A Scalable Packet Routing Mechanism for Chip-Satellites in Coplanar Orbits

VAN HUNTER ADAMS
MASON PECK
Cornell University, Ithaca USA

This article is motivated by the advent of gram-scale spacecraft, or "chipsats," which enable satellite networks composed of hundreds of thousands of nodes. Networks of this size necessitate routing policies unlike any that have been used for collections of conventional spacecraft. This article argues which information should and should not be assumed available to each node in such a network. Based on these argued assumptions, this article uses dynamic programming to derive a routing mechanism for planar collections of chipsats. It, then, shows that the resulting mechanism is optimal for collections of orbits that are all near enough in altitude to communicate with one another, and also for collections of circular orbits. This article shows that the derived mechanism is suboptimal for collections of nested, unconnected orbits, and for stochastic collections of unconnected orbits. The particular form of the routing mechanism derived in this article is unique to planar collections of orbits, but the structure of the mechanism generalizes to three dimensions.

Manuscript received April 30, 2019; revised September 19, 2019; released for publication February 3, 2020. Date of publication March 2, 2020; date of current version October 9, 2020.

DOI. No. 10.1109/TAES.2020.2977791

Refereeing of this contribution was handled by T. Rossi.

Authors' Address: Van Hunter Adams and Mason Peck are with the Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY 14853 USA, E-mail: (vha3@cornell.edu). *(Corresponding author: Van Hunter Adams.)*

I. INTRODUCTION

To a greater extent than ever before, the past two decades has seen scalability become a core design consideration for a handful of engineered systems. Cell phones are the best example of such a system. Consumers purchased 1.56 billion smartphones in the year 2018 [1]. Each of these devices, in addition to being optimized for operation as an autonomous machine, was also optimized for cooperation with billions of other such devices. Manufacturing methods, communications protocols, and distribution strategies were all conceived with the understanding that billions of such devices would be built, distributed, and operate in a cooperative manner. This is what scalability means in the modern era, and it is the definition used in this article. A scalable system is one that works efficiently over and among an arbitrary number, perhaps billions, of iterations and constituents.

Defined as such, engineers do not yet optimize for scalability when designing spacecraft and spacecraft systems [2]. The volume of spacecraft built and launched each year is simply too low for scalability to be a reasonable criterion by which to judge potential spacecraft designs [3]. Instead, the relatively small volume of spacecraft (small relative to the volume of other complex engineered systems, like cell phones) generally favors bespoke solutions for each mission. This method has been radically successful, and has brought decades of commercial and scientific activity in space. Because this strategy works, there is no reason to expect for it to go away. There is, however, reason to expect that new strategies will become viable alternatives to this one.

Not long after people started launching spacecraft, they started launching spacecraft constellations. Many of the early satellite constellations facilitated communication. The first test of a space communications relay system was Pioneer 1 in 1958 [4]. In 1964, Syncom 3 and Relay 1 worked in concert to provide television broadcast in the United States of the summer Olympics in Tokyo, marking the first time that two spacecraft cooperated for that purpose [5]. Today, Globalstar, Inc maintains a constellation of 48 satellites for low-speed data communication [6], ORBCOMM has 31 satellites in orbit to facilitate communication [7], and Iridium has 66 [8]. A few other constellations for other purposes have also been launched, most notably the GPS constellation, composed of 24 spacecraft [9] and Planet Lab's Earth imaging constellation of over 140 spacecraft [10]. Even larger constellations are planned for the near future. SpaceXs Starlink constellation will be composed of nearly 12 000 spacecraft [11]. The trend toward scalable spacecraft (scalable in the spirit of cell phones) has been underway for a long time. We can expect for this trend to continue as the economics surrounding building and launching spacecraft continue to change.

The economics surrounding processors, radios, sensors, and printed circuit boards have changed particularly rapidly due largely to technology industries of scale, such as those surrounding cell phones and gaming [12]. The consequence

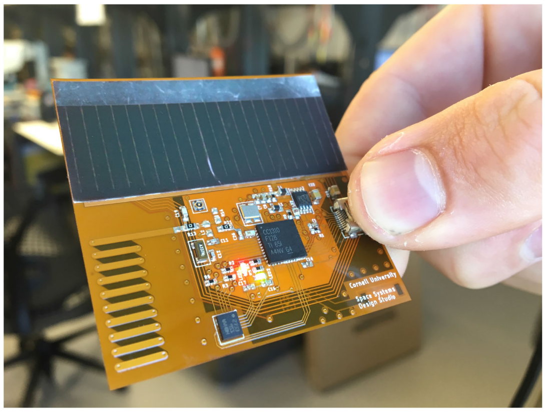

Fig. 1. Monarch chip-satellite.

is that spacecraft, albeit spacecraft of a very different variety than conventional spacecraft, can be manufactured for extraordinarily low cost. The Monarch, shown in Fig. 1, is the current extreme example of one of these low-cost satellites. It is a 5 × 5 cm spacecraft developed at Cornell University that can be manufactured for under \$100 and includes GPS, attitude determination and control, telemetry and command, and a suite of payload sensors [13].

The Monarch is the latest example of what has been termed a chipsat. These are gram-scale spacecraft built through the same automated processes that are used to construct printed circuit boards for cell phones, laptop computers, and other consumer electronics. The earliest work on spacecraft of this size was performed at the Aerospace Corporation in 1999 [14]. Much of the hardware development for these spacecraft has taken place at Cornell University. Justin Atchison characterized the differences in spaceflight mechanics for gram-scale spacecraft while at Cornell [15] and, since then, a series of hardware iterations and experiments have taken place. In 2011, two chipsats flew on the International Space Station. In 2014, Zac Manchester (then of Cornell, now a faculty member at Stanford) launched KickSat, a 3 U cubesat that carried 104 early-generation chipsats to orbit [16]. KickSat 2 (also by Manchester) launched in 2019 and carried 128. A chipsat on the Venta-1 mission verified communication from an orbiting chipsat to a ground station. The Monarch is the latest and most capable example of a chipsat.

These chipsats do not replace conventional spacecraft. They have their own new and unique set of use cases that relate mostly to missions involving massively distributed sensing and distributed monitoring. In distributed sensing missions, many thousands of chipsats are distributed over a vast area and each communicates local measurements through the collection and back to an operator. In distributed monitoring missions, the chipsats are again spread out over a vast area, but each reports detection of some stimulus (a solar event, activity of a spacecraft under surveillance, etc.). Distributed sensing requires high-bandwidth communication of information through the network, distributed monitoring requires comparatively low bandwidth. These constellations could be deployed in LEO for upper atmospheric or heliophysics studies, for spacecraft surveillance, or distributed monitoring for solar flares. Alternatively, they could be used in deep space for missions involving distributed plume sampling around Enceladus or planetary impact missions.

Chipsats solve the mission assurance problem in a new way. Rather than devote time, energy, and money to making certain that each spacecraft does not fail, chipsats solve the problem statistically. An excess are launched with the expectation that some will fail, but that a critical number will remain operational. This is precisely the same strategy that a sea turtle employs when it lays hundreds of eggs with the understanding (in an evolutionary sense) that only a small, critical number will survive to adulthood. Assurance is obtained through redundancy. This fundamental reliance on quantity for the utility of these chipsats means that they are the first spacecraft manufactured with scalability as one of their core design considerations. It also means that many of the standard, nonscalable methods for interacting with conventional spacecraft do not work on chipsats. An example, the example treated in this article, involves telemetry and command [13].

The conventional method for interacting with a spacecraft is to send it commands and receive telemetry when it travels within range of a ground station. Between passes, a conventional spacecraft will log data to onboard memory for later downlink [17]. There are rare examples of collections of satellites in which individual spacecraft that are out of range of a ground station can still communicate data to that ground station by sending them through an intermediate spacecraft. The Iridium constellation, with 66 satellites, is the best-known example of one of these systems. Even for a dynamic network of 66 spacecraft, the number of nodes is small enough for estimates of the positions of every node in the network to be continuously maintained. Consequently, routes through the Iridium constellation and other constellations of conventional spacecraft may be precomputed for each packet. Routing tables are continually updated as the topology changes [18], [19]. This method will not work for swarms of hundreds of thousands of chipsats, for which it is entirely intractable to maintain continuous estimates of individual node positions or to continually update routing tables.

