A track-before-detect algorithm with successive track cancellation

Emanuele Grossi, *Member, IEEE,* Marco Lops, *Senior Member, IEEE,* and Luca Venturino, *Member, IEEE*
DIEI, Università degli Studi di Cassino e del Lazio Meridionale, Italy 03043
Email: e.grossi@unicas.it, lops@unicas.it, l.venturino@unicas.it

***Abstract*—In this work we introduce a novel track formation procedure to overcome the performance losses of the two-stage track-before-detect architecture in [1] when targets are closely-spaced. The proposed strategy adopts a dynamic programming algorithm which successively estimates the trajectories of prospective targets and removes the corresponding measurements from the input data, and which does not need a discretization of the state space. Experimental results demonstrate that its detection capabilities are almost independent of the mutual distance of the targets and equal to those of the system in [1] when targets are widely-spaced.**

I. Introduction

Multi-frame detection (MFD), where multiple, consecutive scans (or *frames*) are jointly elaborated before thresholding, is an effective strategy used in radar applications to detect targets with low signal-to-disturbance ratio (SDR). In the presence of moving targets, MFD requires track-before-detect (TBD) techniques to correctly correlate data over time [1]–[12]. The main limitation of MFD/TBD is the heavy computational cost when the number of sensor resolution elements is *large* and target maneuvers are allowed.

In order to limit complexity and allow real-time implementation, the authors in [1] proposed the detection architecture shown in Fig. 1. At each scan n, the Detector and Plot-Extractor receives the raw data collected by the sensor and produces a list of candidate detections (or plots), say $\boldsymbol{S}_n$. The plot-list $\boldsymbol{S}_n$ is stored in a matrix, whose k-th row is the k-th plot at scan n, defined as $\boldsymbol{s}_{k,n} = (t_{k,n}\ p_{k,n}\ A_{k,n}\ N_{k,n})$, with $t_{k,n}$ the time instant at which the plot has been taken, $p_{k,n}$ the position measurement (usually the pair range-azimuth, but it may also include range-rate and/or elevation measurements if available), $A_{k,n}$ the amplitude of the received signal, and $N_{k,n}$ the power of the disturbance (thermal noise plus clutter). The number of plots at scan n is D_n, and $\boldsymbol{S}_n$ is not defined if $D_n = 0$. The threshold γ_1 of the Detector and Plot-Extractor is set lower than that used in the single-frame detector, causing an increment both in the probability of detection (PD) and in the false alarm rate (FAR), which is the average number of false alarms per minute. The second stage correlates the plots in the current plot-list with those in the past $L-1$ plot-lists and confirms or discard each plot in $\boldsymbol{S}_n$ through a secondary threshold γ_2: the goal is to restore FAR to its original level while maintaining part (if not all) of the PD gain.[1]

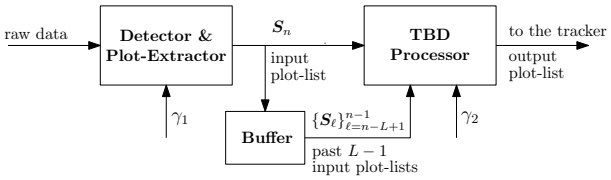

Figure 1. Detection architecture considered in [1].

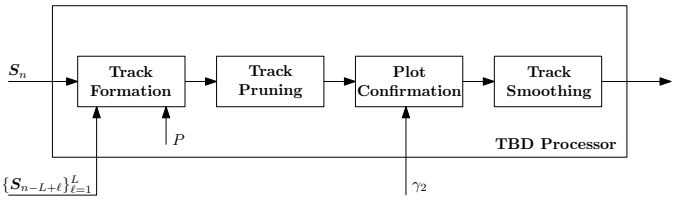

Figure 2. TBD Processor considered in [1].

