Real-Time Software Architecture for EM-Based Radar Signal Processing and Tracking

1st Alan Nussbaum
Georgia Tech Research Institute
Georgia Institute of Technology
&
2nd Byron Keel
Georgia Tech Research Institute

3rd William Dale Blair
Georgia Tech Research Institute
Georgia Institute of Technology
&
4th Umakishore Ramachandran
Georgia Institute of Technology

2022 IEEE High Performance Extreme Computing Conference (HPEC) | 978-1-6654-9786-2/22/$31.00 ©2022 IEEE | DOI: 10.1109/HPEC55821.2022.9926338

***Abstract*— While a radar tracks the kinematic state (position, velocity, and acceleration) of the target, an optimal signal processing requires knowledge of the target's range rate and radial acceleration that are derived from the tracking function in real time. High precision tracks are achieved through precise range and angle measurements whose precision are determined by the signal-to-noise ratio (SNR) of the received signal. The SNR is maximized by minimizing the matched filter loss due to uncertainties in the radial velocity and acceleration of the target. In this paper, the Expectation-Maximization (EM) algorithm is proposed as an iterative signal processing scheme for maximizing the SNR by executing enhanced range walk compensation (*i.e.,* correction for errors in the radial velocity and acceleration) in the real-time control loop software architecture. Maintaining a stringent timeline and adhering to latency requirements are essential for real-time sensor signal processing. This research aims to examine existing methods and explore new approaches and technologies to mitigate the harmful effects of range walk in tracking radar systems with an EM-Based iterative algorithm and implement the new control loop steering methods in a real-time computing environment.**

***Index Terms*— Signal Processing, Tracking, Expectation-Maximization, Moving Targets, Sensor Processing, Computational and Program Steering, Real-Time Software Architecture**

I. Introduction

Optimal radar signal processing requires knowledge of the target's range, range rate, and radial acceleration that are derived in real time from the tracking function. The tracking function estimates the kinematic state (*i.e.,* position, velocity, and acceleration) of the target through range and angle measurements, whose precision are determined by the signal-to-noise ratio (SNR) of the received signal. Measurement precision is inversely related to SNR in a tracking radar and directly impacts tracking performance. The SNR is maximized by minimizing the matched filter loss due to uncertainties in the radial velocity and acceleration of the target. When a target moves and the estimates of the target kinematic state are imperfect, the target echo will exhibit range walk (RW) over the radar's coherent processing interval (CPI). If the range walk and nonlinear slow-time phase changes are not compensated prior to coherent processing over slow-time, a loss in SNR occurs. Compensating for RW requires accurate estimates of the target's range rate and any radial acceleration that is present. Also, when a target maneuvers by suddenly accelerating or decelerating, the estimates of radial velocity and acceleration will lag the true values and significant losses in SNR are likely to occur. Thus, accurate estimates of the kinematic state critical when abrupt changes in the target's velocity and acceleration occur over short time intervals.

A real-time sensor architecture and robust algorithms are needed to obtaining the knowledge required to accurately estimate the target's radial motion and compensate for RW. At the same time, maintaining a stringent timeline and adhering to latency requirements are essential for real-time sensor signal processing. This research aims to examine existing methods and explore new approaches and technologies to mitigate the harmful effects of RW in tracking radar systems with an Expectation-Maximization (EM) based iterative signal process scheme and implement that scheme with new control loop steering methods in a real-time computing environment.

The research builds upon the existing EM algorithm that provides iterative processing to obtain maximum likelihood (ML) estimates of unknown variables in the presence of nuisance parameter. One of the real-world challenges to the successful implementation of the EM algorithm in real-time is convergence to an accurate ML estimate in a timely manner. This research aims to develop an iterative EM algorithm for RW correction that reliably converges and is implemented in a real-time software architecture. The EM-Based algorithm utilizes the output of the tracking function to a seed the iteration and ensure real-time convergence. Computational steering is required to activate the E-Step and M-Step of the iterative algorithm and program steering to monitor and manage the computer resources [11] to achieve latency and timeline requirements. The following steps list the expectation and maximization steps of EM-Based algorithms sequence developed under this work to compensate for RW. The iteration begins with the maximization step.

- **Maximum likelihood (M-step)** – The maximization step consists of using the newest state estimate to update the radial velocity and acceleration estimates for RW compensation in the ML estimation of the measurements of range, azimuth, and elevation. On the first iteration, the newest state estimate is provided by the prediction of the state estimates provided by the tracking function. On subsequent iterations, the newest state estimate is provided by the Kalman filter or IMM estimator (*i.e.,* E-Step) processing the measurement of the previous iteration

M-Step. After the final iteration, the ML estimates of the measurements are provided to the tracking function.

