Opportunity-Adaptive QoS Enhancement in Satellite Constellations: A Case Study

Ann T. Tai Kam S. Tso
IA Tech, Inc.
Los Angeles, CA 90024

Leon Alkalai Savio N. Chau
Jet Propulsion Laboratory
Pasadena, CA 91109

William H. Sanders
University of Illinois
Urbana, IL 61801

Abstract

Systems that are formed by massively distributed mobile resources, such as satellite constellations, often provide mission-critical functions. However, many existing fault tolerance schemes and quality-of-service (QoS) management concepts cannot be applied to those systems in a traditional way, due to the dynamically and continuously changing readiness-to-serve of their mobile resources. In this paper, we describe a case study that investigates a method called "opportunity-adaptive QoS enhancement (OAQ)." The method exploits mobile-resource redundancy to mitigate the effects of a satellite constellation's structural degradation on position-determination (geolocation) accuracy. Accordingly, our algorithm is driven by an application-oriented QoS objective, focusing on a solution that permits a structurally degraded constellation to deliver geolocation results with the best possible quality. More specifically, the OAQ algorithm enables iterative geolocation accuracy improvement by letting neighboring satellites coordinate, and by progressively expanding the scale of this coordination in a window of opportunity that is dynamically determined by time limits, satellites' readiness-to-serve, and signal duration. Further, we define a QoS measure and evaluate it analytically. The results of the evaluation show that the OAQ approach significantly enhances a constellation's ability to deliver service with the quality at the high end of a QoS spectrum, even when there exists structural degradation.

Keywords: Opportunity-adaptive QoS enhancement, mobile resources, satellite constellation, QoS levels, model-based evaluation

1 Introduction

As micro-electro-mechanical systems and wireless networking technologies advance, many critical applications are intended to rely on a class of systems that are composed of massively distributed mobile resources. Examples of these include micro-UAV (unmanned aerial vehicle) swarms which perform coordinated actions in hazardous environments for damage control or monitoring, and micro-satellite constellations in which hundreds nodes coordinate for formation flying, surveillance, and communication. While their quality of service (QoS) is usually mission critical, they are often vulnerable to failures that are caused by adverse space-environment conditions, physical or other types of inadvertent faults, and malicious attacks. In addition, due to their mobile nature, the *readiness-to-serve* [1] of individual computing resources in those systems changes dynamically and continuously, making traditional redundancy-based fault tolerance schemes and QoS management concepts be difficult to apply.

In spite of their importance, fault tolerance and QoS management for systems built on massively distributed mobile resources have not yet received enough attention. To the best of our knowledge, besides the efforts concerning reliable inter-satellite and ground-to-satellite communications (see [2, 3], for example), no significant work has been devoted to method development for mitigating the effects on application-oriented QoS of satellite-failure-caused, constellation-structure degradation.

With the above motivation, we carry out a case study to investigate a method that allows us to exploit resource redundancy in a novel fashion, to mitigate the effects of a satellite constellation's structural degradation on signal-position-determination (geolocation) accuracy. Accordingly, our QoS objective is to guarantee the timely delivery of geolocation results with the best possible accuracy. As this QoS objective necessitates a cohesive formulation of fault-tolerant satellite constellation operation, our method derivation is based on the integration of concepts and techniques across the areas of satellite constellation and fault-tolerant computing. Specifically, it has been shown in the satellite research literature that sensor measurements accumulated successively by neighboring satellites can support an iterative weighted least-square algorithm and thereby enable a mechanism called "sequential localization" to reduce errors in signal-position determination [4, 5]. Although the original purpose of this mechanism was to circumvent the difficulties caused by satellite capacity inadequacy (e.g., insufficient number of onboard sensors) or noisy space environments, the synergy between the theoretical basis of sequential localization and the concepts of data diversity [6] and environment diversity [7] associated with fault-tolerant computing suggests that sequential localization can be judiciously exploited for tolerating the effects of failure-caused loss of satellites on geolocation quality. We thereby develop an algorithm which lets two or more surviving satellites that *consecutively* revisit a signal location coordinate for iterative geolocation-accuracy enhancement, in the situation where satellite failures reduce a constellation's "density" and make it no longer possible to let multiple satellites *simultaneously* "co-visit" the location to ensure result accuracy.

Moreover, the highly dynamic nature of satellite constellations leads us to introduce to the algorithm a concept called *opportunity-adaptive QoS enhancement* (OAQ). Accordingly, the algorithm enables the coordinated, iterative geolocation-accuracy enhancement be carried out in an aggressive fashion, by continuously expanding the scale of the coordination among peer satellites within a "window of opportunity." From temporal perspective, the window of opportunity is dynamically determined by the time allowance and signal duration. From spatial perspective, the opportunity is characterized by the number of mobile resources that are able to join the coordinated iterative geolocation computation. More specifically, those resources include: 1) the satellites that happen to be in the range that allows their footprints[1] to cover the signal location at the initial detection, and 2) those satellites whose routine traveling pattern brings their footprints to the target location subsequent to the initial detection and within the window of opportunity.

The opportunity-adaptive nature of our algorithm implies that the coordinated QoS enhancement is highly distributed, involving no team leader or decision authority. Rather, coordination expansion and termination are solely enabled by peer-to-peer message passing over the crosslink between neighboring satellites. More specifically, a coordination-request message provides sufficient information, which not only allows the receiving peer to carry out another round of accuracy-improvement iteration, but also enables the peer to determine, upon the completion of computation, whether the window of opportunity remains and permits further peer coordination.

The central purpose of this paper is to demonstrate the feasibility and effectiveness of the OAQ framework. Hence, in addition to describing the algorithm, we conduct a model-based quantitative evaluation to analyze the QoS gain from the use of the OAQ algorithm. The model is constructed based on a reference satellite constellation that is designed for detection and position localization of radio-frequency (signal) emitters [8]. Through analyzing the evaluation results, we show that the OAQ framework significantly enhances the system's ability to deliver service with the quality at the high end of a QoS spectrum, even after a significant number of satellites are lost due to faults.

The remainder of the paper is organized as follows. Section 2 provides background information. Section 3 describes the OAQ framework in detail, followed by Section 4 which presents an analytic model and discusses the evaluation results. Concluding remarks are given in Section 5.

2 Degradable QoS in Satellite Constellations

Since our objective is to investigate fault tolerance and QoS issues in the systems that are formed by massively distributed mobile resources, the types of satellite constellations we are concerned with are LEO (low earth orbit) constellations that comprise a large number of small satellites. Moreover, we focus on tactical and strategic applications. Hence, we view accuracy of signal-position determination as a crucial QoS property of a satellite constellation.

For clarity of illustration, we use the constellation shown in Figure 1 as the reference constellation. Nonetheless, the OAQ framework will also be applicable for other systems of similar type, and is anticipated to be more effective for systems built on a very large population of nodes, such as pico-satellite constellations. As mentioned in Section 1, this reference constellation is designed

[1] The area on the earth that is covered by a satellite is referred to as the footprint of that satellite.

for geolocation of radio-frequency (RF) emitters for surveillance applications.

The constellation is formed by seven orbital planes. (Informally speaking, an orbital plane is a ring-shaped trajectory along which satellites travel around the globe.) Each of the planes consists of 14 micro-satellites that are intended to be active in service, and two in-orbit spares that can be deployed to replace any failed satellites in the same orbital plane. Therefore, the constellation consists of 98 active satellites and 14 in-orbit spares (for a total of 112 satellites).

