722872

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Maximum matching with ordering constraints is NP-complete

por
Marcus Ritt

RP 362 Outubro/2009

D| Doador R$ 20,00
12|11|2009

FL 4962

UFRGS-II-PPGCC
Caixa Postal 15064 - CEP 91501-970
Porto Alegre RS BRASIL
Telefone: (051)3308-6155
Fax: (051) 3308-5576
Email: pgcc@inf.ufrgs

Teoria da computação - SBU
Gramática: grafos
grafos

CNPQ 1.03.03.00-6

Abstract

A maximum weighted matching in a graph can be computed in polynomial time. In this paper we show that a variant, where the matching has to respect additional ordering constraints between the vertices makes the problem NP-complete.

1 Introduction

To calculate a maximum matching in general, weighted graphs is one of the cornerstone algorithmic problems for which we know a polynomial solution (Edmonds, 1965) with many practical applications. Consider an undirected graph $G = (V, E)$ with edge weights $w : E \to \mathbb{Q}$. A *matching* in G is a subset $M \subseteq E$ of the edges such that every vertex in V is incident to at most one edge in M. Then, the problem is to find a matching of maximum weight in G, where the weight of the matching is the sum of its edge weights $w(M) = \sum_{e \in M} c(e)$. A formulation as a binary integer program of the problem is

$$
\begin{array}{lll}
\textbf{maximize} & \displaystyle\sum_{e \in E} w_e x_e & \\
\textbf{subject to} & \displaystyle\sum_{e \in \delta(v)} x_e \leq 1, & \forall v \in V, \\
& x_e \in \{0, 1\}, & \forall e \in E,
\end{array}
$$

where $\delta(v)$ is the set of edges incident at vertex v. Variants of the matching problem known to be NP-complete include minimal maximum matching, which is to find the smallest matching which cannot be extended to a larger matching (Garey and Johnson, 1979), and two-stage stochastic matching (Kong and Schaefer, 2006). Both, minimal maximum matching and two-stage stochastic matching can be approximated within a factor of two. Minimal maximum matching has been shown to be APX-hard (Yannakakis and Gavril, 1980).

Motivated by an application in mapping binary to quaternary circuits, we consider additional ordering constraints between the vertices, defined by a directed acyclic graph (DAG) on the same vertex set as the matching. In the following, we focus on the unweighted case. A matching is said to *respect the ordering constraints*, if there is no directed cycle larger than two in the graph defined by the matching itself and the edges of the DAG, where the undirected edges of the matching can be traversed in any direction. Figure 1 shows an example of an instance where the maximum matching has size two, but a maximum matching respecting ordering constraints has size one. The matching of size two shown in Figure 1(b) does not respect the ordering constraints, since it generates the directed cycle $ACDBA$.

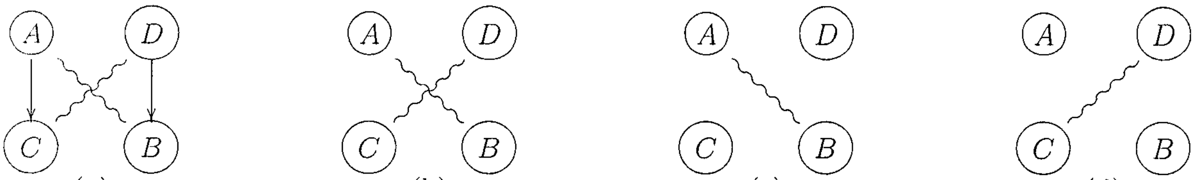

Figure 1: (a) Example for a maximum matching with ordering constraints. Arrows show dependencies between the vertices, and wavy edges show which vertices can be matched. We cannot match vertices A and B and simultaneously C and D, without violating the ordering constraints. (b) Maximum matching without ordering constraints. (c) and (d) The two possible maximum matchings respecting ordering constraints.

We can state the problem in compact form like follows:

MAXIMUM MATCHING WITH ORDERING CONSTRAINTS (MOM)

Input An undirected graph $G = (V, E)$ and a set of directed edges over $D \subseteq V \times V$ over the same vertex set.

Solution A matching $M \subseteq E$, which respects the ordering constraints given by D.

Goal Maximize the cardinality of M.

2 Completeness of MOM

In this section we consider the decision problem associated with MOM, i.e. the problem to decide if MOM has a solution larger than a given value. We are going to show that this version of MOM is NP-complete. We first show the hardness of the problem, and next its membership in NP.

Lemma 2.1
MOM is NP-hard.

