Computational Requirements of a Non-combinatorial Detection of Multiple Targets in High GMTI Clutter

Leonid Perlovsky°, Ross Deming+, Simon Streltsov*, Sergey Petrov*, Ilya Muchnik*

°Air Force Research Laboratory/SNHE, 80 Scott Drive, Hanscom AFB, MA 01731 Leonid.Perlovsky@hanscom.af.mil

+Anteon Corp. 5100 Springfield Pike, Suite 509, Dayton, OH, 45431, onsite consultant to AFRL/SNHE
Ross.Deming@hanscom.af.mil

*LongShortWay, Inc., 10 Dana St, Ste 508 Cambridge MA 02138
{simon, sergey, muchnik}@longshortway.com

Non-combinatorial tracking

The major computational challenge of multi-target tracking in clutter is solving the report-to-track association problem. Multiple hypothesis tracking (MHT) [4] is an approximate solution that tries to limit the combinatorial complexity of assignment data to models in multiple frames. MHT processing and memory requirements grow exponentially with the increased number of frames used to resolve the associations [5]. In addition, a real-time realization of an MHT tracker is difficult due to the complexity of data movement required to manage track hypotheses. This data movement leads to inefficient utilization of embedded processors.

Dynamic Logic (DL) algorithm [1] performs data association without combinatorial complexity. The approach is designed for multi-target high-clutter scenarios in which combinatorial trackers have impractically high complexity. Thus, DL tracker can operate on GMTI data with low detection thresholds and detect very low signal tracks. DL algorithm works directly on contiguous blocks of data making it suitable for embedded applications.

In this paper, we study computational complexity and real-time requirements of multi-target track detection in high GMTI clutter by DL algorithm.

Dynamic Logic tracking algorithm

The Dynamic Logic tracking algorithm is described in [1]. Dynamic Logic maximizes likelihood of batch frames of measurements over possible target trajectories. Direct maximum likelihood methods often require a search over a very fine grid in order to find an initial point of a local optimization algorithm because the likelihood is a multi-dimensional function of the parameters describing the target trajectory with large number of local maxima [6,7,8]. Dynamic Logic algorithm introduces a fuzziness parameter in the likelihood that enables fast convergence without a need for the expensive grid search.

The algorithm maximizes a product of the Gaussian density functions mixture that models cumulative, possibly non-thresholded, measurements from all available frames. The maximization criterion also includes unknown weights of multiple target models S and the fuzziness parameter σ that corresponds to sensor uncertainty (Equation 1). Here $S_k = (x_{0k}, y_{0k}, v_{xk}, v_{yk})$ are track models and r_k – track weights; σ is the fuzziness parameter. [2] modifies the DL algorithm to process observations from multiple moving platforms.

$$L = \sum_n \ln \sum_k r_k G(I_n \mid I_k, CI_k)\ G(x_n \mid S_{kn}, CS_k, \sigma)$$

Equation 1: Log Likelihood

Tracks are detected in a unified optimization process that gradually decreases the fuzziness parameter of the optimization criterion and improves estimates of track parameters and track-signal association. The Dynamic Logic algorithm consists of the following steps (Figure 1):

1. Optimize L over target model parameters S for a fixed value of parameter σ and association variables,

2. Perform an expectation-maximization step over unknown associations,

3. Gradually decrease the value of σ.

At every iteration step, the algorithm simultaneously finds optimal solutions of likelihood L(σ) and decreases the value of the parameter σ. Optimization at smaller s starts with the previously found solution for larger s. As a result, the algorithm arrives at the solution of the original global optimization problem by solving multiple local optimization problems.

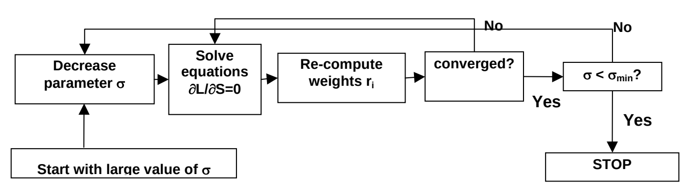

Figure 1: Dynamic Logic Tracker.

Track Detection Performance

Figure 2 demonstrates evolution of the probability function of the estimated tracks for 1st, 5th, 10th and 20th iteration of the algorithm. Algorithm gradually decreases fuzziness of the PDF and uncovers multiple targets hidden in the clutter.

