Distribution Strategies for Collaborative and Adaptive Sensor Networks *

Bryan Horling and Victor Lesser
University of Massachusetts
Amherst, MA 01003-9264
bhorling,lesser@cs.umass.edu

Abstract – *In this paper we describe the CASA project, an ambitious engineering research problem that is drawing on the expertise of a range of disciplines to create a next-generation meteorological sensor network. This new network, named NetRad, will be distributed across a wide area, and consist of a large number of sensors that are both collaborative in their behavior and adaptive in response to changing conditions. A closed-loop control scheme is needed to efficiently and quickly manage the network's complexity, but several barriers exist which confound a simple, centralized solution as the network scales. We will present a range of distribution strategies to cope with this problem, and compare and contrast the effects they will have on the system's performance.*

1 INTRODUCTION

The recent explosion of interest in distributed sensor networks has lead to a wealth of research results in network transport and routing protocols, energy-conscious algorithms and data dissemination and storage techniques. However, the majority of those results target just one class of sensor network, namely those using small, power-limited sensors (sometimes called *motes*) that communicate over wireless networks. In this paper, we will discuss the design of a sensor network on the opposite end of the spectrum. The sensors we consider are significant, fixed installations that may consume large amounts of power and have considerable bandwidth at their disposal. As we will show, this category of sensor has important practical applications, and important research questions remain to be answered when constructing networks of such sensors.

*This material is based upon work supported by the National Science Foundation Engineering Research Centers Program under NSF Award No. EEC-0313747. Any opinions, findings, conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

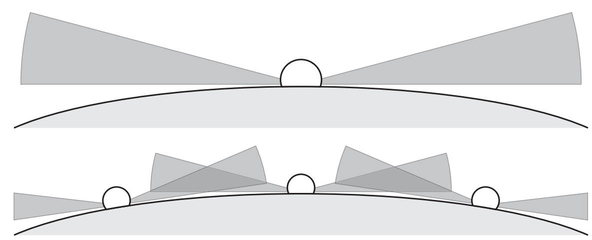

Figure 1 - A comparison with of the current NEXRAD model (top) with the proposed NetRad model (bottom).

The particular domain that we are considering is the tracking of meteorological phenomena, such as storm cells, precipitation and tornadoes. The existing radar-based weather monitoring infrastructure in the United States consists primarily of a collection of fixed, ground-based sensor installations known as NEXRAD. Each installation consists of a single WSR-88D S-band radar that has an effective range of between 230 and 345 kilometers, depending on the type phenomena being observed [1]. At the time of this writing, there are 158 operational NEXRAD radar systems, primarily situated across the continental US. Despite this relatively sparse covering, the long range of the WSR-88D allows the sensor array to provide nearly complete coast-to-coast coverage.

This infrastructure has worked admirably for over 15 years, however, it suffers from a significant drawback under certain conditions. Although the long range of the WSR-88D is able to compensate for the geographic sparsity of the NEXRAD sensor array, in doing so it also creates blind spots due to the curvature of the Earth. Consider the single sensor shown at the top of Figure 1. Radar pulses emitted by the sensor travel in a straight line. Because the Earth curves away from the radar site, an increasing large gap is created between the radar beam and the ground as we move away from the sensor. As a result, nearly 80% of the volume below three kilometers cannot be sensed [7]. Low-altitude phenomena, notably tornadoes, occur in these blind spots, which increases the uncertainty associated with discovery and limits the ability to sense them. This degrades the ability of meteorologists and emergency response workers to track and provide timely warnings for such phenomena.

The solution we are currently working on augments the NEXRAD array with a network of smaller, less expensive sensors, collectively known as NetRad. As shown at the bottom of Figure 1, a denser array of smaller-ranged sensors will not be as affected by the Earth's curvature. This reduces the blind spot's size and correspondingly increases the ability to sense low-altitude phenomena. The redundant arrangement of sensors can also help compensate for other existing line-of-sight barriers, such as buildings, trees and local topography.

