Closed-loop Architecture for Distributed Collaborative Adaptive Sensing of the Atmosphere: Meteorological Command & Control

Michael Zink* and Eric Lyons and David Westbrook and Jim Kurose
Computer Science Department,
University of Massachusetts,
Amherst, MA, USA
E-mail: zink@cs.umass.edu
E-mail: elyons@cs.umass.edu
E-mail: westy@cs.umass.edu
E-mail: kurose@cs.umass.edu
*Corresponding author

Dave Pepyne
Electrical and Computer Engineering Department,
University of Massachusetts,
Amherst, MA, USA
E-mail: pepyne@ecs.umass.edu

Abstract: Distributed collaborative adaptive sensing (DCAS) of the atmosphere is a new paradigm for detecting and predicting hazardous weather using a dense network of short-range, low-powered radars to sense the lowest few kilometers of the earths atmosphere. DCAS systems are collaborative in that the beams from multiple radars are actively coordinated in a sense-and-respond manner to achieving greater sensitivity, precision, and resolution than possible with a single radar; DCAS systems are adaptive in that the radars and their associated computing and communications infrastructure are dynamically reconfigured in response to changing weather conditions and end-user needs. This paper describes an end-to-end DCAS architecture and evaluates the performance of the system in an operational testbed with actual weather events and end-user considerations driving the system. Our results demonstrate how the architecture is capable of real-time data processing, optimization of radar control, and sensing of the atmosphere in a manner that maximizes end-user utility.

Keywords: closed-loop architecture, adaptive sensing, experimental evaluation, environmental sensor network testbed

Reference to this paper should be made as follows:

Biographical notes:

1 INTRODUCTION

Collaborative adaptive sensing of the atmosphere (CASA) is a new paradigm for detecting and predicting hazardous weather using a dense network of short-range, low-powered radars to sense the lowest few kilometers of the earths atmosphere (McLaughlin et al. (2005)). This new paradigm is achieved through a distributed, collaborative, adaptive sensing (DCAS) architecture. Distributed refers to the use of large numbers of small radars, whose range is short enough to see close to the ground in spite of the Earth's curvature and to avoid resolution degradation caused by radar beam spreading. Collaborative operation refers to the coordination of the beams from multiple radars to cover the blind regions of their neighbors (cone of silence) and to simultaneously view the same region in space (when advantageous), thus achieving greater sensitivity, precision, and resolution than possible with a single radar. Adaptive refers to the ability of these radars and their associated computing and communications infrastructure to dynamically reconfigure in response to changing weather conditions and end-user needs. The principal components of a CASA DCAS system includes the sensors (radars); algorithms that detect, track, and predict meteorological hazards; interfaces that enable end-users to access and interact with the system; storage; and an underlying substrate of distributed computation that dynamically processes sensed data and manages system resources. IP1 is a prototype CASA DCAS system, located in southwestern Oklahoma, whose goal is to detect tornados within 60 seconds of formation and to track tornado centroids. At the heart (or perhaps more appropriately, the brains) of IP1 is its Meteorological Command and Control (MC&C) that performs the system's main control loop - ingesting data from the remote radars, identifying meteorological features in this data, reporting features to end-users, and determining each radar's future scan strategy based on detected features and end-user requirements. In this sense, IP1 is truly a closed-loop, end-to-end system, from the sensing radars through the computing and communication infrastructure and algorithms, to the end-users. Unlike existing architectures, like NEXRAD (NEXRAD (2007)) where the radars execute predefined scanning patterns (VCP) and data are pushed to the end-users independent of their preferences, a DCAS system is a pull system where the sensing resources dynamically adapt to end-user needs and preferences.

