characterising the scope of a standard and its provisions is desirable to be incorporated in a standards processing model. Such a facility was incorporated in the system SADA-II Standards Automation Design Assistant-II (Badrah 2000), (Badrah, et. al 2000). This was based on a rule-based approach (Waterman 1986) where the scope rules are extracted from the standard itself and from other sources, e.g. code writers.

Standards contents interpretation relates to the determination of the exact meaning of information or data attributes given in standards clauses. This problem is more persistent in building regulations. One solution for this problem was to model the intention behind the ‘open-textured’ statements in a building regulation such as the American with Disabilities Act Accessibility Guide (1997). So instead of interpreting the meaning of the word ‘equivalent’ in the statement taken from this regulation “persons in wheelchairs must be provided with the same or equivalent access to a building”, the approach was to write an algorithm which can simulate a wheelchair moving through the space and check any obstruction to its path (Han, et. al 1998). However, in a structural code which is usually numerically-oriented, the problem can be solved by raising any undetermined information as queries to the user with a default value. So the user can be aware of interpretations made in relation to the standard and may make his own interpretation.

2.2 Regular and Non-Standard Designs

An issue related to scope interpretation is the handling of non-standard designs i.e. those which can not be handled directly or completely by a design standard. Examples of these are structural components for which requirements for some design issues can not be found in the relevant standard, e.g. an R.C. precast ledger beam (also called spandrel beam) if designed according to BS 8110 (1985). In SADA-II, a case-based approach was adopted for archiving information and design procedures related to non-standard designs.

On the other hand, some clauses in a design standard may state a set of assumptions; so the user needs to apply first principles based on these assumptions to establish a solution such as in the design of columns with irregular section shapes for uniaxial or biaxial bending combined with axial loading. The approach adopted in SADA-II for such situations to write a generic function which can handle an arbitrary structural member section in this case (Badrah, et. al 1999-b). This was realised in AutoCAD/Autolisp so the user needs only to draw the section then he will use the generic function for analysing the section or sizing the reinforcement of RC members. A second approach was adopted in the commercial CAD package Prokon^{TM} (1999) for the design of columns with irregular shapes is by using the finite difference analysis method. The advantage of the SADA-II approach is that the CAD developer need not write algorithms for resolving the mathematical properties of complex shapes because these can be obtained from the AutoCAD object database.

2.3 Human-Computer Interaction

A major factor of success or failure of a CAD application is its user interface. In early CAD applications, the keyboard was the only input device for interactive use of a computer. With the invention of the positioning and pointing devices (such as the mouse) and the windows-based systems, current CAD applications have become much more user-friendly than earlier ones. The interface elements of current computer environments are: windows, icons, pointers, menus, buttons, toolbars, palettes and dialog boxes. In many commercial CAD applications, these elements have been successfully utilised in their interface design. For example, dialog boxes are extensively used for data entry and selection. Pointers are used in some applications , e.g. Robot97™ (1999) as an excellent means of selecting a component (e.g., a beam or a column) by pointing to the area defining the space occupied by the component on the screen. After selecting a specific component, a checking or other process can be carried out. The use of graphics, e.g. 3D models of frames and structural members has also been another factor behind the more effective usability of current CAD applications

2.4 Integration with other CAD applications

Current commercial CAD applications developers such as Master Series™ (1999) and Robot97™ (1999) have developed overall integrated applications, e.g. for analysis, design and preparing production drawings of continuous beams. Another type of integration is the integration of CAD applications which are not necessarily developed by the same developer. In CIMsteel, analysis, design, detailing and other computer-based applications of steel frameworks are integrated through sending and receiving neutral-format files. These files are independent of the applications but their structure conforms to an integration standard developed for steel frameworks called CIS (Watson 1996). The developers of a CAD applications need to write an export translator and an import translator so their application can be integrated with other CAD applications which support the CIS. In SADA-II, a central database approach was proposed where CAD applications can read input data and send generated data to the central database through the Dynamic Data Exchange (DDE) technique (Petzold 1992) provided that all the integrated applications are Windows applications and they support DDE.