The NetRad array offers several other advantages over the existing NEXRAD system. These include the ability to take more measurements, take more directed measurements and take advantage of multiple, overlapping sensor regions to produce higher quality data. These will be covered in more detail in Section 2. Initial deployments of the NetRad array will be relatively small, consisting of between two and nine sensor nodes, and targeted to areas such as the midwest where they offer the most benefit. However, the more generic advantages listed above motivate the use of NetRad in areas not necessarily prone to severe low-altitude phenomena. Because of this, we envision that NetRad arrays may grow significantly larger, to hundreds or thousands of nodes.

It is on these larger arrays that we will devote most of our attention in this paper. The NEXRAD approach, because it operates in one of two modes and does not generally consider areas of sensor overlap, can use direct human interaction as a viable means of real-time control. The increased speed of the NetRad sensors, coupled with their ability to take several types of measurements and even collaborate with neighboring sensors makes controlling such an array in an effective manner a more challenging problem. Although human input will be used to create and weight sensing tasks, we believe the primary control process must be mostly or completely automated, creating a closed-loop cycle where observations suggest actions that produce the next set of observations. As we will discuss in the next section, several factors conspire to make control even more difficult as the network scales, and preclude a purely centralized architecture. A distributed approach that takes advantage of locality to reduce the size of these problems without unduly affecting quality seems to be a more viable strategy.

We will present an overview of our previous work in a similar sensor network domain in Section 3. In that network's design, we demonstrated that partitioning the environment into smaller, more manageable pieces can be an effective way of organizing the distributed solution [5]. A distinguished sensor in each partition serves as a local locus of control, as shown in Figure 2, and the partition boundary itself can be made permeable to data flows or control requests. By limiting interactions to only those sensors within or adjacent to the partition, the burden on this distinguished sensor is made tractable and the control problem becomes more manageable. Furthermore, because it is typically only this proximal set of

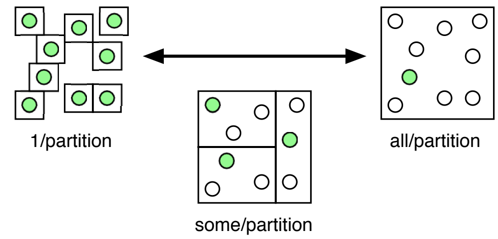

Figure 2 - A range of potential partition layouts. Shaded nodes represent partition controllers.

sensors which are actually relevant to the decision process in question, this limitation should not significantly reduce the quality of those decisions.

Given this template, one must decide how large the partitions should be, where processing is to be performed, what data will be transferred, and how nonlocal requests are prioritized, among other things. Each of these characteristics can affect and be affected by available resources, the required level of quality, timing considerations and other local or global constraints [4]. In Section 4 we will present a range of different distribution schemes based on this pattern, and compare and contrast their ramifications.

2 DCAS

The NetRad array outlined above is being designed as part of the Collaborative Adaptive Sensing of the Atmosphere (CASA) project. It will also be one the first deployed examples of a distributed, collaborative and adaptive sensor network (DCAS) [7]. Sensors in most existing weather tracking systems operate under human control, and are relatively independent of one another; each covers their own area and data fusion takes place outside the network itself. The NetRad array will consist of a network of relatively autonomous sensors which cooperate, to both opportunistically share relevant data and dynamically adapt control policies to meet the needs of a changing environment. These new capabilities make the DCAS approach challenging and rewarding from both research and practical perspectives. Designing such a system requires the expertise and talents from a wide range of disciplines, including sensor design, meteorology and distributed computation. A collaboration of over 100 scientists, researchers, students and industry partners have been assembled in the CASA project to meet these needs. The design outlined earlier and presented in more detail in this section is a result of the combined efforts of many of those individuals, as referenced at the end of this paper. Our main contribution to this effort will be presented in Section 4.