In this article, we describe the architecture of the MC&C and evaluate its performance in an operational testbed, with actual weather events and end-user considerations driving the system. We first introduce the end-to-end architecture that spans the system from the radars to multiple end-user groups. The overall architecture is a user- and data- driven closed-loop soft real-time system, in which a blackboard decouples timing dependencies between data ingest and the generation of radar control commands. Radar control is an essential component of our system, since - in contrast to many other sensor networks actuation is involved in the sensing process. We will see how end-user considerations are deeply embedded into radar control via a utility-based approach. The architecture is also modular (as evidenced by the use of multiple independently-develop meteorological detection algorithms, and the ability to ingest data from multiple exogenous sources), making modifications and the addition of new functionality fairly easy. Finally, we demonstrate and analyze MC&C performance in an operational testbed during a 9-week experiment, focusing on a detailed analysis of two particular severe weather events. Our results show that the system is capable of real-time data processing, radar control optimization, and user-driven collaborative adaptive for sensing of the atmosphere. Finally, we demonstrate and analyze the system performance of the MC&C architecture in an operational testbed during a 9 week experiment. We show detailed results from 2 severe weather events that occurred during the experiment phase. The results of this in-depth analysis show that the system is capable of real-time data processing and optimization for the control of the radars, and sensing of the atmosphere is performed according to the end-users' needs.

The remainder of the article is structured as follows. In Section 2, a brief overview of the IP1 DCAS testbed in southwestern Oklahoma is given. A detailed description of the Meteorological Command and Control (MC&C) architecture and its components is given in Section 3. In Section 4, the adaptive reconfiguration of the system based on atmospheric conditions and end-user needs is described. The first experiment with the system in fully operational mode is described in Section 5 and an analysis of the system during two weather events is presented. Section 6 concludes the article and gives and outlook on future work.

2 DISTRIBUTED COLLABORATIVE ADAPTIVE SENSING

2.1 The IP1 Testbed

The CASA Engineering Research Center [1] has designed and built a 4 radar DCAS prototype testbed which is deployed in southwestern Oklahoma (Junyent and Chrandrasekar (2007), Zink et al. (2005), Kurose et al. (2006)). Figure 1 shows the location and coverage area of the testbed. Operational since the summer of 2006, the testbed consists of four mechanically steered parabolic dish X-band radars (see Figure 2) atop small towers, networked via Internet to a central control site (later to be decentralized as the number of radars increases) at the University of Oklahoma in Norman.

The radar nodes are spaced approximately 30 km apart and together scan a volume of 80km x 80km and up to 3 km in height and communicate with a central control site, known as the SOCC (System Operations Control Center). An overview of the IP1 system architecture is given in Figure 3

Each radar is tasked to scan an angular sector between 60° and 360° degrees in azimuth and in 1° increments in elevation, with a range gate size of 48 meters out to 30 km. While existing meteorological radar systems such as NEXRAD generally operate in sit and spin mode (taking full 360-degree volume scans independently of location and type of meteorological features present), the IP1 radars are tasked by the MC&C to focus on volumes of high interest to end-users, as discussed in Section 4.

As shown in Figure 2, each radar consists of three subsystems:

- The Rotating Tower Top houses the polarimetric Doppler radar (Bringi and Chandrasekar (2001)), an

