



OPTIMIZATION ON PERMUTATIONS: RELATED STRUCTURES, PROBLEMS INTERRELATION, HEURISTIC COMPOSITIONS, APPLICATIONS

Mikalai Miatselski

Chair of Computer Science and Operations Research
University of Warmia and Mazury in Olsztyn

Received 5 October 2016, accepted 2 November 2017, available online 6 November 2017.

Key words: Heuristic algorithm, heuristics composition, optimization on permutations, assignment problem, quadratic assignment problem.

Abstract

A heuristics based approach to practical solving theoretically intractable combinatorial and network problems is discussed. Compound heuristics (heuristics compositions) are suggested to be more efficient procedures for real size problem instances. Some aspects of the heuristics compositions topic are illustrated by optimum permutation problems. We describe a uniform presentation of the chief types of the problems and their interrelations, including the relation “to be a special case of a problem”. We consider a number of algebraic structures and combinatorial constructions on permutation sets and present an inclusion chain of these constructions. The chain enables us to establish and clarify many interrelations for the minimum permutation problems, with algorithmic and complexity aspects taken into account. We also concern the applications of some problems as well.

Introduction

A majority of combinatorial and network problems which one deals with in practice of computer-aided design and manufacturing (CAD/CAM systems) especially in electronics industry and operations research are *NP*-complete or *NP*-hard. First of all, we recall here the traveling salesman problem (LAWLER,

LENSTRA 1985, REINELT 1994) and the quadratic assignment problem (CELA 1998, BURKARD, CELA 1995, BURKARD, CELA 1998). It follows from this fact that obtaining exact solutions in a reasonable time is impossible even with using modern computer systems of higher classes. Moreover, in many cases obtaining an approximate solution with an error bounded by arbitrary constant the same for all input instances is also an *NP*-complete problem. In this extremely complicated situation, it remains only using so-called heuristics, i.e. algorithms that are based on certain ideas, analogies, similarities and often on trust, hopes, intuition and personal observations of their authors, with a final assessment of the solution quality in result of substantial numeric experimentation. Below we concern an approach that is based on designing and using compound heuristics called heuristics compositions. In fact, the developers have been using particular heuristics compositions since the end of the seventies as the approach that has not any alternative when one has to be concerned with complicated combinatorial problems for actual needs of real CAD/CAM systems in electronics industry. KAMBE et al.(1982) was one of the first papers which published a chain of sequential heuristics for problems of placing standard library cells and blocks of integrated circuits developed in CAD/CAM of the SHARP Corporation. It is difficult to draw a definite conclusion whether the concrete composition presented in the paper is practically used in real design process. Our doubt can be explained by the fact that such kinds of developments are very laborious and time-consuming, connected with numerous numeric experimentations on real size instances and usually constitute confidential “know-hows” of big companies. In MIATSELSKI (2009), we formulated the general principles of such kind of approach.

Heuristics compositions

Heuristics should be computationally efficient and tested experimentally in respect of solution quality. As a rule, an individual heuristic involving single evident, sometimes naive, idea gives substitute of solution far from acceptable ones. However, this does not mean that such a heuristic should be irretrievably rejected. Experimentation with compositions of simple heuristics based on different ideas shows a significant growth of collective efficiency of such compositions in many cases. It can be also observed that the efficiency is positively influenced by the following factors:

- 1) engaging as more diverse ideas as possible by individual heuristics;
- 2) proper selection and arrangement of sequential heuristics for a composition to reach a proper trade-off between running time and solution quality;

3) determining proper numbers of iterations to be repeated or running time limits for the selected individual heuristics.

Realization of this principles is mainly based on computer experimentations with using databases of the test instances and benchmarks, public or private.