To better understand the characteristics and ramifications of the distribution schemes we will outline later, we will provide

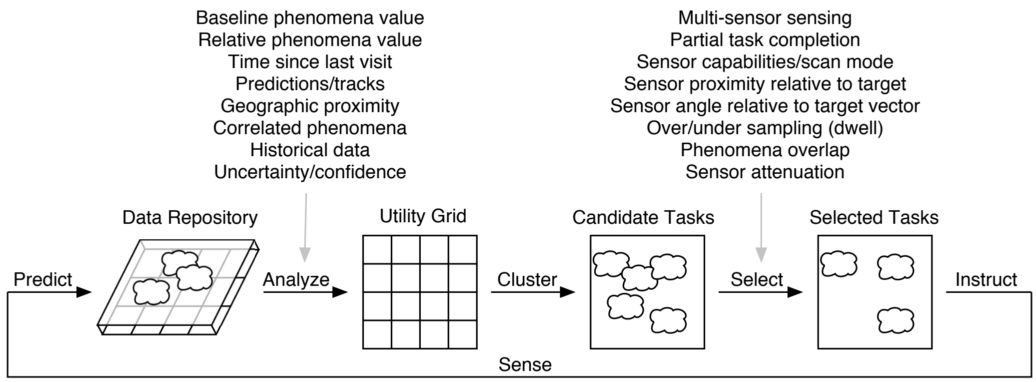

Figure 3 - A high level overview of the control process, along with the domain characteristics which influence it.

some additional insight into NetRad's characteristics, beginning with the sensors themselves. We define the time needed for a radar to complete a desired series of measurements as it's *cycle time*. The sensors in NEXRAD operate in one of two modes: *clear air* mode and *precipitation* mode, which differ principally in the sensitivity of measurements that are taken. In both cases, the radar performs a series of $360°$ azimuthal sweeps to sense different elevations in the atmosphere, which leads to a cycle time of between five and ten minutes. NetRad radars will not be constrained to perform complete sweeps, nor will they have to sense all levels of the atmosphere. This allows the cycle time to be reduced to roughly 30 seconds, so that more measurements can be taken over time.

NEXRAD-style sensors are not inherently confined to $360°$ sweeps, this limitation is due the fact that the mounting apparatus and dish rotation mechanism were not engineered to withstand the stresses caused by continuous, rapid changes in rotational velocity. NetRad sensors will incorporate more robust steering mechanisms, and will therefore be able to target one or several specific points in space during a single cycle. This additional flexibility allows for more sophisticated scan strategies to be employed. For example, one might decide to ignore a particular area which is either uninteresting or who's characteristics are already known, so that the sensor can devote additional sensing time to more uncertain or more important areas.

A consequence of the targeted sensing technique is that some regions will be sensed less frequently than others. When the phenomena in question are confined to a relatively small area, this does not generally cause problems. However, when phenomena are dispersed, the sensor may be forced to choose among targets if high quality measurements are required. To cope with this problem, the NetRad array will be designed with areas of redundant coverage, incorporating regions where multiple sensors may take measurements. In the case where one sensor cannot accomplish its goals by itself it may be possible for other sensors to be employed in its stead. Recent work in meteorological radar interpretation has also demonstrated the benefits of measuring the same region with two or more sensors concurrently. For example, the correct angular component of wind velocity can be determined when two relatively orthogonal sensors take measurements of the same space. This capability can be useful when detecting tornadoes, since they exhibit particular and recognizable wind shear patterns. Other research has shown that more accurate compensation values can be determined for sensor attenuation and clutter if data from multiple sensors are used [2]. By overlapping the sensors' coverage areas, the array can better adapt to demanding scenarios and produce qualitatively different data than a single radar by itself.

Figure 3 gives a high-level view of how the NetRad control process; for more additional details see [9]. Data arrives from the radars and predictions are made, both of which are incorporated into a data repository. This data is then analyzed, and a host of domain-specific characteristics are used to assign a baseline utility to particular areas. Areas of common utility are then clustered into discrete tasks for consideration. A second process, which uses another set of domain-specific criteria, selects the tasks to be pursued, based on their relative importance and the capability of the sensor array to service them. These tasks are then delivered to the relevant sensors, which produce the data to be used on the next cycle.

The flow depicted in Figure 3 is far from comprehensive, but it does illustrate several places where a large scale NetRad array will encounter obstacles. First is the sheer volume of data that will be created by the array. The raw output from a single sensor can produce up to 100 Mbs of data. A more succinct, but still useful version of the same output generates only 2 Mbs. However, because that volume will scale linearly with the number of sensors, it quickly becomes impractical to route all sensor data to one location. Even if were possible to do so, the amount of data would likely overwhelm the meteorological algorithms to the degree that the desired 30 second cycle time would be unattainable. The component with the most constraining complexity performance is the selection process, which must optimize the allocation of sensors to tasks. Consider all the flexibility and adaptability outlined above. These capabilities translate to an enormous number of alternatives at the task selection level, each with potentially different performance characteristics and utility. In fact, the number of alternative configurations grows exponentially with both the number of sensors and the number of tasks. Finding an optimal assignment of tasks quickly becomes intractable as the size of the network grows, and even a satisfactory solution can be elusive under demanding conditions.

