Multi-stage Optimization of Matchings in Trees

Michal Mankowski and Mikhail Moshkov

Computer, Electrical and Mathematical Sciences & Engineering Division
King Abdullah University of Science and Technology
Thuwal 23955-6900, Saudi Arabia
{michal.mankowski, mikhail.moshkov}@kaust.edu.sa

Abstract. In this paper, we propose a method for multi-stage optimization of matchings in trees relative to different weight functions that assign positive weights to the edges of the trees. This method can be useful in transplantology where nodes of the tree correspond to pairs (donor, patient) and two nodes (pairs) are connected by an edge if these pairs can exchange kidneys. Weight functions can characterize the number of exchanges, the importance of exchanges, or their compatibility.

Keywords: tree, matching, weight function, multi-stage optimization

1 Introduction

In this paper, we consider problems of matching optimization connected with kidney transplantation [7, 12]. Let G be an indirected graph which edges and nodes have positive weights. Nodes of this graph can be interpreted as pairs (donor, patient) and two nodes $A = (a_1, a_2)$ and $B = (b_1, b_2)$ are connected by an edge if the donor a_1 can donate a kidney to the patient b_2, and the donor b_1 can donate a kidney to the patient a_2. The weight of a node A can be interpreted as an importance of the transplantation for the patient from the pair A. The weight of an edge connecting nodes A and B can be interpreted as a compatibility of the exchange of kidneys between the pairs A and B.

A matching in G is a set of edges without common nodes. We consider three optimization problems connected with matchings: (c) maximization of the cardinality of a matching, (n) maximization of the sum of weights of nodes in a matching, and (e) maximization of the sum of edges in a matching. The considered problems can be solved in polynomial time [6, 9].

Each solution of the problem (c) allows us to help to the maximum number of patients. It is known [8] that each solution of the problem (n) is also a solution of the problem (c). The situation with the problem (e) is different: a solution of the problem (e) can be not a solution of the problem (c) (see Fig. 1).

In such a situation, it is reasonable to consider multi-stage optimization of matchings relative to different criteria, for example, to describe all matchings with maximum cardinality and later to describe among these matchings all matchings with maximum weight of edges. Another possibility is to describe
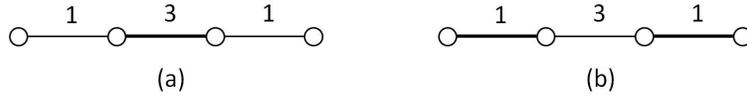
Fig. 1. Matching with maximum weight of edges (a) and matching with maximum cardinality (b)

the whole set of matchings with maximum weight of nodes and after that to describe among these matchings all matchings with maximum weight of edges.

The problem (c) can be formulated as the problem (e) when the weight of each edge is equal to 1. The problem (n) can be formulated as the problem (e) when the weight of each edge is equal to the sum of weights of its ends. So we can consider multi-stage optimization of weights of edges in matchings relative to a number of weight functions each of which assigns a positive weight to each edge of the graph G.

In this paper, we consider a dynamic programming algorithm for multi-stage optimization of matchings in trees. This algorithm can be generalized in a natural way to the forests.

The dynamic programming multi-stage optimization approach was created initially for the decision trees and decision rules [2]. One of the main areas of applications for the approach is the rough set theory [10, 11] in which decision trees and rules are widely used. This approach was extended also to some combinatorial optimization problems [1, 3–5]. Here we consider one more its application.

This paper consists of four sections. In Sect. 2, we consider a graph $D(G)$ corresponding to the tree G. We use this graph to describe the set of matchings in G and to optimize these matchings. Section 3 is devoted to the multi-stage optimization of matchings in G relative to different weight functions. Section 4 contains short conclusions.

2   Graph $D(G)$ Corresponding to Tree G

Let G be a tree. A matching in G is a set of edges without common nodes. We choose a node in the tree G as a root. It will be useful for us to consider G as a directed graph with the orientation of edges from the root. Now each node v in G defines a subtree $G(v)$ of G in which v is the root.

