The Problem of Finding
Two Edge-Disjoint Hamiltonian Cycles

Anna Gorbenko
Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
gorbenko.ann@gmail.com

Vladimir Popov
Department of Intelligent Systems and Robotics
Ural Federal University
620083 Ekaterinburg, Russia
Vladimir.Popov@usu.ru

Abstract
In this paper we consider an approach to solve the problem of finding two edge-disjoint Hamiltonian cycles. This approach is based on constructing logical models for the problem.

Keywords: Hamiltonian cycle, logical model, satisfiability problem, NP-complete

Many different topologies have been proposed for interconnection networks (see e.g. [1]). In particular, there are a number of important applications of the two edge-disjoint Hamiltonian cycles problem (2HC) in interconnection networks and parallel computing (see e.g. [2]). Note that if the network can be decomposed into edge-disjoint Hamiltonian cycles, then the message traffic will be evenly distributed across all communication links. Also, edge-disjoint Hamiltonian cycles form the basis of an efficient all-to-all broadcasting algorithm (see e.g. [3]). Note that problems of object recognition are among the most rapidly developing areas of modern computer science (see e.g. [4] – [6]). There are some applications of 2HC in this area (see e.g. [7]). Also, 2HC can be used for creation of self-testable and path-aware systems (see e.g. [8, 9]). Therefore, as a potential application of 2HC, we can mention creation of robot self-aware systems (see e.g. [10] – [12]). In this paper, we consider simple undirected graphs. Two Hamiltonian cycles in a graph are said to be edge-disjoint if they do not share any common edge.
The two edge-disjoint Hamiltonian cycles problem (2HC):

Instance: A graph $G = (V, E)$ where $V$ is the set of nodes and $E$ is the set of edges.

Question: Are there two edge-disjoint Hamiltonian cycles in $G$?

2HC is NP-complete [13]. However, there are a number of approximation efficient algorithms for the problem (see e.g. [14]). Encoding problems as Boolean satisfiability (see e.g. [15] – [20]) and solving them with very efficient satisfiability algorithms (see e.g. [21] – [23]) has recently caused considerable interest. In this paper we consider an approach to solve 2HC. This approach is based on constructing logical models for the problem.

Let $V = \{v_1, v_2, \ldots, v_n\}$. Let

$$
\varphi[1] = \land_{1 \leq i \leq n} \lor_{1 \leq j \leq n} x[i, j],
$$

$$
\varphi[2] = \land_{1 \leq i \leq n} \land_{1 \leq j[1] < j[2] \leq n} (\neg x[i, j[1]] \lor \neg x[i, j[2]]),
$$

$$
\varphi[3] = \land_{1 \leq i \leq n} \land_{1 \leq i[1] < i[2] \leq n} (\neg x[i[1], j] \lor \neg x[i[2], j]),
$$

$$
\psi[1] = \land_{1 \leq i \leq n} \lor_{1 \leq j \leq n} y[i, j],
$$

$$
\psi[2] = \land_{1 \leq i \leq n} \land_{1 \leq j[1] < j[2] \leq n} (\neg y[i, j[1]] \lor \neg y[i, j[2]]),
$$

$$
\psi[3] = \land_{1 \leq i \leq n} \land_{1 \leq i[1] < i[2] \leq n} (\neg y[i[1], j] \lor \neg y[i[2], j]),
$$

$$
\delta[1] = \land_{1 \leq i < n} \land_{1 \leq j[1] \leq n, 1 \leq j[2] \leq n, (v[i], v[j]) \notin E} (\neg x[i, j[1]] \lor \neg x[i + 1, j[2]]),
$$

$$
\delta[2] = \land_{1 \leq i[1] \leq n, 1 \leq j[2] \leq n, (v[i], v[j]) \notin E} (\neg x[n, j[1]] \lor \neg x[1, j[2]]),
$$

$$
\delta[3] = \land_{1 \leq i < n} \land_{1 \leq j[1] \leq n, 1 \leq j[2] \leq n, (v[j], v[i]) \notin E} (\neg y[i, j[1]] \lor \neg y[i + 1, j[2]]),
$$

$$
\delta[4] = \land_{1 \leq j[1] \leq n, 1 \leq j[2] \leq n, (v[j], v[i]) \notin E} (\neg y[n, j[1]] \lor \neg y[1, j[2]]),
$$

$$
\rho[1] = \land_{1 \leq i < n} \land_{1 \leq j \leq n} \land_{1 \leq k \leq n} (\neg x[i, j] \lor \neg y[i, j] \lor \neg x[i + 1, k] \lor \neg y[i + 1, k]),
$$

$$
\rho[2] = \land_{1 \leq j \leq n} \land_{1 \leq k \leq n} (\neg x[n, j] \lor \neg y[n, j] \lor \neg x[1, k] \lor \neg y[1, k]),
$$

$$
\xi = (\land_{i=1}^3 \varphi[i]) \land (\land_{i=1}^3 \psi[i]) \land (\land_{i=1}^3 \delta[i]) \land (\land_{i=1}^3 \rho[i]).
$$

It is clear that $\xi$ is a CNF. It is easy to check that $\xi$ gives us an explicit reduction from 2HC to SAT.

By direct verification we can check that

$$
\alpha \iff (\alpha \lor \beta_1 \lor \beta_2) \land
(\alpha \lor \neg \beta_1 \lor \beta_2) \land
(\alpha \lor \beta_1 \lor \neg \beta_2) \land
(\alpha \lor \neg \beta_1 \lor \neg \beta_2),
$$

$$
\lor_{j=1}^l \alpha_j \iff (\lor_{j=1}^{l-1} \alpha_2 \lor \beta_1) \land
(\land_{j=1}^{l-1} (\neg \beta_i \lor \alpha_{i+2} \lor \beta_{i+1})) \land
(\neg \beta_{l-3} \lor \alpha_{l-1} \lor \alpha_l),
$$

$$
\lor_{j=1}^4 \alpha_j \iff (\lor_{j=1}^3 \alpha_2 \lor \beta_1) \land (\lor_{j=1}^3 \alpha_2 \lor \beta_1) \land
(\neg \beta_1 \lor \alpha_3 \lor \alpha_4),
$$

$$
\lor_{j=1}^4 \alpha_j \iff (\lor_{j=1}^3 \alpha_2 \lor \beta_1) \land (\lor_{j=1}^3 \alpha_2 \lor \beta_1) \land
(\neg \beta_1 \lor \alpha_3 \lor \alpha_4).
$$
The problem of finding two edge-disjoint Hamiltonian cycles where \( l > 4 \). Using relations (1) – (4) we can easily obtain an explicit transformation \( \xi \) into \( \zeta \) such that \( \xi \Leftrightarrow \zeta \) and \( \zeta \) is a 3-CNF. It is clear that \( \zeta \) gives us an explicit reduction from 2HC to 3SAT.

In papers [21] – [23] the authors considered some algorithms to solve logical models. Our computational experiments have shown that these algorithms can be used to solve the logical model for 2HC.

References


Received: September, 2012