Engineering Design through Constraint-Based Reasoning

Niall Murtagh

My dissertation concerns the application of constraint-based reasoning to parametric engineering design (Murtagh 1991).[1] It deals with the practical application of constraint networks, using automated reasoning to overcome some of the blind spots in conventional iterative design.

Parametric engineering design refers to routine-level design (Brown and Chandrasekaran 1985) in which the parameters and variables describing the design object are known, and the problem is one of finding a consistent set of parameter values that conform to specified requirements. It involves converting a well-established symbolic representation of an object, consisting of a set of parameters and variables, into a specific numeric representation. This conversion involves the attachment of numeric values to the parameters and the use of analysis programs to either verify the consistency of these values or eliminate inconsistent values.

Conventional methods of parametric design rely on the iterative reuse of analysis programs to converge on a satisfactory solution. Finite-element and other analysis programs require considerable computer resources; are unidirectional; and are inflexible in that they require a complete reprocessing, irrespective of how small the change is that was made to the previous design description. Furthermore, exact values are required, and imprecise data cannot be dealt with.

My research proposes a general method to minimize the use of the analysis programs and enable bidirectional reasoning by availing of constraint-based reasoning to carry out redesign (Murtagh and Shimura 1990, 1991). A problem solver, consisting of constraint networks that express basic relationships between individual design parameters and variables, is attached to the analysis programs, so that these programs can provide initial values for the parameters. In redesign and optimization, however, the networks alone can then reason about required adjustments to find a consistent set of parameter values.

Constraint Representation

My dissertation compares various representations of design constraints and sets out how global constraints representing standard design behavioral equations can be decomposed to form binary constraint networks. These networks can then use backward reasoning to communicate required adjustments from performance parameters back to input variables and, thus, determine dependencies between key parameters. Further development of the networks enables the same reasoning system to forward propagate executed adjustments, using relationship information to update only those parts of the design description that are affected by the adjustment. Hence, the redesign process can be simulated by constraint reasoning alone, with the analysis programs used only for setting out an initial description. Single-parameter optimization is implemented by extending the reasoning process used in redesign.

Various issues involved in propagation are then considered, comparing them with the techniques used by Leler (1988), Dechter and Pearl (1988), and Mackworth (1977). Forward propagation through the constraint network is exact because it follows the steps of conventional algebra. Back propagation, however, is generally approximate because of loops or cycles in the constraint networks and generally relies on ad hoc procedures to converge on a solution. At best, back propagation gives exact accuracy (for single-path, or no-loop, dependencies) and good approximations for other multipath dependencies. At worst, it relies on a heuristic correction factor to steer the adjustment in the right direction. In the latter case, the system uses a general hill-climbing strategy as a fallback when back propagation is not adequate, that is, when the reasoning system tends not to converge on a solution. The implementation of interval propagation provides an additional efficient method of overcoming the problem of network cycles.

Interval Propagation

Current parametric engineering design systems can only process problems expressed in terms of exact numbers, mainly because of the requirements of analysis programs. However, constraint networks for propagating exact values can be adapted to hold ranges of values or intervals. Besides the obvious benefit of being able to consider non-exact values, my research shows how the propagation of intervals facilitates the search for a global optimum in continuous, nonmonotonic parameter performance curves, to which conventional hill climbing cannot conveniently be applied. By considering ranges of values simultaneously and determining the approximate maximum and minimum of the performance curve values they produce, it is possible to move more efficiently toward the optimum value than is possible with conventional discrete-value processing. This practical application of interval propagation is compared

with the theoretical work of Hyvonen (1989). It also gives a different perspective on the redesign methods discussed by Orelup et al. (1988), who proposed variations on conventional hill climbing but always with exact values and repeated analysis.

Test Sessions

A constraint network was implemented with an analysis program for structural concrete design, and design sessions were performed to demonstrate how redundant analysis could be avoided and examine different aspects of the reasoning and propagation strategies provided. Interval propagation is shown to generally enable convergence on a global optimum in a finite number of steps. Although interval propagation is generally useful, it is shown to be more expensive than single-value propagation and is only used where nonmonotonic performance functions are involved. When back propagation does not give good dependencies, the reasoning mechanism is shown to resort to a version of hill climbing using forward propagation alone, but it is not necessary to reanalyze at any stage after the initial trial design. The different forms of communication between parameters and variables within the design are compared for accuracy and expense.

Conclusions

My research has shown how the parametric design paradigm can be improved by (1) separating the initial design stage from that of redesign; (2) using reasoning networks to simulate analysis; and (3) using various techniques, including the propagation of intervals, to overcome network cycles. The constraint networks do not replace conventional analysis programs but complement them in a symbiotic system. Thus, it is proposed that the successful combination of representations and strategies is the key to improving the parametric design paradigm.

Acknowledgments

My thanks to Masamichi Shimura, who supervised this work, and the Japanese Ministry of Education for providing a Monbusho Scholarship.

Note

1. To obtain a copy of this dissertation, please contact International Cooperation Section, National Diet Library, Chiyoda-ku 1-10-1, Tokyo 100, Japan, or the author at Mitsubishi Electric Corp., 5-1-1 Ofuna, Kamakura, Kanagawa 247, Japan.

References

Brown, D., and Chandrasekaran, B. 1985. Expert Systems for a Class of Mechanical Design Activity. In *Knowledge Engineering in Computer-Aided Design: Proceedings of the IFIP WG 5.2 Working Conference*, 259–282. Amsterdam: North-Holland.

Dechter, R., and Pearl, J. 1988. Network-Based Heuristics for Constraint-Satisfaction Problems. *Artificial Intelligence* 34:1–38.

Hyvonen, E. 1989. Constraint Reasoning Based on Interval Arithmetic. In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence, 1193–1198. Menlo Park, Calif.: International Joint Conferences on Artificial Intelligence.