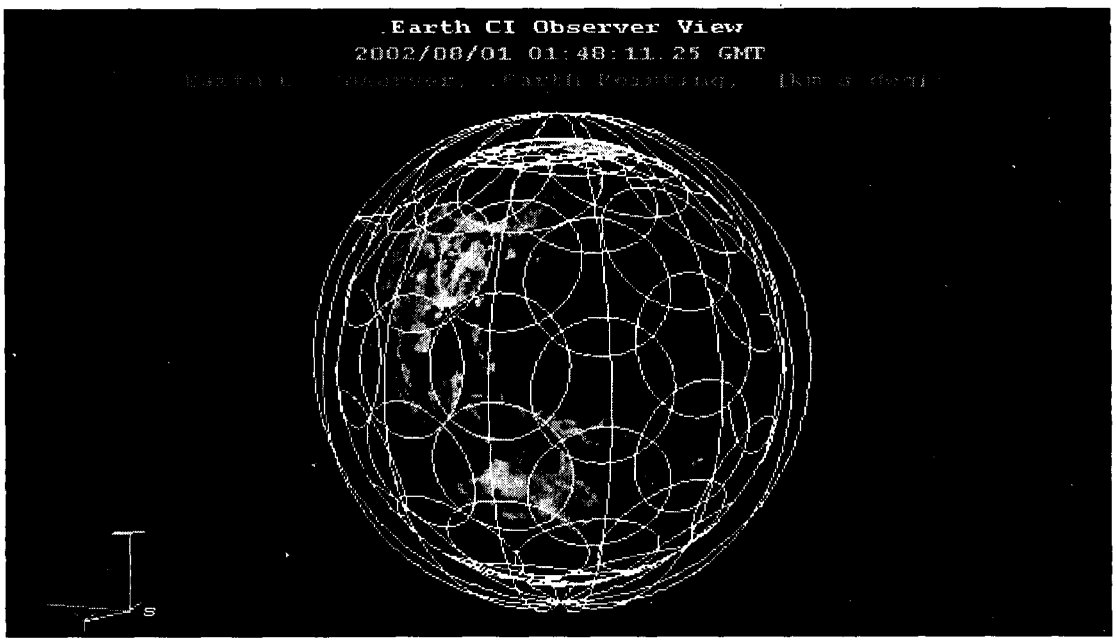
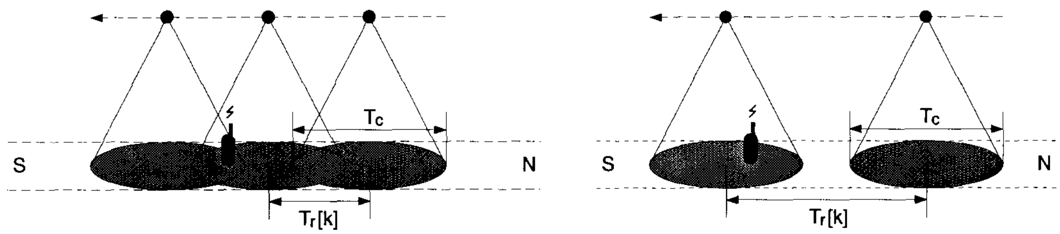


Figure 1: The Reference RF Geolocation Constellation

Figure 1 shows that when the constellation has 98 operational satellites, it offers a full earth coverage. Furthermore, every earth location will be covered by at least one satellite and a large portion of the globe (especially in the areas of high latitude) is covered by overlapped footprints. However, the geometry of the constellation will change if satellites are lost due to physical failures or malicious attacks. Specifically, when an orbital plane loses satellites after exhausting its spares, the surviving satellites will undergo a phasing adjustment so that they can be evenly distributed in the plane again. As a result, the overlapped portion of the footprints of adjacent satellites will shrink, which makes it less likely that a target will be captured simultaneously by multiple satellites. When more satellites fail, the footprints of surviving satellites will eventually become detached. Figures 2(a) and 2(b) illustrate the types of geometric orientation an orbital plane may exhibit. In the figures (where we rotate the axis of the earth 90° clockwise), the top dashed line indicates an orbital plane, while the small solid dots represent the satellites traveling in that plane; the shaded ovals are the satellites' footprints and the cellular phones emitting RF signals are the assumed targets.

As illustrated in Figures 2(a) and 2(b), we define *revisit time*, $T_r[k]$, as the time interval from the instant the center of a satellite's footprint passes a location on the earth to the instant the



(a) Overlapping ($T_r[k] < T_a$) (b) Underlapping ($T_r[k] \geq T_a$)

Figure 2: Node-Failure-Caused Structural Degradation

center of the footprint of the next satellite (in the same plane) passes the same location, given that the orbital plane has k operational satellites. Note that K, the number of operational satellites in a plane, is a random variable since satellites in the plane fail over time. Further, we use the term "coverage time," denoted as T_c, to refer to the maximum amount of time that a location on the earth can be covered by the footprint of a single satellite. Note that the length of T_c can be "visualized" as the diameter of a footprint, as shown in Figures 2(a) and 2(b). From the definitions of $T_r[k]$ and T_c, it follows that the geometric orientation of the footprint trajectory of an orbital plane can be determined by the relations between $T_r[k]$ and T_c. More precisely, $T_r[k] < T_c$ and $T_r[k] \geq T_c$ imply footprint overlapping and underlapping, respectively.

It is worth noting that the geometric orientation changes will affect the QoS of geolocation computation. In particular, when footprints overlap, it is possible that a target will be detected simultaneously by the footprints of adjacent satellites, which we call *simultaneous multiple coverage*. When two or more satellites observe a target at the same time, a measurement collection that is significantly more extensive than that from a single satellite can be obtained. With the added measurements, the ambiguity problem will practically disappear, resulting in a dramatic improvement of positioning accuracy [4]. Nonetheless, even when all satellites in the constellation are functioning, it is still possible that a target is only covered by a single satellite as the earth is not completely covered by overlapped footprints.

When a constellation successively experiences structural degradation due to loss of satellites, footprints will eventually become "underlapping" (this term is regarded as interchangeable with "detached" and "disjoint" in the remainder of this paper). Then, a target will be covered by a single footprint at a time at the best, which prevents a geolocation result from having high accuracy. And in the worst case, a target could escape from surveillance, if 1) the signal starts when its location is not covered by any footprints (i.e., in a "gap"), and 2) the signal stops before the nearest footprint moves to that location.

The above discussion implies that a constellation's structural degradation will lead to its QoS degradation. And since readiness-to-serve of each surviving satellite varies over locations and time, plus signal occurrence and duration are unpredictable, the extent to which we can pursue QoS enhancement in a structurally degraded constellation cannot be determined even if the geometric orientation of the constellation is known. In turn, these factors collectively suggest that an effective solution for QoS optimization should be opportunity-adaptive. Accordingly, we develop a framework as described in the next section.

3 OAQ Framework

3.1 Overview

It has been shown in the research literature that information from diverse sources can help resolve the ambiguity in signal position determination. Those information includes the previously calculated position coordinates and earlier measurements. Further, delayed position determination, termed as sequential localization, may help reduce errors in calculation because another satellite may appear in the range to cover the target, and additional measurements can thus be accumulated to support an iterative weighted least-square algorithm [4, 5]. Although the original purpose of sequential localization was to circumvent the difficulties occurring in the situations where satellites are not adequately equipped (with respect to quantity and capability of sensors) or to tolerate noisy space environments, the mechanism can be judiciously exploited for mitigating the effects of a constellation's structural degradation on geolocation accuracy. Specifically, we can let two surviving satellites that consecutively revisit the target coordinate for iterative position determination, in the circumstance where satellite failures reduce a constellation's "density" and make footprints become underlapped.

We can take a similar approach to QoS enhancement in the situation where the constellation has sufficient operational satellites such that portion of its earth coverage is made up by overlapped footprints. Specifically, if a signal is initially detected by a single satellite, we can withhold the preliminary result and wait to see whether overlapped footprints will arrive at that location before reaching the deadline for alert-message delivery[2]. If so, simultaneous multiple coverage will ensure a high-accuracy geolocation result which requires no further satellite coordination; otherwise the preliminary result will be enclosed in the alert message and sent to the ground.

While reaching a simultaneous coverage in the overlapping case implies the attainment of a geolocation result with the best quality and thus marks the completion of QoS optimization, the iterative QoS enhancement based on sequential localization in the underlapping case can be carried out progressively. Informally speaking, as additional information from diverse sources enables further accuracy-improvement iteration, we can continue to exploit the satellites that consecutively revisit the signal location until 1) the estimated error of the geolocation result drops below a threshold, 2) the alert-message delivery deadline becomes too close to allow another iteration, or 3) the signal terminates. Since this mechanism takes advantage of multiple satellites that revisit a signal consecutively, we call it "sequential multiple coverage."

Our framework is thereby an approach to progressive QoS enhancement via continuously ex-

[2]The deadline for alert-message delivery is the allowed time interval from the initial detection of a target to the final alert-message delivery.

panding the scale of the coordination among peer satellites, throughout a window of opportunity. While satellite coordination plays an important role in the framework, coordination expansion and termination are enabled by the message passing over crosslinks among neighboring satellites, as described in the next subsection (the code-like algorithm is provided and explained in the appendix).

