Multi-objective Quadratic Assignment Problem instances generator with a known optimum solution

Mădălina M. Drugan

Artificial Intelligence lab, Vrije Universiteit Brussel, Pleinlaan 2, B-1050 Brussels, Belgium
mdrugan@vub.ac.be

Abstract. *Multi-objective quadratic assignment problems* (mQAPs) are NP-hard problems that optimally allocate facilities to locations using a distance matrix and several flow matrices. mQAPs are often used to compare the performance of the multi-objective meta-heuristics. We generate large mQAP instances by combining small size mQAP with known local optimum. We call these instances *composite mQAPs*, and we show that the cost function of these mQAPs is additively decomposable. We give mild conditions for which a composite mQAP instance has known optimum solution. We generate composite mQAP instances using a set of uniform distributions that obey these conditions. Using numerical experiments we show that composite mQAPs are difficult for multi-objective meta-heuristics.

1 Introduction

The Quadratic assignment problem (QAP) models many real-world problems like the computer-aided design in the electronics industry, scheduling, vehicle routing, etc. Intuitively, QAPs can be described as the (optimal) assignment of a number of facilities to a number of locations. In general, QAP instances are NP hard problems, and QAP instances are often included in the benchmarks for testing meta-heuristics [1, 2]. Special cases of QAPs solvable in polynomial time are easy to solve [3]. Meta-heuristic search algorithms based on local search are especially useful for large size QAPs, where exact solutions are difficult to obtain. Furthermore, measuring the performance of meta-heuristics is best done when the optimum solution for the test problem is known.

Generating large size QAPs with known local optimum solutions that are difficult and interesting for exact and stochastic algorithms is a current challenge in the field [4, 5]. The algorithms that generate large and hard single objective QAP instances with known optima [6] are rather elaborated and difficult to generate. Drezner et al [4] propose QAP instances that are difficult to solve with heuristics but easy for exact solvers because of the large amount of 0's in the flow matrix.

Recently, Drugan [5] proposes a single objective QAP instance generator with additively decomposable cost function and known local optimum. Problems with additively decomposable cost functions are considered useful test benchmark for meta-heuristic algorithms that explore the structure of the search space. These QAP instances are difficult for both exact methods, like branch and bound, and for meta-heuristics.

Multi-objective Quadratic assignment problems [7] are an extension of QAP with more than two flow matrices. Let us consider N facilities, the $N \times N$ distance matrix $A = (a_{ij})$, where a_{ij} is the distance between location i and location j. Consider an mQAP with m flow matrices $\mathbf{B} = (B^1, \ldots, B^m)$, where $m \leq 2$ and $B^o = (b^o_{ij})$ and b^o_{ij} represents the k-th flow matrix from facility i to facility j. The goal is to minimise the *cost function* in all objectives o

$$c^o(\pi) = \sum_{i=1}^{N} \sum_{j=1}^{N} a_{ij} \cdot b^o_{\pi_i \pi_j} \tag{1}$$

where $1 \leq o \leq m$ and π is a permutation of N facilities and π_i is the i-th element of π. It takes quadratic time to evaluate each of these functions. We consider an mQAP as a tuple $(A, B^1, \ldots, B^m, s)$ where s, if known, is the optimum solution.

The main contribution. We design a multi-objective QAP instance generator that creates meaningful, i.e. large and difficult to solve, benchmark instances for multi-objective meta-heuristics [2, 8]. Our solution introduced in Section 2 is to aggregate several flow and distance matrices with computable optimum solutions, into a larger mQAP such that the optimum of the resulting mQAP is known, called *composite mQAPs*. These mQAPs have additively decomposable cost functions that are the sum of component mQAP's cost functions plus an extra term corresponding to the cost of the region outside these component mQAPs.

In Section 3, we give mild conditions, e.g. upper and lower bounds for the values in the mQAPs matrices such that the composite mQAP instance has the identity permutation as the global optimum solution. However, to verify the global optimum solution we compute a large number of cost functions equivalent with the number of permutations of the component mQAP instances into the permutation of the composite mQAP instance.

In order to simplify the procedure of generating composite mQAPs with known global optima, we consider uniform distributions which are also used to generate other mQAPs from the literature [2, 7]. In Section 4, the conditions on the upper and lower bounds are easily verifiable, and explicit numerical values are proposed.