The TBD processor considered in [1] (and originally proposed in [13], [14]) consists of four blocks, as shown in Fig. 2: the Track Formation stage, which correlates data from multiple scans, the Track Pruning stage, which solves possible data association problems, the Plot Confirmation stage, which compares the decision statistics with γ_2, and the Track Smoothing stage, which improves the measurement accuracy of confirmed plots. This scheme shows some drawbacks in the presence of multiple, closely-spaced targets, in that strong targets may shadow weak ones. A way to cope with this situation is to enlarge the search space to all the t-uples of admissible trajectories, t being the number of closely spaced targets. This solution, however, is hardly implementable, for it requires large computational resources and prior knowledge of t.

In this work, we elaborate on the Track Formation and the Track Pruning stages proposed in [1] and derive a novel procedure, which successively estimates candidate target trajectories and removes the corresponding measurements from the input plot-lists, so that enlargement of the search space is not required. The basic idea is inspired to the successive in-

This works was carried out under a research contract sponsored by Selex Galileo, Nerviano (MI), Italy.

[1]This scheme is general enough to subsume both standard single-frame detector (when $\gamma_1 = -\infty$ and $L = 1$), and classical MFD with raw input data (when $\gamma_1 = -\infty$ and $L \geq 2$).

978-1-4673-3146-3/13/$31.00 ©2013IEEE

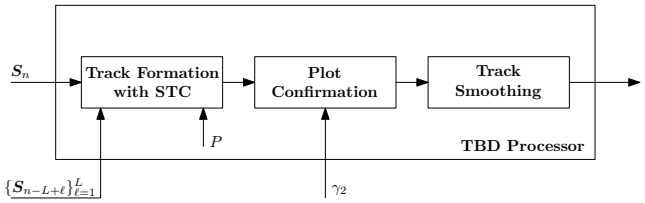

Figure 3. TBD Processor with successive track cancellation.

terference cancellation principle used in communications [15] and adapted to radar systems in [3], [16], [17]. However, differently from these works, at each estimation step, trajectories are formed using a dynamic programming algorithm which does not need a discretization of the measurement space, but directly operates on the plot-lists, thus taking advantage of the fact that the number of plots can be much smaller than the number of resolution elements. The procedure is able to improve the system performance in terms of PD and root mean square error (RMSE) on the estimated target position whenever closely-spaced targets are present in the monitored area with limited complexity increment, while providing the same performance of the procedure in [1] for widely spaced targets.

In the next section, the track formation procedure with successive track cancellation (STC) is described. Some experimental results are discussed in Section III. Concluding remarks are given in Section IV.

II. Track Formation with STC

Some estimated trajectories at the output of the Track Formation stage in Fig. 2 may share a common root. This may be deleterious, since target echoes may be responsible for the confirmation of the alarms they caused *and* of the false alarms in their proximity; moreover, the estimated trajectories of weak targets may erroneously contain plots of strong, close targets. The Track Pruning stage in Fig. 2 tries to solve these data association problems *a posteriori*, i.e., after all candidate target trajectories have been formed. Here we merge the tasks of these two stages and derive a procedure that solves any data association problems while trajectories are formed. Specifically, an initial set of candidate target trajectories is formed starting from the input plot-lists, the *dominant trajectories* (i.e., those with largest metric among all trajectories sharing the same root) are extracted, and the corresponding plots are removed from the input data; the process is iterated until there are no trajectories sharing a common root. This procedure is implemented by Algorithm 1 and is referred to as Track Formation with STC; the resulting TBD processor is shown in Fig. 3.

We now illustrate in detail the operations performed by Algorithm 1. Without loss of generality, assume that the current scan is $n = L$. If a target is present, its trajectory can be uniquely specified by an L-dimensional index vector, say $\boldsymbol{\xi}$,

Algorithm 1 Track Formation with STC

Input: $\{\boldsymbol{S}_\ell\}_{\ell=1}^{L}$ **Output:** $\{\boldsymbol{\tau}_{k,L}, F_{k,L}\}_{k=1}^{D_L}$

