# On the integration of CAD and CAE

C.M. Hoffmann
R. Juan-Arinyo

Report LSI-94-38-R

# On The Integration of CAD and CAE

C.M. Hoffmann
Computer Sciences Department
Purdue University
West lafayette, IN 47907 USA

R. Juan-Arinyo*
Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya
Av. Diagonal 647, 8a, E-08028 Barcelona

September 1994

## Abstract

Between *computer-aided design* (CAD) and *computer-aided engineering* (CAE) there are major divisions. For example while CAD primarily focuses on the design of the form of mechanical parts and their geometric assembly into mechanical systems, CAE focusses particularly on the evaluation of the mechanical and structural performance of parts and assemblies. These divisions are associated with the fact that representations that have evolved to facilitate one or the other of these tasks differ fundamentally.

Integrating CAD and CAE has been attempted before. The principal problem that must be solved to accomplish integration is to devise an algorithmic exchange of information between the two subsystems. It seems to us that a promising approach to achieve closer integration between CAD and CAE would be to approach the problem from a different angle. Instead of seeking to integrate the two on the basis of existing representations, clearly a difficult problem, we propose to recast the problem from a perspective that precedes either. Elsewhere, we have argued that higher-level representations can be instrumental to lowering functional barriers between CAD, CAE and process planning. In this paper, we argue our view in more detail by a case study, and focus in particular on the nature and structure of the information required to solve both the shape design as well as the performance analysis.

The domain we consider in this case study is structural steel design. When approached within traditional engineering practice, this area of design is mature and well-understood. But when departing from the usual repertoire of standard beams and joints to connect them, there is a transition to the frontiers in structural analysis, and many challenging problems remain open.

---

*While on leave in Computer Sciences Department, Purdue University. Supported by a fellowship of the Direcció General de Recerca of the Generalitat de Catalunya.

# 1 Introduction

Design for manufacture is an extensive area of research and practice that is increasingly seen as an information-driven process. Research and development efforts have resulted in a number of commercial systems that address some of the aspects of design for manufacture. There remain major divisions, for example between *computer-aided design* (CAD) that primarily focuses on the design of the form of mechanical parts and their geometric assembly into mechanical systems, and *computer-aided engineering* (CAE) focusing particularly on the evaluation of the mechanical and structural performance of parts and assemblies. These divisions are associated with the fact that representations that have evolved to facilitate one or the other of these tasks differ fundamentally. For instance, CAD systems use primarily a *boundary representation* (Brep) that represents a solid shape as a quilt of surfaces, with appropriate topological information on incidence and connectivity. In constrast, CAE systems use a spatial subdivision representation of solids, meshing their volume with grids or with unstructured meshes of tetrahedra or hexahedra.

Integrating CAD and CAE has been attempted before. The principal problem that must be solved to accomplish integration is to devise an algorithmic exchange of information between the two subsystems. For example, if we have the shape design of a part using a CAD system, we must have an algorithm that meshes the shape. The many excellent approaches to mesh generation must contend with two difficulties:

1. The meshing problem is not entirely a geometric problem. Mesh quality and density may be critical in some spatial regions and less important in others.

2. Some shapes ought to be "abstracted." For instance, a rib may be replaced with a meshed middle surface rather than a meshed volume. Such abstraction depends on the physical problem under investigation, and are difficult to do using Breps.

In the case of the first problem, the main difficulty is that geometric structure and the physical problem interact. This interaction may be difficult to capture algorithmically because the shape representation is at a low level of abstraction and may not provide iportant information relating to more global spatial structure. In the case of the second problem, both application domain knowledge and spatial structure need to be correlated. Again the CAD representation does not provide much of the information needed to determine such correlation.

It seems to us that a promising approach to achieve closer integration between CAD and CAE would be to approach the problem from a different angle. Instead of seeking to integrate the two on the basis of existing representations, clearly a difficult problem, we propose to recast the problem from a perspective that precedes either. Elsewhere, we have argued that higher-level representations can be instrumental to lowering functional barriers between CAD, CAE and process planning. In this paper, we argue our view in more detail by a case study, and focus in particular on the nature and structure of the information required to solve both the shape design as well as the performance analysis. We believe that our approach has the folowing paradigmatic core:

1. By studying the information flow required for the different functional activities in engineering design, we create the prerequisites for formalizing this information and devising a unified global view of design data.