Numerical experiments from Section 5 show that the composite mQAPs are difficult to solve with multi-objective meta-heuristic instances [8] when compared with the other mQAPs from literature [7]. We show that the global optimum is difficult to attain and thus composite mQAPs are difficult to solve. Section 6 concludes the paper.

2 Composite multi-objective QAP instances generator

In this section, we design an algorithm that generates *composite mQAP* instances from small size *component mQAP* instances with computable optimum solution. The values in the composite mQAP not assigned yet are also selected to have known optimum value. Thus, there are three optimisation problems in composite mQAPs: i) optimising the component mQAPs, ii) optimising the region outside these components, and iii) a

Algorithm 1 generate_composite_mQAP

Require: d component mQAP instances $\{(A_1, B_1^1, \ldots, B_1^m, \mathcal{I}), \ldots, (A_d, B_d^1, \ldots, B_d^m, \mathcal{I})\}$
Require: the distributions in the outside region $\mathcal{R}_A$, $\mathcal{R}_{B^1}$, ..., $\mathcal{R}_{B^m}$: the low values distributions $\mathcal{L}_A$, $\mathcal{L}_{B^1}$, ..., $\mathcal{L}_{B^m}$, and the high values distributions $\mathcal{H}_A$, $\mathcal{H}_{B^1}$, ..., $\mathcal{H}_{B^m}$

```
/* I. Aggregate mQAP instances/*
Initialise A, B^1, B^2, ..., B^m with 0s everywhere
for all k = 1 to d do
    for all i, j = 1 to n_k do
        t ← i + Σ_{r=1}^{k} n_r; p ← j + Σ_{r=1}^{k} n_r;
        a_tp ← a_tp + a_kij; b^1_tp ← b^1_tp + b^1_kij; ...; b^m_tp ← b^m_tp + b^m_kij;
    end for
end for
/* II. Generate the set of elements in A, B^1, ..., B^m not assigned yet /*
for all α% elements a_ij ∈ R_A, b^1_ij ∈ R_{B^1}, ..., b^m_ij ∈ R_{B^m} do
    Generate a_ij ∝ H_A, and update the sorted list R_A ← R_A ∪ a_ij
    Generate b^o_t ∝ L_{B^o}, and update the sorted list R_{B^o} ← R_{B^o} ∪ b^o_t, for all o ≤ m
    t ← t + 1
end for
for all (1 − α)% elements a_ij ∈ R_A, b^1_ij ∈ R_{B^1}, ..., b^m_ij ∈ R_{B^m} do
    Generate a_ij ∝ L_A, and update the sorted list R_A ← R_A ∪ a_ij
    Generate b^o_t ∝ H_{B^o}, and update the sorted list R_{B^o} ← R_{B^o} ∪ b^o_t, for all 1 ≤ o ≤ m
    t ← t + 1
end for
for all r = 1 to |R_A| do
    r ← rank of a_ij in R_A
    b^o_ij ← b^o_t with rank |R_A| − r in R_{B^o}, for all o ≤ m
end for
return (A, B^1, ..., B^m)
```

global optimisation problem for the entire mQAP. The pseudo-code for this algorithm is given in Algorithm 1.

The algorithm generate_composite_mQAP has as input d component mQAP instances, $(A_k, B_k^1, \ldots, B_k^m, \mathcal{I})$, $\forall k \leq d$, with identity permutation $\mathcal{I}$ as optimum solution, where $\forall i \in \{1, \ldots, N\}, \mathcal{I}_i = i$. In order to calculate the optimum solution of component mQAPs, we could, for example, exhaustively enumerate all possible permutations. A straightforward method to transform a component mQAP with an optimum solution s into an mQAP instance with the identity permutation as optimum solution is to rename the facilities.

2.1 Aggregate component mQAP instances

For simplicity, we consider that each facility from the composite mQAP corresponds to exactly one facility from a single component mQAP, and, vice-versa, each facility from a component mQAP corresponds to exactly one facility from the composite mQAP. We consider that n_k are the number of facilities of the k-th component mQAP, $(A_k, B_k^1, \ldots, B_k^m, \mathcal{I})$. We call the reunion of all component mQAPs the *component re-*

gion. Note that the number of facilities N for the newly generated composite mQAP is the sum of the number of facilities of the component mQAP, $N = \sum_{k=1}^{d} n_k$.

