A Study of Clustering Applied to Multiple Target Tracking Algorithm[1]

Pavlina Konstantinova, Milen Nikolov, Tzvetan Semerdjiev

Abstract: *In this paper the effectiveness of two Data Association algorithms for Multiple Target Tracking (MTT) based on Global Nearest Neighbor approach are compared. As the time for assignment problem solution increases nonlinearly depending on the problem size, it is useful to divide the whole scenario on small groups of targets called clusters. For each cluster the assignment problem is solved by using Munkres algorithm. Results reveal that the computational time especially for large scenarios decreases significantly when clustering is used.*

Key words: *Multiple Target Tracking (MTT), Clustering , Data association, Global Nearest Neighbor (GNN), Assignment problem, Munkres algorithm.*

INTRODUCTION

The target tracking consists of two basic parts: data association and track filtering. The goal of the first part is to associate the received on each scan observations to existing tracks. This part is often considered as the most important because its result is crucial for overall tracking process. In the general case the problem is difficult since the targets may be closely spaced or not detected in successive scans or to move in large groups. In addition, the sensor returns of the targets may be imprecise due to unavoidable measurement noise, and incorrect due to clutter.

In recent years many data association algorithms have been developed [7,9,11]. Among them, the Suboptimal Nearest Neighbor (SNN), the Global Nearest Neighbor (GNN), the Joint Probabilistic Data Association (JPDA), and Multiple Hypothesis Tracking (MHT) algorithms are considered to be the most powerful techniques for this problem. These algorithms vary widely in their complexity and the resulting tracking performance.

In [9] an experiment with real stressful dynamic environment data is performed. In this experiment the six targets are fighters flying in prescribed routes and formations. For each test, the spacing between aircrafts in the group and between groups is varied. These closely spaced formations are used to establish a raw radar data base for MTT investigations. The results reveal that in general the GNN approach proves to be the best and the most robust. The used JPDA-based algorithms performed poorly in tracking closely spaced manoeuvring targets. In conclusion the authors recommend the GNN method for modern radar systems instead of the SNN methods, which are widely used in current radar systems. In [10] the assignment-based methods are shown to be very effective for data association. In [7] an efficient technique based on clustering is presented for reducing computational time by partitioning the assignment problem into smaller subproblems. A clustering technique is used to remove improbable candidate associations. Also, disjoining candidate assignments leads to reduced dimension – a "divide-and-conquer" technique where the whole problem is broken into several smaller ones.

When clustering is used instead of one big assignment problem a number of smaller assignment problems have to be solved. The Munkres algorithm have to be implemented a number of times but it is expected that the smaller problems will take less execution time and it is important to study the overall computational time for the two cases. It is advisable to test the performances of the algorithms for large complex scenarios.

[1] The research reported in this paper is partially supported by the Bulgarian Ministry of Education and Science under grant No I-1205/2002 and No I-1202/2002, and by Center of Excellence BIS21 grant ICA1-2000-70016.

The goal of our paper is to study the performance of the GNN data association algorithm realised with and without clusters.

PROBLEM FORMULATION

Using assignment algorithm for data association

Let consider n existing target's tracks. If m observations are received for the current scan they have to be assigned to the tracks, on the base of normalized distance function:

$$d_{ij}^2 = \tilde{z}_{ij}^T \; S^{-1} \tilde{z}_{ij}\,.$$

where $\tilde{z}_{ij} = \hat{z}_i - z_j$ is the innovation, $\hat{z}_i$ is the measurement prediction vector of track i , z_j - the observation vector with dimension M, and S is the innovation covariance matrix [3,5]. A threshold constant G is defined for the track gate, such that correlation is allowed if the following relationship is satisfied by the normalized distance function d_{ij}^2

$$d_{ij}^2 < G \qquad (1)$$

The quantity d_{ij}^2 is a sum of squares of M independent Gaussian random variables with zero means and unit standard deviations and has χ_M^2 distribution for correct observation-to-track pairings with M degrees of freedom and allowable probability P of a valid observation falling outside the gate. The threshold constant G can be defined from the table [2] of the chi-square χ_M^2 distribution with M degrees of freedom and allowable probability of a valid observation falling outside the gate.