[1] http://www.casa.umass.edu

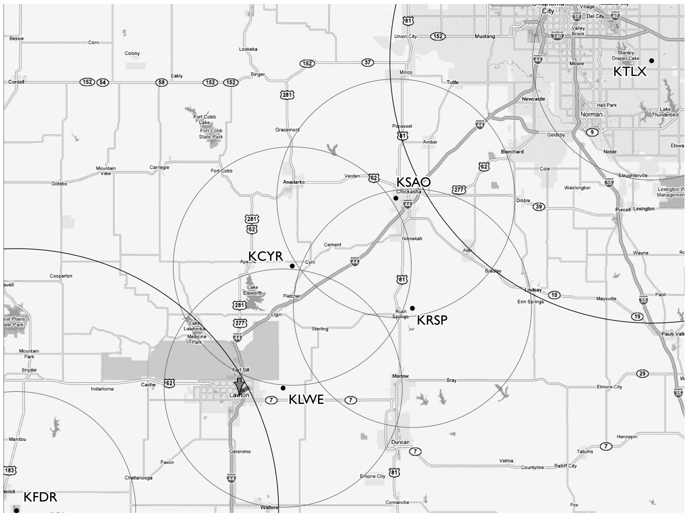

Figure 1: Location of the 4 IP1 radar nodes in Southwestern Oklahoma. The circles around KSAO, KCYR, KLWE, and KRSP show the 30 km coverage area of the IP radars. The nearest NEXRAD sites located near the IP1 testbed are the radars at Twin Lakes (KTLX) and Frederick (KFDR) and are shown here with 30 km and 60 km range rings.

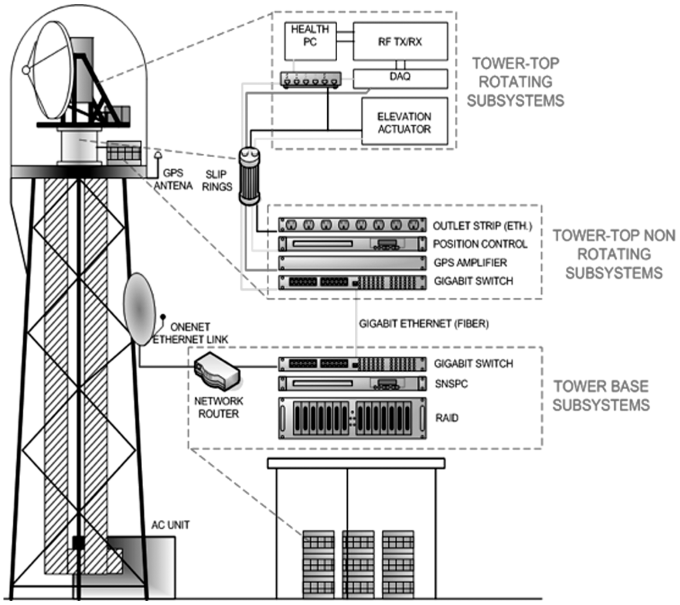

Figure 2: Architecture of an IP1 radar node

embedded system that monitors the radars operational parameters and enables operator actions in the case of anomalies (e.g., mechanical problems), and the data acquisition system (based on Field Programmable Gate Array (FPGA) technology). The FPGA(Khasgiwale et al. (2005)) processes raw digitized data (at a rate of approximately 140 Mbps) into packets that are sent through the rotating joint to Gigabit Ethernet switch in the Non-rotating tower top.

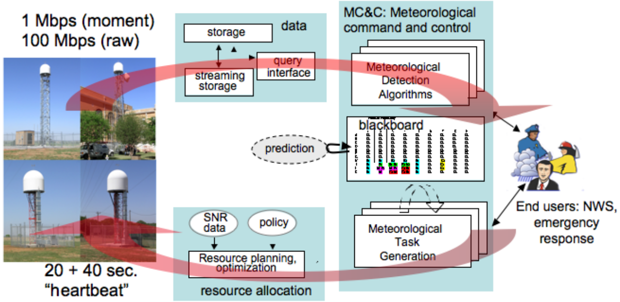
1 Mbps (moment)
100 Mbps (raw)
storage
data
streaming storage
query interface
MC&C: Meteorological command and control
Meteorological Detection Algorithms
blackboard
prediction
End users: NWS, emergency response
SNR data
policy
Resource planning, optimization
resource allocation
Meteorological Task Generation
20 + 40 sec. "heartbeat"

Figure 3: IP1 System Architecture showing the MC&C data acquisition and control loop.

- The Non-rotating Tower Top Subsystem is located below the rotating joint and houses the radar position controller, and a Gigabit Ethernet switch. The radar position controller controls the movement of the antenna via the azimuth and elevation actuators. The switch is connected to another Gigabit Ethernet switch at the tower base via a fiber optic cable.