For each pair of facilities in the k-th component mQAP $(i, j) \in A_k$, there is assigned a pair of facilities in the composite mQAP $(t, p) \in A$, where $t \leftarrow i + \sum_{r=1}^{k} n_r$ and $p \leftarrow j + \sum_{r=1}^{k} n_r$. We update the values $a_{tp} \in A$ and $b_{tp}^{o} \in B^o$ with the corresponding values in $a_{kij} \in A_k$ and $b_{kij}^{o} \in B_k^o, \forall o \leq m$.

2.2 Filling up the composite mQAP instances

Next, we assign the positions in A and $\mathbf{B}$ not assigned yet. Let $\mathcal{R}_A$ and $\mathcal{R}_{B^o}$ be ordered sets containing all unassigned values from A, and B^o, respectively. We call these sets, the *outside region* of the corresponding matrices. The elements in the outside region are generated using the rearrangement inequality [9] [1] such that their cost function has the identity permutation as the optimum solution. Informally, the largest values in the o-th flow matrix B^o correspond to the lowest values in the distance matrix A, and the lowest values in B^o correspond to the largest values in A.

The low values distributions $\mathcal{L}_A$ and $\mathcal{L}_{B^o}$ generate the lowest values of A and B^o, respectively. The high values distributions $\mathcal{H}_A$ and $\mathcal{H}_{B^o}$ generate the highest values of A and B^o. We generate $\alpha\%$ unassigned values in A from $\mathcal{H}_A$ and $(1-\alpha)\%$ from $\mathcal{L}_A$. Because of the rearrangement inequality, $\alpha\%$ values in each of the flow matrices B^o are generated from $\mathcal{L}_{B^o}$ and $(1-\alpha)\%$ are generated from $\mathcal{H}_{B^o}$.

In Algorithm 1, let r be the rank of a_{ij} in $\mathcal{R}_A$. If a_{ij} is generated from $\mathcal{H}_A$, then each value b_{ij}^o is generated from $\mathcal{L}_{B^o}$ such that the rank of b_{ij}^o in $\mathcal{R}_{B^o}$ is $|\mathcal{R}_A| - r$. Similarly, if a_{ij} is generated from $\mathcal{L}_A$, then b_{ij}^o is generated from $\mathcal{H}_{B^o}$ such that the rank of b_{ij}^o in $\mathcal{R}_{B^o}$ is $|\mathcal{R}_A| - r$. Thus, the elements b_{ij}^1, ..., b_{ij}^m have the same ranking in the outside regions of the corresponding flow matrices, B^1, ..., B^m.

3 Designing composite mQAPs with known optimum solution

Cela [3] showed that single QAP instances where all the elements obey the rearrangement inequality are *easy*. This means that if the component mQAPs are degenerated, $n_1 = \ldots = n_d = 1$, then the composite mQAP also becomes "easy". Thus, we consider the component mQAPs to be the "difficult" region, and the outside region to be the "easy" region of a composite mQAP. By design, the component mQAPs and the outside region are optimised by the identity permutation. The composite mQAP, in general, is not optimised by the identity permutation.

In this section, we give mild conditions under which the composite mQAP instances have the identity permutation as the optimum solution. We consider that all the elements in the outside region are either smaller or larger than all the elements in the component mQAPs. Accordingly to the rearrangement inequality, if elements are exchanged between the component mQAPs and the outside region, then the cost of the composite mQAP instance increases.

[1] Let n variables be generated with any two distributions $\{x_1, \ldots, x_n\}$ and $\{y_1, \ldots, y_n\}$ for which $x_1 \leq \ldots \leq x_n$ and $y_1 \geq \ldots \geq y_n$. The rearrangement inequality states that $\sum_{i=1}^{n} x_i \cdot y_i \leq \sum_{i=1}^{n} x_i \cdot y_{\pi_i}$, for all permutations π.

Additively decomposable cost functions for the composite mQAPs. In the following, we show that the composite mQAP instances have *additively decomposable cost functions* with a residual term representing the cost of the outside region.

