Please cite this paper as: J. M. Aristoff, D. J. C. Beach, P. A. Ferris, J. T. Horwood, A. D. Mont, N. Singh, and A. B. Poore, "Multiple Frame Assignment Space Tracker (MFAST): results on UCT processing," in Proceedings of the 2015 AIAA/AAS Astrodynamics Specialist Conference, Vail, CO, August 2015 (Paper AAS-15-675)

Multiple Frame Assignment Space Tracker (MFAST): Results on UCT Processing

Jeffrey M. Aristoff[*], David J. C. Beach[†], P. Alex Ferris[‡], Joshua T. Horwood[§], Alex D. Mont[‡], Navraj Singh[‡], and Aubrey B. Poore[¶]

Numerica Corporation, Fort Collins, CO, 80528, USA

Numerica's Multiple Frame Assignment Space Tracker (MFAST) is a multi-sensor multi-regime space object tracking system that is presently undergoing transition to an operational environment to support improved uncorrelated track (UCT) processing. This paper communicates recent results from MFAST that were obtained by processing real-world historical radar and optical data from the Space Surveillance Network (SSN) in a "UCT processing mode." The results demonstrate that MFAST generally achieves a correctness ratio of 93% or higher, with no cross-tags, and is able to process the data on a consumer-grade laptop computer in real-time or faster.

I. Introduction

On a typical day, the Joint Space Operations Center (JSpOC) is inundated with thousands of sensor reports which do not correlate with existing objects in the space catalog. Such reports (e.g., sensor observations, sensor tracks) are called *uncorrelated tracks (UCTs).* At present, UCT processing is manpower-intensive and existing tools are only able to resolve a small percentage of these UCTs. New sensors, such as the Space Fence, that are soon expected to come online will be able to observe objects much smaller than what is currently possible, thereby greatly increasing the UCT population. An additional challenge involves the persistent surveillance of hard-to-acquire dim and eccentric space (HADES) objects and high area-to-mass ratio (HAMR) objects in deep-space. Such objects are notoriously difficult to track due to their faintness and complex motion. Consequently, many of them get lost and are declared UCTs once they are eventually reacquired. Thus, by embracing new and innovative tools to maintain custody of these HADES and HAMR objects and more efficiently processing UCTs, the U.S. will be more informed on the rapidly evolving space environment and hence will have an increased ability to maintain superiority in space. Therefore, advanced space situational awareness (SSA) will need to leverage data from all sensors, especially in the deep-space geosynchronous Earth orbit (GEO) regime, to support operations in providing improved UCT processing, better prediction capabilities especially for HAMR and HADES objects, and ultimately a more comprehensive catalog containing smaller, dimmer, deep-space, and hard-to-acquire objects.

The GEO Odyssey program is a partnership between the U.S. Air Force Research Lab (AFRL) and other industry, government, and academia team members, to rapidly improve UCT processing. The primary objective of GEO Odyssey is to transition four UCT processing tools to operations, namely:

1. Constrained Admissible Region Multiple Hypothesis Filter (CAR-MHF) of the AFRL Space Vehicles Directorate,
2. Search and Determine Integrated Environment (SADIE) of the AFRL Directed Energy Directorate,

[*]Program Manager, Numerica Corporation, 5042 Technology Parkway, Suite 100, Fort Collins, CO, 80528
[†]Software Engineer, Numerica Corporation
[‡]Research Scientist, Numerica Corporation
[§]Senior Research Scientist, Numerica Corporation
[¶]Chief Scientific Officer, Numerica Corporation

3. Legacy UCT processing tool of Aerospace Corporation,

4. Multiple Frame Assignment Space Tracker (MFAST) of Numerica Corporation.

The transition of more than one UCT processing tool is indicative of the technical challenges associated with UCT resolution and mitigates programmatic risk as rarely does one tool provide a "one size fits all" solution. Nonetheless, it is important to assess the strengths and weaknesses of a given UCT processor, preferably using real-world data. Recent publications by the CAR-MHF team[1] and SADIE team,[2] indicate promising performance when said algorithms were used for UCT resolution. This paper is dedicated to providing a similar analysis for MFAST.

