Cloud-based Space Situational Awareness: Initial Design and Evaluation

Bingwei Liu[a], Yu Chen[a] , Dan Shen[b] , Genshe Chen[b], Khanh Pham[c], Erik Blasch[d]

[a]Department of Electrical and Computer Engineering, Binghamton University, SUNY, Binghamton, NY, USA;
[b]Intelligent Fusion Technology, Inc. Germantown, MD USA;
[c]Air Force Research Laboratory, Kirtland AFB, NM USA;
[d]Air Force Research Laboratory, Rome, NY USA

ABSTRACT

The need for a global collaborating space situational awareness (SSA) network, including radars, optical and other sensors for communication and surveillance, has become a top priority for most countries who own or operate man-made space-crafts. Such a SSA system requires vast storage, powerful computing capacity and the ability to serve hundreds of thousands of users to access the same database. These requirements make traditional distributed networking system insufficient. Cloud computing, which features scalable and elastic storage and computing services, has been recognized as an ideal candidate that can meet the challenges of SSA systems' requirements. In this paper, we propose a Cloud-based information fusion system for SSA and examine a prototype that serves space tracking algorithms. We discuss the benefits of using Cloud Computing as an alternative for data processing and storage and explore details of Cloud implementation for a representative SSA system environment.

Keywords: Cloud computing, space situational awareness, space surveillance and tracking.

1. INTRODUCTION

The spatial density space equipment around the Earth has increased drastically in recent decades. In 2012, there are 1046 operating satellites[1] orbiting the Earth, accompanied by more than 21,000 man-made debris objects that are larger than 10 cm.[2] The number of particles greater than 1cm is estimated to be approximately 500,000[2] and more than 100 million when the object size is less than 1cm.[2] The debris pose huge collision thread to active satellites and other spacecrafts. For example, the intentional destruction of Fengyun-1C and the Iridium 33-Cosmos 2251 collision produced thousands of new debris fragments. Operating man-made spacecraft, such as the International Space Station (ISS), have to conduct debris avoidance maneuvers (DAM) more frequently than before. If the ISS failed to acquire accurate knowledge of its surrounding environment, the cost of a collision will be unacceptable.

Although U.S. Space Surveillance Network (SSN) is maintaining and updating catalogs of orbital debris, they are not updated fast enough for today's demands. The original term of space surveillance was subsumed by space situational awareness (SSA), which is a broader term that includes debris detection and avoidance. The urgent need of a global collaborating SSA network, including radars, optical and other sensors, has become a top priority for most countries who own or operate man-made spacecrafts.

The European Space Agency (ESA) is deploying a SSA system to protect critical space infrastructure against space hazards.[3] The SSA Preparatory Programme (SSA PP), approved in November 2008, proposed a Common SSA Integration Framework (COSIF) based on a Service-Oriented Architecture (SOA) for the SSA system. The ESA SSA system includes three areas: Surveillance and tracking (SST) of objects in Earth orbit, space weather (SWE) monitoring and forecasting and near-Earth objects (NEO) surveillance and tracking. The deployment and implementation of an SSA system requires vast storage, powerful computing capacity and the ability to

Further author information: (Send correspondence to Genshe Chen)
Genshe Chen: E-mail: gchen@intfusiontech.com

Sensors and Systems for Space Applications VI, edited by Khanh D. Pham, Joseph L. Cox, Richard T. Howard, Genshe Chen, Proc. of SPIE Vol. 8739, 87390M · © 2013 SPIE · CCC code: 0277-786X/13/$18 · doi: 10.1117/12.2015525

serve hundreds of thousands of users to access the same database. This makes Cloud computing (CC) a suitable underlying storage and computing infrastructure for an SSA system.

As a new computing paradigm, Cloud Computing (CC) has attracted researchers from the distributed computing community and information technology (IT) service providers. The well-known attractive features of CC include on-demand scalability of highly available and reliable pooled computing resources, secure access to metered services from anywhere, and displacement of data and services from inside to outside the organization. Due to the low cost of storage services provided in a Cloud, compared with purchasing and maintaining a storage infrastructure, it is attractive to companies and individuals to outsource applications and data storage to public Cloud computing services.