3.2 Algorithmic Approach

Figure 3 provides several snapshots of a QoS optimization process, which illustrate how peer satellites coordinate through message passing at different stages. As shown in Figure 3(a), if S_1, the first satellite that detects the signal, sees further opportunity for QoS enhancement after completing its geolocation computation, it will send a coordination-request message to a peer S_2 that is expected to visit the target next. This message contains the initial measurements and preliminary result, so that S_2 can make accuracy improvement from there. By receiving the message, S_2 will obtain the information it needs for the next iteration of geolocation computation. Consequently, when its footprint moves to the target location, S_2 will be able to generate a result with a better resolution.

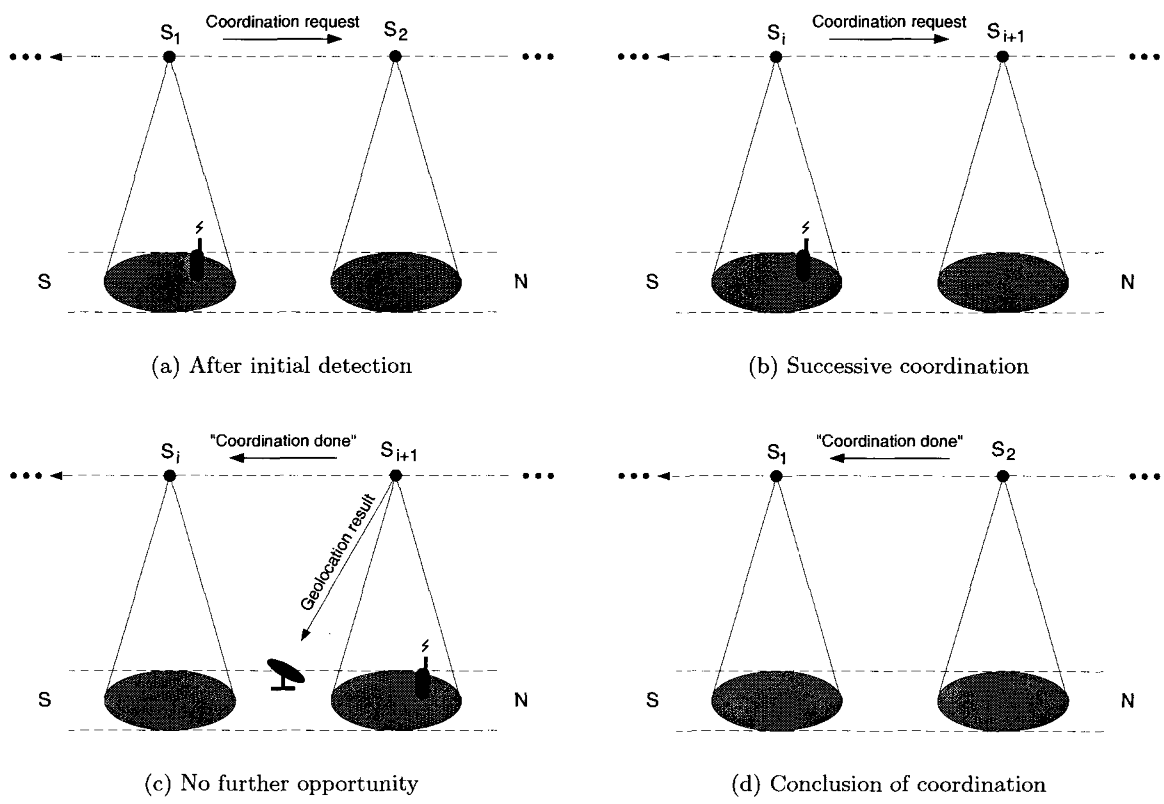


Figure 3: Coordinated QoS Enhancement

The coordination process will continue (see Figure 3(b)), along the chain consisting of satellites that revisit the target one after another[3]. Whereas the coordination will terminate when one of the following conditions becomes true:

TC-1) The estimated error becomes sufficiently small;

TC-2) The elapsed time since the initial detection exceeds a threshold; or

TC-3) The signal stops.

While TC-1 and TC-2 can be routinely checked at the end of each accuracy-improvement iteration so that the satellite that performs the computation can decide on whether it should request another peer to join the coordination, TC-3 can become true after a coordination request is made. Furthermore, the coordinated optimization is highly distributed in nature, meaning that there is no team leader or decision authority. Accordingly, coordination termination is also enabled by message passing between peer satellites, similar to coordination expansion. More specifically, as shown in Figure 3(c), when a satellite S_{i+1} completes computation and realizes that further coordination for QoS enhancement is impossible or unnecessary because one of the termination conditions holds, this satellite will enclose the final result in an alert message and send it to the ground station. The meanwhile S_{i+1} will send a "coordination done" message to S_i. This notified peer will then pass the message to S_{i-1}, and so on. In this manner, S_1, the satellite that performed the initial geolocation, will be notified at the end, as illustrated in Figure 3(d).

Now suppose that S_i does not receive a "coordination done" notification from S_{i+1} when the elapsed time since the initial detection exceeds a threshold which is a function of the alert-message-delivery deadline and S_i's ordinal number i (as described in the next paragraph). Then S_i will assume that S_{i+1} is unable to deliver the alert message because TC-3 becomes true before S_{i+1}'s footprint arrives at that location, as illustrated in Figure 4 (where the the shaded cellular phone with no emission represents a terminated signal). Consequently, S_i will consider its result as the final result and send it to the ground. Analogous to the case shown in Figure 3(c), a "coordination done" notification will be sent to S_{i-1} and propagated along the downstream of the chain.

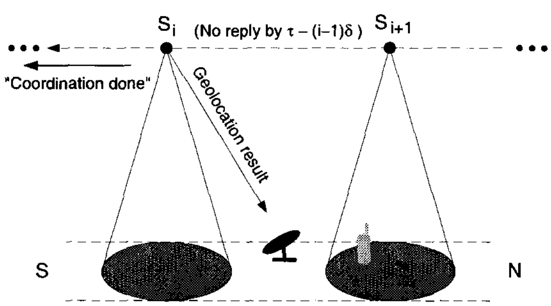


Figure 4: Guaranteed Geolocation Report

[3]For the sake of illustration, we assume here that the target is located near the center line of a plane's footprint trajectory so that the chain of coordinating satellites coincides with a portion of that plane. However, the algorithm itself is general and is not derived based on this assumption.

In order to ensure that all the participating peers, including S_1, receive the "coordination done" notification in time so that they will not be unnecessarily alarmed, the decision (made by a satellite that just completes geolocation computation) on whether to request the next arriving peer to join coordination must be made according to whether TC-2 has become true. More specifically, TC-2 is formulated by the expression $getTime() - t_0 > \tau - (n\delta + T_g)$, where t_0 is the point of initial detection, τ is the (system-level) deadline for alert-message delivery, δ is the maximum inter-satellite message-delivery delay, n is the satellite's ordinal number which identifies its position in the coordination chain, and T_g is the maximum amount of time required for a satellite to perform geolocation computation. Thus the right hand side of the above inequality can be regarded as the "local threshold" of the elapsed time, for S_n to determine whether it should request S_{n+1} to join the coordination. More precisely, if the inequality (i.e., TC-2) holds, then there will be no guarantee that S_{n+1} will complete the next iteration and all the satellites in the downstream can receive the notification from S_{n+1} in time; thus S_n will decide to stop the iterative accuracy improvement and send its geolocation result and "coordination done" message to the ground and S_{n-1}, respectively. By the same token, if TC-2 does not hold, then S_n will send a coordination request to S_{n+1} and will thereafter wait for the "coordination done" message so long as the condition $getTime() - t_0 < \tau - (n-1)\delta$ holds. If no such message is received from S_{n+1} when time expires, S_n will assume that S_{n+1} is unable to completes computation due to TC-3 or becomes fail-silent, and thus S_n will send its geolocation result and "coordination done" message to the ground and S_{n-1}, respectively, as described earlier and shown in Figure 4.

Alternatively, we may let S_{n+1}, the satellite that receives a coordination request but is unable to successfully carry out the computation be responsible for sending the result received from S_n to the ground. This would eliminate the need for the "coordination done" message passing along the downstream of the chain. However, with the backward-messaging scheme, the delivery of the alert message will be guaranteed even if S_{n+1} turns to be fail-silent in the middle of computation.

3.3 Discussion

The opportunity-adaptive nature of our approach thus permits us to strive for the best possible QoS, while guaranteeing that in the worst case, with high probability the preliminary geolocation result of will be delivered in a timely fashion. Therefore, the OAQ framework shares a conceptual basis with the imprecise computation scheme developed by Liu *et al* [9]. However, with the imprecise computation scheme, a task must be decomposed into mandatory and optional subtasks; the mandatory subtask is required for an acceptable result before the task deadline; the optional subtask refines the result and can be left unfinished at its deadline, if necessary, lessening the quality of the computation. In contrast, the OAQ algorithm does not require explicit, static task decomposition at the program-structure level; rather, the sequence and extent of result refinement depends upon a dynamically determined opportunity.