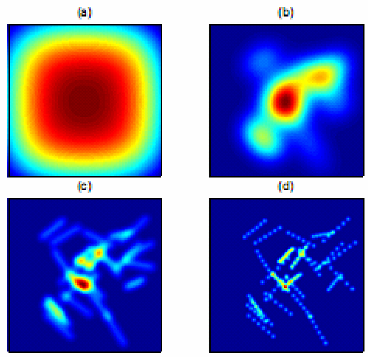

Figure 2: Evolution of the track model during 20 iterations

DL ability to process large number of frames leads to further improvement in detection performance. Figure 3 shows effect of increasing number of frames on detection probability.

In this paper, we study computational complexity of multi-target track detection in high GMTI clutter by DL algorithm.

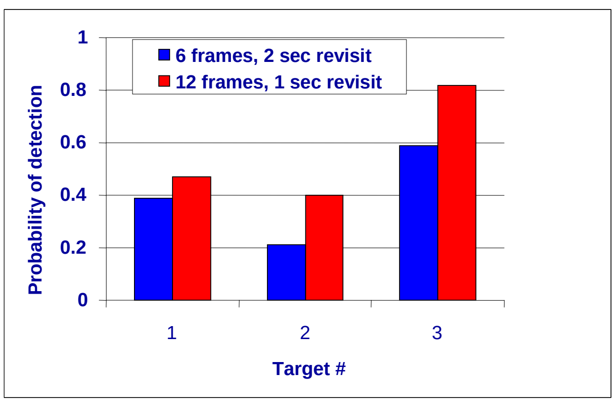

Figure 3: Detection probability

Computational Complexity

In this section, we analyze computational complexity of the DL tracker.

Complexity of each iterative likelihood estimate is linear in number of GMTI detections (and, thus, number of GMTI frames), in number of unknown target parameters (2-D position and velocity for the constant velocity model, RCS) and number of estimated tracks and clutter models. Number of algorithm iterations with the reducing fuzziness parameters is shown experimentally to be low (between 10 and 50). Furthermore, later iterations with decreased fuzziness require processing only part of the data.

Real-time requirements

Algorithm processes simultaneously data from multiple GMTI frames in the surface area several times larger than the possible targets moving through the time of collected data. Raw GMTI data is reduced in size by low amplitude and Doppler thresholds. All consequent operations (Figure 1) are done on full blocks of data requiring no data movement. Each data point in the reduced dataset is described by 4 floating point parameters (range, cross-range, amplitude, range rate). Computations represent identical arithmetic operations on spatially and temporally consequent data points. Therefore, these computations can be efficiently performed on vector processors. In order to avoid memory bottlenecks, the amount of simultaneously processed data should be limited to L1 cache capacity.

Processor and memory requirements can be specified given the GMTI acquisition rate and amount of clutter allowed through the detection threshold.

References

[1] Leonid I. Perlovsky "Neural Networks and Intellect: Using Model-Based Concepts", New York, Oxford University Press, p. 469, 2001

[2] Ross Deming. Leonid Perlovsky "Concurrent Detection and Tracking for GMTI," ASAP Proceedings, MIT Lincoln Laboratory, 2006

[3] Sergey Petrov, Ilya Muchnik, Simon Streltsov "Dynamic Logic Algorithms for Multi-Target GMTI Tracking in Clutter," Phase I SBIR report, contract Fa8718-05-C-0051, LongShortWay Inc. 2006

[4] Thomas Kurien, "Issues in the Design of Practical Multitarget Tracking Algorithms," In Multitarget-Multisensor Tracking: Advanced Applications, 43-83 Y. Bar-Shalom (ed.), Artech House, Norwood, MA, 1990.

[5] Streltsov, S., "Airborne Learning Algorithms. A Case Of a Ground Target Multiple Hypothesis Tracker", Proceedings of IEEE KIMAS, 2003, p. 190.

[6] Sergey Petrov, Ilya Muchnik, Simon Streltsov "Dynamic Logic Algorithms for Multi-Target GMTI Tracking in Clutter," Phase I SBIR report, contract Fa8718-05-C-0051, LongShortWay Inc. 2006

[7] Tonissen, Y. Bar-Shalom, "Maximum Likelihood Track-Before-Detect With Fluctuating Target Amplitude." IEEE Trans on Aerospace and Electronic Systems v 34 n 3 p 796 1998

[8] Pohlig, S. C. "An algorithm for detection of moving optical targets," IEEE Transactions on Aerospace an Electronic Systems, 25, 56-63, 1989