A Comparison study of two approaches for determining ranked set of assignments for multitarget tracking

L. V. Bojilov[1], , P. D. Konstantinova[1]

Central Laboratory for Parallel Processing, Bulgarian Academy of Sciences, acad. "G. Bonchev" str.bl. 25A, Sofia, Bulgaria, e-mail: bojilov@bas.bg pavlina@bas.bg

Abstract. In recent years there is strong interest in computing a ranked set of assignments of measurements to targets. In this report we propose the results of our investigations of two approaches for finding K-best ranked assignments for implementation in multitarget tracking. The first approach includes formulation of data association problem in multitarget tracking as a classical assignment problem, partitioning of all possible solution of this problem and also solving of range of modified forms of the starting problem or its sub problems. The second approach uses different frame describing the association problem as a table with rows and columns corresponding to measurements and targets. The sells of the table are fill with the association probabilities (densities) of any measurement with a given target. The comparison is made on the base of intensive numerical experiments.

1. Introduction.

A strong interest in recent years in determining a ranked set of solutions to assignment problem is connected with the possibility of this approach to be used in solving data association problem in multitarget tracking. The seminal algorithm for determining the ranked set of assignments is due to Murty [Murty]. On the base of optimal solution Murty suggests partitioning of all possible assignments of the problem. A range of subproblems are solved for finding the second best assignment. This second best assignment is used for another partitioning an so on. Recently, Miller, Stone and Cox have proposed an optimized algorithm of Murty achieving a considerable speedup [Mil&Stone&Cox].

Danchick and Newnam [Dan&New] have proposed an algorithm, which is based on the recognition that after determining the best assignment, to determine a ranked set of assignments may be accomplished by solving a series of modified copies of the initial assignment problem. Their suggestion is that the proposed algorithm makes Multiple Hypotheses Tracking (MHT) practical for the first time.

Different approach is used by Nag. Chin. Sharma [Nag&Chin&Sh]. They construct a probability table with rows and columns corresponding to measurements and targets. The sells of this table is fill with probability for any one measurement to originate from any one of the targets. Of course, using a gating technique some of the sells remain empty. The last column contain the probabilities for any measurements to originate from a new target. Dividing any row with the last element (new target sell) authors construct another table - probability ratios table. From this table they derive the preferred measurement table containing in any column the indices of all measurements falling in the target gate, sorted by their probability ratios. Now on, the algorithm proceeds with the last table originating the first K-best association hypotheses.

2. Problem formulation.

The investigated here algorithms are of two different types. We will expose the main ideas of these algorithms and will set forth the most important their steps. We will refer to the first algorithm as Murty's algorithm and to the second - Ngarajan's extended algorithm. The first algorithm is due to Katta Murty [Murty] and is the seminal algorithm in direction of finding firs K-best assignments of the classical assignment problem. The second algorithm is based on the frame proposed by Ngarajan *et al.* in [Nag&Chin&Sh] and extended in our paper [LVB]. The extended algorithm includes two optimizations of the original algorithm which produce a speedup of over a factor of 200.

2.1. Murty's algorithm.

This algorithm deals with the classical assignment problem. If a cost matrix of the task is $n \times n$ there exist $n!$ possible combinations. To denote the set of all possible combinations with A and one particular combination or assignment with $a(k)$. A numerous algorithm have been proposed for finding the optimal solution (say, minimal) of the assignment problem [Survey]. In our experiments we have used extended version of Munkres algorithm due to Bourgeoas and Lassalle [Bur&Lass]. The core idea of the author is to partition the set A into subsets $A_1, A_2, \ldots$ at given level and to find the best solution in any of the subsets which are mutually disjoint:

$$A_i \cap A_j = \varnothing, \quad i \neq j \ .$$

Algorithm starts with finding the optimal solution $a(1)$ with the indices $\left[(1, j_1), (2, j_2), \ldots, (n, j_n)\right]$. The task is to find the best solution among the set of assignments remaining after excluding the optimal assignment $a(2) \in \{A / a(1)\}$. Starting with the first element of the solution $a(1)$ we exchange it with unacceptable big value, so constructing a copy of initial matrix. If we solve a new assignment problem it is clear that all possible assignments with this element will be excluded from the decision. The new optimal decision will be the best decision among all assignments which number is

$$n! - (n-1)! = (n-1)(n-1)!$$

The next step is to repeat the same steps to the remaining *(n-1)!* assignments. We withdraw from the initial matrix the row and column corresponding to the first element and so ensuring its participation in the subsequent decisions. We reconstructing remaining sub matrix in the same way - we exchange it with unacceptable big value and repeat the same steps as above. In that way we obtain the best solution among *(n-2)(n_2)!* assignments out of remaining above *(n-1)!*. Continuing like this we reach the matrix with dimension 2×2 and finding its solution we terminate the stage of obtaining the second best solution. In this stage we have just found *(n-1)* best solutions in *(n-1)* mutually disjoint subsets of A and their union is

$$(n-1)(n-1)!+(n-2)(n-2)!+\cdots$$
$$+2\times2!+1\times1!=n!-1$$

In this union the only missing combination is optimal assignment, i.e. it is equal to the set $\{A/a(1)\}$. The best solution among found *(n-1)* is the second best solution in our problem. Repeating the steps of this stage with some additional rules we can find predetermined number of first K-best assignments.