To contrast our framework with the opportunistic scheduling framework proposed by Raman `et al` [10], the matchmaker in that framework focuses on resource availability and system throughput, while we are concerned with a more comprehensive set of system attributes and our algorithm requires no decision authority. More importantly, since our approach is intended to deal with mobile system resources, the derivation of our algorithm is driven by the resources' readiness-to-serve, rather than the traditionally defined resource availability.

Finally, there are a couple of issues worth mentioning. First, since micro-satellites orbit the earth well below the Global Positioning System (GPS) constellation, each micro-satellite can observe multiple GPS satellites at any time. Then, with a GPS receiver onboard each spacecraft, precise clock synchronization among the micro-satellites can be easily achieved. It follows that the clock-drift problems that typically exist in distributed systems are not a concern in our algorithm. Accordingly, when a satellite makes a coordination request to a peer, to attach the value of t_0 (i.e., the point of initial detection) and ordinal number n will be sufficient for the peer to evaluate TC-2.

Second, if a peer other than the first and last in the coordination chain fails after making a coordination request, it will not be able to forward the "coordination done" notification to the peer in the downstream. In this circumstance, more than one satellite may assume itself as the last node that successfully completes geolocation computation and thus send its result to the ground. Since an alert message will include the ordinal number of the sender satellite, the ground station will choose to use the result marked with the highest ordinal number.

4 Model-Based Evaluation

4.1 Assumptions and Notation

In order to assess the effectiveness of the OAQ framework, we conduct a model-based evaluation. The analytic model is constructed according to the RF constellation described in Section 2. We choose to use this constellation for the quantitative study because 1) the design is conducted in house at JPL, and an interactive simulation model for visualization and coarse-grained quantitative measures (e.g., coverage time) is available, and 2) while its relatively small size allows a closed form solution of and efficient evaluation experiments for the QoS measure, the design principle of this constellation is consistent with those for constellations having a massive number of nodes. Therefore, evaluating this system suffices the purpose of feasibility and effectiveness demonstration.

We assume that the constellation is protected by scheduled and threshold-triggered ground-spare deployment policies. By "scheduled ground-spare deployment policy," we mean that ground spares will be launched to restore the constellation to its original capacity (so that it will again be equipped by a total of 112 satellites). Whereas "threshold-triggered ground-spare deployment policy" refers to the rule with which ground spares will be launched to restore an orbital plane to its original capacity (i.e., 14 active satellite plus 2 in-orbit spares), when the number of operational satellites in the plane drops to a threshold.

As shown in Figure 1, because of the sphere shape of the earth, the ratio of the areas covered by overlapped footprints to those covered by a single footprint changes across different latitudes. In particular, the ratio is the lowest at the equator and the highest at the poles. It follows that our assumed area of interest, which is around 30° north latitude, has a moderately high ratio of coverage. Further, as shown by Figure 1 and the interactive simulation generated by the Satellite Orbit Analysis Program (SOAP), around 30° north (or south) latitude, a location on or near the center line of a footprint trajectory will be least likely to be covered by overlapped footprints, while the two sides of a footprint always have a significant portion that is overlapped with the footprints of those satellites in the adjacent planes[4]. Hence, the situations in which a signal is located at or near the center line of a footprint trajectory can be regarded as the worst case, given that the emitting source is around 30° latitude. In order to be conservative and keep the complexity of the analytic model manageable, we let the QoS measure be formulated based on this worst case. In addition, we assume that satellite failure will not occur in the involved plane from the initial signal detection to the completion of the coordinated geolocation computation. This is a reasonable assumption because based on the conservative assumption for the signal position, the coordination chain will involve at most two satellites for the constellation in question (as shown in Section 4.2.1), and thus the likelihood that one of the coordinating satellites fails during that interval will be negligible. Accordingly, we do not consider the backward-messaging scheme in this evaluation. However, the analytic model we develop in Section 4.2 can be rather easily extended to relax this assumption.

Before we proceed to describe the model, we define the following notation:

- λ Satellite failure rate.
- μ RF signal termination rate.
- ν Iterative geolocation-computation completion rate.
- ϕ Scheduled time interval to the ground-spare deployment for constellation capacity restoration.
- η Threshold for the number of operational satellites in an orbital plane, which will trigger emergency ground-spare deployment for plane capacity restoration.

4.2 Model

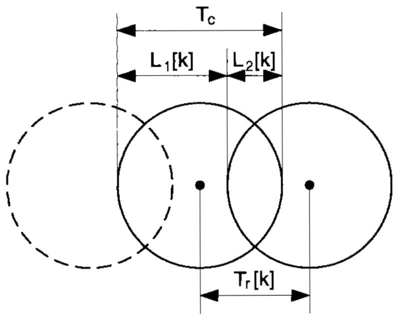
4.2.1 QoS Measure Formulation

Since the goal of the OAQ approach is to strive for the best possible accuracy for position determination with respect to a dynamically determined opportunity, we define a measure that quantifies a system's ability to deliver service in terms of QoS levels. More specifically, if the service delivered by the constellation can be rated by n QoS levels, we can let Y be a random variable that takes its value from the set $\{y \mid y = 1, 2, \ldots, n\}$. We thereby let the QoS measure be the probability that the system will deliver a geolocation result for a target signal with the quality at level y or above (given that a signal occurs). More succinctly, we choose $P(Y \geq y)$ as the QoS measure. In order to determine a QoS spectrum that enumerates all the QoS levels relevant to the system in question,

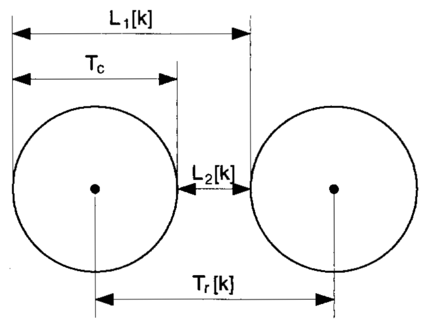
[4] This is ensured by 1) the threshold-triggered ground-spare deployment policy, and 2) phasing adjustment after losing satellites and before the deployment of ground spares.

we begin with analyzing the relationships between system behavior and the geometry properties of the constellation.

As described earlier, when an orbital plane loses satellites, the surviving satellites will undergo phasing adjustment and thus become evenly spaced in a plane again, resulting in changes of the geometrical relations between the footprints of adjacent satellites. And as explained earlier, the geometrical relations between adjacent satellites can be described in terms of $T_r[k]$ and T_c. Clearly, a decrement of k will result in an increased value of $T_r[k]$. Thus, the initial relation $T_r[k] < T_c$ will eventually change to $T_r[k] \geq T_c$, as shown by Figures 5(a) and 5(b). Since $T_r[k] \approx \theta/k$, where θ is the time required for a satellite to orbit through the plane and equals 90 minutes for the constellation in question, we have the estimated values of $T_r[k]$ shown in Table 1. Since a satellite's coverage time in the constellation is 9 minutes (i.e., $T_c = 9$), the values of $T_r[k]$ shown in Table 1 indicate that the underlapping scenario will happen when k is dropped to below 11.



(a) Overlapping ($T_r[k] < T_c$)



(b) Underlapping ($T_r[k] \geq T_c$)

Figure 5: Geometry Relations of Footprints

Table 1: Estimated Values of Revisit Time

k	14	13	12	11	10	9
$T_r[k]$	6.4	6.9	7.5	8.2	9.0	10.0

To facilitate the formulation and solution of the QoS measure, we introduce two auxiliary parameters $L_1[k]$ and $L_2[k]$, as shown in Figures 5(a) and 5(b). They are function of $T_r[k]$ and T_c, and can be calculated as follows:

$$L_1[k] = \left(T_r[k] - \frac{T_c}{2}\right) + \frac{T_c}{2} = T_r[k], \qquad L_2[k] = |T_c - L_1[k]| = |T_c - T_r[k]|$$

We can now analyze $M[k]$, the upper bound of the number of satellites that will consecutively capture a signal S, given that the involved plane has k operational satellite and has an underlapping structure. In accordance with our earlier assumption, the location of S will be along the center line of a footprint trajectory. From Figure 5(b), it follows that for the best case, S would start at a point when its location is covered by the edge of a footprint so that S will have the maximum likelihood to be covered by the forthcoming satellite. Thus, the sufficient condition for S to be covered by another satellite is $\tau > L_2[k]$, given that S does not terminate before the satellite's arrival. Then the number of satellites that will consecutively revisit the location of S thereafter before deadline τ will be bounded above by $\left\lfloor \frac{\tau - L_2[k]}{L_1[k]} \right\rfloor$. The upper bound $M[k]$ can then be expressed as a function of deadline τ, and auxiliary parameters $L_1[k]$ and $L_2[k]$:

$$M[k] = \begin{cases} 2 + \left\lfloor \frac{\tau - L_2[k]}{L_1[k]} \right\rfloor & \text{if } \tau > L_2[k] \\ 1 & \text{otherwise} \end{cases} \tag{1}$$

As explained earlier, adjacent satellites in an orbital plane will be underlapping if $k < 11$. As a result, Equation (1) implies that the upper bound for the number of satellites that will consecutively revisit a signal is two, if τ is less than 9 minutes (i.e., $M[10] = M[9] = 2$).

