Ant Routing Concepts for Dynamic Meshed Satellite Constellations

Matteo Berioli\textsuperscript{(1)}, Laura Rosati\textsuperscript{(1)}, Markus Werner\textsuperscript{(1)}, Gianluca Reali\textsuperscript{(2)}

\textsuperscript{(1)} Deutsches Zentrum für Luft- und Raumfahrt (DLR) Weßling, Germany, Email: \{matteo.berioli,laura.rosati,markus.werner\}@dlr.de

\textsuperscript{(2)} Dipartimento di Ingegneria Elettronica e dell’Informazione (DIEI) University of Perugia, Italy, Email: reali@diei.unipg.it

Abstract

We propose ant routing as a powerful means to enhance communications in meshed regular and irregular satellite constellations. Ant routing resembles basic mechanisms from distributed swarm intelligence in biological systems, and turns out to become an appealing solution when routing becomes a crucial problem in a complex network scenario, where traditional (deterministic) techniques either fail completely or at least face intractable complexity. The principle is to divide forwarding of information elements (packets) from the pure routing and to let ant routing operate to gather routing information, while other techniques, such as MPLS, might be used for forwarding.

INTRODUCTION

Network operators and service providers are facing an increasing demand for broadband access to Internet services. In order to meet this demand, LEO constellations are well suited to satisfy the service requirements of high capacity and low propagation delay. In particular a LEO constellation with Intersatellite Links (ISLs) can be considered as a dynamic network in the sky, which can provide global coverage. While the size of the network is typically limited to some tens of nodes, its topology is dynamic, since the satellites move with respect to earth. Moreover, such a network can be also an autonomous system, where it would be desirable to implement specific techniques for Quality of Service (QoS) and traffic engineering.

The introduction of IP traffic engineering techniques in such dynamic networks is quite a challenging task. A viable and elegant solution is to establish connection-oriented IP flows by means of Multiprotocol Label Switching (MPLS) \cite{1}, and to route flows according to some smart traffic-adaptive routing policies, as it was first proposed in \cite{2}. This is not trivial, considering particularly that the network experiences large traffic demand variations, due to the motion of satellites in their orbit planes and due to the rotation of the earth with its inhomogeneous geographic regions underneath.

This paper gives an overview on how to apply the concept of \textit{swarm intelligence} to satellite constellation networks. Swarm intelligence \cite{3} is a relatively novel discipline focused on self-organizing processes in nature, which take inspiration from the behavior of social insects or from the animals which live in colonies. In particular, some models of distributed organizations have been derived from observation of real ants; they are very successful at different combinatorial optimization and distributed control problems. In particular a series of new distributed routing algorithms (named \textit{ant routing} algorithms) have already been proposed \cite{4, 5, 6, 7, 8}. This approach emerges as a valid alternative to the classic deterministic approach adopted for traffic engineering for several reasons: simplicity, distributed operation, flexibility, robustness, and scalability.

In the remainder of the paper we first present the basic scenario, then we explain the basic ant routing principles and categorize the most prominent algorithms in this context. We further discuss the use of ant routing in ISL networks, starting with the motivation, then summarizing the key principles, and finally outlining applications and first implementations.

SCENARIO

Regular Satellite Constellations

Regular satellite constellation networks are considered in this work as networks composed of Low Earth Orbit (LEO) or Medium Earth Orbit (MEO) satellites, connected by means of ISLs. Normally such constellations
have 50 – 70 satellites (for LEO), and 10 – 15 satellites (for MEO). These satellites may be placed on different kinds of orbits according to the type of constellation. Constellations can be classified in two main groups: (i) polar or near-polar star pattern constellations and (ii) inclined delta pattern constellations [9].

In developing a QoS and networking concept for satellite constellation networks it is first necessary to recall the relevant features setting the framework, as well as the particular challenges.

Earlier studies on the routing in ISL networks of polar or near-polar star pattern constellations have clearly identified the seam between counter-rotating orbits and the on/off-switching of its inter-orbit ISLs as two fundamental drawbacks for the connection-oriented operation [10]. This has stimulated the investigation of moderately inclined delta pattern constellations employing ISLs [11]. It was shown that such constellations generally provide the possibility to set up a number of inter-orbit ISLs that can be maintained permanently. This link permanence is particularly important for optical ISLs, which seems to be the most promising candidate technology for the operation of future broadband LEO systems. Consequently, we consider inclined delta pattern constellations as the strongest candidates to host connection-oriented technologies like MPLS, which have a big potential for providing QoS. The main reasons are their regular mesh topologies and the lack of a seam between ascending and descending satellites [11]. Nevertheless, two types of topology dynamics remain as challenge: the time-varying link lengths of inter-orbit ISLs and the inherent handovers between serving satellites and ground users, both driving rerouting requirements.