Usually (see for example REINELT 1994), two kinds of heuristics are distinguished. A heuristic of the first of kind, so-called a construction heuristic, is intended for obtaining a start approximation in seeking acceptable quality solutions. The first element of any heuristic composition is a constructive heuristic which constructs a feasible solution according to some construction rule. A heuristic of the second kind, so-called iterative or improvement heuristic is trying to improve current feasible solution using some rule of solution modification. Such modifications called moves are accepted or not according to another rule. Every attempt of modifying the current solution constitutes an individual iteration of an algorithm. Thus, an improvement heuristic is characterized by two rules: move rule and move acceptance rule. We conclude our description of simple and compound heuristics by the following formalism: a composition of heuristics is a chain

$$H_0 * H_1 * \dots * H_m,$$

where:

- H_0 – a construction heuristic,
- H_1, \dots, H_m – are improvement heuristics (not necessarily all different),
- *
- denotes the operation of consecutive running of algorithms.

Every improvement heuristic in the chain takes the result obtained by the predecessor as its input data. This result contains the recurrent approximation and possibly some additional information useful for the successors. Obviously, all the heuristics use the instance data of the problem which is being solved. In accordance with the efficiency factors stated above, a designer of a heuristics composition has to select and arrange the heuristics in the composition and also determine positive integers k_1, \dots, k_m which are iteration numbers of the corresponding improvement heuristics. Another approach may require determining the corresponding time limits t_0, t_1, \dots, t_m for the heuristics.

In MIATSELSKI (2009) optimization problems related to designing heuristics compositions are proposed. In particular, the basic problem is reduced to the problem on paths in a weighted digraph that has maximum total length in sense of weights under a restricted number of arcs. This graph problem is

solvable in time $O(n^3 \log m)$, where n is the vertex number of the digraph, m is the limit on the number of arcs. Some kind of our graph problem was considered in CHRISTOFIDES (1985).

Permutations: notation, basic notions, properties

Permutations will be the basic combinatorial objects to deal with in this paper. That is why we have to specify our notation and make some assumptions about permutations.

Let X be a finite linear ordered set, \prec be the order relationship on X , $|X|$ denote the number of members of X . Mainly, we will deal with the following cases of set X : 1) $M = \{1, 2, \dots, m\}$, $N = \{1, 2, \dots, n\}$; 2) $M \times N = \{(1, 2), (1, 3), \dots, (m, n)\}$; 3) $N^2 = \{(1, 2), (1, 3), \dots, (n, n)\}$, where the members of each set are given according to order \prec on X .

We define permutation (substitution) p to be a bijection on X and use notation taking into account the order on X . For example in case 1), we write $p = \left[\begin{array}{c} 1, 2, \dots, m \\ p_1, p_2, \dots, p_m \end{array} \right]$, but mainly we will write simply $p = (p_1, p_2, \dots, p_m)$ when no confusion can arise. The set of all permutations on X is denoted by $S[X]$, with using traditional notation S_m or S_n in case 1), S_{mn} and S_{n^2} for cases 2) and 3).

Cyclic structure $\Sigma(p)$ of permutation p is its important feature. We use symbols $\langle \rangle$ to be brackets for sub-cycles in the cyclic structure. We illustrate these notions by the examples as follows: $m = 8$, $p = \left[\begin{array}{c} 1, 2, 3, 4, 5, 6, 7, 8 \\ 7, 2, 4, 5, 8, 3, 1, 6 \end{array} \right] =$

$$= (p_1, p_2, \dots, p_8) = (7, 2, 4, 5, 8, 3, 1, 6) \quad \Sigma(p) = \langle 1, 7 \rangle \langle 2 \rangle \langle 3, 4, 5, 8, 6 \rangle.$$

$$q = \left[\begin{array}{c} 1, 2, 3, 4, 5, 6, 7, 8 \\ 3, 1, 5, 2, 7, 8, 6, 4 \end{array} \right] = (3, 1, 5, 2, 7, 8, 6, 4); \quad \Sigma(q) = \langle 1, 3, 5, 7, 6, 8, 4, 2 \rangle.$$

We would like to recall that permutations are bijections on set X and thus permutations being mappings can be composed as mappings. We accept the sequence of fulfilling the mappings on rule "from right to left" and use symbol $*$ for composition of permutations. In the context of our examples, we have the

following composition of permutations: $q * p = \left[\begin{array}{c} 1, 2, 3, 4, 5, 6, 7, 8 \\ 4, 7, 8, 2, 1, 6, 3, 5 \end{array} \right]$. It is well-

known that $S[X]$ is a group, called *the symmetric group*, operation $*$ is not commutative, thus $S[X]$ is not an Abelian group, $|S[X]| = |X|!$, for $|S_m| = m!$

We distinguish so-called cyclic permutations, i.e. such that their cyclic structure is reduced to one full cycle. We denote the set of all cyclic permutations by $C[X]$ or in case 1/ above C_m , C_n . Number of permutations $|C[X]| = (|X|-1)!$, with $|C_m| = (m-1)!$ for C_m .

