Integrating Phased Array Path Planning with Intelligent Satellite Scheduling

Randy Jensen[1], Richard Stottler[2], David Breeden[3], Bart Presnell[4], and Kyle Mahan[5]
Stottler Henke Associates, Inc., San Mateo, CA 94404

As the demand for satellite-driven communication increases in both the commercial and military sectors, so do the numbers of active satellite constellations and the parallel requirements for ground-based support capacity. Phased array antennae have been identified as a cost-effective hardware solution for increasing communications capacity at ground stations, due to their ability to support multiple contacts simultaneously and their design compatibility with cost-effective commercial components. However, with increased communications capacity comes added complexity for the task of scheduling satellite supports in a network of satellites and ground stations with multi-beam phased array antennae. This task breaks down into two inter-related goals. First, the network-level challenge remains to allocate contacts to specific local sites. This is already complex in the case where ground stations exclusively use traditional mechanically steered reflector antennae, as schedulers seek to optimize resource usage within the confines of satellite visibilities and equipment availability at different sites. Second, with the introduction of phased array antennae, there is an additional local-level challenge to calculate active areas and paths on the surface of the phased array, to determine whether a candidate allocation with multiple contacts can actually be supported. These are inter-related because the local path planning analysis is predicated on an allocation developed at the network-level, whereas the network-level reasoning is most effective if it can be informed by knowledge of incompatibilities manifested at the local level. This paper describes an Artificial Intelligence based approach for handling these mutual dependencies efficiently while generating nearly optimal solutions.

Nomenclature

BA = Bottleneck Avoidance
GDPAA = Geodesic Dome Phased Array Antenna
LOS = Line of Sight
T/R = Transmit / Receive

I. Introduction

AS the demand for satellite-driven communication increases in both the commercial and military sectors, so do the numbers of active satellite constellations and the parallel requirements for ground-based support capacity. Scheduling and attempting to optimize satellite communication resources is an enormously complex task. There are many constraints to be considered. Obviously the satellite must have LOS (Line Of Sight) and be in-range of the antenna that it communicates with during the entire requested time window. Different stations have antennae with different capabilities, operate at different frequencies, have different data bandwidth, and have different support equipment so different satellites will have different sets of ground stations they can communicate with. Most satellites have requirements to have a certain number of communication events per day with maximum and minimum allowed time separations between events.

---

[1] Group Manager, 951 Mariner's Island Blvd., Suite 360, San Mateo, CA 94404, AIAA Contributor.
[2] President, 951 Mariner's Island Blvd., Suite 360, San Mateo, CA 94404, AIAA Member.
[3] Software Engineer, 951 Mariner's Island Blvd., Suite 360, San Mateo, CA 94404, AIAA Contributor.
[4] Software Engineer, 951 Mariner's Island Blvd., Suite 360, San Mateo, CA 94404, AIAA Contributor.
[5] Software Engineer, 951 Mariner's Island Blvd., Suite 360, San Mateo, CA 94404, AIAA Contributor.

Phased array antennae have been identified as a cost-effective hardware solution for increasing communications capacity at ground stations, due to their ability to support multiple contacts simultaneously and their design compatibility with cost-effective commercial components. However, with increased communications capacity comes added complexity for the task of scheduling satellite supports. This complexity arises from the sheer volume of supports and the unique constraints introduced by specific phased array implementations in handling multiple supports simultaneously.

The scheduling task breaks down into two inter-related goals. First, the network-level challenge remains to allocate contacts to specific local sites. This is already complex in the case where ground stations exclusively use traditional mechanically steered reflector antennae, as schedulers seek to optimize resource usage within the confines of satellite visibilities and equipment availability at different sites. Second, with the introduction of phased array antennae, there is an additional local-level challenge to calculate active areas and paths on the surface of the phased array, to determine whether a candidate allocation with multiple contacts can actually be supported. In contrast to a mechanically steered reflector antenna, a phased array uses an electronic scan area which may change size and location on the surface of the antenna during the duration of a support. Thus, in addition to the traditional satellite network-level scheduling problem of allocating a contact event to a ground station and specific antenna, a phased array presents the additional local-level problem of deconflicting active areas at any one time and in the paths they follow over time.