If the relationship (1) is satisfied the observation is considered in the gate.

Let i =1,…,n is a particular track from the set of existing tracks and j=1,…,m is a particular observation from the latest set of observations. Define binary assignment variable

$$\zeta_{ij} = \begin{cases} 1 & if\ the\ observation\ j\ is\ assigned\ to\ track\ i \\ 0 & otherwhise \end{cases}$$

Then the problem of finding assignment of observations to tracks that minimizes the total distance function can be presented as the 2D assignment problem [10]:

$$\min \sum_{i=1}^{n} \sum_{j=1}^{m} c_{ij} \zeta_{ij} \quad \text{subject to} \quad \begin{cases} \sum_{j=1}^{m} \zeta_{ij} = 1 & i = 1,...,n \\ \sum_{i=1}^{n} \zeta_{ij} = 1 & j = 1,...,m \end{cases}$$

where the cost of assigning observation j to track i is:

$$c_{ij} = \begin{cases} d_{ij}^2 & if\ the\ observation\ is\ in\ the\ track's\ gate \\ \infty & otherwhise \end{cases}.$$

The objective of the assignment method is to assign k observations to k tracks, where $k = \min(n, m)$ in such a way that the sum of k normalized distance functions is minimized. One solution is to find all possible $l!/(l-k)!$ assignments, where $l = \max(n, m)$ and to compute the total distance for each one. Because the computational cost can be prohibitive other methods are used. One of these methods is

Munkres' algorithm [4]. Instead of factorial requirements of the brute force enumeration the requirements of the Munkres' algorithm are on the order of $k^2 l$. If in the attempt to maximize the number of assignments, the algorithm chooses a pairing that does not satisfy the gating condition, the assignment is removed.

ALGORITHM DESCRIPTION

The main blocks of the program are:

1.Clustering
2 For each cluster:
  2.1 Initialization of the assignment matrix
  2.2.Filling up the assignment matrix and solving the assignment problem
  2.3.Checking the validity of the solution and making associations
3.Track filtering

The cluster by definition is a set of closely spaced objects. In our case if two tracks have an observation in their overlapping parts of the gates the tracks form cluster i.e. their clusters are merged in supercluster. The maximal number of clusters can not be more than the number of tracked tracks.

Clustering procedure description

The input parameters for clustering procedure are the array of the existing tracks and array of received observations for the latest scan. The clustering procedure is as follows:

```
For each observation received in the current scan
  NumOfGates=0; //the number of gates in which the observation is fallen
  For each track
     If  the observation is in the track gate
        NumOfGates= NumOfGates+1;
        If the track is not included in cluster form new cluster for the track
           NumOfClusters= NumOfClusters + 1;
           Write the observation in track list.
           If NumOfGates > 1 i.e. the observation falls in the gate of other track
              If OldCluster ≠ <track's cluster>  then MERGE clusters
              If track's cluster is not the last – compress cluster's array
           else
              OldCluster=Track's cluster
  End for each track
End for each observation
Form clusters for tracks without observations (to be filtered "by memory")
```

Track filtering

Track filtering is performed using Converted Measurement Kalman Filter (CMKF) [3] with Interacting Multiple Model [1,6] including two nested models with different process noise. The first model with lower level process noise is intended to ensure more precise evaluation of the updated target's state, the other with higher level of noise keeps larger gate in manoeuvring track segments to prevent missed detections.

PROGRAM REALISATION

The program was first written in MATLAB to use the MATLAB environment for convenient matrix operations. After first test results the program is rewritten in Microsoft Visual C++ to reach fast implementation. For this purpose an object-oriented program package for vector-matrix operations intended for Kalman Filtering is used [8]. In such a way very small changes need to receive effective C++ source code from MATLAB source. An example for comparison source code of one and the same function in MATLAB and MSVC++ is given in table 1.