Basic extreme permutation problems

Let $A = \|a_{ijkl}\|$ be a real $m \cdot m \cdot m \cdot m$ – four-index matrix, S_m denote the set of all permutations on symbols $1, 2, \dots, m$; $p = (p_1, p_2, \dots, p_m) \in S_m$ is a permutation. Define

$$F(A; p) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=m} a_{ijp_i p_j} \tag{1}$$

General Quadratic Assignment Problem (GQAP) consists in seeking a permutation $p_0 \in S_m$ such that $F(p) \geq F(p_0)$ holds for every $p = (p_1, p_2, \dots, p_m) \in S_m$. In other words, permutation p_0 provides the minimum value of objective function $F(A; p)$. In this meaning, permutation p_0 is called an optimal solution of the GQAP.

The GQAP is an intractable problem even for $m = 20-25$, with very special cases being NP-hard in strong meaning. For example, famous the Traveling Salesman Problem (TSP) is a very special case not only of the GQAP, but also of its special case well-known as the Koopmans-Beckman Problem (KBP). The Koopmans-Beckman Problem is defined as follows.

Let $C = \|c_{ij}\|$, $D = \|d_{ij}\|$, $E = \|e_{ij}\|$, be $m \cdot m$ – matrices. Find a permutation $p_0 \in S_m$ such that objective function

$$f(C, D, E; p) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=m} d_{ij} c_{p_i p_j} + \sum_{i=1}^{i=m} e_{ip_i} \tag{2}$$

attains its minimum $f(p_0)$ on S_m . It is easy to see that we obtain the TSP with matrix $C = \|c_{ij}\|$ if we set $e_{ij} = 0$ for all e_{ij} and let matrix $D = \|d_{ij}\|$ take the form:

$$\begin{bmatrix} 0 & 1 \dots & 0 \\ \vdots & \ddots & 1 \\ 1 & \dots & 0 \end{bmatrix}$$

In other words, D is the matrix representation of cyclic permutation $p = \langle 1, 2, 3, \dots, n \rangle$. Certainly, another cyclic permutation can be selected instead of p .

Notice that matrix E is often zero-matrix and the second term in (2) is omitted, i.e. the KBP takes the followings form which is called simply QAP: find $p_0 \in S_m$ that gives the minimum of objective function

$$f(C, D; p) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=m} d_{ij} c_{p_i p_j} \quad (3)$$

on S_m

We would like to emphasize an important role of the QAP with objective (3) as one of the basic models used in different fields such as logistics, allocating production units and especially in computer science for goals of placing the library cells and blocks when integrated circuits are being designed, embedding graphs, forced network clustering, encoding (numeration) network's nodes. The two latter tasks are stated as follows.

I. Forced Network Clustering (cutting, decomposition)

Given a weighted graph $G = (V, E; w: E \rightarrow R^+)$, $|V| = m$, a partition $m = \sum_{k=1}^{k=s} m_k$ find a vertex-disjoint division of graph G into subgraphs G_1, G_2, \dots, G_s with vertex numbers respectively m_1, m_2, \dots, m_s . The objective is to minimize the sum of edge weights w_{ij} of "cut" (external) edges, i.e. edges whose vertices belong to distinct subgraphs. This problem is reduced to the QAP with objective (3), where matrix C is the adjacency matrix of graph G , matrix $D = U - B$, where $U = \|u_{ij}\|$ is matrix with all its elements $u_{ij} = 1$. $B = \text{Diag} [B_1, B_2, \dots, B_s]$ is a diagonal-block matrix, where blocks B_1, B_2, \dots, B_s are quadratic sub-matrices of sizes respectively m_1, m_2, \dots, m_s filled with units.