Numerica's MFAST provides a real-time multi-sensor multiple hypothesis tracking (MHT) solution for SSA supporting UCT resolution, breakup and closely-spaced object processing, and catalog maintenance. It tracks objects in all regimes of space (e.g., GEO, HEO, MEO, LEO) using radar, ground-based optical, and space-based optical sensors. MFAST uses Numerica's special optimization-based formulation of the data association problem, namely the multiple frame assignment (MFA) formulation,[3] that is well-suited for large-scale tracking problems. MFAST includes customized algorithms for non-linear filtering, orbit determination, orbit and uncertainty propagation (including Numerica's implicit Runge-Kutta orbital and uncertainty propagator[4,5]), and advanced physics-based complexity reduction (or gating) techniques that are used to control runtime without sacrificing accuracy. Such methods along with the portability of MFAST allow it to be run on most platform and hardware setups, including laptops. For additional details regarding the design and implementation of MFAST, as well as results on simulated data, see Section II, as well as the papers of Aristoff *et al.*[6] and Singh *et al.*[7]

The purpose of this paper is to demonstrate the performance of MFAST in UCT processing scenarios on data representative of what is being processed under the GEO Odyssey program. Because results from the real UCT data cannot be shown due to its sensitivity, results are instead shown from real-world historical radar and optical data from the Space Surveillance Network (SSN), collected during 2004. To mimic UCT processing, this SSN data is treated as a mock UCT problem, meaning that the true data associations (i.e., satellite number tags) are stripped from the data so that all observations can be treated as UCTs, and a new catalog of candidate orbits is built from scratch. The main performance metric used to validate the performance of MFAST is the *correctness ratio*, defined in the companion paper,[8] that accounts for both incorrectly associated observations (cross-tags) and missing observations. Other secondary metrics that are reported include runtime and number of candidate orbits produced. Scenarios are varied according to the total number of objects. At a high level, in nominal radar and optical UCT processing scenarios, MFAST achieves a correctness ratio of 93% or higher, with no cross-tags, and is able to process the data on a consumer-grade laptop computer in real-time or faster. We remark that the performance of MFAST has also been externally validated through the JSpOC Mission System (JMS) NumVal test cases for UCT and breakup processing. Results can be requested from the JMS Program Office.

The layout of this paper is as follows. Section II provides an overview of multi-target tracking, describes how MFAST fits into the universe of tracking systems, and discusses MFAST's differentiators, architecture, and implementation details. Section III describes the 2004 SSN data and some of the key metrics used to assess the performance of MFAST. Section IV communicates the results from MFAST that were obtained from processing this data and discusses the challenges and issues that were encountered in processing this data and how they were resolved. Section V provides concluding remarks.

II. Overview of MFAST

A multi-target tracking (MTT) system is an essential requirement for surveillance systems that employ one or more sensors along with a computer system to estimate the states of objects in the environment. In SSA, a well-designed MTT system can serve the needs of catalog maintenance, UCT resolution, and cataloguing of new breakup objects. The two fundamental problems of tracking are data association (correlation) and estimation (fusion). The objective of data association is to take as input a stream of reports (e.g., sensor observations) and partition them into reports that emanate from common objects and false alarms. The estimation problem is to combine a sequence of reports emanating from a common object to improve the state or understanding of the object. To solve the data association problem, one must first solve the estimation problem, but to solve the estimation problem, one must know which reports emanate from which object. Thus, tracking is a *joint data association and estimation problem.* When the data associations are unambiguous, such as in the case of widely-spaced objects, the association problem is simple. However, for closely-space objects, the association problem is considerably more difficult.

In the overview of the tracking problem and the ensuing discussion of MFAST, the following fundamental definitions are required.

- **Report:** Any data type that is passed to the tracking system as input. Some examples are listed below.
  - **Sensor track:** A short sequence of time-ordered sensor observations all posited to have emanated from the same object. **Radar tracks** are sensor tracks whose observations are produced from radar sensors. **Optical tracks** are sensor tracks whose observations are produced from optical sensors.
  - **Six-dimensional (6D) tracklet:** A representation of a radar track consisting of a single 6D vector in position-velocity or orbital element space with a corresponding 6×6 covariance matrix characterizing the errors in the state vector. The track state and covariance are typically estimated by solving a batch least squares initial orbit determination problem.
  - **Four-dimensional (4D) tracklet:** A representation of an optical track consisting of a single 4D vector of two angles and two angle rates with a corresponding 4×4 covariance matrix characterizing the errors in the vector. The angles are usually right ascension and declination or azimuth and elevation.
  - **Singleton observation:** A single sensor observation such as a three-dimensional (3D) radar observation (in range, azimuth, and elevation) or a two-dimensional (2D) line-of-sight observation produced from an optical sensor.
  - **Candidate orbit:** A sequence of time-ordered sensor tracks or tracklets all posited to have emanated from the same object. Candidate orbits can be represented by a single state vector and covariance (and can include other meta data), such as the Vector Covariance Message (VCM) format.
- **Frame:** A set of reports generated over a short time interval. In air tracking, an example is all data collected by a radar sensor over a single sweep. In space surveillance, an example is all data collected by a sensor over a single dwell. Internally, MFAST attempts to construct *proper frames* in which an object can have at most one report associated to it (i.e., an object cannot be seen more than once in a proper frame).
- **Hypothesis:** A complete accounting of all reports such that each report is associated to at most one object or is unassigned. This is the definition of "hypothesis" in the context of multiple hypothesis tracking (MHT).