2. Once the design data has been formalized at the proper level of abstraction, the subproblems of shape design and performance analysis can be solved in a coordinated fashion.

Properly approached, there will be algorithms that effect a translation of the common design data into specialized representations familiar to individual CAD or CAE systems.

We believe that a proper level of abstraction is critical to accomplishing a solution of the integration problem that is both robust and technically simple. For this reason, we feel that the proper point of leverage would be at the level of conceptual design, and that the integration further downstream needs to be supported by a dialogue with the information content of the conceptual design.

The domain we consider in this case study is structural steel design. When approached within traditional engineering practice, this area of design is mature and well-understood. But when departing from the usual repertoire of standard beams and joints to connect them, there is a transition to the frontiers in structural analysis, and many challenging problems remain open. A credible solution to the integration problem, for structural engineering, would therefore be one where the information flow is cast into a framework that supports, on the one hand, automating traditional design, and facilitates, on the other hand, nontraditional design, beyond what can be accomplished with the traditional approaches.

## 2 Design Scenario

We envision the following scenario for traditional structural steel design. While we sketch it conceptually with reasonable completeness, note that we are primarily interested in the required information flow. Consequently, conveniences and accelerators for data entry in the various steps are important, but are not considered in deep detail in this paper because they are secondary to the information flow itself.

1. The user defines the topology of the steel structure, as well as the distances and angles of the structural elements. This activity requires minimal support in the form of a wireframe model and a geometric constraint solver that facilitates expressing these design parameters consistently and conveniently. Other support is desirable for minimizing the effort required to specify the abstract structure, as explained later.

2. The user defines live loads on the structure. To support this, the abstract joints must be annotated with loads and their direction.

3. The system performs a static analysis of the wireframe model under the assumption of no deformations. Both members and joints are considered to be perfectly rigid. Stresses in members and reactions in the supports are found. To support this activity, the wireframe model and the loads are input to some static analysis package. This may entail data formatting, but does not require complicated data conversion.

4. In traditional design, the user now chooses from a catalogue of possible beams and joint types, including whether the structural members are welded, bolted or fastened in other ways. No specific dimensions are fixed, only beam shapes and joint types. We need a data base for this as explained in the next step.

5. Based on the static analysis of Step 3, and on the beam shapes and joint types selected, the system suggests possible beam dimensions for each member, and the user selects from among them. To support this activity, a data base of standard structural shapes is needed that contains the material characteristics and shape parameters needed (a)

to compute whether the load requirements are satisfied using standard beam theory, and (b) to compute the detailed three-dimensional shape later. Such data bases simply automate handbooks.

6. Based on the dimensional choices of the user, the system now determines the dead loads and ascertains that the structure will perform as specified. This requires another (first-order) static analysis. If the performance margins are inadequate, the system informs the user, returns to Step 5, and suggests alternatives for the parts of the structure that need to be changed. This step requires simple computations that compute relative or absolute performance margins and can analyze where in particular the structure might be inadequate.
7. As second-order performance analysis, an analysis is performed on the wireframe model under different loads. Members are still axially loaded but, in contrast to the first-order analysis, they are now ideal elastic members and rotation at the joints is permitted. The result of this analysis determines the expected structural deflections in-the-large.
8. With adequate performance according to the first- and second-order analyses, the system now details the structural design by expanding the wire frame members according to beam type and dimensions, and by detailing the standard joints. The association between the detail design and the conceptual wireframe design is established at this point. Note that the association proceeds from the abstract design to the detail design by a process of expanding and coordinating higher-level information collected in the early design steps from the user, from standard libraries, and from the preliminary performance analysis.
9. Finally, a FEM analysis can be performed on the detailed joints as free-body problems using the boundary conditions established by the first- and second-order analyses.