II. Encoding network's nodes

This problem consists in assigning numbers $1, 2, \dots, m$ to vertices of graph $G = (V, E)$ in order to minimize sum $\sum_{i=1}^{i=m} \sum_{j=1}^{j=m} I p_i - p_j I$, where permutation $p = (p_1, p_2, \dots, p_m)$ presents the assigned numbers. In the corresponding QAP, matrix C is the adjacency matrix of graph G . For matrix $D = \|d_{ij}\|$, $d_{ij} = Ii - jI$.

Another version of the QAP, so-called the bottleneck QAP is obtained by substituting objective for objective function $f(C, D; p) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=m} d_{ij} c_{p_i p_j}$ for objective $g(C, D; p) = \text{MIN}_{ij=1, \dots, n} \{d_{ij} c_{p_i p_j}\}$. In the context of problems of encoding the graph vertices, we would like to emphasize that both the problems play an substantial role in operating on sparse matrices. The bottleneck QAP is actually another form of the minimum bandwidth problem for a sparse matrix.

Let us turn to the second term of (2), which actually is the value of objective in the following linear assignment problem (LAP). Given $m \cdot m$ -matrix $E = \|e_{ij}\|$, LAP consists in seeking a permutation $p_0 \in S_m$ that minimizes objective function

$$V(E; p) = \sum_{i=1}^{i=m} e_{ip_i} \tag{4}$$

on S_m .

D.A. Suprunenko (see SUPRUNENKO, MIATSELSKI 1973) introduced conception of general linear assignment problem (GLAP) in the following way. Given matrix $E = \|e_{ij}\|$, and a nonempty subset $H \subseteq S_m$, find a permutation $p_0 \in H$ that minimizes function $V(E; p) = \sum_{j=1}^{j=m} e_{ip_i}$ on H . It can be easily seen that we obtain the traveling salesman problem when $H = C_m$. D.A. Suprunenko also introduced into consideration a special case of the GLAP, so-called problem of minimizing linear form on a permutation set $H \subseteq S_m$, in fact, problem of minimizing the scalar product (MSP) of two n -vectors on $H \subseteq S_n$. That means we have one more objective function defined for two vectors $a = (a_1, a_2, \dots, a_m)$, $b = (b_1, b_2, \dots, b_m)$:

$$v(a, b; p) = \sum_{i=1}^{i=m} a_i b_{p_i} \tag{5}$$

This case is distinguished by the following well-known smart fact. Without loss of generality we can assume that ordering $a_1 \leq a_2 \leq \dots \leq a_m$ holds. Then minimum (maximum) of $v(a, b; p) = \sum a_i b_{p_i}$ on S_m is attained on a permutation $p = (p_1, p_2, \dots, p_m) \in S_m$ such that inequalities $b_{p_1} \leq b_{p_2} \leq \dots \leq b_{p_m}$ ($b_{p_1} \geq b_{p_2} \geq \dots \geq b_{p_m}$) are satisfied.

Here we indicate a few relationships among problems and their objectives given above. First of all, notice that function (4) is evidently reduced to (5) when its matrix $E = \|e_{ij}\|$ is of rank 1. Really, rank 1 implies proportionality of each row of $E = \|e_{ij}\|$ to one of them, for example, to the first column. Then we select this column to be vector a and the sequence of proportionality coefficients to be vector b for the MSP. Moreover, as it is shown in (SUPRUNENKO, MIATSELSKI 1973) the following proposition takes place:

Let $m \cdot m$ -matrix $E = \|e_{ij}\|$ of LAP have rank r , then there exist r pairs of m -vectors $a_{k,k=1,\dots,r}$, $b_{k,k=1,\dots,r}$, such that

$$V(E; p) = \sum_{k=1}^{k=r} v(a_k, b_k; p) \tag{6}$$

holds for any $p \in S_m$ and, conversely, $\sum_{k=1}^{k=s} v(a_k, b_k; p) = V(E; p)$, where $E = \|e_{ij}\|$,

$e_{ij} = \sum_{k=1}^{k=s} a_{ik} b_{jk}$ with rank of matrix E not exceeding number s .