- **Expectation (E-step)** – The expectation step consists of using the most recent range measurement, $\hat{r}$, from the most recent M-Step to feed the Kalman filter or Interacting Multiple Model (IMM) estimator, which update the estimates of the target's kinematic state to include the unobserved variables, range rate $\hat{\dot{r}}$, and radial acceleration $\hat{\ddot{r}}$. After the final iteration of the M-Step, the resulting ML estimates of the measurements of target range and angles are provided to the tracking function that provides the final E-Step in which the target state estimates are updated with the Kalman filter or IMM estimator. The resulting state estimates are predicted to the next signal processing time and used to seed the first iteration of the M-Step.

The paper proposes augmenting the traditional sensor Signal Processor (SP) and Tracker pipeline architecture to support an iterative EM-bases signal processing scheme to enhance measurement fidelity and tracking performance for moving/-maneuvering targets without impacting the usage of critical radar transmit and receive resources.

Experimental results and analysis demonstrate that the EM-based scheme efficiently compensates for the RW of maneuvering targets showing an improvement in SNR of more than *4 db* in many cases. The SNR improvement increases the quality, reliability, and confidence of the estimates of the state for moving/maneuvering targets. These improvements reduce the track timeline to identify and characterize maneuvers, reducing the need for additional valuable sensor resources. The model prototype of the architecture and design adapts the signal processing and tracking traditional pipeline to a control loop steering execution that implements the iterative EM-Based signal processing scheme.

This paper introduces enhancements to the Signal Processor, Tracker, and Processing Manager software architecture that strengthens the radar system's ability to track maneuvering targets within a shorter timeline and with less demand on valuable radar/sensor resources. The paper is organized as follows. Section II provides the motivation to reduce or eliminate the SNR loss and enhance the ability to track maneuvering targets. Section III describes the implementation of the EM-Based signal processing scheme in terms of the iterative E-step and M-step. Section IV demonstrates the reduction of the SNR loss via RW compensation, improvement in tracking precision, and deduction in the demand for valuable sensor resources. Section V addresses the proposed research to validate the execution of the signal processing and tracking in the real-time control loop (with steering) required to support the iterative EM-based algorithm in a functional real-time radar software architecture.

II. Research Motivation

The target's motion model estimate does not match the object's true trajectory resulting from the loss in SNR. This limits the accuracy of the Signal Processors detection logic, which results in a poor parameter estimate for range, elevation, and azimuth for the moving target. The deficient parameter estimate, resulting from SNR loss, is provided to the Tracker to predict a low confidence track state that will be utilized for future track location beam pointing. Figure 1 is the Signal Processor key algorithms and the traditional pipeline architecture of the Signal Processor, Tracker, and Processing Manager (Radar Resource Manager)[5].

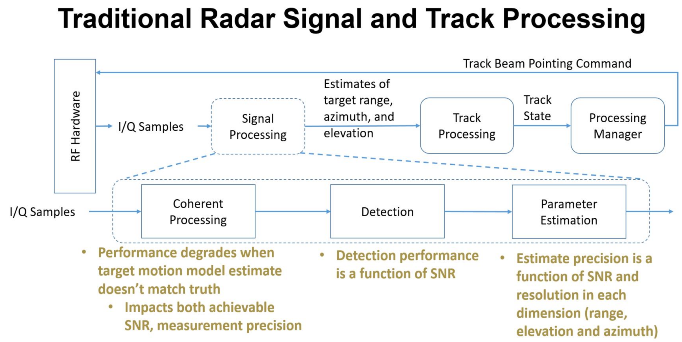

Fig. 1: Signal Processor's estimate precision for a moving target is degraded by loss in SNR. Impacts the Tracker and Processing Manager for track beam pointing.

Figure 2 demonstrate the SNR loss for 1) stationary, 2) moving 300 m/s velocity target, and 3) maneuvering 300 m/s and 9.8 m/s (1g) targets. Figure 2 plots (a), (b), and (c) highlight the RW for the three target configurations. The Figure 2 bottom plots (d), (e), and (f) are the range-Doppler map for each target configuration. Over a single CPI (0.1 seconds), the acceleration only changes the velocity by 0.98 m/sec. You go from 300 m/sec to 300.98 m/sec. As a result, you can't see the slight change in additional range offset; however, there is a quadratic phase term that degrades the Discrete Fourier Transfomr (DFT) and spreads the response across Doppler. The DFT doesn't like quadratic phase terms that change over slow time (pulse index). The DFT expects only a linear phase term (representing the Doppler shift due to a constant velocity target). When the DFT encounters a quadratic phase (changing with pulse index), it spreads the energy across Doppler.