In many online estimation and data fusion problems, such as those implicit in the space surveillance tracking environment, one typically processes the reports sequentially (i.e., through filtering) rather than through a batch process. The data association problem is solved in an analogous framework. In most practical applications one cannot obtain a real-time solution of the data association problem by processing all N frames of data simultaneously. Instead, it is solved sequentially over a *sliding window* of $M < N$ frames. Firm association decisions are held at the back of the sliding window. *Single-frame* methods with $M = 2$ process one additional frame of data (in conjunction with the firm tracks at the back of the window). Examples of single-frame methods include nearest neighbor and global nearest neighbor based on a two-dimensional assignment solver. Such methods are forced to make firm association decisions. Once made, such decisions are *irrevocable*. For many widely-spaced objects and a clear background, single-frame methods are usually appropriate. *Multiple-frame* methods, such as MFAST, use a sliding window with $M > 2$ and, as such, can hold difficult or ambiguous decisions *in abeyance* until more information is available and can change past decisions to improve those at the current time. In dense tracking environments with closely-spaced objects such as breakups in LEO or satellite clusters in GEO, the performance improvements of multiple-frame methods over single-frame methods are substantial.[7]

Multiple hypothesis tracking (MHT) approaches, including MFAST, are the most common multiple-frame methods. They divide into *hypothesis-oriented* and *track-oriented*. The hypothesis-oriented approaches enumerate all of the hypotheses; they tend to work well in cases where one can initiate a full state from a

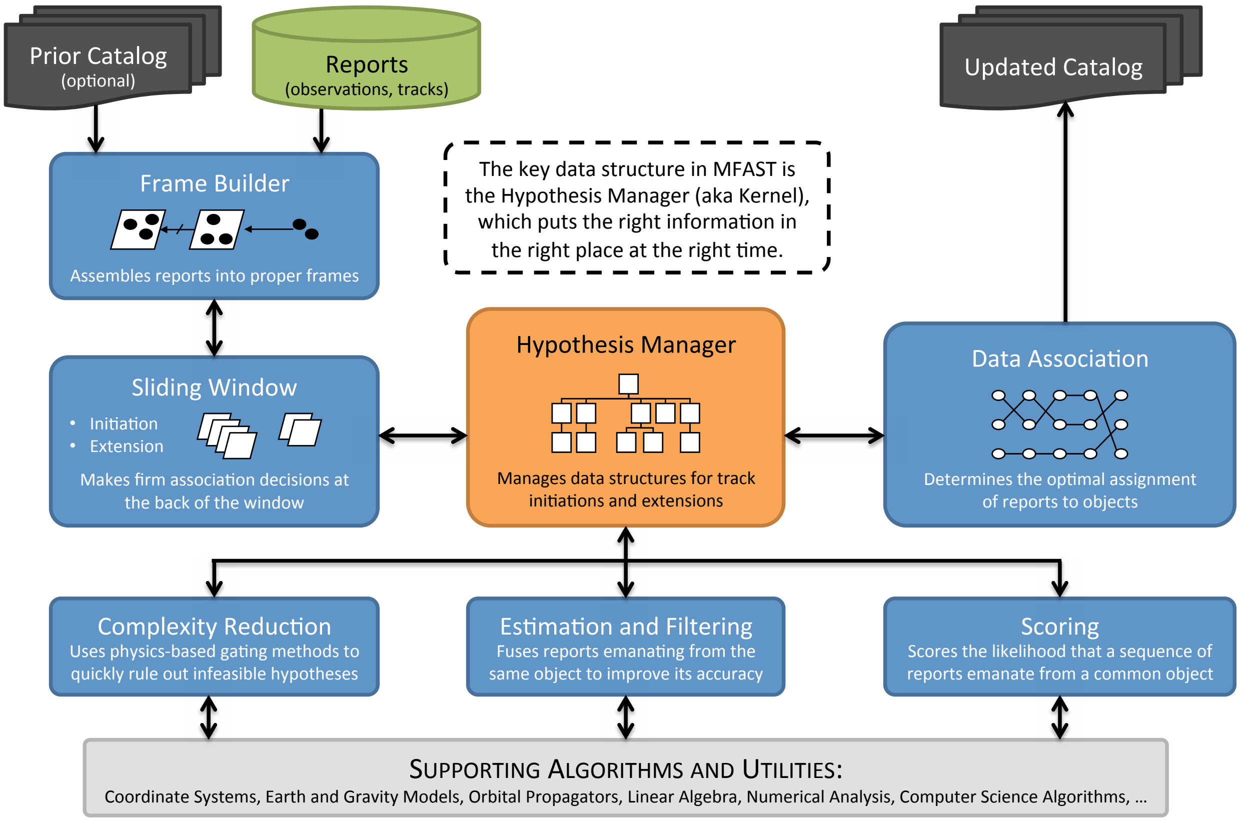

Figure 1. Architecture of the Numerica Multiple Frame Assignment Space Tracker (MFAST) system