The process is iterative in several ways. After checking for dead loads in Step 5, the process may repeat with Step 4. This is an incremental change that allows changing the beam sizes and types, but does not alter the abstract structure itself. It may become necessary to back up to Step 1 and alter the topology. Note that in experimental approaches such as [Papalambros,Bernitzas] there are some capabilities to suggest how to alter the topology so as to optimize the structure. Furthermore, the second-order analysis of Step 7 may suggest returning to earlier design steps, for example when too much deformation in the large is present. Such deformations can be quantified automatically and graded, given suitable models of acceptable deformations.

# 3 Analysis

## 3.1 First Order Analysis

Static analysis of steel structures such as trusses consists of the determination of reactions and internal forces and stresses caused by external loads. In static analysis, structures are usually idealized where each member is represented by a line of given length and each joint is represented by a point. Supports are also idealized to be from among a limited range of types, [11]. Generally, the equilibrium equations needed to carry out the static analysis are written

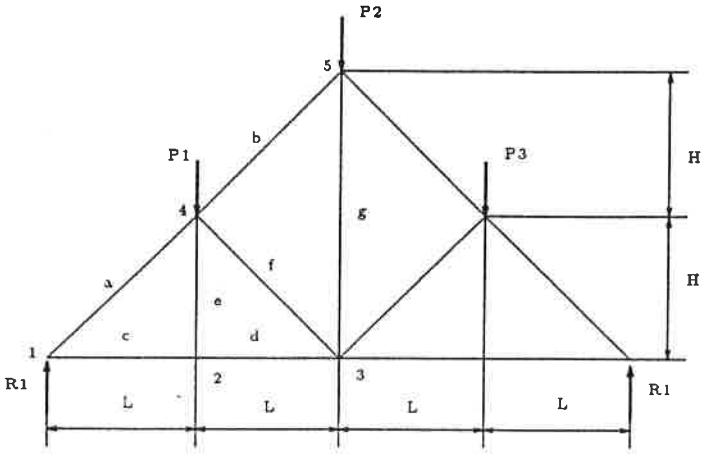

Figure 1: Simple truss showing geometry and external loads.

with respect to the undeformed geometry of the structure. This analysis is then referred to as the *first-order analysis*, [3].

Several methods to perform a first-order analysis of trusses have been reported in the literature. Among them are the method of sections, the method of joints, and graphical analysis; [11, 17]. For example, consider the method of joints. In this method, the computation proceeds by isolating joints one at a time. For each joint, the adjacent members are considered cut, and the forces transmitted by them are considered external forces acting on the isolated joint. Since the joint should be in equilibrium, the external forces must sum to zero. Since only two equilibrium equations are available for each joint, the procedure starts at a joint that has two or fewer unknowns. Usually this happens at a support. Figure 1 shows a simple example of a roof truss while Figure 2 shows the set of forces acting on the joints numbered 1, 2 and 4, respectively.

The information needed to carry out the computations using the method of joints consists of the joint identifiers, the coordinates of every joint, the external load applied to each joint (usually expressed by means of the vertical and horizontal components), and a pointer to each member connected to the joint. In addition, during the static analysis it is helpful to associate with each joint the number of adjacent members whose stresses are not yet computed. Moreover, for each member in the structure there should be a member identifier and a pointer to each of the two nodes it connects. As a result of the analysis computation, each member will be tagged with the stress it has to support. Clearly, all information except

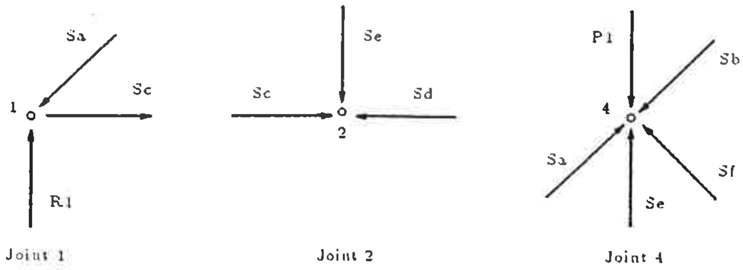

Figure 2: Calculation of stresses in a truss by method of joints.

the joint loads can be automatically determined from the conceptual design, and a convenient data structure to represent all information is a labelled graph whose nodes correspond to joints in the structure and whose edges correspond to the structural members. This data structure provides all the information needed for both statically determinate and statically indeterminate structures, [11].