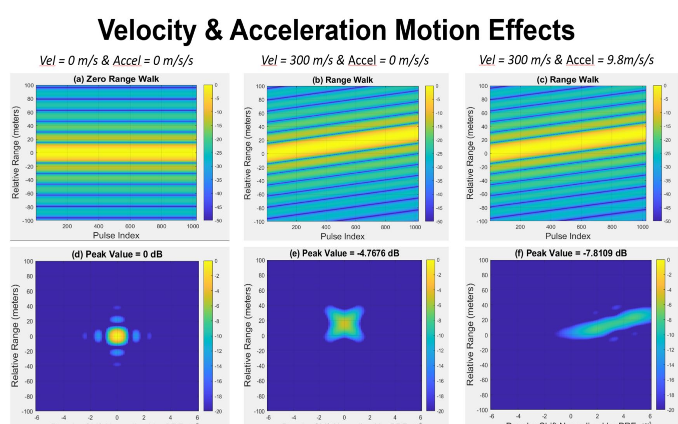

Fig. 2: Plots (a), (b), and (c) x-axis range in relationship to each of the CPI pulses (y-axis). Range-Doppler maps (d), (e), and (f) based on the traditional signal processing fast-time and slow-time processing.

A. Challenges

An expectation-maximization (EM) algorithm is an iterative method to find (local) maximum likelihood (ML) or maximum a posteriori (MAP) estimates of parameters where the algorithm depends on unknown variables. This method for real-time sensor processing has been researched for 50 years without a practical implementation that can support the radar/sensor timelines, and latency requirements [8]. Three challenges are required to be mitigated to implement this paper's EM-Based algorithm.

1) ***Static to Agile Real-Time Approach****:* Eliminate traditional hardcoded and restrictive methods for compensating for a target's range walk. Traditional methods are to have a priori estimates (models) of the target's velocity and acceleration or a bank of filters tuned to the range of possible target velocities and accelerations [3] [9]. These static methods are restrictive and unable to support all possible scenarios. The solution is to calculate the ML for the unknown variables (velocity and acceleration) in real-time. The benefits increase accuracy and higher confidence of the track estimates and predicted state prior to the next scheduled track sensor command [13].

2) ***Rapidly Converge for the Unknown Maximum Likelihood Variables****:* A negative of the EM iterative algorithm's ability to converge to calculate the ML for the unknown variables or achieve convergence in a reasonable latency (initial bias setting)[10] [18]. This paper's research investigation of the EM algorithm for various applications identified developed techniques for seeding the EM algorithm (e.g., K-means clustering, Machine Learning.) The contribution of this research is the solution for the Track Filter initial estimate to provide that seed for the unknown variables. An evaluation of a maneuvering scenario in section IV demonstrates the benefit of calculating reliable ML for the unknown variables in 1 to 2 iterations.

3) ***Control the EM-Base Software Architecture****:* Confidence that software architecture can execute the algorithm's latency while maintaining radar tracking. Department of Energy has been focus on the steering research area for large-scale computational intensive simulations but depend on a human-in-the-loop [4] [6] [16]. Automate steering is required to perform fine-grain decision-making that changes the traditional radar/sensor pipeline architecture to a control loop architecture using computational and program steering. Computational steering is the ability to manually intervene with an otherwise autonomous computational process. Program steering includes monitoring and data interpretation practices to control an application's execution state, architecture configuration, and the utilization of resources. The solution is to automate computational and program steering decisions making, fine-gain control, of the radar/sensor processing with vectors and actuators [12]. The benefit is that the automated steering performs fine-grain real-time decisions that change the traditional pipeline architecture to a control loop architecture.

- **Computational Steering** = Ability to manually intervene with an otherwise autonomous computational process.
- **Program Steering** = Includes the practices of monitoring and data interpretation to control an application's execution state, architecture configuration, and the utilization of resources.

III. Processing Flow for EM-Based Algorithm

A moving target may exhibit an RW through the range and Doppler cells over a coherent processing interval (CPI). The impact is a reduced SNR due to broadening the point target response in the range-Doppler map. The reduced SNR and broadened point target response degrade the precision and accuracy of the range ($\widehat{R}$) and range rate ($\widehat{v}$) estimates. The degraded range and range rate estimates impact the ability of the Tracker to estimate the current state accurately and to respond to an accelerating target. Figure 2(a) is a stationary target, and there are no RW effects, and (b) and (c) demonstrate the effect of RW for a moving target over the entire coherent processing interval. This research utilizes the EM iterative algorithm to compensate for the target's motion with the unknown variables, calculated by the track filter, Kalman, or IMM, interfacing with the Signal Processor.

Figure 3 is the research modification to the traditional radar pipeline algorithms execution to utilize the EM-Based algorithms taking advantage of the EM iterative process to calculate ML for the unknown variables by reprocessing the I/Q current sensor digital data. This section will describe the EM-Based algorithm utilizing proven signal processing, tracking, and cluster algorithms to enable the software architecture to enable a steering control loop execution.

