be

$$\overline{Nu} = f(Pr, Re, z/D, lx/D, m_n/m_c).$$

Where the nozzle exit velocity, U_n, is used in the Reynolds number and the nozzle-to-plate spacing (z/D) is introduced in place of the 'arrival' velocity. Measurements of heat transfer demonstrated a change in slope of the curve of $\overline{Nu} \propto Re$ at a Reynolds number of approximately 3000, suggesting a change from laminar to turbulent flow at this Reynolds number. Correlations developed from these results show that the jet-to-jet spacing (lx/D) and the cross-flow to jet flow ratio (m_c/m_n) are important parameters in determining the average Nusselt number. The jet flow was constrained to flow between the orifice plate and the impingement plate, this flow forms a cross flow. The effect of cross-flow was to reduce the area mean Nusselt number.

The effects of nozzle-to-plate spacing and jet-to-jet spacing on the area mean Nusselt number are not independent. Unfortunately measurements taken at various jet-to-jet spacings appear to give conflicting results on the precise nozzle-to-plate spacing at which the maximum heat transfer coefficients are obtained. Hollworth and Berry (1978) gave results for large jet-to-jet spacings ($ly/D \geq 10$).

At ly/D = 10 and 15 these show the maximum $\overline{Nu}$ at a nozzle-to-plate spacing between 2 and 3 diameters. At larger values of jet-to-jet spacing the cross-flow will have a lower velocity for a given jet flow, and so may be expected to have a less significant effect on the heat transfer coefficients. For jet-to-jet spacings of between 15 and 25 nozzle diameters the Nusselt number, $\overline{Nu}$, remains at or near the maximum measured value between a nozzle-to-plate spacing of 5 diameters and the maximum spacing tested (z/D = 25 for ly/D = 25). Whereas Andrews and Hussain (1984) show that as nozzle-to-plate spacing is varied at a constant jet-to-jet spacing of 12.9D maximums in the area mean heat transfer occur at nozzle-to-plate spacings of 0.56D and 2D. The data of Obot and Trabold (1987) shows that the maximum $\overline{Nu}$ occurs at nozzle-to-plate spacings of less than 2D when the jet-to-jet spacing is in the range 4D to 10D. Other data taken at ly/D = 4, Hollworth and Cole (1987), shows that $\overline{Nu}$ is a maximum when $z/D=11$.

The effect of cross-flow, in the case where this flow discharges on one side of the array only, on the heat transfer for an array has been investigated. Colladay (1975), using the measurements of Kercher and Tabakoff (1970), obtained a correlation for the area mean heat transfer under a row of jets in an array

$$\overline{Nu} = \phi_1 \ \phi_2 \ Re^m \ Pr^{1/3} \ (z/D)^{0.091}$$

where ϕ_1 and m are functions of lx/D and

$$\phi_2 = \frac{1}{1 + a((Gc/Gj)(z/D))^b}$$

where a and b are constants. The correlation suggests a reduction in the area mean heat transfer as the ratio of cross-flow to individual jet flow is increased. The heat transfer measurements on which this correlation is based were averaged over a large area, a much higher resolution was obtained by Florschuetz *et al* (1980) and Florschuetz *et al* (1981). Moving from the upstream end of the array in the direction of the cross-flow the mean Nusselt number under each row of jets was seen to initially reduce. At large jet-to-jet spacings ($lx/D \geq 10$ and $ly/D \geq 4$) and small nozzle-to-plate spacings ($z/D = 1$) this Nusselt number was seen to reach a minimum and then increase. This effect was not observed at the larger nozzle-to-plate and smaller jet-to-jet spacings tested. The effect was attributed to the contribution to heat transfer from the cross-flow, which is significant at small nozzle-to-plate spacings, and to the degradation of heat transfer due to interference between the cross-flow and the jets being less significant because of the large jet-to-jet spacing. These results also demonstrated the significance of the cross-flow to jet flow ratio. A correlation was presented based on this data

$$\overline{Nu} = ARe^{m} \{1-B[(z/D)(Gc/Gn)]^{n}\} Pr^{1/3}$$

where A,m,B and n are functions of lx/D, ly/D, z/D and of the hole pattern (staggered or in-line see Figure 6).

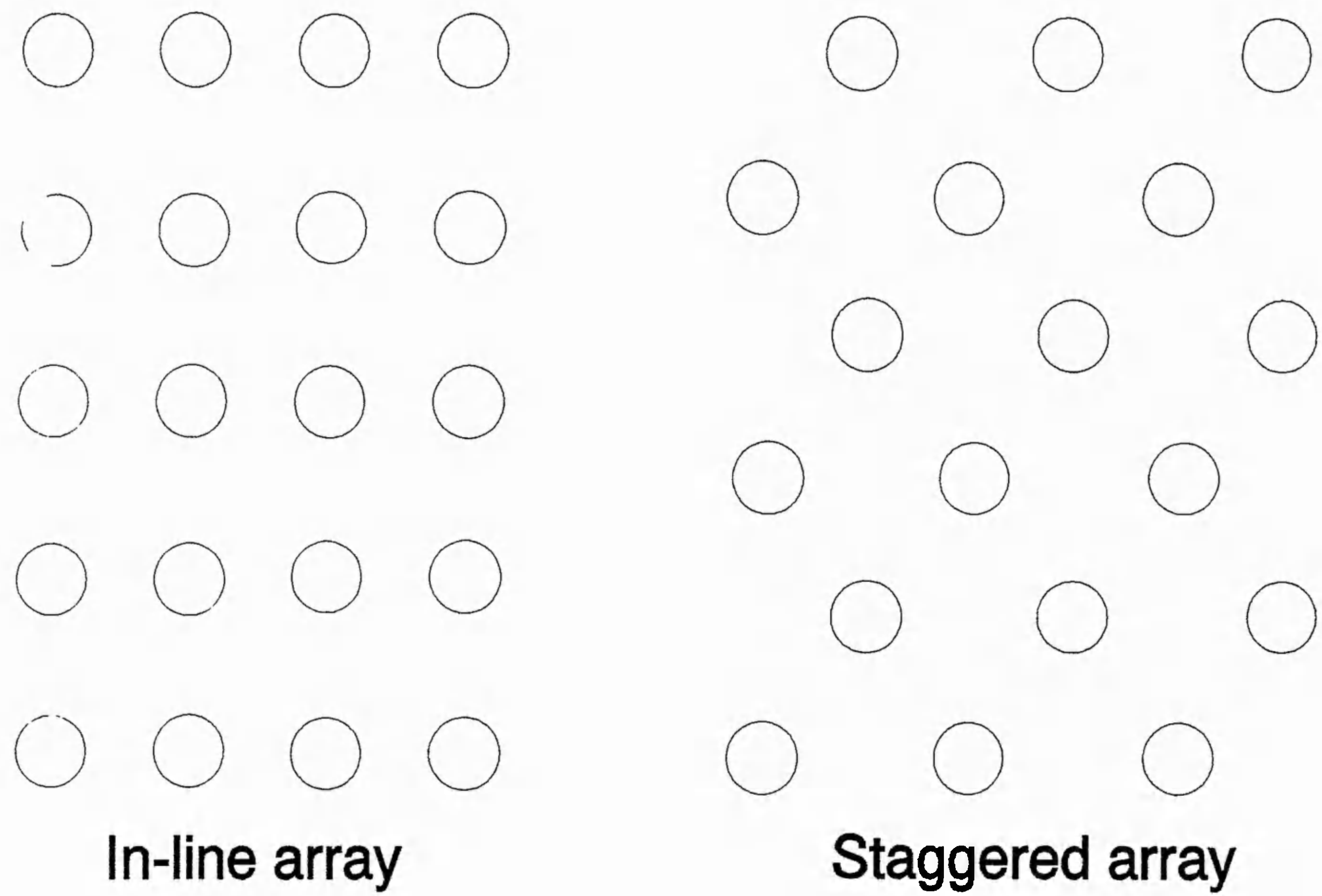


Figure 6. Nozzle arrangement in two possible jet array configurations - view onto the jet plate

Jet flow velocities calculated from pressure measurements (Florschuetz *et al* 1980) show that the jet velocity is higher for the downstream jets because of the higher total flow and so lower static pressure. The higher jet velocities may contribute to the increased value of Nusselt number under the downstream jets found in some jet configurations. Though later measurements by Florschuetz and Isoda (1983) suggest that the jet orifice discharge coefficient will reduce when the ratio of cross-flow to jet mass fluxes (G_c/G_j) is greater than 0.5. This will reduce the jet mass flow and so tend to reduce the heat transfer coefficients.

Measurements of local heat transfer coefficients made at larger nozzle-to-plate spacings (2 to 5 nozzle diameters) by Behbahani and Goldstein (1983) show that (for jet-to-jet spacings of 4 nozzle diameters) the peak levels of heat transfer coefficients under the first three rows of jets are similar. The heat transfer levels fall in subsequent rows. At larger jet-to-jet spacings ($ly/D = 8$) this reduction in heat transfer is much less.

The results of Metzger *et al* (1979) showed that in-line arrays of jets give higher heat transfer rates than staggered arrays of jets. This effect was most significant at lower span wise jet-to-jet spacings and at higher nozzle-to-plate spacings. The differences in the local Nusselt numbers due to the two patterns of jets increased in the direction of the cross-flow. Reduced values of heat transfer coefficients obtained for the staggered hole pattern relative to the in-line pattern were attributed by Florschuetz *et al* (1981) to differences in the span wise distribution of cross-flow velocity. Jets have the effect of locally reducing the cross-flow velocity downstream of the jet and so the jets in the in-line pattern shield the downstream jets from the cross-flow, concentrating the cross-flow between the jet rows. In the staggered pattern the cross-flow velocity distribution across the span is more uniform and so the heat transfer of the impinging jets is degraded further.

The cross-flow induced by an array of impinging jets depends on how the flow is constrained to exit the space between the jet plate and the heat transfer surface. A series of experiments by Obot and Trabold (1987) were performed to investigate this effect. Configurations were tested which constrained the induced cross-flow to exit on one side only, on two opposite sides and on all four sides (Figure 7). These were referred to respectively as the maximum, intermediate and minimum cross-flow configurations. As would be expected the configuration giving the lowest cross-flow rate gives the highest area mean heat transfer coefficients and the configuration giving the highest cross-flow rate gives the lowest area mean heat transfer coefficients. The different configurations all gave the highest heat transfer where the cross-flow as a proportion of the jet flow was lowest. In the minimum and intermediate cross-flow configurations this was at the center of the array, in the maximum cross-flow configuration this was at the end of the array farthest from the exhaust. At the smallest nozzle-to-plate spacing ($z/D = 2$) the heat transfer coefficients seen in the maximum cross-flow configuration exhibited the effect seen by Florschuetz *et al* (1991) where the cross-flow causes a reduction in heat transfer over the first few rows and then an increase in heat transfer in subsequent

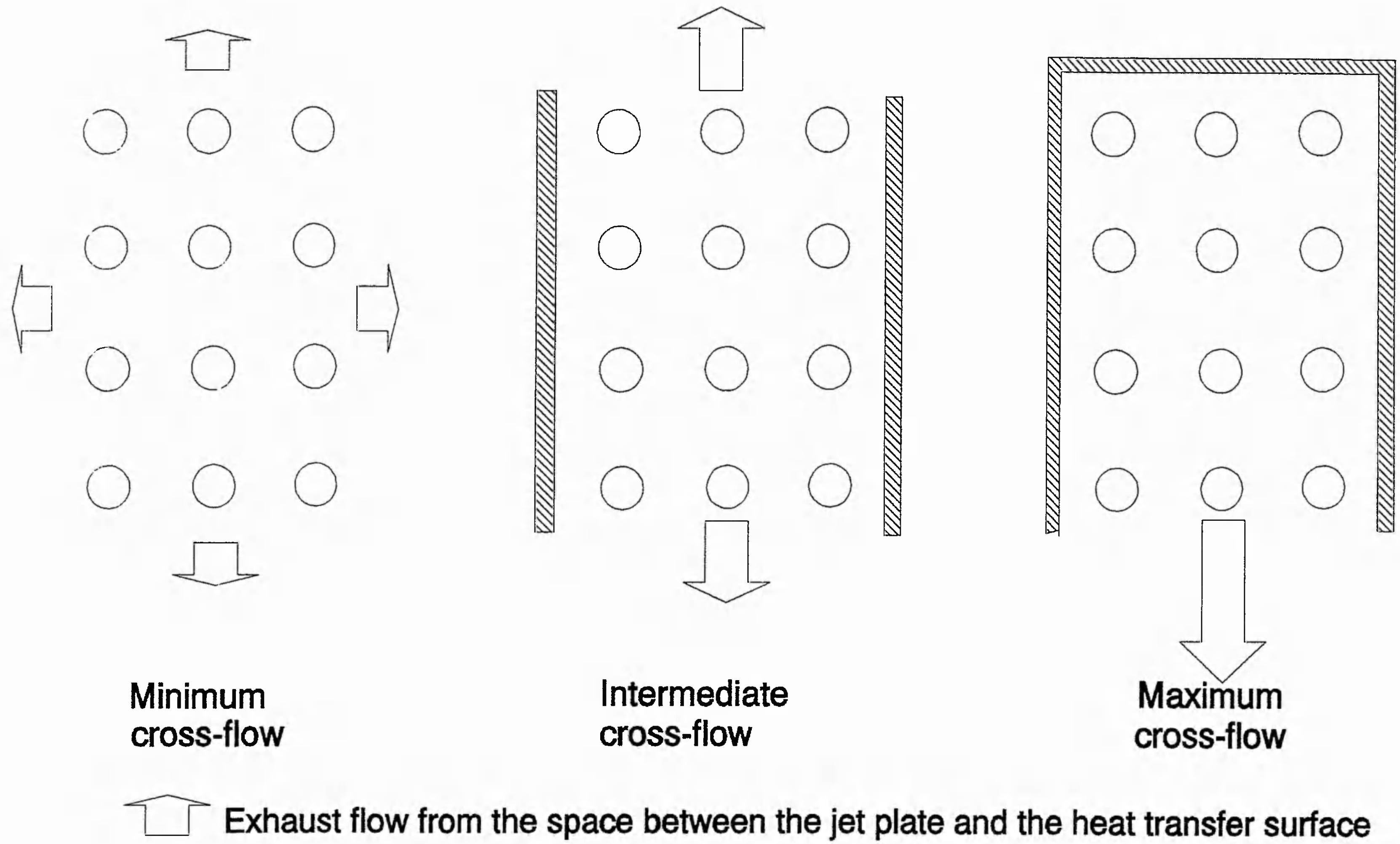


Figure 7. Three possible cross-flow configurations in jet arrays - view onto the jet plate showing side walls constraining exhaust flow

rows.

The number of rows of jets, the constraints placed on the exit of flow from the space between the jet plate and the heat transfer surface and the presence of an initial cross-flow (in the maximum cross-flow configuration) each affect the ratio of cross-flow to jet flow and so also affect the heat transfer coefficients obtained.

Florschuetz and Su (1987) investigated the effect of an initial cross-flow with a temperature different to that of the jet flow. They presented results giving the heat transfer coefficients for each row in the array in terms which included a fluid temperature difference factor

$$\eta_r = (T_{ref,r} - T_{rec,r})/(T_c - T_n) \ ,$$

a temperature recovery factor

$$r_r = (T_{rec,r} - t_n)/((G_n/\rho)^2/2Cp) \ ,$$

and the cross-flow to jet mass flux ratio. Where $T_{ref,r}$ is the surface temperature at which the average heat flux in a region is zero (thus differing from the adiabatic wall temperature where the heat flux is zero at a point), $T_{rec,r}$ is the same parameter measured when $T_c = T_n$. These parameters were used to give an expression for the heat flux

$$q_r = Nu_r \lambda/D \ [(T_w - T_n) - \eta_r (T_c - T_n) + (1 - r_r)((G_n/\rho)^2/2Cp)] \ .$$

This approach is similar to that adopted by Hollworth and Wilson (1985) and by Goldstein and Behbahani (1982) for single impinging jets.

3.5 Concluding Remarks

Jet Impingement Heat Transfer Correlations

A correlation provides a convenient method of presenting and using empirically derived heat transfer coefficients. The work on deriving a correlation (Section 3.1) for single circular impinging jets highlights the difficulties involved in developing a correlation which applies to a wide range of significant variables. These problems include:

(i) The effect of one of the most significant non-dimensional variables, Reynolds number, appears to change with radial displacement from the stagnation point.

(ii) The distribution of local heat transfer coefficient varies non-monotonically with radial displacement from the stagnation point at low nozzle-to-plate spacings.

(iii) The large discrepancies between the results obtained in different experiments mean that these results cannot be simply combined to derive a new correlation.

Despite the efforts to derive a widely applicable correlation, heat transfer coefficients are still most reliably obtained by selecting the experiments with the most appropriate geometry and flow conditions and then deriving heat transfer coefficients from the results of these experiments. However the relationship derived (Section 3.1) does provide a useful method of extrapolating graphical data to new values of Reynolds number.

Because of the increased complexity of the flow field the problems involved in deriving correlations applicable to rows of jets and arrays of jets must be expected to be greater. Indeed the review of the literature did not reveal any correlation expressions giving local heat transfer coefficients due to multiple jet impingement. The distribution of heat transfer coefficient appears to be complex because of interactions between neighbouring jets, effects include:

(i) Mixing between the jet and the ambient air causes a motion in the ambient air. Adjacent jets must be expected to affect each other through this induced motion, this effect will vary in magnitude as the jet-to-jet and nozzle-to-plate spacings change.

(ii) A local increase in the heat transfer coefficient mid-way between each jet is apparently caused by the collision of the wall jets formed by adjacent impinging jets.

(iii) Successive rows of jets in an array of jets induce a cross-flow which affect the heat transfer coefficient due to each jet. Results have shown that the degree to which cross-flow affects the heat transfer coefficient is a complex combination of the jet nozzle arrangement, the relative jet and cross-flow mass flow rates, the nozzle-to-plate spacing and the jet-to-jet spacing.

Impact on the Design of the Knowledge Based Database System

Empirical correlations have been devised which incorporate the effects of many of the significant parameters. Correlations for Nusselt number are given as functions of Reynolds number, Prandtl number, nozzle-to-plate spacing and jet-to-jet spacing. A non-dimensional cross-flow (given in various forms by different authors) is used to quantify the effect of induced or imposed cross-flow on the heat transfer coefficients due to the impingement of both single and multiple jets. The effect of jet-to-jet spacing is introduced for rows and arrays of jets. However the effects of other features have been shown to have a significant effect but have not been incorporated into the available correlations. These features include nozzle geometry, the relative arrangement of the jet nozzles in an array and the constraints imposed on the direction that the exhaust flow from an array of jets may take. These features appear to affect the flow field and hence the heat transfer coefficients but quantification of these effects is difficult. For example the nozzle geometry appears to influence the flow structure in the shear layer between the jet and the ambient air, this affects the generation of turbulent motion in the jet and hence affects the heat transfer. Empirical correlations and graphical data has been obtained for various nozzle geometries, jet nozzle arrangements and exhaust flow configurations. The most appropriate correlation or set of data should be selected in determining the heat transfer coefficients for a particular jet impingement geometry and flow regime. To enable this each correlation in the database must be classified. The classification should be based on a description of the experimental apparatus used. Wherever possible the qualitative description should be extended to include the range of each of the non-dimensional groups which have been shown to be significant. A selection of appropriate correlations should be based on a comparison between a description of a particular geometry and flow regime with each of the descriptions given in the database. The description should be quantified where possible but some significant parameters, such as nozzle geometry, can be described only qualitatively.

CHAPTER 4

THE OUTLINE FOR A DESIGN OF A KNOWLEDGE BASED DATABASE SYSTEM FOR JET IMPINGEMENT HEAT TRANSFER

4.1 The Aims of the System

The tasks of analysis and design of systems of impinging jets are both of practical importance. Methodologies which have been developed for automating both the analysis and design of engineering systems were reviewed in Chapter 2. A feature shared by most of the engineering KBS's discussed is the necessity for a database of design information, though the requirements of the databases differ. For some of the systems discussed a simple tabular database where records are selected by a direct comparison with field values appears to be sufficient, examples include the system for roller bearing selection (Sim and Chan 1991). In contrast Design Support Systems, because they are less specific in their application, must organise and provide access to a broader range of data. Queries to databases from a Design Support System may be formulated by the designer or by the system, for example Arniaiz *et al* (1991) and Fischer and Nakokoji (1992).

In general heat transfer rates vary considerably with geometry and with flow conditions, as a result there are many correlations available, each specific to a different set of geometry and flow conditions. Many correlations have been developed for the various configurations of jet impingement, these are only a small proportion of the total number of heat transfer correlations available. The support of the design of all but the simplest components in which heat transfer is a significant influence on the design will require accurate selection from a diverse collection of correlations.

A consideration of the jet impingement heat transfer literature (Chapter 3) revealed that a large quantity of empirical data exists. This data is already used by engineers in the analysis of heat transfer for systems of impinging jets, but the difficulty in obtaining the best data has prevented the full potential of the empirical data from being realised. The principal aim of this research has been to design and demonstrate a design tool which makes the available empirical data for jet impingement heat transfer more readily available. This has necessitated the extraction from the literature of a sufficiently wide selection of jet impingement correlations, experimental data and methods of applying this information to prove the design of the tool. Because new data frequently becomes available the need to add to or alter the database has also been considered. The methods employed by this system should, if possible, be generally applicable to other forms of convective heat transfer. The starting point for the design of this tool was the work carried out on Expert Database Systems reviewed in Chapter 2.

Each publication presenting heat transfer data tends to cover only a small portion of the possible range of the significant parameters, with one or more parameters being systematically varied by the experimenter in order to determine the effect of that parameter on the heat transfer rate. Over recent years the most significant features in a system of impinging jets have been identified; these features, which were discussed in Chapter 3, include qualitative features such as the arrangement of the jets (a staggered array is one example) and the corresponding key dimensional parameters (jet-to-jet spacing is one such parameter in an array). It has proved possible to incorporate most of these parameters into non-dimensional groups when presenting correlations for heat transfer coefficients, the jet-to-jet spacing (ly) can be combined with the nozzle diameter (D) to give the group ly/D. The choice of suitable non-dimensional groups allows the results from experiments using only a small range of parameter values to be generalised.

For the purposes of a design tool each experiment in which heat transfer coefficients have been measured can be thought of as a separate item which can be used to provide information appropriate to a specific range of parameter values. There are many of these items and new experimental data is regularly published. The design tool must provide a database of some description to store these items, it must enable the most appropriate information to be extracted and it must enable new data to be added as it becomes available. Each of the many items of data can be associated with a qualitative description and a range of numerical parameter values. In selecting an item of data to be used in assessing the heat transfer rate of a given jet impingement system both the qualitative and the quantitative elements should ideally be considered. An item should only be selected for evaluation if the qualitative description of the specified heat transfer problem matches the qualitative description in the database and the dimensions of the problem are such that the important non-dimensional groups lie within the corresponding ranges associated with the correlation in the database. However for certain configurations of jet impingement geometry the heat transfer coefficients are to some extent independent of some of these parameters. It should be possible to take advantage of the knowledge of the circumstances under which limited variations in some of the parameters are not significant in order to select some information which would otherwise be rejected.

At present the most suitable practical solution to the problem of obtaining correlations for the analysis of jet impingement heat transfer appears to be a Knowledge Based System which will select the most appropriate data from a database. This Knowledge Based Database system must be able to direct the user in constructing a specification of the jet impingement geometry which gives the key qualitative features and corresponding parameter values. Based on the description assembled with the help of a Knowledge Base the system must then select the relevant information from a database. Work by Humphrey (1987), Smith *et al* (1987), Parsaye *et al* (1989), Smith *et al* (1989) and Gauch (1990), considered in the review of literature on Expert Database Systems in Chapter 2, investigated the organisation of keywords into hierarchies and the use of hypertext to assist in constructing database queries. These concepts are relevant to the formulation of the qualitative component of the geometry specification. However this work did not consider the combination of words and numeric values which are required to adequately specify data such as heat transfer correlations.

If there is no correlation in the database which satisfies the description then the system should attempt to select an alternative correlation. This correlation should be expected to provide a reasonable approximation to the heat transfer coefficient. It should be possible to progressively relax the criteria for selection so that the errors introduced by choosing an approximate correlation are minimised. The generalisation of the description or query by replacing keywords with those from higher in a hierarchy, suggested by Smith *et al* (1989) and Gauch (1990), will provide one method. However this process should be controlled by a Knowledge Base so that the keywords representing features which are expected to have the least significant effect on the heat transfer coefficients are preferentially removed from the description. Extrapolation of correlations beyond their nominal ranges is a method commonly used by design engineers. The use of a Knowledge Base to control this extrapolation will provide a further method to modify the description.

The KBDS must be able to interpret a wide variety of data, correlations are made up of arithmetic expressions, tables and graphs, in order to yield values for the heat transfer coefficient. The work carried out on Expert Database Systems, described in Chapter 2, is relevant to the correlation selection aspect of the KBDS but these techniques do not consider the problems associated with evaluating the correlations.

Because of low cost the target platform for the Jet Impingement Knowledge Based Database System is a Personal Computer. Publications reviewed in Chapter 2 proposed four different architectures for Expert Database Systems; these included:

(i) A KBS loosely-coupled with a conventional database management system.

(ii) A DBMS enhanced with a query language which supports inference.

(iii) A KBS enhanced to incorporate database management functions.

(iv) A fully integrated data and knowledge management system.

An advantage of the first of these architectures, a KBS loosely coupled to a DBMS, is that the use of available systems is possible. However on a PC with limited memory this approach has a significant disadvantage; the requirement to transfer control between the two systems. Where the memory is sufficiently large for only one of the two processes then each transfer of control between the processes requires the comparatively time consuming task of swapping the programs between memory and disc. The need for integration of the knowledge base and database suggests that transfer of control of this nature may be a frequent requirement. The second of the four architectures, enhanced Database Management Systems, for example Stonebraker and Rowe (1986) and McLean and Weise (1991), are experimental. These systems appear to be aimed at large multi-user computer systems rather than PC's. Some of the functionality required for the system outlined (the evaluation of correlations for example) would require a language in addition to the Database Management language outlined. In addition some of the facilities offered by DBMS, concurrent access and security for example, are not required on a system implemented on a PC. It is possible to enhance KBS, the third architecture, with the necessary database management functions. The structure of Prolog, which is based on first order logic, enables it to be closely coupled to database management functions (Lucas 1988) in such a way that the syntax used to form database queries is a normal part of the language. A further advantage of a system using a logic programming language is that database queries and the expression of constraints in addition to the writing of knowledge bases and other programming tasks can be performed in a single language. The last of these architectures, a fully integrated data and knowledge management system is a research goal (Kerschberg 1990).

To provide a database for jet impingement heat transfer correlations the correlation KBDS must provide a means to represent, retrieve and evaluate data expressed in the form of mathematical formulae, tables and graphs. In general currently available databases appear to operate only on text and numeric values. The scientific database described by Marshall (1991) is an exception. This database provides access to tabular and graphical data, which may be represented as arithmetic expressions and evaluated, so providing interpolation and extrapolation. The purpose was to provide access to a very diverse but inter-related body of data rather than towards making a selection from a large number of discrete and possibly conflicting data items which is the case with the jet impingement correlation database.

The linking of each publication into a network has been suggested; a number of proposals of this type were discussed by Gauch (1990). A network can be formed by using the references given in the documents; links are made from each document to each of the documents to which it refers. A search using this network, moving from document to document via the references, reflects the method a researcher would use to trace useful information (and so would be a very useful research tool). However when applied to the domain of jet impingement heat transfer this method would be unable to discriminate between publications containing empirical heat transfer data and those containing other data. This method of selecting potentially useful documents is not believed to be adequate for the practicing engineer who requires a correlation quickly and who is unable to spend time sifting through publications to determine which do and which do not present empirical data. A more rapid technique for searching for relevant data would be to start with a collection of publications all containing empirical data. An initial selection of documents which are relevant to the type of heat transfer problem under consideration would then be made. Finally a thorough examination of any selected documents to check that the important aspects of the problem are closely matched by the data in the publication would be performed. It is believed that this method would be preferred by practicing engineers, which is the reason such a method has been adopted for the jet impingement heat transfer correlation database.

Knowledge based analysis methodologies reviewed in Chapter 2 included systems for interpreting behaviour from basic principles (Sutherland 1963) or the behaviour of components (Stallman and Sussman 1977). These methodologies are not appropriate because the jet impingement heat transfer process is not well understood (no analytical model is available), nor is the jet impingement process a system in the sense that its behaviour can be inferred from the behaviour of a number of discreet components.

The Scope of the System

The correlations incorporated into the KBDS are used to provide heat transfer coefficients for air only. Data obtained from heat transfer experiments involving fluids other than air and from mass transfer experiments can also be used to derive heat transfer coefficients for air. At present there are no examples of heat transfer obtained using other fluids in the database. The Colburn analogy is applied, using an appropriate choice of Prandtl and Schmidt numbers, to make use of data obtained from mass transfer experiments.

The KBDS includes a representative selection of the available data for jet impingement heat transfer. Heat transfer coefficients for single and multiple impinging jet systems can be derived from correlation expressions and from graphical data. For multiple jet systems only area mean heat transfer coefficients can be derived at present because the KBDS will only interpret the axisymmetric distributions of heat transfer coefficient obtained for a single jet; the more complex operations required to utilise the two dimensional distributions of heat transfer coefficient obtained from obliquely impinging jets, jets in cross-flow and multiple jets have not been incorporated into the system. Heat transfer coefficients and heat flux can be derived for a single non-ambient impinging jet. Methods for deriving heat flux due to multiple non-ambient jets have not been incorporated though work published by Florschuetz and Su (1987) on this subject suggests that analysis methods will be similar in principal to those for a single jet.

4.2 The Use of Knowledge of Jet Impingement Heat Transfer

A Qualitative Characterisation of a System of Impinging Jets

The basis for the selection of a correlation is both qualitative and quantitative. The qualitative aspect can be accomplished by a comparison of the description of a heat transfer problem with the description associated with a correlation in the database. Both of these characterisations can be given by a collection of keywords, the KBDS contains information which controls which keywords may be used. An initial selection of correlations is obtained by a query to the database, the query consists of a set of keywords which describe the heat transfer problem. A correlation is selected if each of these keywords are included in the list of keywords describing the heat transfer process to which the correlation applies. One of the functions of the KBDS is to guide the user towards the formation of a meaningful description. The available keywords can be organised into hierarchies, as a hierarchy is descended the description of the heat transfer process becomes more specific (Figure 8). For example the information on jet impingement can be divided into three topics, the

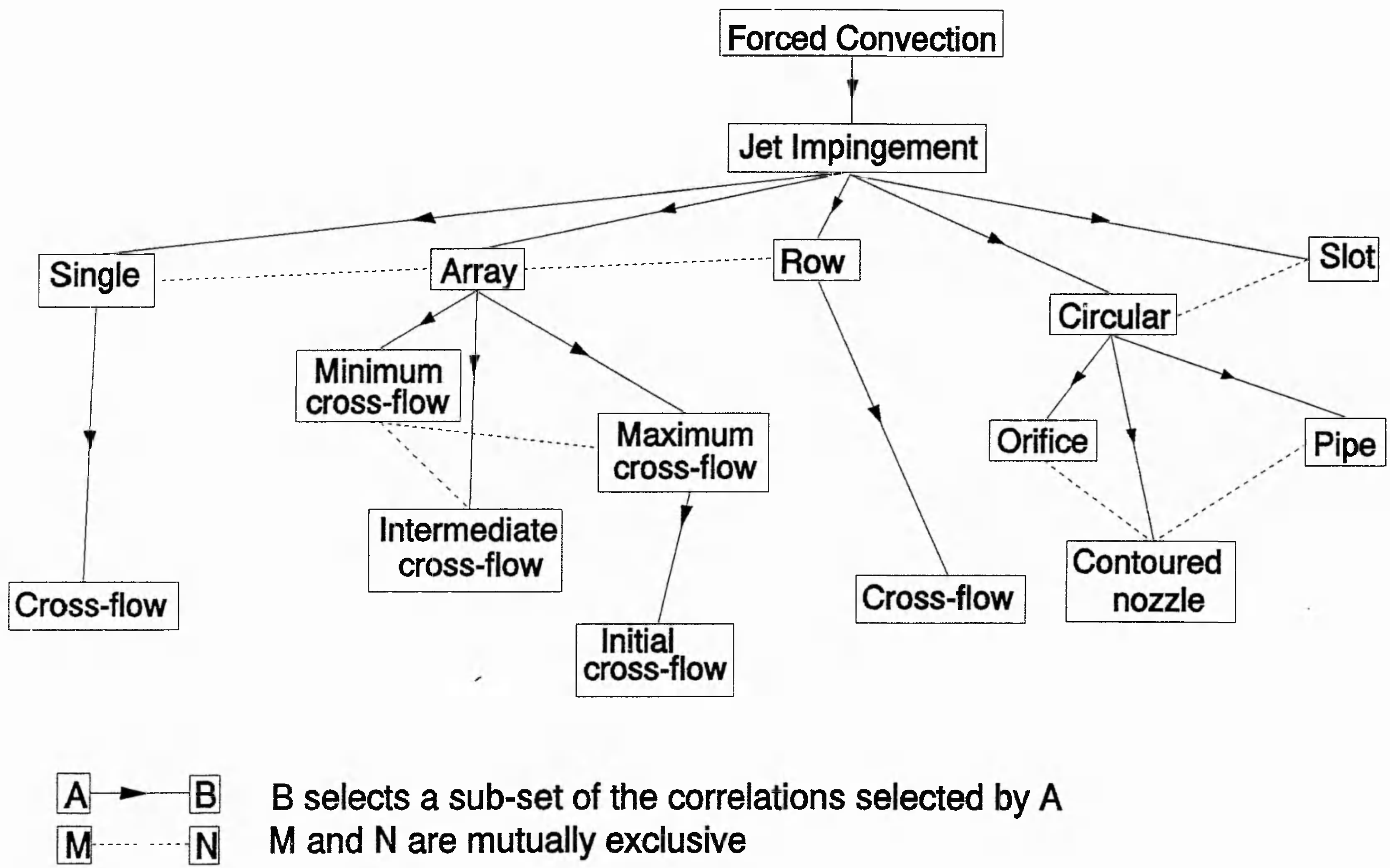


Figure 8. A portion of the network describing jet impingement systems

impingement of a single jet, of a single row of jets and of an array of jets, the three topics are indicated by the keywords single, row and array respectively; each of these topics can in turn be sub-divided. The keywords single, row and array cannot be combined in a meaningful description. However a complication occurs because each topic may be divided in more than one way, for example jet impingement can also be divided by the geometry of the jet nozzles into impingement due to circular jets, slot jets etc. A combination of keywords such as single and circular is meaningful. This information cannot be represented in a simple hierarchy with one type of link between each of its members, a second link must be introduced to express the relationships between keywords. One result of this complication is that the hierarchy cannot simply be followed to its tips to select appropriate heat transfer correlation data. These relationships between keywords will be used by the KBDS to guide the user towards the selection of a meaningful set of keywords.

A Quantitative Characterisation of a System of Impinging Jets

A qualitative description of an impinging jet system does not provide a sufficiently precise definition to enable a heat transfer correlation to be selected or the heat transfer rate to be calculated. To achieve these aims dimensions must be given to the system geometry and flow parameters. As has already been mentioned, the correlations are derived from empirical results obtained over a range of parameter values. This range of parameter values can be considered as the range over which the correlation is valid. The parameters are usually expressed in non-dimensional form using the non-dimensional groups upon which the heat transfer rate has been seen to depend (Chapter 3). For example the flow rate, m, forms part of the Reynolds number and the nozzle-to-plate spacing part of the non-dimensional group z/D. Some parameters, the number of rows in an array for example, are not incorporated into non-dimensional groups (being non-dimensional themselves). Each correlation will have associated with it maximum and minimum limits to the values of these dimensional or non-dimensional parameters. The parameters used in expressing these limits to the correlation validity will vary from correlation to correlation, as will the number of parameters, it must be possible to accommodate these variations.

Parameter values and these non-dimensional groups are used in determining whether a correlation, which has already been shown to satisfy the qualitative description, is appropriate for deriving the heat transfer for a specified jet impingement system. A correlation is appropriate if all the corresponding non-dimensional parameter values for the specified system lie within the limits associated with the correlation.

Aiding the Input of Parameter Values

The number of parameters which are needed to fully specify a system of impinging jets may be large. The number of parameter values needed is further increased by the fact that the various correlations use different parameters in their expressions. This problem can be eased by taking advantage of the relationships between the parameters, these relations allow the values of many of the parameters to be derived from other parameter values. The relationships vary from the simple equation relating nozzle area to diameter,

$$a = \frac{\pi D^2}{4},$$

to the compressible flow equations which enable the mass flow through a nozzle to be calculated from the thermodynamic properties of the fluid the pressure drop across the nozzle,

$$m = f(a, Cd, P1, P2, T1, \gamma, R).$$

Using these inter-relations between the parameters, values for some of the parameters can be derived from values input for other parameters, thus minimising the input required. For example a value for the area, a, can be inferred from the value for the diameter, D, or vice versa. The parameters used to define an array of jets for example are different from those needed to define a row of jets. The system should prompt the user for only those parameters that are relevant to the particular configuration of jet impingement under consideration.

The Use of Domain Knowledge to Direct the Database Search

On an initial search the system may reject all the correlations because either the keywords associated with each correlation do not match the problem specification or the correlation limits are not satisfied by the specified parameter values. Three possible approaches to obtaining useful information under these circumstances are:

(i) To make the problem specification less specific. This can be achieved by removing keywords from the description or query. Keywords are added to the query to make the description more specific, a more specific correlation is expected to produce a more accurate value for the heat transfer rate. If the relative importance of the keywords is known then the keywords to be removed from the query can be chosen so as to minimise the loss of accuracy.

(ii) To rewrite the description given by the user. This will be done to give a description of a heat transfer process which is expected to yield similar heat transfer coefficients to that of the process originally described by the user.

(iii) To selectively relax the limits. In effect this causes the correlation to be extrapolated and so a loss of accuracy would be expected. Knowledge of the relative importance of the parameters in the limits should be used in determining which limits should be preferentially relaxed, so minimising the reduction in accuracy.

The Relative Importance of Parameters Affecting the Heat Transfer Coefficients

By removing a keyword from the description the effect described by the keyword is being ignored. This is acceptable under some circumstances. For instance it has been shown that the geometry of a circular nozzle at its exit has no significant effect on heat transfer coefficients at nozzle-to-plate spacings of 10 nozzle diameters or more. Under these circumstances the keywords specifying a pipe, an orifice or a contoured ASME nozzle can be removed from the database query without significant loss of accuracy.

Another example is taken from the heat transfer measurements of Metzger *et al* (1979), reproduced as Figure 9, which show that the arrangement of the jet nozzles in an array, whether they are in line or form a staggered pattern (Figure 6), can be ignored for arrays with less than eight rows if the jet-to-jet spacing is greater than five nozzle diameters and the nozzle-to-plate spacing is less than 1.5 nozzle diameters. For nozzle-to-plate spacings between 1.5 and 3 nozzle diameters the nozzle arrangement can be ignored for arrays of five rows or less.

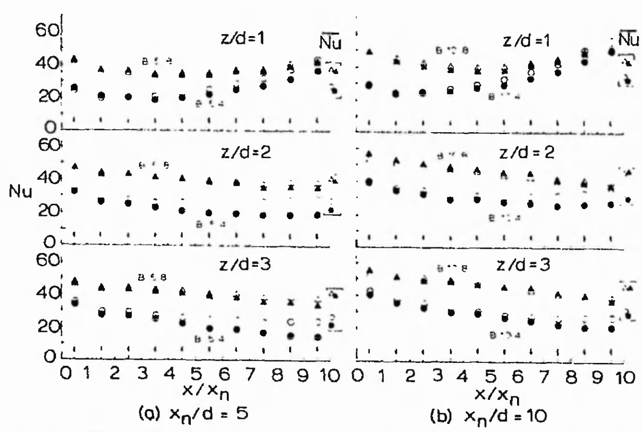


Fig. 6 Effect of hole spacing and hole pattern on chordwise Nusselt number profiles. Open points are inline patterns. Solid points are corresponding staggered patterns. Re* = 196

Figure 9 **Heat transfer due to staggered and in line arrays of jets (x_n = jet-to-jet spacing, x = displacement from the inlet to the array). Reprinted from Metzger *et al* (1979).**

An example of where it is acceptable to relax the limits on the range of parameters over which a correlation can be considered valid may be obtained from the results of Hollworth and Berry (1978). Graphical data of $f(\overline{Nu})$ versus nozzle-to-plate spacing (reproduced as Figure 10) suggests that, at large jet-to-jet spacings (lx/D and ly/D between 15 and 25), the effect of the nozzle-to-plate spacing is small in the range 5 to 25 nozzle diameters. If these conditions are satisfied then the limits of a correlation, initially narrower than $5 \le z/D \le 25$, can be extended, so allowing the correlation to be selected.

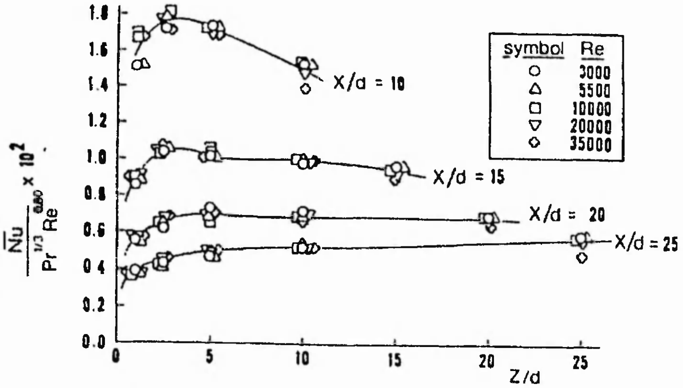


Fig. 8 Effect of Z/d on average heat transfer coefficients

Figure 10 **The effect of nozzle-to-plate spacing on the heat transfer due to arrays with large jet-to-jet spacings (X = jet-to-jet spacing). Reprinted from Hollworth and Berry (1978)[1].**

[1] Permission from the ASME to reproduce the work presented here as Figures 9 and 10 is gratefully acknowledged.

4.3 The addition of correlations to the database

In the field of jet impingement heat transfer new empirical data frequently becomes available. To avoid the Knowledge Base System becoming out of date it must be made possible to incorporate new data. The correct description of the new correlation is vital to enable the Knowledge Base to assist the user in generating a query which will select the new correlation from the database. To achieve this the Knowledge Base must control the keywords which can be used to form the qualitative description of the geometry and flow regime, by informing the user of the non-dimensional groups which are required to give a quantitative description of the range of parameter values over which the correlation is valid and by ensuring that the KBDS will be able to supply values for all the parameters used in the correlation expressions.

Jet impingement is only one of the many heat transfer processes of interest to engineers. Ideally it should be made possible to add correlations for other geometries and flow regimes to the database. A number of facilities have been outlined as desirable features of a correlation Knowledge Based Database system. So that these features will be available for these new correlations it must be made possible to add to the information in the knowledge base which controls access to the database. This will require the integration of the process controlling addition to the Knowledge Base ar ' the Knowledge Base itself. It will also require the development of a format for the knowledge in the Knowledge Base which allows extension of this information.

CHAPTER 5

THE JET IMPINGEMENT KNOWLEDGE BASED DATABASE

Chapter 4 outlined the design for an expert database system for jet impingement heat transfer correlation data. This system will overcome limitations in conventional databases identified in Chapter 1. These limitations include the requirement that a user is familiar with the terminology employed in the database and that methods of specifying information are not sufficiently specific, searches of conventional databases commonly retrieve irrelevant data and fail to retrieve all the relevant data. The system will guide the user in specifying a description of a heat transfer process (overcoming unfamiliarity with terminology and lack of knowledge of the subject of the database) and perform a search of the database selecting all the relevant data. The system makes use of a Knowledge Base to obtain correlation data when none of the information in the database precisely matches the specified description. The system will assist in the addition of new information to the database ensuring that the KBDS will be able to interpret this data. This chapter describes in detail how the design to achieve these goals was implemented.

Diagrams showing the structure of the KBDS are given in Figures 11 to 11d, these diagrams describe the system behaviour by showing the

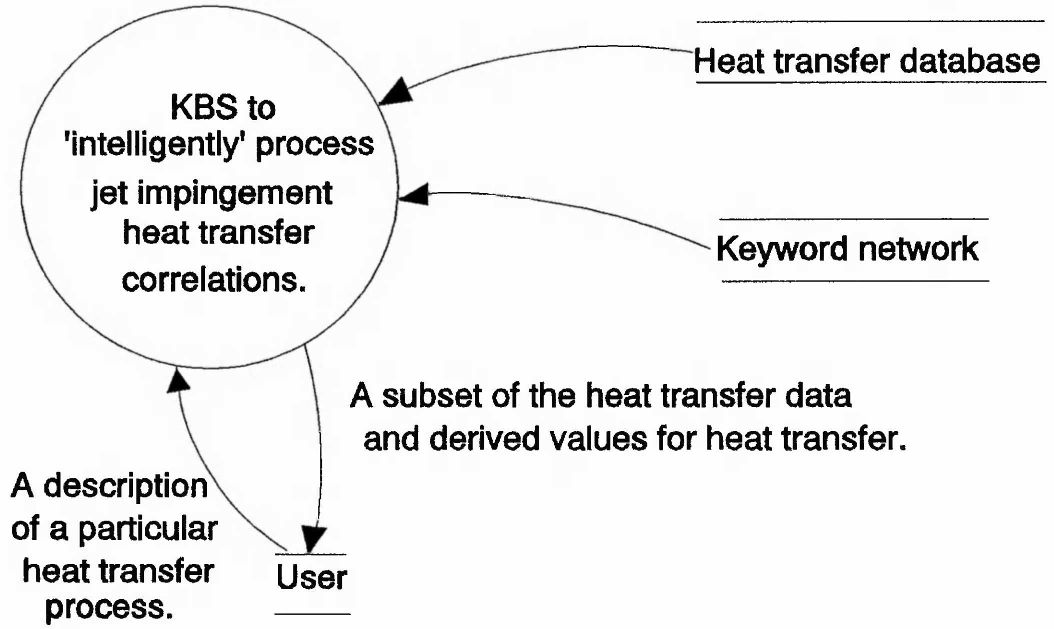


Figure 11. Structure of the jet impingement heat transfer knowledge based system. Context.

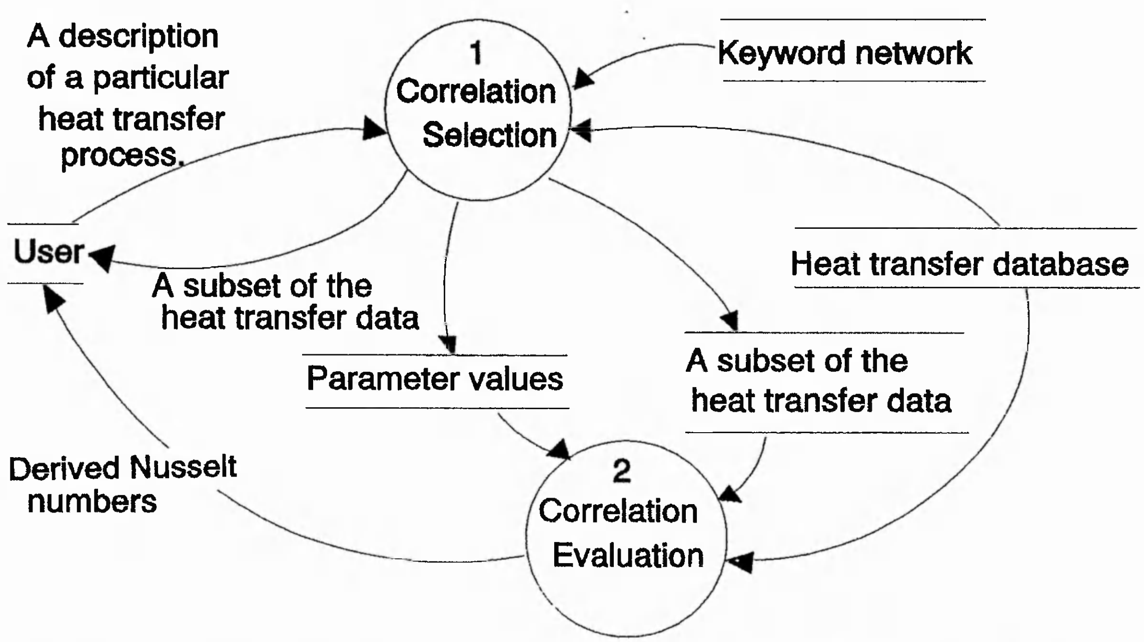


Figure 11a. Structure of the jet impingement heat transfer knowledge based system. Level one.

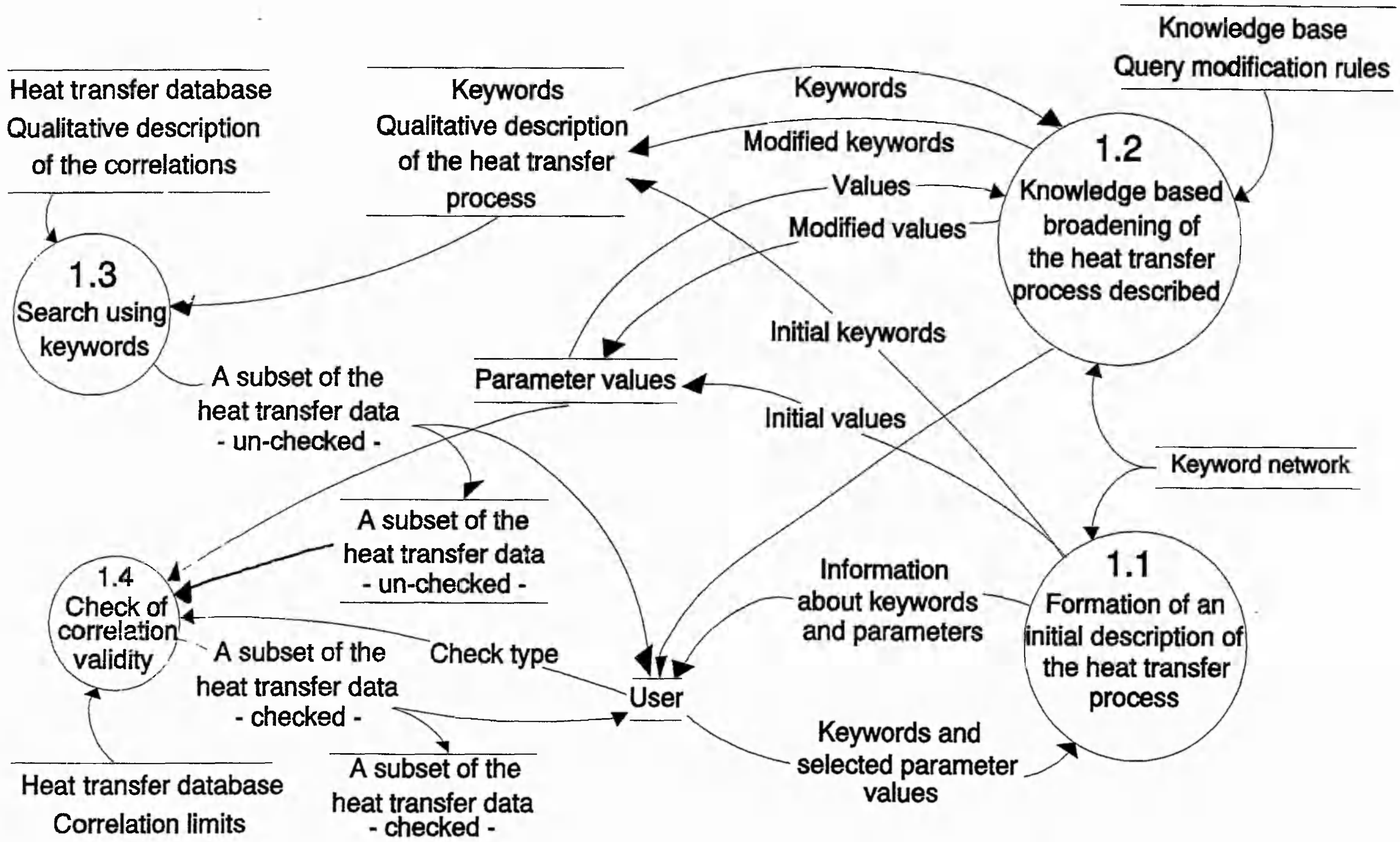


Figure 11b. Structure of the jet impingement heat transfer knowledge based system. Level two, structure of the correlation selection process.

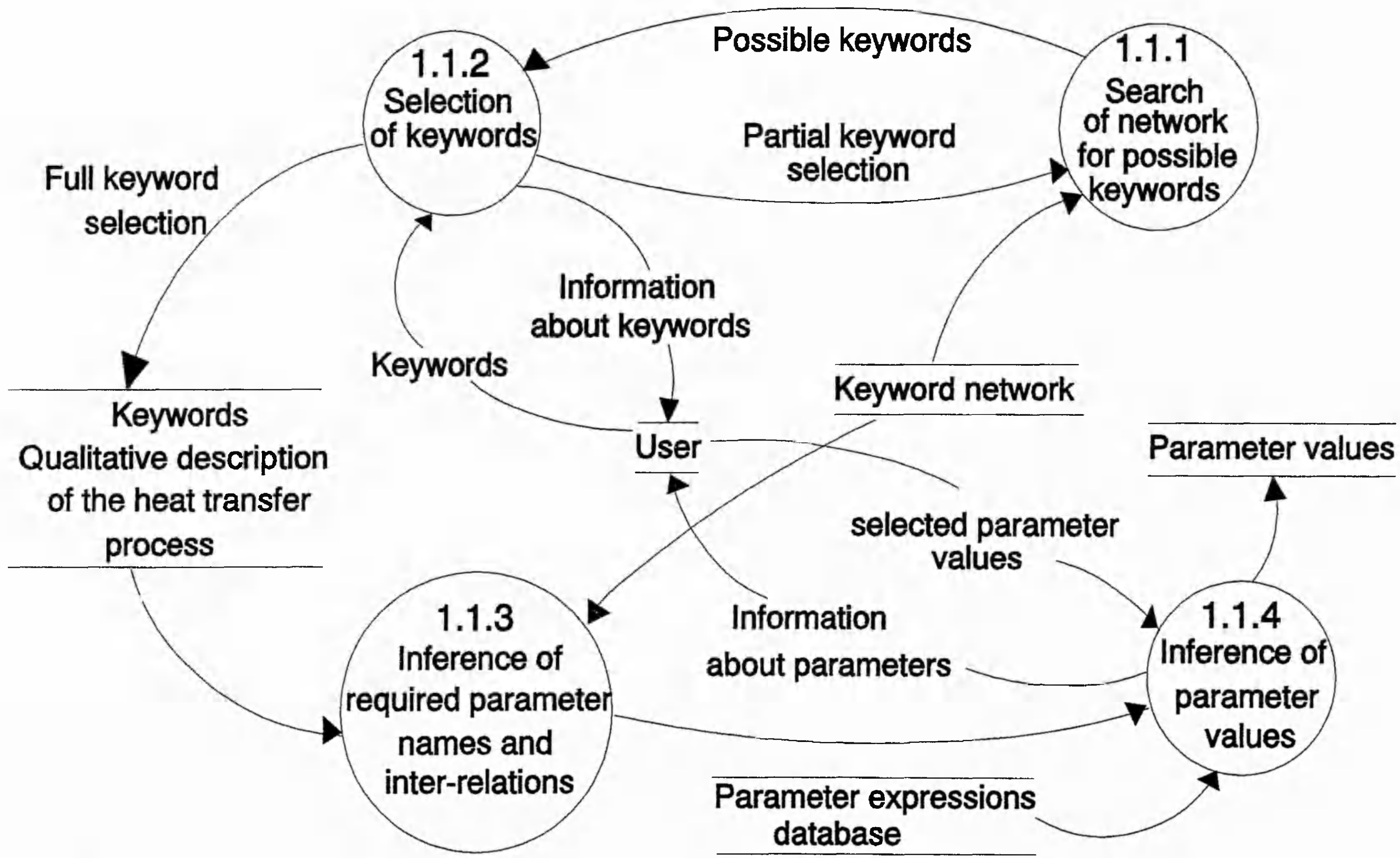


Figure 11c. Structure of the jet impingement heat transfer knowledge based system. Level three, formation of an initial description of the heat transfer process.