- The Tower Base Subsystem consists of a compute cluster and an IDE RAID storage system connected via a Gigabit Ethernet switch to a router. The tower base takes raw radar data, computes moment data (essentially an average of multiple radar-pulse measurements for a given volume in space) and polarimetric variables, while performing quality control (e.g., attenuation correction and range unfolding, 2^nd trip echo suppression, ground clutter suppression) (Junyent (2007), Bharadwaj and Chandrasekar (2005), Lu et al. (2007)) on this data. The 2 Mbps moment data from each radar is sent to the SOCC over OneNet (an IP network operated by the Oklahoma state regents [2]) which is configured to provide guaranteed 4 Mbps connectivity between each radar and the SOCC. Raw radar time series data is archived at the tower base storage and can be transferred to SOCC storage as background traffic (non-real-time).

2.2 System Operations Control Center (SOCC)

The SOCC is a centralized compute cluster (later to be decentralized) interconnected via a Gigabit switch, on which the MC&C algorithms execute. As shown in Figure 3 , the SOCC has five main components (i) data ingest and storage, (ii) meteorological feature detection and multi-radar merging, (iii) feature repository, (iv) utility and task generation, and (v) optimization. We describe each of these

[2] http://www.onenet.net

functions below in more detail, roughly following the path taken by radar data through the MC&C, and the resulting re-tasking of the radars.

3 MC&C ARCHITECTURE

The MC&C is the control part of the DCAS architecture that determines how the radars will scan the atmosphere in the following 60 second heartbeat. The scanning of each radar is determined by several factors. (i) detections from data obtained in the present heartbeat, (ii) historical detections from earlier heartbeats, and (iii) end-user policies. Therefore, an architecture is required that can take these different factors into account and create the control output for the radars in a quick and correct manner. In the following, an overview of the components that, in combination, comprise the MC&C will be given.

3.1 Data Ingest and Storage

In this component, moment data is streamed from the sensor nodes to the MC&C detection algorithms, and written to storage. Data from each elevation scan is sent from a remote radar to the MC&C data ingest routines using LDM a client/server middleware that reliably transfers radar data over a TCP connection[3]. Given the pre-provisioning of OneNet bandwidth for IP1 use, congestion loss is not a concern in our initial testbed. However, we are currently developing a UDP-based transport protocol that uses application- specific selective dropping for congestion-control in bandwidth-constrained environments (Banka et al. (2007)). As illustrated in Figure 4, the arrival of per-radar reflectivity and wind velocity data for an elevation scan is posted and distributed among the MC&C procedures using the Linear Buffer (LB) pub/sub construct of the WDSSII software (Hondl (2003)).

3.2 Meteorological Feature Detection, Multi-radar Data Merging

Once the notifications of available per-elevation reflectivity and wind velocity data have been published in the Linear Buffer, various meteorological detection algorithms that subscribe to it can then read this data and perform feature detection. A detection is the extraction of a meteorological feature from single or merger radar data. This extraction is performed by detection algorithms like RT and SCIT. Figure 4 shows two such algorithms and their location in the overall MC&C architecture: The first one is Storm Cell Identification and Tracking (SCIT), which is a fixed reflectivity threshold algorithm that operates on merged data. The second one, reflectivity threshold (RT) operates on single-radar data and has a variable threshold. While SCIT produces a higher number of detections in areas of high and widespread reflectivity, RT will detect lower levels of reflectivity and generate more detections from small, low threshold areas of reflectivity. Other per-radar detection algorithms can be easily plugged in using the LB pub/sub mechanism. Once a detection algorithm completes its execution, notification is provided through the LB. The merge procedure converts polar coordinate data to latitude/longitude/elevation coordinates, and fuses together spatially overlapping data from multiple radars (Lashmanan et al. (2007)). Figure 5 (a) shows detection for both the RT and SCIT algorithms. These are detections generated from data that has been scanned during the present 60-second interval.