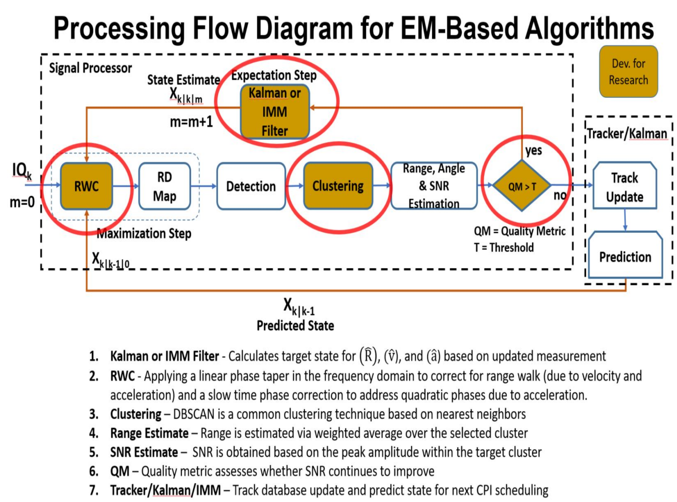

Fig. 3: Taking proven algorithms from domain experts to support iterative EM-Based algorithm processing flow.

A. Tracker's Filter(s)

For a given linear stochastic state and measurement model with white Gaussian errors, the Kalman filter provides the state's conditional mean estimate or expected value estimate [1]. Multiple kinematic motion models are included in the stochastic state model for maneuvering targets (*i.e.,* targets that change modes through a sudden change in acceleration), and the switching between those kinematic models is treated as a finite-state Markov chain. The IMM estimator [1] is well accepted as the best approach to the state estimation of a Markovian switching system when the computational costs are considered.

For signal processing, the estimates of the range, range rate, and range acceleration are the key to reducing SNR loss. Since this paper focuses on reducing signal processing loss, this section only addresses the estimation of the range, range rate, and range acceleration. After the initial signal processing of the I/Q data components, the IMM E-step, Figure 3, is the key to reducing SNR loss for calculating the unknown variables for velocity and acceleration to seed the signal processor to calculate ML. The IMM estimator uses the newest measurement, and the output of the estimator is treated as a temporary state estimate for refining the signal processing to increase the SNR. Once the SNR has been maximized via this iteration of state estimation and signal processing, the final measurement is sent to the tracking function for state estimation, and track management [3] [2].

When the computational costs are considered, the IMM estimator [1] is well-accepted as the best approach to the state estimation of a Markovian switching system. The IMM estimator consists of a Kalman filter for each model, a probability evaluator, an estimate mixer at the input of the filters, and an estimate combiner at the output of the filters. Figure 4 illustrates the functional decomposition of the IMM estimator with two models, Nearly Constant Velocity (NCV) and Nearly Constant Acceleration (NCA), and the data flow. The $X^i_{k|k}$ and $P^i_{k|k}$ denote the state estimate and covariance given that Model i is in effect from time t_{k-1} to time t_k and measurement Z_k. The λ^i_k denotes the likelihood for model i at time t_k, and the μ_k denotes the vector of model probabilities. The output of the IMM Estimator is $X_{k|k}$ and $P_{k|k}$. The next filtering cycle starts with the interacting or mixing step that prepares the inputs for the Kalman filters for possible mode switch. The $X^{0i}_{k-1|k-1}$ and $P^{0i}_{k-1|k-1}$ denote the mixed prior estimates for the Kalman filter based on Model i. This mixing step differentiates the IMM estimator from parallel Kalman filters, for which one of the filters is picked as the best. Parallel Kalman filters and picking the best has no concept of mode switching that the IMM estimator addresses. In the IMM estimator, the multiple models, NCV and NCA, interact through the mixing to track a target maneuvering through an arbitrary trajectory. A derivation and detailed explanation of the IMM algorithm are given in [1].

B. Signal Processor's Range Walk Correction

Over a coherent processing interval (CPI), a target's radial components of motion, which include both velocity and acceleration, present as range walk (RW) in fast-time and a linear and quadratic phase response over slow time . The RW and slow-time quadratic phase (STQP) reduce the resultant signal-to-noise ratio (SNR) when employing traditional discrete Fourier transform (DFT) based Doppler processing. The Cramer-Rao lower bound (CRLB) [14] on measurement precision denotes an inverse relationship between precision and SNR. In a traditional radar pipeline architecture, the

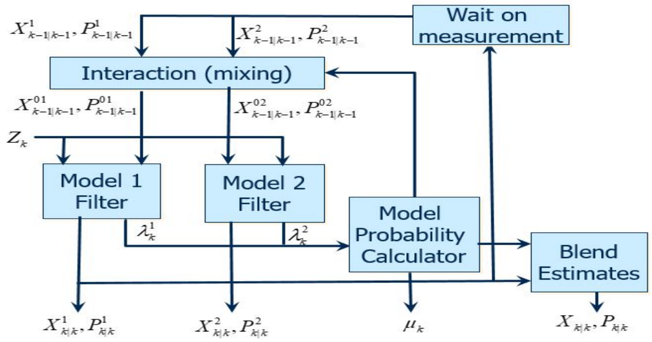

Fig. 4: IMM Estimator with Two Models.