Cloud computing allows users to focus on their application without worrying about an IT infrastructure plan. Central Processing Unit (CPU) cycles, storage space and even network services can be purchased on demand. By offloading data to the Cloud and mitigating part of their computing tasks, space agencies will be able to focus more on collecting and analyzing data without worrying about infrastructure planning and maintenance.

The ESA has formed a partnership with Amazon Web Services (AWS) to deploy its Data User Elements (DUE) program on Amazon's scalable Web storage and resizable computing environment.[4] Data collected by ESA satellites is provided to scientists, governmental agencies and private organization through AWS. The data is used for monitoring the environment, improving the accuracy of weather reporting, and assisting disaster relief agencies. In the future, space applications such as remote sensing can be deployed on the Cloud.

Tracking of space objects (SOs) is an important task for space situation awareness.[5,6] In addition to tracking accuracy,[7–9] the complexity of the tracking algorithms are just as important as storage requirements due to the large amount of SOs (over 20k) currently orbiting the Earth that need to be tracked. The selection of the tracking algorithm and its associated comparisons are important for sensor management,[10–12] collision avoidance,[13,14] and cooperative search.[15] In addition, the selection of the tracking algorithm and its results impact space situation awareness[16] and coordinated game-theoretical threat detection.[10,13,17]

In this paper, we propose a Cloud-based information fusion system for SSA and investigate the resource allocation strategy for a space tracking algorithm. We discuss the benefits of using Cloud Computing as an alternative computing and storage system and explore details of Cloud implementation for SSA.

The rest of this paper is organized as follows: Section 2 discusses the benefit of integrating Cloud Computing into SSA system. Then we propose a framework for Cloud-based SSA system in Section 3. The performance of a Cloud application for a space target tracking algorithm is evaluated for different sizes of virtual machine instances in Section 4. Finally, we conclude this paper in Section 5.

2. BENEFIT OF THE CLOUD

Integrating Cloud computing into the SSA system has at least the following advantages.

2.1 Release from infrastructure maintenance

A space agency usually needs to maintain computing clusters and storage systems in their own data center. If the computing and storage capacities are not satisfied, the space agency will need to purchase, deploy and maintain more resources. This can be costly and inefficient in some cases.

With Cloud computing, a space agency can pay a low price for computing and storage services without considering the scalability and availability of the system. For example, orbital debris position data and space weather data both need vast storage capacity. Most of this data will be shared with other space agencies to obtain higher position accuracy or earlier detection of debris for satellite avoidance maneuvering. The Cloud is a good choice of storing and sharing this data which in many cases is detected in a distributed fashion.

In addition, reducing local computing and storage tasks also means the decrease of energy and its associated cost. The cost of power is reduced because the energy needed for storage and computing is transferred to the Cloud provider.

2.2 Avoid Duplicated Investment

The deployment of ground sensors usually can not exceed a country's boundary. To acquire as much relevant data as possible, a space agency might choose to launch more space sensors, which will certainly increase total cost of the system. A collaborate centralized Cloud-based SSA system can be an optimal strategy to avoid duplicate investment for each space agency. Also, by combining data collected by distributed sensors around the world, the Cloud-based SSA system will have higher accuracy of detected debris positions than any individual agency can have.

2.3 Ease Data Management and Access

Big Data management is a new research area in both science and business applications. Managing Big Data requires a new storage framework and computing architecture. Cloud service providers usually has experience in managing Big Data which can be utilized for space object assessment. The data owner can save time and resources in developing a storage system for their data. In the end users' perspective, research communities can easily access data since it is centralized at the Cloud.

2.4 Flexible Computing

Although the supercomputer of every space agency is extraordinarily powerful, the increase of computing demands will grow beyond the capacity of the supercomputer at some point. A space agency can either plan ahead or take advantage of the pay-as-you-go feature of Cloud Computing. Planning ahead means an accurate prediction of computing needs in the future. When the prediction is not optimal, the advanced investment will become a squandered when there is no need for the advanced deployed resources. On the other hand, on demand computing is one of the most important and attractive features of Cloud computing. Cloud providers like Amazon are able to provide on demand high performance computing. Whenever more computing resources are needed, the space agency can use Cloud computing as a complementary method to augment its current computing center to accelerate flexible data processing.