SpaceXs Starlink will ultimately be composed of nearly 12 000 cooperative spacecraft. Each spacecraft is 227 kg, and the collection is carefully arranged into three orbital shells. Each spacecraft carries thrusters and actuators for attitude control for maintaining formation [20]. As a consequence, routing strategies through this network can exploit the determinism introduced by its careful arrangement and maintenance. Routing paths can be established open-loop, based on the known relative positions of all nodes [21]. The same method cannot be used for chipsats, which trade orbital control for size and expense.

Because of their exceptionally small size and mass, chipsats make spacecraft constellations of an unprecedented size eminently achievable. With the same payload mass as a single Iridium spacecraft, one could launch over 275 000 Monarch spacecraft [8]. It is entirely intractable to maintain continuous estimates of individual node positions or to continually update routing tables for dynamic networks of this size. Furthermore, Monarchs sacrifice much of the capability of large, conventional spacecraft in order to achieve their tiny form factor. A practical and scalable routing mechanism for communicating data through a network of chipsats must rest on realistic assumptions for the information available to each node in the network and the capabilities of each node in the network.

This article argues which information should and should not be assumed to be available to each node in a network composed of an arbitrary number of chipsats, and, then, derives the best achievable routing mechanism for a packet through an Earth-orbiting collection of chipsats under the argued assumptions. It does so by framing the problem as a series of optimal stopping problems and applying the dynamic programming equations. Routing data from any origin node in the network to any destination node in the network can be viewed as a series of decisions. The origin node will encounter a number of intermediate chipsats as it moves along its trajectory. At each of these encounters, it must decide whether the expected time to destination is minimized by relinquishing the data that it carries to the intermediate chipsat that it has encountered, or by retaining that data for itself. This decision is repeated for every encounter with an intermediate chipsat until the data reach their destination. Because the derived mechanism does not consider bottlenecking at ground stations, the scope of application is limited to distributed monitoring missions, wherein a collection of chipsats route low-bandwidth indication of a stimulus through the network to a receiver station.

This article treats only 2-D networks of chipsats (i.e., networks for which all chipsats occupy the same plane). It does so for three reasons. The first is that restraining oneself to two dimensions leads to the simplest form of a generalizable routing mechanism. The second reason is that this is not an unreasonable assumption for a collection of chipsats deployed from a common mothership in low-Earth orbit. At these altitudes, atmospheric drag is by far the dominant perturbing force on spacecraft of the Monarchs area to mass ratio. All chipsats deployed in low-Earth orbit will deorbit after 3–7 days, which is a long enough period of time for extensive dispersion in the direction of travel, but not a long enough time for extensive dispersion in other directions [15]. Finally, restricting oneself to coplanar orbits allows for the performance of the routing mechanism to be evaluated on an exhaustive collection of all possible configurations of orbits.

II. ASSUMPTIONS

A routing mechanism's efficiency through any network is limited by knowledge of the topology of that network. In the case that one knows the exact relative positions of all nodes in a network, then one can (in principle, if not in practice) solve for the fastest path from any node in that network to any other node, or for the set of all best paths [22]. This is the case for the Iridium constellation. The speed with which one arrives at the optimal path is obviously also of relevance for any practical application. This article is interested exclusively in practical routing policies of the sort that could be implemented on existing chipsat hardware in the immediate future, and that scale to networks of the size that chipsats enable. Deriving such a mechanism requires a realistic set of assumptions surrounding the capability of each node and the information available to each node. Each of these assumptions is stated and justified in this section.

A. Information

With respect to available information, it is assumed that each node knows its own position and velocity for all time, each node is able to measure absolute and elapsed time, and each node knows the angular rate of the Earth. The source for position, velocity, and absolute time information is the onboard GPS carried by each chipsat. The source for elapsed time is a timer in the onboard processor. The angular rate of the Earth is, of course, a parameter. This parameter is necessary for each chipsat to calculate its expected time to a ground station, which is co-rotating with the Earth, as explained in Section IV. It is assumed that the GPS does not enter a failure mode in which it reports erroneous positions and velocities, but is instead either totally functional or unresponsive. We also assume negligible drift in the onboard timer over the course of the chipsat's lifetime. The information that is available to each node does not require justification, given the suite of sensors with which each chipsat is equipped [13]. It is the information that is not available to each node (or to human operators or to ground-based equipment) that requires justification.

No node in the network, nor any ground-based equipment, is assumed to have knowledge of the number of nodes in the network. Chipsats are designed such that their high probability of failure (high relative to conventional spacecraft) is offset by the quantity that can be deployed at a single time. The consequence of this is that the number of functional nodes in the network will decrease from a known initial number to zero at a difficult-to-estimate rate. The efficiency of a routing mechanism through a network composed of nodes like these should not, for that reason, depend on knowledge of the number of nodes in the network. Furthermore, chipsats are so inexpensive to launch and deploy, a practical application will likely involve augmenting the network with fresh chipsats through subsequent launches. For this reason too, the routing mechanism should not depend on knowledge of the number of nodes that compose the network, and should instead have performance that is agnostic to this information. Section VI discusses some of the practical considerations associated with replenishing a chipsat swarm from additional motherships.

The topology of the network is also assumed unknown to any of the nodes. Spacecraft with area-to-mass ratios as high as those of Monarchs and other chipsats are extremely

TABLE I
Assumptions on Information Available to Each Node

Available	Not Available
Node's own position	Number of nodes in network
Node's own velocity	Topology of network
Time (absolute and elapsed)	Location of ground station
Angular rate of Earth	Position/velocity of any other nodes

susceptible to orbital perturbation in low-Earth orbit. Atmospheric drag is the most significant perturbing force, with solar pressure also contributing to alterations in trajectory. Both of these forces depend directly on the effective surface area of each chipsat, which, in turn, depends on the attitude of the chipsat [15]. For collections of thousands to hundreds of thousands of nodes, these perturbing forces will cause the topology of the network (i.e., which nodes communicate with which other nodes, and at which times) to change constantly and chaotically. The efficiency of the routing mechanism, therefore, should not rest on assumptions regarding the topology of the network.

The final assumption regarding information availability to each node is one that would not be strictly required but that is, in the estimation of the author, a good design decision for any practical application involving chipsats. No node is assumed to have information regarding the locations of ground stations. Unlike conventional spacecraft that will typically employ high-gain fixed antennas for high-bandwidth communication, chipsats instead communicate to the ground via handheld antennas that interface with a laptop computer [13], [16]. Rather than receiving high-bandwidth information from a single spacecraft, as is conventional, the model for chipsats is low-bandwidth communication from each of many nodes in a network. Considered as an aggregate, the data rates for a single conventional spacecraft and for a swarm of many chipsats are of the same order of magnitude, but the data rate from any particular chipsat is much lower. This enables handheld, portable receiver stations. Consequently, the performance of the routing mechanism should be agnostic to the location of the receiver station. This set of assumptions is summarized in Table I.

B. Capability

Each chipsat is equipped with a low power radio transceiver. They use these transceivers to communicate both to the ground stations and to one another. Substantial signal processing, which requires approximately the computational ability of a commercially available laptop, is required by the ground stations in order to receive these transmissions [16]. Each chipsat's processor has significantly less computational ability than a laptop, and, therefore, the transmission distances from chipsat to chipsat are significantly shorter than from chipsat to ground [23]. The consequence is that, in a collection of very many chipsats, the individual chipsats will drift into and out of communication distance with one another. As discussed in the previous section, the particular chipsats that pass into communicable range with one another will change unpredictably as the topology of the network evolves. When two chipsats are within communicable range, they may share information with one another. When they are out of communicable range, they cannot share information and they do not store any information about the trajectory of the node with which they had previous contact (since these trajectories change so quickly, and since this too does not scale as the number of nodes increases). It is furthermore assumed that each of the chipsats uses code division multiple access, which allows the ground station to differentiate signals from hundreds or thousands of chipsats.