3 DEVELOPMENT TECHNOLOGIES

Some development technologies for detailed structural design applications are described. These include: numerical rule processors, databases, object-oriented software technology, graphics systems, neural networks and genetic algorithms. Although this in not a thorough collection of the development tools for CAD applications, but it gives a CAD developer a comprehensive idea about important tools currently available in the software world.

3.1 Numerical Rule Processors

A numerical rule processor uses a declarative approach to the processing of engineering calculations. The rules which link the parameters are set out in a separate rule set and the order of processing is established at run time (Macleod et. al 1993). Spreadsheets are basic numerical rule processors which have become standard calculation programs. They are preferred by many engineers for quick and custom calculations. The capabilities of spreadsheet programs are increasingly advancing. Their input/output interfaces are improved using dialog boxes and more programming capabilities are provided, e.g. by using Visual Basic™ in Excel™. Many spreadsheet applications have been developed for structural and civil engineering tasks such as:

- Determination of deflection influence line of simply supported beams and for deflection surface of simply supported plates (Hadi 1996).
- Obtaining optimal concrete mix with respect to cost and strength (Kasperkiewicz 1994)
- Calculation of structural system frequencies to avoid resonant conditions (Bohinski and Lee 1994)

3.2 Database Technology

Database systems have tremendous advantages in managing data and producing information in many engineering, business and other fields. The discussion here is concerned with the application of database systems (which are usually coupled with programming capabilities) in the detailed structural design. Current database systems such as MS Access^{TM} have powerful features which can be utilised for the management of structural design data and for other processes. A practical application of database technology is developed and used by Simpson Gumperts & Heger Inc. (Shipley and Peters 1998) for the design and installation of remedial repairs to a high-rise building's exterior wall. The wall is made of reinforced concrete and clad with limestone panels which are supported back to the wall by anchors. About 8700 panels were reviewed and the installation of 15000 remedial anchors were monitored by using this database application. The database was implemented using FoxPro^{TM} and integrated with (1) MSC/NASTRAN^{TM} finite element packages used for the structural analysis of each panel and (2) AutoCAD/AutoLisp used as for automatic drawing of the as-built panels. The main functions of the database in this application are:

- Extract panel and anchor data from the database and create an input data file for the structural analysis package.
- Produce summaries of panel and anchor quantities for bid purposes
- Define the physical attributes of each of the panels
- Generate status reports on anchor installation and create data errors forms automatically from the database
- Generate as-built panel drawing showing anchor information

The advantages of using the database system in this project was saving time to the contracting company and providing with great confidence in the integrity and accuracy of the data (Shipley and Peters 1998).

3.3 Knowledge-Based Systems

A knowledge-based system is a computer program that can manipulate knowledge (i.e., logical relationships) in a declarative mode (i.e., the process of handling the knowledge is separated from the knowledge base). The declarative mode is the basic feature that distinguishes a KBS from a procedural program. Knowledge-based systems have the following main advantages (Andriole 1985), (Adeli 1988):

1. Knowledge in a KBS is more explicit and accessible.
2. The knowledge-base can be gradually and incrementally developed over a period of time.
3. A general system with one inference mechanism can be developed for different types of applications simply by changing the knowledge base.
4. A KBS can explain its behaviour through an explanation facility.

An example of the application of KBS technology to structural engineering is the finite-element modelling framework by Turkiyyah and Fenves (1996). This is a knowledge-based framework for assisting users in setting up, interpreting, and hierarchically refining finite-element models in a structural engineering domain by explicit representation and incremental activation and refraction of modelling assumptions. SADA-II employs the KBS technology for standards scope interpretation, see Section 2.1.