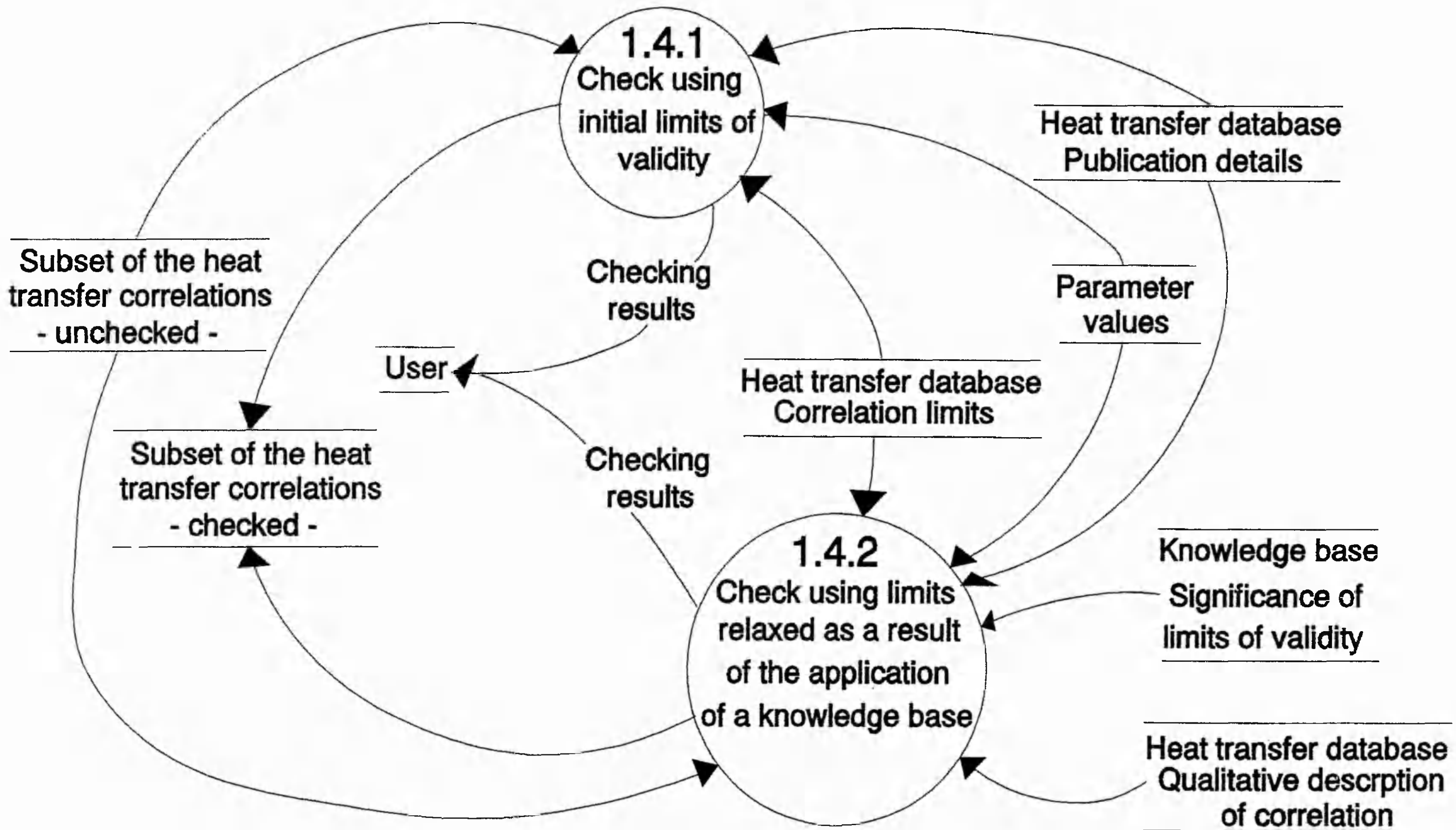


Figure 11d. Structure of the jet impingement heat transfer knowledge based system. Level three, check of correlation validity.

processes and the control and data connections between the processes.[1]

5.1 The Correlation Database

The central items in the heat transfer database are relational tables which give:

(i) A qualitative description of the heat transfer process, ie the geometry and flow conditions, to which each correlation applies.

(ii) A reference to the source of each correlation.

(iii) A quantitative description of the ranges of various parameters or non-dimensional groups over which it is valid to apply each correlation.

(iv) The correlations themselves. This table provides the information needed to derive an appropriate value of Nusselt number for the heat transfer process.

These tables are as described in Lai *et al* (1992), this paper is reproduced as Appendix D. Examples of two entries to the database are given in Figure 12. The records from each of the tables map onto

---

[1] The diagrams follow the conventions described in Ward and Mellor (1985). The diagrams are hierarchical, diagrams at a lower level describe a higher level process in more detail, for example correlation selection (Process 1) in Figure 11a is described as consisting of four processes (Processes 1.1, 1.2, 1.3 and 1.4) in Figure 11b.

PUB.CODE	florsch_81
CONF.	a
KEYWORDS	forced_convection,jet_impingement,array,circular,in_line, orifice,max_crossflow,flat,initial_crossflow

PUB.CODE	gold91
CONF.	a
KEYWORDS	forced_convection,jet_impingement,row,circular,orifice, crossflow,flat

(a) Qualitative descriptions of the heat transfer geometry and flow

PUB.CODE	florsch_81
AUTHORS	Florschuetz,L.W., Truman,C.R. and Metzger,D.E.
YEAR	1981
TITLE	Streamwise flow and heat transfer distributions for jet array impingement with crossflow
JOURNAL	Trans. ASME, J. of heat transfer, 103, 2.

PUB.CODE	gold91
AUTHORS	Goldstein,R.J. and Seol,W.S.
YEAR	1991
TITLE	Heat transfer to a row of impinging circular air jets including the effect of entrainment
JOURNAL	Int. J. Heat Mass Transfer, 34, 8, 2133-2147

(b) References to the source of the correlation

PUB.CODE	CONF.	EXPRESSION	LOWER_VAL	UPPER_VAL
florsch_81	a	rho*v*d/mu	2500.00000	70000.00000
florsch_81	a	lx/d	5.00000	15.00000
florsch_81	a	ly/d	4.00000	8.00000
florsch_81	a	z/d	1.00000	3.00000
gold91	a	rho*d*v/mu	10000.00000	40000.00000
gold91	a	x/d	0.00000	6.00000
gold91	a	z/d	2.00000	8.00000
gold91	a	ly/d	4.00000	8.00000

(c) Limits of validity of the correlations

Figure 12. Two example correlations from the database

PUB.CODE	CONF.	LINE NO.	STATEMENT
florsch_81	a	1	c1=1.18*(lx/d)**-0.944 *(ly/d)**-0.642*(z/d)**0.169
florsch_81	a	2	c2=0.612*(lx/d)**0.059 *(ly/d)**0.032*(z/d)**0.022
florsch_81	a	3	c3=0.437*(lx/d)**-0.095 *(ly/d)**-0.219*(z/d)**0.275
florsch_81	a	4	c4=0.092*(lx/d)**-0.005 *(ly/d)**0.599*(z/d)**1.05
florsch_81	a	5	row=1
florsch_81	a	6	n1=0
florsch_81	a	7	while row=<nx
florsch_81	a	8	re=rho*v*d/mu
florsch_81	a	9	c5=c1*Re**c2*pr**(1/3)
florsch_81	a	10	mc=m*(row-1)+mi
florsch_81	a	11	vc=mc/rho/ly/z
florsch_81	a	12	nu_r=c5*(1-c3*(z/d*vc/v)**c4)
florsch_81	a	13	n1=n1+nu_r
florsch_81	a	14	row=row+1
florsch_81	a	15	end_while
florsch_81	a	16	nu=n1/nx
florsch_81	a	17	write "Area mean htc"
florsch_81	a	18	write nu
florsch_81	a	19	h=nu*lambda/d
florsch_81	a	20	write h
gold91	a	1	re=rho*d*v/mu
gold91	a	2	c1=1
gold91	a	3	nu_line_sum=0.0
gold91	a	4	c2=2.9*re**0.7/(22.8+(ly/d)*(z/d)**0.5)
gold91	a	5	write "Nu (hD/k) on lines parallel to the stagnation points"
gold91	a	6	nu_line_mean_0=c2
gold91	a	7	write nu_line_mean_0
gold91	a	8	while c1=<9
gold91	a	9	nu_line_mean=c2*e**-(0.09*(x/d*c1/10)**1.4)
gold91	a	10	write nu_line_mean
gold91	a	11	nu_line_sum=nu_line_sum+nu_line_mean
gold91	a	12	c1=c1+1
gold91	a	13	end_while
gold91	a	14	nu_line_mean_10=c2*e**-(0.09*(x/d)**1.4)
gold91	a	15	write nu_line_mean_10
gold91	a	16	nu_area_mean=(nu_line_sum +nu_line_mean_0/2+nu_line_mean_10/2)/10
gold91	a	17	write "Area mean htc"
gold91	a	18	write nu_area_mean
gold91	a	19	h=nu_area_mean*lambda/d
gold91	a	20	write h

(d) Correlation statements enabling the heat transfer coefficient to be derived

Figure 12. Two example correlations from the database

Prolog predicates as follows:

description(<u>Publication code,</u>
<u>Configuration,</u>
Keywords),

reference(<u>Publication code,</u>
Authors,
Year),

limits(<u>Publication code,</u>
<u>Configuration,</u>
<u>Expression,</u>
Minimum,
Maximum),

correlation(<u>Publication code,</u>
<u>Configuration,</u>
<u>Line no.,</u>
Correlation statement).

The record key, the field or combination of fields uniquely identifying a record in each table, is shown underlined. The field Configuration is required to distinguish between results from different experimental apparatus described in a single publication.

Correlations could be selected by simply specifying a set of keywords and searching for matching keywords in the 'description' table. This would be the approach adopted in conventional database systems. The application of Knowledge Based System techniques enables an improvement in this process.

The limits of validity for each correlation can be represented simply by a set of expressions, each of which is evaluated as required, and the corresponding minimum and maximum values. The explicit representation of the expressions is important. An alternative database design was investigated which implicitly assigned pairs of fields to each expression (the expression corresponding to each field was determined by the position of the field in a single record). This method was found to become difficult to manage and inefficient as the variety of correlations in the database increased. It was found that a large number of fields would be required to adequately represent all the limits which are significant in the many different possible configurations of impinging jet systems. Furthermore it was found that each correlation would only use a small proportion of the fields available.

5.2 Obtaining access to the most appropriate correlations

The formation of an initial description of the heat transfer process (Process 1.1, Figure 11b) involves the specification of, first, a qualitative description and, second, a quantitative description. The qualitative description of the process in used in a search of the heat transfer database (Process 1.3) to yield a subset of the heat transfer data. The quantitative description is then used in checking the validity if each of these items of data (Process 1.4) to result in a set of correlations which can then be evaluated (Process 2, Figure 11a).

The Keyword Network

All the keywords which can be used to formulate a qualitative description of a heat transfer process are organised into a network. This network is used in helping the user to retrieve relevant correlations from the database.

Two relationships between the various keywords which can be used to describe or classify heat transfer processes were discussed in Chapter 4. A relationship exists between a keyword which can be used to describe a particular aspect of a heat transfer process and another keyword which provides a more specific description of the same aspect of the heat transfer process. Another relationship exists between keywords which describe alternative aspects of a process, these keywords cannot be meaningfully combined in any one description of a heat transfer process. These two relationships can be used to organise the descriptive keywords in a network (Figure 8). A simple network using these relationships was outlined in Lai *et al* (1992), see Appendix D. The resulting network was a directed graph, similar in some respects to an associative network or a semantic net, in having nodes connected by links of different types to represent different concepts. The keywords formed nodes in the network. Two links were used in order to:

(i) Connect one keyword to another keyword which defines a more specific jet impingement process. That is one keyword selects a subset of the heat transfer processes selected by the other keyword. Hierarchies are superimposed on the network using these links.

(ii) Indicate groups of keywords which cannot appear together in a description.

This network was initially constructed in the form of two predicates:

subset(<u>Parent keyword,</u>
<u>Child keyword</u>),
excludes(<u>Keyword one,</u>
<u>Keyword two</u>).

Each keyword being given explicitly as a character string. However the representation of the network in this form prevents the use of any keyword as a label for more than one category of heat transfer. For example 'jet impingement' and 'wall-jet' can be used to identify two categories of heat transfer process, but a wall-jet may be initiated by an impinging jet and so it is reasonable to wish to label a class of wall-jets by the keyword 'jet impingement'. Under some circumstances this use of the keyword 'jet impingement' would cause the program to select incorrect keywords from the network. This failure of the network would occur because the inference mechanism would be unable to distinguish between keywords which were intended to be linked to 'wall-jet' via 'jet impingement', ie those labelling classes of wall-jets initiated by impinging jets, from those which were intended to be linked to 'jet impingement' only, ie those labelling classes of jet impingement. This restriction, which required each keyword in the network to be unique, is not serious in the small example network considered here as alternative keywords can be conceived. However as the database is expanded to cover new categories of heat transfer the restriction would almost certainly prevent the use of widely accepted terminology.

A solution to this problem, which allows the same keyword to be used many times in the network, is to replace each keyword by a unique node number (Figure 13). The keyword character string is then associated with this node via a further link. Navigation through the hierarchy is performed by using the node numbers but interaction with the user is through the associated keywords and text. A query is assembled as a list of node numbers, the corresponding keywords are then obtained for use in the search of the correlation description table. The explicit use of the keywords in this table makes the database more readable and enables the database to be used without the network and the KBDS. The information comprising the network is stored in relational tables (Figure 14) to allow the keyword network to be easily and safely updated. The records in these tables correspond to predicates of the form:

```
subset(    Parent node,
           Child node ),
excludes(  Node one,
           Node two ),
label(     Node,
           Keyword,
           Text ).
```

In addition to the keyword itself a short body of text is associated

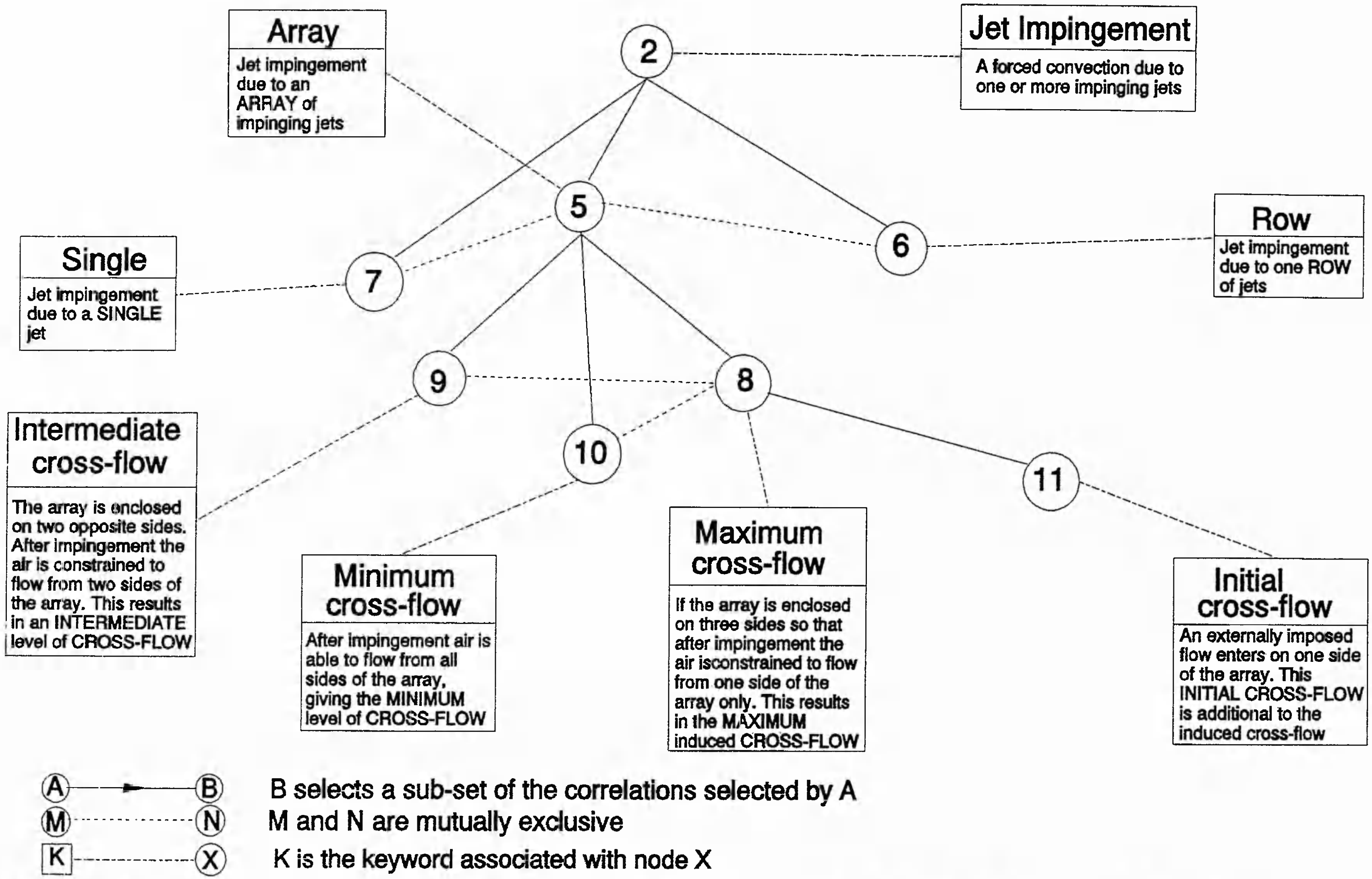


Figure 13. The use of numbered nodes in constructing the keyword network

PARENT	CHILD
1	1
1	2
1	3
2	4
4	5
4	6
4	7
5	8
5	9
5	10
8	11
6	12
7	13
5	22
5	23
4	14
4	15
14	16

PARENT	CHILD
14	17
14	18
15	19
15	20
15	21
35	35
35	24
35	25
35	26
24	28
2	29
26	27
29	30
30	31
29	32
32	33
24	34

The hierarchy linking all the nodes in the network is defined by relations from the table subset(Parent Node, Child Node)

NODE A	NODE B
2	3
5	6
5	7
8	9
8	10
22	23
19	20
19	21
25	26
25	24
14	15
30	32
29	4
28	34
16	17
16	18

Relations from the table excludes(Node A, Node B) are used to link selected nodes

Figure 14. A Portion of the Keyword Network Taken from the Database

NODE NO	KEYWORD	TEXT
1	convection	Root for convection hierarchy
2	forced_convection	FORCED CONVECTION occurs due to an externaly imposed flow, it genarally results in high levels of heat transfer.
3	free_convection	FREE CONVECTION is due to a flow generated by a temperature difference, if the fluid is heated then the fluid will rise due to its increased buoyancy.
4	jet_impingement	A forced convection due to one or more IMPINGING JETS.
5	array	Jet impingement due to an ARRAY or multiple rows of jets.
6	row	Jet impingement due to one ROW of jets.
7	single	Jet impingement due to a SINGLE jet.
8	max_crossflow	If the array is enclosed on three sides so that after impingement the air is constrained to flow from one side of the array only. This results in the MAXIMUM induced CROSS-FLOW.
9	int_crossflow	The array is enclosed on two opposite sides. After impingement air is constrained to flow from two sides of the array. This results in an INTERMEDIATE level of CROSS-FLOW.
10	min_crossflow	After impingement air is able to flow from all four sides of the array, giving the MINIMUM level of CROSSFLOW.
11	initial_crossflow	An externaly imposed flow enters on one side of the array. This INITIAL CROSS-FLOW is additional to the induced cross-flow

Each node in the hierarcy is linked to a keyword and a piece of descriptive text. The definition of this part of the network is given in relations from the table label(Node No., Keyword, Text)

Figure 14a. A Further Portion of the Keyword Network Taken from the Database

with each node. This text is used to describe more fully the meaning of each keyword as it is displayed.

Linked bodies of text form the basis of hypertext systems of which there are a number of commercially available examples. These hypertext systems have been the inspiration for the display which prompts the user to select a keyword. Copies of a series of screens from an example run of the KBDS are given in Appendix A. Screens two to seven (Figures A-2 to A-7) show the hypertext nature of the display and the progression through the keyword hierarchies which enable the building up of an increasingly precise description of the heat transfer process. The network supplies all the keywords and an explanation of the meaning of each of the keywords, as a result an engineer using the system does not require a detailed knowledge of the terminology employed in jet impingement heat transfer before the KBDS can be used to provide useful data.

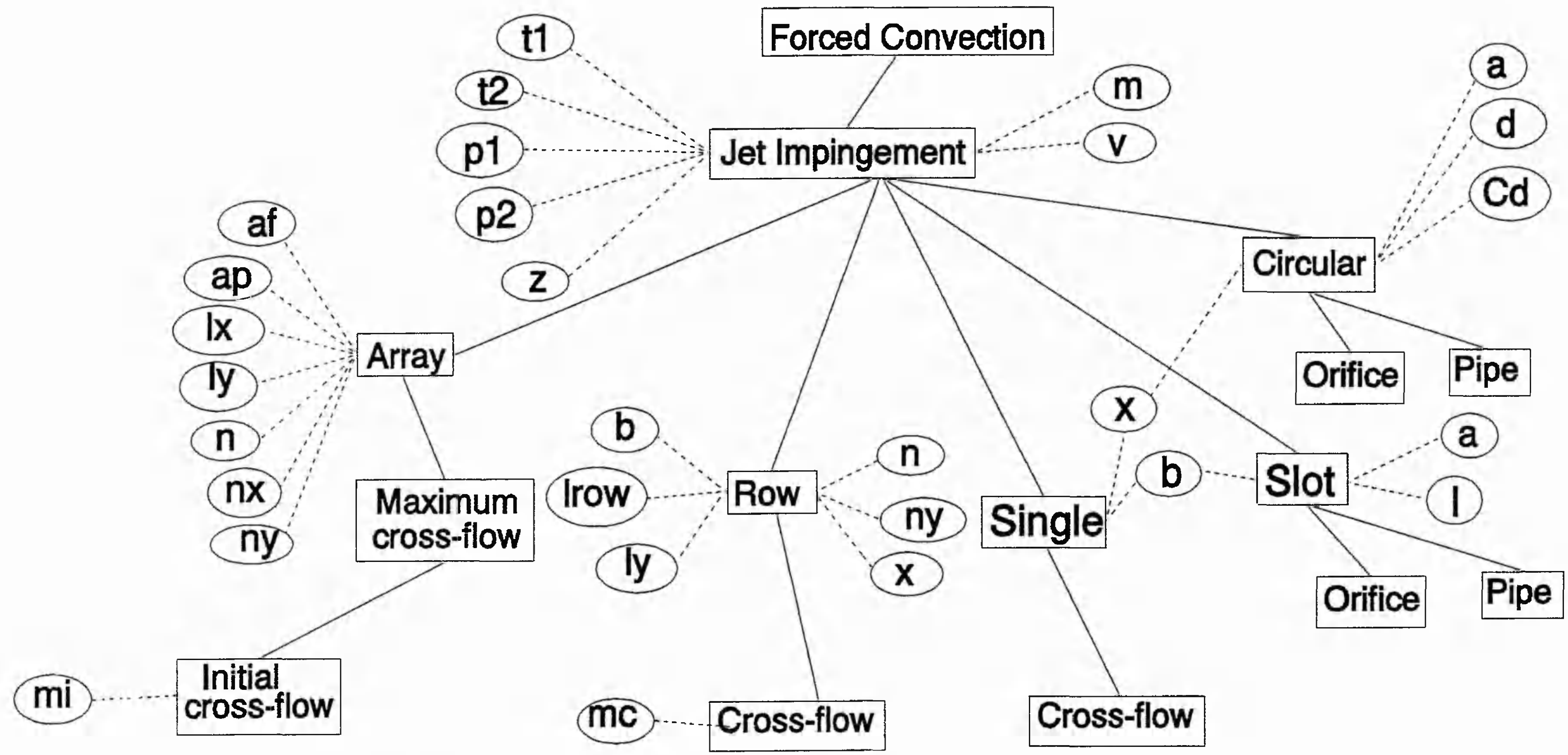
The network contains a hierarchy which classifies and describes the form of the convective flow; in the case of jet impingement this is achieved indirectly by describing how the flow is produced (for example whether the flow is from an array and if so what the geometry of the array is), and the heat transfer surface (whether it is flat, concave...). The hierarchy is descended and keywords selected to give an increasingly specific description of the heat transfer process. To prevent the user from becoming 'disorientated' in the network it would seem preferable that all the alternative keywords, those which further specify a particular keyword, should be displayed together. The link 'excludes' indicates which of the available keywords will be mutually exclusive. This link can also be used to sort the keywords into groups where only one keyword from each group should be selected. The hierarchy is then descended by a breadth-first-search. The search proceeds by extracting all the keywords at the next level of the hierarchy using the link 'subset'. The link 'excludes', which is transitive, is used to put the keywords into groups. Because these links are transitive the network formed is cyclic. The rules which test whether two keywords are connected via links of this type must be written to avoid the potential for non-terminating loops. These two points can be illustrated, to specify the type of jet impingement five keywords are available in the next level of the hierarchy (see Figure 8) these keywords can be organised into two groups so that the user is prompted to select one of array, row or single and then one of circular or slot. The requirement for a breadth first search required the writing of rules to control the search as the built in search method in Prolog is depth first. The Prolog code used to traverse the network is given in Appendix G.

Giving dimensions to the jet impingement system

The dimensions which define the geometry and flow rates in the heat transfer process, described qualitatively by keywords, must be specified to enable an assessment of whether the correlations are within their limits of validity and then to evaluate the correlations. Once the keywords describing a particular heat transfer process have been given the network is then used to infer (Process 1.1.3, Figure 11c) which dimensional parameters are required. Each parameter that may be required is associated with one or more of the nodes in the network (the concept is shown in Figure 15) using a fourth type of link. A parameter is selected if the corresponding nodes from the network are specified by the user in describing the heat transfer process. The parameter names and corresponding keywords are held in a tuple on a database, which can be expressed in predicate form as

```
dictionary(    Parameter name,
               Text,
               Keywords list,
               yes/no),
```

see Figure 16. The text is used to give an explanation of the parameter meaning to the user. A list of keywords rather than the keyword node number is stored in the database. Because keywords may be ambiguous, ie appear more than once in the network, the selection of a parameter only occurs if all the words given in the keyword list associated with the parameter are also given as keywords in the description of the heat transfer process. Using this technique it is very unlikely that unwanted parameter names will be selected. A benefit of this technique is that the database is more readable, which was found to be convenient during the development of the system. For efficiency the keywords in this field of the record could be replaced by the appropriate node number. Parameter selection is performed by Prolog code (Appendix G).



A → B B selects a sub-set of the correlations selected by A

K ········ P Parameter P is required to define a system described using keyword K

NB. For clarity node numbers have been replaced by the corresponding keywords

Figure 15. The identification of required parameters using the keyword network

VARIABLE	TEXT	KEYWORDS	IP y/n
a	Jet area at nozzle exit (m*m)	forced_convection,jet_impingement	y
af	Total area jets / Area jet plate	forced_convection,jet_impingement,array	n
ap	Area of orifice plate or jet plate (m*m)	forced_convection,jet_impingement,array	n
at	Area of target (m*m)	forced_convection,jet_impingement,single	n
b	Equivalent slot width b = d**2*pi/4/lx (m)	forced_convection,jet_impingement,row,circular	n
b	Slot width (m)	forced_convection,jet_impingement,slot	y
d	Jet diameter at nozzle exit (m)	forced_convection,jet_impingement	y
g	Total jet mass flow per unit area target (kg/m/m)	forced_convection,jet_impingement	n
gl	Total jet flow per unit row length (kg/m)	forced_convection,jet_impingement,row	n
gp	Total jet flow per unit area of orifice or jet plate (kg/m/m)	forced_convection,jet_impingement,array	n

Figure 16. A part of the database giving the parameter names

Typically 20 to 30 parameter values are required to give a full set of dimensions to a system of impinging jets. To save the user from calculating and typing all these values, with the associated potential for introducing errors, advantage is taken of the inter-relationships between the parameters. For example for an impinging jet the nozzle area, a, and diameter, D, which may both be required, are related by

$$a = \pi D^2/4.$$

Similarly the nozzle mass flow, m, can be related approximately to the total and static inlet pressures, P and p, the temperature, T, and the effective nozzle area, the product of the nozzle area, a, and the discharge coefficient, Cd,

$$m = f(P,p,T,a,Cd),$$

by the one dimensional isentropic compressible flow equations. When a user enters values these and other equations are used (in Process 1.1.4, Figure 11c) to derive further values where possible. Copies of the screens (Figures A-10 to A- 13 Appendix A) in an example run of the KBDS show how the user is aided in providing the required dimensional input. The fourth field in the parameter table, 'dictionary', is used to indicate whether or not the user should be prompted for a value for the parameter. This enables the prompt to be confined to the more commonly used variables and can considerably reduce the quantity of text the user is required to read. Parameters which are not prompted for must be derived using the expressions inter-relating the parameters. The equations expressing these relationships are held on a relational database (Figure 17), the corresponding predicate has the from

eqn(<u>Parameter name</u>,
<u>Keyword list,</u>
<u>Equation no.,</u>
<u>Line no.,</u>
Equation).

The selection of an equation is determined by two fields in the tuple, the parameter name and the keyword list. The equation is selected and becomes active if the parameter name is required and all the words in the keyword list are also in the description of the heat transfer process. This is necessary, as for the parameter names, to ensure that the correct equations are selected because the same parameter name can be used to represent entirely different parameters in different heat transfer processes.

Using these equations the input required from the user is minimised, it also allows the user flexibility in choosing which parameters he specifies, for example he may enter either the nozzle area or the diameter, he need not enter both. These capabilities improve the usability of the system, but in order to use these equations it was necessary to overcome two problems. Backward chaining inference was found to be unsuitable because this would lead to non-terminating loops. The use of simple forward chaining was also found to be undesirable because for many of the parameters more than one equation is available, this would lead to repeated calculation of

VARIABLE	KEYWORDS	EQN No.	LINE No.	EQUATION
a	forced_convection,jet_impingement,circular	1	1	a=d*d*pi/4
a	forced_convection,jet_impingement,array	2	1	a=af*lx*ly
af	forced_convection,jet_impingement,array	1	1	af=a/(lx*ly)
af	forced_convection,jet_impingement,array	2	1	af=a*n/ap
ap	forced_convection,jet_impingement,array	1	1	ap=lx*nx*ly*ny
b	forced_convection,jet_impingement,row,circular	1	1	b = d**2*pi/4/ly
d	forced_convection,jet_impingement,circular	1	1	d=(4*a/pi)**0.5
d	forced_convection,jet_impingement,row,circular	2	1	d=(b*4*ly/pi)**0.5
g	forced_convection,jet_impingement,single,circular	1	1	g=m/pi/x/x
g	forced_convection,jet_impingement,row,circular	2	1	g=m/x/ly
g	forced_convection,jet_impingement,array	3	1	g=m*nx*ny/ap

Figure 17. A part of the database giving the expressions inter-relating the parameters

each parameter value. These problems were discussed in Lai *et al* (1992), see Appendix D.[2] A solution to these problems, which is to use a hybrid of forwards and backwards chaining, was also outlined. The principles described have been retained but some changes have been made, chiefly to enable the use of parameter equations from the external database. A hybrid rule has the form:

true IF

the value for parameter A is NOT known

AND the value B is known

AND the value C is known

AND A can be derived from the relationship A = *f*(B,C) obtained from the external database.

AND store value for A.

An example of this type of rule, given in the Prolog syntax, is shown below. This rule will derive the jet nozzle pressure ratio from the nozzle inlet and exit pressures.

---

2 At the beginning of the investigation into the feasibility of a Jet Impingement Knowledge Base System an Expert System development tool was used. This tool, Leonardo (Creative Logic Ltd.) was found to have a number of weaknesses which made it unsuitable for this purpose; one of these problems was the inference strategy used by Leonardo. Problems found with Leonardo are discussed in Appendix E.

```
ip_rule :-
        not p_ratio(A),
        p2(B),
        p1(C),
        eval_eqns([eval(p_ratio,[p2,/,p1])],p_ratio(D)),
        assert(p_ratio(D):-!).
```

The Prolog code showing how a relationship is selected and how, using this relationship, a hybrid chaining rule of this form is asserted is given in Appendix G. Before the rule is constructed the equation, or set of equations, forming the relationship is examined to determine which variables are used. The initial conditions of the rule can then be specified, these will determine whether the values for each of these variables are known. The rule will only evaluate the equations if values for all these variables are known. The evaluation of equations and sets of equations will be described in a later section of this chapter.

The user has the freedom to enter values for any of the required parameters. A check (also described in Lai *et al*) is included to ensure that any values entered are consistent both with all the values entered previously and with all those derived using the inter-relations between the parameters.

Consultations with practicing engineers interested in using a KBDS of this type suggested that it was undesirable to prompt the user for all the parameter values. Because many of the parameter values can be derived it is of course unnecessary to do this. As a result a fourth field was introduced into the 'dictionary' table, this field can take the values 'yes' or 'no' to indicate whether or not the user should be prompted for the particular parameter.

Checking that a correlation is valid

The description of a heat transfer process to which a correlation can be correctly applied is both qualitative and quantitative. The qualitative aspect is addessed by a comparison of keywords as detailed previously. The quantitative description can be expressed in the form of minimum and maximum allowable values for the particular parameters or parameter groups associated with a correlation. The limits are used to ensure that the correlation is used only in the appropriate circumstances as discussed in Chapter 4. The limits for a particular correlation, of which there may be any number in the database, are held in tuples of the form:

```
limits( Publication code,
        Configuration,
        Expression,
        Minimum,
        Maximum).
```

The limits appropriate to a correlation are identified by the publication code and the configuration. The limits may be considered in one of four ways:

(i) The limits may be applied strictly.

(ii) A tolerance may be applied to one or more of the numerical parameters used to define the jet impingement system.

(iii) A knowledge base may be used to relax the limits.

(iv) The limits may be ignored.

If the limits are to be strictly enforced each limit expression is evaluated in turn and checked to see that its value lies within the minimum and maximum values. If a value lies outside these limits then the correlation is rejected. The user is informed of the results of the assessment of the limits, see Appendix A (Figures A-16).

A tolerance may be applied to one or more of the parameter values chosen by the user. In this case a test is made to see whether the parameters for which a tolerance have been specified are present in the expression. If they are, then the limits are relaxed by factoring the minimum and maximum values by the appropriate specified tolerance, otherwise they are unchanged. The expression is evaluated and a check is then made to determine whether the derived value lies between the new minimum and maximum values. The correlation is accepted only if this test succeeds for each of the limits.

Knowledge of the influence of a parameter or group of parameters on the heat transfer coefficients can be used to identify a range of conditions where this influence is small. If the particular process for which the heat transfer value is being sought satisfies these conditions the range of values over which a correlation can be considered valid may be expanded. The widening of the limits in this way may allow correlations which would otherwise be rejected to be accepted. For example Behbahani and Goldstein (1983) discovered that the variation in area mean heat transfer coefficients under the first three rows of jets in an array is small. This can be ascribed to the relatively low induced cross-flow in the first three rows of an array of jets. Based on the range of experimental conditions over which Behbahani and Goldstein obtained data the following rule can be derived:

> The number of rows (nx) in an array of jets has a low significance on the heat transfer coefficient when the number of rows is varied in the range 1 to 3
>
> IF
>
> the heat transfer process is forced convection due to an array of impinging jets where the flow is constrained to exhaust from one side of the array only,
>
> AND the jet-to-jet spacing perpendicular to the induced cross-flow is greater than 4 nozzle diameters,
>
> AND the jet-to-jet spacing parallel to the induced cross-flow is between 3.46 and 6.93 nozzle diameters,
>
> AND the nozzle-to-plate spacing is between 2 and 5 nozzle diameters,
>
> AND the nozzle exit Reynolds number is less than 15000.

This rule can be expressed as a rule in Prolog:

```
significance("nx",1,3,Record_keywords,low):-
        all_substrings(Record_keywords,
        [forced_convection,jet_impingement,array,max_crossflow]),
        call_or_eval("nx",E),
        E >= 1,
        E =< 3,
        call_or_eval("ly/d",A),
        A >= 4,
        call_or_eval("lx/d",C),
        C >= 3.46,
        C =< 6.93,
        call_or_eval("z/d",D),
        D >= 2,
        D =< 5,
        call_or_eval("4*m/pi/mu/d",B),
        B =< 15000.
```

A correlation which was only supported by empirical data for three or more rows of jets would, using the strict application of the limits, be rejected for arrays of only two jets. However using the above rule, provided the other conditions were met, the correlation could be accepted. The acceptance of the correlation being justified by the results of Behbahani and Goldstein which showed that under a range of conditions the area mean heat transfer under each of the first three rows of an array of jets is approximately equal. The rules expressing the significance of parameters and the associated Prolog code are given in Appendix G.

Rules such as these to widen the limits should be as general as possible to increase their usefulness. Information obtained from the review of the jet impingement heat transfer literature on the effects of cross-flow suggests that this rule can be used over a larger range of parameters values than those which were actually tested by Behbahani and Goldstein. For example the significance of the cross-flow will reduce as the jet-to-jet spacing on the axis perpendicular to the cross-flow (ly/D) increases, because a larger flow area leads to a lower cross-flow velocity. This reasoning suggests that there should be no upper limit on the value for ly/D in systems to which the rule above can be applied. Furthermore the empirical data was obtained using a staggered array of jets, since the effect of cross-flow is greater in staggered arrays than in in-line arrays the area mean heat transfer coefficients under the first three rows of an in-line array are also expected to be approximately constant. Hence the description of the jet impingement systems to which the rule applies does not include the keyword 'staggered'. Finally high cross-flow rates can lead to a non-uniform pressure distribution under the array, lower static pressures under the downstream nozzles result in higher jet velocities in these nozzles and hence higher heat transfer coefficients on the heat transfer surface below these nozzles. At low flow rates the effects of compressibility are insignificant and so a lower limit on the jet and cross-flow rates are not needed.

The limits are relaxed progressively. These rules which, when applied to allow the widening of the limits, entail the smallest loss in confidence in the derived heat transfer coefficient are tried first. The rules are divided into four groups where the expected maximum error in the heat transfer coefficient or significance is:

very low	10%
low	20%
medium	33%
high	50%

When a rule is used to broaden the limits and so enable the selection of a correlation the user is informed of the significance on the derived heat transfer coefficient of the rule (Figure A-17). This enables an estimate of the potential error in the derived heat transfer coefficient to be made.

If the user requests that the limits are ignored then the limits are simply not checked and an attempt will be made to evaluate all the correlations selected using the quantitative description. If the limits for a correlation are not strictly adhered to then there is a danger that the attempt to evaluate the correlation will fail. Likely causes of failure are attempts to take the root of a negative number obtained by the extrapolation of an expression or table beyond its intended range. The expression evaluator has been written so that arithmetical failures will not cause the program to fail, instead the attempt to evaluate the correlation is halted and processing proceeds to the next item.

Using a knowledge base to broaden the database search

The potential for increasing the number of correlations selected in response to a given description of geometry and flow was outlined in Chapter 4. Additional correlations may be selected if the query to the database is made less specific, this can be carried out by removing keywords from the description of the heat transfer geometry and flow conditions. Rules from the knowledge base are used (Process 1.2, Figure 11b) to modify the initial description of the heat transfer process. The new description is used as before to obtain a set of checked correlations. The selection of less specific correlations can be achieved with little loss in the confidence in the accuracy of the heat transfer coefficients derived from the selected correlations provided the keyword, or keywords, to be removed from the description are chosen with care. The keywords should be removed progressively, removing the least significant keywords first.

A further example of where this idea may be applied is the effect of cross-flow on the heat transfer due to a single impinging jet. A study by Goldstein and Behbahani (1982) suggested that the change in heat transfer due to a cross-flow would be less than 10% provided that the nozzle-to-plate spacing is less than 6 nozzle diameters and that the velocity of the jet at the nozzle exit is 9 or more times greater than the cross-flow velocity. Within this range Goldstein and Behbahani showed that the heat transfer rate was at first enhanced and then diminished by increasing levels of cross-flow. Any cross-flow in heat transfer problems meeting the appropriate conditions can be ignored and the keyword 'cross-flow' omitted from the heat transfer problem description and so from the database query.

Before this rule is selected the heat transfer process to which the rule applies must match the heat transfer process described. This is fulfilled when two conditions are met:

(i) The rule must have associated with it a list of keywords giving a qualitative description of the heat transfer systems to which it applies. This list should be as short as possible so allowing the most general application of the rule. The rule can be applied when the keywords associated with the rule are a sub-set of those used to describe the heat transfer process.

(ii) The permitted range of the significant parameter values must also be represented. The rule can be applied when the parameter values fall within the corresponding range.

A feature in the heat transfer process, described by keywords, may be assigned one of four levels of significance. Each of these levels denote a different loss in the accuracy of the heat transfer coefficient derived by ignoring the feature. The loss in accuracy anticipated by ignoring a feature is based on published empirical data. The levels of significance are:

very low	10%
low	20%
medium	33%
high	50%

The form of the rule based on the work of Goldstein and Behbahani is:

The significance of cross-flow is very low

IF

the heat transfer process is forced convection due to a single impinging jet issuing from a circular nozzle in a cross-flow,

AND the nozzle-to-plate spacing (z/D) is less than or equal to 6,

AND the ratio of jet exit velocity to cross-flow velocity is greater than or equal to 9.

Which expressed as a Prolog rule is:

```
significance("crossflow",Context,Query_keywords,low):-
    Context = ["crossflow","jet_impingement","single","circular",
    "forced_convection"],
    all_members(Context,Query_keywords),
    call_or_eval("z/d",A),
    A =< 6,
    call_or_eval("v/vc",B),
    B >= 9.
```

The aim of the heat transfer KBDS is to select correlations which are expected to provide an accurate value of heat transfer coefficient applicable to the geometry and flow described. Two strategies have been described which extend the number of correlations which can be selected to satisfy a given description; these either make the system described less specific or allow limited extrapolation of correlations. An alternative approach stems from the recognition that many complex heat transfer processes are composed of a number of more simple heat transfer processes. If the data from a complex process is reported in sufficient detail it should be possible to extract results applicable to a simple process from this data. These results can then be used in the evaluation of heat transfer coefficients for processes similar to the simple process. This approach may be applied to jet impingement heat transfer; for example, the use of heat transfer data obtained from measurements made for an array of jets can be used in deriving heat transfer coefficients for a row of jets. One condition which must be satisfied before results for an array of jets can be used is that data specific to each row in the array is available. Measurements of heat transfer coefficients under each row in an array of jets were made by Kercher and Tabakoff (1969) and Florschuetz *et al* (1981). The results for the first row in an array can be used provided that the total width of the surface either side of the row of jets over which heat transfer coefficients are required (2x/D) is within the range of the spacing between the rows of jets in the array (lx/D). In addition other key non-dimensional parameters, Reynolds number, jet-to-jet spacing (ly/D) and nozzle-to-plate spacing (z/D), must also correspond. The application of this rule is illustrated in an example run of the KBDS given in Appendix A, Figures A-22 to A-30.

Given an existing description of a heat transfer geometry and flow a new description, consisting of query keywords, parameter names and corresponding values, can be generated by a rule which performs the following actions:

(i) The existing query keywords are checked to determine whether the given criteria are satisfied,

(ii) the new keyword list is created by adding and deleting keywords,

(iii) parameter values input for the original query are reused where possible in the new query.

An example of a rule of this type, expressed in Prolog, is:

```
generate_query(Old_query,New_query,Names,Values1):-
    all_members(["forced_convection","jet_impingement","row"],
        Old_query),
    not( member("crossflow",Old_query)),
    remove_keyword("row",Old_query,Query),
    append(["array"],Query,New_query),
    x(X),
    ly(Ly),
    ny(Ny),
    Two_x is 2*X,
    intersect(Old_query,New_query,Common_keywords),
    required_vars(Common_keywords,Names,_),
    paramete[illegible]_list(Names,Values),
    append([lx([illegible]_x),ly(Ly),n(Ny),nx(1),ny(Ny)],Values,Values1),
    !.
```

The new set of keywords are then used, as before, to search the heat transfer database for an initial selection of correlations. The quantitative description of the heat transfer geometry and flow may not be complete at this point. If further parameter values are required then these will be obtained using the same algorithm as was used to derive the original set of parameter values but using a set of parameter inter-relations appropriate to the new set of parameters. An assessment of whether the limits of validity of the newly selected correlations are satisfied, or sufficiently close to being satisfied, is then made. Correlations found to be satisfactory can then be evaluated.

A given parameter name will often have either a different meaning or a different value in different heat transfer systems. Figure 15 illustrates this; for example the parameter 'x' is used to give the displacement of a line from the line through the stagnation points under a row of jets, 'x' is also used to give the radial displacement of a point from the stagnation point under a single circular jet. Each parameter name only obtains a firm meaning once the appropriate keywords have been specified, different parameter names requiring different sets of keywords. If the keywords used to describe a heat transfer process are reduced in number so as to broaden a search the meanings or values of some of the parameters may no longer be fixed. A similar argument applies to the equations used to relate the value of one parameter to those of other parameters. For example an expression has been obtained (Bragg 1960) which predicts how a discharge coefficient (Cd), known for one set of flow conditions, will change as the pressure ratio across the orifice varies. The discharge coefficient also depends on the nozzle geometry (Hay and Spencer 1991). Hence a discharge coefficient calculated for one nozzle geometry would not be appropriate for other nozzle geometries.For this reason parameters, or equations inter-relating parameters, which depend on keywords that have been removed from the heat transfer process description cannot be used safely, even though they might provide a useful approximation. The solution adopted to this problem has been to make these parameter values unavailable for use in evaluating expressions. For example a rule exists in the knowledge base which, provided certain conditions are met, removes the keyword 'initial crossflow' from the description of a heat transfer process. In evaluating correlations where cross-flow is a parameter the user is prompted to enter a value, this value is used thus preventing any loss in accuracy as far as possible. This cautious approach prevents the inappropriate use of parameters. However at present there are no rules in the knowledge base which would cause the inappropriate use of parameters. This is largely because the parameter names used to describe the various types of impingement were chosen to avoid inconsistent use of names, this was done to make the system easier to use rather than to avoid this type of conflict.

5.3 The expression interpreter

Three items of information in the database are given as expressions or arithmetic procedures rather than explicitly as numerical values. These items are:

(i) The simple expressions used to give the limits of validity of the correlations.

(ii) The inter-relations between the various parameters.

(iii) The heat transfer correlations themselves.

A language which contains the ability to branch and to perform loops, such as the if-then-else and the do-while features of high level languages such as Fortran or C, is required to express many of the parameter inter-relations and the more recent correlations. See for example the correlation based on the work of Florschuetz et al (1981) given in Figure 12. Values are derived from these expressions or procedures by an interpreter. The statements are obtained from the database in the form of a character string. These statements are then tokenised, that is to say broken into a list of recognised symbols (if, while, write, +, / etc.), numbers and words (variables). This list of symbols is interpreted using a top down or recursive descent parser. The statements may contain variables, when a variable is encountered in a statement the value for the variable is substituted if it is available. If no value is available the the user is prompted for a value. Intermediate results derived during the evaluation of a correlation are discarded after the evaluation of the correlation has been completed, this allows variable names introduced by a correlation to be safely used in subsequent correlations where the meaning and value of the variable may be different.

The interpreter is written in Prolog (Appendix G), the writing of the interpreter was simplified by making use of the Prolog grammar rule notation. The interpreter acts on programs using a conventional arithmetic syntax but with the addition of specialised commands to obtain values from tables and graphs. These are analogous to functions or subroutines, and cause the use of a predicate to perform a complex but standard or commonly required operation. The predicate may be written in Prolog or in a conventional procedural language such as 'C'. Issues in incorporating predicates written in procedural languages will be discussed in Chapter 6.

5.4 The Requirements of a System to Update the Correlation Database

The correlation data is held in four relational tables in an external database. The advantages of storing the information used by a program in a separate database are well established. Advantages for this particular system include the potential for a non-programmer to add further correlation data to the system. This is particularly important as it allows the new data, which is continually being produced, to be incorporated into the database; it also enables the addition of proprietary data. The disadvantage is that the information about each correlation, which can be represented in the database is constrained by the record format. The record format must be carefully chosen so that all the important information can be included.

The relational tables used to hold the correlation data have the following format:

description(<u>Publication code,</u>
<u>Configuration,</u>
Keywords),
reference(<u>Publication code,</u>
Authors,
Year,
Title,
Source),
limits(<u>Publication code,</u>
<u>Configuration,</u>
<u>Expression,</u>
Minimum expression value,
Maximum expression value),
correlation(<u>Publication code,</u>
<u>Configuration,</u>
<u>Line no.,</u>
Correlation statement),

see Figure 12.

To obtain the most appropriate correlations from this database part of the knowledge based database system (in effect a knowledge based 'front end' to the correlation database) makes use of further information. This includes knowledge of how to construct a query using keywords and which parameters are appropriate to define a particular heat transfer process. The information used also includes the equations inter-relating these parameters. These equations are used to derive required parameter values from previous input wherever possible; so minimising the data input task. This information is organised in a network. Non-dimensional expressions which are appropriate for specifying the limits are also incorporated into the network. This information is used to assist in the correct addition of new correlations to the database. The non-dimensional groups are displayed to the user at the appropriate point during the process of addition of a new correlation. The user may choose to enter different non-dimensional groups if required. There is an intimate connection between the front end, and in particular the information used by the front end, and the correlation database; this connection must be maintained when the database is extended to include new correlations. A rigorous validation of a number of aspects of any new data is required to maintain the connection. The rules performing this validation are a form of integrity constraint required to maintain the coherence of the many data tables used by the database.

The network, the keywords, the parameter names, the equations inter-relating the parameters and the recommended non-dimensional groups, is held in external database files allowing permanent changes to the KBS to be made without changing the source code. The network information and parameter equations are held in records which map onto predicates in the KBS, these predicates have the form:

subset(<u>Parent node,</u>
<u>Child node</u>),

excludes(<u>Node one,</u>
<u>Node two</u>),

label(<u>Node,</u>
Keyword,
Text),

dictionary(<u>Parameter name,</u>
Text,
<u>Keyword list,</u>
yes/no),

eqn(<u>Parameter name,</u>
<u>Keyword list,</u>
<u>Eqn. no.,</u>
<u>Line no.,</u>
Equation),

non-dimensional group(<u>Non-dimensional expression,</u>
Text,
<u>Keyword list</u>),

see Figures 12, 16, 17 and 18.

EXPR	TEXT	KEYWORDS
4*m/d/pi/mu	Nozzle exit Reynolds number	forced_convection, jet_impingement,single,circular
z/d	Nozzle_to_plate spacing	forced_convection, jet_impingement,single,circular
x/d	Radial displacement from the stagnation point	forced_convection, jet_impingement,single,circular
vc/v	Cross-flow velocity ratio	forced_convection, jet_impingement,single,crossflow
4*m/d/pi/mu	Reynolds number at exit to a nozzle	forced_convection, jet_impingement,row,circular
z/d	Nozzle-to-plate spacing	forced_convection, jet_impingement,row,circular
ly/d	Jet-to-jet spacing	forced_convection, jet_impingement,row,circular
x/d	Lateral displacement from a line through the stagnation points	forced_convection, jet_impingement,row,circular,flat
vc/v	Cross-flow velocity ratio	forced_convection, jet_impingement,row,circular,crossflow
r/d	Curvature of the heat transfer surface	forced_convection, jet_impingement,row,circular,concave
x/d	Lateral displacement from a line through the stagnation points	forced_convection, jet_impingement,row,circular,concave

Figure 18. Part of the keyword network giving the recommended non-dimensional expressions

5.5 The validation of new correlation data and the controlled adaptation of the knowledge based front end to process this data

During the process of updating the database the KBS enforces a number of constraints on any new information to be added either to the correlation database or to the keyword network. A number of these constraints are complex in comparison to normal database constraints, but they are vital to the correct operation of the knowledge based database system. The normal database constraints are enforced by the KBS, these constraints include data type and range. Also included is the requirement for one field (or combination of fields) to uniquely identify each record in a table, ie form a key (a rule of this type is referred to as an entity constraint). A further important constraint is the requirement for referential integrity, ie if a field in a table refers to information held in a second table then for every instance of the field in the first table a corresponding record should exist in the second table. If this information is missing then the database will be failing to fully represent an aspect of its subject matter. These normal database constraints are enforced by rules in the KBS, see Appendix G. Additional constraints are required to guarantee that the KBS will be able aid a user in developing a description of a heat transfer process, to select appropriate correlations and to evaluate these correlations. The enforcement of these additional constraints requires that the KBS perform an analysis of the contents of data fields.

Correlations for heat transfer processes which are similar to those already in the database can be added provided the following constraints are satisfied:

(i) The keywords used to describe the heat transfer process must satisfy the requirements of the network. That is to say that all the keywords used must be linked, either directly or by implication, by the link 'subset'. In addition none of the keywords should be indicated as being mutually exclusive, again either directly or by implication, by the link 'excludes'.

(ii) Every keyword between each keyword in the description and the root of the keyword hierarchy must be present.

(iii) The parameter names used in the limit expressions and the correlation statements must be included in the group of names associated, either directly or by inheritance, with the specified keywords.

(iv) It must be possible for the expression interpreter to parse the expressions and statements. The interpreter will be unable to evaluate expressions and statements which cannot be parsed.

It is also possible to add correlations for types of heat transfer processes not already in the database; these are types of heat transfer which cannot be described adequately using the currently available keywords and parameters. To allow these types of correlation to be added to the database new keywords and parameter names must first be added to the network. To take full advantage of the capabilities of the knowledge based system two further sets of information should be entered. This information comprises equations relating the new parameters to either other new parameters or to existing parameters and non-dimensional expressions which may be used to specify the limits of validity. The addition of this information to the keyword network is also subject to constraints which must be satisfied before the information is accepted. In addition to entity and referential constraints three further constraints apply:

(v) The nodes and the links of type 'subset' must form a hierarchical network. This part of the network must not be cyclic.

(vi) The link 'excludes' must only be used between nodes at the same level in the hierarchy.

(vii) The constraints on the parameter names used to express the relationships between parameters and the constraints on the form of the statements for these relations are as given above (constraints iii and iv)for the limits expressions and correlation statements

(viii)It is possible for the user to determine for each parameter whether the KBS should prompt for a value for the parameter or whether the KBS should derive a value for the parameter. The equations required to derive the parameter value must exist if the value for the parameter is not to be requested directly. These two constraints are also applied to the recommended non-dimensional expressions.

Figure 19 shows the structure of the program used to control the

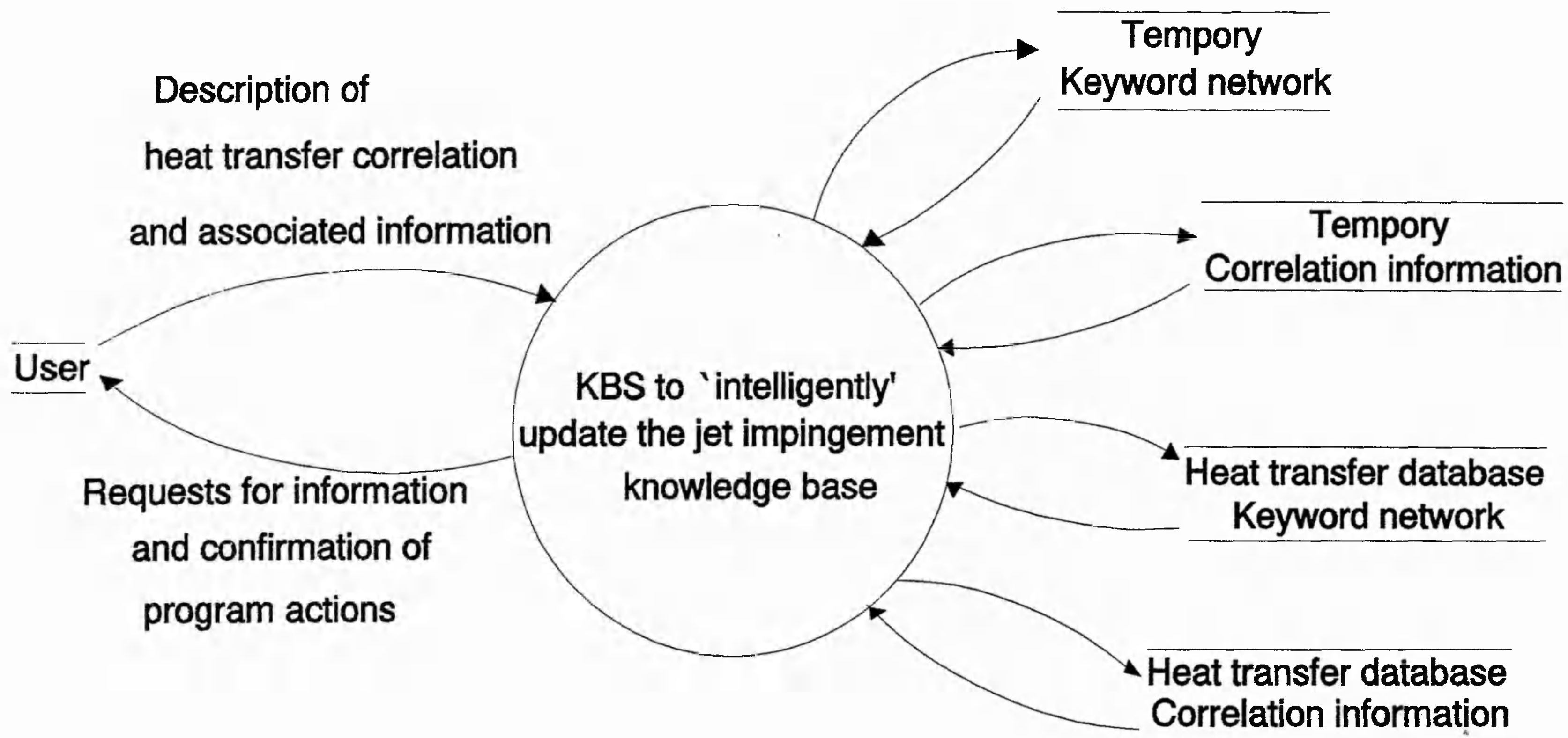


Figure 19. Structure of the update program for the jet impingement heat transfer knowledge based database system. Context.