Fortunately, we do not believe it is necessary to centralize all data and decision making at a single location to obtain reasonable performance. This optimization problem exhibits spatial locality, because distant information should not be required to make an effective local decision. For example, a detailed view of the weather in the far western part of a state is normally not needed to decide an appropriate course of action for a sensor in the far eastern part of that state. Because of this, it should be possible to distribute the problem in a way that reduces the scope of the optimization problem to manageable levels, without a significant reduction in system utility. Deciding how to effectively distribute the problem is a challenge of its own, which we will begin to address in the next two sections.

3 BACKGROUND

Prior to this current work on CASA, our research laboratory participated in a four year, DARPA-funded effort known as the ANTs project (Autonomous Negotiating Teams) [6]. Like CASA, the working environment of ANTs consists of an array of Doppler sensors, which are used to track multiple, mobile targets. In fact, many of the challenges and assumptions of the CASA project are similar to those we experienced in ANTs, including the need to cope with mounting complexity as the network scales. Our initial attempt at addressing CASA's distribution problem will be to build upon those experiences, by exploring the approaches used in ANTs, and determining if they remain viable within the CASA domain.

Although the sensors used in ANTs are much smaller and less expensive than those we described in Section 2 (for example their range is measured in feet, not miles), they still share several other important characteristics. Like CASA, the ANTs sensors have fixed locations, a wired power source, and a fair amount of local processing power. Each sensor has a process that runs on the local processor that is responsible for making decisions and controlling the sensor. We will refer to this local process as an agent.

The sensors used in ANTs can return only simple amplitude and frequency values, so no individual sensor is capable of precisely tracking a target by itself. Instead, the agents that control the sensors must collaborate in some way to achieve their common goal. The sensors must therefore be organized and coordinated in a manner that permits their measurements to be used for triangulation, and geographically distinct groups of such coordinated sensors used to produce a continuous track. More measurements, and particularly more measurements taken in groups at approximately the same time, will lead to better triangulation and a higher resolution track. Additional hurdles include the need to scale to hundreds or thousands of sensor platforms and the ability to operate within a real time, uncertain environment. As with CASA, closed-loop control is necessary to make control decisions in a timely manner. A more detailed description of the entire framework and the environment it operates in can be found in [6].

Unlike the CASA domain, the ANTs sensors are connected with a FM-based wireless network that is divided into eight communication channels. Each channel has limited capacity, and agents may communicate over only one channel at a time. The provided transport protocols are unreliable, and congestion can quickly result in significant losses of messages.

Because the effective range of the ANTs sensors is relatively small, it is important to have a low cycle time, to ensure a reasonable number of measurements can be taken. For example, it is not uncommon for a target to enter and leave a sensor's viewable area in less than ten seconds. A cycle time of one second is used, which in practice results in a tight, but not onerous constraint on computation. This is in contrast to the DCAS architecture, where computational complexity appears to be the most limiting factor. In ANTs, limits on communication proved dominant, and much of our solution is designed to cope with this barrier. The question then, is whether that solution, devised for a somewhat different purpose, can be effective under a new set of conditions.

4 DISTRIBUTION

The concept of *organizational design* is used in many different fields, and generally refers to how members of a society act and relate with one another. This is also true of distributed and multi-agent systems, where the organizational design of a system can include a description of what types of agents exist in the environment, what roles they take on, and how they interact with one another. The objectives of a particular design will depend on the desired solution characteristics, so for different problems one might specify organizations which aim toward scalability, reliability, speed, or efficiency, among other things. The questions we are attempting to answer in CASA, such as where computational responsibilities lie, how data is routed through the network, and how individual sensors behave, can therefore be framed as an organizational design problem.