3.4 Object-Oriented Software Technology

The emergence of object-oriented computing is considered by most specialists as a revolutionary step in computing. A key concept in the OO technology is the software encapsulation of data and procedures in objects (Cox 1986). Users see only the external view of the object but not its internal implementation. They can send the object a 'message' to perform one of its procedures. Booch (1994) describes the main characteristics of object-oriented development as: (1) it encourages the reuse of software components, (2) it leads to systems that are more resilient to change and (3) it appeals to the working of human cognition. In detailed structural design, many object-oriented models have been developed such as:

An example of the application of OOP technology in structural engineering is the generic model for building structures (Biedermann 1996) which combines both analysis and design of buildings. The model was originally developed for steel structure and was easily extended to apply to the design of wood buildings taking the advantage of the flexibility in maintenance and modification of object-oriented systems. SADA-II is built using object-oriented technology. It exploits the delegation -an OOP concept, (Gamma et. al 1995)- for even more support to software reusability (Badrah, et. al 1998).

3.5 Graphics-Based Systems

The use of graphics packages such as AutoCAD has become standard in today's structural design office. Current graphics packages incorporate some powerful features and tools which make them most suitable for design processing tasks in addition to their use as drafting packages. Most commercial packages such as CADS™ (1999) and Master Series™ have supporting functions to aid in detailing processes and production drawings preparation using standard graphics-based packages such as AutoCAD. Additionally, some commercial programs such as Master Series™ and Robot97™ have developed proprietary modules for automatic preparation of production drawings. These drawings can then be exported to AutoCAD for final preparation of the drawings. In SADA-II, the AutoCAD/Autolisp package was used for the development of generic functions which can aid in the analysis and synthesis of an arbitrary RC member sections, see Section 2.2. The same package was also used for extracting geometrical information from production drawings of continuous beams into the relevant SADA-II module for conformance checking of the design entities included.

3.6 Neural Networks and Genetic Algorithms (NNs and GAs)

Neural networks are AI applications which are capable of learning from examples for mapping an unknown relationship between given sets of inputs and outputs. Generic algorithms are adaptive search and optimisation techniques based on the theory of evolution. To develop an NN application, a large number of previously solved examples related to a specific problem usually need to be collected. From these examples, key parameters are to be identified with some of them to be chosen as input and others as output parameters. The NNs and GAs techniques are then applied, so a new problem context can be solved based on the exiting information. A huge number of applications in structural engineering have been developed using NNs and/or GAs. Some of these applications are:

- Optimisation of component design (Rafiq and Williams 1998)
- Prediction of ultimate shear strength of reinforced concrete deep beams (Sanad and Saka 1999)
- Analysis of masonry panels under biaxial bending by using NNs and case-based reasoning (Mathew, et. al 1999)

4 ANALYSIS

Each of the development technologies described in Section 3 has some advantages and disadvantages. The selection of the best technology depends on the type of application and its purpose. Also each of these technologies can be used on their own or as the main technology which is integrated with other technologies. Based on the survey of CAD applications presented in this paper, the following analysis of the development technologies described is made. Numerical rule processors, e.g. spreadsheets are most suitable for in-house software development, e.g. for handling custom calculations. Databases are suitable for structural engineering activities that need management of large amount of data items and repetitive operations. Knowledge-based systems can be used to capture knowledge developed over years of practical experience. Object-oriented software technology are most suitable for large-scale system development. Powerful features of graphics-based systems make them suitable not only for preparation of production drawing but also for assisting in some design activities e.g. for non-standard design processes that depend heavily on complex geometrical data. Neural networks and genetic algorithms can be utilised for non-linear design and analysis activities and for design optimisation. However, their main criticism is that they work in a black box manner, i.e. their answers can not be justified.

REFERENCES

Adeli, H., (Ed.), (1988). *Expert systems in construction and structural engineering*, Chapman and Hall, Bristol, UK.

American with Disabilities Act Accessibility Guidelines. (1997). Access Board, U.S. Architectural and Transportation Barriers Compliance Board, Washington, D.C., USA.

Andriole, S. J. (Ed.), (1985). *Applications in artificial intelligence*, Petrocelli Books, Princeton, USA.

Badrah, M. K., MacLeod, I.A., and Kumar, B. (1998) "Using object-communication for design standards modelling", *Journal of Computing in Civil Engineering*, ASCE, 12(3), pp. 153-161.

Badrah, M. K., MacLeod, I. A. and Kumar, B. (2000) "Design processing or regular and non-standard structural components", accepted for publication in *the Journal of Computing in Civil Engineering*, ASCE.