III. PROBLEM STATEMENT

This article considers the situation in which an arbitrarily large number of chipsats are deployed from a common mothership in low-Earth orbit. The initial conditions imparted to each chipsat are randomized, with some being boosted to orbits with higher altitudes than the mothership and others landing on orbits lower than that of the mothership. This variation in eccentricity and semimajor axis, along with perturbations from atmospheric drag, solar pressure, and other higher-order effects, leads to rapid dispersion of the chipsats in the along-track direction and comparatively little dispersion in altitude, as shown in Fig. 2 [15]. The probability density function for the position of any particular chipsat is a function only of its eccentricity and semimajor axis (see appendices). After a sufficient amount of time has passed, the positions of all chipsats are well approximated by independent distributions [24]. The objective is to develop a mechanism that will select the path which minimizes the expected time that it takes for the packet to reach the ground station, under the assumptions justified in Section II. As explained in the introduction, this article only treats networks for which all chipsats occupy the same orbital plane (same inclination and longitude of ascending node).

The network under consideration is an example of an opportunistic network, where edge connections are unpredictable and the topology of the network is not known to any of the constituent nodes. A path from a particular node to the destination may change or break during the routing process. There are two general strategies for routing through these networks: flooding-based approaches and forwarding-based approaches [25]. In flooding-based approaches, such as epidemic routing, each node broadcasts the packet to every one of its neighbors until the packet reaches the destination. These approaches have the benefit of getting the data to its destination as quickly as possible, at the cost of bandwidth. Although strategies exist for reducing the overhead associated with epidemic routing [26], it still requires more resources than forwarding-based approaches.

Forwarding-based approaches select a single path to the destination, rather than letting all possible paths compete. In a forwarding-based approach, the node carrying the packet chooses, which neighbor to which to handoff the data (or

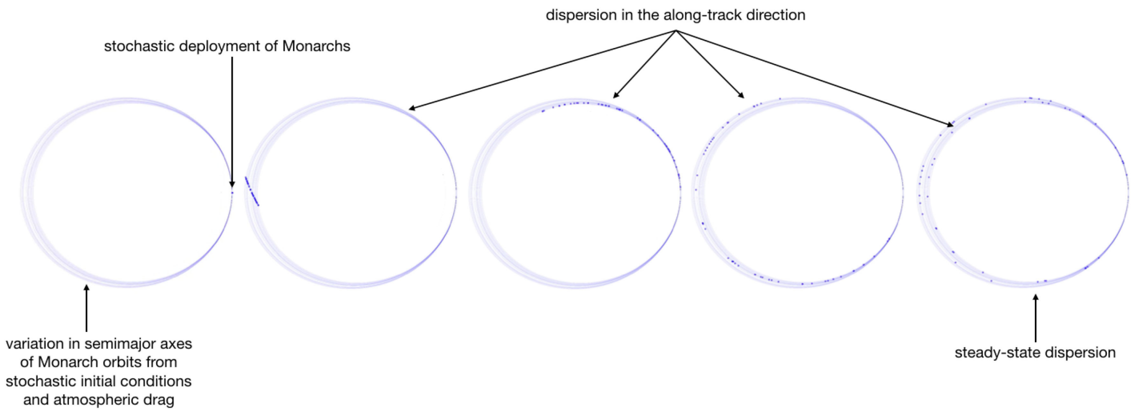

Fig. 2. Example of chipsat deployment and dispersion.

whether to retain the data) based on some piece of information. This information may be the proximity of each neighbor to the destination, or local knowledge of the network. This strategy has the advantage of increased bandwidth because more packets may be routed at once, since less nodes are occupied with a single packet than in an epidemic approach. The cost is potentially choosing a suboptimal path [25]. This article presents a forwarding-based approach through an opportunistic network of chipsats, where routing decisions are based on the instantaneous orbital mechanics of neighboring chipsats. This approach has the advantage of allowing more packets to be transmitted at once than would be allowed by an epidemic approach, and the disadvantage of placing packet delivery at risk. If the chipsat carrying the packet fails, then the packet fails to reach its destination. For many distributed monitoring applications, this is an appropriate tradeoff. The mechanism by which these decisions are made is derived through dynamic programming.

IV. DYNAMIC PROGRAMMING EQUATIONS

The dynamic programming technique is generally useful for solving problems that involve a series of decisions. The objective is to make each decision such that the cost (some quantifiable metric for the undesirableness of an outcome) over a given number of stages is minimized. Typically, each of these decisions involves a tradeoff between the immediate cost incurred at each particular stage and the anticipated future cost incurred as a consequence of each decision. A routing mechanism that optimizes only over each immediate step without consideration for future incurred costs is labeled a myopic mechanism. [22]

Problems exist for which the optimal mechanism is a myopic one (i.e., the cost over all stages is minimized by minimizing the cost at each stage). There exists another class of problems for which a myopic mechanism is suboptimal, but a lack of information about the number of stages in the problem or of the structure of the problem makes such a mechanism the only option. The on-orbit routing problem considered in this article falls into both categories, depending on the configuration of orbits. Each chipsat optimizes over each decision without consideration for future cost incurred as a consequence of that decision. It does so because no chipsat has enough system knowledge to estimate future incurred cost. It will be shown that, for some configurations of orbits, this myopic mechanism yields the optimal mechanism (the mechanism that generates a route with the shortest expected time to ground station). For other configurations of orbits this myopic mechanism is suboptimal, but it is the best that can be achieved under the necessary assumptions for grounding this problem in reality, as described in Section II. Solving a problem using the dynamic programming technique requires a state representation and state update equation, a representation for control input, representations for stopping and stage costs, and an optimal value function. With these defined, one can solve for the optimal control mechanism for minimizing cost over a number of stages.

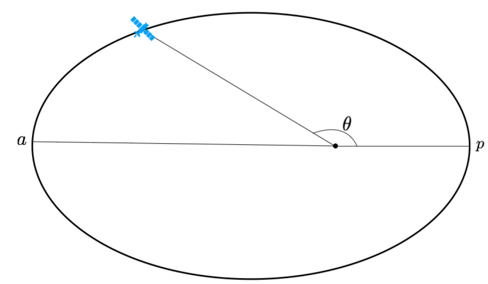

Fig. 3. Illustration of state variables.

The state of the system is specified by the instantaneous perigee altitude of the chipsat carrying data to be routed, the apogee altitude of the chipsat, the angular position of the chipsat measured from its perigee position, and an identifier for which chipsat is presently carrying the data, as shown in (1) and Fig. 3. All of these quantities can be found directly from the GPS data available to each chipsat. In most dynamic programming problems, the state is indexed by time, and the state update equations evolve each state variable from a timestep k to the next timestep k + 1. For this particular problem, the state is indexed by a quantity other than time. Instead, the state is indexed by swept Earth

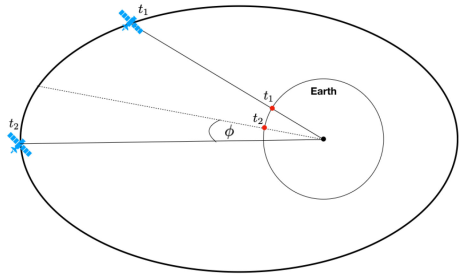

Fig. 4. Illustration indexing variable, ϕ.

angle, as shown in Fig. 4. As explained later in this section, each chipsat maintains an estimate of its own expected time to a ground station, the precise location of which is unknown to any chipsat. As each chipsat sweeps more of the Earth without discovering the ground station, it becomes increasingly confident that it will find it in the near future. In the case of the 2-D problem considered in this article, a chipsat can be completely confident that it will discover a ground station as it approaches a swept angle of 2π radians without having discovered it yet. These estimated times to ground station, which incorporate the orbital mechanics associated with each chipsat's apogee, perigee, and true anomaly, form the basis of routing decisions.

