Multi-index Axial Assignment Problem: 
An Asymptotically Optimal Approach. *

Gimadi E.Kh. Kairan N.M. 
Sobolev Institute of mathematics, Novosibirsk, Russia.

Abstract

The Multi-index Axial Assignment Problem is considered. It known be NP-hard in general case. In the paper the sublinear approximation algorithm is proposed. Performance and probability-of-failure bounds of the algorithm are obtained, and some conditions of asymptotical optimality of algorithm described are proved in the case of matrices with random elements distributed identically and independently.

1 Introduction

Multi-index assignment problem (MAP) [1-3] has numerous applications to communication, logistics, manufacturing and economics will be investigated under additional restrictions (planarity, axiality, permutations and so on).

The MAP (or \(m\)-AP) is formulated as follows:

\[
\text{minimize} \sum_{i_1=1}^n \sum_{i_2=1}^n \ldots \sum_{i_m=1}^n c_{i_1 i_2 \ldots i_m} x_{i_1 i_2 \ldots i_m}
\]

subject to

\[
\sum_{i_1=1}^n \ldots \sum_{i_s-1=1}^n \sum_{i_{s+1}=1}^n \ldots \sum_{i_m=1}^n x_{i_1 i_2 \ldots i_m} = 1
\]

for each \(i_s = 1, 2, \ldots, n\); \(s = 1, 2, \ldots, m\);

\(x_{i_1 i_2 \ldots i_m} \in \{0, 1\}\), for all \((i_1, i_2, \ldots, i_m)\),

where \(c_{i_1 i_2 \ldots i_m}\) are given real numbers.

In the case \(m = 2\) we have a classical (two-index) Assignment Problem (AP) which can be solved by the exact Dinitis algorithm [4] in \(O(n^3)\).

Contrary to the classical AP, the \(m\)-AP, \(m \geq 3\), is unlikely to be efficiently solvable, since the 3-AP is known to be NP-hard [3]. Therefore, we pay attention to design and analysis of approximation algorithms which can solve the MAP in a polynomial time. We use the approach suggested in [9] to discrete optimization problems with random inputs.

*This research was supported by the Russian Foundation for Basic Research (Grants 99-01-0601)
It is known that the classical AP with a random $n \times n$-matrix can be solved by an asymptotically optimal (approximation) algorithm whose running time is lineal in the problem size [8].

In this paper we propose an approximation algorithm for the MAP. We evaluate its running time and formulate conditions providing asymptotical optimality on random inputs specified for some distribution function. Note that algorithm described for the 3-AP has sublineal running time. These results are generalized on the case of the $m$-AP, $m > 3$.

Fast approximation algorithms and probabilistic distribution classes over which an algorithm is asymptotically optimal (i.e. its relative error and probability-of-failure approach to zero as the input size increases) are of a great interest (see [5-13]).

Let us denote:

- $f^*$ is the optimal value of the objective function;
- $f_A$ is the function value of the problem obtained by algorithm $A$.

Due to [9], we say that the algorithm $A$ has bounds $(\varepsilon_A, \delta_A)$ in the class of problems considered if the following inequality is true:

$$\Pr \{ f_A > (1 + \varepsilon_A) f^* \} \leq \delta_A ,$$

where $\varepsilon_A$ is the performance bound (relative error) obtained after the algorithm $A$ performed and $\delta_A$ is the probability-of-failure bound of algorithm $A$.

It is of interest to consider what happens to the bounds $\varepsilon_A$ and $\delta_A$ when the problem size grows.

An algorithm $A$ is called asymptotically optimal in a class of problems considered if there exist bounds $\varepsilon_A$ and $\delta_A$ tending to zero when of the problem size grows.

Note that this line of investigation has been very fruitful (especially in Russian literature) for the Travelling Salesman Problem on minimum and maximum, for the Bin Packing Problem, for the Packing in the Strip, for the Simple Plant Location Problem [see 8-13].

2 Three-index Axial Assignment Problem

Among the MAP, the 3-AP has a vivid geometrical interpretation and the simplest mathematical formulation:

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \sum_{k=1}^{n} c_{ijk} x_{ijk} \rightarrow \min_{(x_{ijk})}$$

subject to

$$\sum_{j=1}^{n} \sum_{k=1}^{n} x_{ijk} = 1, \; i = 1, \ldots, n;$$

$$\sum_{i=1}^{n} \sum_{k=1}^{n} x_{ijk} = 1, \; j = 1, \ldots, n;$$

$$\sum_{i=1}^{n} \sum_{j=1}^{n} x_{ijk} = 1, \; k = 1, \ldots, n;$$

$$x_{ijk} \in \{0, 1\}.$$
for all triples of integers \((ijk), 1 \leq i, j, k \leq n\).