These two goals are inter-related because the local path planning analysis is predicated on an allocation developed at the network-level, whereas the network-level reasoning is most effective if it can be informed by knowledge of incompatibilities manifested at the local level. This paper describes an Artificial Intelligence based approach for handling these mutual dependencies efficiently while generating nearly optimal solutions. This combined network and local scheduling approach has the potential to not only reduce the manpower requirements for scheduling activity, but also to contribute to greater overall satellite communication capacity.

II. Background

In order to explore examples and discuss methodologies, we first introduce definitions for several terms.

Term	Description
Beam	An active LOS contact between a satellite and an antenna, with associated power requirements to accommodate the signal to or from the satellite.
Transmit/Receive (T/R) Module	An individual module on the surface of a phased array, with simultaneous capabilities to send and receive signals to and from a satellite.
Active Area	A bounded region on the surface of a phased array, with an associated shape and quantity of constituent T/R modules to support the communication requirements for a beam. Often the active area is a circular region mapped onto the surface of the phased array.
Optimal Position	The ideal placement of a beam active area on a phased array, based on the angle of visibility to the satellite. A beam's ideal placement when using a circular active area is centered on the optimal position.
Beam Path	A continuous route across the surface of the phased array, plotting the locations of a beam's active area throughout the duration of the contact.
Incompatibility	A condition involving two or more beams whose paths cannot be deconflicted.
Network-Level Allocation	Within a network of satellites and ground stations, network-level allocations are the assignments of beams to ground stations and individual antennae, informed by parameters of the requested support, visibilities, attributes and resource requirements associated with the satellite.
Local-Level Allocation	Local-level allocations apply only to phased array antennae with multi-beam capacity, and involve the calculations of beam paths and assignments to communications ports.

A. Incompatibility and Path Deconfliction

Figure 1 shows an example of two incompatible beams on a Geodesic Dome Phased Array Antenna (GDPAA). In this example, two beams with large active areas overlap when placed at their optimal positions.

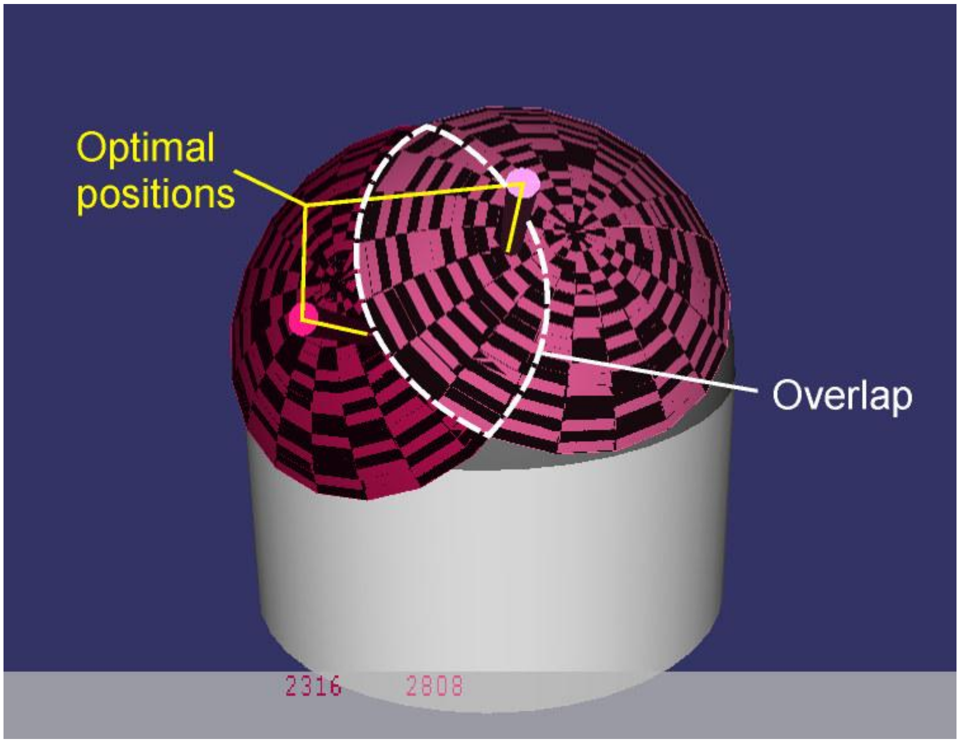

Figure 1. Example of beam incompatibility.