Table 1: Example of the source code in MATLAB and C++ using **Object-Oriented Vector-Matrix Library** for Kalman Filtering

*Source code in **MATLAB***

```
function [X,P]=KFupdating(X,H,R,dz,ObsIC,P,S,p0,fPDA);

W  = zeros(4,2);
A  = zeros(4,4);
uP = zeros(4,4);
I  = eye(4);                   % Unity matrix 4 x 4

W  = P*H'*inv(S);              % Estimated gain
A  = I - W*H;                  % Joseph's form for P
uP = A*P*A' + W*R*W';
X  = X + W*dz;                 % Updated state vector

if fPDA == 1
   dP = W*(ObsIC - dz*dz')*W';
   P  = p0*P +(1-p0)*uP + dP;  %Updated state cov.matrix
else
   P = uP;
end;
```

*Source code in **C++***

```
void KFupdating(Nvector & X, MNmatrix &H, MMmatrix & R,
Mvector & dz,  MMmatrix & ObsIC,NNmatrix & P,MMmatrix &
S, double p0, bool fPDA)
{   NMmatrix W;       W.init(0);
    NNmatrix uP, dP, A, I;
    I.eye();                    // Unity matrix 4 x 4

    W  = P*transp(H)*inv(S);    // Estimated gain
    A  = I - W*H;               // Joseph's form for P
    uP=A*P*transp(A) + W*R*transp(W);
    X  = X + W*dz;               // Updated state vector
    if ( fPDA == 1)
    { dP=W*(ObsIC - dz*transp(dz))*transp(W);
      P= p0*P +(1-p0)*uP + dP;//Updated state cov.matrix
    }
      else
    P = uP;
}
```

SIMULATION RESULTS

The full test scenario (Fig. 1.) consists of 8 groups of targets, each group consists of 21 moving targets. The targets move from south to north with velocity 250 [m/sec] manoeuvre during scans 25 to 27 and 37-39 with transversal acceleration 0.55 g. The initial heading of the odd numbered groups is 13 degrees from the North and for the even numbered groups the heading is 347 degrees. The sampling period is 5 sec.

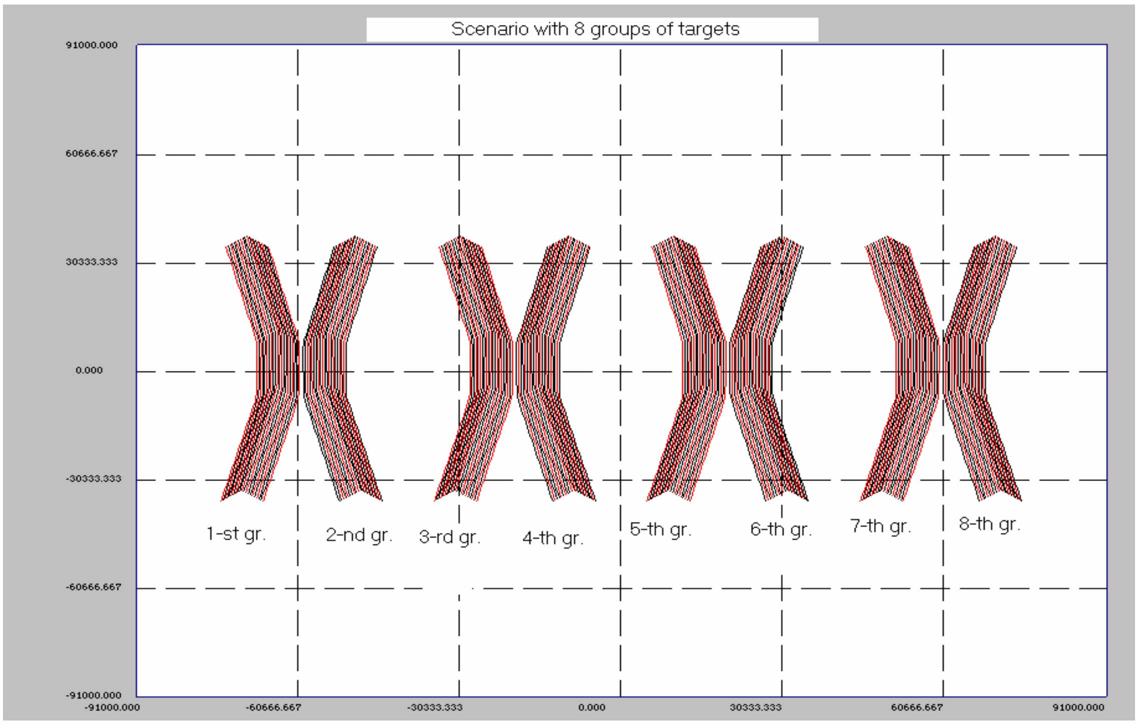

Fig1. Test scenario with eight groups of manoeuvring targets

Eight different experiments have been run for 60 scans. In each run different number of groups are included. The tests are implemented on 4 different computers with different processors. The results shown in table 2 reveal that using clustering speedups the overall tracking performance. The execution time for the whole tracking process – including data association and filtering, is less than the sampling time. The unused part of the sampling time interval can be utilized for processing additional attribute information [11].

