Hybrid PSO-TS Approach for solving the Quadratic Three-Dimensional Assignment Problem

Ait Abderrahim Imene and Loukil Lakhdar

February 28, 2020

Hybrid PSO-TS Approach for solving the Quadratic Three-Dimensional Assignment Problem

Imene Ait Abderrahim
LAPECI laboratory,
Computer science department,
University of Oran 1 Ahmed Ben Bella, Algeria
aitabderrahim_i@hotmail.fr

Lakhdar Loukil
LAPECI laboratory,
Computer science department,
University of Oran 1 Ahmed Ben Bella, Algeria
loukil.lakhdar@univ-oran1.dz

***Abstract*—The quadratic three-dimensional assignment problem (Q3AP) is a generalization of the well-known quadratic assignment problem (QAP). Unlike QAP which has been extensively studied by the combinatorial optimization community, few works have been devoted to the resolution of the Q3AP which is proved to be an NP-hard problem. In this paper, a particle swarm optimization algorithm hybridized with a tabu search is presented to solve the quadratic three-dimensional assignment problem.**

Keywords: Quadratic Assignment Problem (QAP), Quadratic Three-Dimensional Assignment Problem (Q3AP), Particle Swarm Optimization (PSO), Tabu Search (TS), Hybridization.

1. Introduction

The quadratic three-dimensional assignment problem (Q3AP) is a generalization of the well-known Quadratic Assignment Problem (QAP) introduced by Pierskalla [2] in 1967. The Q3AP consists in finding an optimal symbol-mapping over two vectors so as to minimize an objective function. An application example of the Q3AP is the Hybrid Automatic Repeat reQuest (Hybrid-ARQ) error-control mechanism used in wireless communication systems to detect altered bits in transmitted packets [20].

Unlike QAP which has been extensively considered in the litterature, few works have been devoted to Q3AP. The most important one is that achieved by Peter Hahn et al. [3]. In this work, a sequential branch-and-bound algorithm based on dual ascent procedure and four local search methods (simulated annealing, tabu search, fast ant system and iterated local search) have been successfully applied for solving some Q3AP instances.

Q3AP is a NP-hard problem because it is an extension of two problems which are themselves NP-hard: the quadratic assignment problem and the axial 3-assignment problem (A3AP). Typical symbol-mapping (Q3AP) problems may be of sizes 8, 16, 32 and 64. Therefore, we can not consider solving large size instances using exact methods. We also conjecture that combining metaheuristics that have complementary features can significantly improve the effectiveness of the resolution methods. Our proposal in this paper is an approximate optimization method that combines a particle swarm optimization algorithm with a tabu search method.

The remaining of the paper is organized as follows. In section 2, we define the QAP and Q3AP problems. Sections 3 and 4 describe, respectively, the particle swarm optimization (PSO) algorithm and PSO applied to Q3AP. In the section 5, we present our proposal which is a hybrid optimization method that combines PSO with tabu search. Section 6 reports experimental results. Section 7 concludes the work and gives some perspectives.

2. QAP and Q3AP

The quadratic assignment problem (QAP) is one of the most challenging and studied problem by the combinatorial optimization community. It was introduced by Koopmans and Beckmann [21] in 1957 as a mathematical model for the economic problem of assigning a set of economical activities to a set of sites. Many theoretical and real-life problems can be modeled as a QAP.

The problem is expressed as follows. Given three $(n \times n)$ matrices, the flow matrix $F = (f_{ij})$ between each pair of activities i and j, the distance matrix $D = (d_{kl})$ between each pair of sites k and l, and the cost matrix $B = (b_{ij})$ giving the cost of placing facility i at location j, the objective is to find the best assignment of activities to sites so as to minimize the total cost. Mathematically, the QAP can be written as follows:

$$\min_{\pi \in S_n} \left\{ \sum_{i=1}^{n} \sum_{j=1}^{n} f_{\pi(i)\pi(j)} d_{ij} + \sum_{i=1}^{n} b_{\pi(i)i} \right\} \qquad (1)$$

where S_n is the set of all permutations over $\{1, 2, .., n\}$. Each term $f_{\pi(i)\pi(j)} d_{ij}$ means the cost of assigning the facility $\pi(i)$ to location i and the facility $\pi(j)$ to location j.