single observation. Track-oriented MHTs are preferred in MHTs wherein one cannot initiate a full state from a single observation, such as angle-only observations generated from optical sensors. *Numerica's MFAST is a track-oriented optimization-based MHT that uses a special assignment formulation of the data association problem.* Additional details on this assignment problem are provided in Poore.[3]

Figure 1 shows a general multi-object tracking architecture for SSA. Most other tracking systems and "MHTs" have these same components. The input is a sequence of reports commonly in the form of sensor tracks or tracklets. Optionally, one can pass a prior catalog of candidate orbits. Internally, the tracking system solves the data association problem, which determines what reports emanate from what objects in the catalog or what reports initiate new candidate orbits. The output is an updated space catalog (or a new catalog of candidate orbits if no prior catalog is passed as input). All tracking systems have components shown in the bottom-half of the figure including estimation and filtering, scoring, complexity reduction, and other supporting algorithms and utilities. Numerica has innovated on many of these components as evidenced in its work on Gaussian sum filters,[9,10,11] the Gauss von Mises filter,[12] the implicit Runge-Kutta orbital and uncertainty propagator,[4,5] and metrics to test covariance and uncertainty realism.[13,14] Notwithstanding these contributions, what distinguishes MFAST from other tracking systems is primarily the middle row. By grouping reports into multiple frames (e.g., sensor scans) and then processing them over a sliding window, the MFAST system allows one to change decisions in the past to come up with a better decision today. Indeed, only association decisions at the back of the sliding window are made firm; elsewhere decisions are tentative and can be changed once more data is received. From an operational standpoint, MFAST can always return the best association decision at any time, if desired. For the data association component, MFAST uses a multiple frame assignment formulation of the data association problem (the MFA in MFAST) that is solved using an in-house Numerica solver designed for large-scale problems. Because MFAST is able to consider both orbit extensions and initiations within a common framework, it can perform key capabilities such as UCT resolution and catalog maintenance *jointly*, if desired.

In essence, there are many tracking paradigms. We choose MFAST because it works well at the system-level in light to medium clutter, it reduces to the simpler nearest neighbor method in the case of widely-spaced objects, and has the additional desirable features described in this section.

Implementation Details

The MFAST algorithm has been implemented in two concrete forms: a prototype written primarily in Python and used for development and experimentation, and a production version written entirely in C++. Both implementations take as input two types of information:

- A time-ordered list of observations, for example, in B3 format;
- A table of sensor information, including at a minimum latitude-longitude-altitude positions (for ground-based sensors) and sigma/bias estimates.

The production version of MFAST processes this data in a single batch run and upon conclusion returns a list of candidate orbits that contain:

- Orbital state estimate and covariance, as well as a two-line element (TLE);
- The observations associated to the orbit;
- A boolean specifying whether the orbit passed a series of user-configurable quality checks.

The implementations have been designed with flexibility in mind, and can be easily adapted to support alternate data formats. One version under transition, for instance, includes an interface that allows it to query the SQL databases used in that environment; while another version uses a C-level interface that allows the entire MFAST algorithm to be utilized via a single C function call within other executables. In addition, the production version supports a variety of platforms, and has been successfully installed and tested on consumer-grade hardware running Linux CentOS, Mac OS X, and Windows 7.

III. Data and Metrics

MFAST has processed optical and radar sensor data from a variety of real and simulated data sources. Under the GEO Odyssey program, MFAST has processed live UCT data generated from the SSN and other sensors, and has discovered new candidate orbits that have been sent to the JSpOC for follow-on tasking and eventual inclusion into the space catalog. Due to the sensitivity of this data, results cannot be shown in this paper. Instead, we present results from processing real-world historical radar and optical data from the SSN, collected during 2004.

All observations from the 2004 SSN dataset emanate from catalogued objects (and are tagged to said objects meaning they are not bona fide UCTs). To mimic UCT processing, the data is treated as a mock UCT problem, meaning that the true data associations (i.e., satellite number tags) are stripped from the data so that all observations can be treated as UCTs, and a new catalog of candidate orbits is built from scratch. One of the key challenges in processing this data effectively is the wide range of update frequencies encountered between the different objects. While some objects are observed with a high frequency, data collected from others is very sparse. In particular, almost half of the objects observed by optical sensors receive less than one observation per day. This data sparsity can stress the ability of any tracking system to initiate orbits in a timely manner from these less frequently observed objects.