Once the stresses in structure members are known, we can proceed to design each of them individually. Members under tensile stress are designed based on the assumption that the critical concern is the stress on the effective cross-section area. Designing compression members is not governed by stress alone but must account for the stability of the member as well. Since the slenderness ratio of structural members is usually quite high, compressed members are design as columns, [6, 11, 14].

The design procedure starts with the user choosing the desired shape for the members. Then the system will use standard specifications, [13]. In the absence of other known restrictions, the system would select the lightest-weight elements with the selected cross section shape that satisfy the design criteria, [14]. This process yields all the material properties of the structural shapes needed: cross-sectional area, weight, inertia, radii of gyration, plastic modulus, etc. These properties can be added to the graph annotations automatically by querying an engineering data base.

## 3.2 Second Order Analysis

In static analysis the members of a structure are considered to be perfectly rigid bodies subject only to axial loads. This is a reasonable approximation some of the time. Nevertheless, in reality all structures deflect under loading. The effect of this loading, on the deformed structure, creates the *second-order* effects, [3]. Such second-order effects degrade the performance and stability of the structure. Therefore, a static design should be validated by computing whether this degradation is acceptable.

As a first approximation to the real behaviour of structures, we assume that structure members support only axial loads, and that they are no longer rigid but instead ideal elastic members. Moreover, we assume that joints are idealized as points and that members are allowed to rotate with respect the joints.

Under these assumptions, first-order analysis no longer applies and a different method should be used. A standard approach, called *second-order analysis*, [3], allows to take into account the second-order effects in the structure with these assumptions. This approach assumes that if a structure is in equilibrium and small displacement theory is valid, then there is a unique relationship between the deformation of the structure and the loads applied to it. That is, the structure will take up one, and only one, deformed shape under the action of a given set of loads, [2, 11].

Several methods to carry out second-order analysis are known. Among them, the *matricial methods* have been succesfully implemented, [2, 4, 7, 8].

In the *matrix displacements* method, the nodal displacements $\Delta$ are expressed as a linear function $F$ of the applied loads $P$, and the equation $\Delta = FP$ is solved. In the *matrix force* method, also called the *matrix stiffness* method, loads are chosen as the independent variables and are linearly related to the displacements by the structure stiffness matrix $K$, so that the equation $P = K\Delta$ must be solved. The matrices $F$ and $K$ in these computation models are determined from the material properties, and the length, orientation, cross-sectional area and inertia of each member in the structure, [2, 8, 11]. Both matrices can be computed automatically from the graph data generated in the preceding member design phase.

When the matrix stiffness method is applied to problems in continuum mechanics it is commonly known as the *Finite Element Method* (FEM), [5, 9, 18, 20]. Its generality and versatility have made of it one of the most widely used methods in computerized structural analysis, [1, 2, 7, 8, 15], since it is especially well suited for the second-order analysis of structures.

The task of a structural analyst using a typical FEM code [15], can be summarized by the following steps, [6]:

1. Divide the structure into suitable elements.
2. Number the nodes and elements.
3. Record coordinates of each node.
4. Identify the nodes that go with each element.
5. Specify the material properties of each element.
6. Specify the geometric properties of each element.
7. Specify the geometric constraints.
8. Specify the nodal loads.

For the sake of simplicity in the explanation, let us assume that the FEM elements will have a linear behavior. This allows each member in the structure to be properly represented by a bar or rod element, [15, 16], while the joints in the structure are the nodes in the model. The FEM computation will then determine the displacements at the nodes.

It is immediate that all steps required to set up the FEM computation can be done automatically, from the annotated graph determined in the earlier design, and the loads determined from the first-order analysis. If the first order analysis is bypassed, however, the nodal loads would have to be determined explicitly by the user.

## 4 Generation of an Erep

Once first- and second-order analyses have determined the feasibility of the structure as designed up to this point, full detail design is approapriate. The main detail design issue concerns the joints and the details of how the structural members connect at the joints. The detail design subsequently is validated by analyzing the joints in full detail and, if needed, the structural members as well.