Leler, W. 1988. *Constraint Programming Languages, Their Specification and Generation*. Reading, Mass.: Addison-Wesley.

Mackworth, A. K. 1977. Consistency in Networks of Relations. *Artificial Intelligence* 8(1): 99–118.

Murtagh, N. 1991. Engineering Design Using Constraint-Based Reasoning, Ph.D. diss., Dept. of Computer Science, Tokyo Institute of Technology.

Murtagh, N., and Shimura, M. 1991. A Constraint-Based Hybrid Engineering Design System. In *Intelligent CAD, III*, eds. H. Yoshikawa, F. Arbab, and T. Tomiyama, 217–229. Amsterdam: North-Holland.

Murtagh, N., and Shimura, M. 1990. Parametric Engineering Design Using Constraint-Based Reasoning. In Proceedings of the Eighth National Conference on Artificial Intelligence, 505–510. Menlo Park, Calif.: AAAI.

Orelup, F.; Dixon, J. R.; Cohen, P. R.; and Simmons, M. K. 1988. DOMINIC II: Meta-Level Control in Iterative Redesign. In Proceedings of the Seventh National Conference on Artificial Intelligence, 25–30. Menlo Park, Calif.: AAAI.

Niall Murtagh is a research scientist with Mitsubishi Electric, Kamakura, Japan. His research interests include engineering design, constraint-satisfaction problems, and automated reasoning.

Books Received

■ Avresky, Dimitri, editor. *Hardware and Software Fault Tolerance in Parallel Computing Systems*. New York: Ellis Horwood, 1993. 334 pp. ISBN 0-13-381021-6.

■ Blakemore, Colin. *Vision: Coding and Efficiency*. New York: Cambridge University Press, 1993. 448 pp. $34.95. ISBN 0-521-44759-0.

■ Dorr, Bonnie Jean. *Machine Translation: A View from the Lexicon*. Cambridge, Mass.: The MIT Press, 1993. 432 pp. $45.00. ISBN 0-262-04138-3.

■ Farr, Marshall, and Psotka, Joseph, editors. *Intelligent Instruction by Computer: Theory and Practice*. Bristol, Penn.: Taylor and Francis, 1993. 278 pp. ISBN 0-844-81687-6.

■ Feijs, Loe. *A Formalization of Design Methods*. New York: Ellis Horwood, 1993. 502 pp. ISBN 0-13-106113-5.

■ Gonzalez, Avelino J., and Dankel, Douglas D. *The Engineering of Knowledge-Based Systems: Theory and Practice*. Englewood Cliffs, N.J.: Prentice Hall, 1993. 523 pp. ISBN 0-13-276940-9.

■ Heck, André. *Introduction to* MAPLE. Berlin: Springer-Verlag, 1993. 497 pp. ISBN 3-540-976-62-0.

■ Hinton, Jeremy, and Pinder, Alan. *Transputer Hardware and System Design*. Englewood Cliffs, N.J.: Prentice Hall, 1993. 286 pp. ISBN 0-13-953001-0.

■ Hori, Motoo, editor. *Theoretical and Applied Mechanics: Proceedings of the Forty-First Japan National Congress for Applied Mechanics*. Tokyo: University of Tokyo Press, 1993. $160. ISBN 0-86008-493-6.

■ Jones, Joseph L., and Flynn, Anita M. *Mobile Robots: Inspiration to Implementation*. 348 pp. $39.95. ISBN 1-56881-011-3.

■ Kaelbling, Leslie Pack. *Learning in Embedded Systems*. Cambridge, Mass.: The MIT Press, 1993. 176 pp. ISBN 0-262-11174-8.

■ Lajoie, S., and Derry, S. *Computers as Cognitive Tools*. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1993. 401 pp. $79.95, cloth. $32.50, paper. ISBN 0-8058-1082-X.

■ MacGregor, Ronald J. *Theoretical Mechanics of Biological Neural Networks*. Boston: Boston Academic Press, 1993. 377 pp. ISBN 0-12-464255-1.

■ Miikkulainen, Risto. *Subsymbolic Natural Language Processing: An Integrated Model of Scripts, Lexicon, and Memory*. Cambridge, Mass.: The MIT Press, 1993. 391 pp. ISBN 0-262-13290-7.

■ Mitchell, Melanie. *Anology-Making as Perception: A Computer Model*. Cambridge, Mass.: The MIT Press, 1993. 275 pp. $45.00. ISBN 0-262-13289-3.

■ Nagle, Timothy; Nagle, Janice; Gerholz, Laurie; and Eklund, Peter, editors. *Conceptual Structures: Current Research and Practice*. New York: Ellis Horwood, 1993. 643 pp. ISBN 0-13-175878-0.

■ Pereira, Luis Moniz, and Nerode, Anil, editors. *Logic Programming and Nonmonotonic Reasoning*. Cambridge, Mass.: The MIT Press, 1993. 493 pp. $35.00. ISBN 0-262-66083-0.

■ Thanassas, Dimitrios. *The Phenomenon of Commonsense Reasoning*. New York: Ellis Horwood, 1993. 254 pp. ISBN 0-13-663634-9.

■ Walton, Paul, and Maiden, Neil, editors. *Integrated Software Reuse: Management and Techniques*. Brookfield, Vt.: Ashgate Publishing Company, 1993. ISBN 1-85742-146-9.

■ Wexelblat, Alan, editor. *Virtual Reality: Applications and Explorations*. San Diego, Calif.: Academic Press, 1993. 245 pp. ISBN 0-12-745045-9.

■ Wittig, Thies, editor. *ARCHON: An Architecture for Multi-Agent Systems*. New York: Ellis Horwood, 1993. 134 pp.