2.5 Fast Application Development and Deployment

Fast deployment of applications is another benefit that Cloud Computing can provide. The development environment provided by the Cloud platform is designed to ease development of applications. The Cloud computing community has been developing easy to use application development and deployment tools, usually free, for developers and users. An orbital debris visualization application, for example, can be easily developed based on the database serving several space agencies.

3. A CLOUD-BASED SSA SYSTEM

3.1 Framework

A typical scenario of Cloud-based SSA system is depicted in Figure 1. Radars, satellites and other sensors owned by different space agencies collect data and send the data to a center of the owner agency. The owner agency decides what data they are willing to share with the community and transfer it to the Cloud. The Cloud provides almost unlimited storage capacity and optimized solution for Big Data. High performance computing is also available when desired.

An Application Programming Interface (API) layer allows SSA application developers in the community to create new applications. Various algorithms such as data analysis, tracking filters, and space weather monitoring; are ready to use in the API layer. On top of the API layer are applications (APPS) aimed at different purposes. Examples are collision avoidance, orbit determination, space object cataloging, and re-entry prediction.

For all interaction with data, the Cloud provides a uniform web interface for users and administrators.

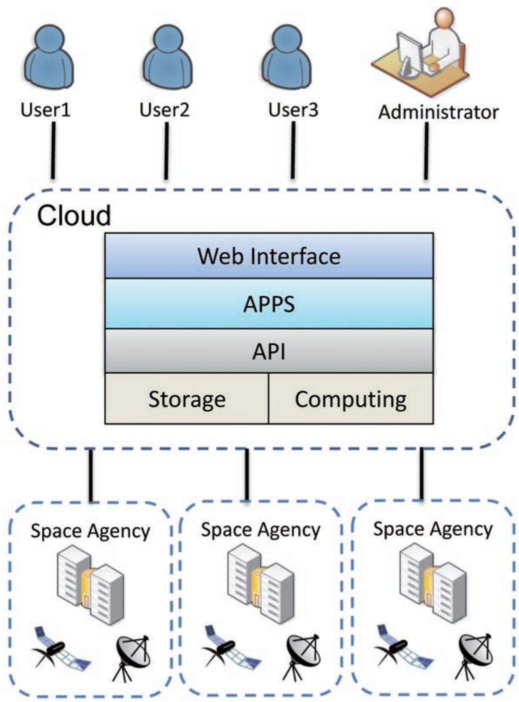

Figure 1. A typical scenario of a Cloud-based SSA system.

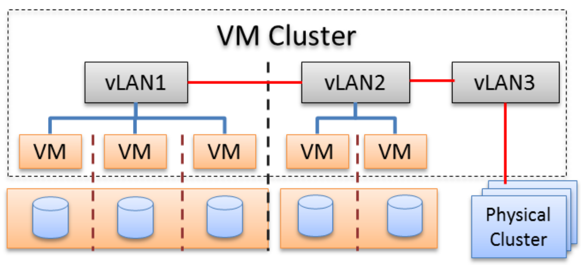

Figure 2. Dynamic computing cluster and storage system.

3.2 Cloud Side Consideration

Figure 2 illustrates a possible implementation at the Cloud side in order to provide high performance computing services to the public. The underlying Cloud infrastructure employs a dynamic virtual machine (VM) computing cluster and storage system. The following aspects are worth noting when designing a Cloud-based SSA system.

1. **Dynamic Computing Cluster.** The dynamic cluster is a computing cluster consisting of VM cluster and physical cluster. According to the system load, the Cloud will allocate appropriate computing resources to different tasks. The dashed lines in Figure 2 represent the appropriate isolation between every two VMs. One VM in the cluster is chosen as a computing scheduler. This scheduler monitors the performance and load of all VMs. We assume that this VM has strong secure environment that is difficult to be compromised. This is reasonable since we can always use a private Internet Protocol (IP) address for this VM and further protect it behind a secure firewall. This elastic computing structure is able to process sophisticated computing intensive tasks.