Some of the metrics used for evaluating the performance of MFAST in "UCT mode" are the number of candidate orbits produced, total runtime, and the correctness ratio. The latter is a metric that other programs at Numerica have identified that addresses key issues such as observations that are left unassociated and cross-tagged observations. For each truth object assigned to an orbit, the correctness ratio is defined by

$$\text{Correctness Ratio} = \frac{C}{C+M+I}.$$

Here, C is the number of reports correctly associated to the orbit, M is the number of missing reports (i.e., reports contained in the dataset that are processed but are not assigned to the orbit), and I is the number of reports incorrectly associated to the orbit (i.e., cross-tags). One can take a (weighted) average of the correctness ratios over all truth objects assigned to orbits to come up with a single number to quantify tracking performance. Indeed, this correctness ratio is 100% if and only if the tracker achieves "perfect performance." If it is not 100%, one can also examine the correctness ratios of the individual truth objects to see which ones contribute to the imperfection. In analogy to the correctness ratio, one can also define an incompleteness ratio and a cross-tag ratio according to

$$\text{Incompleteness Ratio} = \frac{M}{C + M + I}, \quad \text{Cross-Tag Ratio} = \frac{I}{C + M + I}.$$

Note that that three ratios always sum to unity. Thus, in summary, the correctness ratio gives a macro perspective on how a tracking system is performing and, by virtue of the denominator term in its definition, provides an honest assessment of performance since it penalizes both for cross-tags as well as for missing observations. Full details on the correctness ratio and, in particular, the formulation of the two-dimensional assignment problem that is needed for determining how to assign generated orbits to truth objects in the dataset, are provided in the companion paper.[8]

IV. Data Processing Results and Discussion

This section communicates the results from MFAST that were obtained from processing the 2004 SSN dataset in a UCT processing mode. As mentioned in Section III, the ability to effectively process this data and produce accurate candidate orbits presents several challenges to any tracking system. The most notable challenge is that of data sparsity or the infrequency at which sensors observe many objects. This data sparsity can stress the ability of the tracking system to initiate orbits in a timely manner (or even at all). For sparse optical data, generating an initial track state and covariance from angle-only observations challenges batch least squares methods for initial orbit determination and, in some cases, can lead to a failure of the method to converge to an estimate. In the case of the 2004 SSN data, almost half of the objects observed by optical sensors receive less than one observation (i.e., pair of angles) per day. In some extreme cases, objects are only seen once or twice over the entire scenario duration. Since at least three pairs of angles are needed to estimate a full orbital state, there will inevitably be missing orbits; hence, it will be impossible for any tracker to achieve 100% correctness. Indeed, *we can only extract knowledge up to the limit of the information content in the data.*

Though not shown in this paper, using an earlier Python version of MFAST from September 2014, we achieved a respectable correctness ratio of 90% or better (with no cross-tags) when processing the optical portion of the 2004 SSN dataset for objects receiving at least four observations per day. While these initial results were very promising, we embraced some of the challenges described above and made a number of enhancements to MFAST, some of which are listed below.

- Added support to process "two-ob tracks" (i.e., tracks comprised of two angle-only pairs of observations) for ground- and space-based optical sensors (prior to this improvement, MFAST required at least three observations to initiate an optical track).

- Optimized the MFAST gating and complexity reduction algorithms in order to reduce runtime or, equivalently, to process larger data sets without increasing runtime.

- Modified the MFAST non-linear least squares solver (used in some of the gating and orbit determination methods) to improve convergence rates and accuracy of orbital estimates.

- Implemented a new post-processing "orbit stitcher" in MFAST. This feature fuses or stitches compatible orbits generated from radar sensors with compatible orbits generated from optical sensors in order to improve data association correctness, improve the accuracy and precision of the candidate orbits, and reduce the chance of producing a redundant or repeated orbit. The orbit stitcher also mitigates the problem of sparse data from objects that are infrequently observed by its ability to stitch orbits separated by long time gaps.

In what follows, we show results from the latest C++ version of MFAST that includes the improvements described above. To distinguish the performance of MFAST in processing radar and optical data, the optical and radar portions of the dataset were processed separately. These results are shown in Sections IV.A and IV.B, respectively.