Irregular Satellite Constellations

Recently, a new class of irregular, ad-hoc type “constellations” has been introduced mainly in the framework of scientific space exploration missions, where the space-borne nodes of the network typically consist of multiple spacecraft with multiple sensors [12, 13, 14]. In a similar way, one may think of a Delay Tolerant Network (DTN) set up in an ad-hoc fashion by a (varying) number of earth observation satellites in a “stochastic or uncontrolled constellation”. For a generalized view of such a scenario, we introduce the notion of an irregular satellite constellation. It is now instructive to look at such irregular constellation topologies mainly in comparison to the well-known regular ones.

From the protocol perspective, having deterministic ISLs means that there is no medium access contention at link layer. So in regular constellations very simple MAC protocols can be used and many of the problems related to MAC operation can be neglected. This in turn increases the relative importance of the routing protocol in this scenario: there are no packet losses due to access contention, but there may be packets lost because of queue overflow, when a congested link is selected by the routing protocol towards a particular destination.

On the contrary, an irregular constellation behaves more like an ad-hoc network, where there might be medium access contention and related outage periods; on top of that, of course, there is also the challenge of reliable and efficient routing under the regime of a non-deterministic and highly varying topology. That is why ad-hoc network concepts are important in the framework of this paper.

ANT ROUTING

Ant colonies are distributed biological systems that, in spite of the simplicity of their components, show highly structured social organization. As a result, ant colonies can accomplish astonishingly complex tasks that could never be performed by a single insect. The basic principle of an ant routing algorithm is that ants deposit on the ground a hormone, the pheromone, while they roam looking for food. Ants can also smell pheromone and tend to follow with higher probability those paths characterized by strong pheromone concentrations. The pheromone trails allow the ants to find their way to the food source (or back to the nest). The same pheromone trails can be used by other ants to find the location of the food sources discovered by their nestmates. It was demonstrated experimentally that this pheromone-trail-following behavior gives raise to the emergence of the shortest path.

An ant routing algorithm can be briefly described in the following way (cf. also Fig. 1):

- From each network node, a number of discovery packets (forward ants) are sent towards the selected destination nodes. They propagate concurrently and independently.
- In each node routing tables are replaced with stochastic tables, which select next hops according to weighted probabilities. These probabilities are calculated on the basis of the pheromone trails left by previous ants.
- While moving, the ants deposit pheromone on the crossed links, i.e., in the node routing tables they change the probability to select a particular next hop.
Two ants walk towards the food following two paths of different lengths. While walking, they deposit pheromone on the ground.

The ant following the shortest path is the first to reach the food and to walk back to the nest.

The ant following the shortest path is the first to come back to the nest. Now this path is marked by a higher amount of pheromone and for this reason it is more likely to be chosen by following ants.

![Figure 1: Basic principle of ant routing paradigm](image)

Ant routing algorithms can be classified in different ways, according to how the pheromone is updated, how routing table probabilities are calculated, how often and how many ants are sent per request, and so on. In Fig. 2 we present a possible classification. Using the schematic notation as introduced in the right column therein, the most representative ant routing algorithms to be found in the literature can be listed and categorized as follows:

- **AntNet** [4]: \{C4; I2/3; M2; P1/2\}
- **ARAMA** (Ant Routing Algorithm for Mobile Ad hoc Networks) [5]: \{C1; I2/3; M2; P3\}
- **ABC** (Ant-Based Control) [6]: \{C3; I2/3; M1; P3\}
- **RBA** (Routing By Ants) [7]: \{C1; I2/3; M2; P1\}
- **ASGA** (Ant System plus Genetic Algorithm) [8]: \{C1; I2/3; M2; P3\}
ANT ROUTING IN ISL NETWORKS

Motivation

Complexity of Deterministic Approaches

While MPLS has originally been introduced for a number of various purposes, it has revealed to be a valid solution for network management and, in particular, for traffic engineering. The appealing capability of MPLS is essentially its flexibility for deploying solutions which have been determined through optimization techniques. In fact, some simple basic label operations (push, pop, swap) and dynamic bandwidth allocation to LSPs give designers sufficient tools for managing the different sections of the network optimally, through hierarchical and adaptive solutions. Nevertheless, the derivation of the desired solution is not an easy task. In most cases the network management objective includes many aspects (e.g. fault tolerance, load balancing, security, congestion avoidance, performance guarantees, etc.). Thus, also the formulation of the traffic engineering problem is very difficult. In the recent literature some interesting proposals for terrestrial networks have already been presented (e.g. [15, 16]). Nevertheless, they only address particular situations, and unfortunately not the complexity of the general problem.

