The second improvement upon [13] is based upon the observation that an α'-sequence may be used to check the final state of a transition. This property is utilized, in the generation of checking sequences, to allow overlap between the α'-sequences and the test segments. This further contributes to a reduction in the length of the checking sequence.

The method given in this paper might be further enhanced in two ways. First, the connecting transitions might be chosen from the set of transitions of the given FSM M during optimization, rather than being drawn from a cycle-free subset (E'') found prior to optimization. This may be achieved by including a copy of each transition and relying upon properties of the optimization algorithm, which starts with the production of a minimal symmetric augmentation, that guarantee that the set chosen is cycle free. Second, prefixes of the distinguishing sequence may be used to recognize states.

ACKNOWLEDGMENTS

This work is supported in part by the Natural Sciences and Engineering Research Council of Canada under grant OGP0000976. The authors wish to thank the anonymous referees for their comments and suggestions.

REFERENCES

[1] A.V. Aho, A.T. Dahbura, D. Lee, and M.U. Uyar, "An Optimization Technique for Protocol Conformance Test Sequence Generation Based on UIO Sequences and Rural Chinese Postman Tours," *IEEE Trans. Comm.*, vol. 39, pp. 1604-1615, 1991.

[2] F. Belina and D. Hogrefe, "The CCITT-Specification and Description Language SDL," *Computer Networks and ISDN Systems*, vol. 16, pp. 311-341, 1989.

[3] S. Budkowski and P. Dembinski, "An Introduction to ESTELLE: A Specification Language for Distributed Systems," *Computer Networks and ISDN Systems*, vol. 14, pp. 3-23, 1987.

[4] A.T. Dahbura, K.K. Sabnani, and M.U. Uyar, "Formal Methods for Generating Protocol Conformance Test Sequences," *Proc. IEEE*, vol. 78, pp. 1317-1325, 1990.

[5] A. Gill, *Introduction to the Theory of Finite-State Machines.* New York: McGraw-Hill, 1962.

[6] G. Gonenc, "A Method for the Design of Fault Detection Experiments," *IEEE Trans. Computers*, vol. 19, pp. 551-558, 1970.

[7] D. Harel, "Statecharts: A Visual Formalism for Complex Systems," *Science of Computer Programming*, vol. 8, pp. 231-274, 1987.

[8] F.C. Hennie, "Fault Detecting Experiments for Sequential Circuits," *Proc. Fifth Ann. Symp. Switching Circuit Theory and Logical Design*, pp. 95-110, 1964.

[9] D. Lee and M. Yannakakis, "Testing Finite State Machines: State Identification and Verification," *IEEE Trans. Computers*, vol. 43, pp. 306-320, 1994.

[10] D. Lee and M. Yannakakis, "Principles and Methods of Testing FSMs: A Survey," *Proc. IEEE*, vol. 84, pp. 1089-1123, 1996.

[11] I. Pomeranz and S.M. Reddy, "Test Generation for Multiple State-Table Faults in Finite-State Machines," *IEEE Trans. Computers*, vol. 46, pp. 783-794, 1997.

[12] D.P. Sidhu and T.K. Leung, "Formal Methods for Protocol Testing: A Detailed Study," *IEEE Trans. Software Eng.*, vol. 15, pp. 413-426, 1989.

[13] H. Ural, X. Wu, and F. Zhang, "On Minimizing the Lengths of Checking Sequences," *IEEE Trans. Computers*, vol. 46, pp. 93-99, 1997.

[14] M. Yannakakis and D. Lee, "Testing Finite State Machines: Fault Detection," *J. Computer and System Sciences*, vol. 50, pp. 209-227, 1995.

Computing the Shortest Network under a Fixed Topology

Guoliang Xue, *Senior Member*, *IEEE*, and K. Thulasiraman, *Fellow*, *IEEE*

Abstract—We show that, in any given uniform orientation metric plane, the shortest network interconnecting a given set of points under a fixed topology can be computed by solving a linear programming problem whose size is bounded by a polynomial in the number of terminals and the number of legal orientations. When the given topology is restricted to a Steiner topology, our result implies that the Steiner minimum tree under a given Steiner topology can be computed in polynomial time in any given uniform orientation metric with λ legal orientations for any fixed integer $\lambda \geq 2$. This settles an open problem posed in a recent paper [3].

Index Terms—Steiner trees, shortest network under a fixed topology, uniform orientation metric plane, linear programming.