We describe now a graph $D(G)$ (forest of directed trees) which will be used to describe the set of matchings in G and to optimize these matchings. It contains main nodes from G and auxiliary nodes corresponding to the main ones.

Let v be a terminal node of G – see Fig. 2 (a). Then in the graph $D(G)$ there are two nodes v (main) and $v(\emptyset)$ (auxiliary) corresponding to v which are connected by an edge starting in v and entering $v(\emptyset)$ – see Fig. 2 (b).

Let v be a nonterminal node of G which has k outgoing edges $e_1, \ldots, e_k$ entering nodes $v_1, \ldots, v_k$, respectively – see Fig. 3 (a). Then in $D(G)$ there are

Fig. 2. Nodes and edge in $D(G)$ (b) corresponding to terminal node v of G (a)

the main node v, $k + 1$ auxiliary nodes $v(e_1), \ldots, v(e_k), v(\emptyset)$, and $k + 1$ edges starting in v and entering these auxiliary nodes – see Fig. 3 (b).
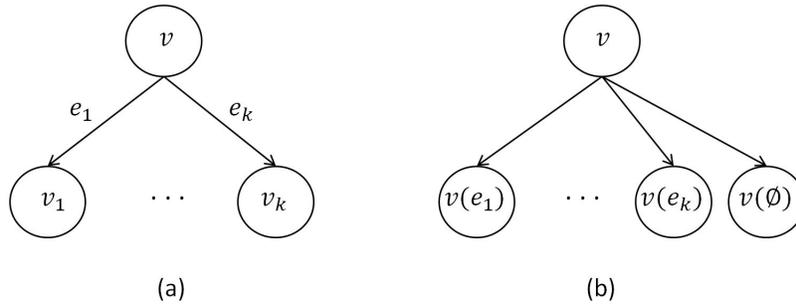

Fig. 3. Nodes and edges in $D(G)$ (b) corresponding to nonterminal node v of G (a)

A proper subgraph Δ of the graph $D(G)$ is a graph obtained from $D(G)$ by removal of some edges such that each main node in Δ has at least one outgoing edge. Let v be a main node in $D(G)$ with children $v(e_1), \ldots, v(e_k), v(\emptyset)$. We denote by $E_\Delta(v)$ the set of all $\sigma \in \{e_1, \ldots, e_k, \emptyset\}$ such that there is an edge in Δ from v to $v(\sigma)$. Proper subgraphs of the graph $D(G)$ can be obtained as results of optimization of matchings in G relative to weight functions.

Let Δ be a proper subgraph of the graph $D(G)$. We correspond a set $M_\Delta(u)$ of matchings in G to each node u of Δ. Let C and D be sets elements of which are also sets. We denote $C \otimes D = \{c \cup d : c \in C, d \in D\}$.

Let v be a terminal node of G. Then $M_\Delta(v) = M_\Delta(v(\emptyset)) = \{\lambda\}$ where λ is the empty matching. Let v be a nonterminal node of G which has k outgoing

edges $e_1, \ldots, e_k$ entering nodes $v_1, \ldots, v_k$, respectively. Then, for $i = 1, \ldots, k$,

$$M_\Delta(v(\emptyset)) = \bigotimes_{j \in \{1,\ldots,k\}} M_\Delta(v_j),$$

$$M_\Delta(v(e_i)) = \bigotimes_{j \in \{1,\ldots,k\} \setminus \{i\}} M_\Delta(v_j) \otimes M_\Delta(v_i(\emptyset)) \otimes \{\{e_i\}\},$$

$$M_\Delta(v) = \bigcup_{\sigma \in E_\Delta(v)} M_\Delta(v(\sigma)).$$

Let $\Delta = D(G)$. One can show that, for any node v of G, $M_{D(G)}(v)$ is the set of all matchings in $G(v)$ and $M_{D(G)}(v(\emptyset))$ is the set of all matchings in $G(v)$ which do not use the node v (have no edges with the end v). For any nonterminal node v of G and for any edge e starting in v, $M_{D(G)}(v(e))$ is the set of all matchings in $G(v)$ containing the edge e.