1: $i = 1$
2: **for** $\ell = 1, \ldots, L$ **do**
3: $\quad H_\ell^{(i)} = \{1, \ldots, D_\ell\}$
4: **end for**
5: $\{\boldsymbol{\tau}_{k,L}^{(i)}, F_{k,L}^{(i)}\}_{k\in H_L^{(i)}} =$ Track-Formation$\big(\{H_\ell^{(i)}, \boldsymbol{S}_\ell\}_{\ell=1}^{L}\big)$
6: $\{K_\ell^{(i)}\}_{\ell=1}^{L}, \{\bar{\boldsymbol{\tau}}_{k,L}^{(i)}\}_{k\in K_L^{(i)}}$
$\qquad =$ Track-Extraction$\big(\{\boldsymbol{\tau}_{k,L}^{(i)}, F_{k,L}^{(i)}\}_{k\in H_L^{(i)}}\big)$
7: $A = \{\bar{\boldsymbol{\tau}}_{k,L}^{(i)}\}_{k\in K_L^{(i)}}$
8: **while** $K_L^{(i)} \neq H_L^{(i)}$ **do**
9: $\quad i = i + 1$
10: $\quad$ **for** $\ell = 1, \ldots, L$ **do**
11: $\quad\quad H_\ell^{(i)} = H_\ell^{(i-1)} \backslash K_\ell^{(i-1)}$
12: $\quad$ **end for**
13: $\quad \{\boldsymbol{\tau}_{k,L}^{(i)}, F_{k,L}^{(i)}\}_{k\in H_L^{(i)}} =$ Track-Formation$\big(\{H_\ell^{(i)}, \boldsymbol{S}_\ell\}_{\ell=1}^{L}\big)$
14: $\quad \{K_\ell^{(i)}\}_{\ell=1}^{L}, \{\bar{\boldsymbol{\tau}}_{k,L}^{(i)}\}_{k\in K_L^{(i)}}$
$\qquad\qquad =$ Track-Extraction$\big(\{\boldsymbol{\tau}_{k,L}^{(i)}, F_{k,L}^{(i)}\}_{k\in H_L^{(i)}}\big)$
15: $\quad A = A \cup \{\bar{\boldsymbol{\tau}}_{k,L}^{(i)}\}_{k\in K_L^{(i)}}$
16: **end while**
17: $\{\boldsymbol{\tau}_{k,L}, F_{k,L}\}_{k=1}^{D_L} =$ Track-Validation$\big(A, \{\boldsymbol{S}_\ell\}_{\ell=1}^{L}\big)$

defined as follows:[2] if $\xi_\ell = k$, with $k \in \{1, \ldots, D_\ell\}$, the target has been observed at scan ℓ, and its plot is $\boldsymbol{s}_{k,\ell}$, while if $\xi_\ell = 0$, the target has been missed at scan ℓ. Following [1], the metric associated with the trajectory indexed by $\boldsymbol{\xi}$ is $\sum_{\ell=1}^{L} z_{\xi_\ell,\ell}$, where

$$z_{k,\ell} = \begin{cases} A_{k,\ell}^2/N_{k,\ell}, & \text{if } k \in \{1, \ldots, D_\ell\} \\ \eta, & \text{if } k = 0 \end{cases}$$

so that it is related to the energy back-scattered by a prospective target, η being a parameter accounting for a missing observation (which is set at the desin stage).

With these definitions, $H_\ell^{(i)}$ (lines 3 and 11 of Algorithm 1) is the set containing the indexes of the plots available at scan ℓ and iteration i to form prospective target trajectories. Starting from the plots indexed by $\{H_\ell^{(i)}\}_{\ell=1}^{L}$, Function Track-Formation (lines 5 and 13) implements a dynamic programming algorithm to estimate the *best* trajectory ending in each alarm at the current scan L and compute the corresponding metric. Specifically, it returns

$$\boldsymbol{\tau}_{k,L}^{(i)} = \arg\max_{\boldsymbol{\xi}\in\mathcal{R}_{k,L}^{(i)}} \sum_{\ell=1}^{L} z_{\xi_\ell,\ell} \quad \text{and} \quad F_{k,L}^{(i)} = \max_{\boldsymbol{\xi}\in\mathcal{R}_{k,L}^{(i)}} \sum_{\ell=1}^{L} z_{\xi_\ell,\ell}$$

for all $k \in H_L^{(i)}$, where $\mathcal{R}_{k,L}^{(i)}$ is the set containing the vectors indexing all trajectories ending in $\boldsymbol{s}_{k,L}$ that can be formed with the plots indexed by $\{H_\ell^{(i)}\}_{\ell=1}^{L}$. These trajectories must