Consider the set $\Pi(N)$ of all permutations of N facilities in the flow matrices. In the permutation group theory, permutations are often written in the cyclic form. If π is a permutation of facilities, we can write it as $\pi = (\pi_1, \ldots, \pi_d)$, where π_k is the k-th cycle containing a set of facilities that can be swapped with each other. These cycles are disjoint subsets. We consider d cycles, each cycle contains the facilities of exactly one component mQAP. If there are n_k facilities in the k-th component mQAP, the corresponding cycle is a n_k-cycle. The cost function of the k-th cycle is

$$c_k^o(\pi) = \sum_{i,j,\pi_i,\pi_j} a_{kij} \cdot b^o_{k\pi_i\pi_j} \tag{2}$$

where $k \in \{1, \ldots, d\}$, d is the number of component QAPs and a_{kij} is an element of the k-th component QAP. Similarly, $b^o_{k\pi_i\pi_j}$ is an element of the k-th component QAP. By design, the optimal cost for each cycle in each objective is $c_k^o(\mathcal{I}) \leftarrow \min_\pi c_k^o(\pi)$.

The cost function of π is now

$$c^o(\pi) = \sum_{i=1}^{N}\sum_{j=1}^{N} a_{ij} \cdot b^o_{\pi_i\pi_j} = \sum_{k=1}^{d} c_k^o(\pi) + R^o(\pi) \tag{3}$$

where $R^o(\pi)$ is a residue defined as the cost in the outside region for the flow matrix o

$$R^o(\pi) = \sum_{a_{ij}\in\mathcal{R}_A,\ b^o_{\pi_i\pi_j}\in\mathcal{R}_{B^o}} a_{ij} \cdot b^o_{\pi_i\pi_j} \tag{4}$$

Swapping facilities *in* a cycle results in swapping elements in the component mQAP and in the outside region. Swapping facilities *between* cycles results in swapping elements between the component mQAPs and the outside region.

3.1 Setting up bounds for the generating distributions

Let m_A and m_{B^o} be the smallest element in the component distance matrices A_k, $m_A \leftarrow \min_{k\leq d}\{a_{kij}\}$, and the component flow matrices $B^o_k, \forall o, m_{B^o} \leftarrow \min_{k\leq d}\{b^o_{kij}\}$, respectively. Similarly, $M_A \leftarrow \max_{k\leq d}\{a_{ij}\}$ and $M_{B^o} \leftarrow \max_{k\leq d}\{b^o_{kij}\}$. Let ℓ_A and L_A be the lowest and the highest bound for the distribution $\mathcal{L}_A$, and let ℓ_{B^o} and L_{B^o} be the lowest and the highest bound for $\mathcal{L}_{B^o}$. Let h_A and H_A be the lowest and the highest bound for $\mathcal{H}_A$ and let h_{B^o} and H_{B^o} be the lowest and the highest bound for $\mathcal{H}_{B^o}$.

The next proposition sets conditions on the bounds for the composite mQAP with the identity permutation as the optimum solution.

Proposition 1. *Let be* $\{(A^k, B^1_k, \ldots, B^m_k, \mathcal{I}) \mid k = 1, \ldots, d\}$ *a set of equal sized mQAP instances with the optimum solution the identity permutation. Algorithm 1 generates a composite mQAP from these component mQAPs. Let following equations hold*

$$\ell_A < L_A < m_A < M_A < h_A < H_A \tag{5}$$

$$\ell_{B^o} < L_{B^o} < m_{B^o} < M_{B^o} < h_{B^o} < H_{B^o} \tag{6}$$

$$\min\{m_A, m_{B^o}\}\cdot(\min\{h_A, h_{B^o}\}+\min\{\ell_A, \ell_{B^o}\}) > M_A\cdot M_{B^o}+\min\{h_A\cdot L_{B^o}, L_A\cdot h_{B^o}\} \tag{7}$$

$$\sum_{k=1}^{d}(c_k^o(\mathcal{I}) - c_k^o(\pi)) + \sum_{a_{ij}\in\mathcal{R}_A,\ b^o_{\pi_i\pi_j}\in\mathcal{R}_{B^o}} a_{ij}\cdot(b^o_{ij} - b^o_{\pi_i\pi_j}) < 0 \tag{8}$$

where π any permutation and for all objectives $o \le m$. Then, the composite mQAP $(A, B^1, \ldots, B^m, \mathcal{I})$ has the identity permutation as the optimum solution.