Let us consider the $(n \times n)$ binary matrix $X = (x_{ij})$ which elements x_{ij} are defined as follows:

$$x_{ij} = \begin{cases} 1 & \text{if facility } i \text{ is assigned to location } j, \\ 0 & \text{otherwise.} \end{cases} \qquad (2)$$

The Koopmans-Beckmann QAP (1) can then be reformulated as:

$$\min \left\{ \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{n}\sum_{l=1}^{n} f_{ik}d_{jl}x_{ij}x_{kl} + \sum_{i=1}^{n}\sum_{j=1}^{n} b_{ij}x_{ij} \right\} \quad (3)$$

subject to:

$$\sum_{i=1}^{n} x_{ij} = 1; \quad j = 1,2,...,n, \quad (4)$$

$$\sum_{j=1}^{n} x_{ij} = 1; \quad i = 1,2,...,n, \quad (5)$$

$$x_{ij} \in \{0,1\} \quad ; \quad i,j = 1,2,..,n. \quad (6)$$

The binary matrix X defined by Eq. (2) and verifying constraints (4)-(6) is named a permutation matrix. Therefore, solving a QAP consists in finding the permutation matrix $X = (x_{ij})$ that minimizes the objective quadratic function (3).

The quadratic 3-dimensional assignment problem (Q3AP) is a extension of the quadratic assignment problem and of the axial 3-dimensional assignment problem (3AP). The Q3AP can be formulated as follows:

$$\min \{ \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{n}\sum_{k=1}^{n}\sum_{n=1}^{n}\sum_{q=1}^{n} c_{ijpknq}x_{ijp}x_{knq} + \sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{p=1}^{n} b_{ijp}x_{ijp} \} \quad (7)$$

where:

$$X = (x_{ijp}) \in I \cap J \cap P, \quad (8)$$

$$x_{ijp} \in \{0,1\}. \quad i,j,p = 1,2,...,n. \quad (9)$$

I, J and P sets are defined as follows:

$$I = \{X = (x_{ijp}) : \sum_{j=1}^{n}\sum_{p=1}^{n} x_{ijp} = 1, \quad \text{for} \quad i = 1,...,n\}, \quad (10)$$

$$J = \{X = (x_{ijp}) : \sum_{i=1}^{n}\sum_{p=1}^{n} x_{ijp} = 1, \quad \text{for} \quad j = 1,...,n\}, \quad (11)$$

$$P = \{X = (x_{ijp}) : \sum_{i=1}^{n}\sum_{j=1}^{n} x_{ijp} = 1, \quad \text{for} \quad p = 1,...,n\}. \quad (12)$$

While for the QAP the problem is to find a 2-dimensional permutation matrix that minimizes the objective function, the problem for Q3AP is to minimize a quadratic function over the 3-dimensional assignment polytope $I \cap J \cap P$. That is why this problem is referred to as quadratic 3-dimensional assignment problem.

An alternative formulation that we particularly used in our implementations is the permutation-based formulation. In permutation-based formulation, the Q3AP given by Eqs. (7)-(9) can be expressed as:

$$\min \{ f(p,q) = \sum_{i=1}^{n}\sum_{j=1}^{n} c_{ip_{(i)}p_{(j)}jq_{(i)}q_{(j)}} + \sum_{i=1}^{n} b_{ip_{(i)}q_{(i)}} \}, \quad (13)$$

where p and q are permutations over the set $\{1,..,n\}$. The first term in (13) can be interpreted as the cost of mapping simultaneously the entity i to the locations $p(i)$ and $q(i)$ and the entity j to the locations $p(j)$ and $q(j)$. The second term can be interpreted as the cost of placing entity i simultaneously to locations $p(i)$ and $q(i)$.

3. Particle Swarm Optimization (PSO) for solving the Q3AP

Particle Swarm Optimization (PSO) method was proposed by Kennedy and Eberhart [6]. PSO is inspired by social behavior patterns of organisms that live and interact within large groups. In particular, PSO incorporates swarming behaviors observed in flocks of birds, schools of fish, or swarms of bees, and even human social behavior.