Let us return to the GQAP with four-index matrix $A = \|a_{ijkl}\|$. We convert matrix A to its two-index representation $\tilde{A} = \|\tilde{a}_{(i,j)(kl)}\|$ which is a two-index $m^2 \cdot m^2$ -matrix, with its elements indexed by ordered set $M^2 = \{(1, 2), (1, 3), \dots, (m, m)\}$ of pairs single indices.

Proposition 1 Let matrix $\tilde{A} = \|\tilde{a}_{(i,j)(kl)}\|$ obtained as result of converting matrix A of the GQAP to a two-index matrix have rank r , then there exist r pairs of matrices $C_{k,k=1,\dots,r}, D_{k,k=1,\dots,r}$ such that

$$F(A; p) = \sum_{k=1}^{k=r} f(C_k, D_k; p)$$

holds for each $p_0 \in S_m$ and, conversely, $\sum_{k=1}^{k=s} f(C_k, D_k; p) = F(A; p)$, where rank r , of \tilde{A} does not exceed value s .

Proposition 1 can be derived from the previous proposition after some preparations including re-indexation of set $M^2 = \{(1, 2), (1, 3), \dots, (m, m)\}$ changing pair indices for single ones.

Operations over permutation sets: some algebraic structures and combinatory constructions

Our definitions of operations over permutation sets will be based on the ways to construct permutations on sets $M + N = \{1, 2, \dots, m, m + 1, m + 2, \dots, m + n\}$, $M \cdot N = \{(1, 2), (1, 3), \dots, (m, n)\}$ and $M^2 = \{(1, 2), (1, 3), \dots, (m, m)\}$ ("big permutations") by means of permutations of S_m or S_n ("building blocks", "small permutations"). These definitions come from some concepts which had been introduced under different names for permutation groups, see for example (SUPRUNENKO 1996). We present these notions defined for arbitrary permutation sets out of group theory context.

Name: Direct sum; Operation symbol: \oplus ; Constructing direct sum of single permutations: $p = (p_1, p_2, \dots, p_m) \in S_m, q = (q_1, q_2, \dots, q_n) \in S_n \xrightarrow{\text{reslt}}$ [Action on united chain $M + N, p \oplus q: i \xrightarrow{\text{def}} p_i$ for $i, 1 \leq i \leq m, p \oplus q: j \xrightarrow{\text{def}} q_j$ for $j, m + 1 \leq j \leq m + n$;] $m + n$;

Constructing direct sum of sets $G \subseteq S_m, H \subseteq S_n: G \oplus H \stackrel{\text{def}}{=} \{(p, q): p \in G, q \in H\}$. Comment: Direct sum of permutation sets $G \oplus H$ acts independently on disjoint parts of chain $M+N=\{1, 2, \dots, m, m+1, m+2, \dots, m+n\}$, $IG \oplus HI = IGI IHI$, in particular $IS_m \oplus S_n I = m!n!$

Name: Direct product; Operation symbol: \otimes ; Constructing direct product of single permutations: $p = (p_1, p_2, \dots, p_m) \in S_m, q = (q_1, q_2, \dots, q_n) \in S_n \xrightarrow{\text{reslt}}$ [Action on set $M+N, p \otimes q: (i, j) \xrightarrow{\text{def}} (p_i, q_j)$ for $i, 1 \leq i \leq m, j, 1 \leq j \leq n$;

Constructing direct product of sets $G \subseteq S_m, H \subseteq S_n: G \oplus H \stackrel{\text{def}}{=} \{(p, q): p \in G, q \in H\}$. Comment: $IG \oplus HI = IGI IHI$, in particular $IS_m \oplus S_n I = m!n!$

Name: Wreath product; Operation symbol: \wr Constructing wreath product of single permutations: $p = (p_1, p_2, \dots, p_m) \in S_m, q^1 = (q_1^1, q_2^1, \dots, q_n^1) \in S_n, \dots, q^m = (q_1^m, q_2^m, \dots, q_n^m) \in S_n \xrightarrow{\text{resit}} [Action\ on\ set\ M \cdot N; p \wr q: (i, j) \xrightarrow{\text{def}} (p_i, q_j^i)]$ for $i, 1 \leq i \leq m, j, 1 \leq j \leq n;$

Construction of wreath product of sets $G \subseteq S_m, H \subseteq S_n: G \wr H \stackrel{\text{def}}{=} \{(p; q^1, q^2, \dots, q^m): p \in G, q^i \in H, 1 \leq i \leq m\}$. Comment: $IG \wr HI = IGI IHI^{IGI}$, in particular $IS_m \wr S_n I = m!n!^m$.