Together with the possible scenarios in which a signal may 1) be captured by a simultaneous dual coverage, 2) be covered by just a single footprint, and 3) escape from surveillance, the above analysis implies that the satellite constellation in question will have a 4-level QoS spectrum, as illustrated in Table 2. We have thus completed measure formulation and are ready to proceed to discuss the method for solution.

Table 2: QoS Levels vs. Geometry Properties

Geometry properties	$Y = 3$ (Simultaneous dual)	$Y = 2$ (Sequential dual)	$Y = 1$ (Single coverage)	$Y = 0$ (Missing target)
$T_r[k] < T_a$	√		√	
$T_r[k] \geq T_a$		√	√	√

4.2.2 Measure Solution

The relationships between the constellation's structural degradation and QoS ranking lead us to choose a decomposition approach for the measure solution. Furthermore, since 1) the measure is defined based on the assumption that the emitter is located at or near the center line of a footprint trajectory, and 2) there are no shared spares between orbital planes, measure solution will be independent of the effect of neighboring planes' structure variation. Accordingly, we start solution derivation from the following expression:

$$P(Y \geq y) = \sum_{Y=y}^{3} \sum_{k=0}^{14} P(Y = y \mid k)\, P(k) \approx \sum_{Y=y}^{3} \sum_{k=9}^{14} P(Y = y \mid k)\, P(k) \tag{2}$$

where we neglect the terms concerning the cases in which $k < 9$ because the scheduled and threshold-triggered ground-spare deployment policies make those cases extremely unlikely. Also, we do not consider the scenario in which a signal occurs at a point when the plane is undergoing phasing adjustment since the probability is very low at the steady state.

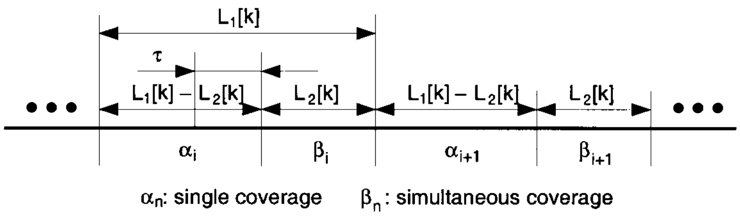
Note that Equation (2) indeed decomposes $P(Y \geq y)$ into two constituent measures:

1) The conditional probability that the system will deliver a geolocation result rated QoS-level y, given that the involved plane has k operational satellites, i.e., $P(Y = y \mid k),\ k \in \{9, \ldots, 14\}$.

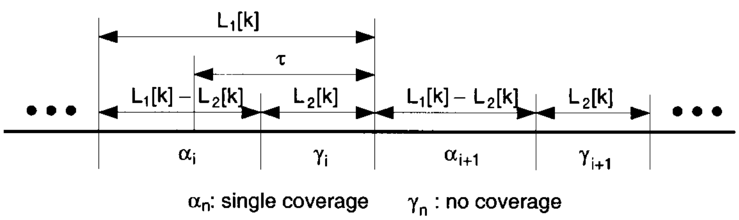
2) The probability that the involved plane has k operational satellites, i.e., $P(k),\ k \in \{9, \ldots, 14\}$.

Note that $P(k)$ suggests a Markov regenerative process because of the scheduled ground-spare deployment policies. Hence we use *UltraSAN* [11] which supports deterministic activity time, to compute the steady-state probability $P(k)$. Steady-state solutions are feasible for Equation (2) because the occurrence of an RF signal is assumed to be a Poisson process and the probability structure of what an arrival observes is identical to the steady state probability structure of the system [12].

Solving for $P(Y = y \mid k)$ is less trivial. So we analyze the problem from the two timing diagrams shown in Figure 6. In Figure 6(a), which is intended to support the analysis for the overlapping case, we break the time horizon into intervals α_n and β_n. Relating this time diagram to Figure 5(a), interval α_n corresponds to the duration (with a length $L_1[k] - L_2[k]$) through which a location on the earth is covered by a single footprint; while interval β_n corresponds to the duration (with a length $L_2[k]$) through which a location is covered by the overlapped footprints. The timing diagram in 6(b), which is intended to support the analysis for the underlapping case, is drawn in an analogous way but interval γ_n corresponds to the duration through which a location on the earth is not covered by any footprints (see Figure 5(b)).



(a) $T_r[k] < T_c$



(b) $T_r[k] \geq T_c$

Figure 6: Timing Diagrams

Based on the timing diagrams, we derive the following theorems, which lead to the solution of the conditional probability $P(Y = y \mid k)$:

Theorem 1 *When $T_r[k] < T_c$, position determination of a signal S can be accomplished by a simultaneous multiple coverage only if it occurs in 1) interval β_i, or 2) interval α_i, with at most τ or $L_1[k] - L_2[k]$ time units, whichever is smaller, away from interval β_i.*

Proof. From the definitions of intervals α_i and β_i (see Figure 6(a)), a necessary condition for a signal S to be captured by a simultaneous multiple coverage is that either S initiates in β_i, or S begins at a point in α_i so that the time required for reaching β_i is less than τ. Hence, if τ is less than the length of α_i, which is $L_1[k] - L_2[k]$ time units, the starting point of S must be at most τ time units away from β_i, since further waiting for footprint overlapping will not be allowed after τ. But if τ is greater than the length of α_i, the starting point of S can be anywhere that is at most $L_1[k] - L_2[k]$ time units away from β_i (otherwise the situation is reduced to the case in which S starts in β_{i-1}). Q.E.D.

Theorem 2 *When $T_r[k] \geq T_c$, position determination of a signal S can be accomplished by a sequential multiple coverage only if 1) $\tau > L_2[k]$, and S occurs in interval α_i with at most $L_1[k]$ or τ time units, whichever is smaller, away from α_{i+1}, or 2) $\tau > L_1[k]$, and S occurs in interval γ_i with at most $L_1[k] + L_2[k]$ or τ time units, whichever is smaller, away from α_{i+2}.*

We omit the proof of Theorem 2 here due to space limitations. The proof is slightly more complicated than but similar to the proof for Theorem 1. Note that the second (alternative) necessary condition in Theorem 2 will never hold in this analysis, since $L_1[k] \geq 9$ is true for all the underlapping cases (see Table 1) while we assume $\tau < 9$ (as described in Section 4.2.1). Accordingly, Theorems 1 and 2 lead us to define two more auxiliary parameters to facilitate the solution derivation for $P(Y \geq y \mid k)$:

$$\hat{L}[k] = \min\{L_1[k] - L_2[k], \tau\}, \qquad \tilde{L}[k] = \min\{L_1[k], \tau\}$$

In addition, since we wish to distinguish the overlapping case from the underlapping case, we define an indicator variable $I[k]$ as follows:

$$I[k] = \begin{cases} 1 & \text{if } T_r[k] < T_a \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

Note that the definition of the QoS measure implies that we are concerned with the system's ability to deliver a position-determination result for an RF signal, given that such an RF emitting event occurs. In turn, this means that the target signal must occur in one of the intervals marked α_n or β_n in the timing diagram shown in Figure 6(a) (or, one of the intervals marked α_n or γ_n in the timing diagram shown in Figure 6(b), for the underlapping case). Then, since we assume that the signal occurrence is a Poisson process, the distribution of the instant when the RF emitting event occurs is the same as the uniform distribution of the event over the same interval [13]. Consequently, if we let $G_3[k]$ denote the probability that the system will deliver a geolocation result rated QoS-level 3 (i.e., the position of a signal is determined by a simultaneous multiple coverage) given that the involved plane has k operational satellites (such that $I[k]=1$), we have

$$\begin{aligned} G_3[k] &= \int_0^{\hat{L}[k]} \frac{1}{L_1[k]} \left(1 - \int_0^{\hat{L}[k]-x} f(y)dy\right) \int_0^{\tau-(\hat{L}[k]-x)} h(z)dz\,dx \\ &\quad + \int_0^{L_2[k]} \frac{1}{L_1[k]} \int_0^{\tau} h(z)dz\,dx \end{aligned} \tag{4}$$

where $f(y) = \mu e^{-\mu y}$ and $h(z) = \nu e^{-\nu z}$ are the probability density functions of signal duration and iterative geolocation computation time, respectively. Note that in the first summon of Equation (4), the expression in the parentheses computes the probability that the signal does not terminate before the arrival of the overlapped footprints, while the third integral evaluates the probability that the iterative computation completes before the deadline is reached. Note also that the upper limits of the integrations in the first summon are defined based on Theorem 1. The expression does not include a term for evaluating the time of the initial geolocation computation because this procedure normally well overlaps with the time to reaching the footprint of the next satellite.