Let elements of the matrix \((c_{ijk})\) be random independent variables, selected from a segment \([a_n, b_n], a_n > 0\), with an identical distribution function. Suppose that the distribution function is described by a distribution function \(F_\xi(x) = Pr\{\xi < x\}\) of a normalized random variable

\[
\xi = \frac{(c_{ijk} - a_n)}{(b_n - a_n)}, \quad 0 \leq \xi \leq 1.
\]

Let \(C_n\) be the set of all problems 3-AP with the matrices \((c_{ijk})\) determined above.

Denote by \(\phi_n\) any function which takes integer values, \(1 \leq \phi_n \leq n\).

Put a description of Algorithm \(A(\phi_n)\) for finding of approximation solution of the 3-AP by stages.

**Algorithm \(A(\phi_n)\):**

1. Let \((d_{ij})\) be \(n \times n\)-matrix which contains elements of the original matrix \((c_{ijk})\) with \(j = k\):

   \[
d_{ij} = c_{ijj}, \quad \text{for all } 1 \leq i, j \leq n.
\]

   Put \(f = 0; \ i = 1,\) and \(J = \{1, 2, \ldots, \phi_n\}\).

2. Pick a number \(\sigma(i)\) as an arbitrary index from the set

   \[
   \text{Arg min}\{d_{ij} \mid j \in J\}
   \]

which consists of column numbers of minimal elements in the \(i\)-th row of the matrix \((d_{ij})\) determined by the column set \(J\).

3. Set \(f := f + d_{i\sigma(i)}; \ J := J \setminus \{\sigma(i)\}; \ j := i + \phi_n\).

4. If \(j \leq n\) then \(J := J \cup \{j\}\)

5. Set \(i := i + 1\).

6. While \(i < n\) repeat the stage 2. Otherwise execute the next stage 7.

7. Represent the solution of the original 3-AP as follows:

\[
x_{ijk} = 1 \quad \text{for all } j = k = \sigma(i), \ i = 1, 2, \ldots, n.
\]

Let every other element of the matrix \((x_{ijk})\) be zero.

8. Stop. As a result of Algorithm \(A(\phi_n)\) we have the value of the objective function

\[
f_{A(\phi_n)}(x) = f.
\]

The description of Algorithm \(A(\phi_n)\) is finished.

It immediately implies the following statement:

**Proposition 1.** Algorithm \(A(\phi_n)\) finds a feasible approximation solution of 3-AP in \(O(n\phi_n)\) time.

**Theorem 1.** If

\[
\frac{b_n}{a_n} = \Theta\left(\frac{n}{S(n, \phi_n)}\right), \tag{1}
\]

and

\[
S(n, \phi_n) \to \infty \quad \text{with} \quad n \to \infty, \tag{2}
\]

then Algorithm \(A(\phi_n)\) finds the asymptotically optimal solution of the 3-AP on the class of problems \(C_n, n = 3, 4, \ldots\).
Corollary 1. Suppose \( F_\xi(x) \geq x, \ 0 \leq x \leq 1; \phi_n = \ln n \). If \( b_n/a_n = o(\ln n) \), then Algorithm \( A(\phi_n) \) yields an asymptotically optimal solution of 3-AP on class of problems \( C_n \) in \( O(n \ln n)\)-time.

Corollary 2. Suppose \( F_\xi(x) \geq x, \ 0 \leq x \leq 1; \phi_n = n \). If \( b_n/a_n = o(n/\ln n) \), then Algorithm \( A(\phi_n) \) yields an asymptotically optimal solution of 3-AP on class of problems \( C_n \) in \( O(n^2)\)-time.

3 \( m \)-AP, \( m > 3 \)

The statements formulated above for 3-AP can be also obtained in the case \( m \)-AP, \( m > 3 \). As elements of a \( n \times n \)-matrix \( (d_{ij}) \) used in Algorithm \( A(\phi_n) \) the numbers \( d_{ij} = c_{i_1, \ldots, i_m} \) for \( i_1 = i \) and \( i_2 = \ldots = i_m = j, \ 1 \leq i, j \leq n \), can be taken. It is of interest that sublineal-time Algorithm \( A(\phi_n) \) applied to 3-AP has sublineal running time in the case \( m \)-APq, \( m > 3 \) as well. Moreover, in this case the running time of Algorithm \( A(\phi_n) \) normalized to the problem size decreases as the function \( n^{2-m} \) when \( m \) grows.

References

710