Name: Diagonal of $S_m \otimes S_m$; Let $m=n$ hold. Notation and definition: $\text{diag}[S_m \otimes S_m] \stackrel{\text{def}}{=} \{(p, q): p \in S_n, q \in S_n, p = q\}$.

The following relationships for sets $G \in S_m, H \in S_n$ can be obtained on the base of constructions of permutation sets described above.

$$G \otimes H \subset G \wr H \subset S_{mn}, (m > 1, n > 1) \tag{7a}$$

$$\text{diag}[S_m \otimes S_m] \subset S_m \otimes S_m \subset S_m \wr S_m \subset S_{m^2}, m > 1 \tag{7b}$$

These inclusions enable us to detect relationships among the permutation problems described above. We will also use abbreviation \bar{S}_m for $\text{diag}[S_m \otimes S_m]$ First of all, we will consider the most general case of matrices, namely our input data are given by four-index matrix $A = \|a_{ijkl}\|$ i.e. we will be concerned with the proper (initial) GQAP which corresponds to position of $\text{diag}[S_m \otimes S_m]$ in the inclusions chain (7b). At this point, we step up in our generalizations and extend the set of feasible permutations of the GQAP permitting the permutations to belong to sets wider than $\text{diag}[S_m \otimes S_m]$, in particular to the successive terms of chain (7b). In this respect, we follow the concept of the GLAP above, with transferring the idea of the LAP to the GQAP. Thus, the next term is $S_m \otimes S_m$ or more general $S_m \otimes S_n$.

The General QAP on $S_m \otimes S_m$ has all indications to be intractable in both theoretical and practical meaning. Therefore, we restrict ourselves by the case when matrix $\tilde{A} = \|\tilde{a}_{(i,j)(kl)}\|$ obtained in result of converting matrix $A = \|a_{ijkl}\|$ of the GQAP to a two-index matrix has rank 1. It follows from proposition 1 that $F(A; p) = f(C_1, D_1; p)$ or simply $F(A; p) = f(C, D; p)$, where according to our conventions above we deals with objective function $f(C, D; p) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=m} d_{ij} c_{p_i p_j}$ or more generally $f(C, D; p, q) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=n} d_{ij} c_{p_i q_j}$ on $S_m \otimes S_m$ or $S_m \otimes S_n$ respectively. Unlike the QAP the problem of minimization $f(C, D; p) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=n} d_{ij} c_{p_i q_j}$ on $S_m \otimes S_n$ so far has attracted insufficient attention of the researches. Moreover, there is not any generally accepted name for the problem. Nevertheless, the

problem has practical applications similar to those of the QAP, however with explicit accent on bipartite weighted graphs (networks).

Let $\bar{p} \in S_m$ be an arbitrary fixed permutation. Then we have $f(C, D; \bar{p}, q)$
 $= \sum_{i=1}^{i=m} \sum_{j=1}^{j=n} d_{ij} c_{\bar{p}i, qj} = \sum_{i=1}^{i=m} \sum_{j=1}^{j=n} d_{ij} c_{\bar{p}i, qj} = \sum_{j=1}^{j=n} e_{jq}$, where $e_{jq} = \sum_{i=1}^{i=m} d_{ij} c_{\bar{p}i, qj} = V(E_{\bar{p}}; q) = \sum_{j=1}^{j=n} e_{jq}$
 and $E_{\bar{p}} = \|e_{jk}\|$ is an $n \cdot n$ -matrix.