The organizational design used in our ANTs system is intended to address the scalability problem, by exploiting spatial locality and organizational constraints to impose limits on

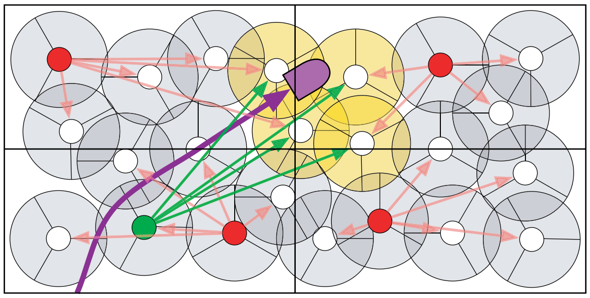

Figure 4 - An example ANTs sensor organization. Each divided circle represents a sensor, with the center color indicating its role. Arrow represent communication.

how far classes of both control and data messages propagate. Our design uses environmental partitioning to create localized regions of interaction, called sectors, as shown in Figure 4. Within these sectors, agents take on different and potentially multiple responsibilities which dictate their individual behaviors. The number of sensors in these sectors affects how efficient the system is, as large regions may create unwelcome disparities in communicative or processor load, and small regions make it difficult to quickly send and receive the necessary information to make decisions. We have shown how sector size affects the overall communication load, load disparity between agents, average communication distance, and the quality of tracking in the system [4]. By varying just this one aspect of the organization, we demonstrated that the performance of the system can be greatly influenced by the organization's design parameters.

This partitioning and role assignment strategy appears to be a viable option for the CASA domain as well. Under such a design, a logical partitioning arrangement would be imposed over the sensor network. Within each partition, the agents controlling the sensors could take on particular roles. Partitions act as semi-permeable barriers, causing agents to limit the types of interactions they have with agents and sensors outside their own partition. An extreme example of such a configuration allows no interactions at all. Each partition contains an isolated fragment of the sensor network, operating independently of its neighbors. A single distinguished node, which we shall call the *manager*, acts as the locus of control, collecting data and assigning tasks to sensors in the sector. By artificially limiting the geographic area in this way, the number of sensors and tasks will be similarly reduced. This will constrain both the computational load incurred by the optimization process and the magnitude of sensor data that must be handled.

As with ANTs, selecting the appropriate design for the DCAS array will depend on identifying relevant characteristics, and determining how they are affected by different organizations. Some of these characteristics are the same as those observed in ANTs, such as communication and computational loads. Others, such as the utility-affecting factors from Figure 3 are more domain-specific. We itemize and expand on a subset of these below:

Computational Load This represents the amount of processing work that must be performed by agents. Both average and individual loads are relevant.

Communication Load The amount of data which must be sent or received by the agents.

Control Quality The quality of task optimization decisions that can be made by the agents. Fewer available sensors reduces the number of ways tasks can be assigned, which can limit the quality attainable by the control process.

Data Availability The percentage of the total amount of relevant data that is available when and where it is needed.

Capability Availability The availability of relevant sensor or processing capabilities. For example, a sensor might be needed to take a measurement in a particular area. Capabilities therefore include location as well as any special functionality (in the case of a heterogeneous sensor network).

In each of these characteristics, we would compare the organizations performance against some ideal or oracle when possible. For example, control quality would be compared to optimal control decisions made by an equivalent, theoretical sensor network that is free of resource or time constraints. So, if the optimal decision in a scenario is to use sensors A and B, but the design causes only A to be accessible, we have sacrificed the control quality. Similarly, if data flow is somehow constrained, we would compare the limitation against what would be available in that theoretical network. The organization being compared is then evaluated on the magnitude, importance, and frequency of those differences.

Other important factors exist that are derivations of those in the list above. For example, the average time needed to discover new phenomena by the system is dependent on the control quality and the capability availability. Similarly, the accuracy of analyses and predictions is dependent on the data availability and computational load. Ideally, we would like to devise a model or set of tests which evaluates a fundamental set of characteristics. The quality of derivative factors could then be estimated from those characteristics, and a utility function incorporating all these values could assess the overall performance of that organization. This process will be discussed briefly in Section 5.