Conceptually, we have designed a complete skeleton structure in which the structure members have nominal lengths and cross sections, as well as a fixed position in 3-space. First- and second-order analyses have validated the choice of cross section, material properties, and the performance of the structure assuming adequate joints. The user would now select joint types and fastening methods. For example, gusset plates could be chosen at selected nodes, and the user can direct on which side the structure members ought to be attached. The design details impacting the subsequent joint analysis will be explained later.

To generate a solid model of the complete detailed structure, we must extrude the member cross sections to their nominal lengths. They are positioned in 3-space according to the coordinates of the adjacent idealized joints. They have to be trimmed using design engineering rules. The skeletal segments idealizing a member must be aligned with the line of action of the member. The orientation of the member about this axis must be chosen by the user.

The gussets or other joint types must be placed in accordance with the sidedness choices of the user. For a gusset, the user has to identify a reference point that is to be aligned with the skeletal joint. Orientation with respect to the adjacent members is accomplished in two steps. A reference plane is identified on the gusset that must be parallel to the plane in which selected adjacent members lie in 3-space. Finally, rotation about the reference point in this plane is determined by identifying a major direction on the reference plane and dimensioning the angle formed with the skeleton line of a selected structural member.

The main design steps, therefore, are at this point to define the detail shape of each joint, selecting a method of fastening, and identifying the spatial position and orientation of each joint by constraints. If a joint is formed by welding trimmed structural members to each other, no further joint specification is required.

Trimming the detail shape of the members can be highly automated. The process is an excellent opportunity to formalize and apply engineering desing rules. Examples of such design rules include:

1. Lines of action of all members connected to a joint should converge to a point, so that there are no bending moments at the joint.

2. Members should not be cut whenever geometry allows them to be connected according to rule 1. This rule minimizes unnecessary shop operations.

3. In roof and bridge trusses we always can identify an upper chord and a lower chord. Web members can be attached to them either perpendicularly or obliquely. Only those members attached obliquely need to be cut, as perpendicularly attached members fit naturally.

4. When a column is built as a truss, those members having the same main direction as the compression loads will be taken as reference members; then rules 2 and 3 will be applied. Such members naturally generalize the notion of upper and lower chord members.

5. In space structures we can often determine a dominant direction such that members in this direction have a length larger than the others; for example, in antennas, solar panel supports and others, such a direction is pronounced and can be taken as reference direction with respect to which rules 2 and 3 are applied.

## 5 Joint Performance Analysis

Let us now consider the problem of assessing the local behaviour of a connection joint in a structure for which the static analysis has been carried out and a detailed design is available in Erep. Assume that the structure has been designed so that the members are double angles that are to be welded to gusset plates at the joints.

For all but the simplest types of gusset plates, there is no direct method of automated design, [19]. The outline of a gusset plate is usually controlled by edge-distance requirements and by the number of bolts or the length of the welding seam required to connect each of the adjoining members. The thickness of the plate is usually established by comparison

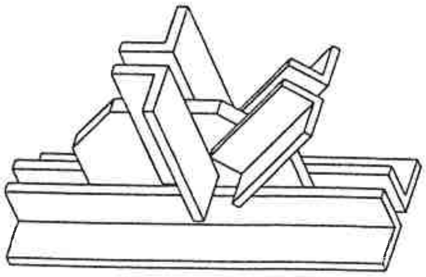

Figure 3: Four members connected by welding through a gusset plate.

with existing similar structures. This practice leads to gusset plates that have generally ample strength. However, it is desirable to assess gusset-plate stresses. When the connection detail design is unusual, with respect to the loads or in the joint layout, moreover, then this assessment is mandatory.

Classical methods, [19], carry out an estimate of the stresses at some critical sections of the gusset plates. One problem with this approach is that it is not obvious how to identify *a priori* what a *critical section* is when shear and buckling is a consideration.