Each time the data-carrying chipsat encounters another chipsat, it updates its state as shown in (4) and (5). It does so by measuring its current apogee altitude, perigee altitude, and angular position from its onboard GPS, and scaling its angular distance traveled by the angular rate of the Earth to determine its updated swept Earth angle, which indexes the state. This update equation is shown in (5). Note that the chipsats angular position, θ, is a form of true anomaly that may exceed 2π radians. It is a measure of angular distance traveled since routing began. The control input to the system is very simple. At each encounter with an intermediate chipsat, the chipsat carrying data to be routed may take one of two actions. It may either relinquish its data to the encountered chipsat, or it may retain the data for itself, as shown in (3). It makes this decision on the basis of the stopping cost, terminal cost, and optimal value function.

State representation

$$x_{\phi_-} = \begin{cases} \gamma & \text{chipsat identifier} \\ p & \text{perigee altitude (km)} \\ a & \text{apogee altitude (km)} \\ \theta & \text{angular position (rad).} \end{cases} \tag{1}$$

Initial condition

$$x_{\phi_0} = \begin{bmatrix} \gamma_0 \\ p_0 \\ a_0 \\ \theta_0 \end{bmatrix}. \tag{2}$$

Control input

$$u_{\phi_-} \in \begin{cases} 1 & \text{Relinquish data} \\ 0 & \text{Retain data} \end{cases}. \tag{3}$$

State update equation

$$x_{\phi_+} = f(x_{\phi_-}, u_{\phi_-}) = \begin{bmatrix} \gamma_+ \longrightarrow \text{from routing decision} \\ p_+ \longrightarrow \text{from GPS} \\ a_+ \longrightarrow \text{from GPS} \\ \theta_+ \longrightarrow \text{from GPS} \end{bmatrix}. \tag{4}$$

Index update equation

$$\phi_+ = \left(\theta_+ - \theta_-\right) \frac{T_{\text{Earth}}}{T_{\text{Earth}} - T_{\text{Node}}}. \tag{5}$$

The stopping cost is the cost incurred if the chipsat carrying data relinquishes that data to the encountered chipsat. This cost is the optimal expected time to a ground station for the chipsat to which the data are relinquished. Note that, with global knowledge, this calculation would include a term that incorporates the probability of encountering another chipsat with a faster expected time to the ground station in the future. Under the assumptions required for making this routing mechanism a practical one, described in Section II, no chipsat has access to the global information required to arrive at these probabilities. As a consequence, the optimal expected time to the ground station (which would include information about probability of future encounters) is approximated by (8). Equation (8) gives the expected time to the ground station without considering the possibility of future encounters. Section V shows that this approximation still yields the optimal route for particular configurations of orbits, since it is only the relationship between the stopping cost and the optimal cost to go that is of consequence for decision making, and not the particular values of each, as shown in (12).

Note also that the stopping cost optimizes only over time, rather than jointly optimizing over energy and time. This is a consequence of the chipsats' architecture. The chipsats do not have any means of propulsion, nor do they have any power storage in the form of batteries. All of the electronics are powered directly from a 300-mW solar cell. The power availability from the solar cell significantly exceeds the power drawn from the electronics. Thus, when a chipsat is illuminated, it has a continuous supply of more-than-ample power. It is for this reason that the stopping cost optimizes over time, and not over energy and time. If the chipsats were storing energy and strategically meeting it out, or if they were using an expendable resource for propulsion, then such a joint optimization would be the prudent choice.

From the expression for the stopping cost, it can be seen that the terminal cost (the cost incurred at a swept angle of 2π radians) is the expected time from the initial swept angle, ϕ_0, if no handoff has occurred. If a handoff has occurred, then the terminal cost is 0. The optimal value function is shown in (11), which again involves an approximation of the optimal expected time to ground station of the same sort used for the stopping cost. This yields the simple thresholding policy shown in (12). At each encounter, the chipsat carrying data uses its onboard GPS unit to update its state and state index (swept angle). It, then, shares this state index with the encountered chipsat, and both calculate their expected time to the ground station by integrating their own probability density functions for position, as shown in (8). The myopic routing mechanism, then, simply chooses whichever chipsat has the shortest expected time to the ground station, and the process repeats until the ground station is encountered.

The routing mechanism is optimal if and only if the approximations for the expected time to ground station involved in the calculation of the optimal time to go, $V_{\phi_+}(x_{\phi_+})$, and the stopping cost, c_s, are such that the relationship between these approximations always yields a correct decision. Equation (12) makes it clear that it is the relationship between $V_{\phi_+}(x_{\phi_+})$ and c_s that is of consequence for decision making, and not the values themselves. For some configurations of orbits, this is the case. For others, it is not.

Stopping cost

$$c_s = \text{optimal expected time to destination for encountered chipsat.} \tag{6}$$

Stage cost

$$c_{\phi_-}(x_{\phi_-}, u_{\phi_-}) = \begin{cases} 0, & u_{\phi_-} = 0 \\ c_s, & u_{\phi_-} = 1. \end{cases} \tag{7}$$

Optimal expected time to ground station (see appendices)

$$E_{\phi_-}[t](x_{\phi_-}) \approx \frac{1}{2\pi - \phi_-} \int_{\phi_-}^{2\pi} \left[\frac{T_{\text{node}} T_{\text{Earth}}}{2\pi\, (T_{\text{Earth}} - T_{\text{node}})} \times \int_{\phi_-}^{y} \frac{(1-e)^{\frac{3}{2}}}{\left(\frac{1}{1+e}\right)^{\frac{3}{2}} \left(e \cos\left((\theta_- + \phi) \frac{T_{\text{Earth}}}{T_{\text{Earth}} - T_{\text{node}}} \right) + 1 \right)^2} d\phi \right] dy. \tag{8}$$

Terminal cost

$$c_{\phi=2\pi} = \begin{cases} E_{\phi_0}[t](x_{\phi_0}) & \gamma_{\phi=2\pi} = \gamma_0 \\ 0 & \gamma_{\phi=2\pi} \neq \gamma_0. \end{cases} \tag{9}$$

Terminal value function

$$V^*_{\phi=2\pi}(x_{\phi=2\pi}) = \begin{cases} E_{\phi_0}[t](x_{\phi_0}) & \gamma_{\phi=2\pi} = \gamma_0 \\ 0 & \gamma_{\phi=2\pi} \neq \gamma_0. \end{cases} \tag{10}$$

Optimal value function

$$\begin{aligned} V^*_{\phi_-} &= \min_{u \in \{0,1\}} E\left[c_{\phi_-}(x_{\phi_-}, u_{\phi_-} \in \{0,1\} + V^*_{\phi_+}(x_{\phi_+}) \right] \\ &= \min\left[c_s, V^*_{\phi_+}(x_{\phi_+}) \right] \\ &\approx \min\left[c_s, E_{\phi_+}[t](x_{\phi_+}) \right]. \end{aligned} \tag{11}$$

Optimal routing mechanism

$$g^*_{\phi_-} = \begin{cases} 1 & c_s < V_{\phi_+}(x_{\phi_+}) \\ 0 & c_s \geq V_{\phi_+}(x_{\phi_+}). \end{cases} \tag{12}$$

TABLE II
Summary of Optimality for Each Orbit Configuration

Orbit Configuration	Routing Mechanism Performance
Nested, circular, connected	Chooses optimal path
Nested, circular, disconnected	Chooses optimal path
Fully connected, non-circular	Chooses optimal expected path
Nested, elliptical, disconnected	May choose suboptimal expected path
Stochastic, disconnected	May choose suboptimal expected path

V. PERFORMANCE

The relationship between the optimal time to go, $V_{\phi_+}(x_{\phi_+})$, and the stopping cost, c_s, will be such that the relationship between these approximations will always yield a correct decision, as shown in (12), if and only if the following conditions are met.