In this condition, an attempt at deconfliction may be made by altering beam paths by shifting their active areas away from the optimal positions. When active areas are offset from optimal positions, this typically requires an increase in the active area size to include additional T/R modules to provide additional signal power as necessitated by the non-optimal orientation. There are also other constraints on how far the active areas may be shifted.

As shown in Fig. 1, both beams have been shifted away from their optimal positions, but cannot be moved far enough to remedy the existing overlap, due to the other constraints on positioning. If a solution cannot be found for deconflicting beam paths, then their overlap is inescapable and they are deemed incompatible.

B. Port Allocation

One Geodesic Dome Phased Array Antenna (GDPAA) design allows up to 4 simultaneous full duplex (both send and receive) contacts. However it cannot be simply treated as 4 separate antennae. First, each transmit/receive (T/R) module on the surface of the array only supports 1 transmit beam, which means that no incompatibilities can be supported for any plurality of simultaneous transmit beams on the antenna. Second, each T/R module supports a maximum of 2 receive beams. This implies that 2 simultaneous incompatible beams can be supported, but no more than 2. In the receive case, there is the additional complexity of allocating the beams to communication ports depending on their mutual incompatibilities, to try to find a valid set of assignments. In order to support 2 simultaneous incompatible beams, they must be allocated to different communication ports.

Table 1. Example port allocation.

Beam	Incompatible with…	Port 1	Port 2
A	B, C	●	
B	A		●
C	A		●
D	-	●	

Table 1 shows an example port allocation for 4 simultaneous beams under this design, dictated by the constraints of pairwise incompatibilities among the beams. Beam A is incompatible with beams B and C, which implies that those must be allocated to a different port from A. If B and C were incompatible with each other, they could not both be supported on the GDPAA at the same time as A, because of the existing requirement that they be assigned to the same port. Consequently, a fourth beam D can only be supported if it is compatible with A and can be allocated to the same port as A. Therefore, the allocation shown is the only successful solution for this set of beams. Note that the choice of port 1 versus port 2 is arbitrary, as the two are interchangeable. So a mirror image of this allocation would also succeed, with the assignments to port 1 and port 2 reversed.

C. Network-Level Scheduling and Allocation

The objective for the phased array path planning and allocation approach is to provide a mechanism by which an external scheduling system can perform analyses and create assignments to phased array antennae, in a manner similar to how this is done with single-beam antennae. For example, a sample use case may involve a network of ground stations, where some have multi-beam GDPAAs and others have traditional single-beam reflector antennae. Consistent with this objective, the phased array planning logic has been implemented in an integrated configuration with an intelligent scheduling engine which has been modified to make specialized function calls for assignments to phased arrays. In the network-level scheduling process, beams that are assigned to traditional antennae are treated exactly the same as they were before, and beams assigned to a phased array antenna go through several extra steps associated with the local-level allocations.

The network-level scheduling component is not the subject of this paper, but a brief description is helpful to establish the context for how beams are handled, and when the local-level allocations must be developed. The interaction between network-level scheduling and local-level allocation principally occurs during a scheduling process called Bottleneck Avoidance (BA), an artificial intelligence technique that is well-suited to satellite scheduling as it mimics the processes of human schedulers.[1,2] In the BA methodology, beam requests are organized into a precedence scheme based on resource contention, where the most heavily requested or loaded resources are treated as having the highest contention (or "bottlenecks"). BA then iterates through beam requests in order of contention, starting with the worst bottlenecks, and schedules them with the goal of reducing bottlenecks. In order to interoperate with this sequential process, local-level allocation logic must provide functions that can be consulted one by one with individual beams during network-level BA scheduling. Table 2 roughly describes how the framework operates.

