A Non-Composite Self-Stabilizing Algorithm For 3-Edge-Connectivity

Abusayeed M. Saifullah
Computer and Inf. Science and Engineering
University of Florida
Gainesville, Florida, USA
Email: asms@cise.ufl.edu

Yung H. Tsin
School of Computer Science
University of Windsor
Windsor, Ontario, Canada
Email: peter@uwindsor.ca

November 17, 2007

ABSTRACT

A *self-stabilizing algorithm* is a distributed algorithm that can start from any initial (legitimate or illegitimate) state and eventually converge to a legitimate state in finite time without being assisted by any external agent. In this paper, we propose a self-stabilizing algorithm for finding the 3-edge-connected components of an asynchronous distributed computer network. It is based on a self-stabilizing depth-first search algorithm and is not composed of a number of self-stabilizing algorithms that run concurrently. The algorithm stabilizes in $O(dn\Delta)$ rounds and every processor requires $O(n \log \Delta)$ bits, where $\Delta(\leq n)$ is an upper bound on the degree of a node, $d(\leq n)$ is the diameter of the communication graph, and n is the total number of nodes in the graph. Both time complexity and space complexity are same as those of the self-stabilizing depth-first search algorithm. The result of the computation is kept in a distributed fashion in the sense that, upon stabilization of the algorithm, each processor is assigned a component identifier which uniquely identifies the 3-edge-connected component to which the processor belongs.

KEY WORDS

Distributed system, fault-tolerance, self-stabilization, depth-first search tree, cut-pair, 3-edge-connected component.