In addition, the environment analyzed in this paper is even more challenging, as in the scenario of satellite constellations with ISLs the network topology is time-variant and input traffic is highly dynamic. One should consider that one satellite may traverse in a short time densely populated regions and deserted ones or oceans. The traffic also varies depending on the time of the day. All this makes the task of optimally routing flows in a constellation very challenging. Already the simulation of such highly varying traffic is quite complex, as proven by a possible approach that was recently presented in [17].

The complexity of the problem suggests finding sub-optimal solutions by breaking it into sub-problems. In this paper we focus on the joint optimization of routing and QoS. Anyway, the simplification introduced is not enough to determine an optimum solution with a polynomially bounded convergence time. This is essentially due to the presence of QoS in the set of optimized quantities. Since in general the resulting metric includes the number of hops, the optimization can be done on an end-to-end basis only. Generally speaking, if a given path results to be optimum, this does not imply that a portion of it is still optimum for the portion of the metric associated with it. This implies the need of exploring all the paths from the source to the destination. As a result, the problem of optimally routing flows in a constellation is NP-complete, and consequently the complexity of the optimized routing algorithms is exponential in the number of satellites; therefore we have decided to resort to a heuristic approach in the present research work.

The Appeal of a Heuristic Solution

For regular dynamic satellite constellations, it has been convincingly argued that deterministic approaches are extremely complex, require a high level of sophistication and control, and need to include a lot of expert knowledge. For irregular dynamic constellations, the case for deterministic solutions is even worse: key components in their methodology may not work at all just because of unpredictable topology changes. This applies to irregular satellite constellations as introduced above as well as to traditional ad-hoc networks.

In contrast to that, a distributed heuristic solution like the ant routing displays several features making it particularly suitable for use in irregular topologies, or topologies which do not vary deterministically:

- the algorithm is fully distributed – there is no single point of failure;
- the operations to be performed in each node are very simple ⇒ simple on-board processing in the satellites;
- the algorithm is based on an asynchronous and autonomous interaction of agents ⇒ there is no need of synchronizing the different satellites;
- it is self-organizing, thus robust and fault tolerant ⇒ there is no need of defining path recovery algorithms;
- it is intrinsically traffic adaptive without the need for complex and yet inflexible metrics;
- it is scalable ⇒ the satellite network can be expanded with additional (also terrestrial) nodes without requiring complex update procedures;
- it is inherently adaptive to all kinds of long-term variations in topology and traffic demand (service and geographic user trends), which are difficult to be taken into account by deterministic approaches.

Key Concepts

Traffic engineering on a dynamic topology becomes especially complex if advanced features such as multiservice or class-based routing need to be taken into account. One thing that cannot be avoided is to calculate link
costs dynamically, depending on the traffic traversing the link (monitoring, e.g., present available bandwidth or queuing delay). In this way routing becomes traffic adaptive; every time a route is calculated the computation takes into account the present network status. This brings the problem that link costs have to be frequently refreshed and retransmitted to all nodes. So one critical point is how often these routing updates should or must be sent for proper operation. This issue entails different flavors depending on the two basic network operation modes:

- In the connectionless operation mode, routes are continuously computed by routing agents and routing tables are continuously updated. This bears the problem of instability (e.g., oscillations) or out-of-order packet arrival, since a route may change during one single flow of packets, even if it is not strictly necessary (old route is still available).
- In the connection-oriented operation mode, routes are computed only once at flow establishment and then maintained as long as possible. If a better route becomes available during the holding time of one flow it may not be used in case of a simple and strict approach. On the other hand, critical topology changes may enforce that flows must be handed over to new routes.

Furthermore, traffic oscillation in ISL networks is a particularly sensitive problem, since the variation of the links causes different routes to be selected and thus instability and constellation movement may force handovers between very different paths in the network. Solutions for this problem have already been explored, not only for connectionless IP traffic [18], but also for connection-oriented kind of traffic [19]. In any case, the signaling load and the accuracy of link-state information is strictly dependent on the rate of routing signaling distribution, and therefore the latter needs to be carefully tuned [20].