The classical PSO model consists of a swarm of particles, which are initialized with a population of random candidate solutions. They move iteratively through a d-dimension problem space to search the new solutions, where the fitness, f, can be calculated as the certain qualities measure. Each particle has a position represented by a position-vector x_i (i is the index of the particle), and a velocity represented by a velocity-vector v_i. Each particle remembers its own best position so far in a vector p_i. The best position-vector among the swarm so far is then stored in a vector p_g. During the iteration time t, the update of the velocity from the previous velocity to the new velocity is determined by Eq. (14). The new position is then determined by the sum of the previous position and the new velocity by Eq. (15) [7]:

$$v_{ij}(t+1) = \omega v(t) + c_1 r_1(p_{ij}(t) - x_{ij}(t)) + c_2 r_2(p_{gj}(t) - x_{ij}(t)), \quad (14)$$

$$x_i(t+1) = x_i(t) + v_i(t+1), \quad (15)$$

where r_1 and r_2 are random numbers in the interval $[0, 1]$. c_1 and c_2 are positive constants called coefficient of the self-recognition and coefficient of the social component respectively. The variable ω is the inertia factor which value is typically setup to vary linearly from 1 to near 0 during the iterated process. From Eq. (14), a particle decides where to move next, considering its own experience, which is the memory of its best past position, and the experience of its most successful particle in the swarm.

In order to guide the particles effectively in the search space, the maximum moving distance during one iteration is clamped in between the maximum velocity $[-v_{max}.v_{max}]$. Inertia weight ω is a parameter to control the impact of the previous velocities on the current velocity [15]
Evaluation of each particle in the swarm requires the determination of the permutation of numbers $1..n$ since he value of function z in Q3AP problem is a result of the sequence. In this paper, we use a heuristic rule called Smallest Position Value (SPV) [8] to enable the continuous PSO algorithm to be applied to all classes of sequencing problems, which are NP-hard in the literature. By using the SPV rule, the permutation can be determined through the position values of the particle so that the fitness value of the particle can then be computed with that permutation. The pseudo-code for particle swarm optimization algorithm is illustrated in Algorithm 1.

Algorithm 1 : Particle Swarm Optimization (PSO) for Q3AP

```
Initialize parameters
Initialize population
Find sequence
Evaluate
Do {
 Find the personal best
 Find the global best
 Update velocity
 Update position
 Find sequence using SPV heuristic
 Evaluate
} While (Not termination)
```

A population of particles is constructed randomly for the PSO algorithm of the Q3AP problem. The continuous values of positions are established randomly. The following formula is used to construct the initial continuous position values of a particle:

$$x_{ij} = x_{min}(x_{max} - x_{min}) \times U(1, 0),$$

where $x_{min} = 0.0$, $x_{max} = 4.0$ and $U(0, 1)$ a uniform random number between 0 and 1.

The initial continuous velocity values of a particle are given with the following formula:

$$v_{ij} = v_{min}(v_{max} - v_{min}) \times U(1, 0),$$

where $v_{min} = -4.0$, $v_{max} = 4.0$.
We give below the steps of PSO for Q3AP:

Step 1: Initialization.

- Let $k = 0$, m=size of swarm.
- Generate m particles randomly as explained.
- Generate initial velocities of particles randomly.
- Apply SPV rule to find the sequence.
- Evaluate each particle i in the swarm using the objective function Z_i for $i = 1, , m$.
- For each particle i in the swarm, set best personal position along with its best fitness value.
- Find the best fitness value $Z_l = \min Z_i$ for $i = 1, , m$ with its corresponding position.
- Set global best with its fitness value.

Step 2: Update iteration counter.

- $k = k + 1$

Step 3: Update inertia weight.

- $\omega = \omega * \alpha$, where α is the damping ratio.

Step 4: Update velocity using Eq. (14).
Step 5: Update position using Eq. (15).
Step 6: Find Sequence.

- Apply SPV rule to find the sequence.

Step 7: Update personal best.

- Each particle is evaluated by using its sequence to test if personal best position will improve. That is, if $f(x) < f(PB)$ for $i = 1, ..., m$, then personal best is updated.

Step 8: Update global best.

- Find the minimum value of personal best.
- If $f(x) < f(GB)$, then the global best is updated.

Step 9: Stopping criterion.

- If the number of iterations exceeds the maximum number of iterations, or maximum CPU time, then stop, otherwise go to step 2.