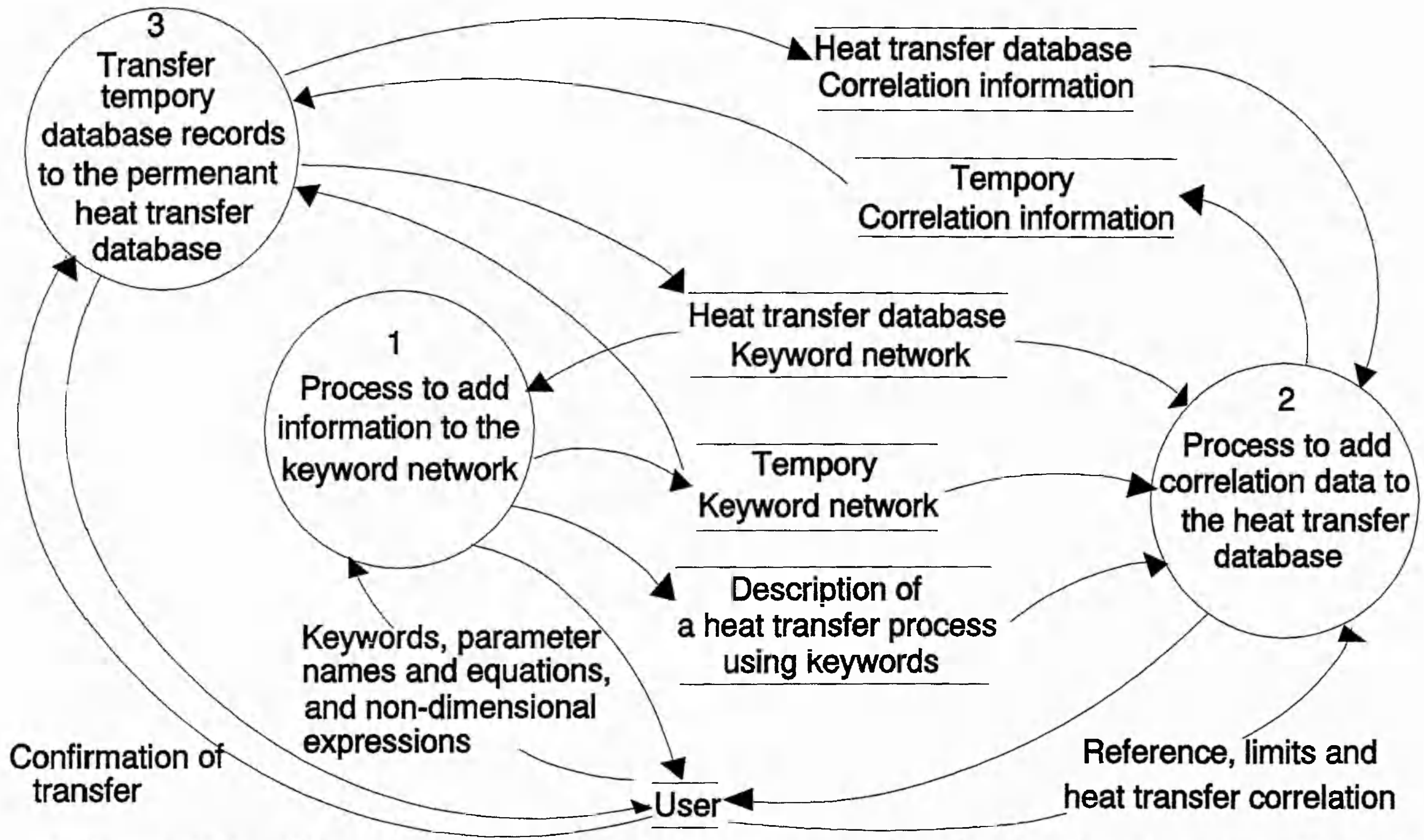


Figure 19a. Structure of the update program for the jet impingement heat transfer knowledge based database system. Level one.

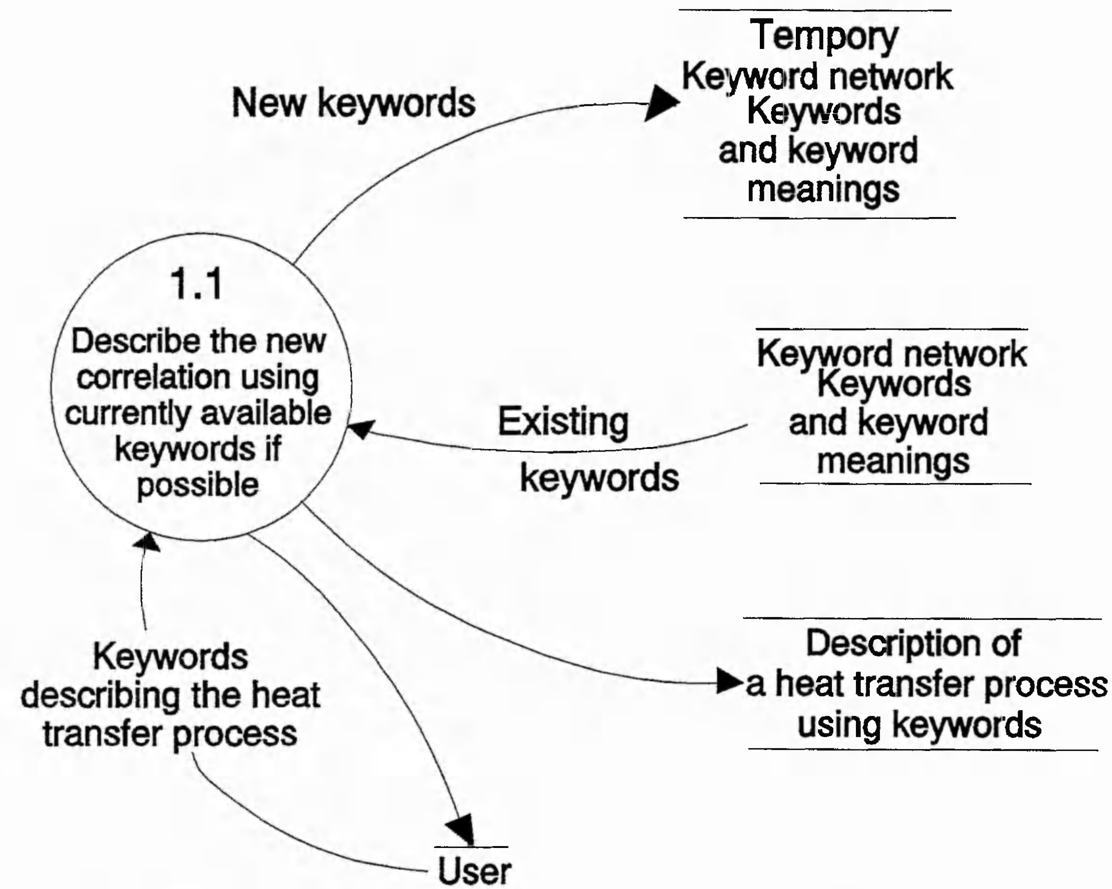


Figure 19b. Structure of the update program for the jet impingement heat transfer knowledge based database system. Level two, process to add information to the keyword network.

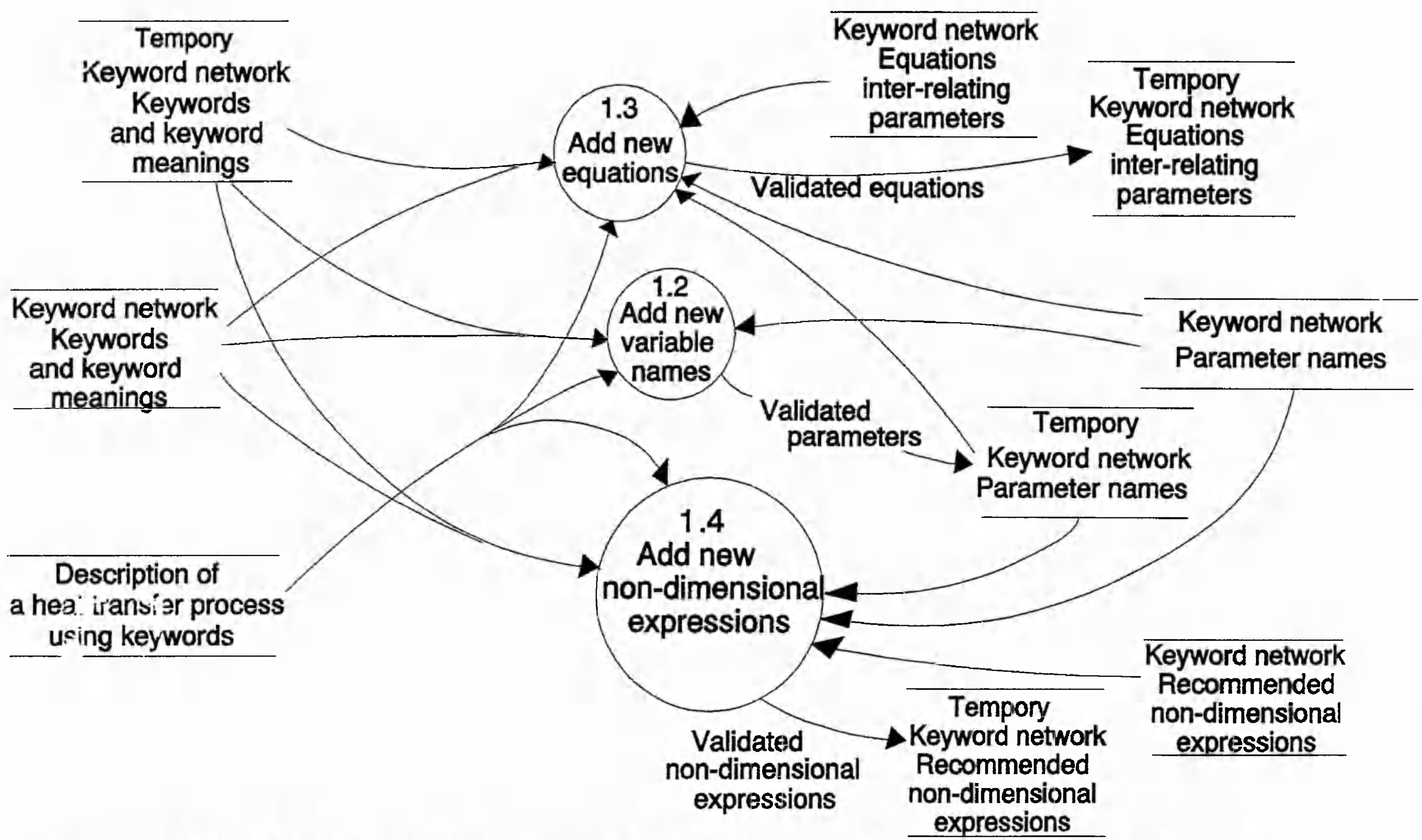


Figure 19c. Structure of the update program for the jet impingement heat transfer knowledge based database system. Level two, process to add information to the keyword network.

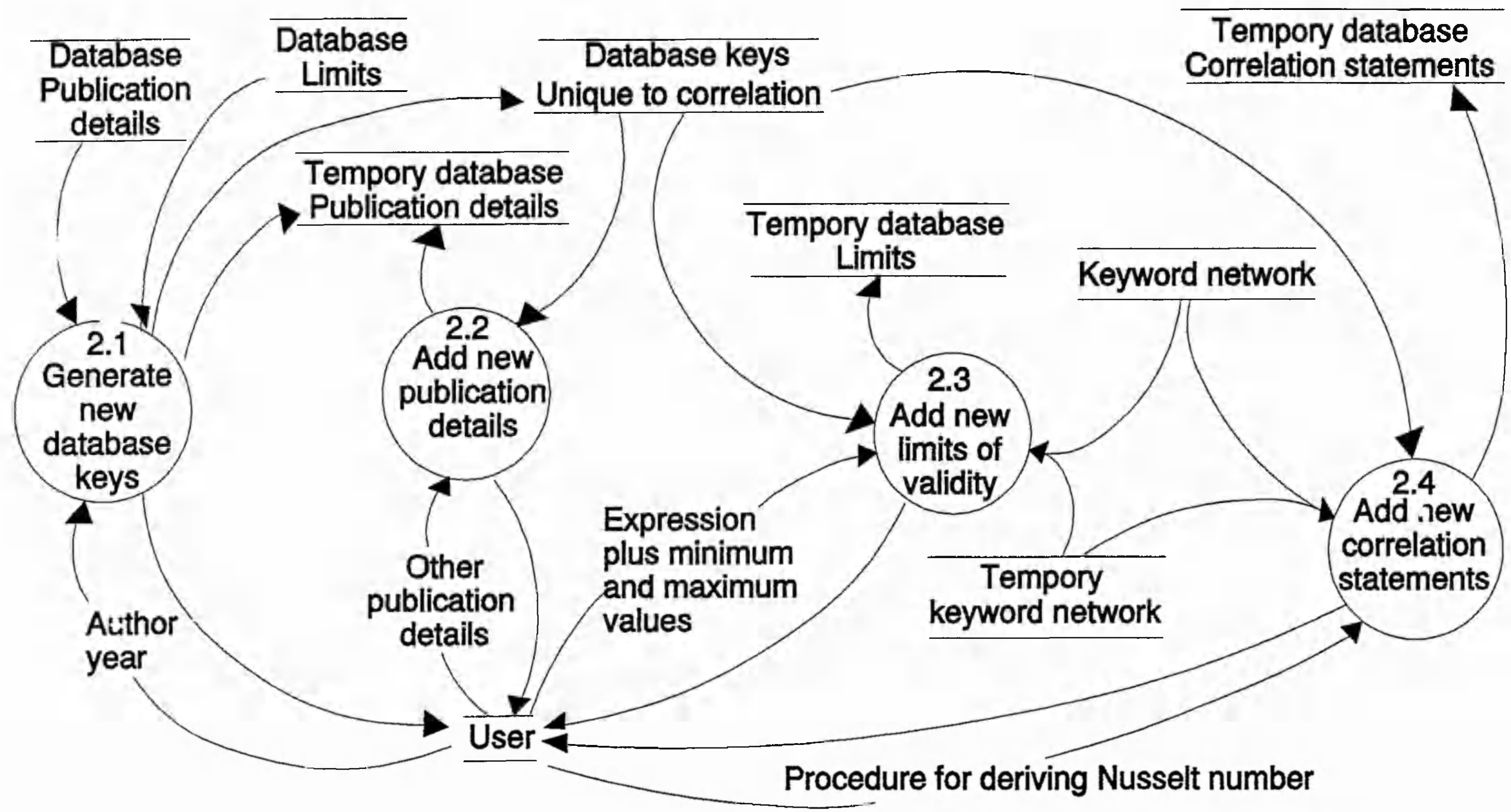


Figure 19d. Structure of the update program for the jet impingement heat transfer knowledge based database system. Level two, process to add correlation data to the heat transfer database

addition of new information to the correlation database. An example run of the update program is presented in Appendix B, this shows the addition of a correlation for a type of forced convection not previously known to the knowledge based system. The example given demonstrates the addition of new keywords and parameter inter-relations as well as the addition of the heat transfer correlation data.

The addition of new correlation data

The simplest additions to the database are those correlations for heat transfer processes which can be described by keywords already in the network. Firstly a list of keywords is built up with the aid of the existing keyword network in the same way as a description during a query to the database. This process (Process 1.1, Figures 19b and 19c) ensures that the keyword list is valid and meaningful, so satisfying the first two of the constraints identified. Secondly the information pertaining to the correlation is specified, this operation can be divided into four processes (Figure 19d). After entering the publication details, the KBS requests the limits of validity for the correlation. An expression and the corresponding minimum and maximum values for the expression are then required for each limit. Recommended non-dimensional expressions are displayed. The user may choose to use the recommended expressions or to use other expressions if required. The user is able to add any number of limits, the number of limits that are required will vary from correlation to correlation. Checks, described above, are performed on each expression entered.

Finally the correlation statements are entered, there is no limit on the number of statements in a correlation. The number of statements required to specify a correlation also varies considerably. After an expression or statement is entered the update program performs two checks, if these checks are successful then the expression or statement will have satisfied the third and fourth of the constraints identified above. The expressions and statements can then be incorporated into the database. A check is made on the parameter names used in the expression, the parameter names must either refer to a constant known to the system (the specific gas constant, R, for example) or to a variable associated via the network with one of the specified keywords (Figure 15). If the parameter is not found by one of these tests the user is informed, the user must either correct the expression, if it is incorrect, or enter the parameter into the network. The addition of a new parameter to the network is discussed below. In the case of the correlation statements a parameter may also be defined by a previous statement from the same correlation. The second check is on the syntax of the expression, this is performed by parsing the expression, if this is successful the expression is assumed to be correct. The rules used to obtain and check limit expressions and correlation statements are given in Appendix G.

Describing a heat transfer process not previously known to the knowledge ba...

If a correlation from a field of heat transfer not previously known to the system is to be added to the database the keyword network will need to be extended. This requires the additional processes shown in Figures 19b and 19c. Referring to the example network given in Figure 8, to add a correlation for heat transfer in a wall jet the keyword ‘wall jet’ would be added at the same level as the keyword ‘jet impingement’. The keyword ‘wall jet’ will select a subset of the correlations selected by ‘forced convection’. The network will also require extending if a more precise description is to be given for a correlation from a field of heat transfer for which data has already been entered. For instance it might be required that the type of circular orifice, whether ‘sharp edged’ or ‘square edged’, is specified. By adding the appropriate keyword at the level below ‘orifice’ the particular type of orifice could be specified.

These modifications to the keyword network are carried out by the update program. Copies of the computer display showing the interaction between the user and the KBDS in adding a new correlation to the database are given in Appendix B. As before the user is prompted for keywords to describe the heat transfer process to which the correlation applies, this is done by offering the available keywords in the network in groups starting from the most general keyword. If, at any point in this process, the keywords available in a group do not suitably describe the heat transfer process then the user may enter a new keyword (Appendix B Figures B-2 to B-4). A new keyword may be entered because the available keywords describe processes from different fields of heat transfer, in this case it is not possible to combine the new keyword in a description with the existing keywords in the group. The user is able to indicate that the choice of the new keyword in a qualitative description of a heat transfer process must exclude the choice of a keyword from one of the alternative keyword groups. This information will be represented in the network by using the 'excludes' and 'subset' links. Alternatively the new keyword may be required to increase the precision of the qualitative description, in this case there will be no alternative keywords. The link 'subset' will be sufficient to represent this information. The user is also prompted for text giving a brief explanation of the keyword's meaning (Figure 11). The new keyword will be incorporated into the network as predicates and will be stored in the corresponding tables in the database.

```
subset(    Parent node no.,
           New node no.),
excludes(  New node no.,
           Alternative node no.).
label(     New node number,
           New keyword,
           Text).
```

By building up the network in this way the fifth and sixth of the constraints are implicitly satisfied.

Under the circumstances where correlations for a new heat transfer process are to be added to the database it is most likely that the parameters to be used in the correlations will not be known to the system. The parameter names and any equations inter-relating the parameter values can be entered into the knowledge base. The user is prompted for parameter names and a brief piece of text describing each parameter. The user will also be prompted to state whether a future user, employing the knowledge base to access the correlation data, should be prompted for a value for the parameter or whether the knowledge base should be left to calculate the value. Figure 14 shows how the parameters are associated with keywords in the network, a parameter is used in giving a quantitative description of all heat transfer processes whose description includes the associated keyword. For each new parameter the user is also prompted to select which of the keywords the parameter should be associated with. These keywords must be one of the keywords specified previously to describe the new heat transfer process. The keywords chosen should be selected from as high in the hierarchy as possible, this will save repeated definition of the parameter in the future. The parameter names and associated information will be added to the network using predicates of the form:

```
dictionary(    New parameter name,
               Text,
               Keyword list,
               yes/no),
```

where 'yes' or 'no' indicates whether or not the user should be prompted for the parameter value.

Procedures inter-relating these parameters may be added to the knowledge base. The statements in these procedures use the same syntax as the statements in the heat transfer correlations. As for the correlations the statements are checked to ensure that the parameter names used are known to the system and that the expressions use the correct syntax (constraint vii).

eqn(<u>New parameter name,</u>

<u>Keyword list,</u>

<u>Eqn. no.,</u>

<u>Line no.,</u>

Equation),

Before the new parameter names are added to the database a further check is made to ensure that the parameter may be derived from equations if no prompt for its value is to be given (constraint viii).

After a qualitative description of the heat transfer process is given to the system, which may involve one or more new keywords, and after the parameter names to be used in the correlations have been identified to the knowledge base the correlations themselves may be added to the database. The method of adding new correlation data and the corresponding constraints have already been described.

All the new records to be added to external database files are initially held as temporary internal predicates in the update program. These predicates are only transferred to permanent storage in the external database files when the process for defining the new correlations has been completed and the user is satisfied that the predicates to be added are correct.

CHAPTER 6

SOME PRACTICAL PROGRAMMING CONSIDERATIONS

6.1 Interfacing between procedural languages and knowledge based systems

Interfacing between a procedural language such as C and a knowledge based system is frequently desirable. Tasks which are procedural in nature, for example mathematical algorithms which are common in engineering problems, are easier to write and execute more efficiently in a procedural language.

Three types of integration can be identified, these are 'loosely-coupled', 'closely-coupled' and 'fully-integrated'. Loosely-coupled systems consist of separate processes or programs which communicate with each other by data passing, usually via files. Loosely-coupled systems are easier to create but if large quantities of data must be transferred or if control must pass frequently between the processes the system can be made to operate more quickly by close coupling. In addition, especially on Personal Computers, if the programs are large the first program will have to be cleared from memory to make way for the second. Closely-coupled processes are combined to form one program where data is transferred through shared memory addresses, this is much faster than using files. A fully-integrated system would be obtained from a program development system which provides a language suitable for expressing both rules and procedures. A fully-integrated system might be expected to provide some benefits relative to closely-coupled systems, particularly during program development, easing the difficulties of interfacing between two languages, however the resulting program is a single process as for a closely-coupled process.

Simple arithmetic expressions involving integers and real numbers can be written in Prolog, however Prolog is not the most suitable language for writing and executing more complex procedures involving arrays and loops. There is no array structure in Prolog; equivalents can be created using lists or predicates with large numbers of arguments but these are less efficient, requiring more space and more computing time. It is desirable to write functions which make use of arrays in a procedural language rather than in Prolog. A number of tasks commonly performed during the course of evaluating a correlation involve the use of arrays, this may include the interpolation of a value from tables or graphs. Arrays are used extensively in the model for the local heat transfer due to a single impinging jet (Chapter 3). The tasks involving arrays have been written in C, and are closely coupled with the Prolog code.

It was found that, on a personal computer, the KBDS and the code used to model the single jet heat transfer could not reside together in the memory when implemented as separate but loosely-coupled processes. When closely-coupled to form a single process however they ran successfully.

The comparison of the keywords specified by the user to describe a heat transfer process with the keywords associated with each correlation in the database is performed frequently. Code written to perform this comparison was closely-coupled to avoid the performance loss associated with data transfer between loosely-coupled processes.

SD-Prolog (Quintec Systems Ltd. 1986 and 1987) variables are stored in data structures or referenced by pointers stored in data structures. Close-coupling requires the creation of an interface which enables the procedural language to obtain values from and return values to these structures. The basic data types transferred through this interface are integers, real numbers and character strings. A number of corrections and additions were made to the interface routines supplied with SD-Prolog, these were made to enable the transfer of the values of the 24 bit integers and the values of members of lists.

6.2 The interface to the database files

The Knowledge Based and database elements of a Knowledge Based Database system (or an Expert Database system) may be 'loosely-coupled', 'closely-coupled' or 'fully integrated' as discussed above. A Knowledge Based system may be loosely-coupled with a conventional database management system. Alternatively the knowledge based and database elements may be combined into one system by extending the capabilities of a database management system (DBMS) to include inference or by adding database query and management functions to a tool primarily for the development of knowledge based systems. On personal computers a closely-coupled system would form a single process. On a network a loosely-coupled system may generate queries to be processed by a separate DBMS. Research into providing a fully integrated system providing the capabilities of knowledge based systems and database management systems (referred to as a Deductive Database) is on-going, for example McLean and Weise (1991). The relative merits of these different approaches are being discussed in the literature, for example Kerschberg (1990).

As stated above loosely-coupled systems consist of separate processes which communicate with each other by passing data. Loosely-coupled systems are generally easier to create (partly because they may make use of available DBMS software), but if large quantities of data must be transferred or if control must pass frequently between the processes then close-coupling is to be preferred.[1] It would appear

---

[1] The combination of Knowledge Base function and database access function into one process is not a guarantee of acceptable performance. Early development of the KBDS was performed with an Expert System development tool, Leonardo (Creative Logic Ltd.). Details of each correlation were stored as a frame (Winston 1992) in a database internal to Leonardo. The control of the search of this small database used code written in a combination of the rule language and the procedural language which also form a part of the Leonardo system. The performance of this system was poor due to frequent disc accesses. The disc accesses were apparently due to inappropriately organised overlays (the storage of part of the data or program on disc which must be read into memory when required, replacing code which must be read onto the disc). The time required by the Leonardo system to

to be feasible to obtain the required records from the heat transfer correlations files by loose-coupling the knowledge based system to a separate DBMS. The following six step process illustrates the necessary procedure:

(i) The knowledge based system is supplied with keywords which describe the heat transfer process and as a result builds an SQL query of the form:

```
SELECT Publication code, Configuration
FROM Description
WHERE Keywords LIKE "%Keyword 1%"
AND Keywords LIKE "%Keyword 2%"
                  ⋮
AND Keywords LIKE "%Keyword n%"
```

(ii) The SQL query is submitted to the DBMS which selects a subset of Publication codes and Configurations from the correlation description table.

(iii) The subset of data is read into the knowledge base as Prolog predicates.

(iv) A second query to the DBMS then retrieves all the records

---

search and select correlations from this database was two orders of magnitude larger than the time required to perform a similar operation using Prolog (Quintec Systems Ltd.).

from the relational tables giving the references, the limits and the correlations where the publication code and the configuration matched one of the pairs successfully retrieved from the correlation description table. For example, in order to select the appropriate publication details for the subset of correlations obtained previously an SQL query of this form is necessary:

```
SELECT Publication code, Authors, Year
FROM Reference
WHERE Publication code = "Publication code 1"
OR Publication code = "Publication code 2"
                      ⋮
OR Publication code = "Publication code m"
```

(v) These sets of data would also be read into the knowledge base as Prolog predicates.

(vi) These predicates are now searched one record at a time using the back tracking mechanism within Prolog.

The procedure outlined suffers from three disadvantages. Firstly, while this procedure is adequate for the four tables making up the correlation database it will not suffice for the tables making up the keyword network used to aid access to the correlation database. The selection techniques provided by SQL would not enable useful sub-sets of these tables to be obtained. The data required from the network tables 'subset', 'excludes' and 'expands' depends on the keywords in the current partially completed description of the heat transfer process. Consequently either many queries to the DBMS will be required or the entire table must be retrieved and the table searched in Prolog. Secondly the selection of a record from the database tables 'diction' (Figure 15), 'eqn' (Figure 16) and 'non-dimensionals' (Figure 18) depend on the results of an analysis of one of the fields. This field must be broken into its component keywords and a test made to determine whether each of these keywords is included in the list of keywords built up by the user to describe the heat transfer process. This cannot be carried out using SQL so for this data the entire table would have to be read into predicates and searched by Prolog. The third disadvantage of loosely coupling to a DBMS is that a second process (the DBMS) must be loaded; the heat transfer correlation knowledge based system has been developed on a personal computer, the computer operating system does not support the running of more than one process at a time.

The KBDS has been developed as a closely-coupled system, the system has been developed in a tool suitable for the development of Knowledge Based systems (Prolog) which has been augmented with a number of database management functions appropriate to a system running on a personal computer. The method that has been adopted allows the records in each of the tables to be treated as Prolog predicates. This replaces the first five steps in the loosely-coupled procedure described. Each database record or predicate is considered one at a time using the normal Prolog backtracking mechanism. The disadvantages of the loosely-coupled method, discussed above, do not apply. However the closely-coupled method involves non-standard

Prolog. Functions must be written to provide random access to the database files and to make available a large cache to prevent frequent disk accesses. A closely-coupled interface which enables each record in a relational table to be treated as a predicate is provided in SD-Prolog and has been used in the development of the heat transfer correlation database.

6.3 The use of a relational database to store a program

The correlations form a program which is interpreted by the knowledge based system to arrive at a value for heat transfer. These programs vary in length from three lines to, currently, about 30 lines; examples of two correlations are given in Figure 12. The database format used by the SD Prolog database interface (Ashton Tate dBase iv 1988) required that the corresponding fields in each of the records should be of the same length. To store each of these programs in a single record would be very inefficient in the use of space as all the records would require a field length long enough for the longest program. The space saving measure adopted here has been to divide the correlation into separate lines or statements, each statement is put in a separate record which contains a field indicating the statement line number.

```
correlation( Publication code,
             Configuration,
             Line number,
             Correlation statement).
```

To evaluate the correlations all the records for a given correlation are retrieved from the database using the keys Publication code and Configuration. The retrieved records are then sorted using the line number. Finally each statement is tokenised and concatenated into a single list before parsing. The program is put into a single list rather than evaluated a statement at a time because some of the language structures, *while* for example, are spread over more than one line.

A database which allowed record fields of variable length would make the steps of collecting the appropriate records and sorting using the line number unnecessary. The database could then be simplified to the form

correlation(<u>Publication code,</u>
<u>Configuration,</u>
Correlation statement).

Database systems are being developed (Stonebraker and Rowe 1986) which will allow the storage of programs of varying length. The potential benefits of this technology to a system such as the correlation Knowledge Based Database System will be discussed in Chapter 7.

CHAPTER 7
DISCUSSION

7.1 The contribution made to the availability of correlation data

Based on a simple outline description of a system of impinging jets the KBDS will automatically select the most appropriate correlation data from a database. The KBDS has the capability to ensure that this data will be valid, in the sense that the key parameter values in the heat transfer problem described will lie within the range of the experimental values used to derive the correlation. If, however, the KBDS is unable to retrieve correlations exactly matching the prescribed geometry and flow then the KBDS will relax the strict selection criteria, automatically relaxing those criteria which are expected to have the least impact on the calculated heat transfer coefficients. The KBDS will then evaluate the selected correlations to give Nusselt numbers and heat transfer coefficients, enabling the user to compare the correlations and select the heat transfer coefficient most suited to his purpose. This simple and reliable method of obtaining heat transfer coefficients from correlations is in marked contrast with the effort that an engineer must apply at present, which is essentially to perform these tasks manually.

The database described here, which was created to demonstrate the function of the KBDS, has over 50 correlations in it even though it is confined to the heat transfer due to impinging jets. It is believed that the techniques developed can be applied to collections of heat transfer correlations obtained for other geometries (a conviction supported by the demonstration of the addition of a correlation for free convective heat transfer from a vertical plate given in Appendix B). The capability to incorporate more diverse heat transfer correlations means that the potential size of the database is very large.

It has been the author's experience that a manual search, or even a search assisted by currently available database systems containing text, for a correlation appropriate to a specific case requires the study of a large number of publications containing either inappropriate correlations or no correlation at all. This process is made more laborious by the fact that the information needed to check whether a particular correlation is appropriate is usually dispersed throughout the publication and is often not explicitly stated. The extraction of both the range of geometry and flow values over which the correlation is valid and the correlation itself is complicated by the fact that different terminology, symbols and units are used by different authors. The collation of this information and its presentation using a single consistent set of terms, notation and units significantly simplifies the task of obtaining an appropriate correlation; the incorporation of this information into a database with a front end which assists in the formulation of queries simplifies and speeds this task further.

A thorough search of the literature without the assistance of the sort offered by the KBDS would be so laborious that it is to be expected that an engineer would not look for a number of correlations so as to obtain a comparison of the heat transfer coefficients predicted by different correlations and instead to be content with extrapolating a single correlation with uncertain results. The assistance provided by the knowledge based database will facilitate the comparison of a number of correlations and so will tend to result in the selection of more accurate correlations and hence the production of better engineering designs.

Neither existing knowledge based systems nor existing database systems deal adequately with the selection of empirical correlations. Database systems have been created which contain the text of documents (either the full text or an abstract). Work reported by other authors (Fenichel 1981) suggested that the success rate for the retrieval of relevant documents from databases of this type was poor, these conclusions were borne out by the authors own experience (Section 1.1). In these systems queries must be formulated manually; this is made more difficult by the use of different terminology by different authors, which requires the construction of multiple queries if all the relevant references are to be selected, and the expression of significant data in the form of diagrams and photographs which are not well handled by current database systems. An expert interface which will ensure the selection of all the relevant information is required.

Work has been carried out with the aim of developing expert interfaces to full text databases. The combination of AI and Expert System techniques with Data Base Management Systems is a subject of current research interest; the development of these techniques for the purposes of the correlation database is discussed in the latter sections of this chapter. The numerical values of parameters are of great importance in the selection of engineering data including correlations and are, of course, essential to the evaluation of correlations. It follows from this that the construction of an interface to a database of correlation information requires a number of features not found in text oriented databases and hence not addressed by previous work on expert database interfaces.

The knowledge based selection procedure is able to obtain correlations even when the prescribed geometry and flow do not precisely match any of the data available. For example the judgment of when it is acceptable to extrapolate a correlation or to use a correlation originally obtained from a different geometry requires a detailed knowledge of the heat transfer process in question. This knowledge can be represented and used in the KBDS, thus the KBDS has the capability of providing this expertise to engineers who do not have this specialised knowledge. This aspect of the knowledge based front end has the potential to be very useful as engineering data is often expressed as empirical correlations, and in many cases none of these correlations will apply precisely to the specified geometry and flow.

When new empirical correlations are added to the system it is only necessary to state the range of each of the non-dimensional groups justified by the work on which the correlation is based. A set of non-dimensional groups sufficient to define the range of parameter values over which the correlations can be used is supplied by the knowledge base. Rules in the knowledge base can be used to represent the conditions under which correlations can be extrapolated with only a given loss in the expected accuracy. Where a correlation can in fact be extrapolated beyond the stated ranges of the non-dimensional parameters then this extrapolation is automatically permitted by the knowledge base. It is not necessary for a user entering a new correlation to have a full understanding of the jet impingement literature and, using this, to specify explicitly the complete range of parameters over which it should be possible to use the correlation. Similarly the knowledge base can select and apply rules to indicate where a correlation, derived for a system with one set of physical features, can be used to derive heat transfer coefficients for qualitatively different configurations, ie those with different physical features. Thus the entry of a new correlation to the database requires only a qualitative description of the geometry and flow, numerical values for the ranges of each of the significant non-dimensional parameters and a statement of the correlation. The determination of whether or not the correlation can be applied to a particular impinging jet system is performed by the KBDS.

The graphical correlation developed for heat transfer due to a single circular impinging jet (Section 3.1) is appropriate for the extrapolation of empirical data over a wide range of Reynolds numbers ($5000 \leq Re_n \leq 124000$). The method used to extrapolate from empirical results obtained at one value of Reynolds number to provide local heat transfer coefficients appropriate to another value of Reynolds number is described in Appendix F. Use of this extrapolation method enables a larger number of correlations to be applied to a particular problem. This extrapolation is especially useful as many sets of data are available from experiments aimed at investigating the effect on the heat transfer coefficients of nozzle-to-plate spacing at a constant nozzle exit Reynolds number. Alternative interpolation methods (see Appendix D for an example) require sets of empirical results obtained at different values of Reynolds number.

7.2 Initial experience of use in a Design Environment

The jet impingement Knowledge Based Database System has been made available within a design environment. A controlled and quantified study of the acceptability and use of the system has not been undertaken. However the following anecdotal evidence of the systems strengths and weaknesses is offered:

(i) Engineers working in the heat transfer field have recognised the requirement for both qualitative and quantitative aspects to be considered in selecting heat transfer correlations. The system has been successfully used to select correlations from the database for design purposes.

(ii) The value of the correlation KBDS as a front-end to a database which assists in the formulation of a suitable description to identify and select correlations has been appreciated. Following from this the front-end has been adapted to select other information where qualitative and quantitative components are required to adequately specify the information, Monico and Chew (1992).

(iii) The design areas into which the system has been introduced are specialised. Only a small number of correlations are used within each of these areas. Hence one of the systems strengths, the ability to make accurate selections from a wide range of correlations, is not so greatly required. It is possible that the system may be of greater benefit within a consultancy where a wider variety of heat transfer geometries are encountered. A second potential application is as a means for standards organisations to distribute correlation information to a large number of clients who collectively employ a wide range of heat transfer geometries.

(iv) Engineers have been reluctant to enter new correlation information into the database. Possible reasons include the lack of immediate benefit to the individual designer and the pressure of other more urgent work.

(v) It is anticipated that in order to maintain the standard of correlations in the database the permission to add new correlation data must be restricted to suitably experienced personnel. This can be achieved using the facilities available in a networked operating system.

(vi) An additional benefit of a networked environment is that additional correlations will be made available to a wider number of engineers, so increasing the potential benefit of adding new correlations.

7.3 The use of Knowledge Based System methods to enhance the selection of empirical data from a database

A semantic network to facilitate the description of heat transfer processes

The identification, by the KBDS, of a particular correlation as being appropriate to the heat transfer problem described by the user is based on a comparison of descriptive keywords and key parameter values. A semantic network represents knowledge of the physical features of impinging jet systems which have a significant effect on the heat transfer coefficient. This network controls both the keywords and the parameters that can be combined to form a problem description. Control is exerted both when obtaining data from the database and when forming the description of an experimental test while adding new data to the database. Thus the network also represents knowledge about the contents of the database. The network ensures that the combination of keywords constructed is sensible by offering only those keywords which can be meaningfully combined with the previously specified keywords. The network al[illegible]s to ensure that the most appropriate keywords are selected by presenting a short piece of text giving a definition of the meaning of each keyword to the user; in this the KBDS embodies some of the ideas of hypertext systems. To prevent the user being overwhelmed with the number of keywords available the user is initially given a choice of a selection of very general keywords. By following a hierarchy built into the network the keywords offered become progressively more specific.

Through the network a given set of keywords implies a particular set of geometric and flow parameters. When querying the database the user supplies values for these parameters. When adding data to the database a range of acceptable values may be assigned to non-dimensional groups formed from these parameters. The network is also used to represent a list of non-dimensional expressions suitable for each geometry. This list is presented to the user when adding a new correlation to the database. Thus the knowledge of which non-dimensional groups are significant can be communicated to a less experienced engineer.

The values of the parameters used to provide dimensions for the geometry of the heat transfer surface and to characterise the flow field are required to select and evaluate a correlation. The parameter values are commonly inter-dependant. The procedures or equations relating these parameters can be expressed as rules. Using these rules parameter values can be inferred from previously specified values, as a result user input is minimised and the potential for error is reduced. These procedures are also incorporated into the semantic network, thus the network contains knowledge on how to derive parts of the description of the heat transfer geometry and flow conditions.

Rules expressing the conditions under which a correlation may be used

Associated with each correlation in the database is a range for each of the most significant non-dimensional groups. Each range represents the limits of the variation in these parameters covered by the experimental observations used in deriving the correlation. This information is held in records in the database, the KBDS interprets these records as rules which can be used to decide whether it is valid to use the correlation in a particular set of circumstances.

The modification of a query to the database using domain knowledge

The KBDS is able to automatically relax the selection criteria applied to each correlation in the database. This is of use in the event that no correlations, or too few correlations, exist in the database which fully satisfy the description entered as a query by the user. Backward chaining rules in conjunction with the keyword network are used to control the relaxation of the selection criteria. The network is used by the KBDS to decide which rules are relevant to the current query. The rules embody knowledge of the influence of given geometric or flow features on the heat transfer coefficients under a range of specified conditions. Under circumstances where the influence of a feature is known to be small it can be omitted from the selection criteria, either by not including a reference to the feature in the database query or by increasing the permissible range of the corresponding parameter values.

7.4 Developments of existing Knowledge Based System methods required for the Knowledge Based Database System

Symbolic AI and Expert Systems methods widely employed in KBS have been used extensively in the creation of the correlation KBDS. In order to achieve the goals set for the correlation KBDS some additions and modifications to the existing methods were required.

The keyword network employed by the correlation KBDS (which contains information in addition to the keywords) differs in significant respects to the keyword structures described by Humphrey (1987), Smith *et al* (1989) and Parsaye *et al* (1989). These differences are:

> The keyword network described here and the methods developed by Humphrey, by Smith *et al* and by Parsaye *et al* each represent information about keywords. This information is used to assist the user in composing a qualitative description of an entry in the database, this description is used either to index a new entry to the database or as a query which can be used to retrieve database entries. The keyword network employed in the correlation database is used to represent the additional information needed to construct a quantitative description. The need for a quantitative description in addition to a qualitative description is

determined by the subject matter of the database, numerical quantities must be derived in order to make sufficiently specific selections from the database and to evaluate the correlations retrieved.

In each of the three methods described by Humphrey, Smith *et al* and Parsaye *et al* keywords are associated with slots in frames. Each slot is considered to represent a different aspect of the system, keywords are chosen to specify a particular quality for each of these aspects. For example a frame may be used to represent a type of disease (Humphrey), slots in this frame may include the location on the body of the disease and the medical intervention prescribed. By the specification of appropriate keywords a description of a document is constructed. The keyword network used in the correlation KBDS, which consists of nodes and links, is implemented as a set of predicates in the logic programming language Prolog. Logic is particularly suitable for representing this type of structure. The present work demonstrates that the knowledge about keywords required to assist in the formulation of database queries can be represented in a logic programming language without the need to implement a frame environment.

The keyword network is employed in the correlation KBDS to construct valid queries. It is also used to ensure that the keywords and other information selected to be associated with a new entry in the correlation database are consistent with the existing entries. The indexing aid described by Humphrey ascribes keywords to frames in order to characterise a new database entry. The systems described by Smith *et al* and Parsaye *et al* are used to construct queries and are intended only for the retrieval of information.

The keyword network can be extended by a user to enable the addition of correlation information in new subject areas. The network is expressed as predicates, each clause representing two nodes and the link between them. New clauses which extend the network incrementally can be added to these predicates. Because the structure of logical clauses are the same as relational records these new clauses can be simply added to a database. The hierarchies of frames and keywords on which the systems of Humphrey, Parsaye *et al* and Smith *et al* are based were all created manually. The extension of these hierarchies by a user engaged in the addition of new information to the database is not addressed.

The keyword network is used to determine whether in the formulation of a query a given pair of keywords are incompatible. The use of a single hierarchy will prevent the entry of unacceptable combinations of keywords by only allowing keywords to be taken from a single line of descent through the hierarchy, ie by following only one link from each node. However to satisfactorily describe more than one feature of a document either multiple hierarchies must be used or it must be possible to follow more than one link from each node in the hierarchy. If either of these methods are adopted then some mechanism must be implemented to prevent unacceptable combinations of keywords from separate hierarchies or as the result of following more than one link from a node. The frame methods described do not appear to be able to prevent the entry of incompatible keywords into slots in the same frame. Though it appears that the system described by Humphrey could be modified to perform this task. Humphrey's system is built using frames in which procedures can be made to act in response to the values entered in a slot. Procedures attached to one slot could be used to modify the list of keywords available for use in another slot, eliminating any keywords which should not be combined with the keyword entered into the first slot.

The keyword network can be employed to compose a more precise description of a particular aspect of the geometry or the flow conditions than the slot filling methods described by Smith *et al* and Parsaye *et al*. This is because these methods allow only one keyword to be assigned to each slot. The method described by Humphrey enables more than one keyword to be used to describe each aspect by generating additional slots when certain keyword slot combinations are given.

The keyword network is also used to perform a number of additional functions, these include using the network as a basis for a hypertext-like interface which informs the used of the meaning of each keyword. The methods described by Humphrey, Smith *et al* and Parsaye *et al* require that the user is familiar with the terminology used in the database. However the keyword frame described by Smith *et al* (1987) includes a short definition of the keyword.

The searching methods of Humphrey and of Parsaye *et al* both require a description of each document in the form of one or more frames containing a set of keywords to be associated with to each document in the database. Suitable keywords are added to the slots in a frame associated with each document, subsequent searches are then based on the contents of these frames. Gauch (1990) argues that the requirement for these keywords, which in the methods of both Humphrey and Parsaye *et al* must be added manually, is a significant shortcoming. The methods of Gauch and of Smith *et al* (1989) do not require any additions to the database itself, obviously a considerable saving in the creation of the database. Instead a query is built up with the benefit of either general knowledge on how to formulate queries (Gauch) or of domain knowledge (Smith *et al*). This query, which relies on the presence of known words or combinations of words in the text, is applied to the text of the document itself. An alternative method is to obtain a description of the document from an analysis of the natural language text. This is the method employed by Lucarella and Morara (1991) in a system which also interpreted natural language queries. These techniques would not be adequate for the retrieval of empirical data such as heat transfer correlations from a database. The reason being that information which is significant in the prediction of heat transfer rates is commonly presented in the form of diagrams in the jet impingement literature and not in the form of text. It is rare that all the necessary information is presented explicitly in the text. For example the details of the numbers and arrangement of nozzles in jet arrays are often presented in photographs or diagrams as are the dimensions of parts of the experimental apparatus; this is typical of publications giving empirical data. The use of numerical values in checking that the range of the specified non-dimensional groups is appropriate forms a vital part of the search process. Hence there is insufficient information in the text of these documents to enable a precise discrimination between data that is and is not relevant. It follows that a description must be associated with each document to enable the selection of sufficiently specific documents. Further complications in the interpretation of the text of a document are differences in terminology and the use of symbols and abbreviations to represent more than one physical quantity. Gauch reported that the system developed results in a significant reduction in the time required to formulate queries but that there was no significant change in the effectiveness of the queries (ie in the proportion of the total number of relevant documents retrieved). This finding tends to support the argument that a controlled description of some form must be associated with each document in a database of correlation information if a high proportion of the relevant documents must be retrieved.

The broadening of a query is achieved in two ways. The first method is based on knowledge of the relative importance of parameters on the heat transfer rate. This will be used in selecting the parameters or combinations of parameters that need not strictly satisfy their limits. Knowledge based control of the search in this way is only applicable to systems where at least part of the selection criteria is numeric; the knowledge based database interfaces developed by Gauch and Smith *et al* perform selection based on text only. The second method of broadening a query is similar to the techniques described by Smith *et al* and by Gauch. Smith *et al* outlined a system in which a keyword is either removed from a slot or changed to a less specific keyword from higher in the keyword hierarchy. Each slot is tried in turn and the results displayed to the user. The system developed by Gauch modifies a Boolean database query either by substituting keywords in the query in a similar manner to Smith *et al* or by modifying the Boolean operators. The correlation database offers an improvement on these methods in that it uses the knowledge of the relative importance of each keyword in selecting which keyword to change; this enables the search to be broadened whilst causing the smallest likely reduction in the accuracy of the answer obtained.

The expert searching systems developed or proposed previously have concentrated on queries formed entirely of words; whereas the jet impingement KBDS requires a combination of words and a corresponding set of numerical values to give a sufficiently specific description of a heat transfer process. Once the user has formed the keyword part of the database query, with the aid of the KBDS and the network, the network is used to automatically generate the list of parameters for which numerical values are required. These values are also used in the database query.

For many of the input parameters mathematical expressions can be obtained which relate the value of the parameter to the values of other parameters. Rules can be derived from these expressions. Inference using these rules allows some of the parameter values to be derived from previous input so minimising the input required from the user. It was discovered that conventional forward and backward chaining inference is inadequate for this task. Many of the parameter values can be derived from more than one equation giving rise to the possibility of repeated evaluation of the same parameter using different rules. In addition the equations also allow the formulation of circular arguments, these must be avoided to prevent non-terminating loops in backward chaining. An inference technique was developed specifically to avoid the problems of non-terminating loops and repeated evaluation of parameter values.

The activation or use of groups of rules only when they become relevant is an established technique in symbolic Artificial Intelligence work. This concept is particularly useful in the provision of a front end to a database. This is because a database will usually contain a wide variety of information enabling the rules to be divided into sets where each set of rules applies to a different subset of the database. If the relevant set of rules can be identified (by a mechanism such as the keyword network) then only a portion of the total number of rules need be considered. The use of the concept of sets of rules for this purpose does not appear to have been attempted before. This feature is used to control the rules used to minimise the number of parameter values that must be entered by the user; only those rules that are required are retrieved from the database.

7.5 Benefits of using an external database in a Knowledge Based System

In the addition of data to a program or of rules to a KBS it is important to be able to predict the effect that the new information will have on the performance of the system. Normally considerable effort is required to understand and predict the behaviour of a modified program. This problem has also been identified as applying to Knowledge Based Systems (for example Watson et al 1992). One of the strengths of a database is that the form and range of values of the data to be added can be strictly controlled, so ensuring that the program can interpret and process the data correctly. These measures increase the confidence that the system behaviour will be maintained. If rules are added to a KBS by entering the new rules into a database where the form of the rule and its possible actions can be controlled then similar benefits can be obtained. When a new correlation is added to the jet impingement correlation database the KBDS enforces constraints on the content of the new data. These constraints ensure that it will be possible for the KBDS to interpret the new information. In addition, because each correlation in the database is independent, an erroneous correlation, whilst misleading in itself, will have no effect on the selection and evaluation of the other correlations.

The difficulty and expense of obtaining new information for

Knowledge Based Systems is widely recognised. It is expected that the use of a database to contain information and a KBS to verify the new information should allow this information to be augmented incrementally by a wider group of people. A potential benefit is that the range and number of correlations in the database can be increased without the need to train staff specifically for the task of interpreting heat transfer publications and of adding information to the database.

The database used by the jet impingement correlation KBDS has a number of limitations which further development would be expected to eliminate.

(i) The arithmetic statements required to derive the heat transfer coefficients are expressed as text. Each statement is stored in a fixed length array in the database. The statements vary considerably in length and so some storage space is wasted.

(ii) The text is interpreted by the KBDS. While the interpreter is suitable for the evaluation of currently available correlations it is expected that the interpreter will require further development as correlations increase in sophistication. For example procedures which include arrays are performed by code written in a conventional procedural language (Kernigham and Ritchie 1978). These procedures are called as and when required by the interpreter. However the procedures form part of the KBDS and it is not possible for users of the database to add to these procedures. The use of one of the widely available compiled languages to write the correlations would overcome this problem provided that execution of these procedures can be initiated by the KBDS.

A database system such as Postgres (Stonebraker and Rowe 1986), designed to store arrays of varying length and to store executable procedures written in a compiled language could form the basis of a solution to these limitations. The use of a language separate from the KBDS to write the correlation evaluation procedures would however require some further development of the KBDS. In order to maintain the coherence between the KBDS and these procedures the variables used by the procedures must be controlled. For a given heat transfer description only a certain set of parameters will be appropriate. The knowledge base makes the values of these parameters available to the correlation evaluation procedures. While it is not possible to ensure the correct use of these variables it should be possible to ensure that the correct values are made available to the procedures. One potential method would be for the program which is used to add new information to the correlation database to generate the source code to be used by these functions to obtain the values for the necessary variables.

7.6 The use of logic in developing the knowledge based database system

Prolog as conventionally implemented (Clocksin and Mellish 1984) does not support the data management functions offered by relational database management systems, but a useful application can be created using only a small set of the DBMS functions. The functions required by the jet impingement correlation database system were the ability to read records from the database files and to add new records to these files. An implementation of Prolog, Quintec systems Ltd. (1986;1987), offering these functions was employed for this work.[1]

A key feature which distinguishes deductive databases (Lloyd 1983) and other enhancements to database query languages (Delacombre 1988 and Stonebraker and Rowe 1986) from current DBMS is the ability to use a rule or rules to infer the answer to a query. In these systems answers may be implicit in the data and rules and so the information need not be stored explicitly. The jet impingement knowledge based database also does not explicitly store the answer to a query. The selection of a correlation for evaluation requires the use of a number of rules. The use of a semantic network to assist in the specification of a database query appears to be particularly compatible with a deductive database. Deductive databases use logic to express both queries and integrity constraints. The keyword network used by the jet impingement correlation KBDS is written as a logic

---

[1] The performance of this interface was found to be satisfactory for reading information from the database files whilst using the small example jet impingement heat transfer correlation database. Unfortunately a fault, apparently in the database interface, resulted in the corrupting of new records appended to the database files. It was found necessary to use a further program to transfer the information, previously validated by the jet impingement correlation KBDS update program, from temporary files to the database files.

program and so it should be possible for the network and associated integrity constraints to be fully integrated with a deductive database. This may reduce the requirement for frequent transfer of control between the KBS and the DBMS during retrieval of information from the keywork network, identifed in Chapter 6 as a potential problem of loosley-coupled systems.

7.7 Modification of the database contents

The work carried out on the jet impingement correlation KBDS addresses the problems associated with obtaining correlation information from the database and of adding new information to the database. This new information may apply to the heat transfer obtained from a different geometry to those already stored in the database. To enable this new information to be incorporated into the database the keyword network, which is used to control the reading of existing data and the addition of new data, must be extended. The KBDS enables the network to be extended, the extra information is stored within the database itself. However the KBDS does not enable changes to be made to the existing information in the database. The problems involved in making alterations to the keyword network would require particular attention. If the keyword network is to be changed rather than extended then this would require complex changes to be propagated throughout the network and the correlation information. These changes would be required to maintain the consistency of all records, both in the correlation database and in the keyword network, to which the affected records in the network refer. For example the insertion of a new node within an existing branch of the keyword network. This would require a new node and a new keyword and associated description. This addition would also require the modification of the links between affected nodes and the addition of the new keyword to affected qualitative descriptions in four database tables. These are the correlations, the parameter names, the inter-relations and the recommended non-dimensional expressions. It is possible that these modifications could be performed automatically provided the appropriated logical rules were developed. A second, more problematic, example is the deletion of a keyword from the network. If a keyword were to be removed this would require either the reassignment of parameters names and inter-relations to new nodes within the network or the removal of correlations using these parameters from the database. The removal of correlations from the database is obviously undesirable.

Chapter 8
Conclusions

8.1 Improvements in the availability of correlation data obtained using Knowledge Based System techniques

A knowledge based database system has been developed which considerably reduces the effort required to obtain correlation data. Based on a simple description of a heat transfer problem outlining the important geometric and flow features and corresponding key dimensions the system automatically retrieves all the relevant and valid correlations from a database. The system then evaluates each of these correlations to give Nusselt number and heat transfer coefficient values. The system performs satisfactorily for a variety of empirical data from the jet impingement heat transfer literature. It is anticipated that the methods developed could be applied to other fields where empirical data is also presented in the form of correlations.

Correlations and empirical data have been extracted from the literature. In the case of the empirical data this has involved digitizing graphical data presented in the literature. The correlations and empirical data have been transformed to use a consistent terminology and have been entered into the database.

Little prior knowledge of the specialised terminology used by the system is required as the knowledge of how to construct a query is provided by the system. Thus overcoming limitations identified in currently available systems. A knowledge based front end to the database guides the user in forming a valid description of a heat transfer problem, this description in effect forms a database query. The user interface, based on hypertext systems provides information on the flow geometry implied by each keyword. An semantic network of keywords (which represent the geometrical and flow features), parameters (which are used to provide values for the significant dimensions), and associated data provides a suitable basis for this interface.

The system may be used to automatically modify the search of the database to improve the performance of the search. This facility provides a useful method of obtaining acceptable correlations when either none or too few correlations are retrieved which satisfy the user's description of the heat transfer geometry and flow regime. Knowledge of jet impingement heat transfer, expressed as rules, is used to adjust either the limits of validity associated with the correlations in the database or the description given by the user. These adjustments are made so as to make available one or more correlations which are expected to provide the best estimate of the heat transfer coefficient for the geometry and flow regime originally described. The selection criteria are adjusted progressively and an indication of the anticipated degree of approximation is provided.

It was found that the text taken from the original publications did not provide a suitable basis for selection from a database. Instead it was necessary to create and associate a set of terms with each document. The search was based on these terms. Much data, significant in the determination of heat transfer coefficients, is not mentioned explicitly in the text and is instead given in diagrams, sketches, photographs and tables. Hence the text alone would not provide sufficient information on the document contents to enable a search to discriminate between publications with and without relevant data. Data in these forms is not accessible to current searching techniques.

A method was found to represent each correlation and the necessary information associated with it in a relational database. The possibility of storing this information in a database means that the task of extending the knowledge in the system is considerably simplified.

The system assists in the addition of new correlation information to the database and in the extension of the keyword network to accomodate the addition of correlations for new flow geometries. In adding new data constraints are enforced which ensure that, at a later date, the system will be able to retrieve the correlation, to check the validity of the correlation and then to evaluate the correlation. The knowledge used by the database in query formulation and in the addition of new correlations is represented in a semantic network which is also stored in a relational database. The system enforces the constraints required to enable this network to be extended, thus enabling correlations for geometries and flow conditions not previously covered by the database to be added.

8.2 Developments of existing Knowledge Based System techniques

The KBDS uses a different technique from previous systems developed to assist in the formulation of database queries. The knowledge used is represented in a semantic network which introduces new information to the execution of the task of query formulation. Previous structures used for this purpose have been either simple hierarchies of keywords or structures comprising frames, the slots of which are filled by keywords taken from hierarchies. Six different meanings are associated with the links used in the semantic network. The network is constructed around keyword nodes linked to form a hierarchy. This hierarchy identifies keywords each of which may be used to contribute to a more specific descriptions of a heat transfer problem. A second link enables the KBDS to present groups of these keywords together as alternatives (presenting the keywords together making the interface easier to comprehend) and to ensure that these keywords are not combined in a single query (preventing the creation of meaningless queries). The network also includes a link from each keyword node to the keyword and to a body of text which enables the KBDS to present text intended to make the meaning of each keyword more apparent. The fourth type of link enables the KBDS to automatically select the parameters which are appropriate to defining the dimensions of the heat transfer problem. Previous Knowledge Based Systems developed to assist in the creation of database queries have based selection from the database purely on the presence or absence of words in a body of text. A qualitative description of this type does not allow a sufficiently specific description of a geometry and flow to enable the most appropriate correlations to be selected; instead a qualitative description, expressed using keywords, must be combined with numerical values for the significant parameters. A further type of link is used to identify non-dimensional groups, built-up from these parameters, the values of which are significant in determining Nusselt number. These are employed in the addition of new correlations to the database to help ensure that an adequate description of the conditions to which the correlation applies is given. Finally a link is used to retrieve equations inter-relating these parameters. These equations are used to minimise the input required from the user by deriving parameter values from previously specified values wherever possible.

It has been demonstrated that the semantic net used to assist in the formulation of queries can also used to guide the user in the creation of a description of the geometry and flow regime to which a new correlation is appropriate. The new correlations can then be added to the database.

The arithmetic equations giving the relationships between the values of the different parameters are converted by the KBDS into rules. An inference strategy was developed to enable these rules to be used which avoids non-terminating loops and repeated evaluation of parameter values.

The network is also used to identify sets of rules which are relevant to the current query. This prevents the unnecessary testing of the conditions of rules and the firing of rules which do not apply to this query. This is significant as a system of the form of the correlation database may be expanded to contain a large number of entries only a few of which are relevant to the current query.

8.3 Jet Impingement Heat Transfer Correlations

A critical review of the literature on the heat transfer due to the impingement of circular air jets has been performed. This review considers the impingement of a single jet on a flat plate, a single row of jets impinging on a flat plate and on a concave surface and in-line and staggered arrays of jets impinging on a flat surface. The effects of cross-flow velocity and ambient temperature relative to the jet, nozzle geometry and turbulence generation are also considered. This review forms the basis of the knowledge embodied in the semantic network and in the correlations contained in the database.