2. **Separated Storage.** Every VM can have one or more associated storage volumes. These volumes are assigned by the hyperviser and cannot be modified without root privilege in the hypervisor. User data security and integrity can also be included in the system by adopting storage integrity auditing service. A distributed storage system like Hadoop Distributed File System (HDFS)[18] could be employed as the underlying storage infrastructure.

3. **Distributing Functionalities into VMs.** Every individual functionality in our model can be implemented on a virtual machine with proper computing and storage resources to handle user requests, message exchanges or performance monitoring. This is an efficient and economical way to implement a complex system. The elasticity of the Cloud will also assure high availability of services. Once any functional VM fails, a new VM can be deployed in minutes to replace the poorly functioning VM.

3.3 Web Interface

Providing computing services through the web has been proved by public Cloud provider like Amazon to be a successful service delivery mechanism. We also recommend using RESTful web services[19] to implement Cloud applications.

The Cloud provides a uniform web interface for administrators, developers and regular users to access data and perform tele-operation to sensors. Administrators monitor the health of VMs, states of sensors and user behavior. Developers store program codes on the revision control and source code management system provided by the Cloud and can easily cooperate with each other on the hosted repository. Regular users visit the web interface to access history data, coordinate tele-operated sensors, request data processing and monitor VMs with owner privilege when desired.

3.4 Security

Security is usually the top concern to both a service provider and a user. The data owner wants to assure the safety of their property in the Cloud. Important and sensitive data in the Cloud should be stored securely and separately. The Cloud should also be able to appropriately defend against attacks to the web server and user data through specialized security groups and policies. The study of security in Cloud computing is still an open area, hence it needs to be considered carefully when putting SSA resources in the Cloud. A compromised server can send malicious commands to sensors causing tremendous lost for sensor owners.

To protect user data and ensure the security of the system, several policies need to be enforced:

1. **VM isolation.** As we mentioned above, VMs must be isolated by the hypervisor even when they are running on the same physical machine.

2. **Secure Storage.** The Cloud must implement secure storage of user data before deploying SSA services. The Cloud can provide an auditing service for users to guarantee data integrity. A third party auditor (TPA) can be employed to audit the Cloud in data integrity. The auditing procedure is usually a challenge-response style. A user or the TPA challenges the Cloud with the integrity of his data. The Cloud then responds with a message to prove that it is actually possessing the user's data and all data blocks are intact in the Cloud storage infrastructure.

3. **Network Management.** Each virtual Local Area Network (vLAN) has strict rules to prevent unauthorized access even within the same vLAN.Usually, the hypervisor will take care of the packet routing and the virtual network optimization. If necessary, a trusted network manager can be deployed to monitor the network with a proper permission.

3.5 Challenges

Although sensors that are distributed on the Earth collect sufficient data to form a global SSA estimate, there are still some obstacles towards a full SSA system.

1. **Data format standard.** Due to the lack of coordination, very versatile data standards or formats have been adopted by different agencies. The incompatibility among variant data formats is currently one of the main obstacles toward a centralized database of space data.

2. **Communication Delay.** The bandwidth from space sensors to the ground is usually low and not suitable for big file transmission. Customized local data pre-processing and aggregation technologies are imperative to reduce the redundancy or the less important data to be sent over the communication channel.

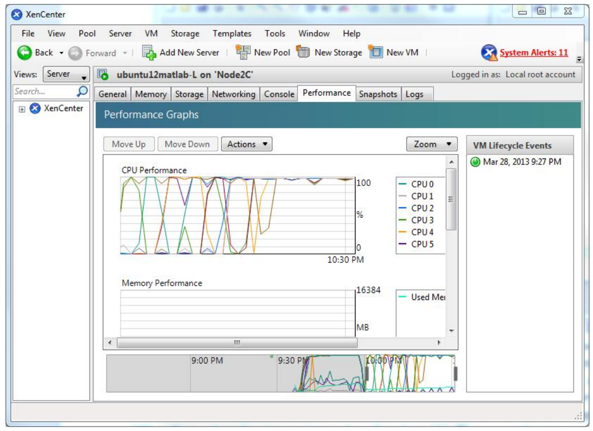

Figure 3. Performance monitoring of the Large VM instance.