[2]Here polar (or sensor) coordinates are used for track formation, and velocities are not considered to limit complexity. Therefore, a trajectory is a sequence of consecutive plots taken during L scans.

Function 2 Track-Formation

Input: $\left\{H_\ell^{(i)}, \boldsymbol{S}_\ell\right\}_{\ell=1}^{L}$ **Output:** $\left\{\boldsymbol{\tau}_{k,L}^{(i)}, F_{k,L}^{(i)}\right\}_{k\in H_L^{(i)}}$

1: **for** $k \in H_1^{(i)}$ **do**
2: $\quad F_{k,1}^{(i)} = A_{k,1}^2/N_{k,1}$
3: $\quad \boldsymbol{\tau}_{k,1}^{(i)} = k$
4: **end for**
5: **for** $\ell = 2, \ldots, L$ **do**
6: $\quad$ **for** $k \in H_\ell^{(i)}$ **do**
7: $\quad\quad \mathcal{M}_{k,\ell}^{(i)} = \big\{(j,p) : p \in \big\{\max\{1, \ell - P - 1\}, \ldots, \ell - 1\big\}, j \in H_p^{(i)}$, and $(\boldsymbol{s}_{j,p}, \boldsymbol{s}_{k,\ell})$ satisfies the velocity constraint$\big\}$
8: $\quad\quad$ **if** $\mathcal{M}_{k,\ell}^{(i)} \neq \varnothing$ **then**
9: $\quad\quad\quad (h,m) = \arg\max_{(j,p)\in\mathcal{M}_{k,\ell}^{(i)}} F_{j,p}^{(i)}$
10: $\quad\quad\quad F_{k,\ell}^{(i)} = F_{h,m}^{(i)} + (\ell - m - 1)\eta + A_{k,\ell}^2/N_{k,\ell}$
11: $\quad\quad\quad \boldsymbol{\tau}_{k,\ell}^{(i)} = \big(\boldsymbol{\tau}_{h,m}^{(i)} \underbrace{0 \cdots 0}_{\ell-m-1} k\big)$
12: $\quad\quad$ **else**
13: $\quad\quad\quad F_{k,\ell}^{(i)} = (\ell - 1)\eta + A_{k,\ell}^2/N_{k,\ell}$
14: $\quad\quad\quad \boldsymbol{\tau}_{k,\ell}^{(i)} = \big(\underbrace{0 \cdots 0}_{\ell-1} k\big)$
15: $\quad\quad$ **end if**
16: $\quad$ **end for**
17: **end for**

satisfy the constraints on the maximum target speed (which is set at the design stage) and on the maximum number of consecutive misses (i.e., of consecutive zeros in $\boldsymbol{\xi}$), say P, between two detections (needed to avoid large *holes* in the trajectory). This is a modified version of Algorithm 1 in [1] to account for the fact that at each iteration the set of input plots indexed by $\left\{H_\ell^{(i)}\right\}_{\ell=1}^{L}$ changes, and its implementation is reported in Function 2. Function Track-Extraction (lines 6 and 14) selects, among all trajectories sharing the same root (i.e., the first plot), the one with largest metric (i.e., the dominant one). All dominant trajectories extracted at iteration i are included in the set A (lines 7 and 15), while the indexes of their plots are stored in the sets $\left\{K_\ell^{(i)}\right\}_{\ell=1}^{L}$. The set containing the indexes of the plots available at iteration i is updated in line 11. The while-loop ends when there are no estimated trajectories sharing the same root. Finally, Function Track-Validation (line 17) shrinks all estimated trajectories with less than a specified minimum value, say Q, of plots (trajectories too short may be unreliable) maintaining only the last one and updates their metric accordingly. Once the algorithm is terminated, each plot $\boldsymbol{s}_{k,L}$, along with the associated trajectory (indexed by $\boldsymbol{\tau}_{k,L}$) and test statistic $F_{k,L}$, is sent to the Plot Confirmation stage in Fig. 3.