The review of the literature for the impingement of a single circular jet revealed that most of the heat transfer is presented in graphical form although a few correlations exist. The empirical data considered in the review confirms that the Nusselt number is a function of Reynolds number, nozzle-to-plate spacing and radial displacement from the stagnation point. The nozzle geometry, confinement of flow in the impingement region and turbulence generation upstream of the nozzle are also significant. However the exponent of Reynolds number is not a constant, the exponent depends on the nozzle-to-plate spacing and on the radial displacement from the stagnation point. This variation in the exponent correspondes to the presence of laminar flow in the thermal boundary layer being restricted to the region local to the stagnation point and occuring only at low nozzle-to-plate spacings. At larger radial distances from the stagnation point and at larger nozzle-to-plate spacings the empirical evidence suggests that the boundary layer will contain turbulent flow. Existing correlations assume a constant exponent of Reynolds number. Graphical correlations have been obtained for the local heat transfer due to a single impinging jet, these correlations have the form

$$\frac{Nu}{Re^{f(x/D,z/D)}} = f_1(x/D,z/D)$$

where

$$f(x/D,z/D) = 0.82 - \frac{0.32}{\left(1 + A\left(\frac{x}{D}\right)^k + B\left(\frac{x}{D}\right)^l\right)\left(1 + C\left(\frac{z}{D}\right)^m + D\left(\frac{z}{D}\right)^n\right)}$$

and $A = -1.95$, $B = 2.23$, $C = -0.21$, $D = 0.21$, $k = 1.8$, $l = 2$, $m = 1.25$ and $n = 1.5$. This correlation gives the exponent of Reynolds number as a function of the nozzle-to-plate spacing and the radial displacement from the stagnation point. The correlation is derived for $z/D = 1.2$, 5 and 10, $5000 \le Re \le 124\ 000$ and $x/D \le 4$. A comparison with empirical data demonstrated that approximate values of heat transfer coefficient in the wall jet can be obtained by extrapolation ot the local value at $x/D \ge 4.5$ using the relationship

$$Nu \, \alpha \, .(x/D)^{-1}.$$

Evidence obtained in the review of heat transfer due to a row of jets or due to an array of jets suggests that the distribution of heat transfer coefficients due to multiple jets will be considerably more complex than the distribution due to a single jet. The flow field associated with multiple impinging jets is expected to be more complex than for a single jet. Empirical data discussed in the review showed that the interaction between the wall jets from neighbouring jets results in local peaks in the distribution of heat transfer coefficient. In an array of jets the air from each successive row of jets results in the build-up of a cross-flow. Both the mean velocity of the cross-flow and the velocity distribution of this flow have been shown to have a significant effect on the heat transfer coefficients due to the impinging jets. Hence the derivation of a single analytical correlation for the distribution of local heat transfer coefficient will present considerable difficulties.

Publications considered in the review demonstrated that the effect of mixing between jets at non-ambient temperature and the ambient air must be considered in the derivation of surface heat flux. Local values for the heat transfer coefficient and the non-dimensional parameters jet effectiveness and recovery factor are required. The requirement for local rather than area mean values for these three parameters will entail either correlation expressions or techniques which are considerably more sophisticated than hitherto or the use of graphical data or a combination of expressions and graphical data. This last approach was adopted in deriving a correlation for the heat transfer due to a single circular impinging jet (Jambunathan *et al* 1992). The wide availability of computers should enable more complex expressions and graphical data to be used conveniently in evaluating local heat transfer rates.

CHAPTER 9

FURTHER WORK

9.1 Developments to Encourage the Use of the Correlation Database Within a Design Environment

The following developments of the correlation database, which could be achieved using currently available computing techniques, might result in improved acceptance and increased use of the correlation database within a design environment. The effect of these developments on the use of the system would need to be assessed by user trials.

(i) The ability to display graphical images taken from the original publications. These would include the correlation expressions, figures showing the experimental apparatus and heat transfer results given in graphical form. It is envisaged that once a correlation was selected the user would be given the opportunity to display the images associated with the correlation.

(ii) Object oriented methods may provide an improved method of correlation evaluation. At present, in order to provide heat transfer coefficient values, a procedure must be supplied in which the correlation expressions are evaluated. The correlation expression or heat transfer data should be incorporated in the appropriate object for evaluation. For example an expression giving an area mean heat transfer coefficient would be evaluated within a different object from data giving the distribution of local heat transfer coefficient. Using this technique only the expression or data would be provided by the user while entering a new correlation into the database. The method of evaluation would be selected and provided by the KBDS.

(iii) Systems exist (for example MathSoft Inc. 1993) which, by use of a suitable graphical user interface, are able to evaluate expressions specified in conventional mathematical form. Thus correlations would be expressed in a familiar way rather than as Fortran-like procedures.

(iv) Suitable modifications to the front-end would enable parametric studies, common in design, to be undertaken and input errors to be corrected. In the present version of the program once the parameter values have been specified they cannot be changed. The use of a dependency network, Stallman and Sussman (1977), would enable individual parameter values to be altered and the changes required by the parameter inter-relations satisfied by re-evaluation of the appropriate equations.

9.2 The Human Computer Interface

An alternative method of selecting the keywords in which the keywords are displayed in a graphical representation of the network could be developed. In addition it may be feasible to provide feedback to the user which would confirm that the description supplied by the user corresponds to the intended heat transfer geometry. The aim of these changes would be to make the database interface more simple to understand and the information presented to the user more easy to assimilate. A study would be required to determine whether these developments were beneficial. The visual presentation of geometry may be found to increase the scope of the database to include information such a stress concentration factors (Peterson 1974) and fluid flow pressure losses (Miller 1978) which are dependent on detailed geometric configuration.

The use of the jet impingement database can be thought of as having four stages:

(i) the description of a heat transfer process using keywords,

(ii) the specification of parameter values,

(iii) selecting and checking the validity of the correlations,

(iv) evaluating the selected correlations.

Of these stages the first two require the most input from the user and so the interface at these points should receive the most attention.

The selection of keywords is performed through a ‘hypertext’ like interface (Appendix A Figures A-2 and A-4 to A-7). Alternative keywords are presented along with text explaining the meaning of each keyword, when one of these keywords is selected the display is changed to show a new set of alternative keywords, the keywords shown will depend on which keywords were selected previously. A 'hypertext' interface presents the user with, potentially, a large quantity of text; the assimilation of this text may discourage some users. An option for the interface design is to present the keyword hierarchy graphically (along the lines shown in Appendix A, Figure A-3) and to enable the keywords to be selected from this display, probably using a pointing device (eg mouse). The display could be made to indicate which keywords might be selected at any one time, the keywords indicated as being available would change as keywords are selected. In comparison to the 'hypertext' interface using this method would result in less text being presented to the user but no explanation of the meaning of each keyword would be immediately available. A number of studies have been carried out into methods of organising terms, such as keywords, on which database searches are based in order to facilitate access to databases. Studies include those of Gauch (1990), Humphrey (1987), Parsaye et al (1989) and Smith et al (1987 and 1989), but no work seems to have been undertaken into the best way of presenting this information.

At present there is little feedback to the user to confirm that the keywords chosen and the dimensions that are given result in the specification of the heat transfer process that is intended. The information that is specified as input to the correlation database describes a physical object rather than an abstract idea, this should make the problem of conveying this information back to the user easier. One possible way is to present a graphical display of the geometry implied by the users description, this could be schematic, confirming the keywords, or dimensioned and to scale, confirming both the keywords and the parameter values. The feasibility and benefits of such a system do not appear to have been investigated.

Values of stress concentration factors (Peterson) and fluid flow pressure losses (Miller) are presented with the aid of diagrams. Concentration factors and pressure losses are dependent on detailed geometric configuration. The description of this geometry requires numerical values, hence the methods developed for the Jet Impingement database are relevant. It is expected that use of appropriate graphical images to present geometrical information and to request appropriate dimensions and flow properties from the use would bring significant benefits.

9.3 The Automatic Acquisition of New Search Control Rules

The KBDS contains an integrated knowledge base and database. The knowledge that the system holds about the correlations in the database consists of:

(i) Knowledge used for guiding the user in the formulation of a description of the heat transfer process.

(ii) Knowledge used to modify a search.

The development of methods which enable the knowledge used in formulating a description to be extended has been discussed (Chapter 5). By using these methods the knowledge in the database about its own subject matter can be extended to cover new fields of heat transfer. At present however the rules used to modify the search are derived without computer assistance. This process is time consuming and requires considerable domain expertise. Two possibilities appear worthy of investigation:

(i) Assistance could be offered which would help ensure that new rules added to the knowledge base are compatible with the existing knowledge.

(ii) New rules could be derived automatically from empirical results.

A medical diagnosis system, TEIRESIAS, assists the domain expert in amending or augmenting the knowledge base by ensuring that new rules are compatible with existing rules. The strategy adopted, described by Davis (1985), relies on the ability of the system to explain the reasoning process to the physician; this is possible because the reasoning method, backward chaining inference, can be readily appreciated. The strategy consists of four steps:

(i) The physician presents a problem to the system for diagnosis.

(ii) The performance of the program is observed.

(iii) The explanation facility is used to monitor the reasoning process and to discover any faults in the rules.

(iv) If the reasoning is at fault, that is a rule is being used which gives an incorrect conclusion, then a 'knowledge acquisition' process can be initiated.

The new rules specified by the physician must be compatible with the existing rules, ie the rule conditions must be capable of matching the conclusions of existing rules and the new rules must draw conclusions which have meaning to the system. By examining its own rules TEIRESIAS forms models, which include typical examples of premises and conclusions, of the rules. In an iterative process with the physician new rules are developed which conform to this model. The action of the diagnosis system including the new rules is then tested by re-presenting the original problem for diagnosis.

An alternative strategy is to derive, or learn, new rules from examples. Buchanan et al (1976) described a system which derives rules to predict which bonds in a molecule will break during mass spectrometry. These rules are used in the prediction of molecular structure based on the significant features on a spectra. The rules were derived from the spectral data obtained from molecules of known structure. The rules derived using this strategy represent new knowledge, whereas in the strategy pioneered by Davis existing knowledge is transferred from the expert to the knowledge base.

The search modification rules currently in the knowledge base have been derived from a manual examination of empirical results. It has been observed that, for some geometries and flow conditions, the heat transfer coefficient varies by a small proportion over a significant range of one (or more) of the non-dimensional groups. It has also been observed that only small variations in the heat transfer coefficients may be obtained as a result of some change in the geometry. This information is used to modify the search process by allowing either extrapolation of correlations or generalisation of the description of the geometry and flow supplied by the user. An appreciation of the principles of the impinging jet process is used to generalise these rules. Components of the strategy developed by Davis appear to be applicable to the task of adding new search modification rules to the knowledge base. These include the requirement that the new rules must be compatible with existing rules. However the search-modification knowledge base is simple; at present the information used to modify the search process is captured in a single layer of rules. It follows that the benefits to be obtained will be comparatively small. However considerable benefit would be obtained from a system which derived search modification rules from the empirical results. The application of a strategy analogous to that developed by Buchanan et al appears to be appropriate. The examples for this rule learning strategy may be provided by empirical results retrieved from the correlation database. An examination of the empirical results would reveal where the value of the heat transfer coefficient is approximately the same over a useful range of one of the non-dimensional groups. A search modification rule can be derived from this information. It appears that a model of the heat transfer process would be required in order to generalise the search modification rules. For example it has been demonstrated experimentally that the effect of cross-flow on the heat transfer coefficient due to an impinging jet reduces as the ratio cross-flow velocity to jet nozzle velocity reduces (Bouchez and

Goldstein 1975 for example). If the total mass flow acting as a cross-flow to a row of jets is constant then as the jet-to-jet spacing is increased the cross-flow velocity will reduce. By applying these two simple rules an experimental result showing a small effect of cross-flow at a given jet-to-jet spacing can be generalised to all larger jet-to-jet spacings. From this it follows that the corresponding search modification rule can also be generalised. It has also been demonstrated that in staggered arrays the effect of cross-flow is more significant than in in-line arrays, hence a rule derived from an experimental result showing a small effect of cross-flow in an in-line array can be generalised to apply to staggered arrays. Causes and corresponding effects, such as these, form precedents (Winston 1992) which can be used in formulating new rules. It is to be expected that the model of the heat transfer process would need to represent these causes and effects.

To summarise the structure of the knowledge base used to modify search rules is simple and so the benefit to be obtained from systems which ensure that rules to be added are compatible with the existing rules may be small. In contrast the ability to derive and generalise rules directly from the empirical data which may then be used to improve the performance of the database search is very attractive.

9.4 The automatic design of a system of impinging jets

The automated design of a system of impinging jets would probably fall into two parts. First the selection of a feasible design option or options, for example a choice between a single jet, a row of jets and multiple rows of jets and between circular and slot jets. The second part would involve the optimisation of the selected design option.

The first part of the automatic design process should be achievable by a rule based system using heuristics, basing the selection of one of the jet configuration options above on the geometry of the heat transfer surface. A long thin heat transfer surface suggesting a row of jets for example.

Optimisation may be considered as the process of deriving values for the main design parameters so as to give the maximum heat transfer rate whilst satisfying all the specified constraints. The optimisation of a system of impinging jets using the data currently available poses considerable challenges. The optimisation process would require a system to derive a heat transfer value for a given system in order to evaluate a design, it follows that an analysis system is required as a component of a design system. This is true of the iterative, generate and test, design methodologies, used by Maher and Fenvers (1985) and Wang and Lin (1989) for example, which would be appropriate for the optimisation of impinging jet designs. It is also true of purely numerical iterative optimisation procedures and of genetic algorithms. At present there is no single continuous mathematical function which will describe the variation of heat transfer coefficient over a wide range of values for all the significant design parameters, nor is it feasible to use either an analytical or a numerical prediction. The only source of heat transfer coefficients are correlations such as those provided by the Jet Impingement Knowledge Based Database System. There is a high likelihood that, at present, an attempt to build a design optimisation system using these techniques alone would fail because of problems with the evaluation of designs. The anticipated problems are:

(i) The choice between different values for heat transfer rate due to the probable lack of agreement between alternative correlations. For example correlations giving an erroneously high value for the heat transfer coefficient might mislead an optimisation algorithm into selecting parameter values within the range of that correlation as the optimum values.

(ii) The simplification of some correlations by the omission of parameters. Simplified correlations of this type would provide misleading information on the effect of changes in parameter values to the optimisation algorithm.

(iii) The inability to derive a heat transfer rate for a given design due to a lack of correlation data.

One possible solution to at least the first of these problems might be to use information on the rate of change of the heat transfer coefficient due to changes in the variable parameters as a guide. The maximum heat transfer coefficient occurring at a point where these partial differentials fall to zero.

9.5 Correlation of Jet Impingement Heat Transfer Data Using Neural Networks

The distribution of heat transfer coefficient due to an array of impinging jets is a function of many parameters

$$Nu = f(Pr, Re, z/D, lx/D, ly/D, m_n/m_c, x/D, y/D)$$

where x and y are co-ordinates on the heat transfer surface. Empirical results reviewed in Chapter 3 (for example Gardon and Carbonpue 1962) show a complex distribution of heat transfer coefficient. The review revealed no empirical correlations for the local heat transfer coefficient; though correlations are available for area mean heat transfer coefficients, for example Florschuetz *et al* (1981). An alternative method, the use of a neural network, appears to offer a method of correlating this data.

The neural network (for example Wythoff 1993) uses an implicit model to transform or map an input to the output. In contrast correlations have conventionally used an explicit mathematical model to perform this function. In the case of the heat transfer due to an array of jets above there would be eight input values and one output value, the Nusselt number. The implicit model is developed from example sets of input values for each of which the corresponding required output value is known. The response of the network is determined by weighted links. These weights are adjusted iteratively to minimise the error between the desired and derived output values.

Thibault and Grandjean (1991) presented results for correlations obtained using neural networks. Correlations were obtained for the area mean Nusselt number due to free convection from a horizontal cylinder

$$\overline{Nu} = f(Pr,Gr)$$

and for the local Nusselt number due to free convection from a slender vertical cylinder

$$Nu = f(Pr,Gr,n,\Omega)$$

where n is the exponent of power law variation of the surface heat flux and Ω is the surface curvature. These results exhibit a close agreement with those obtained from correlation expressions. The correlation of jet impingement data requires a larger number of independent variables.

The use of neural networks to correlate empirical data appears to offers some risk. No method exists (Wythoff) which enables the appropriate number of nodes in a network or the number of sets of examples required to be determined theoretically. It is possible that if a network with too many degrees of freedom (nodes and links) is used or too few sets of examples are provided then interpolation between example sets of data will be unreliable. Either scatter within the results may be fitted or the response of the network may fluctuate at random. At present the most satisfactory number of nodes must be determined by trials and error.

A correlation has been presented (Chapter 3)[1] for the heat transfer due to a single circular air jet impinging orthogonally onto a plane surface. The assignment of values to the eight coefficients in this correlation was performed by trial and error. It is likely that the accuracy of the correlation could be improved by numerical optimisation in which the coefficients were varied to achieve the minimum overall error. This overall error should probably be a weighted sum of the errors between the exponent of Reynolds number derived from regression and from the exponent derived from the correlation expression.

It is possible that a hybrid technique using neural networks and explicit expressions may form a useful method of correlating this data. The neural network may be found to be more suitable for the representation of the complex form of the distribution of heat transfer coefficient. However where the effect of parameters can be explicitly accounted for, for example Reynolds number in the case of the single jet, this should be used. The effect of a change in a parameter value on the heat transfer coefficient is explicit in the form of an explicit correlation, this is not true of a neural network.

9.6 The Incorporation of Computer Fluid Dynamics Modelling

Developments in turbulence modelling techniques, Craft et al (1993), have resulted in improvements in the modelling of flow

[1] Jambunathan *et al* (1992) reproduced as Appendix C

processes, including heat transfer coefficients, in impinging jets. However considerable discrepancies between numerical and experimental results were seen to remain. As the results from Computational Fluid Dynamics (CFD) improves it will become increasingly desirable to create a blend of experimental results and modelling within the correlation database. This will require further development of the KBDS. The new capabilities would include the generation of input to and the interpretation of output from CFD programs, Hartle (1993). It is envisaged that this input would be developed from the geometry and flow descriptions which are already created by the KBDS. Based on knowledge of CFD capability the KBDS could be used to decide whether to retrieve an empirical correlation or to obtain a CFD solution.

REFERENCES

Adeli,H. and Hung,S.L., 1990, A production system and relational database model for processing knowledge of earthquake-resistant design., *Eng. Appli. of AI*, 3, 4.

Andrews,G.E. and Husain,C.I., 1984, Impingement cooling using large arrays of holes., *I. Chem. E. 11th. Ann. Res. Meeting* 86-91.

Arnaiz,A., Forster,J.K.W. and Fothergill,P., 1991, Intelligent retrieval of information from technical reports., *11th Int. Conf. Expert Systems and their Applications*, 8, 55-66.

Ashton Tate (UK) Ltd., 1988, *dBase 4.1.1, Reference manual.*

Behbahani,A.I. and Goldstein,R.J., 1983, Local heat transfer to staggered arrays of impinging circular air jets., *J. Eng. Power. Trans ASME* 105(2), 354-360.

Bouchez,J.P. and Goldstein,R.J., 1975, Impingement cooling from a circular jet in a crossflow., *Int.J.Heat Mass Transfer,* 18 719-730.

Bragg,S.L., 1960, Effect of compressibility on the discharge coefficients of orifices and convergent nozzles., *J. Mech. Eng. Sci.* 2, 1.

Brauer,J. and Stuchly,J., 1991, An information management system for engineering design based on hypertext., *Computer aided engineering journal*. 8, 5, 189-193.

Brodie,M.L. and Jarke,M., 1986, On integrating logic programming and databases., in *Expert database systems - Proceedings from the first international workshop, Kiawah Island, South Carolina 1984.*, ed. Kerschberg,L., Benjamin/Cummings.

Brown,D.R., Knowledge based engineering analysis., *PhD Thesis, Stanford Univ.*, 1988.

Buchanan,B.G., Smith,D.H., White,W.C., Gritter,R.J., Feigenbaum,E.A., Lederberg,J. and Djerassi,C., 1976, Applications of Artificial Intelligence for Chemical Inference. 22. Automatic Rule Formation in mass spectrometry by means of the Meta-DENDRAL program., *J. of the American Chem. Soc.* 98:20, 6168-6178.

Bunker,R.S. and Metzger,D.E., 1988, Local heat transfer in internally cooled turbine airfoil leading edge regions - Part 1: Impingement cooling without film coolant extraction., *ASME HTD* 103, 53-63.

Butler,K.A., 1984, Effect of entrainment on the heat transfer between a flat surface and impinging circular air jets. *MSc Thesis, Univ.of Minnesota*.

Button,B.L. and Wilcock,D., 1978, Impingement heat transfer - a bibliography 1890-1975 *Previews Heat Mass Transfer*, 4(3), 83-89.

Button,B.L. and Jambunathan,K., 1989, Jet impingement heat transfer: a bibliography 1976-1985. *Previews Heat Mass Transfer*,15(2) 149-178.

Chen,Y. and Tseng,L., 1989, EDMS: an engineering data management system for design supporting environments, *Int. J. of Computer Applications in Technology*, 169-175.

Chupp,R.E. et al., 1969, Evaluation of internal heat transfer coeffient for impingement-cooled turbine aerofoils., *J. Aircr.*, 6, 3, 203-208.

Clarke,J.A. and MacRandal,D. 1991, An intelligent front-end for computer-aided building design. *Int. J. A.I. in Eng.*, 6, 1.

Clocksin W.F. and Mellish C.S., 1984, *Programming in Prolog*, Springer-Verlag.

Colladay,R.S, 1975, Turbine cooling. Turbine design and applications, Glassman,A.J. (ed), *NASA, SP*-290, 3, 59-101.

Craft,T.J., Graham,L.J.W. and Launder,B.E., 1993, Impingement jet studies for turbulence model assessment - 2. An examination ot the performance of four turbulence models, *Int. J. Heat Mass Transfer*, 36, 10, 2685-2697.

Creative Logic Ltd., Leonardo v.3.18., Brunel Science Park, Uxbridge, Middlesex, UK.

Cuppens,F. and Demolombe,R., 1989, Cooperative answering: a methodology to provide intelligent access to databases, *Proc. from the second intl. conf. on expert database systems*, Kerschberg,L. (ed), Benjamin/Cummings.

Davis,R.,Buchanan,B., and Shortliffe,E., 1977, Production rules as a representation for a knowledge-based consultation program., *Artificial Intelligence*, **8**, 15-45.

Davis,R., 1985, Interactive transfer of expertise., *Rule-based expert systems., The MYCIN experiments of the Stanford heuristic programming project.*, Buchanan,B.G. and Shortliffe,E.H. (eds.), Ch. 9, 171-205, Addison-Wesley. Originally in *Artificial Intelligence* 12, 121-157, (1979).

Delcambre,L.M.L., 1988, RPL: an expert system language with query power. *IEEE Expert* Winter 1988, 51-61.

Duda,R.O., and Reboh,R., 1984, Artificial Intelligence and decision making: the PROSPECTOR experience., in *A.I. applications for business.*, Reitman,W.,(ed.).

Fenichel,C.H., 1981, Online searching: measures that discriminate among users with different types of experience., *J. of the ASIS,* 32, 1, Jan. 1981, pp23-32, quoted in Gauch (1990).

Florschuetz,L.W. Berry,R.A. and Metzger,D.E., 1980, Periodic Streamwise variations of heat transfer coefficients for inline and staggered arrays of circular jets with crossflow and spent air. *Trans. ASME J. of Heat Transfer*, 102, 1.

Florschuetz,L.W. and Isoda,Y., 1983, Flow distributions and discharge coefficient effects for jet array impingement with initial cross-flow., *J. Eng. Power. Trans ASME*, 105(2), 296-304.

Florschuetz,L.W. and Su,C.C., 1987, Effects of crossflow temperature on heat transfer wthin an array of impinging jets., *J. Heat Transfer, Trans ASME.* 109(1), 74-82.

Florschuetz,L.W. Truman,C.R. and Metzger,D.E., 1981, Streamwise flow and heat transfer distributions for jet array impingement with crossflow. *Trans. ASME J. of Heat Transfer*, 103, 2.

Fischer,G. and Nakokoji,K. 1992, Beyond the macho approach of artificial intelligence: empower human designers - do not replace them. *Knowledge-Based Systems*, 5, 1, pp 15-30.

Fox,R.L., 1971, *Optimisation methods for engineering design.*, Addison-Wesley.

Gardon,R. and Caroonpue,J. 1962, Heat transfer between a flat plate and jets of air impinging on it. *Int. Developments in Heat Transfer*; *Int. Heat Transfer Conf., Univ. Colorado, 28 Aug.-1 Sept.*, 2, 454-460.

Gauch,S.E., 1990, An expert system for searching in full text. *PhD Thesis, Univ. N. Carolina at Chapel Hill.*

Goldstein,R.J. and Behbahani,A.I., 1982, Impingement of a circular jet with and without cross flow., *Int.J.Heat Mass Transfer* Vol.25 No.9 1377-1382.

Goldstein,R.J. and Franchett,M.E., 1988, Heat transfer from a flat surface to an oblique impinging jet., *ASME J. Heat Transfer,* 110, 87-93.

Goldstein,R.J.,Sobolik,K.A. and Seol,W.S., 1990, Effect of entrainment on the heat transfer to a heated circular air jet inpinging on a flat surface., *ASME J.of Heat Transfer*, Vol.112, 608-611.

Hartle,S.L., 1993, The developmemt of a knowledge based front end for a computational fluid dynamics package, *PhD Thesis The Nottingham Trent University.*

Hay,N. and Spencer,A., 1991, Discharge coefficients of cooling holes with radiused and chamfered inlets., *ASME Paper 91-GT-269.*

Hollworth,B.R. and Berry,R.D., 1978, Heat transfer from arrays of impinging jets with large jet-to-jet spacing. *ASME Paper No.* 78-GT-117.

Hollworth,B.R. and Bowley,W.W., 1975, Heat transfer characteristics of an impinging jet in a crossflow., *ASME* 75-WA/HT-100.

Hollworth,B.R. and Cole,G.H., 1987, Heat transfer to arrays of impinging jets in a crossflow., *Trans. ASME J. of Heat Transfer*, 109.

Hollworth,B.R. and Gero,L.R., 1985, Entrainment effects on impingement heat transfer. Part 2. Local heat transfer measurements. *J. Heat Transfer Trans. ASME*, 107, 910-915.

Hollworth,B.R. and Wilson,S.I. 1984, Entrainment effects on impingement heat transfer. Part 1. Measurements of heated jet velocity and temperature distributions and recovery temperatures on target surface. *J. Heat Transfer. Trans.* ASME., 106(4), 797-803.

Howard,H.C. and Rehak,D.R., 1986, KADBASE Interfacing expert systems with databases. *IEEE Expert*, Fall, 65-76.

Humphrey,S.M., 1987, Illustrated description of an interactive knowledge-based indexing system., *Proc. of the ACM SIGIR conf. New Orleans,* 73-90.

Justonis, V.J., 1970, Heat transfer from impinging gas jets on an enclosed concave surface., *J. Aircr.*, 7, 1, 87-88.

Jambunathan,K., Lai,E., Moss,M.A. and Button,B.L., 1992, A review of heat transfer data for single circular jet impingement., *Int. J. Heat and Fluid Flow*, 13, 2, 106-115.

Kercher,D.M. and Tabakoff,W. 1969, Heat transfer by a square array of round air jets impinging perpendicular to a flat surface including the effect of spent air., *ASME Paper No.* 69-GT-4.

Kerschberg,L., 1990, Expert database systems: knowledge/data management environments for intelligent information systems, *Information Systems,* 15, 1 151-160.

Kernigham,B. and Ritchie,D., 1978, The C programming language, *Prentice - Hall.*

Ketonen,T.,Loundamaa,P., and Nurminen,J.K., 1988, An electronic design CAD system combining knowledge-based techniques with optimisation and simulation., in *Artificial intelligence in engineering: design.*, ed. Gero,J.,S., Elsevier/Computational Mechanics Publications.

Kieger,K.K., 1981, Local and average heat transfer from a flat surface to a single circular jet impinging on it. *MSc Thesis*, Univ. of Minnesota, MN, USA.

Koopman,R.N. and Sparrow,E.M., 1976, Local and average transfer coefficients due to an impinging row of jets, *Int. J. Heat Mass Transfer*, 19, 673-683.

Kowalski,R., 1988, Logic for data description. in *Readings in Artificial Intelligence and databases* eds. Mylopoulos,J. and Brodie,M.L., Morgan Kaufmann.

Lai,E., Moss,M.A, Jambunathan,K. and Button,B.L., 1992, The implementation of a deductive database for engineering correlations., *Engng. Applic. Artif. Intell.*, 5, 2, pp 145-153.

Lloyd,J.W., 1983, An introduction to deductive database systems., *The Australian Computer Journal* 15, 2, 52-57.

Lucarella,D and Morara,R., 1991, FIRST: Fuzzy Information Retrieval System., *J. Inf. Sci.* 17,2,81-91.

Lucas,R., 1988, *Database applications using Prolog*, Ellis Horwood.

Maher,M.L., and Fenvers,S.J., 1985, Hi-Rise: An expert system for the preliminary structural design of high rise buildings. in *Knowledge Engineering in Computer Aided Design.* Gero,J.S. (ed.), Elsevier (North-Holland).

Marcus,S., Stout,J. and McDermott,J., 1988, VT: An expert elevator designer that uses knowledge based bachtracking. *AI Magazine*, Spring 1988, pp 95-111.

Marshall,R., 1991, Manipulating full_text scientific databases: A logic-based semantico-pragmatic approach, *The Computer Journal*, 34, 3, 245-253.

MathSoft Inc., 1993, Mathcad 4.0

McDermott,J., 1982, R1: A rule based configurer of computer systems. *Artificial Intelligence*, 19, 39-88.

McLean,S. and Weise,C., 1991, Digress: A deductive interface to a relational database, *J. of the American Soc. for Information Sci.*, 42, 1, 49-63.

Miller,D.S., 1978, *Internal flow systems.* British Hydrodynamics Research Association.

Metzger,D.E., Baltzer,R.T. and Jenkins,C.W., 1972, Impingement cooling performance in gas turbine aerofoils including effects of leading edge sharpness, *Trans. ASME J. of Engineering for Power*, 94, 3, 219-225.

Metzger,D.E., Florschuetz,L.W., Takeuchi,D.I., Behee,R.D. and Berry,R.A., 1979, Heat transfer characteristics for in line and staggered arrays of circular jets with crossflow and spent air, *Transactions of the ASME Journal of heat transfer,* 101, 526-531.

Metzger,D.E. and Korstad,R,J. 1972, Effects of crossflow on impingement heat transfer, *Trans. ASME J. of Engineering for Power*, 35-41.

Metzger,D.E., Yamashita,T. and Jenkins,C.W., 1969, Impingement cooling of concave surfaces with lines of circular air jets, *J. Eng. Power,* 91, 3, 149-158.

Monico,R.D. and Chew,J.W., 1992, Modelling thermal behaviour of turbo machinery discs and casings, *AGARD 80th Symp. of the Propulsion and Energetics Panel on Heat Transfer and Cooling in Gas Turbines, Antalya, 12-15 Oct.*

Obot,N.T., Mujumdar,A.S. and Douglas,W.J.M., 1979, Effect of nozzle geometry on impingement heat transfer under a round turbulent jet. *ASME Paper* 79-WA/HF-53.

Obot,N.T. and Trabold,T.A. 1987, Impingement heat transfer within arrays of circular jets: part 1 - Effects of minimum, intermediate, and complete crossflow for small and large spacings. *Trans. ASME J. of Heat Transfer*, 109.

Parsaye,K., Chignell,M., Khoshafian,S. and Wong,H., 1989, *Intelligent Databases*, Wiley, pp293-346.

Peterson,R.E., 1974, *Stress Concentration Factors*, John Wiley - Interscience.

Popiel,C.O. and Boguslawski, L., 1988, Effect of flow structure on the heat or mass transfer on a flat plate in impinging round jet. *2nd UK National Conf. on Heat Transfer*, Univ. of Strathclyde, UK, September 14-16, 1, 663-685.

Popplestone,R., Smithers,T., Corney,J., Koutsou,A., Millington,K. and Sahar,G., 1986, Engineering design support systems., *D.A.I. Research Paper No.286, Univ. of Edinburgh.*

Powell,D.J., Skolnick,M.M. and Tong,S.S., 1991, Interdiditation: a hybrid technique for engineering design optimisation employing genetic algorithma, expert systems, and numerical optimisation. in *Handbook of Genetic Algorithms*, Davis,L. (ed.) Van Nostrand Reinhold.

Preiss,K., 1980, Data frame model for the engineering design process., *Design studies*, 1, 231-243.

Quintec Systems Ltd., 1986 and 1987, *SD-Prolog reference manuals, SD Prolog ver.2.* Quintec systems Ltd. Midland Ho., Botley, Oxford.

Rowe,N.C., 1988, *Artificial intelligence through Prolog.*, Prentice-Hall.

Seol,W., 1987, Heat transfer from a flat plate to a row of impinging circular air jets including the effect of entrainment., *MS Thesis* Univ. Minnesota.

Sim,S.K. and Chan,Y.W., 1991, A knowledge based expert system for rolling-element bearing selection in mechanical engineering design., *AI in engineering.*, 6, 3, pp125-135.

Smith,P.J., Krawczak,D., Shute,S.J. and Chignell,M.H., 1987, Bibliographic information retrieval systems: increasing cognitive compatibility., *Papers from the 29th NFAIS ann. conf. Arlington 1987, Information services and use.*, 7, 4-5, 95-102.

Smith,P.J., Shute,S.J., Galdes,D. and Chignell,M.H., 1989, In search of knowledge-based search tactics., *Proc. 12th. ann. int. ACM SIGIR conf. on research and dev. in information retrieval.*, 3-10.

Smithers,T., Tang,MX., Tomes,N., Buck,P., Clarke,B., Lloyd,G., Poulter,K., Floyd,C. and Hodgkin,E. 1992, Development of a knowledge-based design support system. *Knowledge-Based Systems*, 5, 1, pp 31-40.

Sparrow,E.M.,Goldstein,R.J. and Rouf,M.A., 1976, Effect of nozzle - surface separation distance on impingement heat transfer for a jet in a crossflow., *ASME J. of Heat Transfer*, 76-HT-F.

Stallman,R.M., and Sussman,G.J., 1977,Forward reasoning and dependency-directed backtracking in a system for computer-aided circuit analysis., *Artificial Intelligence*, 9, 135-196.

Stonebraker,M. and Rowe,L.A., 1986, The design of Postgres, *Proc. of the 1986 ACM SIGMOD conf. Washington DC.*, 15, 2.

Sutherland,I.E., 1980, *Sketchpad, a man machine graphical communication system.*, Garland, Originally published as a PhD Thesis 1963.

Thibault,J. and Grandjean,B.P.A., 1991, A neural network methodology for heat transfer data analysis., *Int. J. Heat Mass Transfer*, 34, 8, 2063-2070.

Van der Spec, 1990, The paint advisor., *Expert Systems*, 7, 4.

Torsun,I.S. and Ng,Y.M., 1988, TCEDI Tightly coupled expert database systems interface, *Proc. expert systems 88, The 8th. ann. tech. conf. British computer soc. spec. group expert systems, Brighton 12-15 Dec.* 210-223.

Wang,H.P., and Lin,A.C., 1989, Camex: An expert system for selecting cam-follower design parameters. *Int.J.Adv.Manuf.Technol.* 4:46-71

Ward,P.T. and Mellor,S.J., 1985, *Structured Development for Real-Time Systems.*, Yourdon Press.

Watson,I., Basden,A. and Brandon,P., 1992, The client-centred approach: expert system maintenance. *Expert Systems*, 9, 4, 189-195.

Waugh,K.G., Williams,M.H., Kong,Q., Salvini,S. and Chen,G., 1990, Designing SQUIRREL: an extended SQL for a deductive database system, *The computer J.,* 33, 6, 535-546.

Winston,P.H., 1992, *Artificial Intelligence*, Addison-Wesley.

Wythoff,B.J., 1993, Backpropagation neural networks: A tutorial, *Chemometrics and Intelligent Laboratory Systems*, 18, 115-155.

Yeh,F.C. and Stepka,F.S., 1984, Review and status of heat-transfer technology for internal passages of air-cooled turbine blades, *NASA technical paper 2232.*

Yoon,S-C. and Henschen,L.J., 1989, Intelligent control in expert database systems using a heuristic approach., *Proc. IEEE int. symp. intelligent control.,* 399-405.

APPENDIX A
EXAMPLE RUNS OF THE JET IMPINGEMENT KNOWLEDGE BASED DATABASE SYSTEM - OBTAINING CORRELATIONS

Copies of some of the screen output taken during sample runs of the program illustrate the function and use of the jet impingement KBDS. In this example the database was searched to obtain heat transfer correlations for a staggered array of circular impinging jets where the jet flow rates were known.

```
═══════════════════════╡Jet Impingement Heat Transfer Database╞═══════════════════════

          JET IMPINGEMENT HEAT TRANSFER CORRELATION DATABASE
          -------------------------------------------------

    Heat transfer correlations can be selected, examined and evaluated.
A typical run of the program will have three stages:
1) An initial selection of correlations is made using keywords. The available
keywords will be offered by menus enabling a progressively more specific
description of the heat transfer process to be built up. Once this description
has been finalised you will be prompted for parameter values to give a more
precise description.
2) The selected correlations may then be checked to ensure that they are
appropriate to the system described.
3) The checked correlations may then be evaluated.

    Selected correlations may be displayed. Further options exist to modify
the search and checking process.                ═══╡Options╞═══
                                                 Initial Search
                                                 Quit

Copyright (c) 1993 Michael A Moss and Nottingham Trent University
All rights reserved.
```

Figure A-1.

The first screen gives a brief introduction to the function of the program and the usual strategy employed in finding appropriate correlations.

N.B. In this example run of the program the menu option selected by the user is shown in **bold** type.

```
Jet Impingement Heat Transfer Database
      Initial Search using Keywords
      -----------------------------

Select keywords to describe your heat transfer process.
Select a keyword from the menu or choose one of the options:
'No keyword' to reject all the displayed keywords,
'Draw keyword tree' to display the keyword hierarchy,
'Back step' to re-do previous selection.

Keyword meanings
forced_convection:
FORCED CONVECTION occurs due to an externaly imposed flow,
it generally results in high levels of heat transfer.
free_convection:
FREE CONVECTION is due to a flow generated by a
temperature difference, if the fluid is heated then the
fluid will rise due to its increased buoyancy.

Select a keyword
forced_convection
free_convection
No keyword
Draw keyword tree
Back step
Quit
```

Figure A-2.

The program helps the user build up a description of the heat transfer geometry and flow, initially this will be description using a collection of keywords. The keywords are arranged in a hierarchy beginning with very general words and progressively becoming more specific. The user selects the most apt keyword from the alternatives offered, a brief explanation of the meaning of each keyword is given to help in this choice. The user may also opt to view the keyword hierarchy below the current keyword by selecting the option **'Draw keyword tree'**.

```
═══════════════════════╡Jet Impingement Heat Transfer Database╞═══════════════════
                           Initial Search using Keywords
                           -----------------------------
Se ══════════════════════════╡Keyword Hierarchy╞══════════════════════════════════
Se│  Hierarcy of keywords below convection      Press Return to continue.
'N│ convection
'D│       ¦-----forced_convection
══        ¦           ¦-----jet_impingement
fo│       ¦           ¦           ¦-----array
FO│       ¦           ¦           ¦           ¦-----max_crossflow
it│       ¦           ¦           ¦           ¦           ¦-----initial_crossflow
fr│       ¦           ¦           ¦           ¦-----int_crossflow
FR│       ¦           ¦           ¦           ¦-----min_crossflow
te│       ¦           ¦           ¦           ¦-----in_line
fl│       ¦           ¦           ¦           ¦-----staggered
  │       ¦           ¦           ¦-----row
  │       ¦           ¦           ¦           ¦-----crossflow
  │       ¦           ¦           ¦-----single
  │       ¦           ¦           ¦           ¦-----crossflow
  │       ¦           ¦           ¦-----circular
  │       ¦           ¦           ¦           ¦-----orifice
  │       ¦           ¦           ¦           ¦-----pipe
  │       ¦           ¦           ¦           ¦-----contoured
```

Figure A-3.

The window displaying the hierarchy can be scrolled (using the arrow keys) to enable the remainder of the keywords to be seen. In this example the user selects the keyword **'forced convection'** after viewing the hierarchy.

Jet Impingement Heat Transfer Database
Initial Search using Keywords

Select keywords to describe your heat transfer process.
Select a keyword from the menu or choose one of the options:
'No keyword' to reject all the displayed keywords,
'Draw keyword tree' to display the keyword hierarchy,
'Back step' to re-do previous selection.

Keyword meanings

jet_impingement:
A forced convection due to one or more IMPINGING JETS.
wall_jet:
A flow confined to a region adjacent to a solid surface

Select a keyword

jet_impingement
wall_jet
No keyword
Draw keyword tree
Back step
Quit

Figure A-4.

Since the keyword 'forced convection' has been selected the available keywords are now 'jet impingement' and 'wall jet', these are the only two types of forced convection currently available in the database. The user selects '**jet impingement**' and is then prompted to choose keywords which further specify the type of jet impingement.

```
Jet Impingement Heat Transfer Database
Initial Search using Keywords
------------------------------

Select keywords to describe your heat transfer process.
Select a keyword from the menu or choose one of the options:
'No keyword' to reject all the displayed keywords,
'Draw keyword tree' to display the keyword hierarchy,
'Back step' to re-do previous selection.

Keyword meanings
array:
Jet impingement due to an ARRAY or multiple rows of jets.
row:
Jet impingement due to one ROW of jets.
single:
Jet impingement due to a SINGLE jet.

Select a keyword
array
row
single
No keyword
Draw keyword tree
Back step
Quit
```

Figure A-5.

```
Jet Impingement Heat Transfer Database
Initial Search using Keywords
------------------------------

Select keywords to describe your heat transfer process.
Select a keyword from the menu or choose one of the options:
'No keyword' to reject all the displayed keywords,
'Draw keyword tree' to display the keyword hierarchy,
'Back step' to re-do previous selection.

Keyword meanings
circular:
Each jet flows from a CIRCULAR hole.
slot:
Each jet flows from a rectangular SLOT.

Select a keyword
circular
slot
No keyword
Draw keyword tree
Back step
Quit
```

Figure A-6.

The keywords which can be used to specify the type of jet impingement are 'array', 'row', 'single', 'circular', and 'slot'. These are arranged into two groups in the keyword hierarchy because they address different aspects of jet impingement. The user is prompted to select one keyword from each group, here the user selects **'array'** and **'circular'**. Note that the search descends the hierarchy breadth first, this enables the user to select all the keywords which are required to specify the type of jet impingement before being required to specify the type of array or the geometry of the circular jet nozzle.

From the following two screens the user selects **'flat'**, to describe the heat transfer surface, and **'max crossflow'** and **'staggered'** which give more detail of the type of jet array.

```
Jet Impingement Heat Transfer Database
Initial Search using Keywords
------------------------------

Select keywords to describe your heat transfer process.
Select a keyword from the menu or choose one of the options:
'No keyword' to reject all the displayed keywords,
'Draw keyword tree' to display the keyword hierarchy,      Select a keyword
'Back step' to re-do previous selection.                   initial_crossflow
Keyword meanings                                           No keyword
initial_crossflow:                                         Draw keyword tree
An externaly imposed flow enters on one side of the array. Back step
This INITIAL CROSS-FLOW is additional to the induced       Quit
crossflow.
```

Figure A-7.

The user selects **'no keyword'** at this point because the jet impingement system being described does not have a cross-flow perpendicular to the first row of jets.

In the subsequent screen the user specifies that the circular jet nozzles are orifices, using the keyword **'orifice'**.

```
Jet Impingement Heat Transfer Database

Keywords selected:
[forced_convection,jet_impingement,array,circular,flat,max_crossflow,staggered
,orifice]
5 correlations selected in 0.270000secs.


              Options
 Carry out another search
 Look at a summary of the search results
 Quit
```

Figure A-8.

Once the list of keywords has been assembled a search is made of the database for records which contain *all* the specified keywords. The user may then opt to try a different selection of keywords, which is what the user does in this example. The option **'Look at a summary of the search results'** enables the user to choose a selection of correlations from a summary of all the keyword searches performed so far.

```
                    Jet Impingement Heat Transfer Database
                      Summary of Keyword Searches
                      ---------------------------

Search No.1.  5 records selected using keywords:
forced_convection jet_impingement array circular flat max_crossflow staggered
orifice
Search No.2.  5 records selected using keywords:
forced_convection jet_impingement array circular flat staggered orifice
Search No.3.  4 records selected using keywords:
forced_convection jet_impingement array circular flat max_crossflow in_line
orifice
Do you want to select a list of records for further investigation? y/n :y
Enter the search number:1
```

Figure A-9.

The first selection of correlations is chosen and will now be checked to determine whether the physical dimensions used in the experimental derivation of the correlations are appropriate to the geometry and flow rates of the impinging jet configuration under investigation. To do this the user must first specify the values for parameters defining the geometry and flow rates. An appropriate list of these parameters is generated automatically by the knowledge base.

```
                  Jet Impingement Heat Transfer Database
          Quantitative Definition of Heat Transfer Geometry and Flow
        ----------------------------------------------------------
a              Jet area at nozzle exit (m*m)
d              Jet diameter at nozzle exit (m)
lx             Jet to jet spacing parallel to crossflow (m)
ly             Jet to jet spacing perpendicular to crossflow (m)
nx             Number of jet holes on X axis in the array
ny             Number of jet holes on Y axis in the array
p1             Total Pressure upstream of nozzle (N/m/m)
p2             Static Pressure at nozzle exit (N/m/m)
ta             Ambient temperature (K)
tw             Impingement plate wall temperature (K)
t1             Total Temperature upstream of nozzles (K)
v              Jet mean velocity at nozzle exit (m/s)
z              Jet to target plate spacing (m)
cd             Nozzle discharge coefficient
m              Mass flow per jet (kg/s)

Enter pairs of parameter names and values eg d 0.001 m 0.005 z 0.03 RETURN
The program will fill in values itself where possible.
cd .7 d .005 m .001 p1 120000 t1 300
```

Figure A-10.

Equations exist which enable the program to derive many of these parameter values from just a few known values; for example five further values are derived from the first five values input by the user.

```
                    Jet Impingement Heat Transfer Database
         Quantitative Definition of Heat Transfer Geometry and Flow
       ----------------------------------------------------------
a    0.000020   Jet area at nozzle exit (m*m)
d    0.005000   Jet diameter at nozzle exit (m)
lx              Jet to jet spacing parallel to crossflow (m)
ly              Jet to jet spacing perpendicular to crossflow (m)
m    0.001000   Mass flow per jet (kg/s)
nx              Number of jet holes on X axis in the array
ny              Number of jet holes on Y axis in the array
p1   120000.00  Total Pressure upstream of nozzle (N/m/m)
p2   118120.49  Static Pressure at nozzle exit (N/m/m)
ta              Ambient temperature (K)
tw              Impingement plate wall temperature (K)
t1   300.00000  Total Temperature upstream of nozzles (K)
v    36.956356  Jet mean velocity at nozzle exit (m/s)
z               Jet to target plate spacing (m)
cd   0.700000   Nozzle discharge coefficient

Enter pairs of parameter names and values eg d 0.001 m 0.005 z 0.03 RETURN
The program will fill in values itself where possible.
lx .02 ly .04 nx 4 ny 10 z .0125 ta 288 tw 400
```

Figure A-11.

In this example the remaining required input is **lx = 0.02, ly = 0.04, nx = 4, ny = 10, z = 0.0125, ta = 288 and tw = 400.**

```
Jet Impingement Heat Transfer Database
        Quantitative Definition of Heat Transfer Geometry and Flow
        ----------------------------------------------------------
a    0.000020  Jet area at nozzle exit (m*m)
af   0.024544  Total area jets / Area jet plate
ap   0.032000  Area of orifice plate or jet plate (m*m)
d    0.005000  Jet diameter at nozzle exit (m)
g    1.250000  Total jet mass flow per unit area target (kg/m/m)
gp   1.250000  Total jet flow per unit area of orifice or jet plate (kg/m/m)
lx   0.020000  Jet to jet spacing parallel to crossflow (m)
ly   0.040000  Jet to jet spacing perpendicular to crossflow (m)
m    0.001000  Mass flow per jet (kg/s)
n    1250.0000 Number of jet holes per unit area (1/m/m)
nx   4.000000  Number of jet holes on X axis in the array
ny   10.000000 Number of jet holes on Y axis in the array
p1   120000.00 Total Pressure upstream of nozzle (N/m/m)
p2   118120.49 Static Pressure at nozzle exit (N/m/m)
ta   288.00000 Ambient temperature (K)
tw   400.00000 Impingement plate wall temperature (K)
t1   300.00000 Total Temperature upstream of nozzles (K)
t2   298.64992 Static Temperature at nozzle exit (K)
v    36.956356 Jet mean velocity at nozzle exit (m/s)
va   36.956356 Axial center line velocity of free jet at given z/d (m/s)
                                                      Press any key ...
```

Figure A-12.

```
Jet Impingement Heat Transfer Database
d    0.005000  Jet diameter at nozzle exit (m)
g    1.250000  Total jet mass flow per unit area target (kg/m/m)
gp   1.250000  Total jet flow per unit area of orifice or jet plate (kg/m/m)
lx   0.020000  Jet to jet spacing parallel to crossflow (m)
ly   0.040000  Jet to jet spacing perpendicular to crossflow (m)
m    0.001000  Mass flow per jet (kg/s)
n    1250.0000 Number of jet holes per unit area (1/m/m)
nx   4.000000  Number of jet holes on X axis in the array
ny   10.000000 Number of jet holes on Y axis in the array
p1   120000.00 Total Pressure upstream of nozzle (N/m/m)
p2   118120.49 Static Pressure at nozzle exit (N/m/m)
ta   288.00000 Ambient temperature (K)
tw   400.00000 Impingement plate wall temperature (K)
t1   300.00000 Total Temperature upstream of nozzles (K)
t2   298.64992 Static Temperature at nozzle exit (K)
v    36.956356 Jet mean velocity at nozzle exit (m/s)
va   36.956356 Axial center line velocity of free jet at given z/d (m/s)
                                                      Press any key ...
z    0.012500  Jet to target plate spacing (m)
rho  1.378101  Density (kg/m/m/m)
p_ra0.984337  Nozzle pressure ratio
cd   0.700000  Nozzle discharge coefficient
                                                      Press any key ...
```

Figure A-13.

Which gives the complete set of parameter values shown in the preceeding two figures.

```
Jet Impingement Heat Transfer Database

                                  Options
                                  Display selected correlations
                                  Check correlation validity
                                  Quit
```

Figure A-14.

Before the correlations are evaluated a check should be made on each of the selected correlations. This check will compare the jet impingement geometry and flow conditions specified by the user with the range of geometry and flow rates used in deriving each of the correlations. The ranges of only the most significant parameters or parameter groups (non-dimensional or semi-dimensional) are recorded in the database and used in this comparison. It is also possible to take a detailed look at each of the correlations at this point.

Jet Impingement Heat Transfer Database

CHECK LIMITS ON CORRELATION RANGE

Limits are given for the range over which each of the correlations are valid.

Options

Accept the existing limits
Use the knowledge base to broaden the limits
Widen the limits by a specified tolerance
Ignore limits and extrapolate
Quit

Figure A-15.

Each correlation can either be accepted or rejected by applying one of four criteria when making this comparison. These criteria are:

(i) to use the range of values of the non-dimensional groups recorded in the database, and to reject any correlation where one or more of the corresponding values derived from the geometry and flow conditions specified by the user lie outside this range.

(ii) to relax this strict interpretation of the conditions where a correlation is valid. A correlation may be accepted if the amount by which one or more of the non-dimensional groups falls outside the limits is not too large. The decision on whether this amount is acceptable is performed automatically using a rule-base embodying knowledge of the effect of changes in value of selected non-dimensional groups on the heat transfer coefficients.

(iii) to request a tolerance on one of more of the parameters specified by the user. The limits involving these parameters will be relaxed by the specified amount.

(iv) to accept all the correlations by ignoring the limits. The correlations will be extrapolated, possibly by a large amount, and the results should be treated with caution.

```
                    ┤Jet Impingement Heat Transfer Database├
                     CHECK LIMITS ON CORRELATION RANGE
                     ---------------------------------
Limits are given for the range over which each of the correlations are valid.
1 correletions accepted, 4 correlations rejected
                                                        Press any key ...

                                           ┤Options├
                                      Display selected correlations
                                      Evaluate selected correlations
                                      Broaden Search
                                      Check correlation validity
                                      Quit
```

Figure A-16.

A strict application of the limits results in all the correlations except one being rejected. To enable a comparison between a number of correlations the check can be repeated, but with less stringent criteria. The option '**Check correlation validity**' is selected again, but this time is followed by the option '**Use the knowledge base to broaden the limits**'.

```
Jet Impingement Heat Transfer Database
Hollworth,B.R. and Cole,G.H. 1987, correlation a.
ly/d = 8.000000 too high, upper limit 4.000000
lx/d = 4.000000 too low, lower limit 8.000000
nx = 4.000000 in limits
4*m/pi/mu/d = 14147.106051 in limits
z/d = 2.500000 in limits
                                                             Press any key...
Behbahani,A.I. and Goldstein,R.J. 1983, correlation a.
Search rule from Hollworth and Cole 1987
Relaxed limits satisfied for nx. Significance low
4*m/d/pi/mu = 14147.106051 in limits
z/d = 2.500000 in limits
ly/d = 8.000000 in limits
lx/d = 4.000000 in limits
                                                             Press any key...
Florschuetz,L.W., Truman,C.R. and Metzger,D.E. 1981, correlation b.
z/d = 2.500000 in limits                          Options
ly/d = 8.000000 in limits                        Display selected correlations
lx/d = 4.000000 too low, lower limit 5.00000     Evaluate selected correlations
rho*v*d/mu = 14147.106051 in limits              Broaden Search
                                                 Check correlation validity
2 correletions accepted, 3 correlations reje     Quit
```

Figure A-17.

Of the original five correlations selected via the keyword description two are now found to be acceptable. The program informs the user which limits are out of range. The program also informs the user which limits have been satisfied by using the knowledge base to relax the limits; in which case the source of the rule used in relaxing the limits is indicated as is the significance or the expected maximum size of the error introduced. These errors are described as 'very low', 'low', 'medium' or 'high', and correspond to maximum expected errors in the heat transfer Nusselt number of 10%, 20%, 33% and 50%.

If at this point there are still too few correlations selected the user may retrieve the original five correlations and either relax the selection criteria still further by specifying tolerances or decide to accept all five correlations. Alteratively a further set of rules in the knowledge base may be used to modify the search itself by choosing the option 'broaden search', this option will be discussed below.

```
═══════════════╡Jet Impingement Heat Transfer Database╞═══════════════
Hollworth,B.R. and Cole,G.H. 1987, correlation a.
ly/d = 8.000000 too high, upper limit 4.000000
lx ══════════════════╡Selected Correlations╞══════════════════
nx│ Current correlation selection              Press Return to continue.
4*│Hollworth,B.R. and Cole,G.H.   1987
z/│Heat transfer to arrays of impinging jets in a crossflow
  │Trans. ASME J. of heat transfer, 109
Be│Descriptive keywords:
Se│forced_convection,jet_impingement,array,circular,orifice,staggered,max_cro
Re│Limits of validity for correlation:
4*│     1.000000 < z/d < 3.000000
z/│     8.000000 < ly/d < 8.000000
ly│     4.000000 < lx/d < 4.000000
lx│     4.000000 < nx < 8.000000
  │     2500.000000 < 4*m/pi/mu/d < 25000.000000
Fl│Correlation Expressions.
z/│ re=m*4/pi/mu/d
ly│ coeff,holl_87b.dat,z/d,table
lx│ nu=re**0.784*pr**(1/3)*coeff/100
rh│ write "Area mean htc"
  │ write nu
2 │ h=nu*lambda/d
```

Figure A-18.

At this point the user chose to examine the correlations, by selecting the option **'Display selected correlations'**. A reference to the source of the correlation, the descriptive keywords, the limits and the correlation statements are displayed for each correlation. The user then obtains derived values for the heat transfer Nusselt number, by selecting the option **'Evaluate selected correlations'**.

```
Jet Impingement Heat Transfer Database
EVALUATE THE SELECTED CORRELATIONS
----------------------------------

Hollworth,B.R. and Cole,G.H.   1987
coeff = 4.350000
Area mean htc
nu = 69.017315
h = 362.202869
                                                  Press any key ...
Behbahani,A.I. and Goldstein,R.J. 1983
coeff = 0.094200
exp = -0.643600
Area mean htc
nu = 90.252126
h = 473.643159

                                                  Press any key ...
                                        Options
                                        Display selected correlations
                                        Evaluate selected correlations
                                        Broaden Search
                                        Check correlation validity
                                        Quit
```

Figure A-19.

The Nusselt numbers derived from the two correlations are displayed.

An alternative method of increasing the number of correlations retrieved from the database is to make the description of the heat transfer problem less specific. This can be done manually by the user simply by specifying fewer keywords. The knowledge base provides an automatic method for choosing which keywords to omit from a description, this method is selected by the option **'Broaden search'**.

```
Jet Impingement Heat Transfer Database

          Automatic Knowledge Based Broadening of Search.
          -----------------------------------------------

Search rule from Metzger et al 1979
The significance of staggered is very_low
9 correlations selected in 0.220000secs.
                                                   Press any key ...
Search rule from Metzger et al 1979
The significance of staggered is low
9 correlations selected in 0.220000secs.
                                                   Press any key ...
Summary of results of Broadening search
Search No. 1. Number of records selected 9
Keywords:
forced_convection jet_impingement array circular max_crossflow orifice flat
Search No. 2. Number of records selected 9
Keywords:
forced_convection jet_impingement array circular max_crossflow orifice flat
Select a list of correlations for further investigation.
Enter the search number:1
```

Figure A-20.