Tracker's performance is directly affected by measurement precision after the Signal Processor; thus, there is a strong desire to maximize SNR RW and quadratic phase variation over a CPI results in a smearing of the target's point spread function in both range and Doppler resulting in a loss in SNR. If the target's radial components of velocity and acceleration were known a priori, motion compensation is achievable via deterministic relationships. However, the target's radial components are not known a priori and must be estimated by external means such as track filtering. In Figure 3 range walk correction (RWC) EM steps, we seek to employ an iterative process known as expectation-maximization, implementing the EM-Based algorithm developed for this research, to estimate the components of radial motion and use these estimates to correct for target motion resulting in a better range and angle estimates with improved SNR. Given a target with initial at range R_0, initial radial velocity, v_0, and radial acceleration a, the received signal's spectrum may be written as Equation 1.

$$X_{rcv}(\Omega, n) = X_{tx}(\Omega) \exp\left(-j\Omega\frac{2}{c}R_0\right) \exp\left(-j\Omega\frac{2}{c}\left(v_o nT + 0.5an^2T^2\right)\right) \tag{1}$$

Where X_{tx}(omega)is the spectrum of the transmitted pulse at baseband, omega is the frequency variable in radians/sec, c is the speed of light, n is the pulse index (n = 0,...,N-1), and T is the pulse repetition interval.The RW over fast-time is governed by the radial components of velocity and acceleration. Given estimates of the target's radial components of initial velocity and acceleration, $\hat{v}_0$ and $\hat{a}$, obtained from the tracker, one may us these components to compensation for range walk. For the nth pulse, RW may be compensated for by applying a linear phase correction term in the frequency domain using Equation 2,

$$Y_{RWC}(\Omega, n) = X_{rcv}(\Omega, n) \exp\left(j\Omega\frac{2}{c}\left(\hat{v}_o(nT - t_c) + 0.5\hat{a}(nT - t_c)^2\right)\right) \tag{2}$$

where the subscript *RWC* denotes that range walk compensation has been applied. The phase compensation is chosen to position the target response in fast-time at a range commensurate with the range of the target within the receive window at a time t_c corresponding to the middle of the CPI. The quantities Δv and Δa are calculated and represent the error in the velocity and acceleration estimates of the radial components.

978-1-6654-9786-2/22/$31.00 ©2022 IEEE

If both quantities are zero, then RW is fully compensated for and the target response is positioned in fast-time at a range corresponding to the middle of the CPI. Suppose one of both quantities isn't zero. In that case, some residual RW is present in the response over slow-time, and based on the quality metric check, another EM-Based iteration can be scheduled per Subsection III-D. In slow-time, the receive phase exhibits both a linear term associated with the Doppler shift and a quadratic term defined by the acceleration denoted in Equation 3 where f_c is the transmit frequency.

$$z(t,n) = y_{RWC}(t,n) \exp\left(j2\pi f_c \frac{2}{c}\left(v_0 nT + 0.5a(nT)^2\right)\right) \tag{3}$$

The DFT serves as a matched filter to the linear Doppler shift, but smears the quadratic term across Doppler resulting in reduce SNR. A slow time quadratic phase correction, in Equation 4, is applied across slow-time to the remove quadratic component.

$$y_{RWC,QPC}(t,n) = z(t,n) \exp\left(-j2\pi f_c \frac{2}{c} 0.5\hat{a}(nT - t_c)^2\right) \tag{4}$$

The following are the six computational processing steps that the signal processor is required to reduce the SNR loss by correcting for RW:

1) For pulse n, apply DFT in fast-time to convert from time to frequency domain.
2) Apply linear phased ramp to in frequency domain to correct for range-walk
3) Apply inverse DFT to convert back to time domain
4) Apply steps 1-3 to all pulses comprising a CPI
5) Having applied RWC to all of the pulses, apply a slow-time quadratic phase correction across the N pulses comprising the CPI.
6) Finally, perform Doppler processing across the slow time via a DFT

The six processing steps, involving multiple iterations, require a significant number of computations that will lengthen the overall Signal Processor's latency and affect the radar's required scheduling and tracking timeline. Traditional radar architectures will not support the iterative EM-Based algorithm. To benefit from the EM-Based algorithm developed for this research, a control loop real-time steering architecture is required and discussed in Section V.