The conclusion is that the problem becomes intractable if all variables want to be considered, the increase of complexity leads to loss of flexibility and generality, since different metrics and parameters have to be calculated requiring complex processing. Some interesting solutions (e.g. [19]) also would need new signaling protocols, tailored to the particular scenario, which are far from being implemented.

We propose to adopt ant routing as a traditional routing algorithm, spreading link-state information and at the same time automatically allowing the computation of traffic-engineered routes without the need to care about sophisticated metrics and parameters. With ant routing, both forward and backward ants can be considered as routing signaling packets. The updates can be periodic or triggered by connection establishments or by significant changes in link state values.

In order to work, ant routing needs packets on the way back from the destination to the source – the backward ants. There are some ways to implement this paradigm without the need of an additional signaling protocol, but by means of commonly-used protocols which already need backward packets, or acknowledgments. TCP is the best example to do that. One possible way of implementing ant routing would be to exploit TCP acknowledgments to transport backward ants. In this way there would not be any specific routing protocol present, so no additional signaling load on the network, which is very appealing. One drawback could be that in case of low load of TCP traffic in the network, routing tables might not be up-to-date: Data packets might be forced to roam around the network a little bit longer before finding the way to their destination, or additional ants could be sent in the network, if routing tables are not fresh enough. This last proposal of TCP-piggy-backing for ant routing will not be considered in this paper and is left for future separate in-depth studies.

**APPLICATION**

We propose a systematic approach to application and implementation of ant routing in satellite constellations as illustrated in Fig. 3. The three inter-related scenarios are discussed step by step in the following sub-sections.

**Application to Regular Constellations**

In recent work it has been shown that the regular movement of the traditional constellations with ISLs allows to calculate various routing information in advance. But as soon as we move from static to adaptive routing, things become much more complicated. The problem can be divided in two sub-problems: selection of the satellites serving ground stations and selection of the route in the satellite segment (see Fig. 3). As already stated, an optimal solution for each of the two sub-problems does not necessarily yield the optimal solution of the overall problem.
The exploitation of MPLS still remains a very powerful approach in order to have control over single flows, but it can be considered just as a way of applying routing solutions that are determined by other mechanisms. So MPLS essentially allows routing to be separated from forwarding. So far different kind of algorithms have been proposed to elaborate routing information [19], but not many considerations have been made on what the required signaling effort is [2].

The scenario is well-studied and readily “available” (i.e., implemented) in various optimization tools and simulators featuring a wide bouquet of deterministic routing and traffic engineering approaches. So before moving to the even more challenging irregular constellations, ant routing may be applied to the regular ones in order to implement and study its basic features and performance as compared to the deterministic solutions.

**Application to Ad-Hoc Networks**

Before ant routing is used to gather routing information from the ISL network, we need to have a good understanding of how it works in the environment where it was already studied.

To this end, we designed and implemented a very simple ant routing approach for ad-hoc networks. Our solution is connectionless and allows nodes to respond to link failures and changes in network topology in a timely manner. It was designed integrating the ant-like behavior within the AODV (Ad Hoc On Demand Distance Vector) protocol [21, 22]. The main purpose of the work was to compare results with the well-known pure AODV in an ad-hoc scenario. The reason for this choice is that AODV is a reactive standard protocol, deeply analyzed and compared in literature with many other ad-hoc routing protocols. Both our ant routing approach and AODV are characterized by the following features:

- They enable dynamic, self-starting and multihop routing in ad-hoc networks.
- The state created in each node along the path is hop-by-hop state, meaning that each node does not know the whole path to the destination, but only the next hop node.
- Every node maintains a routing table with a routing entry for all possible destinations.
- Each node maintains a list of neighbors, i.e., a list of the node lying within the communication range. In order to update the topology, periodic “hello” packets are sent from each node to its neighbors. When a node receives a hello message from the node \( j \), it checks if \( j \) is in the list of its neighbors. If not, \( j \) is added to this list; otherwise its timer is updated. All timed-out neighbors are periodically purged.