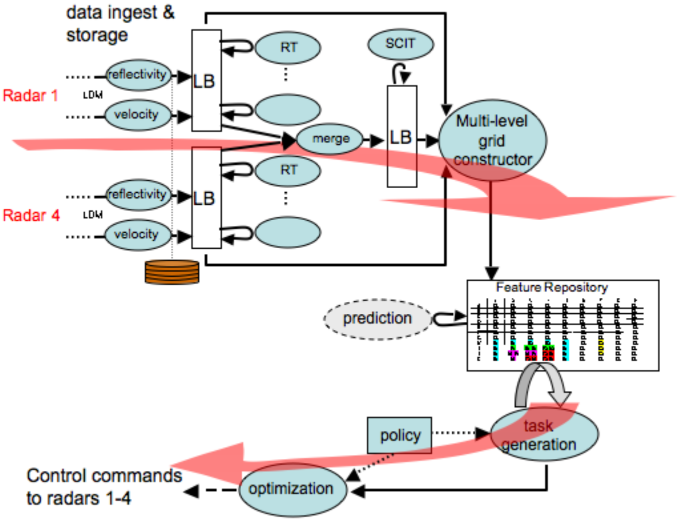

Figure 4: MC&C details of data ingest, detection algorithms, feature repository, and optimization

3.3 The Feature Repository

The IP1 system is a "real-time" system in the sense that radars must be re-tasked by the MC&C every 60 seconds, which defines the system heartbeat interval. This heartbeat interval was chosen based on the physical properties of the mechanically-steered radars, the time scale over which atmospheric conditions change, the system goal of detecting and tracking tornados within 60 seconds, and feedback from end-users after initial operations of the IP1 network. A notion of heartbeat also allows the radars to easily synchronize their operation (e.g., to scan the same volume in order to perform multi-Doppler wind field retrieval), and also helps simplify the optimization of radar targeting. As we evolve from mechanically-scanned radars to rapidly reconfigurable solid-state radars (phased array), we expect to relax the notion of a system heartbeat.

The radars are re-tasked based on detected meteorological features and the projected future evolution of these features in combination with end user policies. In order to decrease the timing dependencies between the ingest/processing of radar data and the generation of radar commands, the MC&C adopts a blackboard-like architecture [Jaganathan 1989]. At the heart of the MC&C is the feature repository, a multi-level grid that stores both the