The KBDS examines each of the keywords originally selected by the user. The knowledge base is used to determine what likely error would be introduced into the derived heat transfer Nusselt number by omitting a keyword from the description. The significance of or size of error likely to be introduced by a given change to the description ranges from 'very low', through 'low' and 'medium' to 'high'. The corresponding likely errors are 10%, 20%, 33% and 50%. The source of the rule used to estimate the likely approximation due to providing a less specific description of the heat transfer process is indicated. The KBDS performs a database search using each modified keyword list, a summary of these searches are displayed and the user is prompted to choose one of the sets of correlations for checking and evaluation as above.

```
                    Jet Impingement Heat Transfer Database
                      EVALUATE THE SELECTED CORRELATIONS
                      ----------------------------------

Hollworth,B.R. and Cole,G.H.    1987
coeff = 4.350000
Area mean htc
nu = 69.017315
h = 362.202869
                                                       Press any key ...
Kercher,D.M. and Tabakoff,W.    1970
coeff = 0.075000
Enter a value for mi :0

coeff1 = 1.002500
coeff1 = 0.913324
coeff1 = 0.843100
coeff1 = 0.786585                            Options
exp = 0.746500                          Retrieve initial selection
Area mean htc                           Display selected correlations
nu = 80.114512                          Evaluate selected correlations
h = 420.440961                          Broaden Search
                                        Check correlation validity
                                        Quit
```

Figure A-21.

As before the correlations must be checked prior to evaluation. In this example the un-modified limits were applied.

A further run of the KBDS illustrating the use of a knowledge base to modify the users query

The following seven figures give a further demonstration of the use of domain knowledge to obtain additional correlations. In this example, rather than just broadening the search by making the description less specific (achieved by eliminating one or more keywords from the query or widening the allowable range on the significant non-dimensional groups), a new description is substituted for the original description.

```
                    Jet Impingement Heat Transfer Database
                     CHECK LIMITS ON CORRELATION RANGE
                     ---------------------------------
Limits are given for the range over which each of the correlations are valid.
Goldstein,R.J. and Seol,W.S. 1991, correlation a.
ly/d = 5.072368 in limits
z/d = 4.440789 in limits
x/d = 5.921053 in limits
Search rule from Jambunathan, Lai, Moss and Button 1992
Relaxed limits satisfied for rho*v*d/mu. Significance low
                                                           Press any key...
Metzger,D.E. and Korstad,R.J. 1972, correlation a.
Enter a value for mc :0

mc/m = 0.000000 too low, lower limit 1.000000
x/d = 5.921053 too low, lower limit 10.000000
ly/d = 5.072368 too high, upper limit 5.000000
z/d = 4.440789 too high, upper limit 2.000000
2*rho*v*b/mu = 16155.757977 too high, upper limit 6000.000000
                                                           Press any key...
```

Figure A-22.

An initial description of the geometry and the flow conditions of a row of jets used to cool a gas turbine component was specified. The keywords required were '**forced convection**', '**jet impingement**', '**row**', '**circular**', '**flat**' and '**orifice**'. The following parameter values were also specified **Cd = 0.8, ny 240, d = 0.00304m, z = 0.0135 m, ly = 0.01542m, p1 = 179263 Pa, p2 =**

101325 Pa, t_{amb} = 420, t_m = 520, t1 = 350 and x = 0.018m.

The options **'Check correlation validity'**, Figure A-14, and **'Use the knowledge base to broaden the limits'**, Figure A-15, were then selected. Even though the limits are relaxed by a rule based on results from Jambunathan *et al* (1992) (Figure A-22) only one correlation was selected. This correlation was then evaluated, resulting in a mean heat transfer coefficient over a strip extending 0.018m on either side of the stagnation points of 1076 W/m^2K (Figure A-23).

```
                    Jet Impingement Heat Transfer Database
                     EVALUATE THE SELECTED CORRELATIONS
                     ----------------------------------

Goldstein,R.J. and Seol,W.S.    1991
Nu (hD/k) on lines parallel to the stagnation points
nu_line_mean_0 = 173.655035
nu_line_mean = 166.310930
nu_line_mean = 154.939467
nu_line_mean = 142.009972
nu_line_mean = 128.525491
nu_line_mean = 115.096452
nu_line_mean = 102.122262
nu_line_mean = 89.863696
nu_line_mean = 78.482532
nu_line_mean = 68.067523
nu_line_mean_10 = 58.653241
Area mean htc                                 Options
nu_area_mean = 116.157246             Display selected correlations
h = 1002.620443                       Evaluate selected correlations
                                      Broaden Search
                                      Check correlation validity
                                      Quit
```

Figure A-23.

It is desirable to select further correlations which can then be evaluated in order to provide a comparison with this value. The option **'broaden search'** is selected to make use of a knowledge base which will modify descriptions of heat transfer geometries and flow conditions, these modified descriptions may then enable the selection of further correlations.

```
Jet Impingement Heat Transfer Database

          Automatic Knowledge Based Broadening of Search.
          ----------------------------------------------
15 correlations selected in 0.330000secs.
                                                  Press any key ...
Summary of results of Broadening search
Search No. 1. Number of records selected 15
Keywords:
array forced_convection jet_impingement circular flat orifice
Select a list of correlations for further investigation.
Enter the search number:1

Adding parameter equtions from the external database.
Equation for a added.
Equation for a added.
Equation for af added.
Equation for af added.
Equation for ap added.
Equation for d added.
Equation for g added.
Equation for gp added.
```

Figure A-24.

The original description of the jet impingement geometry and flow conditions is examined and, provided a suitable alternative is available in the knowledge base, is replaced with a different description. The knowledge base contains rules which identify whether, for a given qualitative description and for a given range of values for each of the non-dimensional groups, an alternative description can be substituted. By the appropriate specification of rules of this type alternative descriptions can be substituted which should enable the selection of correlations which are expected to yield a good approximation to the heat transfer coefficient.

In this example the description of a row of jets is replaced by that of an array of jets for which the heat transfer can be evaluated for the first row in the array. In addition to the keywords used to describe the heat transfer process this also involves the creation of a new list of parameters and the derivation of new parameters from the old parameters. In this example all the new parameter values can be derived from the previous set of values.

```
Jet Impingement Heat Transfer Database
lx/d = 11.842105 in limits
z/d = 4.440789 in limits
Search rule from Jambunathan, Lai, Moss and Button 1992
Relaxed limits satisfied for 4*m/d/pi/mu. Significance low
                                                          Press any key...
Kercher,D.M. and Tabakoff,W. 1970, correlation b.
ly/d = 5.072368 in limits
nx = 1 in limits
lx/d = 11.842105 in limits
z/d = 4.440789 in limits
Search rule from Jambunathan, Lai, Moss and Button 1992
rho*v*d/mu = 52169.689459 too high, upper limit 3000.000000
                                                          Press any key...
Kercher,D.M. and Tabakoff,W. 1970, correlation a.
ly/d = 5.072368 in limits
nx = 1 in limits
lx/d = 11.842105 in limits
z/d = 4.440789 in limits
Search rule from Jambunathan, Lai, Moss and Button 1992
Relaxed limits satisfied for rho*v*d/mu. Significance low
                                                          Press any key...
1 correletions accepted, 14 correlations rejected
                                                          Press any key ...
```

Figure A-25.

Fifteen correlations with descriptions matching the new keywords were selected from the database. The options **'Check correlation validity'** followed by **'Use the knowledge base to broaden the limits'** were then selected. The Knowledge Base is used to determine that extrapolation for Reynolds number is expected to result in a small loss in accuracy. Extrapolation is required because all the experiments were performed at a low Reynolds number, one correlation was found to satisfy the relaxed limits.

```
Jet Impingement Heat Transfer Database
EVALUATE THE SELECTED CORRELATIONS
----------------------------------

Kercher,D.M. and Tabakoff,W.    1970
coeff = 0.003382
Enter a value for mi :0

coeff1 = 1.002500
exp = 0.961421
Area mean htc
nu = 117.731707
h = 1016.210525
                                        Press any key ...

                              Options
                              Retrieve initial selection
                              Display selected correlations
                              Evaluate selected correlations
                              Broaden Search
                              Check correlation validity
                              Quit
```

Figure A-26.

The selected correlation was then evaluated, this correlation produced an area mean heat transfer coefficient of 1016 W/m^2K. This result was sufficiently close to that given by the first correlation (1003 W/m^2K) to increase confidence that this was a reasonable result.

APPENDIX B
A SAMPLE RUN OF THE JET IMPINGEMENT KNOWLEDGE BASED DATABASE SYSTEM - ADDING NEW CORRELATION DATA

Copies of selected screen output are presented to show the use of the KBDS to add a new correlation to the database. The correlation is for free convection from a hot vertical plate, this is a type of heat transfer process not previously given in the database. So that the KBDS can in future provide assistance in obtaining correlations of this type from the database the information used by the KBDS in this process must also be updated.

```
═══════════╡Heat Transfer Knowledge Based Database - Update Program╞═══════════

                Input new correlation data into the database.
                ---------------------------------------------
    This program will offer guidance in inputting new correlations, the
information entered will be checked to ensure that it is compatible with
the database access program
    A typical run of the program may consist of three stages:
(1) A description of the heat transfer process is built up using keywords,
    this description should use existing keywords where possible. However
    new keywords can be used to describe additional types of heat transfer
    process if required.
(2) The specification of:
        Additional variables required to give a precise definition of the
        process.
        Any equations allowing the values of these variables to be derived
        from the values of other variables.
        Recommended non-dimensional expressions to be used to define the
        limits of validity.
(3) The correlation information. This consists of the a reference, the
    limits of validity and the correlation statements.

 Copyright (c) 1993 Michael A Moss and Nottingham Trent University
 All rights reserved
═══════════════════════════════════════════════════════════ Press any key...
```

Figure B-1

The first screen gives a short introduction to the purpose of the program.

```
Heat Transfer Knowledge Based Database - Update Program
                                              Update Options
                                              Keywords
                                              Correlation
                                              Variable(s)
                                              Variable Equation
                                              Non-dimensional Expr
                                              Summary of new records
                                              Read old Summary
                                              Save the new records
                                              Clear all new records
                                              Quit
```

Figure B-2

A qualitative description of the heat transfer process will be built up first using keywords. The option '**Keywords**' is selected to achieve this. Because the new correlation, for free convection from a hot vertical plate, is a type of heat transfer process not previously given in the database new keywords and variable names must be specified. Once this has been completed it is then possible to provide the statements forming the correlation. This example run of the program will also illustrate the addition to the database of an equation relating the value of a variable to the values of other variables and of a non-dimensional expression to be used when defining the range over which the correlation is valid.

A description is made up using the keywords already available if possible; where the existing keywords are not appropriate new keywords may be specified. Two options are provided to help the user assess whether the existing keywords are adequate, these are '*expand*' and '*draw_tree*'; *expand* provides a brief explanation of the meaning of the keyword, *draw_tree* enables the user to browse the keyword hierarchy below the current position in the hierarchy. An example of the output produced by the program in response to the option '*draw_tree*' is given in Appendix A (Figure A-3).

```
Heat Transfer Knowledge Based Database - Update Program
Select or add keywords
Enter either existing keywords or new keywords
A  Keyword information
                                                          Press RETURN

   Keywords available to further specify convection

Y  You may select one keyword from each of the following groups:
s  Group 1:
1  forced_convection:
E  FORCED CONVECTION occurs due to an externaly imposed flow, it genarally
   results in high levels of heat transfer.
   free_convection:
   FREE CONVECTION is due to a flow generated by a temperature difference, if
   the fluid is heated then the fluid will rise due to its increased
   buoyancy.
```

Figure B-3

The user's input and responses to prompts are shown in **bold** type. In response to the request '*expand*' the program displays the text expanding on the meaning of the currently available keywords, given here as Figure B-3.

```
Heat Transfer Knowledge Based Database - Update Program
Select or add keywords
Enter either existing keywords or new keywords
Additional options:
    'draw_tree' to display the existing hierarchy
    'expand' to obtain more information about the keywords
    RETURN to end the selection of keywords in a branch

You may select a keyword from each of the following groups to give a more
specific description of 'convection'
1)   forced_convection free_convection
Enter keyword(s), option or RETURN: expand

Enter keyword(s), option or RETURN: free_convection

No keywords are currently available to give a more specific
description of 'free_convection'
Enter keyword(s), option or RETURN: constant_temperature flat_plate heated_s
urface
Enter text to explain the keyword 'constant_temperature': The heat transfer
surface is at a uniform temperature
Enter text to explain the keyword 'flat_plate': A plane surface
```

Figure B-4

Figures B-4 and B-5 illustrate how a heat transfer process can be described using keywords, the user selects '**free_convection**' from the available keywords, and then proceeds to specify the new keywords '**constant_temperature**', '**flat_plate**' and '**heated_surface**'. These new keywords will be added to the keyword network. For each new keyword the program also prompts for text which will be used to help clarify the meaning of the keyword to a user in the future, this text will also be added to the keyword network.

These three keywords can be meaningfully combined in a description of a heat transfer process. Keywords which cannot be combined in a description, for example '*forced_convection*' and '*free_convection*' are linked together in a group. Figure B-14 illustrates the addition of a new keyword to a group.

```
Heat Transfer Knowledge Based Database - Update Program
Select or add keywords
Enter text to explain the keyword 'flat_plate': A plane surface
Enter text to explain the keyword 'heated_surface': The heat transfer surfac
e is at a higher temperature than the ambient temperature

No keywords are currently available to give a more specific
description of 'constant_temperature'
Enter keyword(s), option or RETURN:

No keywords are currently available to give a more specific
description of 'flat_plate'
Enter keyword(s), option or RETURN: vertical
Enter text to explain the keyword 'vertical': The force of gravity acts para
llel to the surface

No keywords are currently available to give a more specific
description of 'vertical'
Enter keyword(s), option or RETURN:

No keywords are currently available to give a more specific
description of 'heated_surface'
Enter keyword(s), option or RETURN:
```

Figure B-5

The new keyword '**vertical**' is used to qualify the type of 'flat plate'.

```
Heat Transfer Knowledge Based Database - Update Program
Select or add keywords
No keywords are currently available to give a more specific
description of 'heated_surface'
Enter keyword(s), option or RETURN:

Keywords selected:
[free_convection,constant_temperature,flat_plate,heated_surface,vertical]

Additions to keyword database.
------------------------------
 subset_new(3,35).
 subset_new(3,36).
 subset_new(3,37).
 subset_new(36,38).
 label_new(35,"constant_temperature","The heat transfer surface is at a unif
orm temperature").
 label_new(36,"flat_plate","A plane surface").
 label_new(37,"heated_surface","The heat transfer surface is at a higher tem
perature than the ambient temperature").
 label_new(38,"vertical","The force of gravity acts parallel to the surface"
).

                                                           Press any key...
```

Figure B-6

Once the user has built-up the description of the heat transfer process the program produces a summary of the newly created clauses which define the additional portion of the keyword network. These clauses must be added to the database if the user requires a permanent record of the enlarged network.

Selection of the option **'Save the new records'** in the main menu (Figure B-2) causes the program to write formatted files containing the new information, these files can then be appended to the database files. The format of the files is suitable for use with dBase iv (Ashton Tate 1988).

The option **'Summary of new records'** causes the program to write each of the new records to a file, these records are shown as Prolog clauses (Figure B-13). This file can be viewed from within the program. The information in this file, which forms a record of the data entered to the Knowledge Base update program, can be read back into the program be selecting the option **'Read old summary'**.

As the correlation is for a new type of heat transfer new parameters will be required to express both the correlation and the range over which the correlation can be considered valid. The user selects the option **'Variable(s)'** from the main menu (Figure B-2).

```
═══════╡Heat Transfer Knowledge Based Database - Update Program╞═══════
══════════════════╡Add new variable names╞══════════════════

Enter a new variable name or RETURN to end:tw
Enter text to define the variable (max. 80 chars)
Surface temperature (K)
Do you want the user to be prompted for its value? y/n:y
```

Figure B-7

The user specifies the variable or parameter name, **tw**, the text **'Surface temperature (K)'** giving its definition and whether or not the database access program should prompt for the value. The new variable will be incorporated in the keyword network, the keywords to which the variable will be linked must be specified (Figure B-8). It will be automatically selected whenever a description is formed which includes all the keywords which are associated with the variable. The variable must be selected to allow a value for it to be entered (a necessary part of the process of retrieving a correlation from the database) and for it to be used in any expressions (correlations for example).

```
Heat Transfer Knowledge Based Database - Update Program
Add new variable names

Enter a new variable name or RETURN to end:t
Enter text to define the variable (max. 80 c
Surface temperature (K)
Do you want the user to be prompted for its

This value/relationship can
be made more general by removing
keywords from the description
Select keywords to be removed

Keywords
Accept keywords
Original keywords
free_convection
constant_temperature
flat_plate
heated_surface
vertical
```

Figure B-8

The list of keywords given in Figure B-8 as a default is the full list entered by the user. The variable can be made to apply to a wider range of heat transfer processes by making the description of a heat transfer geometry and flow process formed by these keywords more general, this is achieved by eliminating keywords from the list. The update program ensures that the combination of keywords in this list is valid. For example if **'flat_plate'** is selected then both *'flat_plate'* and *'vertical'*, which is below *'flat_plate'* in the keyword hierarchy, will be removed from the keyword list to be associated with the new variable. In this example **'heated surface'** is also removed from the keyword list.

Further parameters which are expected to be required in defining the correlations and the limits can be entered. To proceed with the example the ambient temperature (**ta**), the gravitational constant (**g**), the vertical plate hight (**l**), and the gas properties of air at the film temperature on the heat transfer surface should be specified at this point. Definitions can be found in the summary (Figure B-13).

Selection of the option **'Variable Equation'** (Figure B-2) enables the user to specify arithmetic procedures relating any one of the variables to one or more of the other variables. These procedures are used by the KBDS to automatically derive values for parameters where possible ,so saving a user the task of deriving and specifying these values when searching for a correlation in the database. The keywords in the netword to which the equation will be attached are specified through a menu similar to that in Figure B-8.

```
Heat Transfer Knowledge Based Database - Update Program
Add a new variable equation
Add equations/procedures giving inter-relations between variables
Enter name of variable to be derived:tf

Result variable tf not known.
This variable must be defined before this rule can be accepted

Press any key...
```

Figure B-9

The KBDS checks that the variables used in the expressions have been defined and would be selected from the network by the KBDS if a description of the geometry and flow was specified which included the keywords (*'free convection'* and *'constant temperature'*). Figure B-9 shows the message displayed when an unknown variable **'tf'** is used. The variable can be added by selecting the **'Variable(s)'** option (Figure B-2). Once this is done the variable equation (**tf=(tw-ta)/2**) can be re-entered.

By selecting **'Non-dimensional Expr'** From the main options menu (Figure B-2) the user may elect to enter non-dimensional expressions. Each correlation will be valid over a limited range of values for the significant variables. The ranges of these values are most appropriately expressed as minimum and maximum values for non-dimensional groups. Expressions entered at this point will be used to offer recommended non-dimensional groups for use in specifying the limits of validity of new correlations (see Figure B-11).

```
═══════════╡Heat Transfer Knowledge Based Database - Update Program╞══════════
  ═════════════════╡Add new non-dimensional expressions╞══════════════════
 Add recommended non-dimensional groups for use in limits of validity
 Enter non-dimensional expression or RETURN to end:g*l**3*(tw-ta)*rho_f**2/ta
 /mu_f**2*pr_f
 Enter text to define the expression:Product Grashof and Prandtl numbers
 Enter non-dimensional expression or RETURN to end:

      Additions to non_dimensional expressions.
      -----------------------------------------
  non_dim_new("g*l**3*(tw-ta)*rho_f**2/ta/mu_f**2*pr_f","Product Grashof and
 Prandtl numbers","free_convection,constant_temperature,flat_plate,vertical")
 .

═════════════════════════════════════════════════════════════ Press any key...
```

Figure B-10

The keywords *'free convection'*, *'constant temperature'*, *'flat plate'* and *'vertical'* (specified as before through a menu Figure B-8) will be used to place the non-dimensional expression in the keyword keywork. Checks are performed on the expression. Variables which are undefined or would not be selected by the KBDS if a description of the geometry and flow was specified which included the keywords *'free convection'*, *'constant temperature'*, *'flat plate'* and *'vertical'* will be identified.

To add the correlation statements and the information on the range of parameter values or non-dimensional groups over which the correlation is valid the user selects the option '**Correlation**' from the main menu.

```
Heat Transfer Knowledge Based Database - Update Program
Add new correlation information

Enter Authors' names:Eckert ERG
Enter year of publication:1952

Enter article title:Introduction to the Transfer of Heat and Mass
Enter publication details:New York:McGraw-Hill, 1951, quoted from Welty,J.R.
, Engineering Heat Transfer, Wiley, 1975.

You will be prompted for non-dimensional expressions and corresponding
minimum and maximum limits. Recommended expressions for this geometry are:
g*l**3*(tw-ta)*rho_f**2/ta/mu_f**2*pr_f Product Grashof and Prandtl numbers

Enter an expression or RETURN to end:g*l**3*(tw-ta)*rho_f**2/ta/mu_f**2*pr_f

Enter minimum expression value :0
Enter maximum expression value :1000000000

Enter an expression or RETURN to end:
```

Figure B-11

The user enters the authors' names and the year in which the correlation was published. He then enters the limits over which the correlation is valid, one non-dimensional group (GrPr) is specified along with its minimum and maximum permissible values. The user may enter any number of limits, these may be either the recommended expressions or other expressions if required. Once more checks are performed on the expressions to check that the variables used are appropriate to the keywords provided by the user and that the expression uses valid syntax.

```
Heat Transfer Knowledge Based Database - Update Program
Add new correlation information
g*l**3*(tw-ta)*rho_f**2/ta/mu_f**2*pr_f Product Grashof and Prandtl numbers

Enter an expression or RETURN to end:g*l**3*(tw-ta)*rho_f**2/ta/mu_f**2*pr_f

Enter minimum expression value :0
Enter maximum expression value :1000000000

Enter an expression or RETURN to end:

You will be prompted for the statements making up a correlation.
Enter these one line at a time. RETURN to end.
Enter a new statement :gr = g*l**3*(tw-ta)*rho_f**2/ta/mu_f**2*pr_f
Enter a new statement :a = 0.678*pr_f**0.5*gr**0.25
Enter a new statement :b = (0.952 + pr_f)**0.5
Enter a new statement :nu = a/b
Enter a new statement :write "area mean Nusselt no."
Enter a new statement :write nu
Enter a new statement :h = nu*k_f/l
Enter a new statement :write "area mean heat transfer coefficient"
Enter a new statement :write h
Enter a new statement :
```

Figure B-12

Once the range over which the correlation is believed to be appropriate has been specified the user may enter the correlation statements. These will be used to derive and output a value for the heat transfer Nusselt number if this correlation is selected from the database.

Figure B-13. A Summary of the new Clauses Added in this Example

```
        /*Summary of new records to be added to the database*/
        /*--------------------------------------------------*/

/*Additions to the keyword network*/
/*--------------------------------*/
/*      New keywords      */
 label_new(35,"constant_temperature","The heat transfer surface is at a uniform
temperature").
 label_new(36,"flat_plate","A plane surface").
 label_new(37,"heated_surface","The heat transfer surface is at a higher tempera
ture than the ambient temperature").
 label_new(38,"vertical","The force of gravity acts parallel to the surface").
```

```
/*      New network links       */
/*      Links from more general to more specific keywords       */
 subset_new(3,35).
 subset_new(3,36).
 subset_new(3,37).
 subset_new(36,38).
/*keywords substituted for node numbers*/
/*
 subset_new_display("free_convection","constant_temperature").
 subset_new_display("free_convection","flat_plate").
 subset_new_display("free_convection","heated_surface").
 subset_new_display("flat_plate","vertical").
*/
/*      The following pairs of keywords cannot be combined in a query       */
/*keywords substituted for node numbers*/
/*
*/
/*      Additions to the variables      */
 diction_new("g","Acceleration due to body force eg. Gravity (m/s/s) (Earth's g
 ravity 9.81)","free_convection","y").
 diction_new("rho_f","Density in surface film (kg/m/m/m)","free_convection,cons
 tant_temperature","n").
 diction_new("mu_f","Viscosity in the surface film (Ns/m/m)","free_convection,c
 onstant_temperature","n").
 diction_new("cp_f","Specific heat in the surface film (j/kg/K)","free_convecti
 on,constant_temperature","n").
 diction_new("k_f","Conductivity in the surface film (W/m/K)","free_convection,
 constant_temperature","n").
 diction_new("pr_f","Prandtl no. in the surface film","free_convection,constant
 _temperature","n").
 diction_new("tw","Surface temperature (K)","free_convection,constant_temperatu
 re","y").
 diction_new("ta","Ambient temperature (K)","free_convection","y").
 diction_new("l","Plate hight (m)","free_convection,flat_plate,vertical","y").
 diction_new("tf","Film temperature (K)","free_convection,constant_temperature"
 ,"n").
 diction_new("pa","Ambient pressure (Pa)","free_convection","y").
/*      Equations inter-relating these variables        */
 eqn_new("tf","free_convection,constant_temperature",1,1,"tf=(tw+ta)/2").
 eqn_new("cp_f","free_convection,constant_temperature",1,1,"a0=1049.7").
 eqn_new("cp_f","free_convection,constant_temperature",1,2,"a1=-0.383117*tf").
 eqn_new("cp_f","free_convection,constant_temperature",1,3,"a2=0.000896933*tf**
 2").
 eqn_new("cp_f","free_convection,constant_temperature",1,4,"a3=-0.000000400266*
 tf**3").
 eqn_new("cp_f","free_convection,constant_temperature",1,5,"a4=-0.0000000000314
 785*tf**4").
 eqn_new("cp_f","free_convection,constant_temperature",1,6,"cp_f=a0+a1+a2+a3+a4
 ").
 eqn_new("mu_f","free_convection,constant_temperature",1,1,"a0=0.000001457").
 eqn_new("mu_f","free_convection,constant_temperature",1,2,"a1=110.4").
 eqn_new("mu_f","free_convection,constant_temperature",1,3,"mu_f=a0*tf**1.5/(tf
 +a1)").
 eqn_new("k_f","free_convection,constant_temperature",1,1,"cv=cp_f-287.05").
 eqn_new("k_f","free_convection,constant_temperature",1,2,"gam=cp_f/cv").
 eqn_new("k_f","free_convection,constant_temperature",1,3,"k_f=mu_f*cv*(1.758*g
 am-0.43)").
```

```
 eqn_new("pr_f","free_convection,constant_temperature",1,1,"pr_f=cp_f*mu_f/k_f"
 ).
 eqn_new("rho_f","free_convection,constant_temperature",1,1,"rho_f=pa/gas_const
/tf").
/*     Recommended non-dimensionals for limit expressions     */
 non_dim_new("g*l**3*(tw-ta)*rho**2/ta/mu**2*pr_f","Product Grashoff and Prandt
 l numbers","free_convection,constant_temperature,flat_plate,vertical").

/*     Additional correlations     */
/*     -----------------------     */
/*     Qualitative description     */
 descript_new("Eckert_51","a","free_convection,constant_temperature,flat_plate,
heated_surface,vertical").
/*     Reference     */
 ref_new("Eckert_51","Eckert ERG","1951","Introduction to the Transfer of Heat
and Mass","New York:McGraw-Hill, 1951, quoted from Welty,J.R., Engineering Heat
 Transfer, Wiley, 1975.").
/*     Limits of validity     */

limits_new("Eckert_51","a","g*l**3*(tw-ta)*rho**2/ta/mu**2*pr_f",0.000000,1000
000000.0).
/*     Correlation expressions     */
 corr_new("Eckert_51","a",1,"gr = g*l**3*(tw-ta)*rho**2/ta/mu**2*pr_f").
 corr_new("Eckert_51","a",2,"a = 0.678*pr_f**0.5*gr**0.25").
 corr_new("Eckert_51","a",3,"b = (0.952 + pr_f)**0.25").
 corr_new("Eckert_51","a",4,"nu = a/b").
 corr_new("Eckert_51","a",5,"write \"area mean Nusselt no.\"").
 corr_new("Eckert_51","a",6,"h = nu*k_f/l").
 corr_new("Eckert_51","a",7,"write \"area mean heat transfer coefficient\"").
 corr_new("Eckert_51","a",8,"write h").
```

Figure B-13 shows the file which gives a summary of the new clauses. These clauses must be transferred to formatted files (option **'Save the new records'**) and appended to the database files to give a permanent record of the modifications to the keyword network and of the new correlation.

The database keys 'Eckert_56' and 'a' uniquely identify the information specific to this correlation, these keys are generated automatically. The first key identifies the source of the correlation, the second identifies each correlation taken from this source.

```
Heat Transfer Knowledge Based Database - Update Program
Select or add keywords
Enter either existing keywords or new keywords
Additional options:
    'draw_tree' to display the existing hierarchy
    'expand' to obtain more information about the keywords
    RETURN to end the selection of keywords in a branch

You may select a keyword from each of the following groups to give a more
specific description of 'convection'
1)   forced_convection free_convection
Enter keyword(s), option or RETURN: free_convection

You may select a keyword from each of the following groups to give a more
specific description of 'free_convection'
1)   constant_temperature
2)   flat_plate
3)   heated_surface
Enter keyword(s), option or RETURN: constant_temperature cylinder
Enter text to explain the keyword 'cylinder': Length large compared with dia
meter
```

Figure B-14

A group of keywords is formed by keywords which cannot be combined in a single description. These groups are expressed in the network by links between the keywords. For example 'forced_convection' and 'free_convection' form such a group. Groups may also contain only one keyword. Where one or more groups of keywords are offered to the user, and where the user considers that the alternative keywords do not adequately describe the heat transfer process, then the user may enter a new keyword. New keywords may be combined with existing keywords, for example a new keyword '**cylinder**' may be combined with the existing keyword '**constant_temperature**'.

```
Heat Transfer Knowledge Based Database - Update Program
Select or add keywords
specific description of 'free_convection'
1)    constant_temperature
2)    flat_plate
3)    heated_surface
Enter keyword(s), option or RETURN: constant_temperature cylinder
Enter text to explain the keyword 'cylinder': Length large compared with dia
meter

You may prevent any one of the following groups of keywords being used in
combination with 'cylinder' by entering one keyword from the group
1)    flat_plate
2)    heated_surface
Enter keyword or RETURN to exclude no groups:flat_plate

No keywords are currently available to give a more specific
description of 'constant_temperature'
Enter keyword(s), option or RETURN:

No keywords are currently available to give a more specific
description of 'cylinder'
Enter keyword(s), option or RETURN:
```

Figure B-15

If a new keyword is given then the user is prompted to state whether this new keyword should be incorporated into one of the existing groups. In this case the keyword **'cylinder'** indicates an alternative heat transfer surface geometry to **'flat_plate'** and so is added to that group. Because the keyword **'cylinder'** was specified together with the existing keyword **'constant_temperature'** this group is not offered as one to which the new keyword can be added.

A review of heat transfer data for single circular jet impingement

K. Jambunathan, E. Lai, M. A. Moss* and B. L. Button
Department of Mechanical Engineering, Nottingham Polytechnic, Nottingham, UK

Experimental data for the rate of heat transfer from impinging turbulent jets with nozzle exit Reynolds numbers in the range of 5,000–124,000 have been collated and critically reviewed from the considerable body of literature available on the subject. The geometry considered is that of a single circular jet impinging orthogonally onto a plane surface for nozzle-to-plate distances from 1.2–16 nozzle diameters and over a flow region up to six nozzle diameters from the stagnation point. Existing correlations for local heat transfer coefficient express Nusselt number as a function of nozzle exit Reynolds number raised to a constant exponent. However, the available empirical data suggest that this exponent should be a function of nozzle-to-plate spacing and of the radial displacement from the stagnation point. A correlation for Nusselt number of the form suggested by this evidence has been derived using a selection of the data. The review also suggests that the Nusselt number is independent of nozzle-to-plate spacing up to a value of 12 nozzle diameters at radii greater than six nozzle diameters from the stagnation point. The results from a simple extrapolation for obtaining heat transfer coefficients in the wall jet region compare favourably with published data.

Keywords: jet impingement; forced convection correlation; turbulent flow

Introduction

Impinging jets are widely used where high rates of heat transfer are desired. To improve the design of these systems a knowledge of the parameters affecting the heat transfer rate is required.

The heat transfer rate to or from a jet impinging onto a surface is a complex function of many parameters: Nusselt number (Nu), Reynolds number (Re), Prandtl number (Pr), the nondimensional nozzle-to-plate spacing (z/D), and the nondimensional displacement from the stagnation point (x/D). In addition, the effects of nozzle geometry, flow confinement, turbulence, recovery factor, and dissipation of jet temperature have all been shown to be significant. A selection of publications that consider the effect of these parameters on the heat transfer rate and empirical investigations into the flow phenomena in an impinging jet are reviewed.

The work of Martin (1977), Obot *et al.* (1979a), and Goldstein and Franchett (1988) among others has resulted in the establishment of empirical correlations for evaluating either local or area mean heat transfer. These correlations do not take account of an expected radial variation in the effect of Reynolds number, which is expected to occur as the flow and thermal boundary layers develop.

In many cases the heat transfer data presented only extends a few nozzle diameters from the stagnation point. A means of extrapolating data to give heat transfer rates in the wall jet is required.

*Present address: Rolls-Royce plc, Derby, UK.
Address reprint requests to Dr. Jambunathan at the Department of Mechanical Engineering, Nottingham Polytechnic, Nottingham, NG1 4BU, UK.

Received 20 August 1990; accepted 27 January 1992

A literature review of single circular jet impingement heat transfer

Jet flow characteristics

The flow in an air jet impinging orthogonally on a plane surface is commonly divided into four zones (Figure 1):

(1) There is a developing flow zone where fluid from the surroundings is entrained into the jet, thus reducing the jet velocity. This mixing or shear region surrounds a core where the fluid velocity at the nozzle centerline (U_m) is almost equal to the nozzle exit velocity (U_n). The core region is often referred to as the potential core even though the flow is not inviscid. A common definition of the end of the core region is the point where $U_m = 0.95U_n$. A core length of six nozzle diameters has been suggested by Gautner *et al.* (1970) based on a survey that covered a large range of turbulent Reynolds numbers.

(2) At greater nozzle-to-plate spacings the axial velocities reduce with increasing distance from the nozzle exit. An analysis by Schlichting (1968) showed that the fall of the centerline velocity and the jet half width (width of the jet where $U = U_m/2$) will be directly proportional to the axial distance from the end of the potential core.

(3) The region near the impingement plate is often referred to as the deflection zone where there is a rapid decrease in axial velocity and a corresponding rise in static pressure. The measurements of Tani and Komatsu (1966) showed that this zone extends approximately two nozzle diameters from the plate surface. Giralt *et al.* (1977) proposed a value of $1.2D$ for the height of the deflection zone.

(4) Within the wall jet the transverse velocity profile shows

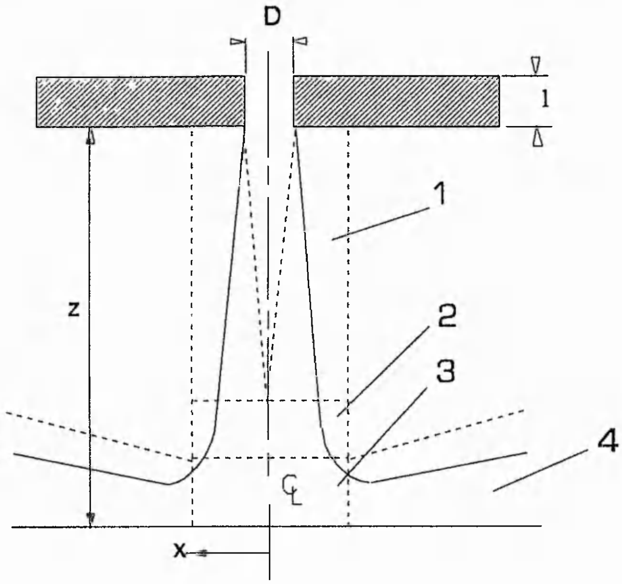


Figure 1 Flow zones in an impinging jet. Zone 1, initial mixing region; zone 2, established jet; zone 3, deflection zone; zone 4, wall jet

that the local velocity rises rapidly to a maximum near to the wall and then falls at greater distances from the wall. The wall jet exhibits higher levels of heat transfer than parallel flow. This appears to be due to turbulence generated by the shear between the wall jet and the ambient air, which is transported to the boundary layer at the heat transfer surface.

Heat transfer characteristics

An analytical solution for heat transfer at a stagnation point in a laminar flow (Sibulkin 1952) shows that

$$Nu \propto U_m^{1/2}$$

suggesting that the Nusselt number should remain roughly

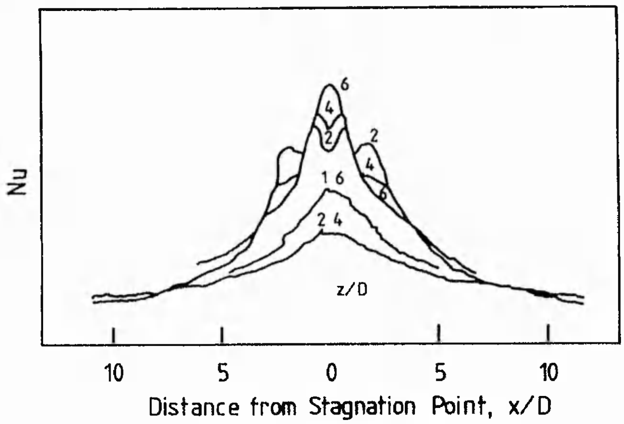


Figure 2 Radial variation of heat transfer coefficient between a plate and an impinging jet. (Data from Gardon and Carbonpue 1962)

constant in the core region and reduce downstream of the core.

The axial variations of velocity and turbulence for slot jets issuing into an unconfined environment were investigated by Gardon and Akfirat (1965). Their results showed that the turbulence intensity in a free jet could reach 30 percent of nozzle exit velocity at approximately eight nozzle diameters downstream of the nozzle. The axial distance from the nozzle at which the maximum turbulence intensity occurs appears to coincide with that of the maximum stagnation point heat transfer. Measurements using circular nozzles have also given similar results; e.g., Schlunder and Gnielinski (1967) found both the maximum turbulence and the maximum stagnation point heat transfer occurring at $z/D = 7.5$. Gardon and Akfirat suggested that the increasing level of turbulence causes the heat transfer rate at the stagnation point to increase even beyond the end of the potential core where the jet centerline velocity is falling. The increase in heat transfer rate ceases when the increase in turbulence does not compensate for the fall in the jet velocity.

The radial variation of heat transfer coefficients measured by Gardon and Corbonpue (1962) are given in Figure 2. These curves show local maxima in the heat transfer rate at the lower

Notation

a	Constant
Cp	Specific heat, w/Kg K
D	Nozzle exit diameter, m
D_{AB}	Diffusion coefficient, m^2/s
h	Heat transfer coefficient, w/m^2 K
k	Coefficient
l	Nozzle length, m
Nu	hD/λ
Pr	Prandtl number
Pu	Potential core length, m
Q	Heat flux, w/m^2
Re	Reynolds number, $D\, U_n/\nu$
Sc	Schmidt number, $\mu/\rho D_{AB}$
Sh	Sherwood number, kD/D_{AB}
T	Temperature, K
t	Static temperature, K
Tu	Turbulence intensity, percentage main velocity component
U	Velocity, m/s
z	Axial distance nozzle to plate, m
x	Radial distance from stagnation point, m

Greek symbols

λ	Conductivity air, w/m K
μ	Viscosity, Ns/m^2
ν	Kinematic viscosity, m^2/s
ρ	Density, kg/m^3

Subscripts

amb	Ambient
aw	Adiabatic wall
m	Centerline value
n	Nozzle exit
rec	Recovery
s	Stagnation point
w	Wall
1/2	Half width

values of nozzle-to-plate spacing ($z/D < 6$). For impinging circular jets at small z/D the heat transfer coefficient is seen to increase between the stagnation point and approximately $x/D = 0.5$. A second maximum is produced by circular jets at approximately $x/D = 2$. Gardon and Carbonpue also reported that at lower Reynolds numbers three maxima were visible at radial displacements of 0.5, 1.4, and 2.5 nozzle diameters from the stagnation point. As the Reynolds number was increased to approximately 20,000 the two outer maxima merged and only two maxima were seen. The presence of these three peaks in the radial distribution of heat transfer was confirmed by Popiel and Boguslawski (1988).

Three causes for the peaks in the radial distribution of heat transfer have been proposed:

The peak at $x/D = 0.5$ has been explained by the change in radial velocity, with displacement from the stagnation point. The radial velocity accelerates rapidly in the deflection region, but at greater radii the spreading of the jet causes the radial velocity to reduce.

Gardon and Akfirat (1965) suggested that the increase in heat transfer coefficient within the range $1 \leqslant x/D \leqslant 2$ can be explained by a transition from laminar to turbulent boundary-layer flow. At greater radial distances from the stagnation point a fall in the heat transfer rate occurs because of the fall in flow velocity in the radial direction. This mechanism will not apply to impinging jets that are fully developed or to impinging jets at large nozzle-to-plate spacings, which result in turbulent boundary-layer flow at the stagnation point. Velocity measurements (Launder 1991) in the boundary layer of an impinging jet for $z/D = 2$ show the presence of turbulence at the stagnation point.

The position of the maximum at $x/D = 2$ coincides approximately with the point where toroidal vortices, which form in the shear region around the circumference of the jet, strike the plate as highlighted in the flow visualization carried out by Popiel and Trass (1982) at $z/D = 1.2$ and 2. Popiel and Trass (1991) suggested that the vortices cause a pulsation in the jet that is responsible for the synchronized appearance of the next toroidal vortex around the entire nozzle circumference. They also show that ring-shaped eddies are formed in the wall jet at approximately the point where the toroidal vortices reach the plate.

Flow visualization photographs obtained by Yokobori *et al.* (1979) also show these toroidal vortices being convected to the impingement plate. These vortices coalesce, and for $z/D > 4$ the large-scale vortex structure appears to break down into small-scale random turbulence that penetrates to the jet axis. Radial movement of the stagnation point of the order of one nozzle diameter was also reported. However, a potential core exists for $z/D < 4$ where the vortices do not penetrate. The penetration of the vortices to the geometric stagnation point and the radial movement of the stagnation point help to break down any distinct flow regions. This may explain why the secondary maxima in the radial distribution of heat transfer coefficient become indistinct and the profiles begin to assume a bell shape for nozzle-to-plate spacings greater than $z/D = 4$.

The apparent disappearance of the large-scale vortex structure at larger nozzle-to-plate spacings is probably due to mixing of the smoke or dye used in these visualizations. Tso and Hussain (1989), using a grid of hot-wire anemometers, showed that ring, helical, and double helical vortex structures exist at large nozzle-to-plate spacings ($z/D = 50$). The helical structure was found to be dominant, this structure appeared to be responsible for significant entrainment of ambient air and jet spreading. The effects of vortices in the deflection zone near to the stagnation point have been detected by Kataoka *et al.* (1987, 1988) using a hot-wire anemometer. They showed a correlation between stagnation point Nusselt number and a "surface-renewal" parameter that is proportional to the frequency and magnitude of these vortices, the correlation was obtained for z/D between 2 and 10.

The flow visualizations of Popiel and Trass, Tso and Hussain, Yokobori *et al.*, and Kataoka *et al.* were conducted using jets issuing from elliptical or bell-shaped convergent nozzles. Popiel and Trass (1991) suggested that a laminar boundary layer formed in nozzles of these types is essential to the formation of a well-defined vortex structure. They gave results for a jet issuing from a nozzle that was designed to disrupt this laminar layer; the visualization shows a much less distinct vortex structure. This suggestion is supported by Lepicovsky (1989), who studied the effect of nozzle exit boundary-layer thickness on the length of the potential core. This study suggested that a thin laminar boundary layer at the nozzle exit resulted in higher rates of jet mixing than a thick boundary layer. However, turbulence measurements made by Yokobori *et al.* at the nozzle exit suggest that the vortex structure can still be obtained when the boundary layer at the nozzle exit is turbulent.

Causes of scatter in the experimental data

Nozzle geometry. Obot *et al.* (1979b, 1979c, 1982) suggested that the variations in turbulence levels due to different nozzle designs have brought about the variations in measured heat transfer rate, particularly in the optimal nozzle-to-plate spacing for maximum heat transfer. For a nozzle with a sharp-edged inlet and a length-to-diameter ratio of 1 (Re = 29,485) the maximum stagnation point Nusselt number, equal to 155, occurs at $z/D = 4$, and with a contoured inlet the maximum, Nu = 125, occurs at $z/D = 8$. The data given by Popiel and Boguslawski (1988) show that the area mean heat transfer due to an impinging jet issuing from a contoured ASME nozzle is 25 percent less than that of a jet from a sharp-edged orifice when $z/D = 4$, Re = 20,000, and $x/D = 1$. The effect of nozzle geometry on heat transfer is most significant in the region near the stagnation point. A similar investigation was made by Gundappa *et al.* (1989), who compared the axial velocity decay and the impingement heat transfer due to orifice jets and to jets issuing from pipes ($l/D = 10$). The axial velocity was seen to decay more slowly in the case of the jet produced by the pipe. Gundappa *et al.* suggested that this led to the higher values of Nusselt number seen for the impinging pipe jet at all radial positions when $z/D = 7.8$. At smaller nozzle-to-plate spacings the different nozzle designs were seen to produce differing shapes of radial Nusselt number distribution.

Nozzle geometry affects the velocity profile at the nozzle exit, which it has been suggested might be expected to affect the behavior of the toroidal vortices around the jet circumference and the turbulence level generated in the shear layer. The turbulence level would in turn affect the heat transfer coefficients. This turbulent behavior also affects the mixing of the jet with the ambient air and so the rate of velocity decay, which also influences the heat transfer coefficients.

Effect of small-scale turbulence. Den Ouden and Hoogendoorn (1974) and Hoogendoorn (1977), among others, have investigated the effect of small-scale turbulence on heat transfer in impinging jet systems. Hoogendoorn showed that the level of turbulence at the nozzle exit has an impact on the heat transfer at the stagnation point. For instance an increase in the axial turbulence intensity from 0.5–3.2 percent (at Re = 60,000 and $z/D = 2$) has resulted in an increase in the Nusselt number at the stagnation point from 180–215 and has eliminated the local minimum in heat transfer coefficient often

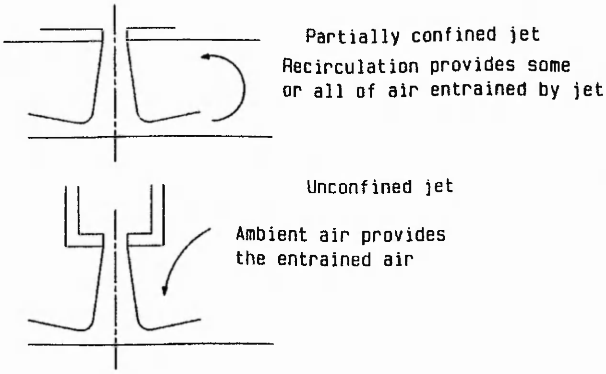


Figure 3 Schematic diagram of confined and unconfined jets

seen at small nozzle-to-plate spacings. However, the results of Gardon and Akfirat (1965) showed that the effect of nozzle exit turbulence on heat transfer is relatively small for $z/D > 6$ where turbulence generated in the shear layer predominates.

Confinement. Obot *et al.* (1982) showed that confinement (Figure 3) causes a reduction in the heat transfer rate. This reduction increases with increasing flow rate. At $z/D = 6$ and $x/D = 6$ a 4 percent reduction in the heat transfer due to flow confinement occurred at a Reynolds number of 29,673 and a reduction of 10 percent occurred at $\mathrm{Re} = 50{,}367$. The data appears to show that this reduction will be least when $z/D = 6$.

The wall jet

Rao and Trass (1964) presented results that suggested that within the range $2.0 \leqslant z/D \leqslant 19.23$ the local mass transfer is independent of nozzle-to-plate spacing at radial positions greater than 4.5 nozzle diameters from the stagnation point. In addition they proposed an expression for Sherwood number, Sh, where

$$\mathrm{Sh} = 1.3\mathrm{Re}^{0.84}(x/D)^{-1.27}$$

Dawson and Trass (1966), by using a momentum integral analysis and assuming a 1/7th power law velocity profile between the wall and the point of maximum velocity in the wall jet, obtained an expression where the Sherwood number is independent of nozzle height:

$$\mathrm{Sh} = 0.0509\mathrm{Re}^{0.832}(x/D)^{-1}$$

This expression was obtained for unity Schmidt number (Sc); the exponents in this relationship are dependent on Schmidt number.

Mitachi and Ishiguro (1977) reported velocity and temperature measurements for a nozzle-to-plate spacing of less than one. They deduced two power laws: a value of between 1/10th to 1/14th for the velocity profile and 1/14th for the temperature profile. By assuming 1/14th power laws for the velocity and temperature profiles they demonstrated that

$$\mathrm{Nu} \propto \mathrm{Re}(x/D)^{-1.08}$$

Hrycak (1978) proposed the following expression based on the Colburn analogy and the wall shear stress derived by Poreh *et al.* (1967):

$$\overline{\mathrm{Nu}} = 1.95\mathrm{Pr}^{0.33}\mathrm{Re}^{0.7}(2x/D)^{-1.23}$$

Figure 4 shows the local heat transfer rate at $x/D = 6$ obtained from the papers summarized in Table 1. This plot supports the suggestion that, for z/D in the range of 2–12, the heat or mass transfer is independent of nozzle-to-plate spacing at sufficiently high radial displacements from the stagnation point. Figure 5 compares the radial distribution of heat transfer in the wall jet region obtained by Goldstein *et al.* (1986) and Hollworth and Gero (1985). It can be seen that for radii greater than about 4 nozzle diameters the local heat transfer is independent of the nozzle-to-plate spacing. Curves fitted to this data for $x/D \geqslant 4.5$ show

$$\mathrm{Nu} \propto (x/D)^{-a}$$

where $0.95 \leqslant a \leqslant 1.02$.

Using the assumption

$$\mathrm{Nu} \propto (x/D)^{-1}$$

based on the wall jet results it can be seen that $\overline{\mathrm{Nu}}/\mathrm{Pr}^{1/3}$ will approach zero asymptotically as x/D increases.

Effect of temperature recovery and of a nonambient jet

Studies were carried out by Hollworth and Wilson (1984) and Hollworth and Gero (1985) using jets heated up to 60°C above ambient temperature. These studies demonstrated that the Nusselt numbers derived using the adiabatic wall temperature are independent of the relative temperature. The adiabatic wall temperature was taken as the temperature that the heat transfer surface assumes when it is in equilibrium with the jet, i.e., when there is no flow of heat between the jet and the heat transfer surface. The results also showed that the axial velocity and temperature profiles downstream of the potential core are very similar. It then follows that the radial profiles of adiabatic wall temperature could be collapsed onto a single curve using the following expression

$$(T_{aw} - T_{amb})/(T_n - T_{amb}) = f(x/x_{1/2})$$

where the jet half width, $x_{1/2}$, is the radius of the free jet at $U = U_m/2$ in the absence of the impingement plate.

At much larger temperature differences (typically up to 300°C) the density of the jet relative to that of the ambient air has a significant effect on the heat transfer coefficients and jet temperature dissipation. Katoaka (1985) correlated the potential core length, Pu, as a function of the Reynolds number and the density ratio.

$$Pu/D = 2.82(\rho_{amb}/\rho_n)^{-0.29}\mathrm{Re}^{0.07}$$

The potential core length reduces if the jet density is lower than the density of the ambient air. Katoaka also showed that the maximum stagnation point heat transfer occurs close to the

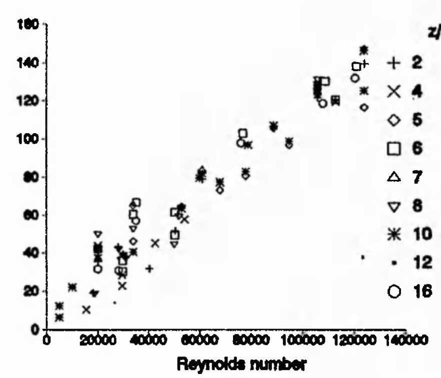


Figure 4 Local heat transfer ($\mathrm{Nu}/\mathrm{Pr}^{1/3}$) at $x/D = 6$ for z/D from 2–16

Table 1 Summary of sources of local heat transfer rates

Authors	Date	Nozzle geometry	Nozzle diameter (mm)	Re	z/D	Data	Other data	Measurement technique
Baughn and Shimizu	1989	Long pipe $l/D = 72$	25	23,750	2–14	Nu		Heated plate, liquid crystals
Baughn *et al.*	1991	Long pipe $l/D = 72$	25	23,300–55,000	2–10	Nu	Effectiveness	Heated plate, liquid crystals
Butler	1984	Contoured	12.7	61,000–124,000	2, 4, 6, 8, 10, 12	Nu	Recovery factor effectiveness	Heated plate
*Button and Wilcock	1982	Orifice $l/D = 0.25$	25.4	25,000	2–6	Nu	Velocity, turbulence	Naphthaline sublimation
Behbahani and Stewart	1989	Sharp orifice	6.35	4,800 10,000	15 10	Sh		Naphthaline sublimation
Chia	1972		32, 19	34,000	1.2–20	Sh	Turbulence, pressures	Naphthaline
Donaldson *et al.*	1971	Contoured	13	30,000–110,000	10–30	htc	Turbulence, pressure	Heat flux gauges
Den Ouden and Hoogendoorn	1974		13, 57	38,000–264,000	1–6	Nu	Turbulence	Heated plate, liquid crystals
*Gardon and Carbonpue	1962	Contoured low l/D	2.3–9	28,000	0.25–24	htc		Heat flux gauges
Gardon and Akfirat	1965	Contoured	0.25 in	2,500–28,000	2	htc	Turbulence	Heat flux gauges
*Goldstein and Behbahani	1982	Pipe $l/D = 7$	12.7	35,200–121,000	6	Nu	Recovery factor	Heated plate
		Orifice $l/D = 1$		35,100–120,500	12			
*Goldstein and Timmers	1982	Orifice $l/D = 1$	10	40,000	2, 6	Nu		Liquid crystals
Goldstein *et al.*	1986	ASME elliptical	12.7	61,000–124,000	2, 5, 8, 10	$Nu/Re^{0.7}$	Recovery factor	Heated plate
Goldstein and Franchett	1988	Orifice $l/D = 1$	10.0	10,000–30,000	4, 6, 10	Nu	Oblique impingement	Heated plate, liquid crystals
Goldstein *et al.*	1990	ASME elliptical	12.7	61,000–124,000	2–12	Nu	Recovery factor effectiveness	Heated plate
Gundappa *et al.*	1989	Pipe $l/D = 10$ Orifice	15.9 15.9	34,000	2.6, 5.2, 7.8	Nu		Heated plate
*Hollworth and Gero	1985	Orifice $l/D = 1$	2.5, 10	5,000–60,000	1, 5, 10, 15	$Nu/Pr^{0.3}$		Heat flux
*Hoogendoorn	1977	Contoured Pipe large l/D	57	66,000	2, 4, 5.9	Nu	Turbulence, pressure	Heated plate, liquid crystals
Katoaka *et al.*	1978	Contoured	28	4,000–15,000	2.5, 6, 8	mtc	Turbulence, velocities	Electro-Chemical
*Kieger	1981	Contoured	12.7	61,000–124,000	4, 5, 10	Nu	Recovery factor	Heated plate
*Lovell	1978	Sharp orifice	0.25 in	2,500–10,000	7, 10, 15	Nu	Impingement angle	Naphthaline sublimation
*Obot *et al.*	1979	Orifice and contoured pipes $l/D = 1–50$	19.05	15,000–54,000	2–12	Nu	Turbulence	Heated plate
*Obot *et al.*	1979	Contoured ASME	10, 20	18,000–52,000	4–24	Local and mean Nu	Turbulence	Heated plate
*Obot *et al.*	1982	Contoured ASME	20	18,000–50,100	2, 4, 6, 8	Nu		
*Popiel and Boguslawski	1988	Contoured and sharp orifice	14, 60 10, 15	4,590–63,100 5,940–44,220	1.2–16	Sh	Turbulence	Naphthaline sublimation
*Schlunder and Gnielinski	1967	Long pipe	20–50	34,000–124,000	1.25, 2.5, 5, 10	Sh		Water vaporization
*Sparrow and Lovell	1980	Sharp orifice	6.35	2,500–10,000	7, 10, 15	Nu	Effect impingement angle	Naphthaline sublimation

* Data used in the present study.

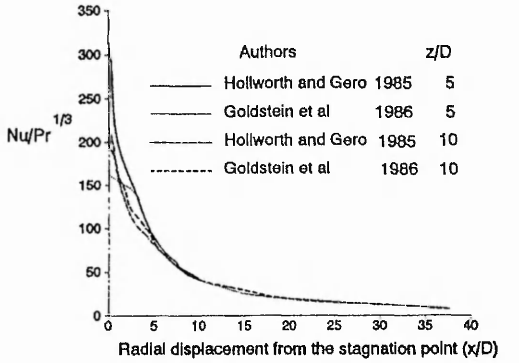


Figure 5 Heat transfer ($\overline{Nu}/Pr^{1/3}$) in the wall jet region based on Goldstein *et al.* (1986) and Hollworth and Gero (1985)

end of the potential core even though its length varies with density ratio.

The reduction of jet temperature due to mixing with the cooler ambient air should be taken into account when determining the adiabatic wall temperature and the heat flux.

A measure of the effect of temperature reduction due to mixing of a nonambient jet issuing from an orifice is given in Hollworth and Wilson (1984). The adiabatic wall temperature can be obtained from the following two correlations. The radial variation of local adiabatic wall temperature (T_{aw}) relative to the adiabatic wall temperature at the stagnation point ($T_{aw,s}$)

$$\alpha = (T_{aw} - T_{amb})/(T_{aw,s} - T_{amb})$$

and the axial variation of the stagnation point adiabatic wall temperature

$$\beta = (T_{aw,s} - T_{amb})/(T_n - T_{amb})$$

Goldstein and Behbahani (1982) expressed the effect of mixing between the jet and an ambient crossflow on the jet temperature in the form of

$$\text{Effectiveness} = (T_{aw} - T_{rec})/(T_{nozz} - T_{amb})$$

where

$$T_{rec} = t_n + R(T_n - t_n)$$

T_n and t_n are the jet total and static temperatures at the nozzle exit and T_{rec} is measured when $T_n = T_{amb}$. The recovery factor R is given by

$$R = (T_{aw} - t_n)/(U_n^2/2Cp)$$

Butler (1984) investigated the effectiveness and the recovery factor of a single circular jet issuing from an ASME nozzle in the absence of a cross flow. The results indicated that the effectiveness is dependent on z/D and x/D.

Figure 6 compares the jet effectiveness obtained by Butler with that derived from the product of α and β using the results of Hollworth and Wilson.