And since QoS-level 3 can be achieved only if footprints overlap, $P(Y=3 \mid k)$ has the following expression:

$$P(Y=3 \mid k) = \begin{cases} G_3[k] & \text{if } I[k]=1 \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

Likewise, we let $G_2[k]$ denote the probability that the system will deliver a geolocation result rated QoS-level 2 (i.e., the position of a signal is determined by a sequential multiple coverage) given that the involved plane has k operational satellites (such that $I[k]=0$), we have

$$G_2[k] = \int_0^{\tilde{L}[k]-L_2[k]} \frac{1}{L_1[k]} \left(1 - \int_0^{\tilde{L}[k]-x} f(y)dy\right) \int_0^{\tau-(\tilde{L}[k]-x)} h(z)dz\,dx \tag{6}$$

where the upper limits of the integrations are defined based on Theorem 2. Since QoS-level 2 is relevant only for the case in which footprints underlap, we have

$$P(Y=2 \mid k) = \begin{cases} 0 & \text{if } I[k]=1 \\ G_2[k] & \text{otherwise} \end{cases} \tag{7}$$

Further, we let $G_0[k]$ denote the probability that the system will be unable to deliver an alert message (which implies the zero QoS level), given that the involved plane has k operational satellites (such that $I[k]=0$), we have

$$G_0[k] = \int_0^{L_2[k]} \frac{1}{L_1[k]} \int_0^{L_2[k]-x} f(y)dy\,dx \tag{8}$$

Finally, based on the relationships between QoS levels and geometry relations of the adjacent satellites, as illustrated in Table 2, we can evaluate $P(Y = 1 \mid k)$ as follows:

$$P(Y = 1 \mid k) = \begin{cases} 1 - G_3[k] & \text{if } I[k] = 1 \\ 1 - G_2[k] - G_0[k] & \text{otherwise} \end{cases} \tag{9}$$

To this end, if we plug the results of Equations (5), (7) and (9), and the results of $P[k]$ computed using *UltraSAN*, into Equation (2), we obtain the QoS measure's final solution.

4.3 Evaluation Results

Applying the analytic model developed in Section 4.2, we evaluate the QoS measure based on various system parameters. In order to analyze the QoS gain from the use of the OAQ algorithm, we also compute the measure for the basic fault-adaptive QoS enhancement (BAQ) scheme for comparison. The BAQ scheme refers to the case in which the constellation is equipped with in-orbit spares and protected by the scheduled and threshold-triggered ground-spare deployment policies, but does not apply the opportunity-adaptive algorithm. Thus under the BAQ scheme, a geolocation result is delivered to the ground after the initial computation (based on a single or simultaneous coverage), which does not take advantage of the possible subsequent revisits by other satellites for coordinated QoS optimization.

We first evaluate the constituent measure $P(Y \geq y \mid k)$ using *Mathematica*™. Tables 3 and 4 display the results for the OAQ and BAQ schemes, respectively. The quantitative results are computed based on the parameter values $\tau = 5$, $\mu = 0.5$, $\nu = 30$ (time is quantified in minutes by default). As described in Section 2, the values of θ and T_c are 90 and 9, respectively, for the constellation in question.

Table 3: Conditional Probability of QoS Level with OAQ Scheme ($\tau = 5$, $\mu = 0.5$)

k	$P(Y = 3 \mid k)$	$P(Y = 2 \mid k)$	$P(Y = 1 \mid k)$	$P(Y \geq 1 \mid k)$
14	0.665889	0.000000	0.334111	1.000000
13	0.563275	0.000000	0.436725	1.000000
12	0.444406	0.000000	0.555594	1.000000
11	0.324039	0.000000	0.675961	1.000000
10	0.000000	0.203672	0.796328	1.000000
9	0.000000	0.104611	0.874083	0.978694

Tables 3 and 4 together show that, the OAQ scheme significantly enhances a system's ability to perform at the high-end of a QoS spectrum, even after a significant number of satellites are lost due to faults. In particular, as shown in Table 3, even when $k = 12$ (which implies two more satellite failures after spare exhaustion and a total loss of 25% orbital-plane capacity), with probability 0.44 the constellation will still be able to deliver a geolocation result rated QoS-level 3, and with

Table 4: Conditional Probability of QoS Level with BAQ Scheme ($\tau = 5,\ \mu = 0.5$)

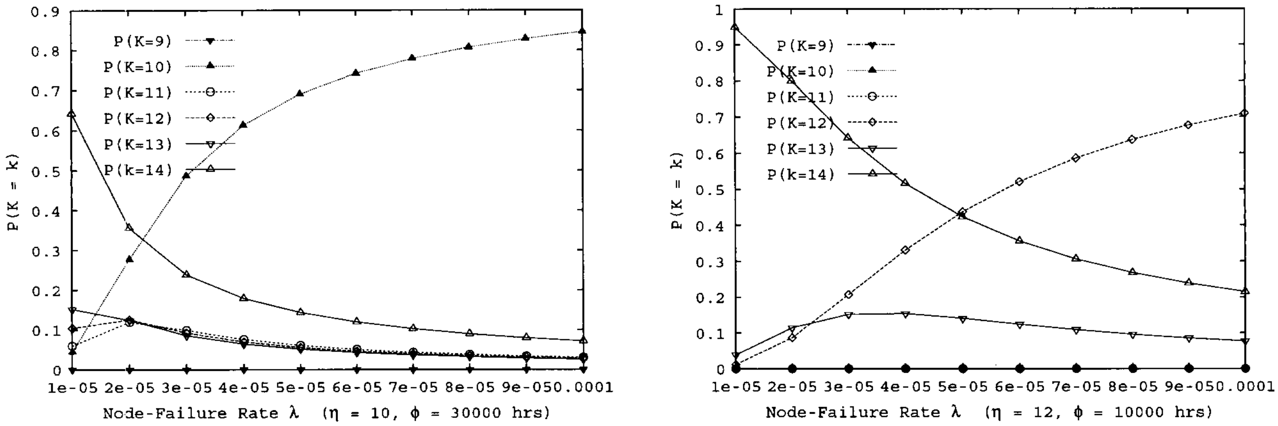
k	$P(Y = 3 \mid k)$	$P(Y = 2 \mid k)$	$P(Y = 1 \mid k)$	$P(Y \geq 1 \mid k)$
14	0.400000	0.000000	0.600000	1.000000
13	0.300000	0.000000	0.700000	1.000000
12	0.200000	0.000000	0.800000	1.000000
11	0.100000	0.000000	0.900000	1.000000
10	0.000000	0.000000	1.000000	1.000000
9	0.000000	0.000000	0.978694	0.978694

probability one the system will deliver a result rated QoS-level 1 or above. The meanwhile, Table 4 reveals that the value of $P(Y = 3 \mid 12)$ is only 0.20 with the BAQ scheme. Nonetheless, the values of $P(Y \geq 1 \mid k)$ shown in the two tables are identical, meaning that the difference between the two schemes is not at how well they guarantee the delivery of a result; rather, the OAQ scheme pushes a system's QoS to the high-end, guaranteeing a result with the best possible quality.