1) After a routing decision among two chipsats, it is impossible that the chipsat that had a longer expected time to the ground station (and, therefore, relinquished the data) at the time of the routing decision will both a) later attain a shorter expected time to ground station than the other chipsat and b) overtake the other chipsat at a distance that exceeds the node-to-node communication distance before the entire Earth has been swept.
2) After a routing decision has been made, it is impossible that the chipsat, which relinquished the data will encounter another chipsat that both a) has a shorter expected time to ground station than the first chipsat to which the data were surrendered and b) will not come into communicable range with that chipsat before the entire Earth has been swept.

If these conditions are met, then the routing mechanism will choose the path which minimizes the expected time to the ground station. These conditions are examined for each of a series of orbital configurations. Table II summarizes the results.

A. Fully Connected Configurations

We consider configurations of orbits for which the maximum altitude separation for all nodes is within the node-to-node communication distance, as shown in Fig. 5. For this particular case, any two chipsats that are at the same angular position are capable of communicating. For such configurations of orbits, it is impossible for any chipsat to overtake any other chipsat without passing within a communicable distance. Any chipsat that overtakes another chipsat will be able to communicate with the chipsat that it is overtaking. Thus, there is no cost incurred by relinquishing data to a chipsat with a faster estimated time to ground station (even if the origin chipsat encounters an even-faster chipsat after handoff has occurred, or if the original chipsat later attains a faster expected time to ground station). The optimality conditions, therefore, hold and the derived mechanism chooses the path, which minimizes the expected time to the ground station for these configurations of orbits. Section V-B shows that, in the special case that all orbits are circular, the mechanism not only chooses the path,

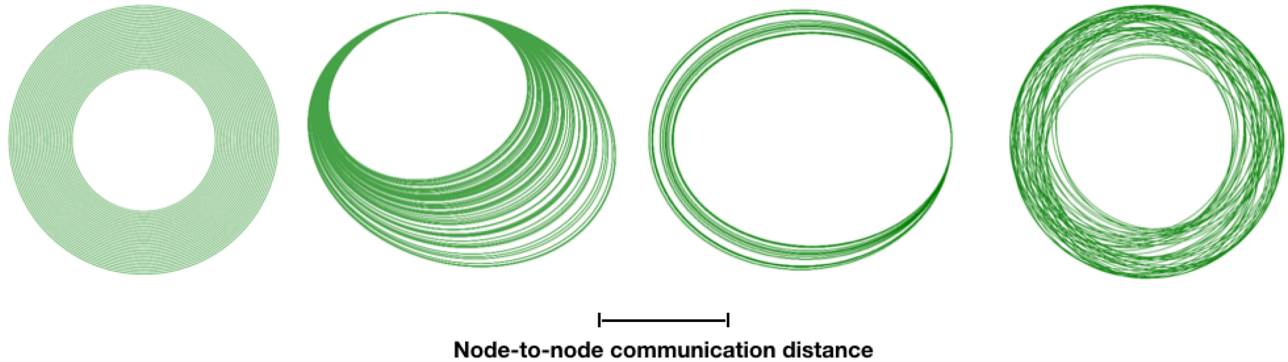

Fig. 5. Examples of fully connected orbit configurations.

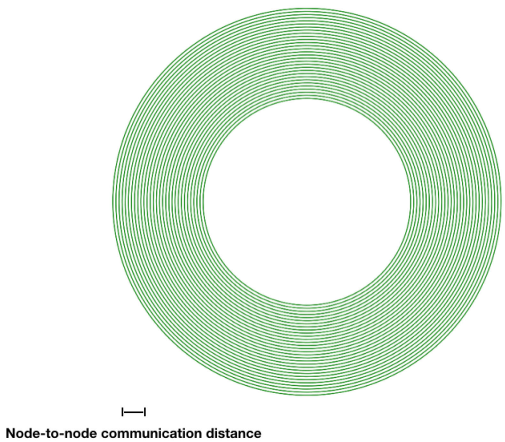

Fig. 6. Nested, unconnected configuration of circular orbits.

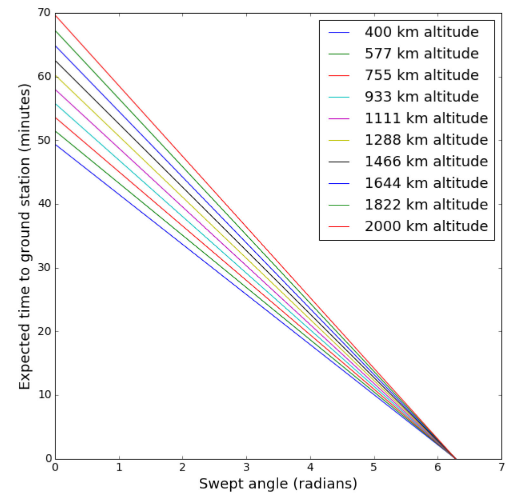

Fig. 7. Expected time to ground station for circular orbits of differing altitude.

which minimizes the expected time to the ground station, it chooses the optimal path, which minimizes absolute time to the ground station.

B. Nested, Circular Configurations

A second configuration of interest is one composed of concentric circular orbits with altitude separations that may or may not prevent complete connectedness, as shown in Fig. 6. Chipsats on circular orbits travel at a constant velocity and have zero eccentricity. The equation for the expected time to ground station, (8), reduces to the form shown below

$$E_{\phi_-}[t](x_{\phi_-}) = \frac{1}{2\pi - \phi_-} \int_{\phi_-}^{2\pi} \left[\frac{T_{\text{node}} T_{\text{Earth}}}{2\pi\,(T_{\text{Earth}} - T_{\text{node}})} \int_{\phi_-}^{y} d\phi \right] dy. \tag{13}$$

The consequence is that chipsats on lower altitude orbits will always have a shorter expected time to ground station, as shown in Fig. 7. Chipsats on lower orbits are always traveling more quickly than chipsats on higher orbits and, as a result, always have a shorter period, T_{node}. All optimality conditions, therefore, hold for this configuration of orbits. The myopic routing mechanism always chooses to pass data down in altitude, which always results in not just the optimal expected path to the ground station, but the time-optimal path. For both connected and disconnected configurations of circular orbits, the derived myopic routing mechanism chooses the time-optimal path to the ground station. Optimizing over each stage leads to a time-optimal path over all stages.

C. Nested, Unconnected Elliptical Configurations

The configuration of particular interest for practical applications is one composed of nested elliptical orbits that are separated by altitudes that exceed the node-to-node communication distance, as shown in Fig. 8. Each chipsat may communicate with other chipsats that occupy orbits of similar altitudes, but not those that occupy orbits of significantly higher or lower altitude. This configuration is of particular practical interest because it is the one that, to good approximation, chipsats deployed from a common mothership will achieve.

Because all chipsats are deployed from a common mothership, the collection of all attainable orbits by each deployed chipsat can be found by adding some amount of along-track velocity (in the forward or reverse direction) that is in the range of possible deployment velocities from

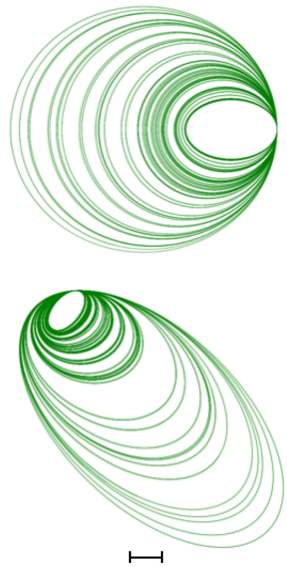

Fig. 8. Nested, unconnected configurations of elliptical orbits. (a) An unskewed collection of nested, unconnected elliptical orbits. (b) A skewed collection of nested, unconnected elliptical orbits.

the mothership. The important property of this collection of attainable orbits is that they do not cross over one another. All are nested inside of one another, intersecting at (at most) one location. If the mothership deploys the chips at its own perigee position, then all chips will have varying apogee altitudes (all at the same angular position) and an instantaneously identical perigee altitude and position, as shown in Fig. 8(a). If the mothership deploys the chips at its own apogee position, then all chips will have varying perigee altitudes (all at the same angular position) and an instantaneously identical apogee altitude and position. If the mothership deploys the chips between apogee and perigee, then all nodes will land on a family of orbits in which each constituent orbit has either higher or lower apogee and perigee than all other orbits, as shown in Fig. 8(b). None of these, however, crossover one another. All orbits are strictly losing energy and, as a consequence, their perigee and apogee altitudes are continuously decreasing. Orbits with lower average altitudes will lose energy more quickly than orbits with higher average altitudes because of the increased amount of atmospheric drag that they experience. The result is that the interior orbits in the initially nested configuration shrink away from higher altitude orbits more quickly than the higher altitude orbits approach them. The nested configuration is, therefore, maintained. A thought experiment can be used to show that the myopic mechanism is suboptimal (it does not necessarily choose the path, which minimizes the expected time to the ground station) on these configurations of orbits.