Table 2. Network-level BA scheduling process. *Given a set of satellite communication requests (beams), the following algorithm schedules beams to antennae.*

1. Organize beam requests and usage profiles	Generate resource usage profiles for each antenna at each ground station in the network. Usage profiles are computed by distributing each beam's usage across the possible resources it can use. So, if there is only one antenna at one station that can support a beam request at a designated time, then the beam contributes a usage of 1.0 to that antenna. If the beam could be scheduled to any of 4 different antennae, it contributes a usage of 0.25 to each.
2. Iteratively schedule bottlenecks	i. Pick the antenna with the worst bottleneck – the highest peak across all antennae. ii. Pick the beam contributing the most to the bottleneck. iii. Of the possible allocations for the beam, schedule where there is the least usage. iv. Update the usage profiles.

III. Integrated Approach

In order to conform to the sequential process of network-level scheduling, the local-level planning and allocation is divided into 2 passes. The **initial pass** is a recurring procedure performed with each beam to be scheduled on a phased array. This generates approximated predictions for incompatibilities between beams that may be allocated to the same phased array, and also attempts to find viable port allocations for both the transmit and receive operations, based on these predicted incompatibilities. The **global pass** is a one-time final verification procedure performed at the conclusion of the network-level allocation process, when an initial schedule has been developed. The task of calculating final beam paths for all beams allocated to a phased array is performed at this stage.

The division of the algorithm into 2 passes is motivated by the problem of mutual dependencies between network-level allocation and local-level incompatibilities and path planning constraints. For example, consider a design where the global pass is the only instance when network-level scheduling checks for phased array incompatibilities. In the likely outcome where initial multi-beam allocations made at the network-level cannot be supported in practice on the antennae, this would require considerable backtracking, and again with incomplete knowledge of the incompatibilities that would impact the likelihood of success or failure in an alternative allocation. Similarly, consider a design where the initial pass includes a complete evaluation of incompatibilities among all beam requests, without any initial input from the network-level scheduler to reduce the space of possible subsets to process. This would be highly inefficient due to all the unnecessary analysis of incompatibilities between beam paths that would not be allocated together anyway. Thus the compromise is to introduce an initial pass analysis that can be used with each beam during the network-level allocations.

The following sections describe elements of the initial pass and global pass operations in more detail.

A. Initial Pass Incompatibility Predictions

Because of the frequency of calls to the initial pass procedure, there is a strong need to optimize its computational efficiency. This motivates an approach where the initial pass attempts to predict incompatibilities through an approximation of the beam path planning procedure, rather than calculating complete deconflicted beam paths for each candidate. Table 3 steps through a high level description of the incompatibility prediction procedure.

Table 3. Initial pass incompatibility analysis. *Given a beam* B_c *under consideration, an antenna, and the n beams (possible zero) previously scheduled on the antenna whose durations intersect with the specified duration for* B_c *, the initial pass performs roughly the following steps to predict incompatibilities.*

1. Calculate the optimal path for beam B_c	The optimal path is the continuous path of optimal positions for the B_c active area on the phased array as it traverses the surface, during the time interval of the requested support.
2. Create temporary port groupings	For all n beams allocated with active support times on the phased array during the duration of the B_c requested support, a complete set of possible port groupings is created, in all permutations. There are separate calculations for the transmit case and the receive case, because these have different port allocation constraints. In the transmit case, there can only be 1 port grouping at any instant in time, with up to 4 beams in an allocation. In the receive case, there can be up to 3 port groupings at any instant in time, where groupings contain up to 2 port allocations each with up to 2 beams.
3. Find closest proximities within port allocations	Each port grouping contains 1-2 port allocations. In order to determine if these allocations are viable, it is necessary to determine if incompatibilities exist within an allocation. This is determined by an approximation that looks for the times when any pair of beams' optimal positions come within closest proximity of each other. These times are roughly considered the "worst cases" for deconfliction.
4. Attempt to deconflict beams within port allocations at closest proximities	For each pair of beams in a shared port allocation, if their active areas overlap at the times of closest proximity, then deconfliction is attempted. This consists of small incremental shifts in the positions of the active areas until either the overlap is eliminated, a constraint is violated (such as moving too far off the optimal position), or a threshold is reached on the allowed number of iterations. The results directly informs incompatibility predictions. If no port grouping can be successfully deconflicted, then B_c is deemed to be incompatible with the beams for which deconfliction failed.