Thus, when one of the permutations in the objective function $f(C, D; \bar{p}, q)$ is fixed we actually deal with the LAP. In respect of computational complexity, there is a principal difference between the QAP and the LAP: the first is NP-hard while the other is efficiently solvable in time $O(n^3)$. For this reason, it seems to be promising for solving the QAP to come down it to a finite not too numerous series of LAPs. The following algorithm exploits this approach.

Algorithm Min-on-direct-product

Input: $C = \|c_{ij}\|$, $D = \|d_{ij}\|$ /* $n \cdot n$ -matrices;

Output: $p_0 \in S_m$, $q_0 \in S_n$ /* a pair of permutations which is accepted to be an approximate solution to the problem of minimizing objective function $f(C, D;$

$p, q) = \sum_{i=1}^{i=m} \sum_{j=1}^{j=n} d_{ij} c_{p_i, q_j}$ on $S_m \otimes S_n$ */;

Main Program:

1) Select: $x \in S_m$; $y \in S_n$; /* Selection of initial permutations;

Do

{2) find y^\wedge that minimize $f(C, D; x, q)$ on $\{x\} \otimes S_n$ while x is fixed; /*in fact, solving a LAP;

2a) $y = y^\wedge$;

3) find x^\wedge that minimize $f(C, D; p, y)$ on $S_m \otimes \{y\}$ while y is fixed; /*in fact, solving a LAP;

3a) $x = x^\wedge$;

Until $x^\wedge = x$ & $y^\wedge = y$;

4) $p_0 := x^\wedge$; $q_0 := y^\wedge$; /* The result has been obtained, further improvement of the current solution in this way is impossible; */

End of Algorithm

This algorithm may be used to be a part of heuristics compositions, both as a construction heuristic and improving one. Obviously, algorithm Min-on-direct-product can be modified by introduction of random repeated choices $x \in S_m$; $y \in S_n$ to instruction 1) intended for selection of initial permutations. The final solution is the best of (p_0, q_0) obtained in result of many trials $x \in S_m$; $y \in S_n$.

Conclusions and future work

In this paper, we have concerned a unified approach to compound heuristics and uniform presentation of the basic extreme permutations problems as well as related structures. This constitutes a basis for further research in connection with the general assignment problem on the wreath product in inclusion chain $(7a, b)$ and using metrics on the symmetric group for seeking heuristic solutions.

References

- BURKARD R.E., CELA E., DEMIDENKO V.M., MIATSELSKI M., WOEGINGER G.J. 1995. *A unified approach to simple special cases of extremal permutation problems*. Optimierung und Kontrolle. K-FU Graz, TUGraz, p 12.
- BURKARD R.E., CELA E., DEMIDENKO V.M., MIATSELSKI M., WOEGINGER G.J. 1998. *A unified approach to simple special cases of extreme permutation problems*. Optimization, 44: 123–138.
- CELA E. 1998 *The quadratic assignment problem: Theory and algorithms*. Kluwer Academic Publishers, Dordrecht, The Netherlands, p. 286.
- CHRISTOFIDES N. 1985. *Graph theory: an algorithmic approach*. Academic Press, p 428.
- KAMBE T., KIMURA S., INUFUSHI T., OKUDA N., NISHIOKA I. 1982. *A Placement Algorithm for Polycell LSI and its Evaluation*. Design Automation, 19th Conference, Las Vegas, p. 123–132.
- LAWLER E.L., LENSTRA J.K., RINNOOY KAN A.H.G., SHMOYS D.B. 1985. *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization*. Wiley.
- MIATSELSKI M. 2009. *Compound heuristics as means for practical solving theoretically hard problems coming from computer-aided design*. Annals of Higher School of Computer Science and Economics, Olsztyn, p. 37–46.
- REINELT G. 1994. *The traveling salesman problem*. Lecture Notes in Computer Science, 840: 324.
- SUPRUNENKO D.A. 1996. *Permutations groups*. Nauka i Technika, p. 366.
- SUPRUNENKO D.A., MIATSELSKI M.M. 1975. *The assignment problem and minimization of the sum of linear forms on the symmetric group*. Cybernetics and Systems, 3: 64–68.