Let G contain n nodes and, therefore, $n - 1$ edges. Then the graph $D(G)$ contains $3n - 1$ nodes and $2n - 1$ edges. It is clear that the graph $D(G)$ can be constructed in linear time depending on n.

3   Multi-stage Optimization of Matchings

Let Δ be a proper subgraph of the graph $D(G)$ and w be a weight function which assigns a positive weight $w(e)$ to each edge e of G. We now describe the procedure of optimization of matchings described by Δ relative to the weight function w. During the work of this procedure, we attach a number $w_\Delta(u)$ to each node u of Δ and, may be, remove some edges from Δ.

Let v be a terminal node of G. Then $w_\Delta(v) = w_\Delta(v(\emptyset)) = 0$. Let v be a nonterminal node of G which has k outgoing edges $e_1, \ldots, e_k$ entering nodes $v_1, \ldots, v_k$, respectively. Then, for $i = 1, \ldots, k$,

$$w_\Delta(v(\emptyset)) = \sum_{j \in \{1,\ldots,k\}} w_\Delta(v_j),$$

$$w_\Delta(v(e_i)) = \sum_{j \in \{1,\ldots,k\} \setminus \{i\}} w_\Delta(v_j) + w_\Delta(v_i(\emptyset)) + w(e_i)$$

$$= w_\Delta(v(\emptyset)) - w_\Delta(v_i) + w_\Delta(v_i(\emptyset)) + w(e_i),$$

$$w_\Delta(v) = \max\{w_\Delta(v(\sigma)) : \sigma \in E_\Delta(v)\}.$$

For each $\sigma \in E_\Delta(v)$ such that $w_\Delta(v(\sigma)) < w_\Delta(v)$, we remove the edge connecting nodes v and $v(\sigma)$ from the graph Δ. We denote by Δ^w the obtained proper subgraph of the graph $D(G)$. It is clear that $E_{\Delta^w}(v) = \{\sigma : \sigma \in E_\Delta(v), w_\Delta(v(\sigma)) = w_\Delta(v)\}$ for each nonterminal node v of the tree G.

One can show that, for each node u of the graph Δ, the number $w_\Delta(u)$ is the maximum total weight of edges in a matching from $M_\Delta(u)$ relative to the weight function w, and the set $M_{\Delta^w}(u)$ is the set of all matchings from $M_\Delta(u)$ that have the total weight of edges $w_\Delta(u)$ relative to the weight function w.

Let G contain n nodes and, therefore, $n-1$ edges. For a terminal node v of the tree G, we do not need arithmetical operations to find values of $w_\Delta(v)$ and $w_\Delta(v(\emptyset))$. Let v be a nonterminal node of G which has k outgoing edges $e_1, \ldots, e_k$ entering nodes $v_1, \ldots, v_k$, respectively. We need $k-1$ additions to compute the value $w_\Delta(v(\emptyset))$, $3k$ additions and subtractions to compute the values $w_\Delta(v(e_1)), \ldots, w_\Delta(v(e_k))$, at most k comparisons to compute the value $w_\Delta(v)$, and at most $k+1$ comparisons to determine edges starting in v that should be removed. As a result, for the node v, the considered algorithm makes at most $6k$ arithmetical operations. To process the tree G, the procedure of optimization makes at most $6n$ arithmetical operations, i.e., has linear time complexity depending on n.

We can use the considered procedure for multi-stage optimization of matchings. Let we have a tree G and weight functions $w_1, w_2, \ldots$ which assign positive weights to the edges of G. We choose a node v of G as the root and construct the graph $\Delta = D(G)$. We know that the set $M_\Delta(v)$ corresponding to the node v of Δ is equal to the set of all matchings in G.

We apply to the graph Δ the procedure of optimization relative to the weight function w_1. As a result, we obtain a proper subgraph Δ^{w_1} of the graph $D(G)$. The set $M_{\Delta^{w_1}}(v)$ corresponding to the node v of Δ^{w_1} is equal to the set of all matchings from $M_\Delta(v)$ which have maximum total weight of edges relative to the weight function w_1.