A correlation for effectiveness is given by Goldstein *et al.* (1990)

$$\text{Effectiveness} = 0.35 + 0.6e^{-0.01(z/D-2)^{2.2}-0.1(x/D)^{2.5}}$$

when $0 \leqslant x/D \leqslant 3.5$, and

$$\text{Effectiveness} = 1.193(x/D)^{-0.98}$$

when $x/D \geqslant 3.5$.

Baughn *et al.* (1991) compared the radial distribution of effectiveness obtained for an impinging jet issuing from a pipe with the distribution given by Goldstein *et al.* for an ASME elliptical nozzle. The results showed that, at all z/D tested, a jet issuing from the pipe has a greater effectiveness when $x/D \leqslant 3$. It has been shown by Tso and Hussain (1989) that vortices surrounding the jet enhance the mixing between the jet and the ambient air. The proposal of Popiel and Trass (1991) that a laminar layer (formed on the wall of a contoured convergent nozzle) between the jet and the ambient air causing the development of strong vortices might provide an explanation for the difference in behavior of the jets issuing from the two types of nozzles. This laminar layer would be thinner at the exit to a long pipe where the flow is fully developed than in an ASME elliptical nozzle, resulting in weaker vortices and so less mixing between the jet and the ambient air.

Butler (1984), Goldstein *et al.* (1990), and Baughn *et al.* (1991) all suggest that effectiveness is independent of nozzle exit Reynolds number.

Correlations for local heat transfer derived from published data

An analysis of the stagnating laminar flow on a cylinder by Sibulkin (1952) showed that the Nusselt number at the stagnation point is a function of Reynolds number to the power 0.5. Popiel and Boguslawski (1988) gave a correlation for the contoured ASME nozzle:

$$Sh_s = (0.508 + 0.051z/D)\,Re^{0.5}\,Sc^{0.4}$$

which applied in the range $1.2 \leqslant z/D \leqslant 5$. However, as the nozzle-to-plate spacing was increased to more than 7 nozzle diameters and the turbulent mixing zone penetrates to the jet axis the correlation reflects a turbulent flow. For $z/D \geqslant 7$

$$Sh_s = 0.461Re^{0.75}(z/D)^{-0.87}Sc^{0.4}$$

The form of the expression used by Hoogendoorn (1977) to correlate the stagnation point heat transfer has been used by a number of authors. Hoogendoorn gave the expression

$$\frac{Nu_s}{Re^{0.5}} = 0.65 + 2.03\left(\frac{Tu\,Re^{0.5}}{100}\right) - 2.46\left(\frac{Tu\,Re^{0.5}}{100}\right)^2$$

where $1 \leqslant z/D \leqslant 10$, $Pr = 0.7$, and Re and Tu (the turbulence intensity) are the values obtained from the free jet at an axial displacement from the nozzle equal to the nozzle-to-plate spacing. This expression shows that as the turbulence increases, the weight given to the exponents of Reynolds number greater than 0.5 also increases.

In the wall jet region ($x/D \geqslant 4.5$) experimental work (e.g., Rao and Trass 1964) shows that $Nu \propto Re^a$, where a is in the region of 0.8 for turbulent flows. A transition from laminar to

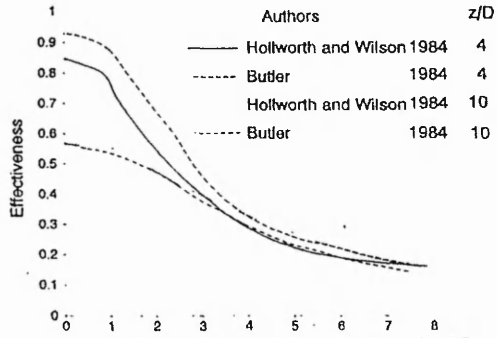


Figure 6 Jet effectiveness at $z/D = 4$ and 10. (From Hollworth and Wilson 1984, and Butler 1984)

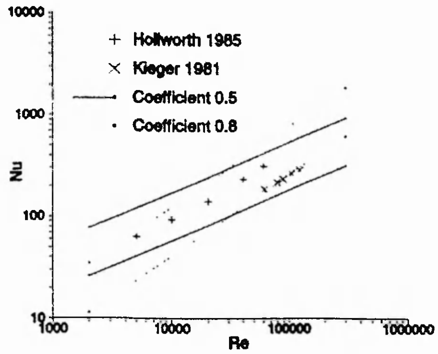


Figure 7 Local heat transfer at the stagnation point at $z/D = 5$

fully turbulent flow is expected to occur between the stagnation point and $x/D = 4.5$ at low nozzle-to-plate spacings. At higher nozzle-to-plate spacings the boundary layer at the stagnation point will be turbulent. This suggests that a correlation for local heat transfer coefficient should be in the form

$$\mathrm{Nu} \propto \mathrm{Re}^a$$

where $a = f(x/D, z/D)$.

Source of information

The review of the jet impingement literature (bibliographies are given in Button and Wilcock [1978] and Button and Jambunathan [1989]) yielded a number of publications giving local values for the heat transfer due to a single impinging circular jet. The data reported covers contoured nozzles, e.g. ASME elliptical type, square-edged orifices, sharp-edged orifices and long pipes. The sources considered in deriving the results presented here are indicated by an asterisk in Table 1. The heat transfer data was extracted by digitizing the graphs given in the publications using a digitizer that has a resolution of 0.1 mm. Small graphs were photographically enlarged; the resulting nonlinearities were less than 2 percent. Errors in the positioning of the digitizer cursor were estimated to be less than 0.5 mm. If a typical data point is assumed to be 100 mm from the axes, then the resulting position error is less than 0.5 percent. A combined error ($= \sqrt{(0.02^2 + 0.005^2)}$) of approximately 2.1 percent is incurred for the conversion of a typical data point from a graph to digital form.

Effect of Reynolds number

Plots of the experimental values of local Nu obtained at the stagnation point for $z/D = 5$ and 10 are given in Figures 7 and 8. These show considerable scatter in the experimental results from the various sources but suggest that the exponent of Re in the relationship

$$\mathrm{Nu} \propto \mathrm{Re}^a$$

will be approximately constant over a range of Reynolds numbers. Figure 9 shows the variation of the exponent with radial displacement from the stagnation point and with nozzle-to-plate spacing. Linear regression using the logarithms of Nusselt number and Reynolds number was used to obtain the exponent a in the relation $\mathrm{Nu} \propto \mathrm{Ra}^a$. The exponent was derived for values of x/D from 0–6 and for values of z/D between 1.2 and 10. Only data from single experiments where three or more data points were available were considered. Of this data the results from six sources have not been included in the derivation of the correlations. The data of Schlunder and Gnielinski (1967) gives a relationship for the stagnation point heat transfer where the exponent for Reynolds number is less than the laminar value of 0.5. The data from Obot *et al.* (1979b, 1979c, and 1982) are based on measurements of the temperature of a series of rings concentric with the stagnation point and varying in width from 1/3 to 1.5 nozzle diameters. The resulting radial distributions of Nusselt number showed a local minimum in the region of the stagnation point even at large nozzle-to-plate spacings. This phenomenon has not been observed in the results of other experiments, suggesting an error in the derivation of the Nusselt number distributions. The exponent of Reynolds number derived from the data of Goldstein and Behbahani (1982) varies only between 0.59 and 0.66 in the range $0 \leqslant x/D \leqslant 6$. It is not clear why this should be so as the apparatus used was similar to that used by Kieger (1981), whose data gives the expected variation in exponent. The results of Sparrow and Lovell (1980), which also show a low value for the exponent of Reynolds number at larger radial displacements from the stagnation point, were obtained at nozzle exit Reynolds numbers between 2,500 and 10,000. These Reynolds numbers are lower than the Reynolds numbers in the other data considered. The trend shown in Figure 9 is for the exponent at the stagnation point to increase from the laminar value of 0.5 as the nozzle-to-plate spacing is increased and for the exponent to increase as the radial displacement from the stagnation point is increased. An expression has been developed to reflect these trends. An expression of this form should be incorporated into a correlation for local heat transfer coefficients. If

$$\mathrm{Nu} \propto \mathrm{Re}^a$$

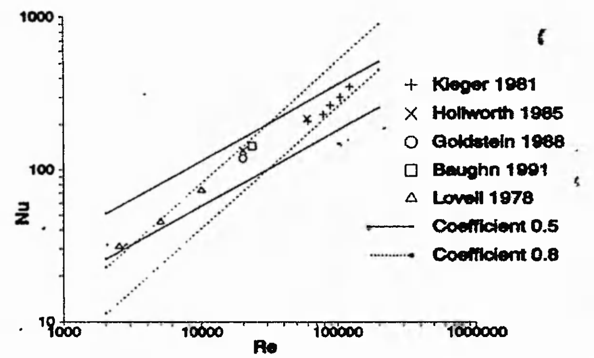


Figure 8 Local heat transfer at the stagnation point at $z/D = 10$

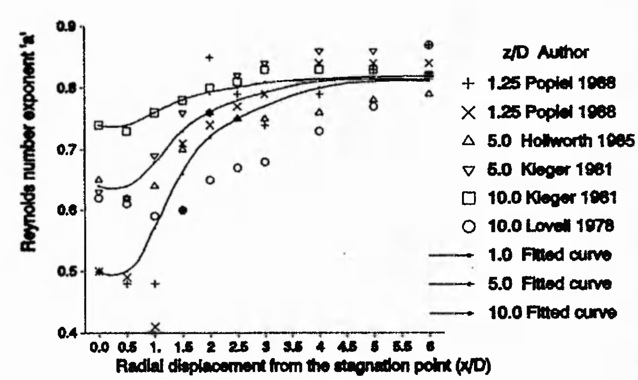


Figure 9 $\mathrm{Nu} = f(\mathrm{Re}^a)$, radial variation in the exponent a

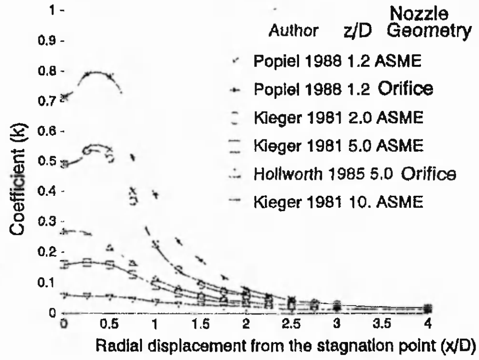


Figure 10 Heat transfer due to a single circular jet impinging on a flat plate

Table 2 Maximum uncertainties for the correlation coefficient

z/D	Nozzle geometry	x/D	95% Confidence interval (%)
1.2	Contoured	2.0	11
1.2	Orifice	1.0	25
2.0	Contoured	3.0	23
5.0	Orifice	0.0	4.1
5.0	Contoured	4.0	4.2
10.0	Contoured	0.0	4.1

then

$$a = 0.82 - \frac{0.32}{\left(1 + A\left(\frac{x}{D}\right)^k + B\left(\frac{x}{D}\right)^l\right)\left(1 + C\left(\frac{z}{D}\right)^m + D\left(\frac{z}{D}\right)^n\right)}$$

where $A = -1.95$, $B = 2.23$, $C = -0.21$, $D = 0.21$, $k = 1.8$, $l = 2$, $m = 1.25$, and $n = 1.5$.

If this assumption is made it is possible to develop a correlation for local Nusselt number in the form

$$Nu = f(Re^a, z/D, x/D)$$

where $a = f(x/D, z/D)$.

A curve of the form

$$Nu = kRe^a$$

where k is a constant and a is the function of x/D and z/D given above, was fitted through the results for 14 values of x/D between 0 and 4 for each of the values of z/D where sufficient data were available. This relationship assumes no heat transfer at zero nozzle flow, implying that the contribution of free convection is insignificant. This assumption is justified by the small temperature differences between the plate and the ambient air in the majority of the experiments on which this work is based.

The resulting curves of k or Nu/Re^a against x/D are plotted in Figure 10. The scatter of the data on which k is based varies with the radial displacement from the stagnation point. This results in a radial variation in the uncertainty of the values derived for k. Table 2 gives the 95 percent confidence interval for the value of k at the radial position where the scatter in the measured data is the greatest and hence where the confidence interval is also the largest. The confidence interval gives the range within which the fitted curve might be expected to fall, assuming that the scatter in the measured results is random.

The effect of nozzle-to-plate spacing and of radial displacement from the stagnation point

The curves shown in Figure 10 could be approximated by correlations of the form

$$\frac{Nu}{Re^a} = \frac{K}{1 - (x/D)^n}$$

where $1 \leqslant n \leqslant 2$, $a = f(x/D, z/D)$, and K is a constant. Since the curves are dissimilar, a different value of n would be required depending on the nozzle-to-plate spacing. Also there would be a considerable loss of accuracy in the region of the stagnation point where there is a local minimum in the value of k.

Discussion

Many papers are available that consider the effect of nozzle-to-plate spacing and Reynolds number on the Nusselt number. More recently the significance of other parameters such as nozzle geometry and confinement have been appreciated and investigations made into their effects. It appears that nozzle geometry affects the generation of turbulence in the shear layer. At nozzle-to-plate spacings of less than 10 diameters the use of an orifice will probably yield higher rates of heat transfer than a contoured nozzle. Other work suggests that the use of a pipe will also result in higher rates of heat transfer than a contoured nozzle at larger nozzle-to-plate spacings. A definitive statement on which nozzle geometry should be used to give the highest rates of heat transfer is probably not possible at present because this would involve a comparison between results from different experiments where factors apart from the nozzle geometry are significant. The optimal design of a jet impingement system is further complicated by the mixing of the ambient air with the jet. Where the temperature difference between the jet and the heat transfer surface is large compared with the temperature difference between the ambient air and the heat transfer surface, mixing will tend to reduce the exchange of energy between the jet and the heat transfer surface. The factors that cause an increase in turbulence, and so an increase in heat transfer rate, may also be expected to increase the dissipation of the jet temperature and so may in fact cause a reduction in the heat energy transferred.

The experimental data can be divided into three groups according to the type of jet nozzle used: ASME elliptic nozzles, orifices, and pipes. Figure 10 shows that the Nusselt number for an orifice jet is significantly greater than for an ASME elliptic nozzle. The effect of nozzle geometry is most significant at the stagnation point and is reduced as x/D increases. The effect of nozzle geometry is also greater at smaller nozzle-to-plate spacings.

Figures 7 and 8 show that $Nu \propto Re$ and suggest that at the stagnation point the exponent a is independent of Reynolds number in the range 5,000–124,000. Figure 9 shows that a is dependent on the radial displacement from the stagnation point and the nozzle-to-plate spacing. Previous correlations have either assumed that the exponent of Reynolds number is independent of both z/D and x/D or that the correlation applies to only a very limited range of values for z/D and x/D. For example, Goldstein and Franchett (1988) proposed a correlation,

$$Nu/Re^{0.7} = A\,e^{-(B - C\cos\phi)(x/D)^m}$$

where ϕ is the impingement angle, and gave different values for each of the coefficients for nozzle-to-plate spacings from 4–10 nozzle diameters. This correlation is based on data obtained for Reynolds numbers in the range $10{,}000 \leqslant Re \leqslant$

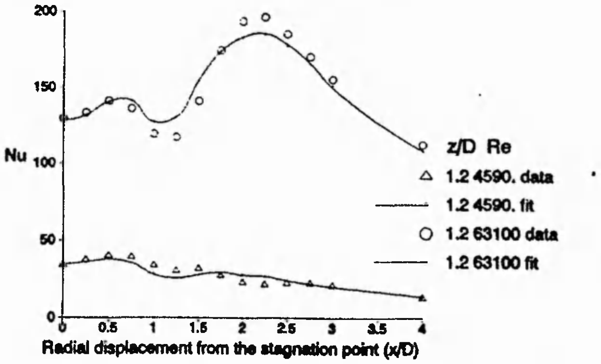


Figure 11 Radial variation in heat transfer between a plate and an impinging jet. Comparison of measurements and correlations. (Data obtained by Popiel and Boguslawski [1988] using an ASME nozzle)

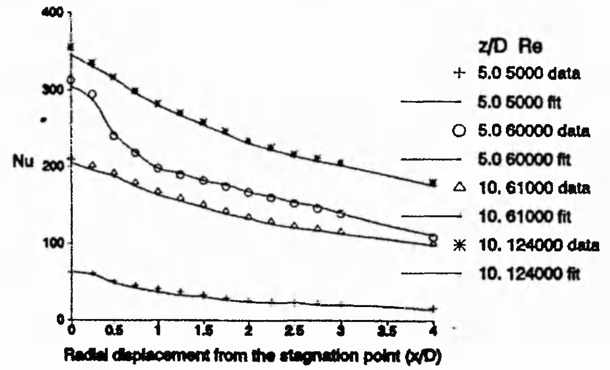


Figure 12 Radial variation in heat transfer between a plate and an impinging jet. Comparison of measured data and correlations. (Data from Hollworth and Gero [1985] obtained at $z/D = 5$ using an orifice and from Kieger [1981] obtained at $z/D = 10$ using an ASME nozzle)

30,000. A correlation in which the exponent of Reynolds number is given as a function of both x/D and z/D should be applicable over a wider range of parameter values without the need to change the correlation coefficients.

Figures 11 and 12 compare the radial distribution of Nu derived from the correlations with experimental values taken from the literature. These show a reasonable agreement between the correlations and experimental values.

Conclusions

A review of the empirical results confirms that the simplest correlations for local heat transfer coefficient should be of the form

$$\mathrm{Nu} = f(\mathrm{Re}, z/D, x/D, \mathrm{Pr})$$

but that these correlations will not account for the significant effects of nozzle geometry, confinement, and the generation of turbulence upstream of the jet nozzle. It has been demonstrated that the prediction of heat transfer also must include a consideration of the dissipation of jet temperature. This requires the use of local rather than area mean heat transfer coefficients.

The review clearly shows that details of the nozzle geometry, the confinement of flow, and the turbulence intensity at the nozzle exit are required if comparisons between different sets of experimental results are to be used to obtain an improvement in the understanding of the jet impingement heat transfer process.

Graphical correlations have been obtained for the local heat transfer due to a single impinging jet in the form

$$\frac{\mathrm{Nu}}{\mathrm{Re}^{f(x/D, z/D)}} = f_1(x/D, z/D)$$

where

$$f(x/D, z/D) = 0.82 - \frac{0.32}{\left(1 + A\left(\frac{x}{D}\right)^k + B\left(\frac{x}{D}\right)^l\right) \times \left(1 + C\left(\frac{z}{D}\right)^m + D\left(\frac{z}{D}\right)^n\right)}$$

and $A = -1.95$, $B = 2.23$, $C = -0.21$, $D = 0.21$, $k = 1.8$, $l = 2$, $m = 1.25$, and $n = 1.5$. This correlation is shown to be satisfactory at the nozzle-to-plate spacings for which heat transfer data are available over a range of Reynolds numbers. This correlation gives Nusselt number as a function of Reynolds number raised to an exponent where the exponent depends on the nozzle-to-plate spacing and the radial displacement from the stagnation point. A relationship of this form is suggested by the empirical data. It is suggested that future work in developing correlations when more data are available should follow a similar approach.

In conjunction with the results given by other authors, e.g., Goldstein *et al.* (1990), who investigated the dissipation of a nonambient jet, the correlations can be used to derive the heat transfer to a plate when the air temperature at the nozzle exit is not equal to the ambient temperature.

Jets issuing from square-edged orifices give higher heat transfer than jets from ASME elliptical nozzles.

The data considered in this review support the suggestion that the heat transfer coefficients in the wall jet are independent of nozzle-to-nozzle spacing z/D in the range of 1–12 nozzle diameters. Approximate values of heat transfer coefficient in the wall jet can be obtained by the extrapolation of the local value at $x/D \geqslant 4.5$ using the relationship

$$\mathrm{Nu} \propto (x/D)^{-1}$$

Acknowledgment

The post held by M. A. Moss is funded by the Polytechnics and Colleges Funding Council (PCFC) under their special research initiative and is gratefully acknowledged. The authors also wish to thank D. Corlett for her work in maintaining the collection of jet impingement literature on which this work is based.

References

Baughn, J. W., Hechanova, A. E. and Xiaojun, Y. 1991. An experimental study of entrainment effects on the heat transfer from a flat surface to a heated circular impinging jet. *Proc. of the ASME/JSME Joint Conf. Thermal Engineering*, Reno, NV, USA, March 17–22

Baughn, J. W. and Shimizu, S. 1989. Heat transfer measurements from a surface with uniform heat flux and an impinging jet. *J. Heat Transfer Trans. ASME*, **111**, 1096–1098

Behbahani, A. I. and Stewart, B. W. 1989. Measurement of the local mass transfer coefficient by holographic interferometry. *National Heat Transfer Conf. HTD*, **112**

Butler, K. A. 1984. Effect of entrainment on the heat transfer between a flat surface and impinging circular air jets. M.Sc. thesis, University of Minnesota, MN, USA

Button, B. L. and Jambunathan, K. 1989. Jet impingement heat transfer: a bibliography 1976–1985. *Previews Heat Mass Transfer*, **15**, 149–178

Button, B. L. and Wilcock, D. 1978. Impingement heat transfer—a bibliography 1890–1975. *Previews Heat Mass Transfer*, **4**, 83–89

Button, B. L. and Wilcock, D. 1982. Effect of nozzle geometry on circular jet impingement heat transfer. *2nd Polytechnic Symposium on Thermodynamics and Heat Transfer*, Leicester, UK, November 16–17, 86–99

Chia, C. J. 1972. Mass transfer in an impinging jet. Ph.D. thesis, University of Toronto, Ontario, Canada

Dawson, D. A. and Trass, O. 1966. Mass transfer in a turbulent radial wall jet. *Canadian J. of Chemical Engineering*, **V44**, 121–129

Den Ouden, C. and Hoogendoorn, C. J. 1974. Local convective-heat-transfer coefficients for jets impinging on a plate: experiments using a liquid-crystal technique. *Proc. 5th Int. Heat Transfer Conf.*, Tokyo, Japan, September 3–7, **5**, 293–297

Donaldson, C. DuP., Snedeker, R. S. and Margolis, D. P. 1971. A study of free jet impingement. Part 2. Free jet turbulent structure and impingement heat transfer. *J. Fluid Mech.*, **45**, 477–512

Gardon, R. and Akfirat, J. C. 1965. The role of turbulence in determining the heat-transfer characteristics of impinging jets. *Int. J. Heat Mass Transfer*, **8**, 1261–1272

Gardon, R. and Carbonpue, J. 1962. Heat transfer between a flat plate and jets of air impinging on it. *Int. Developments in Heat Transfer, Int. Heat Transfer Conf.*, University of Colorado, CO, USA, August 28 to September 1, **2**, 454–460

Gauntner, J. W., Livingood, J. N. B. and Hrycak, P. 1970. Survey of literature on flow characteristics of a single turbulent jet impinging on a flat plate. *NASA* TN D-5652 NTIS N70-18963

Giralt, F., Chia, C. J. and Trass, O. 1977. Characterization of the impingement region in an axisymmetric turbulent jet. *Ind. Eng. Chem. Fundam.*, **16**, 21–28

Goldstein, R. J. and Behbahani, A. I. 1982. Impingement of a circular jet with and without crossflow. *Int. J. Heat Mass Transfer*, **25**, 1377–1382

Goldstein, R. J., Behbahani, A. I. and Kieger Heppelmann, K. 1986. Streamwise distribution of the recovery factor and the local heat transfer coefficient to an impinging circular air jet. *Int. J. Heat Mass Transfer*, **29**, 1227–1235

Goldstein, R. J. and Franchett, M. E. 1988. Heat transfer from a flat surface to an oblique impinging jet. *ASME J. Heat Transfer*, **110**, 87–93

Goldstein, R. J., Sobolik, K. A. and Seol, W. S. 1990. Effect of entrainment on the heat transfer to a heated circular air jet impinging on a flat surface. *ASME J. Heat Transfer*, **112**, 608–611

Goldstein, R. J. and Timmers, J. F. 1982. Visualization of heat transfer from arrays of impinging jets. *Int. J. Heat Mass Transfer*, **25**, 1377–1382

Gundappa, M., Hudson, J. F. and Diller, T. E. 1989. Jet impingement heat transfer from jet tubes and orifices. *National Heat Transfer Conf.*, Philadelphia, PA, USA, August 6–9, **107**, 43–50

Hollworth, B. R. and Gero, L. R. 1985. Entrainment effects on impingement heat transfer. Part 2. Local heat transfer measurements. *J. Heat Transfer Trans. ASME*, **107**, 910–915

Hollworth, B. R. and Wilson, S. I. 1984. Entrainment effects on impingement heat transfer. Part 1. Measurements of heated jet velocity and temperature distributions and recovery temperatures on target surface. *J. Heat Transfer. Trans. ASME*, **106**, 797–803

Hoogendoorn, C. J. 1977. The effect of turbulence on heat transfer at a stagnation point. *Int. J. Heat Mass Transfer*, **20**, 1333–1338

Hrycak, P. 1978. Heat transfer from a row of jets impinging on concave semi-cylindrical surface. *Proc. 6th Int. Heat Transfer Conf.*, Toronto, Ontario, Canada, August 2, 67–72

Katoaka, K. 1985. Optimal nozzle-to-plate spacing for convective heat transfer in nonisothermal, variable density impinging jets. *Drying Technol.*, **3**, 235–254

Kataoka, K., Harada, T. and Sahara, R. 1988. Mechanism for enhancement of heat transfer in turbulent impinging jets. *Current Research in Heat and Mass Transfer: A Compendium and a Festschrift for Professor Arcot Ramachandran* (M. V. Krishna Murthy *et al.*, Eds.), Hemisphere, New York

Katoaka, K., Komai, T. and Nakamura, G. 1978. Enhancement mechanism of mass transfer in a turbulent impinging jet for high Schmidt numbers. *ASME* Paper 78-HT-5

Kataoka, K., Suguro, M., Degawa, H., Maruo, K. and Mihata, I. 1987. The effect of surface renewal due to large-scale eddies on jet impingement heat transfer. *Int. J. Mass Transfer*, **30**, 559–567

Kieger, K. K. 1981. Local and average heat transfer from a flat surface to a single circular jet of air impinging on it. M.Sc. thesis, University of Minnesota, MN, USA

Launder, B. E. 1991. Private communication, October 10

Lepicovsky, J. 1989. The role of nozzle-exit boundary layer velocity gradient in mixing enhancement of free jets. *Proc. 3rd Joint ASCE/ASME Mechanics Conf.*, San Diego, CA, USA, July 9–12

Lovell, B. J. 1978. Local transfer coefficients for impingement of an axisymmetric jet on an inclined flat plate. M.Sc. thesis, University of Minnesota, MN, USA

Martin, H. 1977. Heat and mass transfer between impinging gas jets and solid surfaces. *Advances in Heat Transfer*, **13**, 1–60

Mitachi, K. and Ishiguro, R. 1977. Heat transfer of radial wall jets. Second report; Measurements of temperature field and heat transfer for radial wall jet. *Heat Transfer-Japan Res.*, **6**, 55–65

Obot, N. T., Mujumdar, A. S. and Douglas, W. J. M. 1979a. Design correlations for heat and mass transfer under various turbulent impinging jet configurations. *Developments in Drying*, 388–402

Obot, N. T., Mujumdar, A. S. and Douglas, W. J. M. 1979b. Effect of nozzle geometry on impingement heat transfer under a round turbulent jet. *ASME* Paper 79-WA/HF-53

Obot, N. T., Mujumdar, A. S. and Douglas, W. J. M. 1979c. Heat transfer under axisymmetric turbulent impinging jets. *AIChE 72nd Ann. Conf.*, San Francisco, CA, USA, A1-C3

Obot, N. Y., Mujumdar, A. S. and Douglas, W. J. M. 1982. Effect of semi-confinement on impingement heat transfer. *Proc. 7th Int. Heat Transfer Conf.*, Munchen, Germany, September 6–10, **3**, 395–400

Popiel, C. O. and Boguslawski, L. 1988. Effect of flow structure on the heat or mass transfer on a flat plate in impinging round jet. *2nd UK National Conf. on Heat Transfer*, University of Strathclyde, UK, September 14–16, **1**, 663–685

Popiel, C. O. and Trass, O. 1982. The effect of ordered structure of turbulence on momentum, heat, and mass transfer of impinging round jets. *Proc. 7th Int. Heat Transfer Conf.*, Munchen, Germany, September 6–10, **6**, 141–146

Popiel, C. O. and Trass, O. 1991. Visualization of a free and impinging round jet. *Exp. Thermal Fluid Sci.*, **4**, 253–264

Poreh, M., Tsuei, Y. G. and Cermak, J. E. 1967. Investigation of a turbulent radial wall jet. *J. Appl. Mech. Trans. ASME*, **34**, 457–463

Rao, V. V. and Trass, O. 1964. Mass transfer from a flat surface to an impinging turbulent jet. *Can. J. Chem. Eng.*, **42**, 95–99

Schlichting, H. 1968. *Boundary-Layer Theory*. McGraw-Hill, New York, 681

Schlunder, E. U. and Gnielinski, V. 1967. Heat and mass transfer between surfaces and impinging jets. *Chem. Ing. Tech.*, **39**, 578–584 (Translation held by BLDSC)

Sibulkin, M. 1952. Heat transfer near the forward stagnation point of a body of revolution. *J. Aeronautical Sci.*, August

Sparrow, E. M. and Lovell, B. J. 1980. Heat transfer characteristics of an obliquely impinging circular jet. *J. Heat Transfer. Trans. ASME*, **102**, 202–209

Tani, I. and Komatsu, Y. 1966. Impingement of a round jet on a flat surface. *Proc. of the Eleventh International Congress of Applied Mechanics* (H. Gortler, Ed.), Springer-Verlag, New York, 672–676

Tso, J. and Hussain, F. 1989. Organized motions in a fully developed turbulent axisymmetric jet. *J. Fluid Mech.*, **203**, 425–448

Welty, J. R., Wicks, C. E. and Wilson, R. E. 1976. *Fundamentals of Momentum, Heat and Mass Transfer*. Wiley, New York, 376–380

Yokobori, S., Kasagi, N., Hirata, M., Nakamaru, M. and Haramura, Y. 1979. Characteristic behavior of turbulence and transport phenomena at the stagnation region of an axi-symmetrical impinging jet. *Proc. 2nd Symp. Turb. Shear Flows*, London, UK, **4**, 4.12–4.17

Engng Applic. Artif. Intell. Vol. 5, No. 2, pp. 145–153, 1992
Printed in Great Britain.

0952-1976/92 $5.00+0.00

Contributed Paper

The Implementation of a Deductive Database for Engineering Correlations

E. LAI
Nottingham Polytechnic

M. A. MOSS
Nottingham Polytechnic

K. JAMBUNATHAN
Nottingham Polytechnic

B. L. BUTTON
Nottingham Polytechnic

A vast quantity of data in the form of correlations is available for use in design and analysis. Even using hitherto-available databases there are significant difficulties in obtaining the best correlation for a particular purpose. The limitations of current databases in this respect are discussed and the implementation of a deductive database to overcome these limitations is described.

A database has been developed which makes available a wide variety of correlation information; the data includes arithmetic expressions, tables, graphs and procedural methods on the topic of jet impingement heat transfer. Logic programming is used to provide access to the relational database and to declare rules which act on this data: the resulting knowledge-based system is able to confine its selection of information to relevant and valid correlations. In addition, facilities created using knowledge-based programming techniques assist the engineer in the task of providing a full description of the heat transfer system and in evaluating the correlations, returning values for heat transfer coefficients and flux for the system described. The knowledge-based methods described provide improved access to the correlation data and are believed to be applicable to other fields where large quantities of data are presented in the form of correlations.

Keywords: Deductive database, knowledge-based system, heat transfer correlations, expert database system.

INTRODUCTION

Heat transfer data for design or analysis is typically dispersed over a vast number of sources. The subject of jet impingement heat transfer alone has generated more than 1000 English-language publications.[1,2] An engineer is unlikely to become familiar with more than a small proportion of the data available. He may be satisfied to extrapolate a familiar correlation, potentially introducing an error, rather than embark on a search for a more-suitable correlation.

A conventional database containing the titles, abstracts and publications can improve the situation by selecting a subset of the total number of publications available, but this approach has a number of limitations. The selection of publications is typically based on the presence of specified words in the title and abstract, and there is a tendency for authors to address the abstract to contemporary research issues rather than describe the data presented. As a consequence, publications containing relevant data may not be selected. In addition, the range of conditions over which data is valid is generally not described in the abstract and so papers which are not relevant may be selected. Finally, once the selection has been made the engineer must still distinguish between appropriate and inappropriate data, extract the data and derive values for heat transfer rates. A knowledge-based system (KBS) which selects relevant valid data from a database and derives

Correspondence should be sent to: M. A. Moss, Department of Mechanical Engineering, Nottingham Polytechnic, Burton Street, Nottingham NG1 4BM, U.K.

values from this data would be of considerable benefit. This paper describes the implementation of such a system.

Related work

Two examples of applications combining KBSs and databases are a system for processing knowledge on the effects of earthquakes on buildings[3] and a system for advising DIY customers on the selection of paint for a particular job.[4] Adeli and Hung[3] outline a system for making available data describing the design of a building and the observed effect of an earthquake on the building. A large quantity of observed data is stored, either as production rules or in a relational database. The addition of further earthquake data is simplified where the relational database is used in the place of rules, though the basis for deciding whether information should be represented as production rules or in the database is not made clear. The KBS described by Van der Spec[4] offers advice to DIY customers on painting and paints. Knowledge is divided between a rule-base and a database; the database contains details of paint characteristics, prices etc., while the rule base contains the knowledge needed to interpret and apply this data. For example, rules specify which paint characteristics are required to perform various tasks and so the knowledge base is able to select suitable paints for a particular job.

A conventional database contains only facts and gives access to these facts. Lloyd[5] gives an introduction to the "deductive database" which uses logic to represent both facts and rules; the rules are used to deduce further facts when required. Logic provides a single language which can be used to express not only data definitions but also queries, views, integrity constraints, and programs.[6] Prolog is the most common implementation of a language for programming in logic, and a program written in Prolog can combine sophisticated database access and knowledge-based programming. Brodie and Jarke[7] have considered the pros and cons of logic programming. In addition to the beneficial qualities mentioned above they have also stated that the semantics of logic as a knowledge representation are formally well defined; this is not always the case with expert-system shells. Brodie and Jarke also mentioned a number of limitations; these include difficulties in representing procedural knowledge and in updating the database from within a KBS.

The problems and alternative methods of interfacing Prolog with databases have been discussed by a number of authors, and an overview of this work is given by Torsun and Ng.[8] Lucas[9] described the requirements for an interface between Prolog and a relational database, which provides access to the facts in an external database in the same way as to those in the internal Prolog database.

The selection of database records is often based on the matching of a set of keywords specified by the user with members of a set of keywords associated with a record. Problems associated with this method are discussed by Parsaye *et al.*,[10] these problems include the tendency of different persons assigning keywords to a document not to use the same words to describe identical subject matter. The same difficulty applies to persons formulating queries using keywords. By using a hierarchy of keywords Parsaye *et al.* were able to overcome many of these problems.

A DESCRIPTION OF THE HEAT TRANSFER DATABASE

An overview of the heat transfer data and of access methods

Heat transfer data is presented in papers in the form of arithmetic expressions, tables and graphs. A publication may contain data in these forms in any combination. For convenience, a particular combination of data which can be used to derive values for heat transfer rates will be referred to as a "correlation." Data on the heat transfer due to the impingement of single and multiple jets has been incorporated into a database. The database system developed enables an engineer to access these diverse types of data in order to derive heat transfer rates.

The method used to obtain an appropriate correlation from the database follows the method that would be used by an engineer in a thorough search for a correlation. Potentially suitable correlations are initially selected, based on a general description of the physical system concerned. Correlations are generally only applicable over a limited range of parameter values, normally expressed as non-dimensionals. For instance, the range of flow velocity over which a correlation for jet impingement heat transfer is valid is typically constrained so that the nozzle exit Reynolds number lies within a limited range. For each correlation the limiting parameters or expressions will be evaluated and a comparison made with the corresponding allowable maximum and minimum numerical values. A final selection is based on the results of this comparison. The selected correlations are then used to derive values for heat transfer flow rates.

The database

The information is represented in the database in a tabular (relational) form (Fig. 1); the meaning attributed to the fields in each of the tables is given below.

A correlation is uniquely identified by the publication code, which identifies the source of the correlation, and the configuration identifier, which is used to distinguish between different correlations given in a single source.

Table 1 uses keywords to give a general indication of which system configurations a correlation is applicable to.

Table 2 gives the publication details of the source of

TABLE 1.

PUBLICATION CODE.
CONFIGURATION IDENTIFIER.
KEYWORDS.

TABLE 2.

PUBLICATION CODE.
AUTHORS NAMES.
YEAR.

TABLE 3.

PUBLICATION CODE.
CONFIGURATION IDENTIFIER.
EXPRESSION.
MINIMUM LIMIT.
MAXIMUM LIMIT.

TABLE 4.

PUBLICATION CODE
CONFIGURATION IDENTIFIER.
LINE NUMBER.
CORRELATION STATEMENT

Fig. 1. Relations in the heat transfer correlation database.

the heat transfer data.

Table 3 indicates the range of validity of the heat transfer data. The expression quantifies some aspect of the impingement system, the jet Reynolds number for example. The minimum and maximum limits indicate the range in which the value of the expression must lie for the associated heat transfer data to be valid. This technique allows the database to hold as many sets of expressions and corresponding limits as are required to give the limits of validity of the correlation data. The technique also allows a significant simplification of the database; the conventional alternative would pre-specify pairs of fields for the maximum and minimum values of each non-dimensional parameter. Under this arrangement not all the fields would be used for every correlation and the database would also be much less flexible, as the addition of new expressions and limits would be impossible.

Table 4 gives the heat transfer correlation data. The heat transfer data may be in the form of arithmetic expressions, tables or graphs. In general the storage of the heat transfer data requires more than one record. The line number field is used to specify the order of evaluation of the records.

Examples of typical database entries are given in Fig. 2.

The KBS has been developed using an implementation of Prolog[11] which provides access to facts in the external database in the same way as to facts in the internal database.

Correlation selection

Correlations are initially selected using a search of the keywords (Fig. 2, Table 1), which describe the jet impingement system. For example database entries containing information about an array of jets flowing from circular square-edged holes would contain the keywords "array" "circular" and "orifice". A query is made by specifying a list of keywords. If all these are present then the entry for that correlation will be selected. Knowledge-based aids to this section process have been implemented and are described below.

Once a set of correlations has been selected by using their keywords then the decision whether or not to derive a value for the heat transfer coefficient can be made in one of three ways. These are to require that the values obtained for each of the expressions obtained from Table 3 (Fig. 2) are either within the specified limits or within a specified tolerance of the limits, or to ignore the limits. If the second option is selected tolerances are requested from the user for the jet flow rate, the impingement plate-to-target spacing and (in the case of multiple jets only) nozzle-to-nozzle spacing. When an expression is evaluated the system also examines it to see if it contains parameters which are dependent on one or more of the jet flow rate, impingement plate-to-target spacing and nozzle-to-nozzle spacing. If the expression does depend on these parameters then the limits are adjusted accordingly. This allows extrapolation up to a specified limit. If the third option is selected the limit expressions are not evaluated and an attempt is made to extrapolate the correlations. If an attempt is made to extrapolate correlation data it is quite likely that a failure will occur; typical causes are an attempt to divide by zero or an attempt to take the square root of a negative number. The system must recover from failures of this nature, inform the user and continue to evaluate the remaining selected correlations.

THE FRONT END—KBS-BASED AIDS TO DATABASE ACCESS

In addition to the usual input and output functions the front-end uses KBS techniques to perform two active tasks: to generate or obtain a list of keywords used to describe the heat transfer system, and to deduce the values required to fully define the geometry and flow properties.

An aid to the input of keywords

The words used in describing a heat transfer process can be organised into a hierarchy where the description becomes more specific as the hierarchy is descended;

```
TABLE 1.
behb_83      a     array,staggered,circular,orifice,flat,max_crossflow
kercher_70   a     array,circular,orifice,in_line,max_crossflow,flat

TABLE 2.
behb_83      a     Behbahani and Goldstein      1983
kercher_70   a     Kercher and Tabakoff         1970

TABLE 3.
behb_83      a     lx/d            4       8
behb_83      a     ly/d            4       8
behb_83      a     z/d             2       5
behb_83      a     4*m/d/pi/mu     5000    15000
behb_83      a     nx              5       10
kercher_70   a     rho*v*d/mu      3000    30000
kercher_70   a     z/d             1.0     4.8
kercher_70   a     lx/d            3.1     12.5
kercher_70   a     nx              4       16

TABLE 4.
behb_83      a     1    re=4*m/d/pi/mu
behb_83      a     2    coeff,behb_83.dat,z/d,table
behb_83      a     3    exp,behb_83a.dat,z/d,table
behb_83      a     4    nu=coeff*re**0.78*(z/d)**exp
kercher_70   a     1    re=rho*v*d/mu
kercher_70   a     2    coeff,kerc_70.dat,lx/d,table
kercher_70   a     3    row=1
kercher_70   a     4    coeff2=0
kercher_70   a     5    while row =< nx
kercher_70   a     6    mc=m*(row-1)+mi
kercher_70   a     7    coeff1,kerc_70c.dat,mc/(lx*d*rho*v),table
kercher_70   a     8    row=row+1
kercher_70   a     9    coeff2=coeff1+coeff2
kercher_70   a     10   end_while
kercher_70   a     11   coeff2=coeff2/nx
kercher_70   a     12   exp,kerc_70d.dat,lx/d,table
kercher_70   a     13   nu=coeff*coeff1*re**exp*pr**(1/3)*(z/d)**0.091
```

Fig. 2. A fragment of the heat transfer correlation database.

"array", "row" and "single" describe three possible configurations on "impinging jet". Furthermore, some combinations of keywords do not make physical sense or are mutually exclusive, for example the words orifice, pipe and contoured are used to describe the cross-sections of three different jet nozzles. These inter-dependencies may be represented in directed graphs similar to semantic net structures.[12] The keywords and the relationships between them are represented as nodes and links in hierarchical graphs. A portion of one graph is shown in Fig. 3. Two types of link are used: "Subset", which indicates that one keyword specifies a subset of the other, and "Excludes", which indicates keywords that are mutually exclusive, for example:

subset (row, pipe)
subset (row, orifice)
excludes (orifice, pipe)
excludes (orifice, contoured)

Keywords can be mutally exclusive by implication, in the above example "orifice" excludes "pipe" and "orifice" excludes "contoured", so it can be inferred that "pipe" excludes "contoured". The graph formed by using the links "subset" is acyclic, whereas the graph formed by the transitive links "excludes" is cyclic. The heat transfer correlation database uses two hierarchies of keywords, the first to describe the convection process or flow, and the second to describe the geometry of the heat transfer surface. Using the relationships between keywords represented in these graphs, the front-end can either assist the user in generating a list of keywords or check that a list of keywords entered by the user is valid.

(1) The front-end is able to suggest keywords to the user in such a way that a valid list of keywords is developed. This is achieved by first displaying a list of the keywords which are directly linked to the root of the tree, one of which is selected by the user. The front-end then offers keywords which are connected to the selected keyword by means of the link "subset" but which are not inferred as being excluded by the link "excludes". This process is repeated, offering keywords linked to previously selected keywords and progressively descending the hierarchy, until either no further valid keywords are available or the engineer decides that the keywords chosen are sufficiently specific. At any point during this process the engineer may opt to view a graphical display of the keyword hierarchy

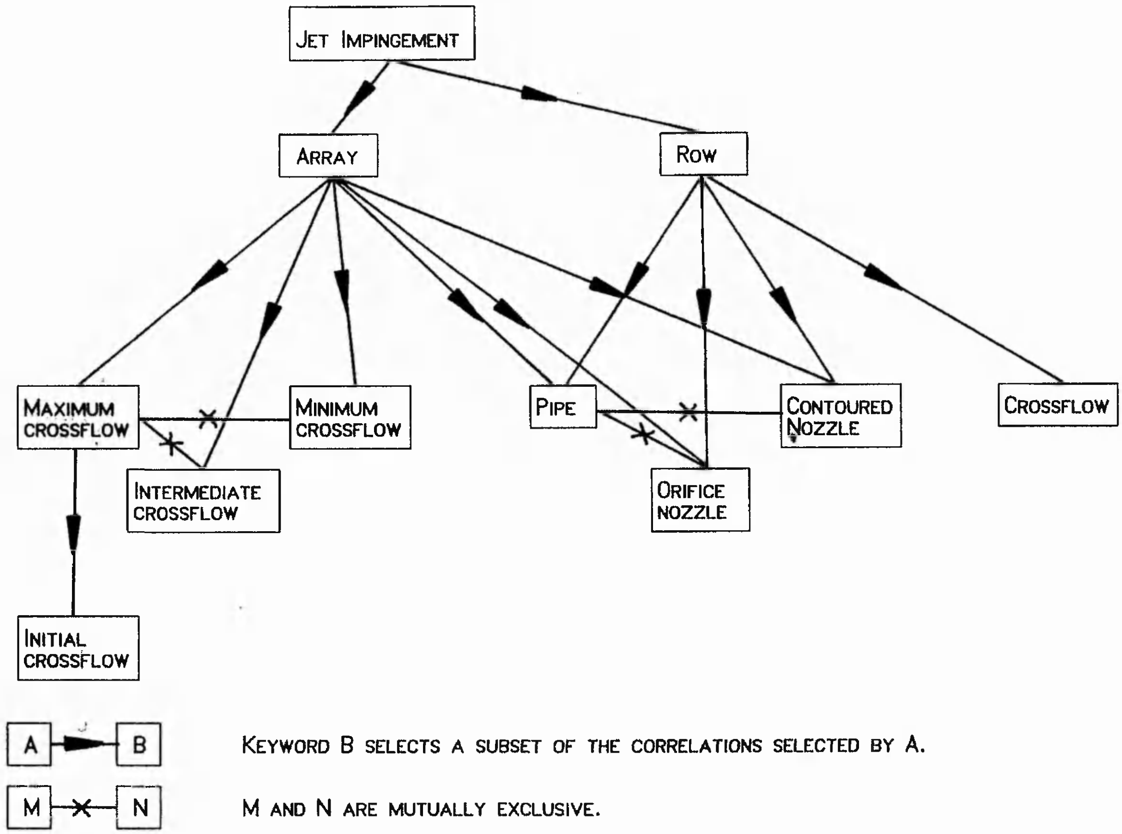


Fig. 3. The relationships between selected keywords expressed as a graph.

from the current level downwards. This provides a visual display of all the remaining possible keywords and their relationships.

(2) The front-end will check the validity of a list of keywords entered by the user. The specified keywords are valid if a set of paths exists, one for each keyword, where there are no pairs of keywords which should exclude each other. A path is a list of nodes connecting a specified keyword with the root of its hierarchy, described by the link "subset". The fact that two keywords are mutually exclusive is derived from the link "excludes".

An aid to the input of numerical values

For a given physical system a corresponding set of parameter values is used in the correlations and in the expressions for the limits of validity. The front-end derives appropriate lists of parameter names and the rules specifying the relationships between these parameters, the selection of these lists is based on the keywords specified. The parameter names from the list are then displayed and the user may input one or more parameter values of his choice. The front-end considers each input value in turn and cycles through all the input rules in order to derive new parameter values. This process is repeated until either all the parameters have been derived or all the input values have been considered. If additional parameters need to be derived the list of parameter names (and values where available) is redisplayed and the user prompted for more input.

The heat transfer geometry and flow properties can both be specified in a number of ways, for example the jet flow rate ($\overset{\circ}{m}$) can be specified directly or derived from the equations for compressible flow. Two of the many possible expressions are given below:

$$\overset{\circ}{m} = A.\,P1\left(\frac{2\gamma 1}{1-\delta R.T1}\left(\left(\frac{p2}{P1}\right)^{\gamma-1/\gamma} - \left(\frac{p2}{P1}\right)^{\gamma}\right)\right)^{0.5}$$

$$\overset{\circ}{m} = \rho.V.A$$

where $\rho = f(T, P)$ and

$$V = \left(\frac{2\gamma}{1-\gamma}R.T1\left(\left(\frac{p2}{P1}\right)^{\gamma-1/\gamma} - 1\right)\right)^{0.5}.$$

It is an objective of the KBS that all the geometric and flow parameters can be fully specified using any one of many possible selections of parameters from the whole list of parameters. To make this possible the front-end must allow the user to enter values for any of the parameters, which introduces two potential problems.

Firstly if a nozzle mass flow rate is entered along with the nozzle area, the flow temperature and pressures, it is possible that the nozzle mass flow rate will differ from that derived analytically. This form of inconsistency in the input must be prevented.

The second problem occurs because of the nature of the inter-relations between the parameters. The inter-relations allow non-terminating loops to be set up during pure backwards chaining. For example, in an attempt to derive nozzle mass flow from known values of $T1$, $P1$ and D using the rule

$$\mathring{m}=f(V,\rho,D) \quad \text{where} \quad \rho=f(T1,P1),$$

backwards chaining might select the rule

$$V=f(m,D,\rho)$$

which leads to a non-terminating loop because mass-flow is required. The inference method used must avoid this problem and instead select the rule

$$\mathring{m}=f(T1,P1,P2,D)$$

from which mass flow can be calculated.

A hybrid of forwards and backwards chaining along with a consistency check is used to overcome these foreseen difficulties. The rules used to derive the system geometry and flow property values all have the format:

$X1$ is true IF
value parameter X is NOT known
AND value Y is known
AND value Z is known
AND X can be derived from the function $X=f(Y,Z)$
AND store value for X

or in Prolog syntax

```
m1:-
    v(V),
    a(Area),
    rho(Rho),
    M is Rho*Area*V,
    asserta(m(M):-!).
```

The value for X is derived using the forward chaining rule ($X1$), values for Y and Z are obtained by backward chaining. The rules for deriving Y and Z, $Y1$ and $Z1$ respectively, have a similar format. A non-terminating loop is avoided because the rules to derive Y and Z do not refer back to the rule $X1$ but instead use the value X. The system is implemented in Prolog. A fragment of the code is given in Appendix A.

A consistency check is made on all input values before they are used. When a value is entered a check is made to see if the value has already been derived by the system. If it has then the values are compared; if the values are not equal the input value is rejected and the user informed. The method is based on the rule cycle hybrid described by Rowe[13] with two exceptions. First, once the rules have derived all the possible new values, inference stops. The user is then prompted for more data, where Rowe's method continues inference until all rules are satisfied. Second, is the inclusion of a consistency check.

EVALUATION OF HEAT TRANSFER

If a correlation is selected by the keywords and not eliminated by considering its limits, the system will evaluate the correlation to give a value for heat transfer. The lines of the correlation are stored as records in the database, which are retrieved from the database tables and sorted on the line number field. The correlations are expressed in a high-level language designed for their representation, which is procedural rather than declarative, (Fig. 2, Table 4). Thus, to obtain a value from a table of X vs Y and Z all that is required is the location of the table, two expressions for Y and Z and a name to which the derived value of X will be assigned, e.g.

```
[behb_83  a  2  coeff,behb_83.dat,z/d,table.]
```

The correlation language is interpreted by the KBS. The Prolog grammar rule notation is used to define a definite clause grammar for the language. Recursive descent parsing is used in evaluating the arithmetic expressions.

Once all the lines in the correlation have been processed and the heat transfer obtained, the parameter values derived during evaluation of the correlations are discarded. This allows the parameter names to be safely reused in other correlations.

Evaluation of arithmetic statements

If a line of the correlation is recognised as an arithmetic statement, then the statement is divided and the right hand side is evaluated as an arithmetic expression. The syntax implemented is similar to Fortran, the exceptions being that variables are case-insensitive and are not restricted in length.

The expression is input as a string. This is processed in two steps, tokenisation and recursive descent parsing. Tokenisation separates the string into words (variables), numbers and symbols (operators, e.g. +, − and parentheses). Each of these tokens is stored separately in a list. The list is parsed to yield a number which is the value of the expression. The parser will evaluate expressions with any order and number of operators and with parentheses nested to any depth.

Evaluation of heat transfer for a single circular jet

In addition to the general methods such as evaluating arithmetic statements and obtaining values from tables, more-specialised methods can be implemented. An example is the derivation of area mean heat transfer coefficients and flux for a single circular jet from a set of graphs giving the radial distributions of local Nusselt number and jet effectiveness (a measure of the dissipation of the jet temperature).

An analysis is performed using an empirical model for jet temperature dissipation and heat transfer described by Goldstein and Behbahani.[14] This analysis is widely applicable.

Publication code
Configuration identifier.
Line number.
Graph Y parameter name ($f(Nu)$).
Graph X parameter name (x/D).
Expression, variable or value A.
Expression, variable or value B.
(Typically Re and z/D.)
The string 'graph'.

Data is often presented in publications as curves of $f(Nu)$ vs x/D at given values of Reynolds number and z/D. The database records facts about each of these curves in the form of a table. There will typically be a number of curves in a graph and a number of graphs in a reference.

Publication code
Configuration identifier.
File name
Value for Re or z/D.
Value for Re or z/D.

For instance, if the keywords and limits of a correlation indicate an analysis using this empirical model, the following steps are carried out. The information pertaining to the graphical data is searched to determine the number of curves which should be used in interpolation to give a curve of $f(Nu)$ vs x/D at the required values of Re and z/D (Fig. 4). The search proceeds by generating alternative combinations of curves, (a_1, b_1) (a_2, b_2) and (a_3, b_3), and then performing a two-stage test on the curves. The first part of the test determines if the required point (a, b) lies within the triangle (with vertices (a_1, b_1) (a_2, b_2) and (a_3, b_3)) formed by the values of Re and z/D for the generated combination of curves. If this test succeeds the evaluation function, the sum of the distances from the required point to the vertices of the triangle, is calculated. The set of curves giving the smallest evaluation function is selected for use in interpolation.

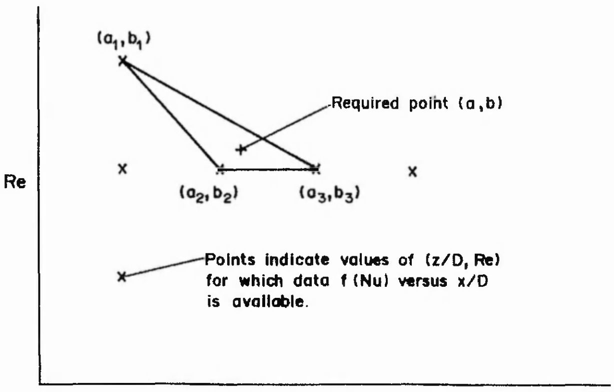


Fig. 4. Selection of three curves for use in interpolation.

The process to find two curves for interpolation is similar. While interpolation involving two curves is simpler, the process is likely to be successful because the required values of Re and z/D often will not fall between the values available in the graphs.

Once the selection of the best curves for interpolation is complete, a weight is calculated for each curve. The remaining analysis is largely procedural and involves arrays of numeric data. Processing of this type is more appropriately done using a procedural language. The routine is written in C and is closely-coupled with the Prolog KBS.

DISCUSSION

Logic programming using Prolog was found to provide a suitable method for implementing the correlation database, in that it provides a single language which facilitates easy access to relational database tables through pattern-matching. It also permits the addition of knowledge-based capabilities (for example the hierarchical network of keywords used to ensure that the information selected from the database is relevant) which are not found in conventional programs accessing databases.

A method was developed to store the information relevant to a correlation in a relational form. The system, working in the way an engineer would work, is able to apply rules to this information to select relevant valid correlations.

The application of knowledge-based techniques greatly improves the ease and reliability with which the most appropriate data can be obtained, and also provides the means by which the data extracted can be manipulated to provide an answer in the form required by an engineer.

A provisional selection of correlations is based on keywords, as would the selection of correlations in a more conventional database. Logic programming is used to represent the relationships between the keywords in the form of a graph, and the resulting information is used to ensure that the user specifies a meaningful and valid combination of keywords.

The retrieval of information from text databases has been considered by Parsaye et al.,[10] who proposed a method known as "concept indexing". This method makes use of a group of keywords which have been selected by human experts to describe a document. The keywords are organised into a hierarchy (again by human experts) linking each keyword to other keywords which are either more-general or more-specific. Each document is represented or indexed by a frame, and the value of each slot in the frame is filled by a word taken from the corresponding keyword hierarchy. Document retrieval is based on the contents of these slots. The concept of hierarchies of keywords has been successfully applied to improve the selectivity of the retrieval of publications from the correlation database, but without using frames. A problem was foreseen in the event of a keyword appearing more than once in a hierarchy (i.e., being ambiguous). The frame-indexing method would retrieve documents some of which may be inappropriate. The method developed in this study does not suffer from this problem, as any ambiguous keywords are searched for along with other keywords higher in the hierarchy, thus putting the keywords concerned in a context and giving a unique combination of keywords.

A final selection of correlations is based on a check that specified parameter expressions are within the limits given or implied in a publication. The representation method chosen, which uses an interpreter to evaluate an expression which is then compared with the associated minimum and maximum limits, allows any number of limits to be specified. These expressions can be made up of any of the parameter names known to the system. This approach is more flexible and requires less storage space than conventional database methods, which would use pairs of fields to represent minimum and maximum values for pre-determined expressions.

The knowledge base is used to decide which parameters are relevant in describing numerically a particular physical situation. The KBS then selects equations relating to these parameters and, using an inference technique developed specifically for this purpose, allows the user flexibility in the choice of the parameters he may use to give a quantitative description of the heat transfer system.

The selection of correlations is aided by the above techniques. Once correlations have been selected, the system can derive values for the heat transfer coefficient from them. The symbolic manipulation capability of Prolog and the provision of a definite clause grammar notation made the speedy implementation of an interpreter for the correlation data a practical proposition. The high-level language developed to represent the correlations allows easy access to tabular data and to complex but standard calculation methods.

The main disadvantage found in using logic programming was the representation of procedural processes, particularly those involving arrays. The difficulty was overcome by programming those parts of the system in a procedural language, in this case C. In order to give an efficient system, the facility to closely couple the two languages to give one process must exist. This facility is widely available in recent implementations of Prolog.

CONCLUSIONS

A large quantity of heat transfer data of interest to engineers is available in the form of correlations. Severe limitations in the ability of conventional text based databases to provide access to this data and to present this data in a form convenient for an engineer are identified. Logic programming and knowledge-based system techniques have been applied to overcome these limitations.