2.2. Ngarajan's extended algorithm.

In this algorithm a different approach is used. This approach is directly derived from a multitarget tracking problem. Let to consider a given tracking problem with N tracked targets. To assume M measurements are received at a given scan. We can construct a table with columns corresponding to targets and rows corresponding to measurements referred to as probability table. The sells of the table contain the probabilities (or probability densities, or their logarithms) the given measurements to originate from corresponding target. There is additional column corresponding to the hypothesis that no one target raises a given measurement and it is originated from a new target. A preliminary assumption is that the probability of association any measurement with a new target is definitely less then the probability of associating it with a known target. If we divide every row by its last element we receive a probability ratios table. To accept that probability ratio in any sell of the latter table is $p(m,t_m)$. Our task is to find a hypothesis ψ of associating measurements to every target with highest probability

$$P(\psi)=\prod_{m=1}^{m_k} p(m,t_m).$$

To simplify the procedure of hypotheses generation we construct a new table - preferred measurement table:

Preference index	TARGETS			
	T1	T2	T3	T4
0	5	7	7	1
1	4	8	3	5
2	3	5	2	2
3	6	3	8	8
4	8	2	4	3
5	1	6	6	-
6	-	4	-	-
7	-	1	-	-

This table correspond to an example with four targets and eight measurements in their validation regions. The most left column contain the preference index and al the rest four columns correspond to any one of the targets. Any of these columns contains the numbers of the measurements falling in validation region of corresponding target sorted by their probability ratios to originate from this target. For example the highest probability hypothesis is $(5,7,7,1)$, however it is unfeasible: any measurement can be associated to one target only. Another hypothesis can be $(5,7,3,1)$, which is feasible. For convenience to express hypotheses as a sequence of preference indices: the highest probability hypothesis will be $(0,0,0,0)$ and the next hypothesis $(0,0,1,0)$. It is obvious that any one hypothesis can produce four different hypotheses every one with less probability then the origin. On the other hand, for the two hypotheses $(0,1,0,1)$ and $(3,1,2,0)$ no conclusion can drown regarding their probabilities. Using this frame the authors of [Nag&Chin&Sh] construct un algorithm of quick generation of hypotheses dividing them into two lists: a) feasible hypotheses in 'candidate hypotheses list', and b) unfeasible hypotheses in 'retained hypotheses list' for future processing. In addition, if consecutive hypothesis can be derived from any of the hypotheses out of the 'candidate hypotheses list'. If it is so, we discard it. If not, we check if this hypotheses for feasibility. If it is feasible we add it to the 'candidate hypotheses list' and compute their probability, of not, we include it to the 'retained hypotheses list'. As a result the authors achieve considerable reduction of processing time, avoiding generation of numerous hypotheses and avoiding computation of their probabilities. In our extension we manage to cut down positions for hypotheses generation and achieve additional reduction of processing time.

3.The experiments' frame.

For our experimental investigation we construct the next frame. In a square surveillance field we place in random manner n targets using the random generator of the computer. From the coordinates of any of the targets using predetermined σ and Gaussian random generator place n measurements and in addition k false alarms is drown, or $m=n+k$ measurements. We choose the σ large enough to construct scenario with overlapping validation regions and shared measurements. We produce a range of such scenarios solving the problem of finding *K*-best assignments using the Murty's algorithm. Starting with the same value of random generator seed we produce the same range of scenarios solving any problem of finding firs *K*-best associations using Nagarajan's extended algorithm.

The structure of data of the latter algorithm ensure obtaining decision in which a new targets probabilities are included and so, at any scan we can distinguish correctly updated targets and new targets.

To achieve the same result with the Murty's algorithm we have to construct the cost matrix of the problem in a special way. In this particular investigation we fill up the probability table (the first table) of Nagarajan's extended algorithm and the cost matrix of Murty's algorithm with negative logarithms of the probability densities of measurements and so the best hypotheses (and assignments) are hypotheses with the least values. The cost matrix of the Murty's contains three sub matrices. The left upper is square $m \times m$ diagonal matrix containing in its diagonal elements negative logarithms of probability densities any of the measurements to originate from a new target. The right upper is rectangular $m \times n$ matrix containing negative logarithms of probability densities of any of the measurements to originate any of the targets in which validation region the measurement is fell in. The third matrix is also rectangular with dimensions $n \times (m+n)$ and is placed below the firs two matrices. This matrix contain only elements with unacceptable high values and serves to add up the cost matrix to square matrix.

The frame of our experiments is, in some extent, different compare to most common used. Most often, the considered matrices are filled up directly with an array of random numbers. In our approach our matrices and tables are filled up with the values drawn from a particular multitarget scenario even though constructed in a random manner. The latter approach leads to tasks which are more close to real world problems. The only drawback using this approach is when the algorithms have to be compared with different levels of sparsity, whereas, in the former approach it is straightforward to receive exact the requisite percent of sparsity of a matrix. We avoid this drawback defining only three levels of sparsity controlling them with different values of σ .