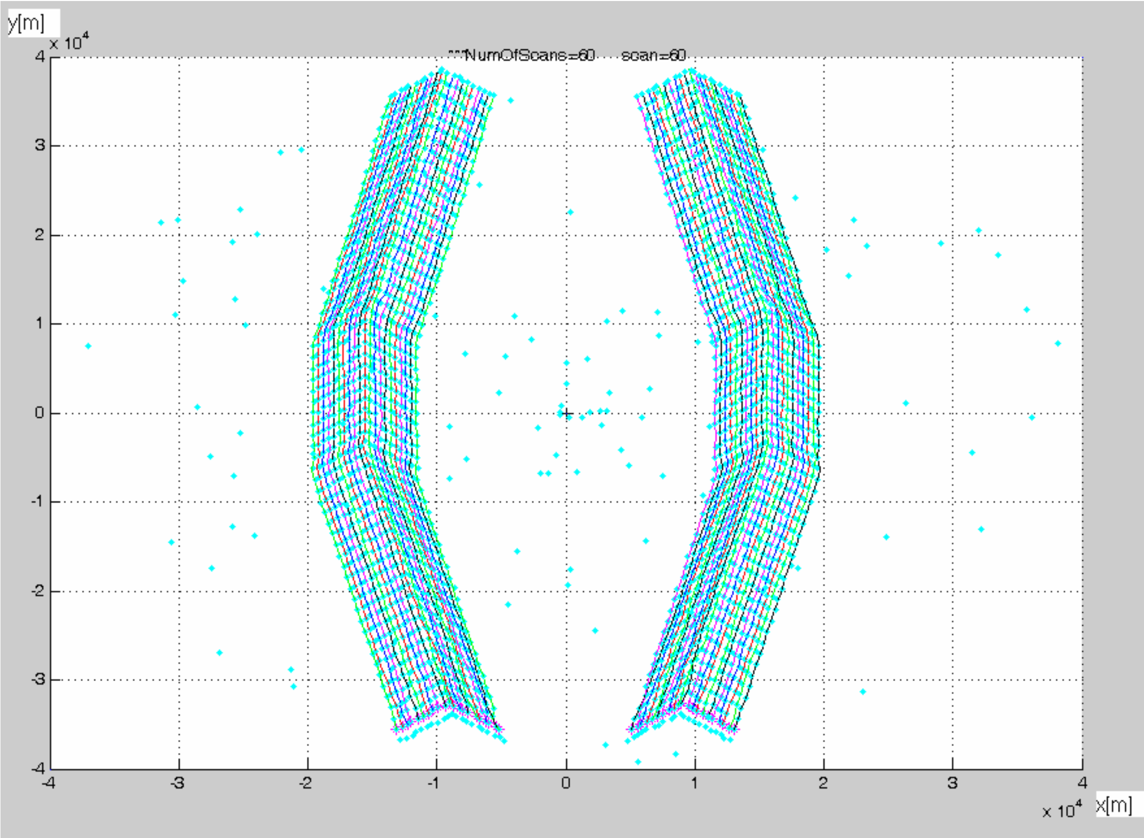

Fig2. Implementation of test scenario with two groups of targets in MATLAB

Table 2: Execution time in seconds for data association and filtering for 60 scans of the different parts of scenario 1 (fig.1)