C. Detection Clustering

For detection processing in Figure 3 the paper is taking advantage of Density-Based Spatial Clustering of Applications with Noise (DBSCAN), an algorithm for density-based clustering. The measurements from the coherent processing step in the signal processor, a range-Doppler matrix based on the sum channel, are considered the data points inputs to DBSCAN, and their data points are assigned to clusters. The cluster with the maximum number of measurements is assigned to the core cluster and is the input to the Signal Processor's parameter estimation function.

D. Quality Metric

The quality metric threshold step in Figure 3 will identify when the EM-Based algorithm converges. This threshold is set to *.1 db* but in the future research, it will be controlled and real-time calculated by the processing manager in the steering architecture that will be described in section 5. If the SNR change is greater than *.1 db* from the previous iteration. Then the EM-Based algorithm executes another iteration to improve the ML resulting from reprocessing the I/Q data to minimize the RW effect.

IV. Evaluation

Previous EM's research approaches for the unobserved variables may not consistently support the sensor's near-real-time timelines [7] and its decision-making ability to attain convergence [17][15]. However, the evaluation will demonstrate that the proposed EM-Based algorithm supporting RW correction meets our objectives to enhance measurement precision, convergence timeliness, and reduce valuable radar resources. In the scenario selected to demonstrate the value of the EM-Base algorithm, the target for measurements 1 to 3, the object is at a constant velocity of *300 m/s* and then performs a *3g* maneuver between measurements 3 and 4. Between measurements 6 to 7, the object performs another *1g* maneuver. Finally, for measurement 9, the target ends its maneuvering, and its trajectory is at the *300 m/s* constant velocity. The ground base radar configuration uses a center frequency of 10 GHz, and a pulse width of 10 us. Each coherent processing interval contains 1024 pulses. The rate of tracking for the target is 1 Hz. Figure 5 demonstrates the EM-Based algorithm's ability to compensate for RW is performed promptly and rapidly adapts to the object trajectory changes. Each measurement in Figure 5 contains the reduction in SNR, the number of detections in the core cluster, calculated range, velocity, and acceleration, and the NCV and NCA IMM filter probability. Highlighted in the red boxes, there are four unique events and the associated measurements:

1) Measurement 3, the EM-Based algorithm reduces the SNR loss by *4.8 dB* for the *300 m/s* velocity target in 2 iterations.
2) Measurement 4, the algorithm detects that a *3g* maneuver is being performed and a *10.7 dB* SNR reduction. Immediately, the first and/or second EM-Based iteration reduces the SNR loss by *9.3 dB*, detects the probability of the maneuver, and at measurement 5 calculates a nearly accurate acceleration.
3) Measurement 7, the object performs the second maneuver of *1g* and the SNR loss is *7.9 dB*, and the EM-Based algorithm eliminates the SNR loss in 1 iteration.
4) Measurement 9, the object trajectory has no acceleration (maneuver is completed), and the algorithm is immediately impacted with an SNR loss *6.3 dB* because it uses the unknown variables velocity ($\widehat{v}$) and acceleration ($\widehat{a}$) previous calculated in measurement 8. The first iteration of the research algorithm reduces the SNR loss by *5.7 dB* and calculates a realistic acceleration ($\widehat{a}$) estimate.

V. Agile Concurrent Real-Time SW Architecture for EM-Based Algorithm

Traditional pipeline architecture reference in section II needs to transform into an implementation that needs a robust C++ Linux software architecture that exhibits control loop features that provide real-time steering sensor functions and algorithms adhering to the EM-Based sequences of events.

978-1-6654-9786-2/22/$31.00 ©2022 IEEE

Target Preforms 2 Maneuvers

Signal Processor Results			Tracker and IMM			
Meas	SNR Reduction dB	Cluster NumDet	Range m	Vel m/s	Accel m/s/s	PROB NCV/NCA Modes
1	4.8	14440	4517.1	0.0	0.0	.8/.2
2	4.1	1848	4817.9	300.6	0.0	.8/.2
3	4.9	718	5117.5	299.9	-0.2	.8/.2
3	0.2	307	5117.2	299.7	-0.1	.8/.2
3	0.1	326	5116.9	299.7	-0.1	.8/.2
4	10.7	581	5427.9	320.5	19.3	.8/.2
4	1.5	321	5430.8	321.0	14.3	.8/.2
4	1.4	317	5416.9	321.0	14.3	.56/.43
5	1.0	302	5780	363.1	32.2	.56/.43
5	0.0	267	5780	363.1	32.2	.05/.95
6	0.2	257	6156.7	390.9	28.2	.05/.95
6	0.0	198	6156.7	390.9	28.2	.03/.97
7	7.9	263	6553.1	402.0	11.6	.03/.97
7	0.0	203	6551.6	399.2	9.0	.26/.74
8	0.0	148	6957.2	410.4	9.6	.26/.74
8	0.0	148	6957.2	410.4	10.0	.05/.95
9	6.3	99	7364.7	406.6	-1.7	.05/.95
9	.6	120	7365.9	408.3	-0.6	.37/.63
10	0	128	7775.3	409.5	.5	.37/.63
10	0	131	7775.3	409.6	.5	.7/.3