4.1 Strict Partitions

We will refer to the rigid partitioning example we outlined above as *strict* partitioning. In this design, there is just one

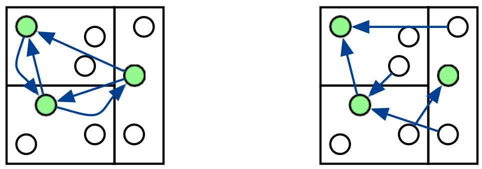

Figure 5 - Data-permeable partitions, either manager initiated (left) or sensor initiated (right).

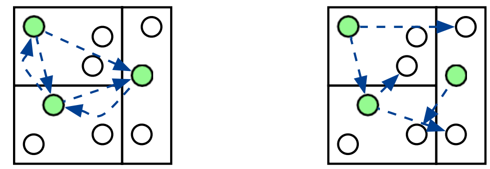

Figure 6 - Control-permeable partitions, either manager directed (left) or sensor directed (right).

parameter - the number of sensors to include in each sector. No communication or other forms of interaction take place across sector boundaries. This strategy has almost ideal scalability characteristics. Because each sector is independent, if an appropriate size can be determined for one sector, the array as a whole can grow to an arbitrary size by simply dividing the sensor population accordingly. As shown earlier in Figure 2, there are a range of possible sector sizes, from just a single sensor, to the size of the entire array. The single-sensor design is similar to the current NEXRAD organization. Each sensor is an island, operating without regard to its neighbors. The all-sensor design represents our baseline approach, where anyone can communicate with everyone.

These two approaches provide good examples of the tradeoffs we are concerned with. The single-sensor design has very good load characteristics, but limiting the access to just one sensor results in poor control quality, data and capability availability. The all-sensor design has very good control quality and availability, but as we have shown, the computational and communication burden on the manager will be too high. This matches our experience with the ANTs sensor array, where larger sectors correlated with both better RMS error rates and worse load distribution [4].

Without constructing an accurate model, or performing empirical tests, it is hard to estimate just how much the lack of interaction caused by the boundary will affect the performance of the system. However, it is clear that those areas near the edge will not be able to take advantage of one of the principle benefits of the NetRad design, namely the ability to sense a region with multiple radars. This will limit the types of measurements that can be taken in those areas, and create a more difficult sensor allocation problem by ignoring the built-in redundancy. We will continue by exploring how these deficiencies can be addressed.

4.2 Data-Permeable Partitions

One possible strategy is to make the partition boundaries permeable to data flows. By this we mean that some sensor data will be exchanged among sensors. Data could either be automatically pushed to neighboring sectors, or opportunistically pulled when the need arises. As shown in Figure 5, this arrangement can route data in two different ways. One option is to take advantage of the fact that the local manager is already centralizing the sector's data, and have it serve as the source. The drawback to this approach is the manager already suffers from a significant communication load, which is exacerbated by this design. The other option is to directly involve the sensor(s) producing the data, which avoids loading the manager but creates a larger query pool in the case the pull mechanism is used. Conversely, a push mechanism would require the agent at each sensor to locally determine nonlocal relevance or importance, which may not always be possible.

Because no control messages are passed between sectors, any data which is shared through this scheme is undirected. The recipient will have had no influence on the original measurement type or direction. In a pull-based system it may request measurements of only a particular type, but sending that request will not affect the remote manager's task optimization process. So, to a certain extent, this mechanism requires some amount of luck for the correct data to be produced and then shared. That said, we can expect that some emergent behaviors may make this technique more viable than it initially appears. For example, if an interesting phenomena is occurring near a partition border, it is likely that both managers will assign that phenomena a high utility. Therefore, it is also likely that both managers will choose to sense in that area, so that synergistic measurements will taken and potentially shared. This raises this design's data availability characteristic. However, because the two managers do not coordinate these activities, they cannot expect that this cooccurrence will happen, and the control quality and capability availability will remain compromised.

4.3 Control-Permeable Partitions

The natural next step in this process is to make the partition permeable to control messages. As with data flow, this can occur in two ways, as seen in Figure 6. In the manager directed approach, one manager would contact another, requesting the use of a particular sensor, or requesting a measurement in a particular area. In the sensor directed approach, the manager would contact a particular sensor directly. This latter option has rather profound affects on the control cycle as we have described it. In this design, individual sensors would regain some of the autonomy they had previously ceded to their local manager. Upon receipt of conflicting tasks, the sensor would decide locally which it would service, and potentially how it would service them. Although this does simplify the control process of the managers somewhat, it does so at the expense of quality, since an individual sensor will likely not have the context necessary to always make the globally correct decision. Sensing tasks could go unsatisfied, which would reduce the predictability of the system. This does not seem to be a viable strategy.