4.Simulation results.

We have performed two groups of experiments. In the first group we choose the parameters of the task (standard deviation σ and gate threshold γ) so, to ensure low level of sparsity of the matrices' sells (i.e., high level of filling up - more then 66 percent). In the second group of experiments the level of sparsity is high - the filling up is no more then 25 percent. The experiments in the two groups are with three different dimensions of the tasks. For Ngarajan's extended algorithm the notation n/m means that the algorithm search the first K-best hypotheses in a table with n columns and m rows. For the Murty's algorithm the same notation means that the algorithm search the first K-best hypotheses in a square matrix with dimension $(n+m) \times (n+m)$. The results of the experiments are given in the tables below. Every value of the tables is averaged over the 50 Monte Carlo runs. For every set of runs for every one of the compared algorithms the same random number streams were used.

First group of experiments concerns scenarios with low level of sparsity. The numerical results are given in Table 1 and Table 2. As it can be seen, in more cases Murty's algorithm outperform Ngarajan's extended algorithm. An exception is the times for finding the first 100-best hypotheses (the last columns in the firs two tables). So, for low level of sparsity Murty's algorithm is preferable especially for scenarios with higher dimensions.

Table 1. Times in seconds for the Murty's algorithm

Scenario dimension	Number of hypotheses generated			
	10	20	50	100
9/16	0.02	0.043	0.122	0.283
10/18	0.08	0.058	0.142	0.351
11/20	0.047	0.08	0.207	0.44
12/22	0.077	0.105	0.259	0.568

Table 2. Times in seconds for the extended algorithm

Scenario dimension	Number of hypotheses generated			
	10	20	50	100
9/16	0.087	0.099	0.069	0.087
10/18	0.151	0.149	0.115	0.196
11/22	0.283	0.262	0.334	0.48
12/22	0.531	0.513	0.449	0.815

The second group of experiments concerns scenarios with comparatively high level of sparsity - the filling up is no more then 25 percent. The results are given in Table 3 and Table 4. It is obvious that for this kind of scenarios the extended algorithm is superior in time processing to Murty's algorithm, especially for longer ranges of hypotheses to be found.

Table 3. Times in seconds for the Murty's algorithm

Scenario dimension	Number of hypotheses generated			
	10	20	50	100
9/16	0.02	0.037	0.102	0.241
10/18	0.0.02	0.045	0.131	0.304
11/20	0.034	0.0.068	0.169	0.358
12/22	0.052	0.093	0.231	0.493

Table 4. Times in seconds for the extended algorithm

Scenario dimension	Number of hypotheses generated			
	10	20	50	100
9/16	0.004	0.005	0.008	0.012
10/18	0.027	0.017	0.013	0.024
11/20	0.04	0.033	0.051	0.041
12/22	0.093	0.085	0.07	0.078

From the results of the two groups of experiments the overall conclusion can be drawn as regard to work of the compared algorithms. The Murty;s algorithm is less sensitive to the dimensions of solving problems whereas Ngarajan's extended algorithm is more sensitive from this parameter in any of the experiments. As regard to number of hypotheses to be found, extended algorithm, as it was referencing in our previous work [LVB] , shows very weak dependence on this number, whereas in the Murty;s algorithm this dependence is much stronger. As a final inference, the both algorithms can be successfully implemented in problems where a range of firs K-best hypotheses have to be extracted. The Murty's algorithm is more appropriate for tasks with low level of sparsity whereas Ngarajan's extended algorithm is more acceptable for tasks with higher level of sparsity.

5. Conclusions.

References

1. Murty, K.G., An algorithm for finding all the assignments in order of increasing cost, *Operations Research*, 16(1968), 682-687.
2. Miller, M.L., H.S. Stone, I.J. Cox, Optimizing Murty's ranked assignment method, *IEEE Transactions on Aerospace and Electronic System*, 33, 3(1997), 851-862.
3. Danchick, R., G.E. Newnam, A fast method for finding the exact *N*-best hypotheses for multitarget tracking, *IEEE Transactions on Aerospace and Electronic System*, 29, 2(1993), 555-560.
4. Nagarajan, V., M.R. Chidambara, R.N. Sharma, Combinatorial problems in multitarget tracking - A comprehensive solution, *IEE Proceedings*, pt. F, 134, 1(1987), 113-118.
5. Bojilov, L.V., An improved version of a multiple targets tracking algorithm, *Information & Security*, 9,(2002), 130-140.q
6. Carpaneto, G., S. Martello, P. Toth, Algorithms and codes for the assignment problem, *Annals of Operations Research*, 13 (1988), 193-223.
7. Bourgeoas, F., J.C. Lassalle, An extension of Munkres algorithm for the assignment problem to rectangular Matrices, *Communications of the ACM*, Vol. 14, (1971), 802-804).