Next we evaluate the other constituent measure $P[k]$, using the software package *UltraSAN*. The results are shown in Figure 7, where we use the term "orbital-plane capacity" to refer to the number of (active) operational satellites that an orbital plane is equipped with. From the curves, we observe that when protected by the scheduled and threshold-triggered ground-spare deployment policies, the full orbital-plane capacity (i.e., $k = 14$) will dominate when node-failure λ rate is low. Whereas the threshold capacity (i.e., $k = \eta$) tends to become dominant as failure rate increases. Specifically, the values of $P[10]$ in Figure 7(a) and $P[12]$ in Figure 7(b) are both very small when $\lambda = 10^{-5}$, but they rapidly increase and become dominant as λ increases. This is because when satellites become more vulnerable to failure, the capacity of an orbital plane is likely to drop to the threshold sooner. Nonetheless, the threshold-triggered ground-spare deployment policy tends to prevent the scenario in which the plane's capacity drops below the threshold from happening. As a result, the likelihood that the system is operating at its threshold capacity becomes dominant when λ is high.

It is also interesting to note that the probability that a plane has a capacity that is close to the threshold will reach a maximum as λ increases (i.e., $P[11]$ and $P[12]$ in Figure 7(a) and $P[13]$ in Figure 7(b)). This phenomenon is also resulted from the ground-spare deployment policy. Specifically, as λ increases, the likelihood that a plane has a lower capacity tends to increases; however, the threshold-triggered ground-spare deployment policy will be exercised more often and thus make the threshold capacity become increasingly dominant. In turn, this reduces the probability that the orbital plane will be operated at a capacity other than the threshold capacity.

Based on the results of the constituent measures, we compute $P(Y = y)$ and $P(Y \geq y)$. Figure 8(a) compares the probabilities that OAQ and BAQ schemes will deliver a result rated QoS-level 3. For this evaluation experiment, we set η to 12 and let ϕ remain 30000 hours (over 3 years). The curves show that under the OAQ scheme, the system will achieve level-3 QoS with a greater

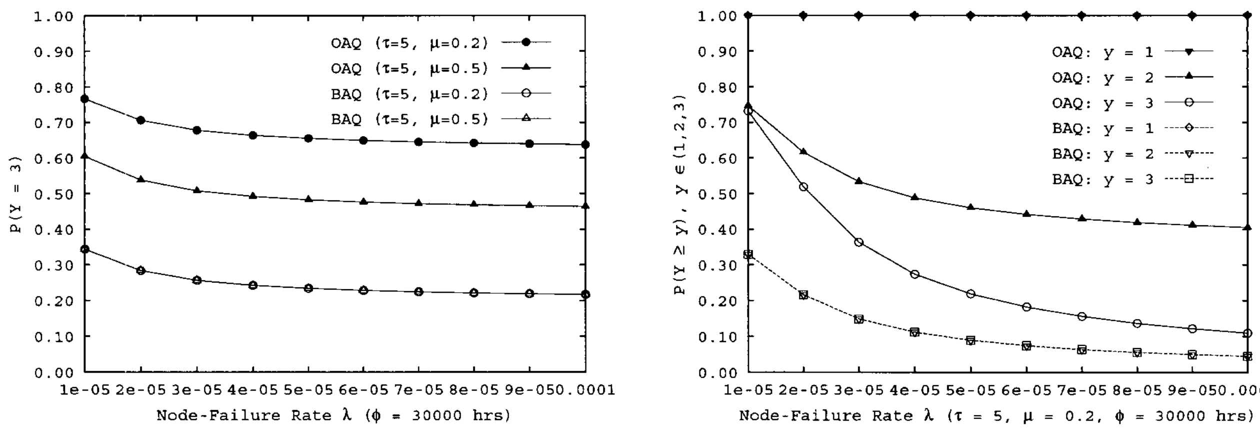


(a)

(b)

Figure 7: Probability of Orbital-Plane Capacity

probability when signal completion rate decreases (i.e., mean signal duration increases). More specifically, when μ decreases from 0.5 to 0.2, $P(Y = 3)$ increases up to 38% over the domain of λ considered. On the other hand, the same variation does not lead to any differences in the behavior of the BAQ scheme. This exemplifies that the QoS gain from the use of the OAQ scheme is due to its awareness and exploitation of conditions in the operational environment, while the BAQ scheme ignores potential opportunities.



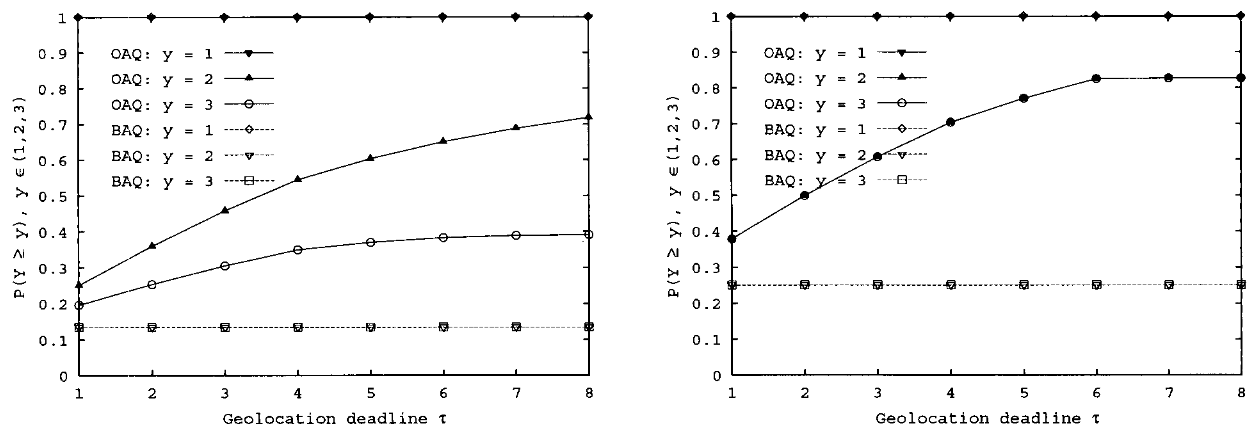
(a) $P(Y = 3)$

(b) $P(Y \geq y)$

Figure 8: QoS Gain from the use of the OAQ Algorithm

Figure 8(b) provides the results of the QoS measure $P(Y \geq y)$. The curves reveal that OAQ is always significantly more likely to achieve level-3 and level-2 QoS than BAQ. On the other hand, we can observe that the values of $P(Y \geq 1)$ are always equal for the two different schemes (both are equal to one always), meaning that OAQ and BAQ perform equally good with respect to guaranteeing the delivery of a result rated QoS-level 1 or above at any time. These results indeed confirm, from a different angel, that the merit of the OAQ scheme is that it pushes a system's performance to the high-end of a QoS spectrum.

Figures 9(a) and 9(b) illustrate the results of the QoS measure as a function of τ. While the data again confirm our earlier observation on the merit of the OAQ scheme, the curves exemplify that the OAQ scheme achieves better QoS by taking the full advantage of the "time allowance." In addition, by contrasting the results shown by 9(a) with those displayed in 9(b), we observe that although OAQ performs consistently superior over BAQ, BAQ receives a more significant impact from the variation of η on the QoS gain. This is indeed a desirable result, because it is obviously more costly to heavily rely on ground-spare deployment for QoS guarantee, relative to an algorithmic approach that takes advantage of opportunities arising during system operation.



(a) $P(Y \geq y)$ when $\eta = 10$

(b) $P(Y \geq y)$ when $\eta = 12$

Figure 9: Benefit from Opportunity Exploitation

5 Concluding Remarks

We have developed an approach to coordinated, progressive QoS optimization in satellite constellations which are vulnerable to structural degradation. This approach is characterized by the notion of "opportunity-adaptive," which implies that the scale of the coordination is dynamically determined by the readiness-to-serve of peer satellites, duration of the target signal and rate at which the coordination consumes the amount of allowed time. By letting peer satellites to successively join the coordinated geolocation computation as they become ready to serve within a window of opportunity, the OAQ algorithm is able to guarantee a timely delivery of the result with the best possible QoS. More specifically, as accuracy of iterative geolocation computation is generally an increasing function of the number of participating satellites, optimality in this framework means the best possible QoS with respect to a dynamically determined opportunity. Our model-based evaluation results demonstrate that as originally surmised, the OAQ algorithm is effective in pushing a system's performance to the high-end of a QoS spectrum even after a constellation loses a significant number of satellites. In addition, the analysis exemplifies that the opportunity-adaptive framework will amplify the benefit from redundancy-based fault tolerance strategies.