3. **Real-time Application.** Some applications are not able to provide real-time access or remote control because the delay of communication channels. It is an interesting topic to consider some pseudo-real-time solutions, which can meet the time constraints of applications that are tolerable to certain delay.

4. PERFORMANCE EVALUATION

To evaluate the performance of the Cloud to enhance space target tracking filter algorithms, we setup an experiment in our Cloud testbed. This experiment explores the appropriate Central Processing Units (CPUs) and memory resources that should be allocated to a virtual machine on which the algorithm are running.

4.1 Experiment Setup

4.1.1 Cloud Testbed

Our Cloud testbed consists of 16 servers in our data center, all using Xen Cloud Platform (XCP) 1.6.[20] We choose Citrix XenCenter as our management software, which provides convenient management features for our experiment purpose. We can easily create, clone and move a virtual machine within the XenCenter. Live migrate, within the same pool or across different pools, is also an attractive feature. Except one Cloud server, each Cloud server is equipped with two Intel Xeon E5405 Quad-core processors at 2.4GHz, 32GB memory and 3TB storage. A more powerful physical server was used to run a VM with 24 cores and 60 GB memory to compare with other VMs. This server has 24 cores and 64GB memory in total. For this experiment, we only use a pool of four servers. Figure 3 shows the real-time monitor of the CPU, memory and network performance of the large instance in the experiment.

4.1.2 Virtual machines under comparison

Table 1. Specification of compared virtual machines

Item	Small	Medium	Large	XLarge	XXLarge
CPU	Xeon E5-2609				Xeon E5-2640
Frequency	2.4GHz				2.5GHz
Cores	2	4	8	8	24
Cache(MB)	10				16
Memory(GB)	4	8	16	30	62

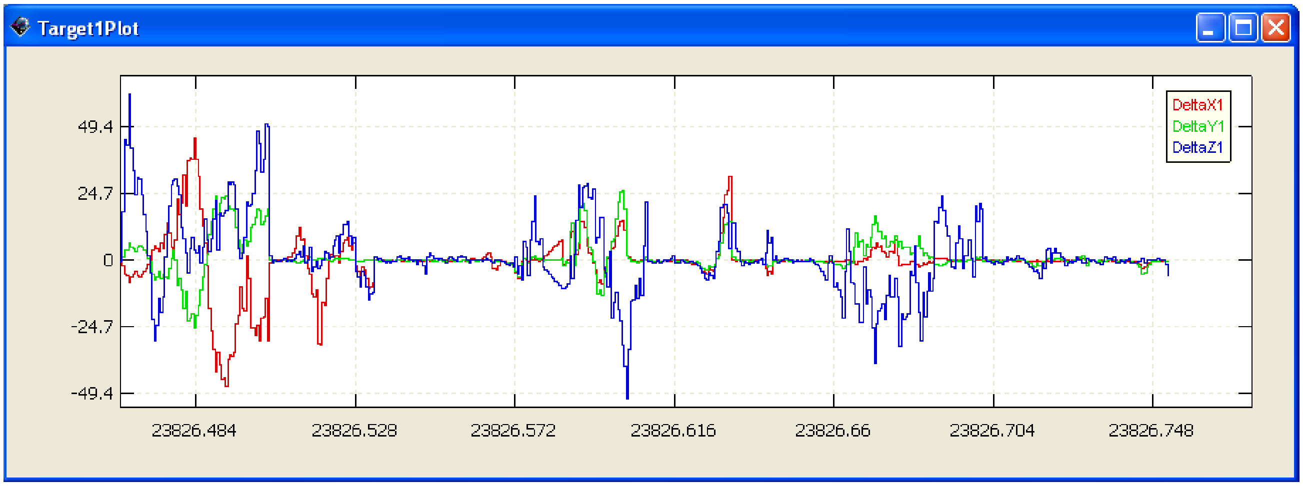

Figure 4. Tracking results of EKF (where DeltaX1, DeltaY1, and DeltaZ1 are the tracking in X-axis, Y-axis, and Z-axis, respectively)

To evaluate the performance differences among different virtual machines in the Cloud, we compared five types of instances in the Cloud. Table 1 lists the specification each type of virtual machine instances we used for performance comparison. We denote the VMs as Small, Medium, Large, XLarge and XXLarge instances according to their computing capacities. The small, medium and large instances double the number of virtual CPUs and memory each time. The XLarge instance only doubles memory compared to the Large instance. The XXLarge instance is intended to examine the effect of the increase of cores to the performance of the algorithm. All virtual machines use a Ubuntu 12.04 LTS operating system.