Proof. The proof follows directly from the proof of Proposition 1 from [5]. Intuitively, the set $\Pi(N)$ of all possible permutations is split in three subsets: i) exchange facilities *within* a cycle, ii) cycle that completely switch their facilities with other cycles, and iii) the general case where facilities are switched at random between cycles. The proof considers the difference between the identity permutation and another permutation for all these three cases. □

In Proposition 1, for Inequality 5 and 6, the rearrangement inequality holds. From Inequality 7 and the rearrangement inequality, we have that a permutation where facilities are swapped between the outside and the component region has a higher cost than a permutation where solutions are swapped *in* the composite or *in* the outside region. The condition in Inequality 7 can be fulfilled by setting the bounds for the distributions $\mathcal{H}_A$ and $\mathcal{H}_{B^o}$ high enough.

Inequality 8 states that if swapping elements in the outside region generates more variance than swapping elements in the component mQAPs, then the identity permutation is the global minimum for the subset of permutations where cycles are completely swapped. To decide if the generated composite mQAP has the identity permutation as optimum solution, we need $d!$ evaluations of Inequality 8 corresponding to all combinations of the component mQAPs on the diagonal of the composite mQAP.

4 A practical composite mQAP instance generator

In this section, we generate composite mQAP instances to fulfil the conditions from Proposition 1. The current mQAP instance generators [2, 7] use uniform distributions to generate mQAPs. Thus, we also use uniform distributions to generate composite mQAPs. Note that even though component mQAPs and the elements in the outside region are generated by uniform random distributions, the values of the corresponding composite mQAP instances are not generated by a uniform random distribution.

An uniform random distribution $\mathcal{D}$ generates all the component mQAPs. Let $\mathcal{L}$ and $\mathcal{H}$ be the uniform independent distributions generating the outside region of the distance matrix A and the flow matrices $\mathbf{B}$.

We study the relationship between the inequalities from Proposition 1 on the bounds for the uniform distributions. Let the two terms from Inequality 8 be denoted as the variance of the composite region and of the outside region

$$\Delta_{\mathcal{C}} = \sum_{k=1}^{d} c^k(\mathcal{I}) - c^k(\pi), \quad \Delta^o_{\mathcal{O}} = R^o(\mathcal{I}) - R^o(\pi)$$

We explicitly compute the values of $\Delta_{\mathcal{C}}$ and $\Delta^o_{\mathcal{O}}$. If $\Delta^o_{\mathcal{O}} + \Delta_{\mathcal{C}}$ is non-negative, the identity permutation is the global optimum solution.

Consider that there are $L - \ell + 1$ values in $\mathcal{L}$, $\ell = s_0, s_1, \ldots, s_{L-\ell} = L$, and $H - h + 1$ uniformly generated values in $\mathcal{H}$, $(h = t_0), t_1, \ldots, (t_{H-h} = H)$. Let's assume that $L - \ell = H - h$. With a perfect random generator, in any row and column of mQAPs' matrices values of $\mathcal{L}$ and of $\mathcal{H}$ are equally represented.

The variance in the outside region. Assuming that all the values of the distributions $\mathcal{L}$ and $\mathcal{H}$ are uniformly distributed, the cost of the outside region has the approximative value of

$$R^o(\mathcal{I}) = \sum_{a_{ij} \in \mathcal{R}_A} a_{ij} \cdot b^o_{ij} \approx \frac{|\mathcal{R}_A|}{H - h + 1} \cdot \left(\sum_{i=0}^{L-\ell} s_i \cdot t_{H-h-i} \right) \tag{9}$$

When $\alpha = 0.5$, the elements in the flow and distance matrices are equally generated from low and high distributions. The swapped elements are randomly distributed in the corresponding matrices and, thus, the cost of the outside region in each objective o is upper bounded by

$$R^o(\pi) \leq \frac{|\mathcal{R}_A|}{L - \ell + H - h + 2} \cdot \left(\sum_{i=0}^{L-\ell} s_i + \sum_{j=0}^{H-h} t_j \right)^2$$