Consider chipsats A, B, and C that are orbiting on nested, elliptical orbits, as shown in Fig. 8(a). Chipsat A carries data to be routed to a ground station. It is overtaken by chipsat B, which has a shorter expected time to the ground station than chipsat A and, therefore, receives the data from A. Later, chipsat A is overtaken by chipsat C, which has a shorter expected time to the ground station than A or B. If it is possible for chipsat C to overtake chipsat B at a distance greater than the distance at which it overtook chipsat A, then the routing mechanism is suboptimal because a shorter expected time to ground station could have been achieved if A had waited to handoff to C. Optimality condition 2 would not hold. This is only possible if the rate at which the perpendicular separation between the orbits of B and C increases is greater than the rate at which the angular separation between B and C decreases.

A simple argument proves that this is the case for parts of the orbits of A, B, and C. Between perigee and apogee, the distance between orbits in the nested configuration shown in Fig. 8(a) monotonically increases. Between apogee and perigee, the distance between orbits monotonically decreases. Thus, the rate of change of the distance between orbits in the nested configuration is zero at apogee and zero at perigee, with a maximum rate of change at a location somewhere between apogee and perigee. The rate of angular separation behaves differently. For orbits in the configuration under consideration, the velocity of chipsats with higher apogee altitudes is greater at perigee than the speed of chipsats with lower apogee altitudes. At apogee, however, chipsats with lower apogee altitudes have greater velocity than those with higher apogee altitudes (see Fig. 9). Therefore, the rate of change of angular separation between chipsats is negative at perigee and positive at apogee. There must, then, be a location, somewhere between perigee and apogee, for which the rate of change of angular separation is zero.

This proves that there must be a range of locations for which the rate of change of separation between orbits exceeds the rate of change of angular separation between chipsats on those orbits. It is possible, therefore, for an incorrect routing decision to be made. A chipsat may handoff to another chipsat before encountering a third that has a shorter expected time to ground station. It has been shown that it is not always the case that this third chipsat will pass within communicable distance of the first, and, therefore, the routing mechanism does not always yield an optimal expected route. Optimality condition 2 does not hold for nested configurations of elliptical orbits like that shown in Fig. 8(a). Through a nearly identical argument, it can be shown optimality condition 2 does not hold for configurations like that shown in Fig. 8(b) either.

Chipsat deployment between apogee and perigee leads to the skewed configuration of orbits shown in Fig. 8(b), in which all orbits have higher or lower apogee and perigee than all other orbits. The consequence is that the rate of change of separation between orbits is increased for one half of the network and decreased on the other half of the network. Using precisely the same reasoning as was employed for the previous configuration, it can be shown that there must exist a range of locations in this network for which the rate of change of separation between orbits exceeds the rate of change of angular separation between chipsats on those orbits. Therefore, for this configuration also, it is

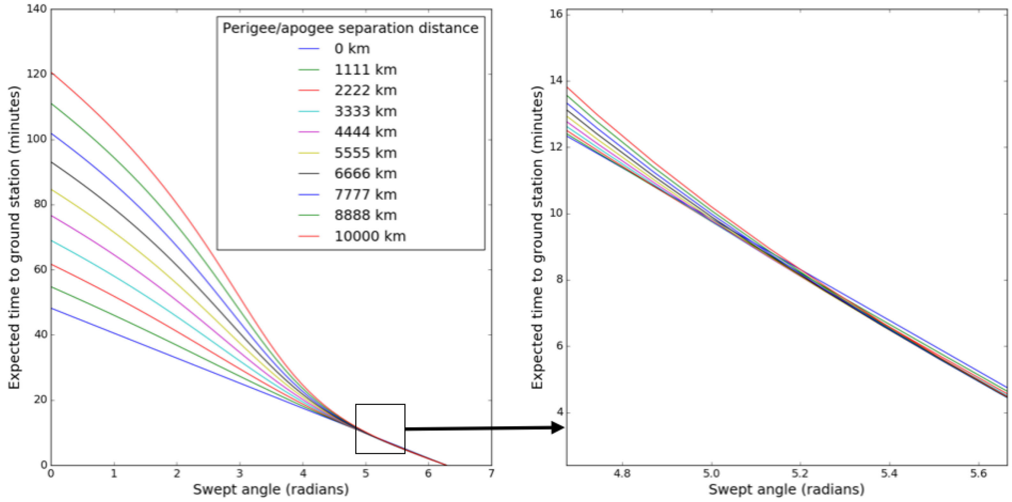

Fig. 9. Expected time to ground station for nested elliptical orbits, as in Fig. 8(a).

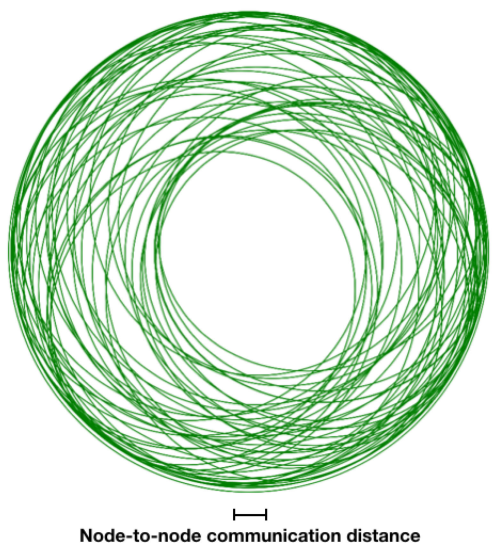

Fig. 10. Stochastic, unconnected configurations of orbits.

possible for incorrect routing decisions to be made and the myopic mechanism is suboptimal. It may choose a path which does not minimize the expected time to the ground station. Suboptimal, however, does not necessarily mean not worth doing. The expected time to the ground station is still reduced by performing the handoffs as prescribed by the routing mechanism. Suboptimality only means that a better handoff could have been made.

D. Stochastic, Unconnected Configurations

For a stochastic collection of unconnected orbits, like that shown in Fig. 10, no guarantees whatsoever may be made for optimality conditions 1 or 2. For this configuration, therefore, the myopic routing mechanism is suboptimal. As in the nested, unconnected configuration, the expected time to the ground station is still reduced by performing the handoffs as prescribed by the routing mechanism. Suboptimality only means that a better handoff could have been made.

VI. PRACTICAL CONSIDERATIONS

Implementation of the system described in this article will lead to a number of off-nominal situations that would need to be accommodated. These include situations in which a collection of chipsats is replenished from a secondary mothership, situations in which a packet misses a communication opportunity with the ground station, and noncatastrophic GPS failure modes. Each is considered in this section.

The only assumption on the collection of chipsats is that they all occupy the same orbital plane. As long as subsequent motherships all occupy the same orbital plane (which is not difficult, practically, to achieve), then the assumptions are not affected. A replenishment may, however, affect the configuration of the collection of orbits and consequently the optimality of the routing mechanism. If, for example, the original collection of chipsats occupied a fully connected nested configuration of orbits, then the routing mechanism would be routing optimally over that collection chipsats (see Section V-A). A replenishment may change the configuration from nested and fully connected to nested and unconnected, or stochastic. The routing mechanism still works over such collections of orbits but, as explained in the article, it is not optimal over such configurations.