1 INTRODUCTION

LET $p_1, p_2, \ldots, p_n$ be n terminal points (whose locations are fixed) in a plane with distance function d and $p_{n+1}, p_{n+2}, \ldots, p_{n+m}$ be m Steiner points (whose locations are to be determined) in the same plane. A topology for these terminal and Steiner points is a graph $\mathcal{T} = (V, E)$, where $V = \{v_1, v_2, \ldots, v_{n+m}\}$ is the set of *ordered* vertices (with v_i corresponding to p_i) and E is the set of undirected edges. A network T under topology $\mathcal{T}$ is obtained by mapping vertex v_i to location $l^{[i]}$ such that $l^{[i]} = p_i$ for $i = 1, 2, \ldots, n$. The cost of a network T is the sum of edge costs where the cost of each edge is measured using the distance between the locations of its two end vertices:

$$cost(T) = \sum_{(v_i,v_j)\in E} d(l^{[i]}, l^{[j]}). \tag{1}$$

The shortest network under topology $\mathcal{T}$ is a network T under topology $\mathcal{T}$ with the minimum possible cost. A shortest network under a given topology $\mathcal{T}$ can be obtained by finding the *optimal locations* of the Steiner points which correspond to an optimal solution to the following optimization problem:

$$\min_{l^{[n+1]},\ldots,l^{[n+m]}} \sum_{(v_i,v_j)\in E} d(l^{[i]}, l^{[j]}). \tag{2}$$

Problem (2) has been studied extensively under the name *multifacility location problem* (see [1], [2], [12], [18], [19], [22], [23], [24] and the references therein). It also has important applications in the computation of Steiner minimum trees when $\mathcal{T}$ is a tree graph where the degree of v_k is less than or equal to 3 for $k = 1, 2, \ldots, n$ and exactly 3 for $k > n$. Such a topology is called a Steiner topology and a shortest network under a Steiner topology $\mathcal{T}$ is called a Steiner minimum tree under topology $\mathcal{T}$. The Steiner tree problem with Euclidean and rectilinear distances has attracted much attention due to its applications in telecommunications and in design of printed circuit boards [4], [5], [7], [8], [9], [10], [13].

- *G. Xue is with the Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287-5406. E-mail: xue@cse.asu.edu.*
- *K. Thulasiraman is with the School of Computer Science, University of Oklahoma, Norman, OK 73109. E-mail: thulasi@ou.edu.*

Manuscript received 7 Sept. 2001; revised 11 Dec. 2001; accepted 18 Jan. 2002.

For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number 114931.

The second improvement upon [13] is based upon the observation that an α'-sequence may be used to check the final state of a transition. This property is utilized, in the generation of checking sequences, to allow overlap between the α'-sequences and the test segments. This further contributes to a reduction in the length of the checking sequence.

The method given in this paper might be further enhanced in two ways. First, the connecting transitions might be chosen from the set of transitions of the given FSM M during optimization, rather than being drawn from a cycle-free subset (E'') found prior to optimization. This may be achieved by including a copy of each transition and relying upon properties of the optimization algorithm, which starts with the production of a minimal symmetric augmentation, that guarantee that the set chosen is cycle free. Second, prefixes of the distinguishing sequence may be used to recognize states.

ACKNOWLEDGMENTS

This work is supported in part by the Natural Sciences and Engineering Research Council of Canada under grant OGP0000976. The authors wish to thank the anonymous referees for their comments and suggestions.

REFERENCES

[1] A.V. Aho, A.T. Dahbura, D. Lee, and M.U. Uyar, "An Optimization Technique for Protocol Conformance Test Sequence Generation Based on UIO Sequences and Rural Chinese Postman Tours," *IEEE Trans. Comm.*, vol. 39, pp. 1604-1615, 1991.

[2] F. Belina and D. Hogrefe, "The CCITT-Specification and Description Language SDL," *Computer Networks and ISDN Systems*, vol. 16, pp. 311-341, 1989.

[3] S. Budkowski and P. Dembinski, "An Introduction to ESTELLE: A Specification Language for Distributed Systems," *Computer Networks and ISDN Systems*, vol. 14, pp. 3-23, 1987.

[4] A.T. Dahbura, K.K. Sabnani, and M.U. Uyar, "Formal Methods for Generating Protocol Conformance Test Sequences," *Proc. IEEE*, vol. 78, pp. 1317-1325, 1990.

[5] A. Gill, *Introduction to the Theory of Finite-State Machines.* New York: McGraw-Hill, 1962.

[6] G. Gonenc, "A Method for the Design of Fault Detection Experiments," *IEEE Trans. Computers*, vol. 19, pp. 551-558, 1970.

[7] D. Harel, "Statecharts: A Visual Formalism for Complex Systems," *Science of Computer Programming*, vol. 8, pp. 231-274, 1987.

[8] F.C. Hennie, "Fault Detecting Experiments for Sequential Circuits," *Proc. Fifth Ann. Symp. Switching Circuit Theory and Logical Design*, pp. 95-110, 1964.