This analysis is an approximation because it only considers the points of closest proximity between beams, in any of the possible port groupings. For obvious reasons, it is much less computationally intensive to attempt deconfliction on a collection of single points for beam pairs, as opposed to multiple samples along the optimal paths. Since this considers the worst cases of closest proximities, it is still also likely to identify incompatibilities in most cases where they exist. As the initial pass is repeated with successive beams, it maintains persistent records of incompatibilities identified earlier, to short-circuit the logic for subsequent calls examining the same set of beams.

B. Initial Pass Port Allocation

For each beam processed in the initial pass, it is also necessary to determine if it is possible to find a viable port allocation for the beam, given any predicted incompatibilities with other simultaneous beams on the phased array. A natural assumption may be that for an initial allocation, it is only necessary to find a solution for the beam itself, and the port it will be assigned to during its duration.

However, the problem is more complex than that. When the scheduling task involves an extended period such as a 24 hour interval of communications between ground stations and satellites, the port allocation task requires an analysis of longer overlapping sequences of beams. A long string of beams, often as many as 50, may overlap partly in time and therefore affect each other's assignments. It would be possible to assemble a collection of port allocations such that each allocation is successful within the narrow timeframe of an individual beam, but the collective set of allocations cannot succeed in the complete sequence over a longer period. The following example illustrates this condition. Figure 2 shows a sequence of 6 beams A-F, and their associated time intervals.

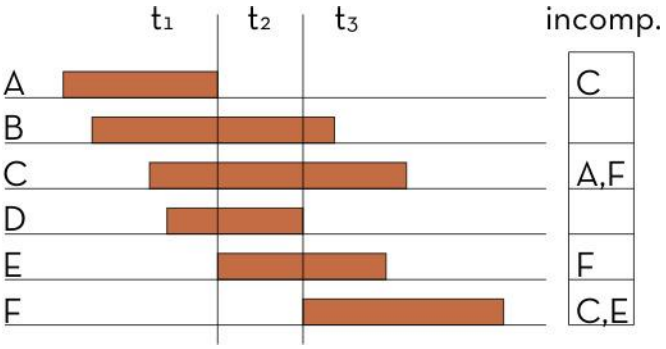

Figure 2. Sequence of 6 beams with overlapping time windows. *Vertical lines delineate time intervals t_1, t_2, and t_3, which represent 3 distinct periods during which different combinations of 4 beams are active. Also in this example, there are 3 pairs of beams that are incompatible: (A,C), (C,F), and (E,F). All other beam pairs are compatible.*

This example is concerned only with "receive" incompatibilities, which means that it is possible to support 2 incompatible beams on the phased array, as long as they are allocated to separate ports. Table 4 steps through one possible form of the initial pass where each beam is tested for viable allocations, but only by considering a narrow time frame of the candidate beam's duration.

Table 4. Initial pass for a sequence of 6 beams, using narrow allocation context. *In this sequence, the search for viable allocations is only performed within the context of the duration of each beam under consideration, as opposed to the longer chain of overlapping beams.*

Step	Temporary Allocations			Result
	Port 1	Port 2	Viable Allocation?	
1. Add beams A and B	(A,B)	()	Yes	2 viable allocations, A, B added
	(A)	(B)	Yes	
2. Try beam C	(A,B)	(C)	Yes	2 viable allocations, C added
	(A)	(B,C)	Yes	
	(A,C)	(B)	No – (A,C) incomp.	
3. Try beam D	(A,B)	(C,D)	Yes	2 viable allocations, D added
	(A,C)	(B,D)	No – (A,C) incomp.	
	(A,D)	(B,C)	Yes	
4. Try beam E	(B,C)	(D,E)	Yes	3 viable allocations, E added
	(B,D)	(C,E)	Yes	
	(B,E)	(C,D)	Yes	
5. Try beam F	(B,C)	(E,F)	No – (E,F) incomp.	1 viable allocation, F added
	(B,E)	(C,F)	No – (C,F) incomp.	
	(B,F)	(C,E)	Yes	