Badrah, M. K. (2000). *Information technology strategies for detailed structural design*, PhD thesis, Department of Civil Engineering, University of Strathclyde, Glasgow, UK.

Biedermann, J. D. (1996). "Addressing current issues in structural design software", Journal of Computing in Civil Engineering, ASCE, 10(4), pp. 286-294.

Bohinsky, J. A. and Lee, J. P. (1994). "Computer applications in structural analysis and design, Computing in Civil engineering, ASCE, pp. 462-65.

Booch, G. (1994). *Object-oriented analysis and design with applications*, 2ed edition, The Benjamin / Cummings Publishing Co., Redwood City, CA., USA.

BS 8110 (1985). *Structural Use of Concrete* BS *8110, Part 1-3*, British Standards Institution, London.

CADS (1999), Compter &Design Service Ltd., "http://www.cads.co.uk".

Cox, B. J. (1986). *Object-oriented programming: An evolutionary approach*, Addison-Wesley Publishing Co., Reading, Mass., USA.

Gamma, E., Helm, R., Johnson, R. and Vlissides, J. (1995). *"Design Patterns: Elements of Reusable Object-Oriented Software"*, Addison-Wesely, USA.

Hadi, M. N. S. (1996). "Utilising the capabilities of spreadsheets for designing structures", The International Journal of Construction Information Technology, Vol. 4, No. 2, pp. 15-28.

Han, C. S., Kunz, J. C. and Law, K. H. (1998-a). "A hybrid prescriptive-/ performance-based approach to automated building code checking", *Proc. of the International Computing Congress held in Conjunction with 1998 ASCE Annual Convention & Exhibition*, October 18-21, Boston, Mass, USA, pp. 537-548.

Kasperkiewicz, J. (1994). "Optimisation of concrete mix using a spreadsheet package", ACI Materials Journal, 91(6), pp. 551-59.

MacLeod, I. A., Ades, N. and Kor, F. H. (1993). "The use of numerical rule processors in engineering calculations", *Proceedings of the fifth international conference (V-ICCBE),* Anaheim, California, Vol. 2, pp. 1600-1607.

Master Series (1999), Civil and Structural Computer Services Ltd., Newtownabby, UK.

Mathew, A., Kumar, B., Sinha, B. P. and Pedreschi, R. F. (1999). "Analysis of masonry panel under biaxial bending using ANNs and CBR", *Journal of Computing in Civil Engineering,* ASCE, 13(3), pp. 170-177.

Petzold, C. (1992). *Programming Windows 3.1.* Microsoft Press, Redmond, Wash, USA.

Prokon (1999), Software Consultants Ltd, "http://www.prokon.com".

Rafiq, M. Y. and Williams, C. (1998). "An investigation to the integration of neural networks with the structured genetic algorithm to aid conceptual design", *in Ian Smith (Ed.): Artificial intelligence in structural engineering -Information technology for design, collaboration, maintenance and monitoring,* Springer-Verlag, Berlin, Germany, pp. 295-307.

Robot97 (1999), Integrated Structures Software Ltd., "http://www.robot97.com".

Sanad, A. and Saka, M. P. (1999). "Prediction of ultimate shear strength of reinforced concrete deep beams using neural networks", *Artificial intelligence Applications in civil and structural engineering, AICivil-Comp Conference*, Oxford, England, September, 13-15, pp. 147-157.

Shipley, S. F. and Peters, D. J. (1998) "Databases and curtain wall remediation - from design to filed work", *Proceedings. of the International Computing Congress held in Conjunction with 1998 ASCE Annual Convention & Exhibition*, Boston, Mass., USA, pp. 88-91.

Turkiyyah, G. M. and Fenves, S. J. (1996). "Knowledge-based assistance for finite-element modelling", *IEEE Expert Intelligent Systems and Their Applications*, 11(3), pp. 23-32.

Waterman, D. A. (1986). *A guide to expert systems*, Addison-Wesley Publishing Company, USA.

Watson, A. (1996). "The CIMsteel integration standards*", National Steel Construction Conference*, Phoenix, USA.