Fig. 5: EM-Based algorithm mitigates for entire scenario the SNR loss for 1) 300 m/s constant velocity, 2) 3g maneuver, 3) 1g maneuver, and 4) 300 m/s constant velocity

A. Software Architecture for EM-Based Algorithm

Previous steering research, mainly for the Department of Energy (DOE), has identified that vectors and actuators are vital functions controlled by an analyst-in-the-loop for adapting the high-performance scientific application. The next phase of this research is to automate the application steering Figure 6 by building upon the previous research concepts for steering applications in section II-A3.

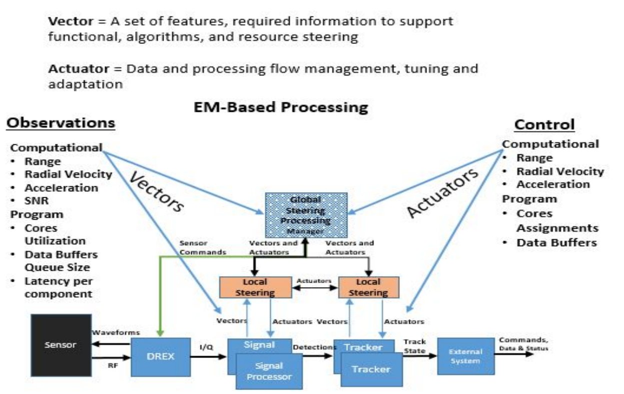

Fig. 6: Closed-Loop Computational and Program Steering

B. Performance Requirements

For the ground-based radar utilized for the evaluation in section IV the performance required for this sensor configuration is the following budgets for each EM-Based algorithm iterative sequence:

1) M-Step: SP create initial measurement (range, az, el) - 55 ms
2) E-Step: Filter creates track state estimate - 1 ms
3) M-Step: SP reprocess data increase SNR - 35 ms
4) E-Step: Filter creates accurate track state - 1 ms
5) M-Step: SP reprocess data increase SNR - 35 ms
6) EM algorithm converges
7) SP calls the Tracker update object state - 2 ms
8) Processing manager schedule next CPI - 1 ms

C. Need for Agile Concurrent SW Architecture

Processing Manager based on rules and situational awareness [12] one second frames subdivided into ten 100 ms CPI. Each CPI has 1024 pulses, and Pulse Repetition Interval (PRI) is 100 us. For 1 object, EM-Based algorithm has no problem completing its E-steps and M-steps in 127 ms, adhering to the 1 Hz track rate. If attempting to track 10 targets, one per CPI, only 7 out of 10 can be processed in the 1 Hz track rate and scheduling frame. Figure 7 assign multiple Signal Processors and Trackers using Program Steering of computing resources and Computational Steering of E-Step and M-Step processing. Steering of the application is required to take advantage of the EM-Based implementation and object tracking benefits.

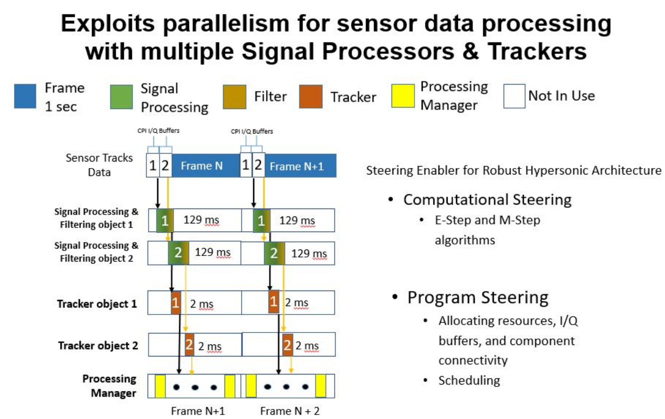

Fig. 7: Benefit the EM-Based Algorithm needs an agile concurrent software architecture

D. Concurrent Computational Processing Framework

EM-Base research-generated algorithm is currently being prototyped to an architecture documented in this section that must adhere to define latency and timeline performance requirements and transition from a pipeline to a real-time steered control loop architecture. Steering architecture design needs an event-driven approach. The Actor Model provides a high-level abstraction for concurrent applications to support parallel processing for the Worker thread (Parent) and Sub-thread (child) levels.