[9] D. Lee and M. Yannakakis, "Testing Finite State Machines: State Identification and Verification," *IEEE Trans. Computers*, vol. 43, pp. 306-320, 1994.

[10] D. Lee and M. Yannakakis, "Principles and Methods of Testing FSMs: A Survey," *Proc. IEEE*, vol. 84, pp. 1089-1123, 1996.

[11] I. Pomeranz and S.M. Reddy, "Test Generation for Multiple State-Table Faults in Finite-State Machines," *IEEE Trans. Computers*, vol. 46, pp. 783-794, 1997.

[12] D.P. Sidhu and T.K. Leung, "Formal Methods for Protocol Testing: A Detailed Study," *IEEE Trans. Software Eng.*, vol. 15, pp. 413-426, 1989.

[13] H. Ural, X. Wu, and F. Zhang, "On Minimizing the Lengths of Checking Sequences," *IEEE Trans. Computers*, vol. 46, pp. 93-99, 1997.

[14] M. Yannakakis and D. Lee, "Testing Finite State Machines: Fault Detection," *J. Computer and System Sciences*, vol. 50, pp. 209-227, 1995.

Computing the Shortest Network under a Fixed Topology

Guoliang Xue, *Senior Member*, *IEEE*, and
K. Thulasiraman, *Fellow*, *IEEE*

Abstract—We show that, in any given uniform orientation metric plane, the shortest network interconnecting a given set of points under a fixed topology can be computed by solving a linear programming problem whose size is bounded by a polynomial in the number of terminals and the number of legal orientations. When the given topology is restricted to a Steiner topology, our result implies that the Steiner minimum tree under a given Steiner topology can be computed in polynomial time in any given uniform orientation metric with λ legal orientations for any fixed integer $\lambda \geq 2$. This settles an open problem posed in a recent paper [3].

Index Terms—Steiner trees, shortest network under a fixed topology, uniform orientation metric plane, linear programming.

1 INTRODUCTION

LET $p_1, p_2, \ldots, p_n$ be n terminal points (whose locations are fixed) in a plane with distance function d and $p_{n+1}, p_{n+2}, \ldots, p_{n+m}$ be m Steiner points (whose locations are to be determined) in the same plane. A topology for these terminal and Steiner points is a graph $\mathcal{T} = (V, E)$, where $V = \{v_1, v_2, \ldots, v_{n+m}\}$ is the set of *ordered* vertices (with v_i corresponding to p_i) and E is the set of undirected edges. A network T under topology $\mathcal{T}$ is obtained by mapping vertex v_i to location $l^{[i]}$ such that $l^{[i]} = p_i$ for $i = 1, 2, \ldots, n$. The cost of a network T is the sum of edge costs where the cost of each edge is measured using the distance between the locations of its two end vertices:

$$cost(T) = \sum_{(v_i, v_j) \in E} d(l^{[i]}, l^{[j]}). \qquad (1)$$

The shortest network under topology $\mathcal{T}$ is a network T under topology $\mathcal{T}$ with the minimum possible cost. A shortest network under a given topology $\mathcal{T}$ can be obtained by finding the *optimal locations* of the Steiner points which correspond to an optimal solution to the following optimization problem:

$$\min_{l^{[n+1]}, \ldots, l^{[n+m]}} \sum_{(v_i, v_j) \in E} d(l^{[i]}, l^{[j]}). \qquad (2)$$

Problem (2) has been studied extensively under the name *multifacility location problem* (see [1], [2], [12], [18], [19], [22], [23], [24] and the references therein). It also has important applications in the computation of Steiner minimum trees when $\mathcal{T}$ is a tree graph where the degree of v_k is less than or equal to 3 for $k = 1, 2, \ldots, n$ and exactly 3 for $k > n$. Such a topology is called a Steiner topology and a shortest network under a Steiner topology $\mathcal{T}$ is called a Steiner minimum tree under topology $\mathcal{T}$. The Steiner tree problem with Euclidean and rectilinear distances has attracted much attention due to its applications in telecommunications and in design of printed circuit boards [4], [5], [7], [8], [9], [10], [13].

- *G. Xue is with the Department of Computer Science and Engineering, Arizona State University, Tempe, AZ 85287-5406. E-mail: xue@cse.asu.edu.*
- *K. Thulasiraman is with the School of Computer Science, University of Oklahoma, Norman, OK 73109. E-mail: thulasi@ou.edu.*

Manuscript received 7 Sept. 2001; revised 11 Dec. 2001; accepted 18 Jan. 2002.
For information on obtaining reprints of this article, please send e-mail to: tc@computer.org, and reference IEEECS Log Number 114931.