Modern theories model joints by means of the relationship between the moment transmitted to the joint by a connected member and the rotation the member undergoes. This is known as the *M-Φ model*, [3, 10]. Despite intense research on the M-Φ model, the complexity of accurately modeling the large number of structural actions present in a joint has so far limited attempts to produce M-Φ relationships using theoretical approaches, [10]. Moreover, the M-Φ model represents the behavior of the joint as a whole and does not provide much insight into local deformations. Thus, the analysis of stresses in gussets using FEM appears to be the best approach. Consider the connection in Figure 3, of four structural members. The gusset can be modeled as a plate element and represented and analyzed in terms of the displacements of its middle surface. The part of each connected member that supports the welding through which loads are delivered can be considered as a bar or a beam depending on whether the neutral axis and the axis of shear centers of the element coincide or not. Because the neutral axis of these bars do not intersect the middle surface of the gusset, and since they do not necessarily intersect each other, they should be modeled as eccentric members with respect to the plate, [12, 15]. Hence, the gusset plate connecting the members can be considered as an eccentric, multiply-stiffened plate. The forces acting on it are membrane loads, transmitted through the members; there are no bending loads, [5, 15]. Figure 4 shows the upper part of a finite element model for the joint in Figure 3.

Deriving this finite element model from the joint geometry given in the Erep representation is not difficult. First the set of members connected to the plate is determined through the mating rules present in the Erep. The geometry of the plate is just the geometry of the gusset. As the location and orientation of each connected member with respect to the gusset plate

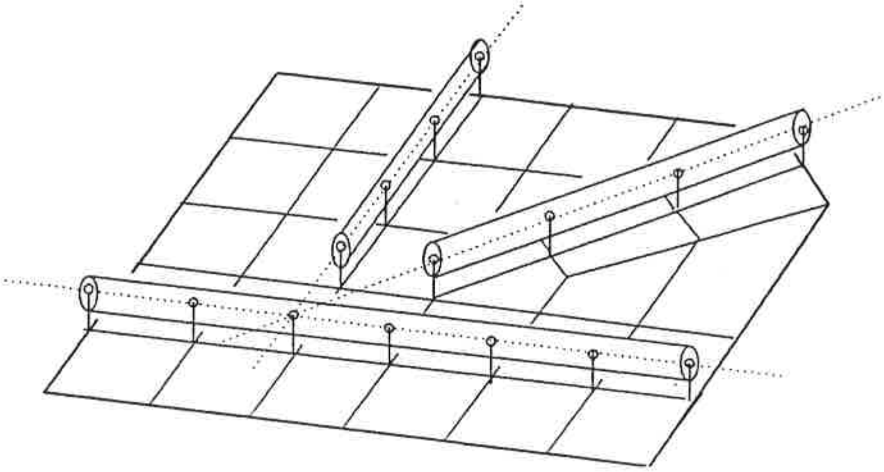

Figure 4: A finite element model of a connection.

are known from the Erep, the length of the bars in the model can be computed. Finally, plate thickness, and material and geometric properties of connected members determine the offset at which the bar's neutral axis, or beam's axis of shear centers, should be placed.

## 6 Ideas

1.- Structural design is NOT feature-oriented; at least in the first step. It does not make sense to go into further detail until a feasibility design has been completed.

2.- It is a bad practice to try to design universal solutions. The approach "specific problem ==> specific solution" has been proven to be much more productive. Hence,

  a) Can we extend the Erep to fit the first step of structural design? Is it convenient to make such an extension?

  The answer seems to be NO.

  b) Can we fit the structural design problem in the already available Erep features? We can consider the following Erep "customization":

    b.1) The analysis graph can be represented "geometrically" by menas of features such that, for the moment, only have a datum segment which represents the ideal structural member. Taking into account that, in the end, all members will be generated as extrusions of their cross-section areas, these features could be viewed as templates where to hold the final geometry of the member.

What we called "feature interference graph" could be of help here.

b.2) Other information needed or derived by computation (loads, sections, inertias, etc) could be represented as attributes.

In order to facilitate a later compilation, each attribute could be named.

3.- Once the feasibility study is completed, deriving the Erep representation can be highly automated.

4.- Now the Erep contains de full, detailed geometry and we are in a situation similar to that of feature-oriented design. Furthermore, Erep has the graph that represents the skeletonal structure.

Further and more accurate analysis, such those on local behaviour of joints, can now be performed.

HENCE, OUR RATIONAL IS THAT EVEN IN A NO FEATURE-ORIENTED ENVIRONMENT AN Erep CAN BE OF MUCH HELP.

# References

[1] R.J. Astley. *Finite Elements in Solids and Structures*. Chapman and Hall, London, 1992.