The manager directed approach avoids this problem by allowing individual managers to retain control of their local sensors. However, their individual control processes are made more complicated, because this design has the same effect as increasing the size of the sector so that neighboring sectors overlap. When the manager has the option of requesting measurements from its neighbors, its partition has in some ways grown in size to incorporate neighboring sensors, and the number of alternative configurations it must consider increases accordingly. Moreover, some of these configurations now have some probability of failure, because requesting a remote measurement does not guarantee it will actually be taken. One could incorporate a request-commit phase, where the target manager agrees or declines to perform the requested task, but the communication delay will add delays to an already time-bounded process. Finally, when conflicts do arise between local and nonlocal tasks, it is not always straightforward to choose among them. For example, one could simply resolve conflict by selecting the task with the higher expected utility. However, if that task could be accomplished by another sector at slightly reduced utility without conflict, that may be the globally better option. Maintaining or obtaining the state necessary to correctly make that decision adds further overhead.

The benefit to this approach is that activities requiring multiple sensors, or tasks that could benefit from redundancy, can now be accommodated. This increases the control quality, and data and capability availability, with the drawbacks outlined above. It is possible to use this design, but to ignore or heuristically solve the more complex aspects (such as seeking globally correct decisions, ensuring success for all tasks). In this case, it is unclear how much performance will degrade, particularly in pathological scenarios. Again, it is hard to precisely predict how much these additions will improve the performance of the system in the absence of real-world data.

4.4 Dynamic Partitions

To conclude our design alternatives, we will mention two ways in which the partitioning process can be made dynamic and adaptive. The first, shown on the left in Figure 7, is to incorporate a form of *roaming* partition. In this design, noncritical phenomena are sensed using one of the partitioning strategies outlined above. However, when a particularly important object is detected, a partition is dynamically created to encompass it. A new manager is also dynamically selected to control the sensors in that new partition. This manager's tasks will take precedence over those of the existing managers. As the phenomena moves, the roving sector follows it, maintaining a continuous track over its lifetime. The organizational design implemented by our ANTs system incorporated a variant of this approach. The highlighted sensors in the center of Figure 4 represent such a roaming sector. This strategy avoids the partition boundary problems, as the sector is assumed to always encompass the sensors that are needed, and therefore the control quality and capability availability characteristics are good. If the size of the sector can be kept to a reasonable level, this approach also has good computational and communication loads. The drawback to this design lies in the additional complexity needed to implement it. In addition, it requires a conflict resolution strategy, since multiple roaming sectors may intersect and contend for the same sensors.

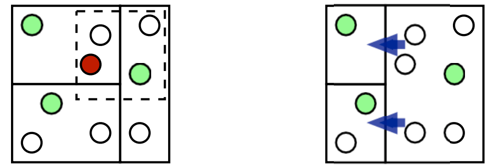

Figure 7 - Dynamic partitions, either roaming (left) or resizable (right).

On the right side of Figure 7 is an example of resizable partitions. In this design, sectors may grow and shrink in response to demand. For example, instead of making a single request to use a particular sensor, a manager could request to transfer that sensor to its control. Like the roaming strategy, this technique can avoid the partition boundary problem by effectively moving the boundary away from the phenomena. However, deciding when it is appropriate to both request and relinquish control of a sensor is a complex problem, with many tradeoffs that can be difficult to gauge in a timely manner. This strategy can also suffer from conflicts when two or more managers desire the same sensor. This strategy has been explored in greater detail in [8].

5 Conclusion

In this paper we have explored a range of different organizational possibilities for a DCAS-style sensor network. The alternatives presented are by no means exhaustive, and completely different organizational paradigms may ultimately replace or augment the partitioning approach. For example, a data distribution hierarchy that incorporates aggregation and summarization capabilities might help manage large data flows, while coexisting with a partitioned organization.