It is also possible that a chipsat will fail to find a ground station during a single pass. The action taken in this case would depend on the mission and the importance of the packet being routed. For missions in which other chipsats can be expected to be routing similar information (a solar activity monitoring mission, for example), then the chipsat might retire the data after it has swept their entire search space (θ, Section IV). Alternatively, for missions requiring a greater guarantee on each packets delivery, the chipsat may make the conservative assumption that a technical problem has prevented it in particular from communicating, and it will surrender its packet to a neighboring chipsat.

Practical implementation will also require each chipsat to verify that its GPS measurements are reasonable, perhaps

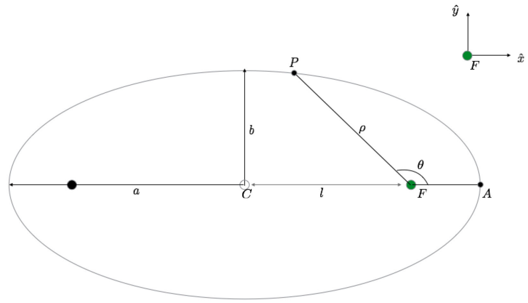

Fig. 11. Swept area geometry, origin at focus.

through comparison to a simple onboard dynamics model of its trajectory. The performance of the routing mechanism is sensitive to erroneous GPS measurements. In the rare event that a GPS modules failure mode leads to it reporting erroneous positions and velocities, the chipsat should be programmed to detect this error and refuse to receive packets.

The ultimate practical consideration is whether latency benefit from the routing mechanism is worth the implementation burden, as opposed to having each chipsat wait for direct downlink to a ground station. Chipsats enable swarms of hundreds of thousands of spacecraft. In swarms of such number, the connectivity of the network will remain intact until the chipsats begin to deorbit. The neighboring nodes will change but, if enough are launched, a path will remain from each node in the network to each of the other nodes in the network through intermediate nodes. Consequently, the speed with which a packet can be routed from any origin node in this network to the destination is limited only by the time that it takes to decide on a handoff, and the light-travel time of the packet. In this limiting case of very large constellations, the packet transmission time savings will be radically reduced by using the strategy described in the article as opposed to the nonstrategy of waiting for direct downlink.

APPENDIX

A. Swept Area From Angular Position

Consider an elliptical orbit with semimajor axis a and semiminor axis b, as shown in Fig. 11.

Earth sits at focal point F, with perigee at A. As the spacecraft traverses the orbit, it sweeps out the area AFP. The position of the spacecraft is specified by its distance from the Earth (ρ) and the angle from perigee (θ). Some of the geometric relationships among the above quantities are given by

$$e = \text{eccentricity} = \sqrt{1 - \frac{b^2}{a^2}} \tag{14}$$

$$l = \text{linear eccentricity} = ae \tag{15}$$

$$\rho(\theta) = \frac{a(1-e^2)}{1 + e\cos\theta}. \tag{16}$$

The location of the spacecraft (in Cartesian coordinates, with the origin on the Earth) can be parametrized as shown below

$$^{F}\mathbf{P} = \begin{bmatrix} \rho\cos\theta \\ \rho\sin\theta \end{bmatrix} \tag{17}$$

$$= \begin{bmatrix} \frac{a(1-e^2)}{1+e\cos\theta}\cos\theta \\ \frac{a(1-e^2)}{1+e\cos\theta}\sin\theta \end{bmatrix}. \tag{18}$$

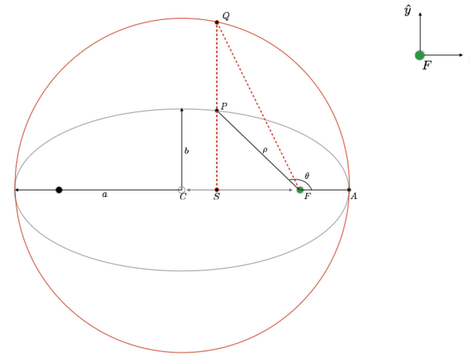

Fig. 12. Swept area geometry, scaled to circle.

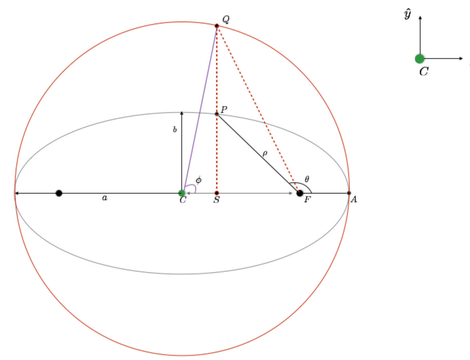

Fig. 13. Swept area geometry, scaled to circle, origin at center.

Consider scaling the y-coordinate such that the trajectory is a circle of radius a (the semimajor axis), as shown in Fig. 12.

This can be done by scaling the y-axis by $\frac{a}{b} = \frac{1}{\sqrt{1-e^2}}$. The point P gets mapped to Q, and the new position can be parametrized as shown below

$$^{F}\mathbf{Q} = \begin{bmatrix} \rho\cos\theta \\ \frac{1}{\sqrt{1-e^2}}\rho\sin\theta \end{bmatrix} \tag{19}$$

$$= \frac{a}{1+e\cos\theta}\begin{bmatrix} (1-e^2)\cos\theta \\ \sqrt{(1-e^2)}\sin\theta \end{bmatrix}. \tag{20}$$

Move the origin to the center of the circle, as shown in Fig. 13.

The position of Q is now given by

$$^{C}\mathbf{Q} = \begin{bmatrix} \rho\cos\theta + ae \\ \frac{1}{\sqrt{1-e^2}}\rho\sin\theta \end{bmatrix} \tag{21}$$

$$= \frac{a}{1+e\cos\theta}\left[\begin{matrix}(1-e^2)\cos\theta + e(1+e\cos\theta)\\ \sqrt{(1-e^2)}\sin\theta\end{matrix}\right]. \tag{22}$$

The angle ϕ is the eccentric anomaly. The geometry of Fig. 13 yields

$$\tan\phi = \frac{\sqrt{1-e^2}\sin\theta}{(1-e^2)\cos\theta + e + e^2\cos\theta}. \tag{23}$$

Using a Wiererstrauss substitution, this can be simplified to

$$\tan\frac{\phi}{2} = \sqrt{\frac{1-e}{1+e}}\tan\frac{\theta}{2}. \tag{24}$$

Solving (24) for ϕ yields

$$\phi = \operatorname{atan2}\left(\sqrt{1-e^2}\sin\theta, (1-e^2)\cos\theta + e + e^2\cos\theta\right) \tag{25}$$

$$= 2\tan^{-1}\left[\sqrt{\frac{1-e}{1+e}}\tan\frac{\theta}{2}\right]. \tag{26}$$

Solving (24) for θ yields

$$\theta = 2\tan^{-1}\left[\sqrt{\frac{1+e}{1-e}}\tan\frac{\phi}{2}\right]. \tag{27}$$

With this angle ϕ, one can calculate the area of sector ACQ. This sector includes the region of interest

$$A_{ACQ} = \frac{1}{2}a^2\phi. \tag{28}$$

To obtain the area of the region of interest, subtract and scale (28). To begin, subtract off the area of triangle FCQ, leaving the region AFQ remaining, as shown below

$$A_{AFQ} = A_{ACQ} - A_{FCQ} \tag{29}$$

$$= \frac{1}{2}a^2\phi - \frac{1}{2}a^2 e\sin\phi \tag{30}$$

$$= \frac{1}{2}a^2\left[\phi - e\sin\phi\right]. \tag{31}$$

This is still the area for a circle. To get back to the area for the ellipse, undo the initial scaling by multiplying by $\frac{b}{a}$ (since scaling in the y-direction scales the area by the same factor). Doing so yields

$$A_{AFP} = \frac{1}{2}ab\left[\phi - e\sin\phi\right]. \tag{32}$$

B. Probability Density From Swept Area

Because the spacecraft sweeps equal areas in equal times, the likelihood of finding the spacecraft in this angular region is given by

$$P(AFP) = \frac{A_{AFP}}{A_{\text{total}}} \tag{33}$$

$$= \frac{\frac{1}{2}ab\left[\phi - e\sin\phi\right]}{\pi ab}. \tag{34}$$

This leads directly to the probability distribution function in ϕ, shown in

$$P(\phi) = \frac{\phi - e\sin\phi}{2\pi}. \tag{35}$$

Substituting the expression for θ yields the probability distribution in θ, shown below

$$P(\theta) = \frac{2\tan^{-1}\left(\sqrt{\frac{1-e}{e+1}}\tan\left(\frac{\theta}{2}\right)\right) - e\sin\left(2\tan^{-1}\left(\sqrt{\frac{1-e}{e+1}}\tan\left(\frac{\theta}{2}\right)\right)\right)}{2\pi}. \tag{36}$$