A. Optical Data Processing

The optical portion of the 2004 SSN dataset under consideration contains 13830 observations (angle-only pairs) over 627 objects with a total duration of 10 days. As mentioned earlier, data sparsity is a challenge in this dataset; the average number of observations per day per object is only 2.2. Knowing that it would be impossible to generate orbits from all 627 objects and hence achieve perfect correctness, we filtered out the data by only processing data from those objects that received 30 observations or more over the scenario duration (translating to an average of 4 observations per day per object or roughly one optical track per day per object). This resulted in 9045 observations over 165 objects.

Results of processing this data are shown in the left column of Figure 2. In addition, we show results from processing random subsets of this data. In these smaller scenarios, we processed data from a random set of $N < 165$ objects. Several Monte-Carlo trials were performed; averages are shown on the graphs along with their respective one-standard-deviation error bars. The (averaged) correctness ratio ranged from 93.5% to 97.1%. The reason why perfect tracking performance was not achieved was not due to cross-tags (as all cross-tag ratios were 0%) but due to some of the observations not being associated to orbits, which contributed to non-zero incompleteness ratios.

In the second row of graphs in Figure 2, we show the number of orbits established versus the number of objects in the scenario. (Note that we required an "established" orbit to contain at least four tracks.) Here, perfect tracking performance implies that all data points lie on the dashed green line corresponding to a generated orbit from each truth object in the scenario. Given that the correctness ratios were 93.5% or higher across all scenarios, it is not surprising to see some missing orbits. In particular, in the largest 165 objects scenario, 153 orbits were discovered. Those missing tended to be very sparsely observed.

The graphs in the third row of Figure 2 show runtime versus scenario size. We see that this relationship is approximately linear (on a log-log plot) which implies that the runtime scales according to a power law in the number of objects. From the slope of the line, we estimate this scaling to be $O(N^{1.29})$. In particular, the largest 165-object 10-day scenario completed in just 12.6 minutes. We remark that all runtime results were obtained from running MFAST on a single-core of a consumer-grade laptop. Based on these results, MFAST scales well both in terms of runtime and tracking performance (correctness ratio) as the scenario size increases.

B. Radar Data Processing

The radar portion of the 2004 SSN dataset under consideration contains 334744 observations (range-azimuth-elevation triples) over 2742 objects with a total duration of 10 days. As was performed with the optical data, we filtered out the radar data by only processing observations from those objects that received at least 30 observations over the scenario duration. This reduced the total number of objects to 1892.

Results of processing the data over the 1892 objects are shown in the right column of Figure 2 as well as results from smaller scenarios generated by randomly selecting subsets of the 1892 objects. The (averaged) correctness ratio ranged from 95.2% in the smallest 5-object scenario to 98.2% in the largest 1892 object scenario. Similar to the results of the optical data processing, imperfect correctness was attributed to missing reports (i.e., observations in the dataset that were processed but not associated to any orbits) and not to cross-tags (as all cross-tag ratios were 0%). Evidently, the non-zero incompleteness ratios lead to missing orbits, as shown in the plot in the second row of the right column of Figure 2. In particular, in the largest 1892-object scenario, orbits for 98.4% of the objects (1862 in total) were discovered. With regards to runtime, the plot in the third row of Figure 2 demonstrates that MFAST scales exceptionally well in radar tracking scenarios in the number of objects N with a complexity of $O(N^{0.96})$. In particular, the largest 1892-object 10-day scenario completed in 42.0 minutes on a single core of a laptop computer.

V. Conclusions

Operators need a time-efficient way to track and maintain a more comprehensive catalog of small, dim, deep-space, and hard-to-acquire debris objects in space. Such tracking capabilities are needed to support conjunction assessment (out to 10 days or more) and other SSA functions. With the instantiation of new sensors that will be able to see all objects down to the size of a marble, there will be a proliferation of newly-detected objects and other UCTs. Although the JSpOC is currently meeting requirements, in the future it may struggle to manage a much-larger catalog with as many as half-a-million space objects. The future JSpOC will need to embrace new and innovative tools for tracking and data association in order to maintain custody of these hard-to-track objects, more efficiently process UCTs, and ultimately to see that the U.S. maintains its superiority in space.

This paper has demonstrated that Numerica's Multiple Frame Assignment Space Tracker (MFAST) provides a viable solution to support the future needs of space surveillance. Radar observations, optical sensor observations, and an optional prior catalog of space objects provide input data to MFAST. The MFAST software solves the problem of associating data to specific catalogued objects or to new objects. This solution is facilitated by the use of Numerica's multi-dimensional assignment solver, specifically designed for large-scale problems. MFAST outputs new orbits and updates to existing catalogued orbits. There are several key components contained within MFAST that have been customized for space: physics-based methods to quickly rule out infeasible hypotheses, new estimation and fusion techniques to improve orbital accuracy and covariance realism, and hypothesis scoring techniques based on a probabilistic Bayesian framework. Another innovative feature of MFAST is the concept of a "sliding window," which facilitates changes in past decisions to improve current ones and is especially valuable for resolving closely-spaced objects. The MFAST software is provided with a variety of user interfaces that can be customized to the specific tracking problem under consideration by the operator.