We apply to the graph Δ^{w_1} the procedure of optimization relative to the weight function w_2. As a result, we obtain a proper subgraph Δ^{w_1, w_2} of the graph $D(G)$. The set of matchings $M_{\Delta^{w_1, w_2}}(v)$ corresponding to the node v of Δ^{w_1, w_2} is equal to the set of all matchings from $M_{\Delta^{w_1}}(v)$ which have maximum total weight of edges relative to the weight function w_2, etc.

In particular, we can maximize the cardinality of matchings and after that among all matchings with maximum cardinality we can choose all matchings with maximum total weight of edges.

It is easy to extend the considered approach to forests: we can apply the optimization procedures to each tree from a forest independently.

4   Conclusions

In this paper, we proposed a method for multi-stage optimization of matchings in trees relative to a sequence of weight functions. This method can be useful in transplantology if besides of the maximization of transplanted kidneys we would like to maximize the compatibility of the transplantations. In the future, we will try to generalize this method to other classes of graphs.

Acknowledgements

References

1. AbuBekr, J., Chikalov, I., Hussain, S., Moshkov, M.: Sequential optimization of paths in directed graphs relative to different cost functions. In: Sato, M., Matsuoka, S., Sloot, P.M.A., van Albada, G.D., Dongarra, J. (eds.) International Conference on Computational Science, ICCS 2011, Nanyang Technological University, Singapore, June 1-3, 2011. Procedia Computer Science, vol. 4, pp. 1272–1277. Elsevier (2011)

2. Alkhalid, A., Amin, T., Chikalov, I., Hussain, S., Moshkov, M., Zielosko, B.: Optimization and analysis of decision trees and rules: dynamic programming approach. International Journal of General Systems 42(6), 614–634 (2013)

3. Alnafie, M., Chikalov, I., Hussain, S., Moshkov, M.: Sequential optimization of binary search trees for multiple cost functions. In: Potanin, A., Viglas, T. (eds.) Seventeenth Computing: The Australasian Theory Symposium, CATS 2011, Perth, Australia, January 2011. CRPIT, vol. 119, pp. 41–44. Australian Computer Society (2011)

4. Chikalov, I., Hussain, S., Moshkov, M.: Sequential optimization of matrix chain multiplication relative to different cost functions. In: Cerná, I., Gyimóthy, T., Hromkovic, J., Jeffery, K.G., Královic, R., Vukolic, M., Wolf, S. (eds.) SOFSEM 2011: Theory and Practice of Computer Science – 37th Conference on Current Trends in Theory and Practice of Computer Science, Nový Smokovec, Slovakia, January 22-28, 2011. Lecture Notes in Computer Science, vol. 6543, pp. 157–165. Springer (2011)

5. Chikalov, I., Hussain, S., Moshkov, M., Odat, E.: Sequential optimization of global sequence alignments relative to different cost functions. In: ACM International Conference on Convergence and Hybrid Information Technology, ICHIT 2010, Daejeon, Korea, August 26-28, 2010. ACM (2010)

6. Edmonds, J.: Paths, trees, and flowers. Canadian Journal of Mathematics 17, 449–467 (1965)

7. Gentry, S.E.: Optimization over graphs for kidney paired donation. In: Lim, G.J., Lee, E.K. (eds.) Optimization in Medicine and Biology, pp. 177–195. Auerbach Publications, Taylor & Francis Group, Boca Raton, FL (2008)

8. Halappanavar, M.: Algorithms for Vertex-weighted Matching in Graphs. Ph.D. thesis, Old Dominion University, Norfolk, VA, USA (2009)

9. Lovász, L., Plummer, M.D.: Matching Theory (AMS Chelsea Publishing). Akadémiai Kiadó - North Holland, Budapest (1986)

10. Pawlak, Z.: Rough Sets – Theoretical Aspect of Reasoning About Data. Kluwer Academic Publishers, Dordrecht (1991)

11. Pawlak, Z., Skowron, A.: Rudiments of rough sets. Information Sciences 177(1), 3–27 (2007)

12. Segev, D.L., Gentry, S.E., Warren, D.S., Reeb, B., Montgomery, R.A.: Kidney paired donation and optimizing the use of live donor organs. Journal of the American Medical Association 293(15), 1883–1890 (2005)