The complexity of Algorithm 1 is ruled by that of Function 2, whose complexity is linear in the number of integrated scans and quadratic in the average number of alarms in the input data-set [1]. Since the size of the input-data set progressively reduces at each iteration, the computational burden required by Function 2 become negligible after few iterations.

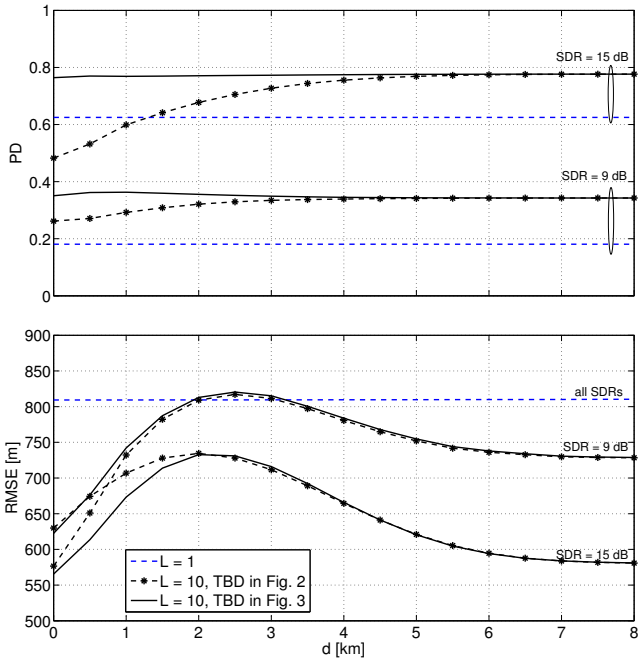

Figure 4. Probability of detection and root mean square error on the estimated target position versus the distance among the closely-spaced targets when FAR $= 1$ per min and $\text{FAR}_{\text{in}} = 10^3$ per min.

III. Numerical results

We discuss a numerical example, where a Swerling I fluctuation model is assumed. Each variable $A_{k,\ell}^2/N_{k,\ell}$ is exponentially distributed, and the constant η is set equal to zero. Range and azimuth measurement errors are Gaussian, with standard deviation 20 m and 0.5°, respectively. The scan period is 1 s, and the search area is $\pm 60^\circ$ and 40 to 140 km. Four, widely-spaced triplets of targets are present, and their positions and number are unknown to the detector. Targets follow a constant velocity model, where the initial position of each triplet is randomly chosen in $50 - 130$ km and $\pm 50^\circ$, and velocity has a random direction and modulus randomly chosen in $0 - 300$ m/s. The mutual distance among targets in the same triplet is constant and equal to d. All targets have the same SDR, which evolves with range according to the radar equation, and the value at scan L is reported in all plots. The Track Formation with STC operates with $L = 10$, $v_{\max} = 300$ m/s, $P = 4$, and $Q = 3$, while the the Track Smoothing stage uses a standard linear regression. For the sake of comparison, the performance of the scheme in Fig. 2 and that of the single-frame detector ($L = 1$) are reported in all plots.

Fig. 4 shows PD and RMSE on the estimated target position versus d,[3] when the SDR is 9 and 15 dB, FAR $= 1$ per min, and $\text{FAR}_{\text{in}} = 10^3$ per min, the latter denoting the false alarm rate in the input plot-list. It is seen that the proposed scheme with STC guarantees a better accuracy in the estimated position than that of the single-frame architecture, performing

[3]Observe that the position measurements of the three targets are different with probability one due to measurements errors, even in the limiting case $d = 0$.