For a permutation π, let assume that $(1 - p) \cdot |\mathcal{R}_A|$ percent of the outside region is optimised and the remaining $p \cdot |\mathcal{R}_A|$ percent of the outside region is uniform randomly positioned in the matrix. Then the cost of the outside region in each objective o is

$$R^o(\pi) \approx \frac{(1 - p) \cdot |\mathcal{R}_A|}{L - \ell + 1} \cdot \left(\sum_{i=0}^{L-\ell} s_i \cdot t_{L-\ell-i} \right) + \frac{p \cdot |\mathcal{R}_A| \cdot \left(\sum_{i=0}^{L-\ell} s_i + \sum_{j=0}^{H-h} t_j \right)^2}{L - \ell + H - h + 2}$$

Given a certain value for p, the variance is the outside region is

$$\Delta^o_{\mathcal{O}} \approx \frac{p \cdot |\mathcal{R}_A|}{L - \ell + 1} \cdot \left(\sum_{i=0}^{L-\ell} s_i \cdot t_{L-\ell-i} \right) - \frac{p \cdot |\mathcal{R}_A| \cdot \left(\sum_{i=0}^{L-\ell} s_i + \sum_{j=0}^{H-h} t_j \right)^2}{L - \ell + H - h + 2} \tag{10}$$

The variance in the component mQAPs. The minimum cost of all d component mQAPs is approximatively equal because all the values are generated from the same uniform distribution. This cost could be increased by the imperfection of the random generator, and the limited size of the component mQAP. Consider that there are $M - m + 1$ values in $\mathcal{D}$, such that $(m = v_0), v_1 \ldots, (v_{M-m} = M)$. Let $N^2 - |\mathcal{R}_A| = d \cdot n \cdot (n - 1)$ be the total number of elements in the component mQAPs. Following the same line of reasoning, the maximum variance is

$$\Delta_{\mathcal{C}} < \sum_{k=1}^{d} c^k(\mathcal{I}) - \frac{N^2 - |\mathcal{R}_A|}{(M - m + 1)^2} \left(\sum_{i=0}^{M-m} v_i \right)^2 \tag{11}$$

The elements of the component matrices are uniform randomly generated and positioned. Thus, when the component mQAPs are optimised and $N \to \infty$, we have

$$\sum_{k=1}^{d} c^k(\mathcal{I}) \approx \frac{N^2 - |\mathcal{R}_A|}{(M-m+1)^2} \left(\sum_{i=0}^{M-m} v_i \right)^2 \tag{12}$$

and $\Delta_{\mathcal{C}} \to 0$.

The variance in the composite mQAPs. Note that if $N \to \infty$, then $\Delta_{\mathcal{C}}$ is approaching 0, and $\Delta_{\mathcal{O}}$ has a negative value, $\Delta_{\mathcal{O}} < 0$, and the identity permutation is the optimal solution. This concludes our reasoning.

4.1 An example

We choose the bounds for the composite mQAPs to be the same with the bounds for the uniform randomly generated mQAPs from [2, 7] with the purpose of comparing the mQAP instances. These bounds are also set to cover a large number of values between 1 and 99, the same bounds as the randomly generated mQAP instances. Let's consider the following numerical values: i) $m = 21$ and $M = 40$, ii) $h = 80$ and $H = 99$, and iii) $\ell = 1$ and $L = 20$. Thus $L - \ell = H - h = 20$. Let $n = 8$ be the number of facilities in component mQAPs, where $d \geq 2$. Further, $s_i = i$ and $t_i = i + 80$, where $i \in \{1, \ldots, 20\}$.

If $d = 2, 3, \ldots$, then $|\mathcal{R}_A| = 128, 384, 768, \ldots$. Using Equation 9, the cost of the outside region is $R^o(\mathcal{I}) = |\mathcal{R}_A| \cdot \frac{1 \cdot 99 + \ldots + 20 \cdot 80}{20} = |\mathcal{R}_A| \cdot 906.5$. From Equation 10, we have that $\Delta^o_{\mathcal{O}} \approx p \cdot |\mathcal{R}_A| \cdot 906.5 - p \cdot |\mathcal{R}_A| \cdot 6703.2 \approx -p \cdot |\mathcal{R}_A| \cdot 5807.7$. Note that the second term it is negative and dominates $\Delta^o_{\mathcal{O}}$. From $N^2 - |\mathcal{R}_A| = 112, 168, 224, \ldots$ and Equation 11, we have that $\Delta_{\mathcal{C}} < \sum_{k=1}^{d} c^k(\mathcal{I}) - (N^2 - |\mathcal{R}_A|) \cdot 29241$. From Equation 12, we have that $\Delta_{\mathcal{C}} \to 0$, and thus the identity permutation is the optimum solution for all the composite mQAPs with these uniform distributions. Note that the condition from Inequality 7 was relaxed for this numerical example.