Proof. The proof will be by reduction from 3-SAT. Let $\varphi = C_1 \wedge C_2 \wedge \cdots \wedge C_m$ be an instance of 3-SAT over variables $X = \{x_1, \ldots, x_n\}$ and m clauses $C_i = c_{i1} \vee c_{i2} \vee c_{i3}$ for literals c_{ij}. The idea of the reduction is to represent each variable x_i by a cycle C_{4m} (of length $4m$). This graph has a maximum matching of size $2mn$, since in each cycle at most $2m$ edges can be matched. There are exactly two possible maximum matchings, which represent the Boolean value "true" or "false" of a variable (see Figure 2). To represent the clauses, we add ordering constraints between the variables in such a way that there exists a cycle in the graph iff the clause is unsatisfied. Figure 3 shows an

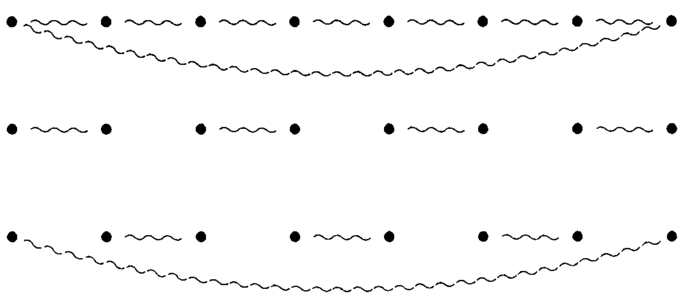

Figure 2: Representation of a variable. Top: Graph with all candidate edges. Middle and bottom: the two possible maximum matchings, representing Boolean values true and false.

example for the clause $x_1 \vee x_2 \vee \neg x_5$. Ordering constraints are added for each clause separately. Each clause is represented by four adjacent edges in the cycle, and the inner two edges are used for constructing the ordering dependencies of the clause. This makes sure that the cycles corresponding to the clauses are kept separate. Therefore, if φ has a satisfying assignment, we can construct a matching of size $2mn + m$ which respects the ordering constraints in this graph: for each cycle representing a variable choose the matching corresponding to the value of that variable. Since in each clause at least one literal is satisfied, the corresponding cycle has an unmatched edge at some variable, and therefore we can include the additional edge corresponding to the clause in the matching. Conversely, if there exists a matching of size $2nm + m$, every cycle has to contribute its maximum of $2m$ edges, and the extra edges corresponding to the clauses have to be part of matching. If we choose the value of each variable corresponding to one of the two possible matchings in each cycle, φ has to be satisfied: in each clause at least one literal is true, since otherwise the graph would have a cycle. The whole construction can be done in polynomial time. ■

Lemma 2.2
MOM is in NP.

Proof. Evidently, we can guess any subset of edges in polynomial time, and verify that it is indeed a matching. Therefore, it suffices to show that the ordering constraints can be verified in polynomial time. To check that $E \cup D$ has no directed cycle, where the edges in E can be traversed in any direction is equivalent to say that the graph $G = (V, E \cup \vec{D})$ has no strongly connected component larger than two. (We write $\vec{D} = \{(u, v) \mid \{u, v\} \in D\}$ for the set where each undirected edge of D has been replaced by a pair of directed edges in both directions.). This can be checked in time $O(|V| + |E|)$ by an algorithm which determines the strongly connected components of G (Cormen et al., 2001), and verifies that none of them is larger than two. ■

Theorem 2.1
MOM is NP-complete.

Proof. By lemma 2.1 and 2.2. ■

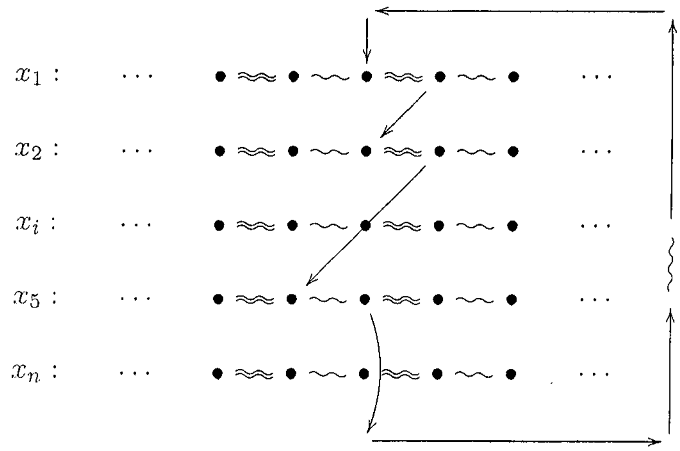

Figure 3: Representation of the clause $x_1 \vee x_2 \vee \neg x_5$. The matching with singly or double wavy edges represents the variable being true and false, respectively.

3 Conclusions

We have shown a variation of the maximum matching problem with ordering constraints to be NP-complete. This problem has an application in matching binary to quaternary circuits. It remains an open question, if we can find other application of this problem, for example in scheduling or assignments. Another open question is the approximability of this problem.

References

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms.* The MIT Press, 2nd edition, 2001.

J. Edmonds. Paths, trees, and flowers. *Canad. J. Math*, 17:449–467, 1965.

Michael R. Garey and David S. Johnson. *Computers and intractability: A guide to the theory of NP-completeness.* Freeman, 1979.

Nan Kong and Andrew J. Schaefer. A factor 1/2-approximation algorithm for two-stage stochastic matching problems. *European Journal of Operational Research*, 172(3):740 – 746, 2006.

M. Yannakakis and F. Gavril. Edge dominating sets in graphs. *SIAM J. Appl. Math*, 38:364–372, 1980.