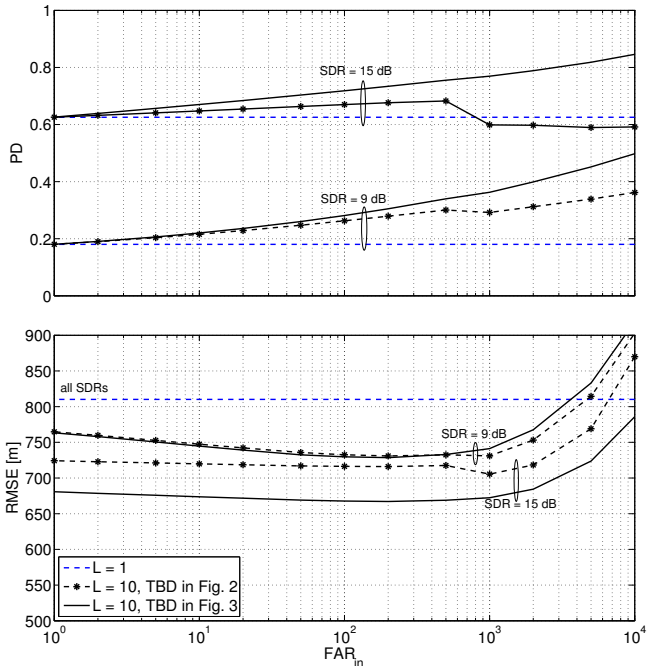

Figure 5. Probability of detection and root mean square error on the estimated target position versus FAR_{in} when $d = 1$ km and $\text{FAR} = 1$ per min.

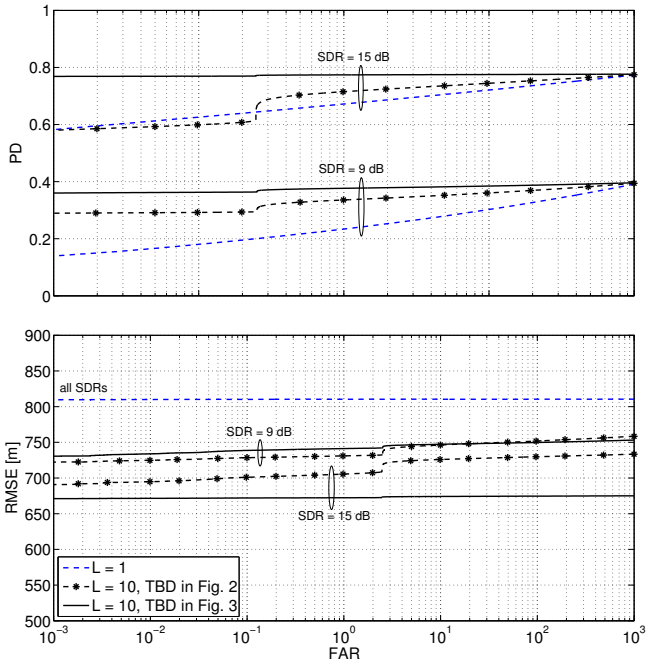

Figure 6. Probability of detection and root mean square error on the estimated target position versus FAR when $d = 1$ km and $\text{FAR}_{\text{in}} = 10^3$ per min.

similarly to the scheme in Fig. 2. As to PD, the detection capability of the scheme in Fig. 2 may become even worse than that of the single-frame detector in the presence of closely-spaced targets. The proposed STC strategy is able to restore the PD gain with respect to the single-frame detector when targets are closely-spaced, making PD almost independent of d. Fig. 5 shows PD and RMSE on the estimated target position versus FAR_{in} when $\text{FAR} = 1$ per min, while Fig. 6 shows the same performance metrics versus FAR when $\text{FAR}_{\text{in}} = 10^3$ per min. In both figures, d is 1 km and SDR is equal to 9 and 15 dB. Again, the proposed scheme exhibits a detection performance superior to that of the scheme in Fig. 2, the latter becoming seriously impaired for large values of SDR and FAR_{in} (see Fig. 5), or large SDR's and low values of FAR (see Fig. 6).