In this example, it is possible to find at least one successful port allocation for each beam, given the predicted incompatibilities, if allocations are only evaluated within the time frame of the beam itself. However, each local allocation has secondary implications on allocations for other time intervals, which in this case ultimately lead to failure. Consider the final step where only one viable allocation for beam F is found. The allocation of (B,F) on Port 1 and (C,E) on Port 2 is only considered within the scope of the time interval t_3, but it implies allocations for the other time intervals as well. Figure 3 shows the same set of beams, re-grouped into the port assignments that are logically mandated by this singular viable allocation found for beam F.

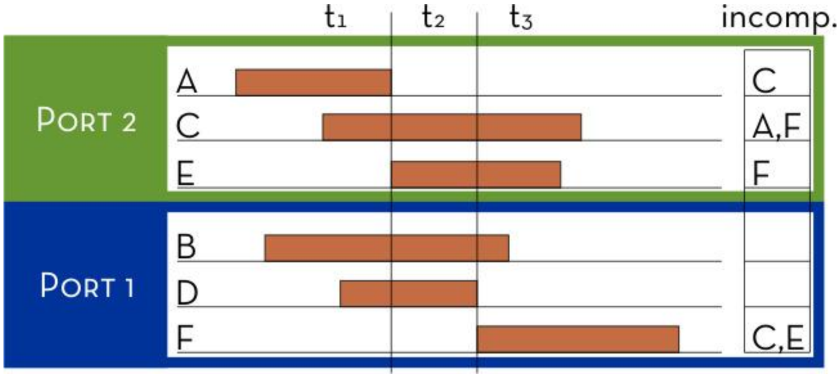

Figure 3. Re-grouping of beams by port allocations. *The only viable allocation for beam F entails the allocation of beams C and E on Port 2 for the time interval t_3. This leads to the allocation of beams A and C on Port 2 for the time interval t_1, which fails due to incompatibility.*

In the resulting allocations we see that there are at most 2 beams to a port at any time, as expected. The allocation of (C,E) on Port 2 implies the allocation of (B,D) on Port 1 for the time interval t_2. This in turn implies the allocation of (A,C) on Port 2 for the time interval t_1. But beams A and C are incompatible, and cannot be allocated together on the same port, so this allocation fails and any subsequent allocation that depends on it fails as well. Thus there is no viable allocation for beam F. But this can only be determined through an analysis of the entire sequence as opposed to focusing strictly on the duration of beam F.

This is the motivation for the current design of the initial pass logic to consider complete contiguous sequences of beams when evaluating a new beam to determine if a viable port allocation can be found. Given the frequency of the calls to the initial pass logic, this suggests a possible inefficiency in computing time. However, the current implementation of the initial pass logic attempts to minimize this impact by caching previous findings about incompatibilities as beams are tested and then added to an antenna. The search for contiguous allocations with each beam becomes a simple constraint satisfaction problem, for which highly efficient algorithms have been developed. Additionally, if the port allocation task is significantly postponed to a global pass verification at the end of initial scheduling, then this raises the question of what should be done when port allocation failures are discovered at that later stage. From the preceding example, this would present the problem of where to allocate beam F, if not on this antenna. And if a truly optimal solution is desired, this would trigger going “back to the drawing board” to determine if another beam allocated elsewhere could have been assigned to this antenna instead. If such port allocation issues can be identified efficiently during the initial pass, then this is a much more effective place to find a solution as well.