Accelerators (GPUs) will be investigated for the signal processing of the sensor data for coherent and detection processing. This can significantly benefit the Signal Processor latency and the reprocessing of the I/Q data required for the EM-Based algorithm.

VI. Conclusion

Introduced an EM-Based approach for addressing RW over a CPI to improve the measurement quality of a sensor used to track moving objects. Motion during a processing interval negatively impacts track performance for many applications, including air defense, autonomous vehicles, cell phones, image processing, sonar, etc. This work will compensate for a target's motion resulting in higher SNR and improved radar measurement precision in range and angle. The current phase of this research is the implementation of a control loop steering architecture that adapts the processing management, signal processing, and tracking functions with the incorporation of the iterative EM-Based algorithm research.

978-1-6654-9786-2/22/$31.00 ©2022 IEEE

References

[1] Y Bar-Shalom. “X. r. Li, and t. Kirubarajan”. In: *Estimation with Applications to Tracking and Navigation* (2001).

[2] Yaakov Bar-Shalom, X Rong Li, and Thiagalingam Kirubarajan. *Estimation with applications to tracking and navigation: theory algorithms and software*. John Wiley & Sons, 2004.

[3] W Dale Blair and T Bar-Shalom. “Tracking maneuvering targets with multiple sensors: Does more data always mean better estimates?” In: *IEEE Transactions on Aerospace and Electronic Systems* 32.1 (1996), pp. 450–456.

[4] Greg Eisenhauer, Karsten Schwan, and Fabián E Bustamante. “Publish-subscribe for high-performance computing”. In: *Internet Computing, IEEE* 10.1 (2006), pp. 40–47.

[5] Liron Frenkel and Meir Feder. “Recursive expectation-maximization (EM) algorithms for time-varying parameters with applications to multiple target tracking”. In: *IEEE Transactions on Signal Processing* 47.2 (1999), pp. 306–320.

[6] Weiming Gu et al. “Falcon: On-line monitoring and steering of large-scale parallel programs”. In: *Frontiers of Massively Parallel Computation, 1995. Proceedings. Frontiers’ 95., Fifth Symposium on the*. IEEE. 1995, pp. 422–429.

[7] Pakorn KaewTraKulPong and Richard Bowden. “An improved adaptive background mixture model for real-time tracking with shadow detection”. In: *Video-based surveillance systems*. Springer, 2002, pp. 135–144.

[8] Todd K Moon. “The expectation-maximization algorithm”. In: *IEEE Signal processing magazine* 13.6 (1996), pp. 47–60.

[9] Richard L Moose, Hugh F Vanlandingham, and DH McCabe. “Modeling and estimation for tracking maneuvering targets”. In: *IEEE Transactions on Aerospace and Electronic Systems* 3 (1979), pp. 448–456.

[10] Robert D Nowak. “Distributed EM algorithms for density estimation and clustering in sensor networks”. In: *IEEE transactions on signal processing* 51.8 (2003), pp. 2245–2253.

[11] A. Nussbaum, S.M.C. Choodamani, and K. Schwan. “ObsCon: Integrated Monitoring and Control for Parallel, Real-Time Applications”. en. In: *Cluster Computing (CLUSTER), 2015 IEEE International Conference on*. IEEE, 2015.

[12] Alan Nussbaum et al. “Application of Computional and Program Steering of Radar Resource Management”. In: *Military Sensing Symposia Tri-Service Radar* (2018).

[13] Alan Nussbaum et al. “EM-based Radar Signal Processing and Tracking”. In: *2021 IEEE Radar Conference (RadarConf21)*. IEEE. 2021, pp. 1–6.

[14] Shimon Peleg and Boaz Porat. “The Cramer-Rao lower bound for signals with constant amplitude and polynomial phase”. In: *IEEE Transactions on Signal Processing* 39.3 (1991), pp. 749–752.

[15] Silvana Silva Pereira, Roberto Lòpez-Valcarce, and Alba Pagès-Zamora. “A diffusion-based EM algorithm for distributed estimation in unreliable sensor networks”. In: *IEEE Signal Processing Letters* 20.6 (2013), pp. 595–598.

[16] John A Stankovic et al. “Feedback control scheduling in distributed real-time systems”. In: *Real-Time Systems Symposium, 2001.(RTSS 2001). Proceedings. 22nd IEEE*. IEEE. 2001, pp. 59–70.

[17] CF Jeff Wu. “On the convergence properties of the EM algorithm”. In: *The Annals of statistics* (1983), pp. 95–103.

[18] Yan Yang and Jinwen Ma. “A single loop EM algorithm for the mixture of experts architecture”. In: *International Symposium on Neural Networks*. Springer. 2009, pp. 959–968.

978-1-6654-9786-2/22/$31.00 ©2022 IEEE