[2] J.A.D. Balfour. *Computer Analysis of Structural Frameworks. Second edition.* Oxford University Press, New York, 1992.

[3] W.F. Chen and E.M. Lui. *Stability Design of Steel Frames.* CRC Press, Boca Raton, Florida, 1991.

[4] D. Clarke. *Computer Aided Structural Design.* John Wiley and Sons, Chichester, UK, 1978.

[5] R.D. Cook, D.S. Malkus, and M.E. Plesha. *Concepts and Applications of Finite Element Analysis. Third Edition.* John Wiley and Sons, New York, 1989.

[6] J.G. Eisley. *Mechanics of Elastic Structures. Classical and Finite Element Methods.* Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[7] J.F. Fleming. *Computer Analysis of Structural Systems.* McGraw-Hill Book Company, New York, 1989.

[8] S.M. Holzer. *Computer Analysis of Structures.* Elsevier, New York, 1985.

[9] B. Irons and S. Ahmad. *Techniques of Finite Elements.* Ellis Horwood Ltd., Chichester, UK, 1980.

[10] R.F. Lorenz, B. Kato, and W.-F. Chen, editors. *Semi-Rigid Connections in Steel Frames.* McGraw-Hill, Inc., 1993.

[11] F.S. Merrit, editor. *Structural Steel Designers Handbook.* McGraw-Hill, New York, 1972.

[12] C. Meyer, editor. *Finite Element Idealization.* American Society of Civil Engineers, New York, 1987.

[13] American Institute of Steel Construction, editor. *Manual of Steel Construction, 8th edition.* Chicago, 1980.

[14] H. Parker and J. Ambrose. *Simplified Design of Steel Structures. Sixth edition.* J. Wiley and Sons, Inc., New York, 1993.

[15] H.G. Schaeffer. *MSC/Nastran Primer.* Wallace Press, Inc., Milford, NH, 1979.

[16] J.W. Schwalbe. *Finite Element Analyisis of Plane Frames and Trusses.* Robert E. Krieger Publishing Co., Malabar, Florida, 1989.

[17] S.P. Timoshenko and D.H. Young. *Theory of Structures.* McGraw-Hill Book Co., New York, 1965.

[18] W. Weaver and P.R. Johnston. *Finite Elements for Structural Analysis.* Prentice-Hall, Englewood Cliffs, N.J., 1984.

[19] C.D. Williams and E.C. Harris. *Structural Design in Metals. Second Edition.* The Ronald Press Company, New York, 1957.

[20] O.C. Zienkiewicz. *The Finite Element Method.* McGraw-Hill, London, 1989.

**Departament de Llenguatges i Sistemes Informàtics**
Universitat Politècnica de Catalunya

## Research Reports – 1994

LSI-94-1-R “Logspace and logtime leaf languages”, Birgit Jenner, Pierre McKenzie, and Denis Thérien.

LSI-94-2-R “Degrees and reducibilities of easy tally sets”, Montserrat Hermo.

LSI-94-3-R “Isothetic polyhedra and monotone boolean formulae”, Robert Juan-Arinyo.

LSI-94-4-R “Una modelización de la incompletitud en los programas” (written in Spanish), Javier Pérez Campo.

LSI-94-5-R “A multiple shooting vectorial algorithm for progressive radiosity”, Blanca Garcia and Xavier Pueyo.

LSI-94-6-R “Construction of the Face Octree model”, Núria Pla-Garcia.

LSI-94-7-R “On the expected depth of boolean circuits”, Josep Díaz, María J. Serna, Paul Spirakis, and Jacobo Torán.

LSI-94-8-R “A transformation scheme for double recursion”, José L. Balcázar.

LSI-94-9-R “On architectures for federated DB systems”, Fèlix Saltor, Benet Campderrich, and Manuel García-Solaco.

LSI-94-10-R “Relative knowledge and belief: SKL preferred model frames”, Matías Alvarado.

LSI-94-11-R “A top-down design of a parallel dictionary using skip lists”, Joaquim Gabarró, Conrado Martínez, and Xavier Messeguer.