C. Global Pass Verification and Path Planning

The global pass performs two main tasks to verify the viability of the schedule assembled by the combined process of network-level scheduling with local-level inputs returned by the calls to the initial pass logic. The first task is to verify that a viable chain of port allocations can be constructed for all beams assigned to each phased array antenna. This is treated as a constraint satisfaction problem in precisely the same way as it is performed in the initial pass, only the scope is the entire duration of all scheduled supports under consideration. The second task is to generate complete beam paths for all supports allocated to phased array antennae. Recall that during the initial pass, beam paths or only considered in terms of worst case conditions, as an approximation to predict incompatibilities. However, in order to prepare for the step of actually executing support tasks for beams allocated to phased array antennae, it is necessary to calculate full paths as instructions for the local beam managers on the antennae. This process also provides final verification of the compatibility of simultaneous assigned beams.

The path planning procedure is adapted from the Probabilistic Road Maps algorithm, a motion planning algorithm in robotics, which uses a random sampling technique to find a collision-free path between a starting and goal configuration of the robot. Random sampling replaces the costly step of computing an explicit representation of free space, by a pair of tests that check for collision on every randomly picked sample from the robot’s configuration space (which is state * time dimensional) and connection between samples. The result is an undirected graph called a probabilistic roadmap, whose nodes consist of sampled, collision-free points from the *state * time* space called *milestones.* Milestones are connected by short admissible trajectories called *local paths*, that form the edges of the roadmap. The start and goal configurations of the robot are connected to this roadmap to find a collision-free path between those points in the space.

As described by Hsu, Kindel, Latombe & Rock[3], “A *complete* motion planner is one that returns a solution whenever one exists and indicates that no such path exists otherwise.” But such planners are usually exponential in the number of degrees of freedom of a robot. They further describe probabilistic completeness as follows.

> A planner based on random sampling cannot be complete … A planner is *probabilistically complete* if the probability that it returns a correct answer goes to 1 as the running time increases. Suppose that a randomized planner returns a solution path as soon as it finds one, and indicates that no such path exists if it cannot found one after a given amount of time. If the planner returns a path, the answer must be correct. If it reports that no path exists, the answer may be sometimes wrong.

Thus it shows that every new milestone added to the roadmap further reduces the chance of a path not being found where one exists.

The global pass beam path planner uses a modified version of the Probabilistic Road Maps algorithm to plan beam paths from a starting configuration to the goal configuration on the surface of the phased array. Here the beam centers are analogous to the robots. The configurations are from the state * time space of the beam. The state of the beam is represented by the position of the beam center on the surface of the phased array.

The path planner iteratively generates a tree-shaped roadmap where the root of the tree is the starting configuration of the beams. Child milestones are created from randomly selected parent milestones that already exist in the tree, and are added to the tree only if they are collision free and can be reached from their parents without collisions as well. The process continues until one of the children satisfies the goal test, which consists of a check that determines whether or not the particular milestone can reach the goal in a collision free way. If a valid path was found, the sequence of milestones at random times from the start milestone to the goal milestone needs to be converted to milestones at specific evenly spaced interpolated times.

IV. Initial Results

Figure 4 shows a sampling of the output from running the local-level allocation and planning procedures integrated with the network-level BA scheduling process.