4. Hybridizing PSO with TS for solving Q3AP

A tabu search algorithm has been considered (see Algorithm 2). Basically, the tabu search enhances the performance of a local search method by using memory structures. Indeed, the main memory structure called the tabu list represents the history of the search trajectory. In this way, using this list allows to avoid cycles during the search process. More details of this algorithm are given in [13].

Algorithm 2 Tabu Search (TS)

Choose an initial solution
Evaluate the solution
Initialize the tabu list
repeat
  for each generated neighbor do
    Incremental evaluation of the candidate solution
    Insert the resulting fitness into the neighborhood fitnesses structure
  end for
  Select the best admissible neighboring solution
  Update the tabu list
until a maximum number of iterations reached

A hybridization scheme has been proposed to improve the results obtained in [10]. The SPV (Smallest Position Value) heuristic that was intended to generate the next explored solution in the search space is replaced by a tabu search that explores the neighborhood of the current solution and chooses the best neighbor solution. The motivation for replacing PSV by TS is that PSV breaks the permutation structure of the solution which causes a problem of convergence.

5. Experimental results

In this section, we report and comment the experimental results. The Q3AP PSO-TS has been coded in C++ programming language.
Experiments have been conducted on an intel core i5-5200U 2GHz laptop

PSO parameters have been set to the following values:

- Size of swarm $m = 80$.
- $c_1 = c_2 = 2$.
- $r_1 = U(0, 1)$, $r_2 = U(0, 1)$.
- Initial inertia weight $\omega_0 = 1.0$.
- The decrement factor $\alpha = 0.99$.

It is clear that parameters tuning can affect the performance of an algorithm. So, we begin by a tuning process to find the best parameter values. We tes our algorithm on two Q3AP instances derived from nug8 and nug12 QAP instances of QAPLIB.

The number of iterations of PSO and TS are set to 1000 and 3000 respectively. The evolution of the fitness function over time for nug8 and nug12 instances are given in Fig. 1 and Fig. 2 respectively. We notice that as the number of iterations increases, the fitness of solution improves. There is another point noticeable in which the cost decreased; it is the choice of the lower and upper bound of variables as well as the choice of the learning coefficients $(c_1.c_2)$.
The number of TS iterations is set to 100, and the size of the tabu list is set to $(n * (n - 1))^2/16$ where n is the size of the instance.

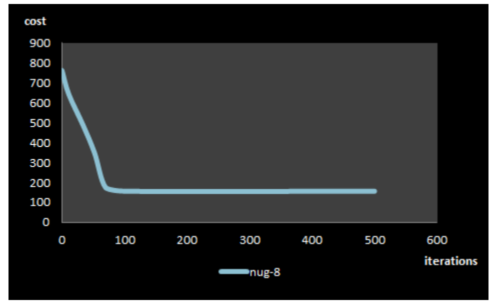

Figure 1: Change of cost over time of the instance nug8

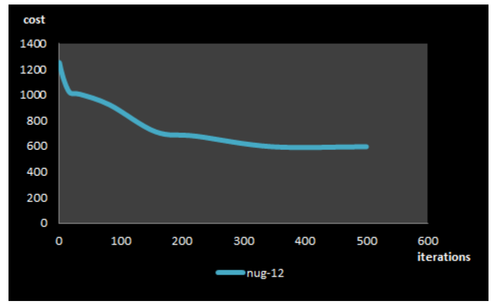

Figure 2: change of cost over time of the instance nug12

6. Conclusion

The quadratic 3-dimensional assignment problem (Q3AP) is a challenging combinatorial optimization problem.

In this paper, we have proposed an optimization approach that consists of a particle swarm optimization hybridized with a tabu search algorithm for solving this problem.
The experiments carried out on 2 instances showed that our approach is promising but not completely satisfactory. As a short-term perspective, we intend to study the performance of our algorithm in terms of execution time and quality of solutions on larger instances.

References

[1] P. M. Hahn, Y. R. Zhu, M. Guignard and J. M-G. Smith. *Exact Solution of Emerging Quadratic Assignment Problems*. International Transactions in Operational Research, 2013.

[2] W. P. Pierskalla. *The multi-dimensional assignment problem*. Technical Memorandum No. 93, Operations Research Department, CASE Institute of Technology, 1967.