Our approach to the problem is to keep “ant-like” agents extremely simple, since we want to investigate the basic behavior and achievable performance of ant routing algorithms at low implementation cost. In particular
we try to see whether it is possible with an ant-like approach to achieve a performance comparable with that of existing ad-hoc routing algorithms. If so, ant-like algorithms with their simplicity and the capability to adapt to severe environment changes would reveal to be very suitable for satellite constellations with ISLs, and in particular for irregular constellations. For this purpose we use the parameters of the model as tuning knobs, and performance is estimated by means of simulation.

At each node, the routing table has an entry for each possible destination in the network, which stores the quantity of pheromone associated with the links departing from that node. When a node receives a datagram with destination $d$, if the routing entry for $d$ is available, then the datagram is forwarded to the neighbor characterized by the maximum probability. Otherwise the datagram is buffered at the node and forward ants are sent in order to search a path to $d$. According to the principle of maximum algorithmic simplicity, we assume that ants can only deposit a constant amount of pheromone while moving and that they can only be influenced by the presence of the pheromone in the path selection. So the forward ants store only the identities of the crossed nodes in order to avoid cycles and to inform backward ants about the path they have to follow. As a backward ant travels, it updates the routing tables of each crossed node, as described in the following.

Being $i$ the current node, $k$ the node the ant comes from and $r$ a constant value, with $0 < r < 1$, the amount of pheromone $p(i,k)$ on the link $(i,k)$ is multiplied by $(1 - r)$ and then $r$ is added. The probabilities for the others links are multiplied by $(1 - r)$, so that the sum of all probabilities remains equal to 1. This simulates the deposit of a constant amount of pheromone.

In real ant colonies pheromone also evaporates; this process permits to choose new directions without being over-constrained by past decisions. This is also simulated by updating at regular time intervals the values of the pheromone on every link: If at instant $n$ the value of the pheromone on one particular link is $p_n$, evaporation is performed according to

$$p_{n+1} = (p_0 - p_n)k_{evap} + p_n$$

where $p_0$ is the initial value of pheromone on each link, which normally is $1/N$ if $N$ is the number of neighbors (at the beginning the probability to select a particular next hop is the same for all neighbors); $k_{evap}$ is a constant $< 1$. It can be easily demonstrated by induction that according to this evaporation formula, for every possible value of $p_n$, $p_{n+m}$ tends to $p_0$ if $m$ goes to infinity, that is selection probabilities become uniform if pheromone keeps evaporating without being updated:

$$p_{n+m} = k_{evap} p_0 \sum_{i=0}^{m-1} (1 - k_{evap})^i + p_n (1 - k_{evap})^m$$

$$\lim_{m \to +\infty} p_{n+m} = p_0 .$$

Even if the algorithm is very simple there are several parameters to be tuned, e.g. $r$, $k_{evap}$, $t_{evap}$, the latter being the time interval after which evaporation is performed. Proper tuning of these parameters becomes more difficult if the ant routing algorithm is not in its simplest version (according to the classification shown in Fig. 2) and if it involves several additional parameters and functions to be set. On the other hand, only such parameter tuning opens some freedom for a proper trade-off among the variables which serve as key performance measures for the protocol. For example, the higher the frequency of emission of the forward ants, the shorter is the time required for finding a path, and the higher the signaling load for routing updates on the network.

As an overall conclusion we state that ant routing is very powerful and flexible, but algorithm performance can be tuned and changed a lot by changing algorithm parameters.

**Application to Irregular Constellations**

In irregular constellations the main goal is not always to have high performance communications; this might be the case if the constellation is well meshed and the concentration of satellites in space is high. On the other hand, when the constellation is very sparse and when it presents long outage periods, it might also be that the main goal is simply to find a way across an irregular constellation at all and at some time. That is why it is likely that techniques like MPLS will not be always used, or only in some portions of the network.

The problem then becomes the following: the scenario can change within a short time between the two extremes, from a very dense topology to a very sparse one (or vice versa), or in some cases portions of the constellation might be very dense and others very sparse at the same time. Then we need a routing protocol that is able to operate between these two opposites, and ant routing seems very suitable for all the reasons outlined at various places in this paper.
SUMMARY

In this work we proposed ant routing as a powerful means to enhance communications in meshed regular and irregular satellite constellations. The solution applies when routing becomes a crucial problem (to optimize QoS and traffic engineering or to find a route across an irregular topology), that cannot be solved with traditional techniques, at least not without facing intractable complexity. The principle is to divide forwarding from routing and to let ant routing operate to gather routing information, while other techniques like MPLS might be used for forwarding.

REFERENCES