The information describing the correlations, and the correlations themselves (with the exception of the numerical values in tabular form), have been represented in a form which can be stored in a relational database. Data in this form can be easily accessed using logic programming techniques.

The benefits obtained from the use of knowledge-based techniques include:

(i) The database will be both more specific and more reliable in the selection of relevant correlations because of the assistance given to the user in specifying keywords.

(ii) The user is given the maximum choice in the parameters he uses to specify the system geometry and flow properties: all required parameters are derived from the minimum user input.

(iii) The evaluation of correlations can be optionally restricted to those that are valid, i.e. the system geometry and flow properties are within the range covered by the correlation.

(iv) A high-level language has been developed to represent the correlations. Prolog provided a suitable tool for the development of an interpreter which forms part of the database system. The interpreter is used to evaluate the correlations at the conditions specified by the engineer, and so provide a numerical answer to his query for a heat transfer value.

Acknowledgements—The post held by Mr M. A. Moss is funded by the Polytechnics and Colleges Funding Council (PCFC) under their special research initiative, and is gratefully acknowledged. The authors also wish to thank Mrs D. Corlett for her work in maintaining the collection of jet impingement literature on which much of this work is based.

REFERENCES

1. Button B. L. and Wilcock D. Impingement heat transfer—a bibliography 1890–1975. *Previews Heat Mass Transfer* **4**(3), 83–89 (1978).
2. Button B. L. and Jambunathan K. Jet impingement heat transfer: a bibliography 1976–1985. *Previews Heat Mass Transfer* **15**(2), 149–178 (1989).
3. Adeli H. and Hung S. L. A production system and relational database model for processing knowledge of earthquake-resistant design. *Engng Applic. Artif. Intell.* **3**, (1990).
4. Van der Spec *et al.* The paint advisor. *Expert Syst.* **7**, 4 (1990).
5. Lloyd J. W. An introduction to deductive database systems. *Aust. Comput. J.* **15**, 52–57 (1983).
6. Kowalski R. Logic for data description. In *Readings in Artificial Intelligence and Databases* (Mylopoulos J. and Brodie M. L., Eds). Morgan Kaufmann (1989).

7. Brodie M. L. and Jarke M. On integrating logic programming and databases. In *Expert Database Systems—Proceedings from the first International Workshop, Kiawah Island, South Carolina 1984* (Kerschberg L., Ed.). Benjamin/Cummings, Reading (1986).
8. Torsun I. S. and Ng Y. M. Tightly-coupled expert database systems interface. In *Research and Development in Expert Systems V—Proc. Expert Systems '88 the 8th Ann. Tech. Conf. British Computer Soc. Spec. group Exp. Syst. Brighton 12–15 Dec. 1988* (Kelly B. and Rector A. L., Eds). Cambridge Univ. Press, Cambridge (1989).
9. Luicas R. *Database Applications using Prolog*. Ellis Horwood (1988).
10. Parsaye K., Chignell M., Khoshafian S. and Wong H. *Intelligent Databases*, pp. 293–346. Wiley, London (1989).
11. SD-Prolog Reference Manuals, Quintec Systems Ltd (1986) and (1987).
12. MacRandal D. Semantic networks. In *Approaches to Knowledge Representation: an Introduction* (Edited by Ringland G. A. and Duce D. A.), pp. 45–79. Research Studies Press (1988).
13. Rowe N. C. *Artificial Intelligence Through Prolog*. Prentice-Hall, Englewood Cliffs, NJ (1988).
14. Goldstein R. J. and Behbahani A. I. Impingement of a circular jet with and without crossflow. *Int. J. Heat Mass Transfer* **25**(9), 1377–1382 (1982).

APPENDIX A

Portion of Prolog code used to obtain and derive input

```
/*---------
  DERIVE GEOMETRIC AND FLOW
  CONDITIONS FROM MINIMUM INPUT

Parameter values are derived from rules 'ip_rule/0' using method
similar to the rule-cycle hybrid from Rowe 1988.
------------*/
ip_values(Keywords):-
  obtain_required_parameters(Keywords,Names,Texts),
  ip_values1(Names,Texts),
  show_parameter_names_texts_and_values(Names,Texts,17).

/*--------
Check whether all required values have been derived, if not ask for
input and use rule cycle hybrid to derive further values
----------*/
ip_values1(L,_):-
  ip_values2(L).
ip_values1(Names,Texts):-
  show_parameter_names_texts_and_values(Names,Texts,17),
  write ("\n\nEnter pairs of parameter names and values"),
  write ("eg d 0.001 m 0.005 z 0.03\n"),
  read_list_name_value_pairs,
  not (hybrid),
  ip_values1(Names,Texts).

/*-------
All required values have been derived if calls to all the predicates
holding the parameter values (eg for mass flow rate m(Value):-!.)
succeed.
---------*/
ip_values2([]).
ip_values2([H:T]):-
  X=.. [H, A],
  call(X),
  ip_values2(T).

/*-------
Hybrid/0 repeats calls to the set of input rules ip_rule/0,
  eg for mass flow rate ip_rule:- m1.
The call to these rules is repeated until a call is made where no new
facts are derived.
Differs from Rowe in starting the call to ip_rule at the start of the set
of rules after each new fact derived.
----------------*/
hybrid:-
  call(ip_rule),
  !,
  hybrid.

/*-------------
  RELATIONS BETWEEN VARIABLES
These are the rules called, indirectly, by the rule hybrid/0.
-----------*/
/*-------
Mass flow through a single jet
---------*/
m1:-
  v(V),
  a(Area),
  rho(Rho),
  M is Rho*Area*V,
  asserta(m(M):-!).

m1:-
  t1(T1),
  a(Area),
  p1(P1),
  p_ratio(Rp),
  gas_const(R),
  gamma(G),
  A is (G+1)/G,
  B is 2/G,
  C is 2*G/(1-G)/R,
  p_pow(Rp,A,Rp1),
  p_pow(Rp,B,Rp2),
  D is C/T1*(Rp1-Rp2),
  p_pow(D,0.5,E),
  M is P1*Area*E,
  asserta(m(M): - !).
```

APPENDIX B

Nomenclature

A, a	constant
Cp	specific heat (w/kg.K)
D, d	nozzle exit dia. (m)
h	heat transfer coefficient (w/m²K)
lx	Jet to jet spacing parallel to crossflow
ly	Jet to jet spacing perpendicular to crossflow
m	mass flow of one jet
mc	cross flow, mass flow per jet
mu	dynamic viscosity
Nu	hD/λ
nx	Number of jet holes on X axis in the array
ny	Number of jet holes on Y axis in the array
$P1$	Total pressure
$p2$	Static pressure
Pr	Prandtl No.
R	Gas Constant.
Re	Reynolds No.
T	temperature (K)
t	static temperature (K)
v	velocity (m/s)
z	axial distance nozzle to plate (m)
x	radial distance from stagnation point (m)

Greek symbols

ρ	density (kg/m³)
γ	Ratio of specific heats.

Appendix E
The Importance of Inference Strategy When Considering a Set of Simultaneous Equations

The comparison between Leonardo (Creative Logic Ltd.) and SD Prolog (Quintec Systems Ltd.) is based on experience gained during the initial development of a knowledge based system for the analysis of jet impingement heat transfer. Prolog and Leonardo use or make available different inference strategies, the effect of these strategies on the inference of numerical values using the one dimensional compressible steady state flow equations was investigated.

A number of methods for inferring or propagating values from rules have been developed. The most widely used are forwards chaining (also known as data driven chaining) and backwards chaining (also known as goal directed chaining). These inference methods can act in either a depth first or a breadth first manner.

Leonardo implements depth first backwards and depth first forwards chaining, the chaining methods can be used together or separately. Prolog performs only depth first backwards chaining automatically but rules can be written to carry out all the other inference methods mentioned above.

Rules are defined in Leonardo in the form

if <conditions> then <conclusions>.

A condition tests the value of an object (eg. mass flow < 0.1 or nozzle type is orifice). A conclusion assigns a value to an object. While backward chaining if the value of any object in the conditions is not known then Leonardo will attempt to obtain the value from another rule where the value of the object is given by the last of the conclusions.

Rules in Prolog derive only one conclusion. The advantage of being able to derive more than one conclusion from a rule in Leonardo is severely limited by the fact that its inference engine will only chain to the last of the conclusions given in a rule.

It is possible to create a set of rules in which more than one path through the rules is available to derive a given conclusion, but the information required to satisfy all the conditions contained in the rules in the different paths is not the same. To successfully derive the conclusion using a given set of input data the inference engine must select the path through the rules for which the conditions can all be satisfied by the data available. Backtracking is the name given to a mechanism where if the first choice of rule does not start the selection of the appropriate chain of rules then another rule is tried and another path through the rules constructed. Backtracking cannot be carried out in Leonardo and so if an unsuitable path is begun then the inference engine is unable to switch to a suitable path. The appears to be a serious limitation.

The one dimensional compressible flow equations can be expressed in the form of rules. The equations allow each parameter to be derive from more than one equation. The static temperature, t2, can be derived from the following equations, assuming constant gas properties, by specifying five different combinations of parameter value.

$$t2 = f(T_{ratio}, T1) \quad (1)$$

$$T_{ratio} = f(P_{ratio}) \quad (2)$$

$$P_{ratio} = f(P1, p2) \quad (3)$$

$$P_{ratio} = f(P1, T1, m) \quad (4)$$

$$P_{ratio} = f(p2, T1, m) \quad (5)$$

Where total pressure and temperature are P1 and T2 respectively, the static pressure is p2 and the mass flow is m.Other relationships can be derived but these are considered the most likely to be useful.

The effect of three inference methods will be considered, opportunistic forwards chaining and depth first backwards chaining, as implemented in Leonardo, and depth first backwards chaining with backtracking as occurs in Prolog.

Forwards chaining

Forwards chaining uses a rule to produce a result when the information exists to satisfy the conditions. If P1, T1 and m were specified then (4) would fire to give P_{ratio}, this would enable (2) to fire to give T_{ratio} and then (1) to give t2 as required.

Further equations could be specified to make the knowledge base more general, for example:

$$p2 \quad = f(P_{ratio}, P1) \qquad (6)$$

The inference strategy used by Leonardo causes any rule that can fire to be fired. The addition rule (6) would cause p2 to be derived after (4) was used, this would be followed by P_{ratio} being needlessly rederived using (3) and (5). The rederivation of values can be avoided by putting a condition in the rule which prevents it being used if another rule has already been used to derive the same conclusions. The derivation of other values which are not required cannot be prevented except at the loss of generality in the knowledge base.

Backwards chaining without backtracking

To derive t2 using backwards or goal directed chaining Leonardo would try (1) , and because T_{ratio} is not known would then try (2) in order to find a value for T_{ratio}. Similarly (3) would then be tried. If T1, P1 and p2 had been specified (3) would provide P_{ratio} and then (2) would provide T_{ratio} enabling (1) to successfully give t2. If however T1, P1 and m had been provided then backwards chaining would again lead to (3), a rule to derive p2 would then be looked for, because no rule exists to derive p2 from T1, P1 and m Leonardo would make no further progress towards finding t2. This inference strategy fails to find the alternative route to a solution using (4).

Because Leonardo can use forwards and backwards chaining at the same time a value for t2 can be derive as shown using forwards chaining.

Backwards chaining with backtracking

The backtracking mechanism in Prolog enables backwards chaining alone to arrive at a solution for t2. This works because the backtracking mechanism records its choice of rule, after choosing (3) and failing to make any further progress Prolog backtracks to the point before (3) was chosen and tries a different rule, (4), which the enable it to eventually give t2.

The backtracking method will enable Prolog to derive a solution if the requisite rules exist. Leonardo relys on the use of forward chaining under some circumstances, if the number of rules in the knowledge base is large this will result in the needless use of many rules at the cost of computing time.

Appendix F
Extrapolation of Empirical Heat Transfer Results using the Single Circular Jet Correlation

A correlation for the local heat transfer Nusselt number has been derived (Jambunathan *et al* 1992). This correlation expresses Nusselt number as a function of nozzle exit Reynolds number raised to an exponent, this exponent varies with nozzle-to-plate spacing (z/D) and with jet-to-jet spacing (x/D).

$$\frac{Nu}{Re^{f(x/D,z/D)}} = f_1(x/D,z/D)$$

This relationship can be used to extrapolate radial distributions of Nusselt number. Comparisons between the correlation and experimental results suggested that reasonable results will be obtained for extrapolation for Reynolds number in the range $5000 \le Re \le 124000$. A simple algorithm was developed to perform the extrapolation:

(i) A curve of Nu vs. x/D at the appropriate z/D is obtained, probably by interpolation. Many sets of empirical results are available from experiments in which measurements were performed at various nozzle-to-plate spacings whilst maintaining a constant nozzle exit Reynolds number.

(ii) f_1 is independent of Reynolds number so the following expression can be derived

$$Nu_b = Nu_a \left(\frac{Re_b}{Re_a}\right)^{f(x/D,z/D)}$$

which yields the extrapolated local Nusselt number Nu_b for the Reynolds number Re_b where the Nusselt number Nu_a was obtained empirically at a Reynolds number Re_a.

APPENDIX G
AN APPROACH TO OBTAINING A NEW CORRELATION FOR THE DATABASE

Each correlation in the database may be selected, evaluated and used in design. Hence only correlations of acceptable accuracy should be added to the database. The single most important practice observed whilst selecting correlations from the literature for addition to the database was to perform a comparison with other data. Care must be exercised to ensure that values compared are formed from the same parameters and expressed in the same units. The following comparisons were made during the construction of the database:

(i) The form of the Nusselt number distribution should be similar to that obtained from experiments using similar geometry and flow conditions.

(ii) The trends in variation of the Nusselt number should be similar. For example similar increases in Reynolds number should result in similar increases in Nusselt number.

(iii) Given similar geometric and flow conditions the values of Nusselt number obtained should be similar. However it should be borne in mind that heat transfer coefficients are very sensitive to changes in geometry (for example jet nozzle profile in the case of jet impingement). The new correlation should not lie significantly outside the range of scatter for similar correlations. If a large number of correlations are available then this function could be performed statistically.

(iv) If there are significant differences between correlations then these should be explicable (even if the cause identified is speculative to some degree). This tentative explanation should be reflected in the description associated with the correlation.

In adding new correlation data to the database the following conversions may be required:

(i) The nomenclature must correspond to that used in the database.

(ii) The expressions and data must be converted to SI units.

In addition some publications may present results for a number of different geometries. In cases where these variations cannot be summarised by non-dimensional parameters, for example studies have been performed on the effect of nozzle geometry on the heat transfer Nusselt numbers, then a separate description must be constructed for each geometry tested.

Appendix H
Source Code

Code written to realise the jet impingement heat transfer knowledge based database is given. The code is presented in five sections:

(i) *Formulation of a query.*

(ii) *Database search.* The code given in this section performs a search of the database and carries-out the checks on correlation validity. This code also makes use of a knowledgebase to control these checks and to modify the database query. Example rules used in modifying the search or the query are given in the main text (Chapter 5).

(iii) *Interpreter.*

(iv) *Network extension.*

(v) *Heat transfer correlation addition.*

Libraries for list manipulation, conversion between strings and other data-types, input and output (including a menu system designed for IBM compatible Personal Computers) are not included. Also omitted is procedural code (written in C) for evaluation of the local heat transfer due to a single circular impinging jet. The Prolog compiler employed (Quintec Systems Ltd 1986,1987) differs from the Edinburgh standard (Clocksin and Mellish 1985) in the following respects:

(i) The 'string' (an array of characters) is a valid data-type. The Edinburgh standard employs lists of characters.

(ii) The code provides access to external database files. Each record in these files is available to the Prolog code as if it were asserted as a clause in the internal Prolog database.

(iii) Procedural code can be invoked from Prolog, a clause functor corresponds to a particular procedural function, the clause arguments correspond to the function arguments. Data is trasnferred from Prolog to the procedural code and fron the procedural code to Prolog. Callbacks from the external code to the Prolog code are not used.

(iv) The position of the cursor on the display console can be controled, a menu system built using this feature was used in creating the user interface.

FORMULATION OF A QUERY

```
/*--------------------------------------------------------------------
Copyright (c) Michael A Moss and The Nottingham Trent University 1993

All rights reserved.
Use and copying of this software and preparation of derivative works based
on this software are permitted provided that this complete copyright and
permission notice is maintained, intact, in all copies and that full credit
is given to Michael A Moss and The Nottingham Trent University.

Michael A Moss and The Nottingham Trent University provide this software
without express or implied warranty.
--------------------------------------------------------------------*/
/*-------

        KEYWORD NETWORK INSPECTION
        ******************************

    Aid to input of Keywords for paper selection.
    ***********************************

The keywords are organised into hierarchies.
The keywords devide the possible physical systems into groups and
sub-groups.

A valid combination of keywords must be linked via subset/2 and not via
exclude/2. The keywords are represented as node numbers, the keyword name and
text explaining the meaning of each keyword is given in label/3.

subset(Parent,Child)
exclude(Node_1,Node_2)
label(Node,Keyword,Text)

subset/2,exclude/2 and label/3 are in external .dbf database files.

Predicates used by database access and update programs to obtain information
from this network.
---------*/

/*-------
Inheritance rules for sub groups. 'Subset' forms an acyclic graph.
Connectivity in the direction from root to branches.
sub_inherit/2 G is an ancestor of K.
sub_inherit1/2 G is either an ancestor of K or K itself.
path/2 gives the keywords connecting A to the root of the graph
which is indicated by subset(A,A).
----------*/
sub_inherit(G,K):-
    subset(G,K).
sub_inherit(G,K):-
    subset(G,G1),
    G \== G1,
    sub_inherit(G1,K).

sub_inherit1(G,K):-
    sub_inherit(G,K).
sub_inherit1(K,K).
```

```
sub_not_root(A,B):-
      subset(A,B),
      A \== B.

path(A,[]):-
      subset(A,A).
path(A,[A|Path]):-
      subset(Ancestor,A),
      Ancestor \== A,
      path(Ancestor,Path).

/*-------
exclude/2 is transitive
----------*/
exclude_trans(A,B):-
      exclude(A,B).
exclude_trans(A,B):-
      exclude(B,A).

/*--------
Inheritance rules for exclusions.

exclude_implys/2 gives the keywords B excluded by A, both explicitly
given in and implied by the exclude graph, but not considering
inheritance in the sub-group graph.

gives keywords connected to A in the cyclic graph 'exclude'.
--------*/
exclude_implys(A,B):-
      exclude_implys(A,B,[A]).

exclude_implys(A,B,Visited):-
      exclude_trans(A,B),
      not( member(B,Visited) ).
exclude_implys(A,C,Visited):-
      exclude_trans(A,B),
      not( member(B,Visited) ),
      exclude_implys(B,C,[B|Visited]).

/*------
excluded/2 succeeds if a member of the list exclude A,
excluded1/2 succeeds if any member of the first list is excluded
by the second.
-------*/
excluded(A,[H|T]):-
   exclude_implys(H,A).
excluded(A,[H|T]):-
   excluded(A,T).

excluded1([P|Ps],Ks):-
      excluded(P,Ks).
excluded1([P|Ps],Ks):-
      excluded1(Ps,Ks).
```

```
/*-------
Returns a list of the keywords which will be excluded by A
-----*/
will_exclude(A,Excluded):-
   bagof(Ex,exclude_implys(A,Ex),Excluded).

/*---------
all_child_keys/2 returns a list of the keywords at the next level in the
hierarchy. This list has the form [[Keyword nodes],[Keyword nodes 1]...]
where only one keyword node from each group can be selected
-------*/
all_child_keys(Parent,Cs):-
      bagof(B,sub_not_root(Parent,B),Bs),
      all_child_keys1(Bs,Cs).

all_child_keys1([],[]).
all_child_keys1([A|As],[Cs|Ds]):-
      bagof(B,exclude_implys(A,B),Bs),
      intersect([A|As],[A|Bs],Cs),
      delete_members(Bs,As,A1s),
      !,
      all_child_keys1(A1s,Ds).

/*------------
node_label_list(Nodes,Labels) - converts between a list of nodes and a
list of corresponding keywords
--------------*/
node_label_list([N|Ns],[K|Ks]):-
      label(N,K,_),
      !,
      node_label_list(Ns,Ks).
node_label_list([],[]).

/*------------
node_label_list(Nodes,Labels,Texts) - converts between a list of nodes and a
list of corresponding keywords and explainatory texts
--------------*/
node_label_list([N|Ns],[K|Ks],[T|Ts]):-
      label(N,K,T),
      !,
      node_label_list(Ns,Ks,Ts).
node_label_list([],[],[]).

/*--------
check_keys/1 succeeds if a list of keywords is a valid combination.
The check will test all combinations of paths up the tree from each
keyword and will succeed provided one of these combinations is valid.
----------*/
check_keys([]).
check_keys([K|Ks]):-
      node_label_list([N|Ns],Keywords,_),
      not( excluded(K,Ks) ),
      path(N,Path),
      delete_members(Path,Ns,N_rem),
      check_keys1(Path,N_rem),
      check_keys(N_rem).
```

```
check_keys1(Path,[]).
check_keys1(Path1,[N|Ns]):-
	path(N,Path2),
	delete_members(Path1,Path2,Path2a),
	delete_members(Path2,Path1,Path1a),
	not( excluded1(Path1a,Path2a) ),
	check_keys1(Path1,Ns).

/*------
draw_tree/1 draws the tree sub/2 starting at node number A.
Output is to file which is then browsed using view/2.
----*/
draw_tree(Node):-
	label(Node,Keyword,_),
	stdout(Out),
	open(tree,"tree.txt",write),
	stdout(tree),
	write("Hierarcy of keywords below "),write(Keyword),
	write("     Press Return to continue.\n"),
	write(Keyword),write("\n"),
	Depth is 0,
	draw_tree1(Node,Cs),
	length(Cs,L),
	draw_tree2(Depth,[L],Cs),
	write("\n"),
	stdout(Out),
	close(tree),
	!,
	open(tree1,3:3,21:76,b:cyan:w,'Keyword Hierarchy'),
	stdout(tree1),
	view("tree.txt",13),
	stdout(Out),
	close(tree1),
	!.
draw_tree1(Node,Cs):-
	bagof(C,sub_not_root(Node,C),Cs),
	!.
draw_tree1(Node,[]).
draw_tree2(Depth,Ls,[]):-
	!.
draw_tree2(Depth,[L|Ls],[N|Ns]):-
	label(N,K,_),
	reverse([L|Ls],LR),
	draw_tree3(Depth,LR),
	write("|------"),write(K),write("\n"),
	draw_tree1(N,Cs),
	length(Cs,L1),
	Depth1 is Depth + 1,
	draw_tree2(Depth1,[L1,L|Ls],Cs),
	L2 is L - 1,
	draw_tree2(Depth,[L2|Ls],Ns).
draw_tree3(0,_):-
	!.
draw_tree3(Depth,[P|Ps]):-
	P > 1,
	write("|"),
	tab(5),
	Depth1 is Depth - 1,
```

```
        !,
        draw_tree3(Depth1,Ps).
draw_tree3(Depth,[P|Ps]):-
        tab(6),
        Depth1 is Depth - 1,
        !,
        draw_tree3(Depth1,Ps).

draw_tree_and_fail(A):-
    draw_tree(A),
    fail.

/*-----------------

   PROMPT FOR KEYWORDS TO DESCRIBE THE HEAT TRANSFER PROCESS
   ********************************************************

Use a hypertext-like display to show the available keywords and their
meanings. The user may select a keyword from the menu

-----------------*/

/*--------
hprompt/1 returns a list of valid keyword node numbers
open a window for the keyword explainations.
---------*/
hprompt(Ks):-
    stdout(Out),
    clear(Out),
    write("                    Initial Search using Keywords\n"),
    write("                    ----------------------------"),
    write("\nSelect keywords to describe your heat transfer process.\n"),
    write("Select a keyword from the menu or choose one of the options:"),
    write("\n'No keyword' to reject all the displayed keywords,"),
    write("\n'Draw keyword tree' to display the keyword hierarchy,"),
    write("\n'Back step' to re-do previous selection."),
    bagof(R,subset(R,R),Rs),
    stdout(Out),
    open(exp,8:0,17:61,w:blue:w,'Keyword meanings'),
    stdout(exp),
    hprompt1(Rs,Ns),
    stdout(Out),
    close(exp),
    clear(Out),
    node_label_list(Ns,Ks,_),
    write("\n\nKeywords selected:\n"),write(Ks),write("\n"),
    !.

/*----
Prompt for keywords from each hierarchy
-----*/
hprompt1([],[]).
hprompt1([R|Rs],All_nodes):-
    hprompt2([R],Nodes1),
    hprompt1(Rs,Nodes2),
    append(Nodes1,Nodes2,All_nodes).
```

```
/*-----
Given a keyword node list, find the keyword nodes in the next level in the
hierarchy for the first keyword node. Obtain a selection of these. Using this
selection make a recursive call to the predicate. Then make recursive
calls for the remaining predicates in the first keyword node list.
-----------*/
hprompt2([],[]).
hprompt2([P|Ps],Ns):-
   hprompt3(P,N1s),
   hprompt2(N1s,N2s),
   append(N1s,N2s,N3s),
   hprompt2(Ps,N4s),
   append(N3s,N4s,Ns).

/*-----
Find all groups of keyword nodes at the next level in the hierarchy.
--------*/
hprompt3(Parent,Selected_children):-
   all_child_keys(Parent,Children),
   hprompt4(Parent,Children,Selected_children).

/*-------
hprompt4(Parent node,[[Child nodes],[Child nodes],...],[Nodes])
Prompt for a keyword from each of these groups in turn, return a list
of keyword node numbers.
---------*/
hprompt4(_,[],[]).
hprompt4(Parent,[C|Cs],Selected_children):-
   hprompt5(Parent,C,S),
   hprompt4(Parent,Cs,Ss),
   append(S,Ss,Selected_children).

/*----
To allow backtracking to respecify previous keyword selection
-----*/
hprompt5(Parent,Nodes,Sel_node):-
   stdout(Out),
   clear(Out),
   node_label_list(Nodes,Keywords,Texts),
   show_expansions(Keywords,Texts),
   hprompt_options(Parent,Nodes,Keywords,Options_list),
   menu_and_fail("Select a keyword",vertical,6:60,white:blue,Options_list),
   selected_keyword([-1]),
   !,
   retract( selected_keyword([-1]) ),
   fail.
hprompt5(Parent,Nodes,Sel_node):-
   retract( selected_keyword(Sel_node) ).
hprompt5(Parent,Nodes,Sel_node):-
   hprompt5(Parent,Nodes,Sel_node).

/*---
Call to menu/5 which fails on backtracking
----*/
menu_and_fail(A,B,C,D,E):-
      menu(A,B,C,D,E),
      !.
```

```
/*------
Write the text explaining the meanings of the keywords
-------*/
show_expansions(_,[]).
show_expansions([K|Ks],[Text|Ts]):-
   write(K),write(":\n"),
   write_line(58,Text),
   write("\n"),
   !,
   show_expansions(Ks,Ts).

/*-----
Create a list of Keywords and other options for use by menu/5
--------*/
hprompt_options(Parent,[],[],
   [
   "No keyword":assert(selected_keyword([])),
   "Draw keyword tree":draw_tree_and_fail(Parent),
   "Back step":assert(selected_keyword([-1])),
   "Quit":abort
   ]).
hprompt_options(Parent,[C|Cs],[K|Ks],[K:assert(selected_keyword([C]))|Options]):-
   hprompt_options(Parent,Cs,Ks,Options).

□
/*---------

   DERIVE GEOMETRIC AND FLOW CONDITIONS FROM MINIMUM INPUT
   *****************************************************

Primary values derived from rules 'ip_rule/0' using method similar to
the rule-cycle hybrid from Rowe 1988.
Modified to obtain the rules from an external database. 5/11/91
Modified to use only a limited set of the required parameters for input 9/5/92
-----------*/

ip_values(Keywords_in,Names,Values):-
   input_vars(Keywords_in,Names,Texts),
   collect_eqns(Keywords_in),
   list_string_atom(Names,Names_a),
   required_vars(Keywords_in,All_names,All_texts),
   list_string_atom(All_names,All_names_a),
   repeat,
   ip_values1(Names_a,Texts,All_names_a),
   stdout(Out),
   clear(Out),
   write("          Quantitative Definition of Heat Transfer Geometry and Flow\n"),
   write("          --------------------------------------------------------"),
   show_results(All_names_a,All_texts,20),
   write("\n"),tab(58),write("Press any key ..."),
   getkey(_),
   parameter_values_list(All_names,Values).
```

```
/*--------
(i) use rule cycle hybrid to derive values from any currently known
values, useful when the description has been partially formed by the
knowledge base in the course of broadening a search
(ii) check whether all required values have been derived (ip_values2/1),
if not
(iii) ask for input and use rule cycle hybrid to derive further values
----------*/
ip_values1(Names,Texts,_):-
    hybrid.
ip_values1(Names,_,_):-
    ip_values2(Names).
ip_values1(Names,Texts,All_names):-
    stdout(Out),
    clear(Out),
    write("       Quantitative Definition of Heat Transfer Geometry and Flow"),
    write("\n       -------------------------------------------------------"),
    show_results(Names,Texts,20),
    write("\n\nEnter pairs of parameter names and values "),
    write("eg d 0.001 m 0.005 z 0.03 RETURN"),
    write("\nThe program will fill in values itself where possible."),
    write("\n"),
    read_name_val(NameValPairs),
    !,
    ip_values3(All_names,NameValPairs),
    ip_values1(Names,Texts).

ip_values2([]).
ip_values2([H|T]):-
    X =.. [H,A],
    call(X),
    ip_values2(T).

ip_values3(_,[]).
ip_values3(All_names,[Name,Val|T]):-
    check_consistancy(All_names,Name,Val),
    not(hybrid),
    ip_values3(All_names,T).

/*-------------
Check that a value for a parameter has not already been derived
or entered. If it has and it differs from the previous value
then offer options.
---------*/
check_consistancy(All_names,Name,Val):-
    X =.. [Name,A],
    not( call(X) ),
    !,
    Y =.. [Name,Val],
    asserta(Y:-!).
check_consistancy(All_names,Name,Val):-
    X =.. [Name,A],
    call(X),
    A =:= Val,
    !.
check_consistancy(All_names,Name,Val):-
    X =.. [Name,A],
```

```
   call(X),
   A =\= Val,
   !,
   write("\n"),write(Name),write(" = "),write(A),write(" has already been "),
   write("derived.\nOPTIONS:Continue discarding "),write(Name),write(" = "),
   write(Val),write(" <c>\n        Restart I/P <r>"),
   write("\n        Quit <q>\nEnter Option :"),
   readln("c",Ans),
   check_consistancy1(All_names,Ans).

check_consistancy1(_,"c").
check_consistancy1(_,"q"):-
   break.
check_consistancy1(All_names,"r"):-
   clear_ip(All_names),
   fail.

/*-------
Hybrid repeats calls to the set of input rules ip_rule/0.
The call to these rules is repeated until a call is made where no new facts
are derived.
Differs from Rowe in starting the call to ip_rule at the start of the set
of rules after each new fact derived.
---------------*/
hybrid:-
   call(ip_rule),
   !,
   hybrid.

/*--------
Compile list of parameter names and associated text for which values are
required.

The required parameters is based on the list of keywords describing the
heat transfer process and information from the keyword network given in
the database diction/4.
-------*/
required_vars(Query_keywords,Names,Texts):-
   required_vars1(Query_keywords),
   required_vars2(Names,Texts).

required_vars1(Query_keywords):-
   diction(Name,Text,L,_),
   split_list_string(Keywords,L),
   all_members(Keywords,Query_keywords),
   assert(req_parm(Name)),
   assert(req_parm_text(Text)),
   fail.
required_vars1(_).

required_vars2([Name|Names],[Text|Texts]):-
   retract(req_parm(Name)),
   retract(req_parm_text(Text)),
   !,
   required_vars2(Names,Texts).
required_vars2([],[]).
```

```
/*--------
Select input parameters.

As for required_vars/3 but only parameters marked for input are selected
-------*/
input_vars(Query_keywords,Names,Texts):-
   input_vars1(Query_keywords),
   input_vars2(Names,Texts).

input_vars1(Query_keywords):-
   diction(Name,Text,L,"y"),      /*Add only names marked for input*/
   split_list_string(Keywords,L),
   all_members(Keywords,Query_keywords),
   assert(req_parm(Name)),
   assert(req_parm_text(Text)),
   fail.
input_vars1(_).

input_vars2([Name|Names],[Text|Texts]):-
   retract(req_parm(Name)),
   retract(req_parm_text(Text)),
   !,
   input_vars2(Names,Texts).
input_vars2([],[]).

/*--------
Display a list of the required parameters showing values of parameters
derived so far.
----------*/
show_results([],[],A).
show_results(A,B,Lines):-
   Lines =< 0,
   write("\n"),tab(58),write("Press any key ..."),
   getkey(_),
   show_results(A,B,20).
show_results([Name|Names],[Text|Texts],Lines):-
   X =.. [Name,Val],
   call(X),
   write("\n"),write(Name),
   stdout(Out),
   cursor(Out,R:C),cursor(Out,R:4),write(Val),
   cursor(Out,R:13),write(" "),
   cursor(Out,R:14),write(Text),
   Lines_left is Lines-1,
   show_results(Names,Texts,Lines_left).
show_results([Name|Names],[Text|Texts],Lines):-
   write("\n"),write(Name),
   stdout(Out),
   cursor(Out,R:C),cursor(Out,R:14),write(Text),
   Lines_left is Lines-1,
   show_results(Names,Texts,Lines_left).

/*--------
Clear any existing values for the input parameters
-------*/
clear_ip([Name|Names]):-
   R =.. [Name,A],
```

```
   call(R),
   retract(R:-!),
   !,
   clear_ip(Names).
clear_ip([Name|Names]):-
   !,
   clear_ip(Names).
clear_ip([]).

/*------------
Common constants.
These are defined here to save repeated entry into database.
Constants given for air at 300K
--------*/
mu(0.000018).
lambda(0.02624).
cp(1006.3).
pi(3.141592654).
gamma(1.4).
gas_const(287.0).
e(2.718281828).

pr(Pr):-
      mu(Mu),
      cp(Cp),
      lambda(La),
      Pr is Mu*Cp/La.

/*-----------
Obtain the equations inter-relating the relevant parameters.
Equations/statements given in strings are obtained from the database
eqn/5 and converted into a forward chaining Prolog rule.
------*/
collect_eqns(Query_keywords):-
   write("\nAdding parameter equtions from the external database."),
   eqn(Name,Keywords,Eqn_no,1,_), /*First record in a series*/
   split_list_string(K1,Keywords),
   all_members(K1,Query_keywords),
   collect_eqn1(Query_keywords,Name,Eqn_no),
   collect_eqn2(1,Eqns),
   make_tokens(Eqns,Eqns1),
   math_parse(Eqns1,Eqns2),
   check_unknowns(Eqns2,Unknowns),
   no_duplicates(Unknowns,Unknowns1),
   assert_eqns(Name,Eqns2,Unknowns1),
   write("\nEquation for "),write(Name),write(" added."),
   fail.
collect_eqns(_).

/*-------
Collect all the statements for a given relationship.
---------*/
collect_eqn1(Query_keywords,Name,Eqn_no):-
   eqn(Name,Keywords,Eqn_no,Line_no,Statement),
   split_list_string(K1,Keywords),
   all_members(K1,Query_keywords),
   assert(selected_statement(Line_no,Statement)),
   fail.
```

```
collect_eqn1(_,_,_).

collect_eqn2(N,[S|Ss]):-
   retract(selected_statement(N,S)),
   !,
   N1 is N + 1,
   collect_eqn2(N1,Ss).
collect_eqn2(_,[]).

math_parse(Tokens,Structures):-
      parse(Tokens,Structures,_),
      !.

/*-------
Find all the parameters not derived by the statements.
--------*/
check_unknowns(Eqns2,Unknowns):-
   check_unknowns(Eqns2,[],_,Unknowns),
   !.

check_unknowns([],_,[],[]):-
   !.
check_unknowns([eval(R,Tokens)|Es],Results_vars,[R|Rs],Ip_vars):-
   !,
   extract_variables(Variables,Tokens,_),
   check_unknowns(Es,[R|Results_vars],Rs,Ip_vars_old),
   delete_all1(Results_vars,Variables,Ip_vars_new),
   append(Ip_vars_new,Ip_vars_old,Ip_vars).
check_unknowns([while(_,E1)|Es],Results_vars,Rs,Ip_vars):-
   !,
   check_unknowns(E1,Results_vars,R1,Ip1),
   append(R1,Results_vars,R2),
   check_unknowns(Es,R2,R3,Ip_vars_old),
   append(R1,R3,Rs),
   append(Ip1,Ip_vars_old,Ip_vars).
check_unknowns([if(_,E1,E2)|Es],Results_vars,Rs,Ip_vars):-
   !,
   check_unknowns(E1,Results_vars,R1,Ip1),
   check_unknowns(E2,Results_vars,R2,Ip2),
   append(R1,R2,R3),
   append(R3,Results_vars,R4),
   check_unknowns(Es,R4,R5,Ip_vars_old),
   append(R3,R5,Rs),
   append(Ip1,Ip2,Ips),
   append(Ips,Ip_vars_old,Ip_vars).
check_unknowns([if(_,E1)|Es],Results_vars,Rs,Ip_vars):-
   !,
   check_unknowns(E1,Results_vars,R1,Ip1),
   append(R1,Results_vars,R2),
   check_unknowns(Es,R2,R3,Ip_vars_old),
   append(R1,R3,Rs),
   append(Ip1,Ip_vars_old,Ip_vars).
check_unknowns([E|Es],R1s,Rs,Is):-
   !,
   check_unknowns(Es,R1s,Rs,Is).
```

```
module add_ip_rule.

export extract_variables/3.

/*--------
Parser
- input a list of tokens (integers, variable names and operator symbols).
- returns a list of variables.

Goal in form - extract_variables(Variables,Tokens,_).
---------*/

extract_variables(Variables) -->
	expression(Variables).

expression(Vars) -->
	constant(X),arithmetic_op(Op),expression(Ys),
	{
	append(X,Ys,Vars)
	}.
expression(X) -->
	constant(X).

arithmetic_op([*,*,-]) --> [*],[*],[-].
arithmetic_op([*,*]) --> [*],[*].
arithmetic_op([+]) --> [+].
arithmetic_op([-]) --> [-].
arithmetic_op([*]) --> [*].
arithmetic_op([/]) --> [/].

constant(Vars) -->
	['('],expression(Vars),[')'].
constant([X]) -->
	variable(X).
constant([]) -->
	[X],
	{
	number(X)
	}.
constant([]) -->
	[-],[X],
	{
	number(X)
	}.

variable(X) -->
	[X],
	{
	atom(X)
	}.

end_of_module.
```

```
/*-------
Convert the statements into the hybrid forwards/backwards
chaining rule of the form

ip_rule:-
   X is not known,
   obtain values for Y,Z...,
   evaluate the equations in the relationship X=f(Y,Z...),
   assert the new value for X.
--------*/
assert_eqns(Name,Eqns,Unknowns):-
   make_predicate(Unknowns,Predicate),
   gc(stack),
   !,
   string_atom(Name,N),
   X =.. [N,Dummy_var],
   Y =.. [N,Result],
   assert(ip_rule:-(not(X),Predicate,eval_eqns(Eqns,Y),assert(Y:-!)) ),
   !.

/*------
Converts list variable names [a,b,c...] to
a single predicate (a(A),b(B),c(C)...).
--------*/
make_predicate(Vs,P):-
      make_predicate1(Vs,Xs),
      gc(stack),
      make_predicate2(Xs,P).

make_predicate1([],[]):-
      !.
make_predicate1([V|Vs],[X|Xs]):-
      X =.. [V,A],
      make_predicate1(Vs,Xs).

make_predicate2([A],A):-
      !.
make_predicate2([A,B],P):-
      !,
      P =.. [',',A,B].
make_predicate2([A|As],P):-
      make_predicate2(As,Ps),
      P =.. [',',A,Ps].

/*-------
called to evaluate the equations when a hybrid rule is selected.
--------*/
eval_eqns(Eqns,Y):-
      eval_expr_list(Eqns,R),
      call(Y),
      clear_results(R).
```
□