The probability density function is obtained by taking the derivative of the distribution function with respect to ϕ (or θ in the case of $P(\theta)$). The result is shown in

$$p(\phi) = \frac{\partial P(\phi)}{\partial\phi} = \frac{1 - e\cos(\phi)}{2\pi} \tag{37}$$

$$p(\theta) = \frac{\partial P(\theta)}{\partial\theta} = \frac{(1-e)^{3/2}}{2\pi\left(\frac{1}{e+1}\right)^{3/2}(e\cos(\theta)+1)^2}. \tag{38}$$

ACKNOWLEDGMENT

The authors would like to thank Dr. M. von Gagern. His derivation of the swept area of an ellipse as measured from a focus, as posted on Stack Overflow, was tremendously helpful for finding the probability density function for the position of a spacecraft on an elliptical orbit, as shown in (38). The derivation of swept area from angular position in the first section of the appendix [concluding with (32)] is based on Dr. Martin von Gagern's posted derivation.

REFERENCES

[1] Gartner Number of smartphones sold to end users worldwide from 2007 to 2020 (in Million Units) Graph. In *Statista*, Aug. 1, 2019. [Online]. Available: https://www.statista.com/statistics/263437/global-smartphone-sales-to-end-users-since-2007/. Accessed: Mar. 7, 2020.

[2] W. J. Larson and J. R. Wertz *Space Mission Analysis and Design*. Torrance, CA, USA: Springer, 1992.

[3] G. Krebs Orbital Launches of 2019 *Gunters space page* Recuperado el 27.11 (2012): 2017. Accessed: Mar. 7, 2020.

[4] G. Marcus, Merritt Island, and US Air Force Major General Bernard Pioneering space part II *Quest*, vol. 14, no. 3, 2007, Art. no. 18.

[5] D. H. Martin *Communications Satellites*, 4th ed. El Segundo, CA, USA: The Aerospace Press, 2010, pp. 8–9.

[6] F. J. Dietrich, P. Metzen, and P. Monte The Globalstar cellular satellite system *IEEE Trans. Antennas Propag.*, vol. 46, no. 6, pp. 935–942, Jun. 1998.

[7] M. J. Evans and T. D. Maclay Mission design of the Orbcomm constellation *Mission Des. Implementation Satell. Constellations*, Dordrecht, pp. 103–112, 1998.

[8] K. Maine, C. Devieux, and P. Swan
Overview of IRIDIUM satellite network
in *Proc. WESCON*, Nov. 1995, p. 483.
[9] J. Foust
SpaceX's space-Internet woes: Despite technical glitches, the company plans to launch the first of nearly 12,000 satellites in 2019
IEEE Spectrum, vol. 56, no. 1, pp. 50–51, Jan. 2019.
[10] Planet Labs Website, planet.com, Accessed on: Sep. 18, 2019
[11] C. G. B. Green, P. D. Massatt, and N. W. Rhodus
The GPS 21 primary satellite constellation
Navigation, vol. 36, no. 1, pp. 9–24, 1989.
[12] R. R. Schaller
Moore's law: Past, present and future
IEEE Spectrum, vol. 34, no. 6, pp. 52–59, Jun. 1997.
[13] V. H. Adams and M. Peck
R-Selected spacecraft
J. Spacecraft Rockets, vol. 57, no. 1, pp. 90–98, 2020.
[14] H. Helvajian and S. W. Janson
Microengineering aerospace systems
Microengineering aerospace systems, vol. 29, 1999.
[15] J. A. Atchison and M. A. Peck
Length scaling in spacecraft dynamics
J. Guid., Control, Dyn., vol. 34, no. 1, pp. 231–246, 2011.
[16] Z. Manchester, M. Peck, and A. Filo
Kicksat: A crowd-funded mission to demonstrate the world's smallest spacecraft
in *Proc. Small Satell. Conf.*, 2013.
[17] E. Agasid *et al.*
Small spacecraft technology state of the art
Tech. Rep. NASA/TP-2015-216648/REV1, NASA, Ames Research Center, Moffett Field, CA, USA, Mission Design Division, 2015.
[18] R. S. Pratt *et al.*
An operational and performance overview of the IRIDIUM low earth orbit satellite system
IEEE Commun. Surveys, vol. 2, no. 2, pp. 2–10, Apr.–Jun. 1999.
[19] H. Keller and H. Salzwedel
Link strategy for the mobile satellite system iridium
in *Proc. Veh. Technol. Conf.*, 1996, vol. 2, pp. 1220–1224.
[20] Spacex Non-Geostationary Satellite System Attachment A, Technical Information to Supplement Schedule S, Nov. 15, 2016.
[21] M. Handley
Delay is not an option: Low latency routing in space
in *Proc. 17th ACM Workshop Hot Topics Netw.*, Nov. 2018, pp. 85–91.
[22] D. P. Bertsekas *et al.*
Dynamic Programming and Optimal Control. vol. 1. Belmont, MA, USA: Athena Sci., 1995.
[23] Instruments, Texas CC1310 simplelink ultra-low power sub-1 GHz wireless MCU
2016.
[24] V. H. Adams and M. A. Peck
A probabilistic network formulation for satellite swarm communications
in *Proc. AIAA Inf. Syst.-AIAA Infotech@ Aerospace*, 2018, p. 1802.
[25] C.-M. Huang, K.-C. Lan, and C.-Z. Tsai
A survey of opportunistic networks
in *Proc. 22nd Int. Conf. Adv. Inf. Netw. Appl.-Workshops*, 2008, pp. 1672–1677.
[26] T. Spyropoulos, K. Psounis, and C. S. Raghavendra
Spray and wait: An efficient routing scheme for intermittently connected mobile networks
in *Proc. ACM SIGCOMM Workshop Delay-Tolerant Netw.*, 2005, pp. 252–259.

V. Hunter Adams received the B.A. degree in physics from Cornell University, Ithaca, NY, USA in 2015, and the Ph.D. degree in aerospace engineering from Cornell University, in 2019.

He is currently a Lecturer in electrical engineering with Cornell University.

Mason A. Peck received the B.S. degree in aerospace engineering from the University of Texas at Austin, Austin, TX, USA, in 1994, and the M.S. and Ph.D. degrees from the University of California, Los Angeles, Los Angeles, CA, USA, in 1999 and 2001, respectively.

From 1993 to 1994, he was with Bell Helicopter on structural dynamics. From 1994 to 2001, he was an Attitude Dynamics Specialist and Systems Engineer with Hughes Space and Communications (now Boeing Satellite Systems). From 1998 to 2001, he was a Howard Hughes Fellow. During his years at Boeing, he served as an Attitude Dynamics Lead with the Boeing Mission Control Center, participating in real-time spacecraft operations and helping to resolve spacecraft performance anomalies. In 2001, he joined Honeywell Defense and Space Systems, where he became a Principal Fellow in 2003. He has several patents on his name. In July 2004, he joined as a Faculty with Cornell University, where he teaches courses in dynamics and control and in the mechanical and aerospace engineering program, where he was promoted to an Associate Professor in Fall 2010. In 2012, he was appointed as NASA's Chief Technologist.