On a real-world historical dataset from the SSN processed in a "UCT mode," MFAST achieves a correctness ratio of 93.5% or higher on the optical portion of the data and 95.2% or higher on the radar portion of the data (for objects that receive, on average, at least three updates per day). These results are obtained with no cross-tags, and MFAST is able to process the data on a single core of a consumer-grade laptop computer in real-time or faster. Speaking to its high efficiency, the MFAST runtime scales well in the number of objects N with a complexity of $O(N^{1.29})$ for optical data and $O(N^{0.96})$ for radar data.

Acknowledgments

This work was funded, in part, by a Phase II SBIR from the Air Force Research Laboratory Aerospace Systems Directorate (FA8650-14-C-7472) and by a Phase II SBIR from the Air Force Research Laboratory Information Directorate (FA8750-12-C-0080).

Distribution

Approved for public release (Case Number 377ABW-2015-0613).

References

[1]Stauch, J., Jah, M., Baldwin, J., Kelecy, T., and Hill, K., "Mutual application of joint probabilistic data association, filtering, and smoothing techniques for robust multiple space object tracking," *Proceedings of the 2014 AAS/AIAA Astrodynamics Specialist Conference*, San Diego, CA, August 2014, Paper AIAA-2014-4365.

[2]Schumacher, P., Sabol, C., Segerman, A., Hoskins, A., and Coffey, S., "Search and Determine Integrated Environment (SADIE) for automated processing of space surveillance observations," *Proceedings of the 2014 AAS/AIAA Astrodynamics Specialist Conference*, San Diego, CA, August 2014, Paper AIAA-2014-4165.

[3]Poore, A. B., "Multidimensional assignment formulation of data association problems arising from multitarget tracking and multisensor data fusion," *Computational Optimization and Applications*, Vol. 3, 1994, pp. 27–57.

[4]Aristoff, J. M., Horwood, J. T., and Poore, A. B., "Implicit Runge-Kutta-based methods for fast, precise, and scalable uncertainty propagation," *Celestial Mechanics and Dynamical Astronomy*, Vol. 122, No. 2, 2015, pp. 169–182.

[5]Aristoff, J. M., Horwood, J. T., and Poore, A. B., "Orbit and uncertainty propagation: a comparison of Gauss-Legendre-, Dormand-Prince-, and Chebyshev-Picard-based approaches," *Celestial Mechanics and Dynamical Astronomy*, Vol. 118, 2014, pp. 13–28.

[6]Aristoff, J. M., Horwood, J. T., Singh, N., and Poore, A. B., "Multiple hypothesis tracking (MHT) for space surveillance: theoretical framework," *Proceedings of the 2013 AAS/AIAA Astrodynamics Specialist Conference*, Hilton Head, SC, August 2013, Paper AAS 13-705.

[7]Singh, N., Horwood, J. T., Aristoff, J. M., and Poore, A. B., "Multiple hypothesis tracking (MHT) for space surveillance: results and simulation studies," *Proceedings of the 2013 Advanced Maui Optical and Space Surveillance Technologies Conference*, Wailea, HI, September 2013.

[8]Horwood, J. T., Aristoff, J. M., Beach, D. J. C., Ferris, P. A., Mont, A. D., Singh, N., and Poore, A. B., "A correctness ratio metric for assessing data association methods in space surveillance," *Proceedings of the 2015 AIAA/AAS Astrodynamics Specialist Conference*, Vail, CO, August 2015, Paper AAS 15-673.

[9]Horwood, J. T. and Poore, A. B., "Adaptive Gaussian sum filters for space surveillance," *IEEE Transactions on Automatic Control*, Vol. 56, No. 8, 2011, pp. 1777–1790.

[10]Horwood, J. T., Aragon, N. D., and Poore, A. B., "Gaussian sum filters for space surveillance: theory and simulations," *Journal of Guidance, Control, and Dynamics*, Vol. 34, No. 6, 2011, pp. 1839–1851.

[11]Aristoff, J. M., Horwood, J. T., Singh, N., and Poore, A. B., "Non-linear uncertainty propagation in orbital elements and transformation to Cartesian space without loss of realism," *Proceedings of the 2014 AAS/AIAA Astrodynamics Specialist Conference*, San Diego, CA, August 2014, Paper AIAA-2014-4167.