IV. Conclusion

In this work we elaborated on the TBD processor in [1] and proposed a novel track formation procedure with successive track cancellation. The analysis indicates that the proposed solution provides improved detection capabilities in the presence of closely-spaced targets, while giving the same performance as the scheme in [1] when targets are widely-spaced.

References

[1] E. Grossi, M. Lops, and L. Venturino, "A novel dynamic programming algorithm for track-before-detect in radar systems," *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2608–2619, May 2013.

[2] J. Arnold, S. Shaw, and H. Pasternack, "Efficient target tracking using dynamic programming," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 29, no. 1, pp. 44–56, Jan. 1993.

[3] S. W. Shaw and J. F. Arnold, "Design and implementation of a fully automated OTH radar tracking system," in *IEEE Proc. Int. Radar Conf.*, Alexandria, VA, USA, May 1995, pp. 294–298.

[4] S. M. Tonissen and R. J. Evans, "Peformance of dynamic programming techniques for track-before-detect," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 32, no. 4, pp. 1440–1451, Oct. 1996.

[5] S. M. Tonissen and Y. Bar-Shalom, "Maximum likelihood track-before-detect with fluctuating target amplitude," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 34, no. 3, pp. 796–806, Jul. 1998.

[6] H. Im and T. Kim, "Optimization of multiframe target detection schemes," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 35, no. 1, pp. 176–186, Jan. 1999.

[7] L. A. Johnston and V. Krishnamurthy, "Performance analysis of a dynamic programming track before detect algorithm," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 38, no. 1, pp. 228–242, Jan. 2002.

[8] Y. Boers and J. N. Driessen, "Multitarget particle filter track before detect application," vol. 151, no. 6, pp. 351–357, Dec. 2004.

[9] M. G. Rutten, B. Ristic, and N. J. Gordon, "A comparison of particle filters for recursive track-before-detect," in *Int. Conf. Inform. Fusion (FUSION)*, vol. 1, Philadelphia, PA, USA, Jul. 2005, pp. 169–175.

[10] W. R. Blanding, P. K. Willett, Y. Bar-Shalom, and R. S. Lynch, "Multiple target tracking using maximum likelihood probabilistic data association," in *Proc. IEEE Aerosp. Conf.*, Big Sky, MT, USA, Mar. 2007.

[11] S. J. Davey, M. G. Rutten, and B. Cheung, "A comparison of detection performance for several track-before-detect algorithms," in *Int. Conf. Inform. Fusion (FUSION)*, Cologne, Germany, 2008.

[12] G. Pulford and B. La Scala, "Multihypothesis Viterbi data association: algorithm development and assessment," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 46, no. 2, pp. 583–609, Apr. 2010.

[13] E. Grossi, M. Lops, and L. Venturino, "Track-before-detect with censored observations," in *IEEE Int. Conf. Acoustic, Speech, Signal Process. (ICASSP)*, Kyoto, Japan, Mar. 2012, pp. 3941–3944.

[14] E. Grossi, M. Lops, and L. Venturino, "A two-step multi-frame detection procedure for radar systems," in *Proc. Int. Conf. Inform. Fusion (FUSION)*, Singapore, Jul. 2012, pp. 1196–1201.

[15] P. Patel and J. Holtzman, "Analysis of a simple successive interference cancellation scheme in a DS/CDMA system," *IEEE J. Sel. Areas Commun.*, vol. 12, no. 5, pp. 796–807, Jun. 1994.

[16] S. Buzzi, M. Lops, L. Venturino, and M. Ferri, "Track-before-detect procedures in a multi-target environment," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 3, pp. 1135–1150, Jul. 2008.

[17] W. Yi, M. Morelande, L. Kong, and J. Yang, "An efficient multi-frame track-before-detect algorithm for multi-target tracking," vol. 7, no. 3, pp. 421–434, Jun. 2013.