LSI-94-12-R “Analysis of an optimized search algorithm for skip lists”, Peter Kirschenhofer, Conrado Martínez, and Helmut Prodinger.

LSI-94-13-R “Bases de dades bitemporals” (written in Catalan), Carme Martín and Jaume Sistac.

LSI-94-14-R “A volume visualization algorithm using a coherent extended weight matrix”, Daniela Tost, Anna Puig, and Isabel Navazo.

LSI-94-15-R “Deriving transaction specifications from deductive conceptual models of information systems”, María Ribera Sancho and Antoni Olivé.

LSI-94-16-R “Some remarks on the approximability of graph layout problems”, Josep Díaz, María J. Serna, and Paul Spirakis.

LSI-94-17-R "SAREL: An assistance system for writing software specifications in natural language", Núria Castell and Àngels Hernández.

LSI-94-18-R "Medición del factor modificabilidad en el proyecto LESD" (written in Spanish), Núria Castell and Olga Slávkova

LSI-94-19-R "Algorismes paral·lels SIMD d'extracció de models de fronteres a partir d'arbres octals no exactes" (written in Catalan), Jaume Solé and Robert Juan-Arinyo.

LSI-94-20-R "Una paral·lelització SIMD de la conversió d'objectes codificats segons el model de fronteres al mdel d'octrees clàssics" (written in Catalan), Jaume Solé and Robert Juan-Arinyo.

LSI-94-21-R "Clausal proof nets and discontinuity", Glyn Morrill.

LSI-94-22-R "A formal method for the synthesis of update transactions in deductive databases without existential rules", Joan A. Pastor.

LSI-94-23-R "On the sparse set conjecture for sets with low density", Harry Buhrman and Montserrat Hermo.

LSI-94-24-R "On infinite sequences (almost) as easy as $\pi$", José L. Balcázar, Ricard Gavaldà, and Montserrat Hermo.

LSI-94-25-R "Updating knowledge bases while maintaining their consistency", Ernest Teniente and Antoni Olivé.

LSI-94-26-R "Structural facilitation and structural inhibition", Glyn Morrill.

LSI-94-27-R "Formalising existential rule treatment in the automatic synthesis of update transactions in deductive databases", Joan A. Pastor.

LSI-94-28-R "Proceedings of the Fifth International Workshop on the Deductive Approach to Information Systems and Databases" (Aiguablava, 1994), Antoni Olivé (editor).

LSI-94-29-R "Towards a VRQS representation", Mariona Taulé Delor.

LSI-94-30-R "MACO: Morphological Analyzer Corpus-Oriented", Sofía Acebo, Alicia Ageno, Salvador Climent, Javier Farreres, Lluís Padró, Francesc Ribas, Horacio Rodríguez, and Oscar Soler.

LSI-94-31-R "Lexical mismatches: a semantic representation of adjectives in the LKB", Salvador Climent and Carme Soler.

LSI-94-32-R "LDB/LKB integration", German Rigau, Horacio Rodríguez, and Jordi Turmo.

LSI-94-33-R "Learning more appropriate selectional restrictions", Francesc Ribas.

LSI-94-34-R "Oracles and queries that are sufficient for exact learning", Nader H. Bshouty, Richard Cleve, Ricard Gavaldà, Sampath Kannan, and Christino Tamon.

LSI-94-35-R “Acquisition of lexical translation relations from MRDs”, Ann Copestake, Ted Briscoe, Piek Vossen, Alicia Ageno, Irene Castellón, Francesc Ribas, German Rigau, Horacio Rodríguez, and Anna Samiotou.

LSI-94-36-R “The query complexity of learning context-free grammars”, Carlos Domingo and Víctor Lavín.

LSI-94-37-R “Learning minor closed graph classes with membership and equivalence queries”, John Shawe-Taylor, Carlos Domingo, Hans Bodlaender, and James Abello.

LSI-94-38-R “On the integration of CAD and CAE”, C.M. Hoffmann and R. Juan-Arinyo.

---

Copies of reports can be ordered from:

Nuria Sánchez
Departament de Llenguatges i Sistemes Informàtics
Universitat Politècnica de Catalunya
Pau Gargallo, 5
08028 Barcelona, Spain
`secrelsi@lsi.upc.es`