[12]Horwood, J. T. and Poore, A. B., "Gauss von Mises distribution for improved uncertainty realism in space situational awareness," *SIAM Journal of Uncertainty Quantification*, Vol. 2, 2014, pp. 276–304.

[13]Horwood, J. T., Aristoff, J. M., Singh, N., Poore, A. B., and Hejduk, M. D., "Beyond covariance consistency: a new metric for uncertainty consistency," *SPIE Proceedings: Signal and Data Processing of Small Targets 2014*, Vol. 9092, 2014.

[14]Horwood, J. T., Aristoff, J. M., Singh, N., and Poore, A. B., "A comparative study of new non-linear uncertainty propagation methods for space surveillance," *SPIE Proceedings: Signal and Data Processing of Small Targets 2014*, Vol. 9092, 2014.

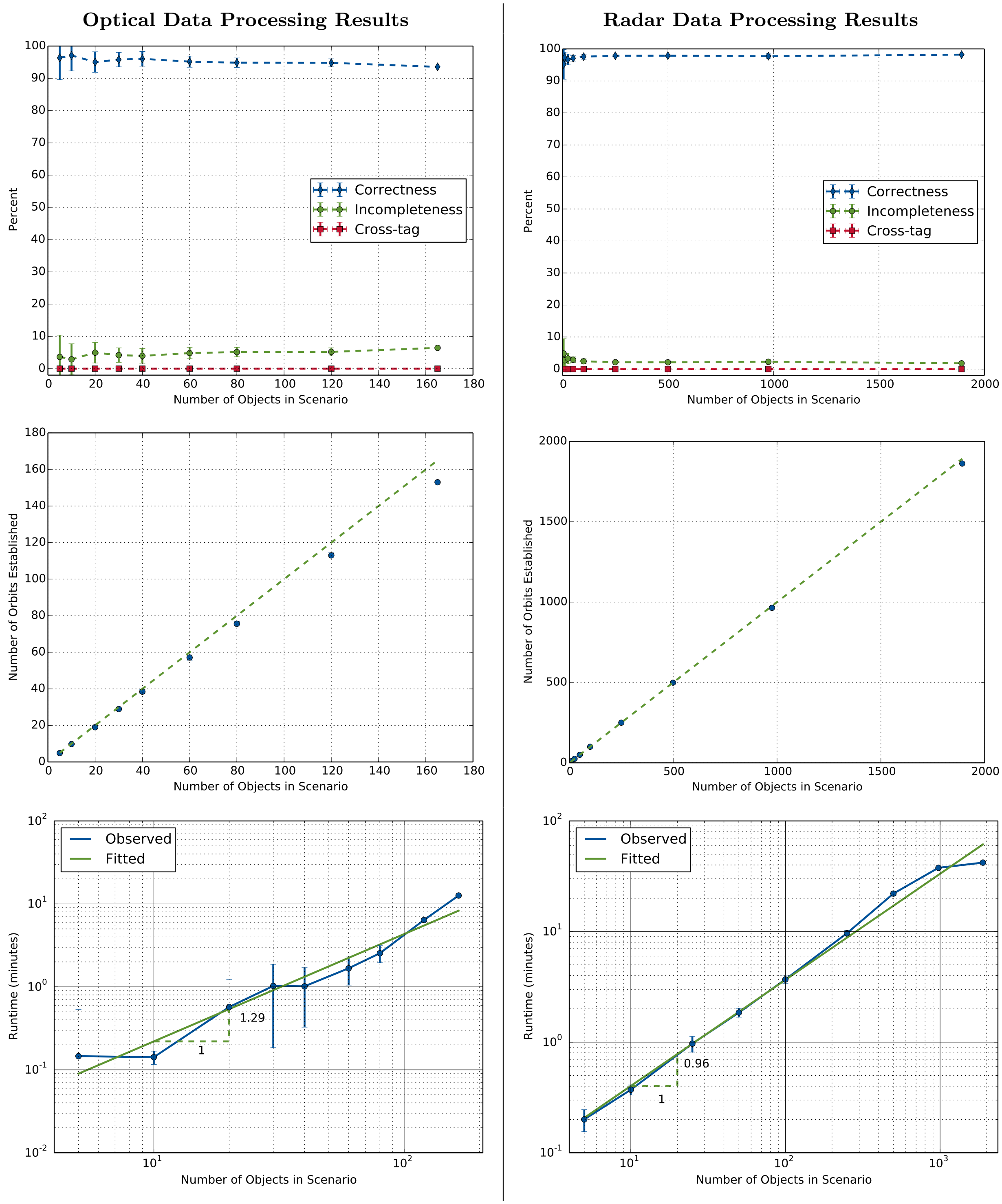

Figure 2. MFAST results from processing the 2004 SSN dataset in a UCT processing mode