Although in the previous section we compared and contrasted the qualitative characteristics of the different approaches, we feel that a quantitative approach is needed to more rigorously evaluate candidate designs in context. To this end, we have developed a domain-independent organizational design modeling language, able to capture quantitative details of organizations and environments, and use that information to predict runtime characteristics [3]. Using such a tool, one can determine, for example, what the appropriate sector size is for a given design, and how information and control decisions flow through the organization to produce measurements or create load imbalances. This type of model can define primary and derivative factors such as those listed in Section 4, how they interact, and how they combine to determine a utility value for the organization as a whole. Given this value, one can then rank and select the most appropriate organizational design.

6 Acknowledgments

CASA's high-level architecture, as outlined in the introduction, is the responsibility of a number of individuals from different institutions, including but not limited to David McLaughlin, V. Chandrasekar, Kelvin Droegemeier, Stephen Frasier, Jose Colom, Jim Kurose, and Sandra Cruz-Pol. The control architecture presented in Section 2 was designed by members of CASA's distributing thrust, and influenced through interactions with many other CASA participants. Principle design was the effort of David Westbrook, Michael Zink, Jim Kurose, Sherief Abdallah, Armen Babykian, Jerry Brotzge, Kurt Hondl, Eric Lyons, Victoria Manfredi, Mark Preston and the authors. We also wish to thank a number of people who provided background meteorological information for this paper, including Jerry Brotzge, V. Chandrasekar, Kurt Hondl, Pat Kennedy and Eric Lyons.

References

[1] T. Crum and R. Alberty. The WSR-88D and the WSR-88D operational support facility. *Bulletin of the American Meteorological Society*, 74(9):1669–1687, 1993.

[2] J. E. Evans. Achieving higher integrity in NEXRAD products through multi-sensor integration. Presented at the American Meteorological Society 79th Annual Conference, January 1999.

[3] B. Horling and V. Lesser. Quantitative Organizational Models for Large-Scale Agent Systems. In *Proceedings of the International Workshop on Massively Multi-Agent Systems*, Kyoto, Japan, December 2004.

[4] B. Horling, R. Mailler, and V. Lesser. A Case Study of Organizational Effects in a Distributed Sensor Network. In *Proceedings of the International Conference on Intelligent Agent Technology (IAT 2004)*, Beijing, China, September 2004.

[5] B. Horling, R. Mailler, J. Shen, R. Vincent, and V. Lesser. Using Autonomy, Organizational Design and Negotiation in a Distributed Sensor Network. In V. Lesser, C. Ortiz, and M. Tambe, editors, *Distributed Sensor Networks: A multiagent perspective*, pages 139–183. Kluwer Academic Publishers, 2003.

[6] V. Lesser, C. Ortiz, and M. Tambe, editors. *Distributed Sensor Networks: A Multiagent Perspective (Edited book)*, volume 9. Kluwer Academic Publishers, May 2003.

[7] D. J. McLaughlin, V. Chandrasekar, K. Droegemeier, S. Frasier, J. Colom, J. Kurose, and S. Cruz-Pol. Distributed collaborative adaptive sensing (DCAS) for improved detection, understanding, and prediction of atmospheric hazards. In *Proceedings of the 2005 American Meteorological Society (85th AMS Annual Meeting), Ninth Symposium on Integrated Observing and Assimilation Systems for the Atmosphere Oceans, and Land Surface (IOAS-AOLS)*, 2005.

[8] M. Sims, C. Goldman, and V. Lesser. Self-Organization through Bottom-up Coalition Formation. In *Proceedings of Second International Joint Conference on Autonomous Agents and MultiAgent Systems (AAMAS 2003)*, pages 867–874, Melbourne, AUS, July 2003. ACM Press.

[9] M. Zink, J. Brotzge, V. Chandresakar, K. Droegemeier, J. Kurose, D. McLaughlin, B. Philips, M. Preston, P. Rees, S. Sekelsky, and D. Westbrook. Distributed collaborative adaptive sensing for hazardous weather detection, tracking, and predicting. In *Lecture Notes in Computer Science (Dynamic Data Driven Application Systems (DDAS))*. Kluwer Academic Publishers, 2004. Also, Technical Report 04-93, Department of Computer Science, University of Massachusetts.