The results of this effort are meaningful for several reasons. First, the OAQ framework advocates the marriage between satellite constellation and fault tolerance technologies. In particular, although sequential multiple coverage was studied and proved to be sound mathematically in the research literature, the techniques have neither been discussed in the context of applications, nor been considered as a solution for tolerating the loss of satellites in a constellation. The OAQ framework demonstrates a novel, yet practical application of this satellite technology for fault tolerance in structurally degraded constellations. Moreover, we exploit inter-satellite crosslink communication as the enabling technology for realizing dynamic, progressive peer-satellite coordination that requires no intervention from ground stations.

Second, this effort exemplifies that while the continuously changing readiness-to-serve of satellites creates many challenges for fault tolerance, the mobile nature of their behavior can indeed be exploited to enable novel utility of resource redundancy. More generally, from the perspective of fault tolerance in systems comprising a large population of mobile resources, the results of this investigation demonstrate the feasibility of adaptation, extension, and generalization of various existing fault tolerance concepts, such as analytic redundancy, data diversity, environment diversity, imprecise computation, and active/passive replication.

Third, our analytic evaluation is based on a decomposition approach. Unlike traditional model decomposition that divides a model according to the behavioral or structural characterization of the system in question, our decomposition is done at the boundary between the system and the application, which results in two sets of constituent measures. While one set concerns the system-structure-level degradation and restoration, the other focuses on the application-oriented QoS that reflects the influence from the OAQ algorithm. The decomposition thus effectively reduces model complexity and enables a hybrid-composition approach to measure solution.

It is worth noting that when transient-fault-caused value errors in a large constellation are the major concern, we may allow individual satellites or small groups of them that make consecutive revisit to a target to perform computation independently, based on their own sensor data and potentially under different space environments; thus a majority-voting-based decision algorithm can be executed by the last member in the coordination chain (the effect of single-point-of-failure can be mitigated by a backward messaging scheme similar to that implemented in the OAQ algorithm) and produce a dependable final result, since data diversity and environment diversity make common-mode failure unlikely. As researchers in the wireless-networking area have been investigating advanced applications of TCP/IP and multicasting in satellite constellations, we believe that adaptation of those well-known fault tolerance schemes, including group membership protocol, will be feasible for application in micro- and pico-satellite constellations.

Accordingly, our current work is directed toward investigating the feasibility of adapting the concept of active and passive replications to satellite constellations, for implementing fault tolerance schemes that consider both inadvertent and malicious faults. Furthermore, as the development of the OAQ algorithm is aimed at the systems that are constituted by a large number of "light weight" mobile resources that are densely distributed, it is anticipated that the effectiveness of this framework will be scalable with respect to the size of a system. Hence, we plan to conduct additional case studies and use modeling and simulation techniques to validate this hypothesis and to guide algorithm extension.

References

[1] J.-C. Laprie, editor, *Dependability: Basic Concepts and Terminology*, vol. 5 of *Dependable Computing and Fault-Tolerant Systems.* Vienna, Austria: Springer-Verlag, 1992.

[2] M. Gerla *et al.*, "TCP/IP via satellite constellations," in *Proceedings of the 4th European Workshop on Mobile/Personal Satcoms (EMPS 2000)*, (London, UK), pp. 82–91, Sept. 2000.

[3] L. Wood, G. Pavlou, and B. Evans, "Managing diversity with handover to provide classes of service in satellite constellation networks," in *Proceedings of the AIAA International Communication Satellite Systems Conference*, (Toulouse, France), pp. 17–20, Apr. 2001.

[4] N. Levanon, "Quick position determination using 1 or 2 LEO satellites," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 34, pp. 736–754, July 1998.

[5] Y. T. Chan and J. J. Towers, "Sequential localization of a radiating source by Doppler-shifted frequency measurements," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 28, pp. 1084–1090, Oct. 1992.

[6] P. E. Ammann and J. C. Knight, "Data diversity: An approach to software fault tolerance," *IEEE Trans. Computers*, pp. 418–425, Apr. 1988.

[7] Y. Huang and C. Kintala, "Software implemented fault tolerance: Technologies and experience," in *Digest of the 23rd Annual International Symposium on Fault-Tolerant Computing*, (Toulouse, France), pp. 2–9, June 1993.

[8] J. Chase, D. Collins, T. George, E. Gralla, and R. Shotwell, "FAST Study: Candidate Satellite Constellations," Presentation, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, CA, 2002.

[9] J. W. S. Liu, K.-J. Lin, W.-K. Shih, A. C. Yu, J.-Y. Chung, and W. Zhao, "Algorithms for scheduling imprecise computation," *IEEE Computer*, vol. 24, pp. 58–68, May 1991.

[10] R. Raman, M. Livny, and M. Solomon, "Matchmaking: Distributed resource management for high throughput computing," in *Proceedings of the Seventh IEEE Symposium on High Performance Distributed Computing*, (Chicago, IL), pp. 140–146, July 1998.

[11] W. H. Sanders, W. D. Obal II, M. A. Qureshi, and F. K. Widjanarko, "The *UltraSAN* modeling environment," *Performance Evaluation*, vol. 24, no. 1, pp. 89–115, 1995.

[12] S. A. Nozaki and S. M. Ross, "Approximations in finite-capacity multi-server queues with Poisson arrivals," *Journal of Applied Probability*, vol. 15, pp. 826–834, 1978.

[13] L. Kleinrock, *Queueing Systems. Volume I: Theory.* New York: John Wiley & Sons, 1975.

A OAQ Algorithm

The OAQ algorithm is shown in Figure 10. The invocation of the algorithm is supported by a software geolocation agent GA which functions as the mediator between the onboard sensors and the middleware-implemented algorithm. In particular, the agent GA on satellite S_n always keeps updating the knowledge about the geometric orientation of the constellation, so that upon the invocation of the OAQ algorithm, S_n will be informed about with which geolocation mode (indicated by `GLmode`, a system parameter that is maintained by the GA) it should carry out computation. In addition, the GA constantly monitors the sensor readings to check whether the signal terminates and updates the system parameter accordingly.

Therefore, each satellite participating the coordination will execute this algorithm, upon arriving at the target location, using the data received from its predecessor and the system parameter values supplied by the GA as the input arguments. Further, upon completion of computation, satellite S_n ($n \geq 1$) will learn from the returned value `GLmode` (from `GLcomp()`), with which mode the next accuracy-improvement iteration shall be carried out (by peer S_{n+1}). Specifically, a zero value of this parameter means that one of the termination conditions, namely TC-1, TC-2, and TC-3, is met by the end of S_n's (successful) computation. Thus S_n will send its geolocation result and "coordination done" message to the ground and S_{n-1}, respectively.

If none of TC-1, TC-2, and TC-3 holds when S_n successfully completes computation, S_n will send a coordination request to S_{n+1}, meaning that the next iteration will be carried out through simultaneous or sequential coverage (indicated by a `GLmode` value 1 or 2, respectively). After making such a request, S_n will enter a waiting state and become active again when 1) a "coordination done" notification is received from S_{n+1} (identified by the GA via the function `nextpeer()`), or 2) the condition $getTime() - t_0 < \tau - (n-1)\delta$ no longer holds. For the former case, `OAQdone()` will return a value of one, so that S_n will forward the "coordination done" notification to S_{n-1}; for the latter case, `OAQdone()` will return a value of zero, so that S_n will assume that S_{n+1} is unable to finish the computation and will then send its geolocation result to the ground and notify S_{n-1}.

Finally, if S_n is unable to successfully finish its computation, `GLcomp()` will return a value of -1. The satellite will then simply withdraw from the coordination. Its silence will subsequently be

```
GeoLocation(n, t0, GLmode, GL_data) {
  GLmode = GLcomp(n, t0, GLmode, GL_data);
  switch (GLmode) {
    // no further improvement is required, or possible
    case 0: send(n, GLdata, groundStation, prePeer());
            break;
    // further improvement by sequential or simultaneous coverage
    case 1,2: reqCoordination(n, t0, nextPeer(), GLmode, GL_data);
              if (OAQdone(rcvMsg(nextPeer()), getTime()-t0 < τ - (n - 1)δ)) {
                // a peer in the upstream has delivered alert msg
                if (n > 1) {
                  send("Coordination done", prePeer());
                }
              } else { // peer in the upstream unsuccessful
                send(n, GLdata, groundStation, prePeer());
              }
              break;
    // unsuccessful geolocation computation
    case -1: return();
             break;
  }
}
```

Figure 10: OAQ Algorithm

detected by S_{n-1}, which will take responsibility to send the geolocation result and "coordination done" notification to the ground and S_{n-2}, respectively.