[3] P. M. Hahn, B-J. Kim, T. Stützle, S. Sebastian, W. L. Hightower, H. Samra, Z. Ding, M. Guignard. *The Quadratic Three-dimensional Assignment Problem : Exact and Approximate Solution Methods*. European Journal of Operational Research, 2008.

[4] B. Zhang, X. Zhang, J. Hu, D. Yang. *A hybrid ARQ scheme with constellation rearrangement and power adjustment*. Proceedings of the International Conference of Wireless Communications, Networking and Mobile Computing, pages 464468, 2005.

[5] H. Samra, Z. Ding, P. M. Hahn. *Symbol mapping diversity design for multiple packet transmissions*. IEEE Transactions on Communications 53 (5), 810817, 2005.

[6] J. Kennedy, R. Eberhart. *Particle swarm optimization*. Proceedings of IEEE international conference on neural networks. doi: 10.1109/ICNN.1995.488968: 19421948, 1995.

[7] H. Liu, A. Abraham, J. Zhang. *A Particle Swarm Approach to Quadratic Assignment Problems*. Proceedings of The Eleventh Online World Conference on Soft Computing in Industrial Applications, Springer, 2007.

[8] A. S. Mamaghani, M. R. Meybodi. *Solving the Quadratic Assignment Problem with the Modified Hybrid PSO Algorithm*. 6th IEEE International Conference on Application of Information and Communication Technologies (AICT), pp. 1-6, 2012.

[9] Y. Shi, R. C. Eberhart. *A modified particle swarm optimizer*. In Proceedings of the IEEE International Conference on Evolutionary Computation, pp. 6973, 1998.

[10] I. Ait Abderrahim. *On Solving the Quadratic Three-Dimensional Assignment Problem with the PSO Algorithm*. 6th International Conference on Metaheuristics and Nature Inspired Computing, META2016, 2016.

[11] P. Szwed, W. Chmiel. *Multi-swarm PSO algorithm for the quadratic assignment problem: a massive parallel implementation on the OpenCL platform*. Computing Research Repository, 2015.

[12] T. V. Luong, L. Loukil , N. Melab, E.-G Talbi. *A GPU-based Iterated Tabu Search for Solving the Quadratic 3-dimensional Assignment Problem*. ACS/IEEE International Conference on Computer Systems and Applications (AICCSA), Workshop on Parallel Optimization in Emerging Computing Environments (POECE), 2010.

[13] F. Glover. *Future paths for integer programming and links to artificial intelligence*. Computers and Operations Research, vol. 13, no. 5, pp. 533549, 1986.

[14] T. Stützle. *Iterated local search for the quadratic assignment problem*. European Journal of Operational Research, 174(3):1519 (1539), 2006.

[15] L. Loukil, M. Mehdi, N. Melab, E.-G.Talbi, P.Bouvry. *Parallel Hybrid Genetic Algorithms for Solving Q3AP on Computational Grid*. International Journal of Foundations of Computer Science, Volume 23, Issue 02, 2012.

[16] R. E. Burkard , S. E. Karisch , and F. Rendl. *QAPLIB - a quadratic assignment problem library*. European Journal of Operational Research, 55:115119, 1991.

[17] C. Blum, J. Puchinger, G.Raidl, A. Roli. *A brief survey on hybrid metaheuristics*. 4th International Conference on Bioinspired Optimization Methods and their Applications, pp. 3-16, Ljubljana, 2010.

[18] G. Raidl. *A Unified View On Hybrid Metaheuristics*. HM'06 Proceedings of the Third international conference on Hybrid Metaheuristics, 2006.

[19] P. Lipinski. *A Hybrid Evolutionary Algorithm to Quadratic Three-Dimensional Assignment Problem with Local Search for Many-Core Graphics Processors*. Springer-Verlag Berlin Heidelberg, Volume 6283, pp. 344-351, 2010.

[20] L. K. Rasmussen, B. W. Wicker. *Trellis-Coded, Type-I Hybrid-ARQ Protocols Based on CRC Error-Detecting Codes*. IEEE Trans. Commun., vol. COM-43, pp. 25692575, Oct. 1995.

[21] T. C. Koopmans and M. J. Beckmann. *Assignment problems and the location of economic activities*. Econometrica, 25:5376, 1957.