4.1.3 Algorithm

We consider a single Low Earth Orbit (LEO) observer that tracks a Geostationary Earth Orbit (GEO) satellite with possible maneuvering capability. The LEO orbit is nearly circular with a radius around 6600km and its position is assumed to be accurately calibrated by the Global Positioning System (GPS). The target being tracked is in a GEO with radius of approximately 42164km. We consider range, azimuth, and elevation (i.e. volume) measurements of the target without false alarms or missed detections except when the line of sight between the observer and the target is blocked by the Earth. The standard deviations of measurement error are 0.1km, 2mrad/sec, 2mrad/sec, for range, azimuth, elevation; respectively. The observer has a fixed sampling interval of 50s.

We consider the case that both the observer and the target share the same orbit plane. An Extended Kalman filter (EKF) was used to obtain the target position and velocity estimate. Figure 4 shows the target position estimate during 500 iterations. Along the orbit plane, the true target trajectory is close to a circle. The deviations in the estimated trajectory are due to Earth blockage. Note that as soon as the new measurements are available, the EFK can follow the target trajectory with a fairly accurate position estimate.

In our experiment, however, we turn off the display functionality and only calculate the estimate target position data.

4.1.4 Methodology of the experiment

Because of the flexible resource allocation of the Cloud, the first issue in designing a computing environment for a specific application is to decide how many resources we should assign to each VM for the application. In this experiment, we simulate a computing environment that many users, up to 30, are concurrently request computing for the EKF tracker algorithm. The average time to finish each request is used as our primary principle of comparison.

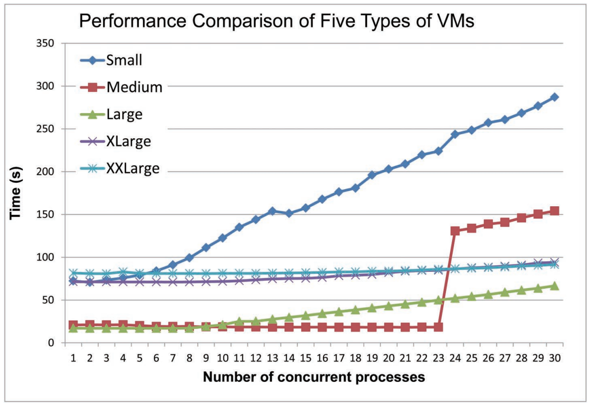

Figure 5. Performance comparison of five types of virtual machines.

4.2 Results

Figure 5 shows the results of the experiment. The Small instance has a response time proportional to the number of concurrent processes because of the limited number cores and amount of memory. The XLarge and XXLarge had almost the same response time from 1 to 30 concurrent processes, but both took more time than the Large instance. The Large instance, overall, has the best performance, at least for less than 30 processes. The number of cores in the XLarge and XXLarge instances did not help reduce the time for each process. The Medium instance satisfies a request less than that of 23.

As the results indicate, more cores and memory for a VM will not necessarily improve the performance of the application. For this specific tracking application, 4 cores and 8 GB of memory are enough for several dozens of requests. When the number of requests reach the level of hundreds of thousands, the Medium instance can be easily duplicated and deployed in a few minute. If the Cloud has a good prediction model for the number of users, it can easily scale up to hundreds of thousands of users without losing any performance.

5. CONCLUSION

A Cloud-based SSA system was proposed in this paper and a primitive experiment of resource allocation was investigated. We believe that the computing and storage power, combined with flexibility and scalability of Cloud Computing, will play an important role in deploying a global SSA system among multiple space agencies.

There are still a lot of effort needed for a relatively complete SSA network, such as the technical challenges we identified in this paper or an international agreement in sharing sensor data among multiple countries.