No of experi-ments	Number of groups	Number of targets	Execution time [sec]							
			Pentium S, 133 MHz		Pentium 2, 266 MHz		Celeron, 300 MHz		AMD Athlon, 1050 MHz	
			without clusters	with clusters	without clusters	with clusters	without clusters	with clusters	without clusters	with clusters
1	8	168	122.02	67.22	47.89	22.68	42.53	27.31	10.11	5.60
2	7	147	79.15	54.32	28.89	18.80	28.71	21.68	6.43	4.45
3	6	126	53.16	42.10	17.87	13.95	20.62	16.89	4.34	3.46
4	5	105	38.94	31.92	12.42	10.43	14.79	12.68	3.14	2.58
5	4	84	26.36	23.95	8.24	7.36	10.12	8.98	2.14	1.81
6	3	63	16.76	16.37	5.90	4.78	6.29	5.80	1.32	1.15
7	2	42	9.23	8.73	2.75	2.70	3.51	3.42	0.66	0.66
8	1	21	3.80	3.80	1.10	1.10	1.50	1.50	0.22	0.22

CONCLUSIONS AND FUTURE WORK

The performances of two algorithms for data associations based on global nearest neighbour approach are compared. The simulation results show significant computational savings over the assignment approach without clustering especially for large scenario with many targets. The execution time of the C++ program is less than the sampling time period of the sensor observation and is suitable for real time applications. The unused part of the sampling time interval can be utilized for processing additional attribute information.

REFERENCES

[1] Angelova D., Vesselin Jilkov, Tzvetan Semerdjiev, Tracking Maneuvering Targets by Interacting Multiple Models, Comptes rendus de L'Academie Bulgare des Sciences, vol 49, 1996, pp.37-40.

[2] Bar-Shalom Y, Fortmann T., Tracking and Data Association, Academic Press, 1988.

[3] Bar-Shalom Y, Xiao-Rong Li, Multitarget-Multisensor Tracking: Principles and Techniques, New York, YBS, 1995.

[4] Blackman S. Multiple – Target tracking with Radar Applications, Artech House, 1986.

[5] Blackman S., R. Populi. Design and Analysis of Modern Tracking Systems, Artech House, 1999

[6] Bojilov L., Alexiev K., Konstantinova P., "An algorithm unifying IMM and JPDA approaches", Comptes rendus de L'Academie Bulgare des Sciences, vol 56, No 12, 2003, pp.53-60.

[7] Chummun M. R., Kirubarajan T., Pattipati K.R., Bar-Shalom Y., -Fast Data Association Using Multidimensional Assignment With Clustering, IEEE trans. On Aerospace and Electronic Systems, vol. 37, No 3 July 2001, pp.898-912

[8] Djerassi E., Konstantinova P., "Object-Oriented Environment for Assessing Tracking Algorithms", Information and Security – An International Journal. Vol. 9: "Sensor Data Fusion", Sofia, 2002, pp. 93-106;.

[9] Leung H., Zhijian Hu, M.Blanchette, Evaluation of Multiple Radar Target Trackers in Stressful Environments, IEEE Trans. On Aerospace and Electronic Systems, vol. 35, No 2, April 1999.

[10] Popp R., Pattipati K., Bar-Shalom Y., "Dynamically Adaptable m-Best 2-D Assignment Algorithm and Multilevel Parallelization", IEEE trans. on AES, Oct. 1999, pp. 1145-1158.

[11] Tchamova A., Dezert J., Semerdjiev Tzv., Konstantinova P., Target Tracking with generalized Data Assocoation based on the General DSm Rule of Combinaton, Proceedings of Fusion 2004, Stockholm, Sweden, June 28th-July 1st 2004.

[12] Zhou B., Bose N.K., "Multitarget Tracking in Clutter: Fast Algorithms for Data Association", IEEE Trans. on Aerospace and Electronic Systems, vol.29, No 2, April 1993, pp.352-363.

ABOUT THE AUTHORS

Dr. Pavlina Konstantinova, PhD, senior researcher at Institute for Parallel Processing, BAS, Sofia, Phone: +359 979-66-20, E-mail: pavlina@bas.bg.

Milen Nikolov, PhD-student, Institute for Parallel Processing, BAS, Sofia, Phone: +359 979-66-20, E-mail: milenkec@yahoo.com.

Prof. Tzvetan Semerdjiev, D.Sc., Institute for Parallel Processing, BAS, Sofia, Phone: +359 979-67-47, E-mail: signal@bas.bg.