5 Difficulty of mQAP instances

Our goal is to generate instances that are difficult to solve with local search. We propose to use as difficulty measures for mQAPs: i) the covariance coefficients of the elements in two different flow matrices, and ii) the correlation between the cost functions of two objectives.

Dominance [7] is a measure of the amplitude of the variance for the flow matrix and distance matrix. Note that there are $m+1$ dominance values: m flow dominance values and one distance dominance value. We denote the distance and flow dominances with $d_a = \frac{\sigma_a}{\mu_a}\%$ and $d_k = \frac{\sigma_{b^k}}{\mu_{b^k}}\%$, where μ_a and σ_a is the mean and the standard deviation for the matrix a. A matrix with low epistasis has the dominance close to the lower bound, 0. The dominance's upper bound is 100.

We propose to measure the amplitude of the sample covariance between two flow matrices, b^k and b^r. The *dominance* of the flow matrices b^k and b^r is defined as $d_{kr} = \frac{1}{\sqrt{\mu_{b^k} \cdot \mu_{b^r}}} \cdot \sqrt{\frac{\sum_{i,j=1}^{n} (b^k_{ij} - \mu_{b^k}) \cdot (b^r_{ij} - \mu_{b^r})}{n^2}}\%$.

type mQAPs	N	n	Dominance d_a	d_{b1}	d_{b2}	d_{12}	Ruggedness ϕ_{b1}	ϕ_{12}	Asymptotic behav Best Know	Invers	Gap	Empirical behav % opt	mean	Gap
Knowless'	25		60	64	63	18	96	92	646561				672236	3.8
	50		59	61	58	13	98	98	5264742				5333790	1.2
	75		60	59	60	13	99	98	12285680				12476382	1.5
Composite	25	1	94	99	95	97	95	95	183427	1381042	87	100	183427	87
	50	1	92	97	93	94	98	97	737069	6010180	87	7	1965328	67
	75	1	91	96	91	94	99	99	1923900	13417974	88	0	4766598	64
	25	5	83	93	83	88	95	95	189465	1352778	86	100	189465	86
	50	5	85	95	85	90	96	96	745782	5560362	87	12	1707298	69
	75	5	83	92	83	87	98	98	3317244	12324804	73	0	4768865	61
	20	10	83	97	90	93	98	98	244998	536258	54	100	244998	54
	50	10	84	93	84	89	98	97	742401	4807446	85	1	1790843	63
	80	10	82	92	83	87	98	98	3870974	14129836	73	0	5351231	62

Table 1. Analytical and empirical properties of 9 bi-objective QAP instances.

Ruggedness [10] is a normalisation of the *autocorrelation* coefficient for the cost function c^k when a (m)QAP is explored with local search. By definition, the autocorrelation coefficient for the k objective is $\epsilon_{c^k} = \frac{2\cdot(\mathbb{E}[(c^k)^2]-\mu^2_{c^k})}{\mathbb{E}[(c^k(\pi)-c^k(\pi'))^2]}$, where μ_{c^k} is the average of c^k and π and π' are *any* two permutations. The ruggedness coefficient for the k-th objective is $\phi_{b^k} = 100 - \frac{400}{n-2}\cdot(\epsilon_{b^k} - \frac{n}{4})$. A ruggedness coefficient close to 0 indicates a flat landscape, whereas a large ϕ^k, close to 100, indicates a steep landscape with lots of local optima.

We propose to measure the correlation between the cost functions of two objectives, c^k and c^r, $\epsilon_{kr} = \frac{\mathbb{E}[c^k(\pi)-\mu_{c^k}]}{\sigma_{c^k}}\cdot\frac{\mathbb{E}[c^r(\pi)-\mu_{c^r}]}{\sigma_{c^r}}$. The *ruggedness* of the objectives k and r is defined as $\phi_{kr} = 100 - \frac{400}{n-2}\cdot(\epsilon_{kr} - \frac{n}{4})$.