DATABASE SEARCH

```
/*--------------

JET IMPINGEMENT DATABASE SEARCH AND ANALYSIS
*************************************************

--------------*/

/*-----
Sel_list has the form [Node nos.,[Keys for selected correlations]]
-------*/
search1(N,L,Sel_list):-
   hprompt(Query_keywords),
   time_seconds(Time),
   search3(Query_keywords,List,Num_pub,Time),
   N1 is N + 1,
   append([[N1,Num_pub,Query_keywords,List]],L,All_list),
   search2(N1,All_list,Sel_list).

search2(N,All_list,Sel_list):-
   menu("Options",vertical,15:3,w:blue:w,
   ["Carry out another search":assert(opt(search1(N,All_list,Sel_list))),
    "Look at a summary of the search results":assert(opt(search_summary(N,All_list,Sel_list))),
    "Quit":abort]),
   retract(opt(Option)),
   call(Option),
   Option =.. [_,N,All_list,Sel_list].

/*-------
Try each correlation in turn, select if keywords match
----------*/
search3(Query_keywords,_,_,T):-
   descript(Key,Key1,Keywords),
   search4(Query_keywords,Keywords),
   asserta(selected(Key,Key1)),
   fail.
search3(_,List,N,T1):-
   time_seconds(T2),
   Time is T2 - T1,
   search5(List,N),
   write(N),
   write(" correlations selected in "),write(Time),write("secs.\n"),
   !.

search4([],_).
search4([W|Ws],Keywords):-
      substring(Keywords,W),
      !,
      search4(Ws,Keywords).

/*--------
Collect selected references in a list
---------*/
search5([[Key,Key1]|List],N1):-
      retract(selected(Key,Key1)),
      search5(List,N),
```

```
    N1 is N + 1.
search5([],0).

/*--------
search_summary/3 summarise the results of all the searches
--------*/
search_summary(N,All_list,Sel_list):-
   stdout(Out),
   clear(Out),
   write("                 Summary of Keyword Searches\n"),
   write("                 ---------------------------\n"),
   reverse(All_list,R),
   write_pub_list(R),
   search_summary1(N,All_list,Sel_list).

/*---------
Display Keywords and number of records selected
---------*/
write_pub_list([]).
write_pub_list([[N,N1,Keywords,List]|T]):-
   writef("\nSearch No.%d.  %d records selected using keywords:\n",N,N1),
   write_line(78,78,Keywords),
   write_pub_list(T).

/*--------
Allow user to select a search or to try new searches
--------*/
search_summary1(N,All_list,Sel_list):-
   write("\nDo you want to select a list of records "),
   write("for further investigation?"),
   write(" y/n :"),
   readln("y","y"),
   write("Enter the search number:"),
   read_integer(Int),
   search_summary2(Int,All_list,Sel_list).
search_summary1(N,All_list,Sel_list):-
   search2(N,All_list,Sel_list).

search_summary2(N,[[N,_,Key_list_in,Sel_list]|T],[Key_list_in,Sel_list]).
search_summary2(N,[H|T],Sel_list):-
   search_summary2(N,T,Sel_list).


/*-------------

DISPLAY JET IMPINGEMENT RECORDS
********************************

-------------*/
show_record_list(Keys):-
   stdout(Out),
   open(htc_data,"htc_data.txt",write),
   stdout(htc_data),
   write("Current correlation selection                  "),
   write("Press Return to continue."),
   show_record(Keys),
   stdout(Out),
   close(htc_data),
```

```
   open(htc_data1,3:3,21:76,b:cyan:w,'Selected Correlations'),
   stdout(htc_data1),
   view("htc_data.txt",13),
   stdout(Out),
   close(htc_data1),
   write("\nThe selection has been saved as 'htc_data.txt'.").
show_record_list(_).

show_record([]).
show_record([[Key,Key1]|T]):-
   show_record1(Key,Key1),
   !,
   show_record(T).

show_record1(Key,Key1):-
   ref(Key,Authors,Year,Title,Source),
   writef("\n%-30s %t",Authors,Year),
   write("\n"),write_line(75,Title),
   write("\n"),write_line(75,Source),
   descript(Key,Key1,Keywords),
   write("\nDescriptive keywords:\n"),
   write(Keywords),
   write("\nLimits of validity for correlation:"),
   get_limits(Key,Key1,Limits),
   show_limits(Limits),
   write("\nCorrelation Expressions."),
   collect_expressions(Key,Key1,Expr),
   show_expressions(Expr).

show_limits([]).
show_limits([[Var,Low,High]|T]):-
   writef("\n     %t < %s < %t",Low,Var,High),
   show_limits(T).

show_expressions([]):-
   write("\n").
show_expressions([A|T]):-
   write("\n "),write(A),
   show_expressions(T).

/*-------------
Evaluate the correlations given for each selected publication
------------*/
eval_pub_list([]).
eval_pub_list( [[Key,Key1]|T] ):-
   eval_expr(Key,Key1),
   eval_pub_list(T).
```

```
/*---------

    KNOWLEDGE BASED BROADENING OF THE DATA BASE SEARCH.
    ******************************************************

----------*/

broaden1(Query,_,_,Values,New_query,New_selection,All_new_names,All_new_values):-
   broaden2(Query),
   write("\nSummary of results of Broadening search"),
   broaden3(1),
   broaden4(New_query,New_selection,New_names,New_values),
   remove_parameters(Values),
   retrieve_parameters(New_values),
   retractall(ip_rule),
   ip_values(New_query,All_new_names,All_new_values).
broaden1(Query,Selection,Names,Values,Query,Selection,Names,Values):-
   write("\nNo rules available to broaden search for this geometry.").

/*------

Broadening of the search proceeds by selectively deleting the least
significant keywords from the query and performing a keyword search

------------*/

broaden2(Query_keywords):-
   get_significance(Keyword,Context,Query_keywords,Significance),
   write("\nThe significance of "),write(Keyword),write(" is "),
   write(Significance),write("\n"),
   remove_keyword(Keyword,Context,Query_keywords,New_query),
   time_seconds(Time),
   search3(New_query,New_selection,Num,Time),
   required_vars(New_query,New_names,_),
   parameter_values_list(New_names,New_values),
   assert(selected_list(New_selection,Num,New_query,New_names,New_values)),
   tab(58),write("Press any key ..."),
   getkey(_),
   fail.

/*------

OR...

by generating a new query, and performing a keyword search.

The results of the search will be held in a structure
selected_list/5.
-------*/
broaden2(Old_query):-
   generate_query(Old_query,New_query,New_names,New_values),
   time_seconds(Time),
   search3(New_query,New_selection,Num,Time),
   assert(selected_list(New_selection,Num,New_query,New_names,New_values)),
   tab(58),write("Press any key ..."),
   getkey(_),
   fail.
```

```
/* succeeds only if search has been broadened */
broaden2(_):-
   selected_list(_,_,_,_,_),
   !.

broaden3(N):-
   retract( selected_list(New_selection,Num,New_query,New_names,New_values) ),
   assert( selected_list(N,Num,New_query,New_selection,New_names,New_values) ),
   write("\nSearch No. "),write(N),
   write(". Number of records selected "),write(Num),
   write("\nKeywords:\n"),
   write_line(76,76,New_query),
   N1 is N + 1,
   broaden3(N1).
broaden3(_).

broaden4(New_query,New_selection,New_names,New_values):-
   write("\nSelect a list of correlations for further investigation."),
   write("\nEnter the search number:"),
   read_integer(N),
   selected_list(N,Num,New_query,New_selection,New_names,New_values),
   retractall(selected_list/4).

/*-----------
get_significance/4
Choose the rules on the significance of the keywords to consider the least
significant changes first.

Significance   Approx. max. %ge error.
very_low       10
low         20
medium         33
high         50
-------*/
get_significance(Keyword,Context,Query_keywords,very_low):-
   significance(Keyword,Context,Query_keywords,very_low).
get_significance(Keyword,Context,Query_keywords,low):-
   significance(Keyword,Context,Query_keywords,low).
get_significance(Keyword,Context,Query_keywords,medium):-
   significance(Keyword,Context,Query_keywords,medium).
get_significance(Keyword,Context,Query_keywords,high):-
   significance(Keyword,Context,Query_keywords,high).

/*--------
remove_keyword(Keyword,Context,Old_query,New_query)
Remove keywords from the old query to give the new query.
The keyword to be removed is given by significance/4
Remove Keyword and all keywords in the hierarchy below Keyword provided
they are in the context given in the significance/4 rule.
---------*/
remove_keyword(K,Context,_,_):-
      label(N,K,_),
      sub_inherit(N,Child_node),
      label(Child_node,Child),
/*       path(Child,Path),
      all_members(Context,Path), */
      assert(remove_keyword(Child)),
```

```
	fail.
remove_keyword(K,_,Old_query,New_query):-
	fact_listing(remove_keyword/1),
	remove_keyword1(K,_,Old_query,New_query).
remove_keyword1(K,_,Old_query,New_query):-
	retract(remove_keyword(Child)),
	delete_all(Child,Old_query,Query),
	remove_keyword1(K,_,Query,New_query).
remove_keyword1(K,_,Old_query,New_query):-
	delete_all(K,Old_query,New_query),
	!.

/*-------
remove_keyword(Keyword,Old query,New Query)
removes keyword and all descendants of the keyword from the old query
NB. May cause problems if the Keyword occers more than once in the network.
This predicate will attempt to delete all the descendants of all occurances
of the keyword in the network, this may cause keywords to be removed
unintentionally from the query
---------*/
remove_keyword(Keyword,Old_query,New_query):-
	bagof(D,sub_inherit(Keyword,D),Descendants),
	delete_list_list(Descendants,Old_query,Q1),
	delete_member(Keyword,Q1,New_query).

/*------
remove_parameters/1 retract all the listed predicates
---------*/
remove_parameters([]):-
   !.
remove_parameters([V|Vs]):-
   retract(V:-!),
   !,
   remove_parameters(Vs).

/*------
retrieve_parameters/1
Restore the listed parameter values to the database.
--------*/
retrieve_parameters([]):-
   !.
retrieve_parameters([V|Vs]):-
   assert(V:-!),
   !,
   retrieve_parameters(Vs).

/*------
Generate a new query, the new query will select correlations which should
provide information of relevance to the original query.
Parameter values input for the original query are reused where possible in the
new query. The description of the heat transfer geometry and flow may not be
complete at this point.
-------*/

generate_query(Old_query,New_query,Names,Values1):-
	all_members(["forced_convection","jet_impingement","row"],Old_query),
	not( member("crossflow",Old_query)),
	remove_keyword("row",Old_query,Query),
```

```
        append(["array"],Query,New_query),
        x(X),
        ly(Ly),
        ny(Ny),
        Two_x is 2*X,
        intersect(Old_query,New_query,Common_keywords),
        required_vars(Common_keywords,Names,_),
        parameter_values_list(Names,Values),
        append([lx(Two_x),ly(Ly),n(Ny),nx(1),ny(Ny)],Values,Values1),
        !.

generate_query(Old_query,New_query,Names,Values1):-
        all_members(["forced_convection","jet_impingement","row","crossflow"],Old_query),
        remove_keyword("row",Old_query,Query),
        append(["array","initial_crossflow"],Query,New_query),
        x(X),
        ly(Ly),
        mc(Mi),
        ny(Ny),
        Two_x is 2*X,
        intersect(Old_query,New_query,Common_keywords),
        required_vars(Common_keywords,Names,_),
        parameter_values_list(Names,Values),
        append([lx(Two_x),ly(Ly),mi(Mi),n(Ny),nx(1),ny(Ny)],Values,Values1),
        !.


/*-----
parameter_values_list(Required_variables,Available_values)
predicate to generate list of the available parameters and values,
list in from [name(val),name1(val1),...].
------*/
parameter_values_list([Parameter|Ps],[X|Xs]):-
        string_atom(Parameter,A),
        X =.. [A,Val],
        call(X),
        !,
        parameter_values_list(Ps,Xs).
parameter_values_list([Parameter|Ps],Xs):-
        !,
        parameter_values_list(Ps,Xs).
parameter_values_list([],[]).

□
/*--------------

CHECK WHETHER EXPRESSION IS WITHIN LIMITS OR TOLERANCE
*****************************************************

--------------*/

check_limits(Key_in,L,L1):-
    stdout(Out),
    clear(Out),
    write("                CHECK LIMITS ON CORRELATION RANGE\n"),
    write("                --------------------------------"),
    write("\nLimits are given for the range over which each of the correlations"),
    write(" are valid."),
```

```
   menu("Options",vertical,7:3,w:blue:w,
   ["Accept the existing limits":assert(opt(a)),
    "Use the knowledge base to broaden the limits":assert(opt(b)),
    "Widen the limits by a specified tolerance":assert(opt(t)),
    "Ignore limits and extrapolate":assert(opt(e)),
    "Quit":abort]),
   retract(opt(Ans)),
   check_limits1(Ans,Key_in,L,L1),
   check_summary(Ans,L,L1).

/*-------

    APPLY THE LIMITS STRICTLY
    *************************

--------*/
check_limits1(a,_,[],[]).
check_limits1(a,_,[[Key,Key1]|T],[[Key,Key1]|T1]):-
   get_limits(Key,Key1,Limits_list),
   in_limits(Limits_list),
   check_limits1(a,_,T,T1).
check_limits1(a,_,[H|T],T1):-
   check_limits1(a,_,T,T1).

/*-------
Evaluate limit expression, succeed if value within limits
--------*/
in_limits([]).
in_limits( [[Funct,Lo,Hi]|T] ):-
      call_or_eval(Funct,Val),
      !,
      Lo =< Val,
      Hi >= Val,
      in_limits(T).

/*--------

      KNOWLEDGE BASED BROADENING OF THE DATA BASE SEARCH.
      ****************************************************

---------*/

/*---------
Use the knowledge base rules to widen the limits
------*/
check_limits1(b,_,Initial_selection,Modified_selection):-
   widen_limits(Initial_selection,Modified_selection).

/*--------
Test the limits of a set of correlations using rules on the allowable
variation of parameter values.

widen_limits(Keys for correlations,Keys for tested correlations)
The limits for a list of correlations are retrieved and
tested. If the limits are satisfied the keys for the correlation are
returned.
---------*/
widen_limits([],[]).
```

```
widen_limits([[Key,Key1]|L],[[Key,Key1]|L1]):-
    get_limits(Key,Key1,Limits),
    ref(Key,Authors,Year,Title,Source),
    write("\n"),write(Authors),write(" "),write(Year),
    write(", correlation "),write(Key1),write("."),
    descript(Key,Key1,Record_keywords),
    widen_limits1(Record_keywords,Limits,Result),
    write("\n"),tab(58),write("Press any key..."),
    getkey(_),
    Result == satisfied,
    !,
    widen_limits(L,L1).
widen_limits([[Key,Key1]|L],L1):-
    widen_limits(L,L1).

widen_limits1(_,[],satisfied):-
    !.
widen_limits1(_,[],not_satisfied):-
    !.
widen_limits1(Record_keywords,[[Expr,Min,Max]|Limits],Result):-
    call_or_eval(Expr,Val),
    Min =< Val,
    Max >= Val,
    write("\n"),write(Expr),write(" = "),write(Val),
    write(" in limits "),
    !,
    widen_limits1(Record_keywords,Limits,Result).
widen_limits1(Record_keywords,[[Expr,Min,Max]|Limits],Result):-
    get_significance(Expr,Low,High,Record_keywords,Significance),
    Low =< Max,
    High >= Min,
    !,
    write("\nRelaxed limits satisfied for "),write(Expr),
    write(". Significance "),write(Significance),
    widen_limits1(Record_keywords,Limits,Result).
widen_limits1(Record_keywords,[[Expr,Min,Max]|Limits],not_satisfied):-
    call_or_eval(Expr,Val),
    Min > Val,
    !,
    write("\n"),write(Expr),write(" = "),write(Val),
    write(" too low, lower limit "),write(Min),
    widen_limits1(Record_keywords,Limits,not_satisfied).
widen_limits1(Record_keywords,[[Expr,Min,Max]|Limits],not_satisfied):-
    call_or_eval(Expr,Val),
    Max < Val,
    !,
    write("\n"),write(Expr),write(" = "),write(Val),
    write(" too high, upper limit "),write(Max),
    widen_limits1(Record_keywords,Limits,not_satisfied).
```

```
/*------

Rules giving the range over which a parameter or expression can vary
without having an unacceptable effect on the calculated heat transfer rate.

get_significance(Expression,Low value,High value,Keywords,Significance)
significance(Expression,Low value,High value,Keywords,Significance)

Choose rule so that the limit relaxations which have the least significant
effect on the heat transfer rate are tried first.

Keywords used to ensure that a rule is relevant.
---------*/
get_significance(Expr,Min,Max,Record_keywords,very_low):-
   significance(Expr,Min,Max,Record_keywords,very_low).
get_significance(Expr,Min,Max,Record_keywords,low):-
   significance(Expr,Min,Max,Record_keywords,low).
get_significance(Expr,Min,Max,Record_keywords,medium):-
   significance(Expr,Min,Max,Record_keywords,medium).
get_significance(Expr,Min,Max,Record_keywords,high):-
   significance(Expr,Min,Max,Record_keywords,high).

/*----------

   APPLY A TOLERANCE TO THE CHECK
   ******************************

-----------*/

/*--------
Obtain percentage tolerances
---------*/
check_limits1(t,Keywords_in,Initial,Modified):-
   write("\nYou may specify a tolerances for any of the following parameters.\n"),
   required_vars(Keywords_in,Names,Texts),
   write_line(75,75,Names),
   write("\nEnter pairs of parameter names and percentage tolerances"),
   write("\neg m 10 z 5, enter RETURN to end: "),
   read_name_val(NameValPairs),
   check_tolerance(NameValPairs,Initial,Modified).

/*-------
Obtain the un-modified limits,
call predicate to apply the tolerances and check the limits.
--------*/
check_tolerance(_,[],[]).
check_tolerance(Tolerances,[[Key,Key1]|T],[[Key,Key1]|T1]):-
   get_limits(Key,Key1,Limits_list),
   ref(Key,Authors,Year,Title,Source),
   write("\n"),write(Authors),write(" "),write(Year),
   write(", correlation "),write(Key1),write("."),
   in_tolerance(Tolerances,Limits_list,Result),
   write("\n"),tab(58),write("Press any key..."),
   getkey(_),
   Result == satisfied,
   !,
   check_tolerance(Tolerances,T,T1).
```

```
check_tolerance(Tolerances,[H|T],T1):-
   check_tolerance(Tolerances,T,T1).

/*-----
check each limit in turn
--------*/
in_tolerance(Tolerances,[],satisfied):-
   !.
in_tolerance(Tolerances,[],not_satisfied):-
   !.
in_tolerance(Tolerances,[[Exp,Lo,Hi]|T],Result):-
   call_or_eval(Exp,Val),
   write("\n"),write(Exp),write(" = "),write(Val),
   string_compact_list(Exp,Exp1),
   in_tolerance1(Tolerances,Exp1,Lo,Hi,New_low,New_high),
   in_tolerance2(Val,New_low,New_high,Result),
   !,
   in_tolerance(Tolerances,T,Result).

/*---------
Apply all tolerances cumalatively.
----------*/
in_tolerance1([],Exp,Low,High,Low,High).
in_tolerance1([Name,Tol|Tols],Exp,Low,High,New_low,New_high):-
   member(Name,Exp),
   !,
   L is Low * (1 - Tol/100),
   H is High * (1 + Tol/100),
   in_tolerance1(Tols,Exp,L,H,New_low,New_high).
in_tolerance1([Name,Tol|Tols],Exp,Low,High,New_low,New_high):-
   not ( member(Name,Exp) ),
   !,
   in_tolerance1(Tols,Exp,Low,High,New_low,New_high).

in_tolerance2(Val,New_low,New_high,Result):-
   Val >= New_low,
   Val =< New_high,
   write(" in limits"),
   !.
in_tolerance2(Val,New_low,New_high,not_satisfied):-
   Val =< New_low,
   write(" too low, lower limit "),write(New_low),
   !.
in_tolerance2(Val,New_low,New_high,not_satisfied):-
   Val >= New_high,
   write(" too high, upper limit "),write(New_high),
   !.

/*-----

IGNORE LIMITS AND RETAIN ALL CORRELATIONS
*****************************************

----------*/
check_limits1(e,_,L,L).

check_limits1(_,K,L,L1):-
   check_limits(K,L,L1).
```

```
/*------
UTILITIES
---------*/

/*-----
Obtain limits from database for a publication
Test for variable within limits
-------*/
get_limits(Key,Key1,Limits):-
        get_limits1(Key,Key1),
        get_limits2(Limits),
        !.
get_limits1(Key,Key1):-
        limits(Key,Key1,Var,Lo,Hi),
        asserta( selected( [Var,Lo,Hi] ) ),
        fail.
get_limits1(_,_).
get_limits2([L|Ls]):-
        retract( selected(L) ),
        get_limits2(Ls).
get_limits2([]).

/*-------
Use previously derived value if available, else derive a value.
N.B. need to convert limit expressions from strings (the external database
form) to atoms (the form of functors for Prolog predicates).
--------*/

call_or_eval(S,Val):-
        string_atom(S,Funct),
        X =..[Funct,Val],
        call(X),
        !.
call_or_eval(S,Val):-
        string_compact_list(S,Tokens),
        exp_1(Tokens,Val),
        string_atom(S,Funct),
        X =..[Funct,Val],
        asserta(X:-!),
        !.

/*----------
check_summary(A,Original list,Checked list)

Give no output is A is set to 'e'
------------*/
check_summary(e,_,_):-
        !.
check_summary(_,L,L1):-
        length(L,N),
        length(L1,N1),
        D is N - N1,
        write("\n"),write(N1),write(" correletions accepted, "),write(D),
        write(" correlations rejected").
```
□

```
?-op(while,fx,600).
?-op(if,fx,600).

/*------------

                    EVALUATE THE CORRELATIONS
                    **************************

Evaluation of Nu from correlation using an interpreter

-----------*/


/*------------
Evaluate all the correlation expressions given for each
selected publication
-----------*/
eval_pub_list([]).
eval_pub_list( [[Key,Key1]|T] ):-
   eval_expr(Key,Key1),
   eval_pub_list(T).

/*---------
Evaluate all the correlation expressions for
one publication
---------*/
eval_expr(Key,Key1):-
   ref(Key,Authors,Year,Title,Source),
   writef("\n%-30s %t ",Authors,Year),
   collect_expressions(Key,Key1,Expr_list),
   !,
   make_tokens(Expr_list,Expr_list3),
   parse(Expr_list3,Expr_list4,_),
   !,
   assert(now_evaluating(Key,Key1)),
   eval_expr_list(Expr_list4,R),
   clear_results(R),
   retract(now_evaluating(Key,Key1)),
   write("\n"),tab(58),write("Press any key ..."),
   getkey(_),
   !.

/*---------
Obtain all correlation expressions from the database for one publication
Put expression strings in order in a flat list [str1,str2,...].
---------*/
collect_expressions(Key,Key1,Expressions):-
   collect_expressions1(Key,Key1),
   collect_expressions2(1,Expressions),
   !.

collect_expressions1(Key,Key1):-
   corr(Key,Key1,N,Expr),
   asserta( selected(N,Expr) ),
   fail.
```

```
collect_expressions1(_,_).

collect_expressions2(N,[E|Es]):-
   retract( selected(N,E) ),
   N1 is N + 1,
   collect_expressions2(N1,Es).
collect_expressions2(_,[]).

/*-----
Convert all expression strings into a single list of tokens,
use string_compact_list/2
--------*/
make_tokens([],[]).
make_tokens([E|Es],All_tokens):-
      string_compact_list(E,Tokens),
      !,
      make_tokens(Es,Ts),
      append(Tokens,Ts,All_tokens).

/*----------
Convert a list of tokens into a single string.
------------*/
de_tokenise([T],S):-
      string_atom(S,T),
      !.
de_tokenise([T|Ts],String):-
      string_atom(S,T),
      !,
      de_tokenise(Ts,Rest_string),
      concat(S,Rest_string,String).

/*---------
An INTERPRETER for arithmetic algorithms
----------*/

/*--------
Parser
- input a list of tokens (integers, variable names and operator symbols).
- returns a logical structure of the program for evaluation.
--------*/
parse(Tokens,Structures,_):-
      statements(Structures,Tokens,_).

/*------
Definite Clause Grammar for a programming language to represent the
correlation data.
-------*/

statements(Sl) -->
      statement(S),statements(Ss),
      {
      append([S],Ss,Sl)
      }.
statements([]) --> [].

statement(eval(X,V)) -->
      identifier(X),[=],expression(V).
statement(table(X,File,Y,Z)) -->
```

```
        identifier(X),identifier(File),expression(Y),expression(Z),[table].
statement(table(X,File,Y)) -->
        identifier(X),identifier(File),expression(Y),[table].
statement(graph(N,A,B,C)) -->
        identifier(N),expression(A),expression(B),expression(C),[graph].
statement(if(C,S1,S2)) -->
        [if],condition(C),[then],statements(S1),[else],statements(S2),[end_if].
statement(if(C,S1)) -->
        [if],condition(C),[then],statements(S1),[end_if].
statement(while(C,S)) -->
        [while],condition(C),statements(S),[end_while].
statement(write(X)) -->
        [write],identifier(X).
statement(write_string(X)) -->
        [write],text_string(X).
statement(exp_error(X)) -->
        [error],text_string(X).

expression(Expr) -->
        constant(X),arithmetic_op(Op),expression(Y),
        { append(X,Op,E) },
        { append(E,Y,Expr) } .
expression(Expr) -->
        constant(X),
        {
        append(X,[],Expr)
        } .

arithmetic_op([*,*,-]) --> [*],[*],[-].
arithmetic_op([*,*]) --> [*],[*].
arithmetic_op([+]) --> [+].
arithmetic_op([-]) --> [-].
arithmetic_op([*]) --> [*].
arithmetic_op([/]) --> [/].

constant(Expr) -->
        ['('],expression(Y),[')'],
        { append(['('],Y,E) },
        { append(E,[')'],Expr)} .
constant([X]) -->
        identifier(X).
constant([X]) -->
        [X],
        {
        number(X)
        } .

identifier(X) -->
        [X],
        {
        atom(X)
        } .

text_string(X) -->
        [X],
        {
        string(X)
        } .
```

```
condition(compare(Op,X,Y)) -->
        expression(X),
        comparison_op(Op),
        expression(Y).

comparison_op(=:=) --> [=],[:],[=].
comparison_op(=\=) --> [=],[\],[=].
comparison_op(>=) --> [>],[=].
comparison_op(=<) --> [=],[<].
comparison_op(>) --> [>].
comparison_op(<) --> [<].

/*---------------
Process list of structures obtained from parsing to give numerical values
for the correlation.
----------*/

eval_expr_list([],[]).
eval_expr_list([eval(Name,Expr)|Rem_expr],[X|Results]):-
        exp_1(Expr,Ans),
        X =.. [Name,Ans],
        asserta(X:-!),
        !,
        eval_expr_list(Rem_expr,Results).

eval_expr_list([while(C,Es)|Rem_expr],R3):-
        eval_while_loop(C,Es,R1),
        !,
        eval_expr_list(Rem_expr,R2),
        append(R2,R1,R3).
eval_expr_list([if(compare(Op,Expr1,Expr2),E1s,E2s)|Rem_expr],R3):-
        exp_1(Expr1,A1),
        exp_1(Expr2,A2),
        Condition =.. [Op,A1,A2],
        eval_if_blocks(Condition,E1s,E2s,R1),
        !,
        eval_expr_list(Rem_expr,R2),
        append(R2,R1,R3).
eval_expr_list([if(compare(Op,Expr1,Expr2),E1s)|Rem_expr],R3):-
        exp_1(Expr1,A1),
        exp_1(Expr2,A2),
        Condition =.. [Op,A1,A2],
        eval_if_blocks(Condition,E1s,R1),
        !,
        eval_expr_list(Rem_expr,R2),
        append(R2,R1,R3).
/*------
Read a table
-------*/
eval_expr_list([table(Funct,File,X,Y)|Rem_expr],[Result|Results]):-
    exp_1(X,Xval),
    exp_1(Y,Yval),
    string_atom(File_str,File),
    string_list(File_str,L1),
    string_list("\\jetdat\\",L2),
    append(L2,L1,L3),
    home_dir(Home),
```

```
   string_list(Home,L4),
   append(L4,L3,L5),
   string_list(File_str1,L5),
   read_table(File_str1,Xval,Yval,Ans),
   writef("\n%t = %t",Funct,Ans),
   Result =.. [Funct,Ans],
   asserta(Result:-!),
   !,
   eval_expr_list(Rem_expr,Results).
/*--------
Read a table of form X's'CRLF'Y's'CRLF' and interpolate a value for Y
at the given X.
----------*/
eval_expr_list([table(Funct,File,X)|Rem_expr],[Result|Results]):-
   exp_1(X,Xval),
   string_atom(File_str,File),
   string_list(File_str,L1),
   string_list("\\jetdat\\",L2),
   append(L2,L1,L3),
   home_dir(Home),
   string_list(Home,L4),
   append(L4,L3,L5),
   string_list(File_str1,L5),
   read_graph(File_str1,1,Xval,Ans),
   writef("\n%t = %t",Funct,Ans),
   Result =.. [Funct,Ans],
   asserta(Result:-!),
   !,
   eval_expr_list(Rem_expr,Results).

/*-----------
Evaluation of heat transfer from graphs, single jet
------*/
eval_expr_list([graph(Name,X,Re,Z)|Rem_expr],Results):-
   now_evaluating(Key,Key1),
   !,
   srj_corr(Key,Key1,Name,X,Re,Z),
   !,
   eval_expr_list(Rem_expr,Results).

eval_expr_list([write(X)|Rem_expr],Results):-
      Y =.. [X,Z],
      call(Y),
      write("\n"),write(X),write(" = "),write(Z),
      !,
      eval_expr_list(Rem_expr,Results).
eval_expr_list([write_string(X)|Rem_expr],Results):-
      write("\n"),
      write(X),
      !,
      eval_expr_list(Rem_expr,Results).
eval_expr_list([exp_error(X)|Rem_expr],Results):-
      write("\nERROR!\n"),
      write(X),
      press_any_key,
      !,
      eval_expr_list(Rem_expr,Results).
```

```
/*---------
If the evaluation of a correlation fails display message and
proceed to next correlation
-------*/
eval_expr_list(A,[]):-
   !,
   write("\nFailure in evaluating correlation.\n").


eval_while_loop(compare(Op,Expr1,Expr2),Es,R):-
       exp_1(Expr1,A1),
       exp_1(Expr2,A2),
       Condition =.. [Op,A1,A2],
       eval_while_loop1(Condition,compare(Op,Expr1,Expr2),Es,R).

eval_while_loop1(Condition,C,Es,R3):-
       call(Condition),
       !,
       eval_expr_list(Es,R1),
       eval_while_loop(C,Es,R2),
       append(R2,R1,R3).
eval_while_loop1(_,_,_,[]):-
       !.

eval_if_blocks(Condition,E1s,E2s,R):-
       call(Condition),
       !,
       eval_expr_list(E1s,R).
eval_if_blocks(Condition,E1s,E2s,R):-
       !,
       eval_expr_list(E2s,R).
eval_if_blocks(Condition,E1s,R):-
       call(Condition),
       !,
       eval_expr_list(E1s,R).
eval_if_blocks(Condition,E1s,[]):-
       !.

/*----------
Clear database of all facts derived in evaluating this correlation.
The facts asserted are recorded in the list.
--------*/
clear_results([]).
clear_results([X|Rs]):-
   retract( :-(X,!) ),
   clear_results(Rs).

/*---
Evaluate correlations for single round impinging jet
----*/
srj_corr(Key,Key1,Name,X,Re,Z):-
   srj_corr_two(Key,Key1,Name,X,Re,Z),
   srj_corr_three(Key,Key1,Name,X,Re,Z).

srj_corr_two(Key,Key1,Name,X,Re,Z):-
   exp_1(X,X1),
   exp_1(Re,Re1),
   exp_1(Z,Z1),
```

```
    write('\nEvaluating local flux using Re extrapolation...\n'),
    two_interpolation_curves(Key,Key1,Name,X1,Re1,Z1,L),
    srj_corr_eval(L,X1,Re1,Z1),
    press_any_key.

srj_corr_three(Key,Key1,Name,X,Re,Z):-
    exp_1(X,X1),
    exp_1(Re,Re1),
    exp_1(Z,Z1),
    write('\nEvaluating flux by interpolation between three curves...\n'),
    three_interpolation_curves(Key,Key1,Name,X1,Re1,Z1,L),
    srj_corr_eval(L,X1,Re1,Z1),
    press_any_key.

srj_corr_eval(L,X1,Re1,Z1):-
    L \== [],
    t1(T1),
    t2(T2),
    tw(Tp),
    ta(Tamb),
    d(D),
    home_dir(Home),
    concat(Home,"\\jetdat\\butl84a.dat",Fl1),
    concat(Home,"\\jetdat\\butl84.dat",Fl2),
    local_flux_srj(L,X1,Fl1,1,Fl2,1
       ,Re1,Z1,T1,T2,Tp,Tamb,D,Nu,Htc,Flux),
    write("\n Area mean values derived from curves of local heat transfer"),
    write(" values."),
    write("\n Nu = "),write(Nu),
    write("\n htc = "),write(Htc),
    write("\n Flux = "),write(Flux),write("\n").
srj_corr_eval([],X1,Re1,Z1):-
       write("\nSuitable curves not found\n").

□
/*-------
Predicates to parse and evaluate an expression held in a string
by Recursion-descent parsing.
--*/

module eval_expr.
import string_float/2,read_float/2.
export calculator/0,eval_expr1/2,string_compact_list/2,exp_1/2.
export get_number/3,get_variable/3.

eval_expr1(Expr_stg,Ans):-
       string_compact_list(Expr_stg,Expr),
       !,
       write("\n"),
       exp_1(Expr,Ans),
       write(Expr_stg),
       write(" = "),
       write(Ans).

/*-----
Convert expression string to a list of numbers, variable names
and symbols, ie produce list of tokens. Discard other characters.
-----*/
```

```
string_compact_list(Expr_stg,Expr):-
	string_list(Expr_stg,Ascii_list),
	compact_list(Ascii_list,Expr),
	!.

compact_list([],[]).
compact_list(List,[H|T]):-
	get_number(List,Short_list,H),
	!,
	compact_list(Short_list,T).
compact_list(List,[H|T]):-
	get_variable(List,Short_list,H),
	!,
	compact_list(Short_list,T).
compact_list(List,[H|T]):-
	get_string(List,Short_list,H),
	!,
	compact_list(Short_list,T).
compact_list([H|T],[Symbol|T1]):-
	member(H,[40,41,42,43,45,47,58,60,61,62,92]),
	!,
	name(Symbol,[H]),
	compact_list(T,T1).
compact_list([H|T],T1):-
	!,
	compact_list(T,T1).

/*-----
Get number from front of string
-------*/
get_number(List,Short_list,Number):-
	get_number1(List,Short_list,Num_list),
	not(length(Num_list,0)),
	string_list(Num_string,Num_list),
	string_float(Num_string,Number).

get_number1([H|T],Short_list,[H|Num_string]):-
	H =< 57,
	H >= 48,
	get_number1(T,Short_list,Num_string).
get_number1([H|T],Short_list,[H|Num_string]):-
	H == 46,
	get_number1(T,Short_list,Num_string).
get_number1(L,L,[]).

/*-------
Get variable from front of string.
The variable must start with a letter, subsequent characters may be letters,
numbers, points or under-scores.
-----------*/
get_variable([Letter|List],Short_list,Variable):-
	lexis(Letter,letter),
	get_variable1([Letter|List],Short_list,[H|T]),
	lower_case(H,H1),
	name(Variable,[H1|T]).

get_variable1([H|T],Short_list,[H|Var_list]):-
```

```
        lexis(H,letter),
        get_variable1(T,Short_list,Var_list).
get_variable1([H|T],Short_list,[H|Var_list]):-
        H > 47,
        H < 58,
        get_variable1(T,Short_list,Var_list).
get_variable1([95|T],Short_list,[95|Var_list]):-
        get_variable1(T,Short_list,Var_list).
get_variable1([46|T],Short_list,[46|Var_list]):-
        get_variable1(T,Short_list,Var_list).
get_variable1(L,L,[]).

/* Force first letter of variable name to be lower case */
lower_case(H,H1):-
        H < 91,
        H > 64,
        H1 is H + 32.
lower_case(H,H).

/*-------
Get a string from the front of the list,
The string is any text enclosed by double quotes, "any text".
------*/
get_string([34|T],Short_list,String):-
        get_string1(T,Short_list,Chars),
        string_list(String,Chars).
get_string1([H|T],Short_list,[H|Chars]):-
        H \== 34,
        !,
        get_string1(T,Short_list,Chars).
get_string1([34|Short_list],Short_list,[]).

/*--------
Begin evaluation of expression
---------*/
exp_1([Ans],Ans):-
        number(Ans).
exp_1(Expr,Ans):-
        exp_2(Expr,Rest_expr,Ans1),
        exp_1([Ans1|Rest_expr],Ans).

/*---------
Add or Subtract two terms.
-----*/
exp_2(Expr,Rest_expr,Ans):-
        exp_3(Expr,Rest_expr1,Ans1),
        exp_2a(Rest_expr1,Ans1,Rest_expr,Ans).
exp_2a([+|Expr],Ans1,Rest_expr,Ans):-
        exp_3(Expr,Rest_expr,Ans2),
        Ans is Ans1+Ans2.
exp_2a([-|Expr],Ans1,Rest_expr,Ans):-
        exp_3(Expr,Rest_expr,Ans2),
        Ans is Ans1-Ans2.
exp_2a(Rest_expr,Ans,Rest_expr,Ans).

/*------
Multiply or devide two factors
------*/
```

```
exp_3(Expr,Rest_expr,Ans):-
      exp_4(Expr,Rest_expr1,Ans1),
      exp_3a(Rest_expr1,Ans1,Rest_expr,Ans).
exp_3a([*|Expr],Ans1,Rest_expr1,Ans):-
      exp_4(Expr,Rest_expr,Ans2),
      Ans3 is Ans1*Ans2,
      exp_3a(Rest_expr,Ans3,Rest_expr1,Ans).
exp_3a([/|Expr],Ans1,Rest_expr1,Ans):-
      exp_4(Expr,Rest_expr,Ans2),
      Ans3 is Ans1/Ans2,
      exp_3a(Rest_expr,Ans3,Rest_expr1,Ans).
exp_3a(Rest_expr,Ans,Rest_expr,Ans).

/*---------
Raise to a power
---------*/

exp_4(Expr,Rest_expr,Ans):-
      exp_5(Expr,Rest_expr1,Ans1),
      exp_4a(Rest_expr1,Ans1,Rest_expr,Ans).
exp_4a([*,*|Expr],Ans1,Rest_expr,Ans):-
      exp_5(Expr,Rest_expr,Ans2),
      p_pow(Ans1,Ans2,Ans).
exp_4a(Rest_expr,Ans,Rest_expr,Ans).

/*------
Unary positive or negative
--------*/

exp_5([+|Expr],Rest_expr,Ans):-
      exp_6(Expr,Rest_expr,Ans).
exp_5([-|Expr],Rest_expr,Ans):-
      exp_6(Expr,Rest_expr,Ans1),
      Ans is -1*Ans1.
exp_5(Expr,Rest_expr,Ans):-
      exp_6(Expr,Rest_expr,Ans).

/*------
Either have a number or a variable or an expression in parentheses
----*/

exp_6(['('|Expr],Rest_expr,Ans):-
      exp_7(Expr,Rest_expr,Ans).
exp_6([Number|Short_list],Short_list,Number):-
      number(Number).
/*-----
exp_6([Variable|Rest_expr],Rest_expr,Ans):- If any new values can
      call(fc_rule),                        be derived do so
      fail.
-------*/
exp_6([Variable|Rest_expr],Rest_expr,Ans):-  /*If a value has already*/
      name(Variable,[H|T]),                  /*been calculated use it*/
      lexis(H,letter),
      Var_expr =..[Variable,Ans],
      call(Var_expr).
exp_6([Variable|Rest_expr],Rest_expr,Ans):-  /*Ask for the value      */
      name(Variable,[H|T]),
```

```
    lexis(H,letter),
    read_float(Variable,Ans),
    Var_expr =..[Variable,Ans],
    asserta(Var_expr:-!).

/*--------
Evaluate expression in parentheses
--------*/
exp_7([Number,')'|Rest_expr],Rest_expr,Number):-
    number(Number).
exp_7(Expr,Rest_expr,Ans):-
    exp_2(Expr,Rest_expr1,Ans1),
    exp_7([Ans1|Rest_expr1],Rest_expr,Ans).

end_of_module.
```
□

```
y,Key1]|L1]):-
   get_limits(Key,Key1,Limits),
   ref(Key,Authors,Year,Title,Source),
   write("\n"),write(Authors)ance(Expr,Min,M
/*--------
get_keys/2 returns a list of valid keywords.
---------*/
add_key_prompt:-
   retractall(current_keywords/1),
   write("Enter either existing keywords or new keywords\n"),
   write("Additional options:\n"),
   write("    'draw_tree' to display the existing hierarchy\n"),
   write("    'expand' to obtain more information about the keywords\n"),
   write("    RETURN to end the selection of keywords in a branch\n"),
   subset(R,R),
   get_keys([R],Ns),
   node_label_list(Ns,Ks),
   write("\nKeywords selected:\n"),write(Ks),write("\n"),
   assert(current_keywords(Ks)),
   !,
   write("\nAdditions to keyword database."),
   write("\n------------------------------\n"),
   fact_listing(subset_new/2),
   fact_listing(exclude_new/2),
   fact_listing(label_new/3),
   press_any_key.

/*-----
Given a keyword node list, find the keyword nodes in the next level in the
hierarchy for the first keyword node. Obtain a selection of these. Using this
selection make a recursive call to the predicate. Then make recursive
calls for the remaining predicates in the first keyword node list.
-----------*/

get_keys([],[]).
get_keys([P|Ps],Ns):-
   get_keys1(P,N1s),
   get_keys(N1s,N2s),
   append(N1s,N2s,N3s),
   get_keys(Ps,N4s),
   append(N3s,N4s,Ns).

/*-------
get_keys1/2, obtains one or more keyword nodes in the hierarcy below Parent
If there are no keyword nodes below parent a prompt is made for new keywords.
-------*/
get_keys1(Parent,Ns):-
   show_child_keys(Parent,Child_nodes),
   get_keys1a(Ns,Parent,Child_nodes).

get_keys1a(Ns,Parent,Child_nodes):-
   repeat,
   write("\nEnter keyword(s), option or RETURN: "),
   read_list(Ks),
   get_keys2(Ks,Parent,Child_nodes,Ns).
```

```
get_keys2(["draw_tree"],Parent,_,_):-
      draw_tree(Parent),
      fail.
get_keys2(["expand"],Parent,Child_nodes,_):-
      keyword_info(Parent,Child_nodes),
      fail.
get_keys2(Ks,Parent,Child_nodes,Selected_nodes):-
      not( member("draw_tree",Ks) ),
      not( member("expand",Ks) ),
      flatten_l(Child_nodes,Child_nodes1),
      node_label_list(Child_nodes1,Child_keys),
      compare_lists(Ks,Child_keys,New_keys,Unused_keys,Old_keys),
      get_keys3(Child_nodes1,Old_keys,Old_nodes),
      not( excluded1(Old_nodes,Old_nodes)),
      add_keywords(Parent,New_keys,New_nodes),
      append(Old_nodes,New_nodes,Selected_nodes),
      add_excludes(Child_nodes,Old_nodes,New_nodes).
get_keys2(_,_,_,_):-
      fail.

/*----
get_keys3 return the node numbers for the keywords entered by the user and
which are already known to the system
------*/
get_keys3(Child_nodes,[Old_key|Oks],[Old_node|Ons]):-
      label(Old_node,Old_key,_),
      member(Old_node,Child_nodes),
      !,
      get_keys3(Child_nodes,Oks,Ons).
get_keys3(_,[],[]).

/*------
For each new keyword: assign a node number, obtain text and
add the new keywords to a tempory database.
-------*/
add_keywords(_,[],[]):-
      !.
add_keywords(Parent,New_keywords,New_nodes):-
      next_available_node(Av_node),
      add_keywords1(Parent,Av_node,New_keywords,New_nodes).

add_keywords1(_,_,[],[]).
add_keywords1(Parent,N,[K|Ks],[N|Ns]):-
      write("Enter text to explain the keyword '"),
      write(K),write("': "),
      readln(254,"",Text), /*max 254 chars*/
      assert(label_new(N,K,Text) ),
      assert( subset_new(Parent,N) ),
      N1 is N + 1,
      add_keywords1(Parent,N1,Ks,Ns).

/*-----
Find the next unused keyword node number
----*/
next_available_node(Av_node):-
      bagof(N,label(N,_,_),Ns),
      max(Ns,Node),
```

```
    Av_node is Node + 1.

/*-------
Keywords are organised into groups if incompatible keywords.
Show which keywords can be made incompatible with each new keyword,
offer opportunity to make these keywords incompatible
---------*/
add_excludes(_,_,[]):-
    !.
add_excludes([],_,_):-
    !.
add_excludes(Child_nodes,Old_nodes,[New_node|Ns]):-
    add_excludes1(Child_nodes,Old_nodes,Groups),
    add_excludes2(Child_nodes,New_node,Groups),
    add_excludes(Child_nodes,[N|Old_nodes],Ns).

/*---
A new keyword can be added to a group if none of the keywords entered
along with the new keyword are in the group. Return all the groups
which can be used
----*/
add_excludes1([],_,[]):-
    !.
add_excludes1([C|Cs],Old_nodes,[C|Gs]):-
    no_members(C,Old_nodes),
    !,
    add_excludes1(Cs,Old_nodes,Gs).
add_excludes1([C|Cs],Old_nodes,Gs):-
    add_excludes1(Cs,Old_nodes,Gs).

/*---
prompt for a keyword from one of the groups
-----*/
add_excludes2(Child_nodes,New_node,Groups):-
    label(New_node,New_key,_),
    !,
    write("\nYou may prevent any one of the following groups of keywords"),
    write(" being used in\ncombination with '"),write(New_key),
    write("' by entering one keyword from the group"),
    write_child_list(Groups),
    write("\nEnter keyword or RETURN to exclude no groups:"),
    readln("",Excluded_key),
    add_excludes3(Child_nodes,New_node,Groups,Excluded_key).

/*---
Check that the keyword, if any, is a member of one of the valid groups.
Assert new exclude relation to the Prolog database or redisplay prompt
------*/
add_excludes3(_,_,_,""):-
    !.
add_excludes3(Child_nodes,New_node,Groups,Excluded_key):-
    label(Excluded_node,Excluded_key,_),
    flatten_l(Child_nodes,Child_nodes1),
    member(Excluded_node,Child_nodes1),
    flatten_l(Groups,Groups1),
    member(Excluded_node,Groups1),
    !,
    assert(exclude_new(New_node,Excluded_node)).
```

```
add_excludes3(Child_nodes,New_node,Groups,_):-
      write("Invalid keyword, enter a keyword or RETURN :"),
      readln("",Excluded_key),
      !,
      add_excludes3(Child_nodes,New_node,Groups,Excluded_key).

/*-------
Show options for next keyword, A is the current keyword node
B is a list giving the nodes in the next level in the hierarchy
------*/
show_child_keys(A,B):-
      all_child_keys(A,B),
      B \== [],
      !,
      label(A,Parent,_),
      write("\nYou may select a keyword from each of the following "),
      write("groups to give a more\nspecific"),
      write(" description of '"),write(Parent),write("'"),
      write_child_list(B).
show_child_keys(A,[]):-
      label(A,Parent,_),
      write("\nNo keywords are currently available to give a more specific"),
      write("\ndescription of '"),
      write(Parent),write("'").

/*---------
all_child_keys/2 returns a list of the keywords at the next level in the
hierarchy. This list has the form [[Keyword nodes],[Keyword nodes 1]...]
where only one keyword node from each group can be selected
-------*/
all_child_keys(Parent,Cs):-
      bagof(B,sub_not_root(Parent,B),Bs),
      all_child_keys1(Bs,Cs).

all_child_keys1([],[]).
all_child_keys1([A|As],[Cs|Ds]):-
      bagof(B,exclude_implys(A,B),Bs),
      intersect([A|As],[A|Bs],Cs),
      delete_members(Bs,As,A1s),
      !,
      all_child_keys1(A1s,Ds).

/*------
write_child_list/1, prints a compound list eg [[a],[b,c]], starting a new
line for each new sub-list encountered.
---*/
write_child_list(A):-
      write_child_list(A,1).

write_child_list([],_).
write_child_list([A|B],N):-
      list(A),
      write("\n"),write(N),write(")   "),
      write_child_list1(A),
      N1 is N + 1,
      write_child_list(B,N1).
write_child_list1([]).
write_child_list1([A|B]):-
```

```
        atomic(A),
        label(A,Keyword,_),
        write(Keyword),write(" "),
        write_child_list1(B).

if_then_else(Condition,Then,Else):-
        call(Condition),
        !,
        call(Then).
if_then_else(Condition,Then,Else):-
        call(Else).

/*----------
Display more information on a keyword.
-------*/
keyword_info(Parent,Children):-
        stdout(Out),
        open(key_info,"key_info.txt",write),
        stdout(key_info),
        tab(60),write("Press RETURN to continue"),
        write("\n\nKeywords available to further specify "),
        label(Parent,Parent_key,_),
        write(Parent_key),
        write("\n\nYou may select one keyword from each of the following groups:"),
        keyword_info1(1,Children),
        open(key_info1,3:3,21:76,b:cyan:w,'Keyword information'),
        stdout(key_info1),
        close(key_info),
        view("key_info.txt",13),
        stdout(Out),
        close(key_info1).

keyword_info1(N,[A|As]):-
        write("\nGroup "),write(N),write(":\n"),
        node_label_list(A,Ks,Ts),
        show_expansions(Ks,Ts),
        N1 is N + 1,
        !,
        keyword_info1(N1,As).
keyword_info1(_,[]).

/*------
Write the text explaining the meanings of the keywords
-------*/
show_expansions(_,[]).
show_expansions([K|Ks],[Text|Ts]):-
    write(K),write(":\n"),
    write_line(74,Text),
    write("\n"),
    show_expansions(Ks,Ts).

/*--------
check_keys/1 succeeds if a list of keywords is a valid combination.
The check will test all combinations of paths up the tree from each
keyword and will succeed provided one of these combinations is valid.
----------*/
check_keys([]).
check_keys([K|Ks]):-
```

```
        not( excluded(K,Ks) ),
        path(K,Path),
        delete_list_list(Path,Ks,K_rem),
        check_keys1(Path,K_rem),
        check_keys(K_rem).

check_keys1(Path,[]).
check_keys1(Path1,[K|Ks]):-
        path(K,Path2),
        delete_list_list(Path1,Path2,Path2a),
        delete_list_list(Path2,Path1,Path1a),
        not( excluded1(Path1a,Path2a) ),
        check_keys1(Path1,Ks).

□
/*-----

    OBTAIN AND ADD A NEW INPUT PARAMETER NAME.
    ************************************************

---------*/
add_variable:-
        current_keywords(Keywords),
        write("\nEnter a new variable name or RETURN to end:"),
        readln(12,"",Name), /*max 12 chars*/
        add_variable1(Name,Name1,Keywords,Text),
        write("Do you want the user to be prompted for its value? y/n:"),
        readln("y",Y_or_n),
        some_keywords(Keywords,Keywords,Keywords1),
        concat_list_string(Keywords1,Keywords2),
        assert( diction_new(Name1,Text,Keywords2,Y_or_n) ),
        add_variable.
add_variable:-
        write("\n    Additions to variable dictionary."),
        write("\n    ---------------------------------\n"),
        fact_listing(diction_new/4),
        press_any_key.

add_variable1("",_,_,_):-
        !,
        fail.
add_variable1(Name,Name,Query_keywords,Text):-
        not( add_variable2(Name,Query_keywords) ),
        !,
        write("Enter text to define the variable (max. 80 chars)\n"),
        readln(80,"",Text). /*max 80 chars*/
add_variable1(Name,Name2,Query_keywords,Text):-
        write("\nEnter a new variable name or RETURN to end:"),
        readln(12,"",Name1), /*max 12 chars*/
        add_variable1(Name1,Name2,Query_keywords,Text).

/*------
Succeeds if the variable name has already been used for a geometry of
the description given or for a more general version of the geometry
described.
----------*/
add_variable2(Name,_):-
    string_list(Name,[A|As]),
```

```
   not( lexis(A,letter) ),
   write("\nInvalid variable name - Initial character must be a letter").
add_variable2(Name,Query_keywords):-
   diction(Name,_,Keywords,_),
   split_list_string(Keywords1,Keywords),
   all_members(Keywords1,Query_keywords),
   write("\n'"),write(Name),write("' is already used in this context").

/*----
Integrity constraint called before new parameter record if saved to
database files. Succeed if variable can be prompted for or if eqn
to derive parameter exists (Out is stream for messages).
-----*/
add_variable3(Out,Name,Text,Query_keywords,Y_or_n):-
     Y_or_n == "y",
     !.
add_variable3(Out,Name,Text,Query_keywords,Y_or_n):-
     eqn(Name,Keywords,Eqn_no,1,Eqn),
     split_list_string(Keywords1,Keywords),
     split_list_string(Query_keywords1,Query_keywords),
     similar_list(Query_keywords1,Keywords1),
     !.
add_variable3(Out,Name,Text,Query_keywords,Y_or_n):-
     stdout(File),
     stdout(Out),
     write("\nEquation to derive '"),write(Name),write("' required."),
     write("\nRecord for '"),write(Name),write("' not written.\n"),
     stdout(File),
     fail.

/*----------

     OBTAIN PARAMETER EQUATION AND ADD TO PROLOG DATABASE
     *************************************************************

------*/
/*-------
Obtain a set of equations defining the relationship between input variables,
perform checks and add the equations to the database
---------*/
add_eqn:-
   current_keywords(Query_keywords),
   write("Add equations/procedures giving inter-relations between variables"),
   write("\n"),
   some_keywords(Query_keywords,Query_keywords,Query_keywords1),
   sys_vars(Sys_vars),
   required_vars(Query_keywords1,Req_vars,_),
   write("Enter name of variable to be derived:"),
   readln(12,"",Result), /*max 12 chars*/
   !,
   add_eqn2a(Query_keywords1,Sys_vars,Req_vars,Result).
```

```
/*---
Check that the result variable exists -
if blank then end
if it does then continue
if it dosn't then reprompt
----*/
add_eqn2a(_,_,_,""):-
   !.
add_eqn2a(Keywords,Sys_vars,Req_vars,Result):-
   append(Sys_vars,Req_vars,Vars),
   member(Result,Vars),
   !,
   add_eqn1(Keywords,Sys_vars,Req_vars,Result,[],0).
add_eqn2a(Keywords,Sys_vars,Req_vars,Result):-
   write("\nResult variable "),write(Result),write(" not known."),
   write("\nThis variable must be defined before this rule can be accepted\n"),
   press_any_key,
   write("Enter name of variable to be derived:"),
   readln(12,"",New_result), /*max 12 chars*/
   !,
   add_eqn2a(Keywords,Sys_vars,Req_vars,New_result).

/*-----
Obtain new statement
------*/
add_eqn1(Keywords,Sys_vars,Req_vars,Result,Deriv_vars,N):-
   write("Enter a new statement or RETURN to end:"),
   readln(80,"",Statement), /*max 80 chars*/
   add_eqn2(Keywords,Sys_vars,Req_vars,Result,Deriv_vars,N,Statement).

/*-----
Either
(i)  add statements to the database
(ii) parse statement and check that the variables in the statement exist
(iii) re-prompt if statement not validated
------*/
add_eqn2(Keywords,_,_,Result,_,_,""):-
   add_eqn4(Keywords,Result),
   !.
add_eqn2(Keywords,Sys_vars,Req_vars,Result,Deriv_vars,N,Statement):-
   string_compact_list(Statement,Tokens),
   vars_in_statement(Variables,Tokens),
   list_string_atom(Str_variables,Variables),
   list_head_tail(Str_variables,Deriv_new,Ip_vars),
   delete_all1(Sys_vars,Ip_vars,Ip_vars1),
   delete_all1(Req_vars,Ip_vars1,Ip_vars2),
   delete_all1(Deriv_vars,Ip_vars2,Ip_vars3),
   add_eqn3(Keywords,Sys_vars,Req_vars,Result,[Deriv_new|Deriv_vars],
                                   N,Statement,Ip_vars3),
   !.
add_eqn2(Keywords,Sys_vars,Req_vars,Result,Deriv_vars,N,Statement):-
   write("\nFailure to validate statement, re-enter\n"),
   add_eqn1(Keywords,Sys_vars,Req_vars,Result,Deriv_vars,N).
```

```
/*----
If the statement is OK make tempory note else print message and end
----*/
add_eqn3(Keywords,Sys_vars,Req_vars,Result,Deriv_vars,N,Statement,[]):-
   Line_no is N + 1,
   assert(eqn_new(Line_no,Statement)),
   add_eqn1(Keywords,Sys_vars,Req_vars,Result,Deriv_vars,Line_no).
add_eqn3(_,_,_,_,_,_,_,L):-
   write("\nVariables "),write(L),write(" not known. "),
   write("\nThese variables must be defined before the rule can be accepted"),
   write("\n"),
   retractall(eqn_new/2),
   write("\n"),tab(56),write("Press any key ..."),
   getkey(_).

/*-----
Find a unique combination of attributes to identify reference the
relationship - use Result parameter name, Query keywords and Equation
number
-------*/
add_eqn4(Query_keywords,R):-
   current_eqns(R,Query_keywords,Ns),
   max(Ns,Max),
   Eqn_no is Max + 1,
   concat_list_string(Query_keywords,Query_keywords1),
   write("\n       Added inter-relations."),
   write("\n       ---------------------\n"),
   add_eqn5(R,Query_keywords1,Eqn_no).

/*----
Assemble and assert the new database records
------*/
add_eqn5(R,Query_keywords,Eqn_no):-
   retract(eqn_new(Line_no,Statement)),
   assert(eqn_new(R,Query_keywords,Eqn_no,Line_no,Statement)),
   write(eqn_new(R,Query_keywords,Eqn_no,Line_no,Statement)),
   write("\n"),
   add_eqn5(R,Query_keywords,Eqn_no).
add_eqn5(R,_,_):-
   write("\nEquations for "),write(R),
   write(" have been added to a tempory database"),
   press_any_key.

/*-------

   Obtain recommended NON-DIMENSIONAL EXPRESSIONS and add to Prolog database
   *******************************************************************

------*/
add_non_dim:-
   current_keywords(Query_keywords),
   write("Add recommended non-dimensional groups for use "),
   write("in limits of validity\n"),
   add_non_dim1(Query_keywords),
   write("\n     Additions to non_dimensional expressions."),
   write("\n     ----------------------------------------\n"),
   fact_listing(non_dim_new/3),
```

```
   press_any_key.

add_non_dim1(Query_keywords):-
   write("Enter non-dimensional expression or RETURN to end:"),
   readln(40,"",Expr), /*max 40 chars*/
   add_non_dim2(Expr,Query_keywords).

add_non_dim2("",_):-
   !.
add_non_dim2(Expr,Query_keywords):-
   write("Enter text to define the expression:"),
   readln(80,"",Text), /*max 80 chars*/
   some_keywords(Query_keywords,Query_keywords,Query_keywords1),
   sys_vars(Sys_vars),
   required_vars(Query_keywords1,Req_vars,_),
   string_compact_list(Expr,Tokens),
   vars_in_expr(Ip_vars_a,Tokens),
   list_string_atom(Ip_vars,Ip_vars_a),
   delete_all1(Sys_vars,Ip_vars,Ip_vars1),
   delete_all1(Req_vars,Ip_vars1,Ip_vars2),
   add_non_dim3(Ip_vars2,Expr,Text,Query_keywords1),
   !,
   add_non_dim1(Query_keywords).
add_non_dim2(Expr,Query_keywords):-
   write("Failure in validation of "),
   write(Expr),
   write("\n"),
   add_non_dim1(Query_keywords).

add_non_dim3([],Expr,Text,Query_keywords1):-
   concat_list_string(Query_keywords1,Query_keywords2),
   assert(non_dim_new(Expr,Text,Query_keywords2)),
   !.
add_non_dim3(L,_,_,_):-
   write("\nThe following variables are not available with this"),
   write(" combination of keywords:\n"),
   write(L),write("\n").

/*-------
vars_exist/1 checks that a variable exists in the dictionary
appropriate to the current set of keywords.
-----*/
vars_exist(Vs,Keys):-
      vars_exist1(Vs,Keys),
      not( call(flag_no_var) ).
vars_exist(Vs,Keys):-
      call(flag_no_var),
      retractall(flag_no_var),
      write("\nThese variables must be defined before this expression"),
      write(" can be accepted."),
      fail.

vars_exist1([],_).
vars_exist1([V|Vs],Query_keys):-
      string_atom(V_str,V),
      diction(V_str,_,Keys),
      split_list_string(Keys1,Keys),
      all_members(Keys1,Query_keys),
```

```
    !,
    vars_exist1(Vs,Query_keys).
vars_exist1([V|Vs],Query_keys):-
    write("\nVariable "),write(V),write(" is undefined."),
    assert(flag_no_var),
    vars_exist1(Vs,Query_keys).

/*--------
Parser
- input a list of tokens (integers, variable names and operator symbols).
- returns a list of variables.
---------*/
vars_in_expr(Variables,Tokens):-
    expression(Variables,Tokens,[]),
    !.
vars_in_statement(Variables,Tokens):-
    statement(Variables,Tokens,[]),
    !.

statement(Variables) -->
    variable(X),[=],expression(Vs),
    { append([X],Vs,Variables) } .
statement(Variable) -->
    [write],constant(Variable).
statement(Variable) -->
    [error],constant(Variable).
statement(Variables) -->
    [if],condition(Variables),[then].
statement([]) -->
    [else].
statement([]) -->
    [end_if].
statement(Variables) -->
    [while],condition(Variables).
statement([]) -->
    [end_while].
statement(Vars) -->
    variable(X),variable(File),expression(Y),expression(Z),[table],
    { append(X,Y,Y1) },
    { append(Y1,Z,Vars) } .
statement(Vars) -->
    variable(X),variable(File),expression(Y),[table],
    { append(X,Y,Y1) } .
statement(Vars) -->
    variable(X),expression(A),expression(B),expression(C),[graph],
    { append(X,A,A1) },
    { append(A1,B,B1) },
    { append(B1,C,Vars) } .

expression(Vars) -->
    constant(X),arithmetic_op(Op),expression(Ys),
    {
    append(X,Ys,Vars)
    } .
expression(X) -->
    constant(X).

arithmetic_op([*,*,-]) --> [*],[*],[-].
```

```
arithmetic_op([*,*]) --> [*],[*].
arithmetic_op([+]) --> [+].
arithmetic_op([-]) --> [-].
arithmetic_op([*]) --> [*].
arithmetic_op([/]) --> [/].

constant(Vars) -->
	['('],expression(Vars),[')'].
constant([]) -->
	[X],
	{
	number(X)
	}.
constant([]) -->
	[-],[X],
	{
	number(X)
	}.
constant([X]) -->
	variable(X).
constant([]) -->
	[X],
	{
	string(X)
	}.

variable(X) -->
	[X],
	{
	atom(X)
	}.

condition(Vars) -->
	expression(X),
	comparison_op(Op),
	expression(Y),
	{ append(X,Y,Vars) } .

comparison_op(=:=) --> [=],[:],[=].
comparison_op(=\=) --> [=],[\],[=].
comparison_op(>=) --> [>],[=].
comparison_op(=<) --> [=],[<].
comparison_op(>) --> [>].
comparison_op(<) --> [<].

/*------
Sys_vars/1 obtains a list of constants eq. pi given in the dictionary
------*/
sys_vars(Sys_vars):-
sys_vars1,
sys_vars2(Sys_vars).

sys_vars1:-
   diction(Var,Text,"convection","n"),
   assert(sys_var(Var)),
   fail.
sys_vars1.
sys_vars2([V|Vs]):-
```

```
   retract(sys_var(V)),
   !,
   sys_vars2(Vs).
sys_vars2([]).

/*--------
Compile list of parameter names and associated text for which values are
required.
-------*/
required_vars(Query_keywords,Rev_names,Rev_texts):-
   required_vars1(Query_keywords),
   required_vars2(Names,Texts),
   reverse(Names,Rev_names),
   reverse(Texts,Rev_texts).

required_vars1(Query_keywords):-
   diction(Name,Text,L,Rule),
   split_list_string(Keywords,L),
   all_members(Keywords,Query_keywords),
   asserta(req_parm(Name)),
   asserta(req_parm_text(Text)),
   fail.
required_vars1(_).

required_vars2([Name|Names],[Text|Texts]):-
   retract(req_parm(Name)),
   retract(req_parm_text(Text)),
   !,
   required_vars2(Names,Texts).
required_vars2([],[]).


/*--------
current_eqns/3 returns a list of the equation numbers which also derive this
variable.
--------*/
current_eqns(Var,Query_keywords,L):-
   eqn(Var,Keywords,Eqn_no,1,Eqn),
   split_list_string(Keywords1,Keywords),
   similar_list(Query_keywords,Keywords1),
   assert(eqn_no(Eqn_no)),
   fail.
current_eqns(_,_,[N|Ns]):-
   retract(eqn_no(N)),
   current_eqns(_,_,Ns).
current_eqns(_,_,[0]).

/*----
recommended_non_dims/3 - returns a list of non-dimensional expressions
and corresponding text
----*/
recommended_non_dims(Query_keywords,Non_dims,Texts):-
      recommended_non_dims1(Query_keywords),
      recommended_non_dims2(Non_dims,Texts).

recommended_non_dims1(Query_keywords):-
      non_dim(Non_dim_expr,Text,Keywords),
      split_list_string(Keywords1,Keywords),
```

```
        all_members(Keywords1,Query_keywords),
        asserta(rec_non_dim(Non_dim_expr)),
        asserta(rec_non_dim_text(Text)),
        fail.
recommended_non_dims1(_).

recommended_non_dims2([Name|Names],[Text|Texts]):-
    retract(rec_non_dim(Name)),
    retract(rec_non_dim_text(Text)),
    !,
    recommended_non_dims2(Names,Texts).
recommended_non_dims2([],[]).

/*---------
Use menu from MMI.Pro to select a sub-list of the list of strings.
some_keywords(Query_keywords,Previous_selection,Final_selection)
-----------*/
some_keywords(Query,Prev_sel,Final_sel):-
        length(Query,A),
        B is A+9,
        some_keywords1(Prev_sel,M),
        D =.. [:,"Accept keywords",assert(selected_keyword(accept))],
        E =.. [:,"Original keywords",assert(selected_keyword(original))],
        append([D,E],M,M1),
        open(sk,3:46,B:34,w:blue:w,_),
        stdout(Out),
        stdout(sk),
        write("This value/relationship can\n"),
        write("be made more general by removing\n"),
        write("keywords from the description\n"),
        write("Select keywords to be removed"),
        menu("Keywords",vertical,8:46,w:blue:w,M1),
        retract(selected_keyword(K)),
        some_keywords2(Query,Prev_sel,K,Final_sel),
        stdout(Out),
        close(sk),
        !.

/*---
Create list for menu
------*/
some_keywords1([],[]).
some_keywords1([K|Ks],[D|Ds]):-
        D =.. [:,K,assert(selected_keyword(K))],
        !,
        some_keywords1(Ks,Ds).

/*------
Remove keyword and all descendants from the keyword list
--------*/
some_keywords2(Query,Prev,accept,Prev):-
       !.
some_keywords2(Query,_,original,Final):-
       !,
       some_keywords(Query,Query,Final).
some_keywords2(Query,Prev,Pk,Final):-
       label(Pn,Pk,_),
       bagof(Cn,sub_inherit(Pn,Cn),Cns),
```

```
node_label_list(Cns,Cks),
delete_list_list(Cks,Prev,A),
delete_member(Pk,A,A1),
!,
some_keywords(Query,A1,Final).
```
□

HEAT TRANSFER CORRELATION ADDITION

```
/*------------------------------------------------------------------
Copyright (c) Michael A Moss and Nottingham Trent University 1993

All rights reserved.
Use and copying of this software and preparation of derivative works based
on this software are permitted provided that this complete copyright and
permission notice is maintained, intact, in all copies and that full credit
is given to Michael A Moss and Nottingham Trent University.

Michael A Moss and Nottingham Trent University provide this software
without express or implied warranty.
------------------------------------------------------------------*/
/*----------

        Input new correlation data.
        ***************************

   Info. required  Authors + Year
                   Limits
                   Correlation expressions

----------------*/
add_data:-
   current_keywords(Query_keywords),
   write("\nEnter Authors' names:"),
   readln(80,"",Authors), /*max 80 chars*/
   write("Enter year of publication:"),
   readln(4,"",Year), /*max 4 chars*/
   write("\nEnter article title:"),
   readln(254,"",Title), /*max 254 chars*/
   write("Enter publication details:"),
   readln(254,"",Source), /*max 254 chars*/
   add_1st_key(Authors,Year,Key),
   add_2nd_key(Key,Key1),
   assert( ref_new(Key,Authors,Year,Title,Source) ),
   concat_list_string(Query_keywords,Query_keywords1),
   assert( descript_new(Key,Key1,Query_keywords1) ),
   sys_vars(Sys_vars),
   required_vars(Query_keywords,Req_vars,_),
   write("\nYou will be prompted for non-dimensional expressions and corresponding"),
   write("\nminimum and maximum limits. Recommended expressions for this "),
   write("geometry are:\n"),
   recommended_non_dims(Query_keywords,Non_dims,Non_dim_texts),
   write_table(Non_dims,Non_dim_texts),
   add_limits(Query_keywords,Req_vars,Sys_vars,Key,Key1),
   write("\nYou will be prompted for the statements making up a "),
   write("correlation.\nEnter these one line at a time."),
   write(" RETURN to end.\n"),
   add_corr(Query_keywords,Req_vars,Sys_vars,Key,Key1),
   write("\nTuples added to the database\n"),
   fact_listing(ref_new/5),
   fact_listing(descript_new/3),
   fact_listing(limits_new/5),
   fact_listing(corr_new/4),
   press_any_key.
```

```
/*-----
Add limit expressions and minimum and maximum values.
Check that variables used in the expression are known,
if not known print a warning.
---------*/

add_limits(Query_keywords,Req_vars,Sys_vars,Key,Key1):-
   write("\nEnter an expression or RETURN to end:"),
   readln(40,"",Exp), /*max 40 chars*/
   add_limits1(Query_keywords,Req_vars,Sys_vars,Key,Key1,Exp).


add_limits1(Query_keywords,Req_vars,Sys_vars,Key,Key1,""):-
   !.
add_limits1(Query_keywords,Req_vars,Sys_vars,Key,Key1,Exp):-
   string_compact_list(Exp,Tokens),
   vars_in_expr(Variables,Tokens),
   list_string_atom(V1,Variables),
   delete_all1(Sys_vars,V1,V2),
   delete_all1(Req_vars,V2,V3),
   unknown_vars(V3),
   write("Enter minimum expression value :"),
   read_float(Min),
   write("Enter maximum expression value :"),
   read_float(Max),
   assert( limits_new(Key,Key1,Exp,Min,Max) ),
   !,
   add_limits(Query_keywords,Req_vars,Sys_vars,Key,Key1).
add_limits1(Query_keywords,Req_vars,Sys_vars,Key,Key1,Exp):-
   write("Failure in validation of "),write(Exp),write(", re-enter\n"),
   add_limits(Query_keywords,Req_vars,Sys_vars,Key,Key1).

/*-----
Add correlation expressions.
Check that variables used in the expression are known,
if not known print a warning.
---------*/

add_corr(Query_keywords,Req_vars,Sys_vars,Key,Key1):-
   add_corr1(Sys_vars,Req_vars,[],0),
   concat_list_string(Query_keywords,Query_keywords1),
   add_corr4(Query_keywords1,Key,Key1).

add_corr1(Sys_vars,Req_vars,Result_vars,N):-
   write("Enter a new statement :"),
   readln(100,"",Statement), /*max 100 chars*/
   add_corr2(Sys_vars,Req_vars,Result_vars,N,Statement).

add_corr2(_,_,_,_,""):-
   !.
add_corr2(Sys_vars,Req_vars,Result_vars,N,Statement):-
   string_compact_list(Statement,Tokens),
   vars_in_statement(Variables,Tokens),
   !,
   add_corr3(Sys_vars,Req_vars,Result_vars,N,Statement,Tokens).
add_corr2(Sys_vars,Req_vars,Result_vars,N,Statement):-
   write("\nParsing failed for: "),
```

```
   write(Statement),
   write(", re-enter"),
   !,
   add_corr1(Sys_vars,Req_vars,Result_vars,N).

add_corr3(Sys_vars,Req_vars,Result_vars,N,Statement,[write,S]):-
   string(S),
   Line_no is N + 1,
   assert(corr_new(Line_no,Statement)),
   !,
   add_corr1(Sys_vars,Req_vars,[Result|Result_vars],Line_no).
add_corr3(Sys_vars,Req_vars,Result_vars,N,Statement,Tokens):-
   statement(Variables,Tokens,_),
   list_string_atom(Str_variables,Variables),
   list_head_tail(Str_variables,Result,Ip_vars),
   delete_all1(Sys_vars,Ip_vars,Ip_vars1),
   delete_all1(Req_vars,Ip_vars1,Ip_vars2),
   delete_all1(Result_vars,Ip_vars2,Ip_vars3),
   unknown_vars(Ip_vars3),
   Line_no is N + 1,
   assert(corr_new(Line_no,Statement)),
   !,
   add_corr1(Sys_vars,Req_vars,[Result|Result_vars],Line_no).

add_corr4(Query_keywords,Key,Key1):-
   retract(corr_new(Line_no,Statement)),
   assert(corr_new(Key,Key1,Line_no,Statement)),
   add_corr4(Query_keywords,Key,Key1).
add_corr4(_,Key,Key1):-
   write("\nAll statements have been added to a tempory database").

/*----
Warning for unknown variable names
-------*/
unknown_vars([]):-
   !.
unknown_vars(L):-
   write("\nUnknown Variables "),write(L),
   write("\nYou will need to add these variables before "),
   write("using this statement\n").

/*----------
Given the Authors and Year for a publication add_1st_key/2 finds a unique
key (this may already exist), the letters 'a' to 'z' may be appended if
required.
-----------*/
add_1st_key(Authors,Year,Key):-
      sub_string(Year,S1,3,2),
      concat("_",S1,S2),
      string_list(Authors,As),
      add_1st_key1(As,[],S3),
      concat(S3,S2,S4),
      add_1st_key2(S4,Authors,Year,Key).

add_1st_key1([A|As],Ls,S):-
      lexis(A,letter),
      length(Ls,Length),
      Length < 12,
```

```
    !,
    add_1st_key1(As,[A|Ls],S).
add_1st_key1(_,Ls,S):-
    reverse(Ls,Lr),
    string_list(S,Lr).

/* The reference is already in the database */
add_1st_key2(Key,Authors,Year,Key):-
    ref(Key,_,Authors,Year),
    !.
/* Use the new key */
add_1st_key2(Key,Authors,Year,Key):-
    not( ref(Key,_,_,_) ),
    !.
/* Derive a new key */
add_1st_key2(S,Authors,Year,Key):-
    add_1st_key3(S,Authors,Year,97,Key).

add_1st_key3(S,Authors,Year,A,Key):-
    string_list(S1,[A]),
    concat(S,S1,Key),
    ref(Key,_,Authors,Year),
    !.
add_1st_key3(S,Authors,Year,A,Key):-
    string_list(S1,[A]),
    concat(S,S1,Key),
    not( ref(Key,_,_,_) ),
    !.
add_1st_key3(S,A,Key):-
    B is A + 1,
    add_1st_key3(S,B,Key).

/*----------
Given the first key to a correlation record add_2nd_key/2 finds a unique
second key, trying 'a' first and progressing to 'z'.
----------*/
add_2nd_key(Key,Key1):-
    add_2nd_key1(Key,97,Key1).
add_2nd_key1(Key,A,Key1):-
    string_list(Key1,[A]),
    not( descript(Key,Key1,_) ),
    !.
add_2nd_key1(Key,A,Key1):-
    B is A + 1,
    add_2nd_key1(Key,B,Key1).
```

□