REFERENCES

[1] Union of Conerned Scientists, "Ucs satellite database," (2012). http://www.ucsusa.org.

[2] NASA, "Orbital debris frequently asked questions," (2012). http://orbitaldebris.jsc.nasa.gov/faqs.html.

[3] Johnston, S. J., O'Brien, N. S., Lewis, H. G., Hart, E. E., White, A., and Cox, S. J., "Clouds in space: Scientific computing using windows azure," *Journal of Cloud Computing: Advances, Systems and Applications* **2**(1), 2 (2013).

[4] "AWS case study: European space agency," (2011). http://aws.amazon.com/solutions/case-studies/european-space-agency/.

[5] Chen, G., Blasch, E., Chen, H., and Pham, K., "Multi-agent modeling and analysis for space situation awareness," (2009). http://spie.org/x36954.xml.

[6] Oliva, R., Blasch, E., and Ogan, R., "Applying aerospace technologies to current issues using systems engineering: 3rd aess chapter summit," *Aerospace and Electronic Systems Magazine, IEEE* **28**(2), 34–41 (2013).

[7] Chen, H., Chen, G., Blasch, E., and Pham, K., "Comparison of several space target tracking filters," in [*SPIE Defense, Security, and Sensing*], 73300I–73300I, International Society for Optics and Photonics (2009).

[8] Chen, H., Shen, D., Chen, G., Blash, E., and Pham, K., "Space object tracking with delayed measurements," in [*Proceedings of SPIE, the International Society for Optical Engineering*], Society of Photo-Optical Instrumentation Engineers (2010).

[9] Chen, H., Shen, D., Chen, G., and Pham, K., "Sensor scheduling for space object tracking and collision alert," in [*Dynamics of Information Systems: Mathematical Foundations*], 175–194, Springer (2012).

[10] Chen, G., Chen, H., Pham, K., Blasch, E., and Cruz Jr, J. B., "Awareness-based game-theoretic space resource management," in [*SPIE Defense, Security, and Sensing*], 73300P–73300P, International Society for Optics and Photonics (2009).

[11] Chen, H., Chen, G., Shen, D., Blasch, E. P., and Pham, K., "Orbital evasive target tracking and sensor management," in [*Dynamics of Information Systems*], 233–255, Springer (2010).

[12] Tian, X., Chen, G., Blasch, E., and Pham, K., "Comparison of three approximate kinematic models for space object tracking," in [*submitted to 2013 international conference on informationfusion*], (2013).

[13] Shen, D., Pham, K., Blasch, E., Chen, H., and Chen, G., "Pursuit-evasion orbital game for satellite interception and collision avoidance," in [*SPIE Defense, Security, and Sensing*], 80440B–80440B, International Society for Optics and Photonics (2011).

[14] Xu, P., Chen, H., Charalampidis, D., Shen, D., Chen, G., Blasch, E., and Pham, K., "Sensor management for collision alert in orbital object tracking," in [*SPIE Defense, Security, and Sensing*], 80440E–80440E, International Society for Optics and Photonics (2011).

[15] Shen, D., Chen, G., Pham, K., and Blasch, E., "A trust-based sensor allocation algorithm in cooperative space search problems," in [*SPIE Defense, Security, and Sensing*], 80440C–80440C, International Society for Optics and Photonics (2011).

[16] Blasch, E., Shen, D., and Pham, K., "Game-theoretical approach evaluation for space situation awareness," in [*Nat. Symp. On Sensor and Data Fusion*], (Oct. 2011).

[17] Blasch, E. P., Pham, K., and Shen, D., "Orbital satellite pursuit-evasion game-theoretical control," in [*Information Science, Signal Processing and their Applications (ISSPA), 2012 11th International Conference on*], 1007–1012, IEEE (2012).

[18] "Hadoop," (2013). http://hadoop.apache.org/.

[19] Fielding, R. T. and Taylor, R. N., "Principled design of the modern web architecture," *ACM Transactions on Internet Technology (TOIT)* **2**(2), 115–150 (2002).

[20] "Xen cloud platform," (2013). http://xen.org.