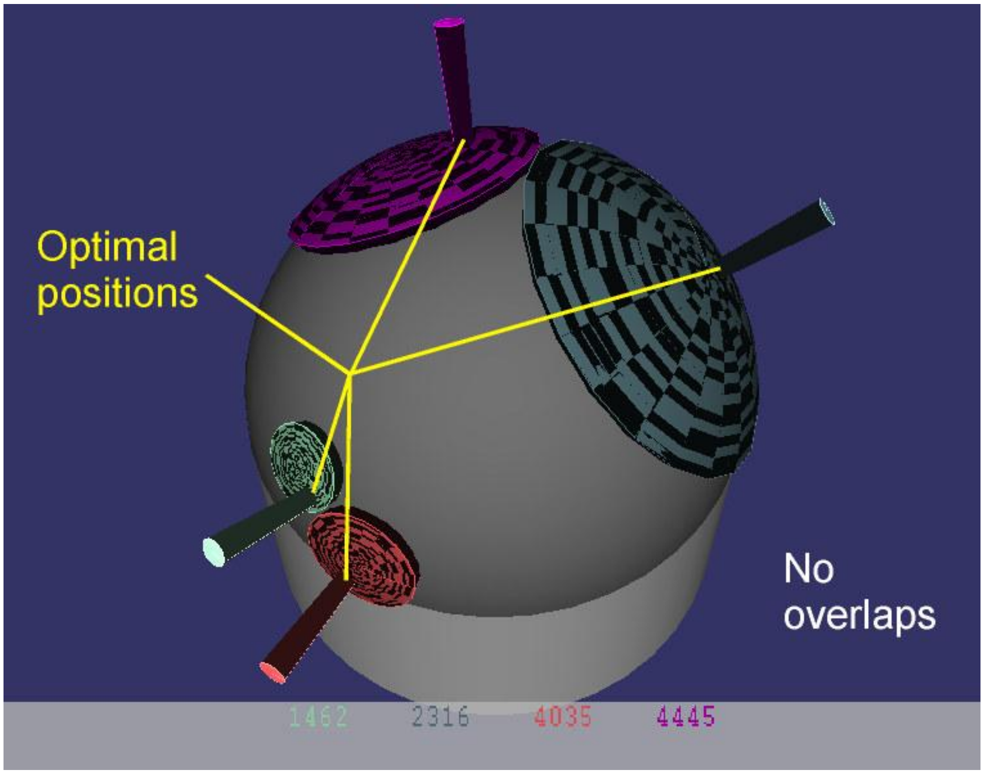

Figure 4. Four simultaneous beams with deconflicted paths. *All four beams in this example have active areas shifted off of their optimal positions for deconfliction.*

For the 4 beams allocated to this phased array antenna, the local-level analysis generated port allocations and deconflicted paths satisfying both the transmit and receive communications requirements for the satellite supports. In this instance, all 4 beams needed to be deconflicted, with active areas shifted away from their optimal positions, and the automated mechanisms accomplished this successfully. This is a typical example of the increased capacity that can be achieved with these methods, without increased burden on human schedulers.

In initial experiments scheduling a sample data set of 192 requests in an 8 hour time interval, the bottleneck scheduling process was executed with the inclusion of local-level allocation and planning for phased array antennae. The data set contains two phased array antennae, on which roughly 1/3 of requests are allocated, so 65 beams are analyzed for local incompatibilities, assigned allocations, and given calculated beam paths. This rough experiment was performed on several standard Windows computers. In this experiment, the initial pass procedure was completed in an average of 0.0044 seconds for each call, totaling 0.286 seconds for all 65 calls. The global pass procedure was completed in an average of 0.1995 seconds, called once for the entire data set after an initial schedule is developed at the network-level. Roughly speaking, with this 8 hour data set, the local-level allocation procedures are performed in less than half a second. This is well within the tolerance for computational performance, although the way forward will involve exploring further optimizations as well as additional experiments with larger data sets.

V. Conclusion

Many of the elements of the phased array logic discussed in this paper are designed to generalize for a broader range of potential constraint satisfaction problems with different kinds of phased arrays. The goal for the initial automated planning and allocation methods is to contribute to the overall feasibility of phased array utilization, by demonstrating an automated means to optimize the increased capacity that comes with such a platform. This has the potential to not only reduce the manpower requirements for scheduling activity, but also to contribute to greater overall satellite communication capacity.

References

[1] Stottler, R., “General Scheduling Service using Intelligent Resource Management Techniques,” *Infotech@Aerospace 2010 Conference, American Institute of Aeronautics and Astronautics*, Atlanta, GA, 2010.

[2] Stottler, R., Mahan, K., and Jensen, R., “Bottleneck Avoidance Techniques for Automated Satellite Communication Scheduling,” *Infotech@Aerospace 2011 Conference, American Institute of Aeronautics and Astronautics*, Reston, VA (submitted for publication).

[3] Hsu, D., Kindel, R., Latombe, J. C., and Rock, S.. “Randomized Kinodynamic Motion Planning with Moving Obstacles,” *International Journal of Robotics Research*, 21(3):233-255, March 2002.