A difficult (m)QAP instance has both large dominance and ruggedness.

Asymptotic behaviour of mQAPs calculates the difference between the optimum value (or a known feasible solution) and the value of the solution generated with the inverse permutation of that solution. Here, we consider the inverse of the optimum (or, if optimum is unknown, the best known solution) an approximation of the worst solution of an instance of composite mQAP. Thus, for mQAPs with the optimum solution $\mathcal{I}$, we assume that the reverse of the identity permutation, $\mathcal{I}^{-1}$, is an approximation of the worst solution for that instance. The gap is $\frac{inverse\_solution - best\_known\_solution}{inverse\_solution} * 100\%$, where the $best\_known\_solution$ is the best solution returned by an algorithm and $inverse\_solution$ is the inverse solution for the best known so far.

Numerical examples. Let consider the numerical example from Section 4. In Table 1 we compare the difficulty of several bi-objective QAP instances from [7] and the composite bi-objective QAPs. We consider the correlation between the flow matrices $\rho = 0.75$, and $N = \{25, 50, 75\}$. The asymptotic and empirical behaviour is shown only for the first objective. To compute the empirical behaviour of bQAPs we run iterated Pareto LS [8] for 50 times each run for 10^6 position swaps in a permutation. The composite bQAPs are most difficult tested instances because they have the largest dominance values, ruggedness coefficients and gaps. The small composite bQAPs, $N = 25$, have a lower ruggedness than the large composite bQAPs, $N = 75$. The dominance values and the gap decrease with the size increase of the component bQAPs because there is less variance in the values of the component matrices. Note that the empirical gap is much smaller than the asymptotic gap. To conclude, the analytical (difficulty measures) and empirical properties of the composite bQAPs outperform the same properties of the uniformly randomly generated bQAPs.

6 Conclusion

We propose a multi-objective quadratic assignment instance generator that aggregates several small multi-objective QAP instances into a larger mQAP instance. Both the component mQAP instances and the cost of the elements outside these components have, by design, the identity permutation as the optimal solution. We give mild conditions under which the resulting composite mQAP instances have identity permutation as the optimum solution. We propose difficulty measures to compare the proposed composite mQAPs with other mQAPs from literature. We conclude that composite bQAP instances are more difficult than the uniform random bQAPs, and in addition, they have a known optimum solution.

References

1. Puglierin, F., Drugan, M.M., Wiering, M.: Bandit-inspired memetic algorithms for solving quadratic assignment problems. In: Proc of CEC, IEEE (2013) 2078–2085
2. Drugan, M.M.: Cartesian product of scalarization functions for many-objective QAP instances with correlated flow matrices. In: Proc. of GECCO, ACM (2013) 527–534
3. Çela, E.: The Quadratic Assignment Problem: Theory and Algorithms. Kluwer Academic Publishers, Dordrecht (1998)
4. Drezner, Z., Hahn, P., Taillard, E.: Recent advances for the quadratic assignment problem with special emphasis on instances that are difficult for meta-heuristic methods. Annals of Operations Research **139**(1) (2005) 65–94
5. Drugan, M.: Generating QAP instances with known optimum solution and additively decomposable cost function. Journal of Combinatorial Optimization (2014)
6. Palubeckis, G.: An algorithm for construction of test cases for the quadratic assignment problem. Informatica, Lith. Acad. Sci. **11**(3) (2000) 281–296
7. Knowles, J., Corne, D.: Instance generators and test suites for the multiobjective quadratic assignment problem. In: Proc. of. Evolutionary Multi-Criterion Optimization EMO 2003. Number 2632 in LNCS, Springer (2003) 295–310
8. Drugan, M.M., Thierens, D.: Stochastic pareto local search: Pareto neighbourhood exploration and perturbation strategies. J. Heuristics **18**(5) (2012) 727–766
9. Wayne, A.: Inequalities and inversions of order. Scripta Mathematica **12**(2) (1946) 164–169
10. Angel, E., Zissimopoulos, V.: On the hardness of the quadratic assignment problem with metaheuristics. J. Heuristics **8**(4) (2002) 399–414