[3] http://my.unidata.ucar.edu/content/software/ldm

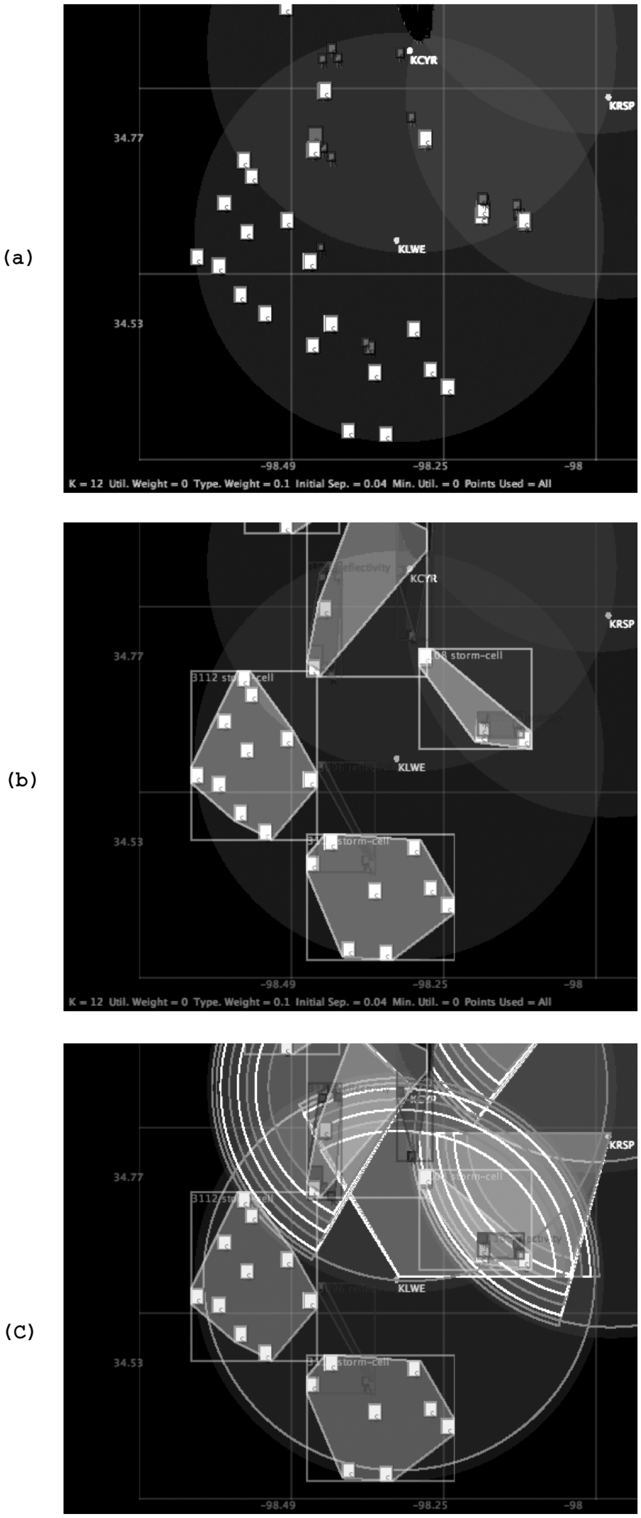

Figure 5: MC&C example of a weather event during the Spring Experiment. (a) shows the detections of the RT and SCIT algorithm, the resulting task are shown in (b), and (c) shows the scanning pattern for each radar. As shown, each radar does a pie shaped sector scan, the number of arcs on the edge denoting the number of elevation angles in the scan. This snapshot of the MC&C was taken from data on May 9th 2007 1:39:37 (UTC)

underlying per-voxel reflectivity and wind velocity data, as well as higher-level spatially-coherent meteorological objects such as storms cells, areas of high wind shear or precipitation, and tornados. Each feature board object has a position, a spatial extent for non-point objects, and a tag representing the meteorological phenomenon that the object represents (e.g., storm-cell, mesocyclone, and tornado). The multi-level grid-construction procedure writes this information into the feature repository as needed data becomes available via the linear buffer, as shown in Figure 4.

Another advantage of implementing the feature board as a blackboard architecture is that it allows the input processing of the data to be decoupled from the control of the sensors. In particular, the generation of radar commands (the lower half of the control loop in Figure 3) proceeds asynchronously from the input processing of data (the upper half of the control loop). In this decoupled architecture, detection algorithms continuously post their results to the feature repository. At 60 second intervals the task generation component posts a set of tasks based on current state of the feature repository, and the optimization component then processes this task set and generates a scan strategy for the radars for the next 60 second cycle. In this design, we have relieved the time pressure on the detection components and somewhat relieved the time pressure on the MC&C components, task generation and optimization. One consequence of this design is that data that is not processed and posted on the feature repository before the task generation begins will not be acted upon until the next cycle of the system. This allows the system to avoid stalling, while waiting for late-arriving data (e.g., due to unanticipated network and processing delays or even component failures like a radar node that going off-line). We are interested in the effects of this "decision lag" and also its relationship to the selection of the value of the system heartbeat.

We note here that the feature repository is the central system data structure. It is from here that meteorological objects can be obtained and subsequently delivered/displayed to end-users. Here, assimilated exogenous data (e.g., from satellite or from NEXRAD) can be stored and merged with data generated in the testbed.

3.4 Utility, Prediction and Task Generation

Resource allocation in IP1 is performed on the basis of a multi-user, multi-attribute utility approach. Within the feature repository, each task has such an associated utility (see Section 4 for the definition of utility) that represents the value of scanning that task during the next heartbeat. The utility value weights attributes such as the time since the task was last scanned, the object type (e.g., scanning an area with a tornado vortex will have higher utility than sensing clear air), and user-based attributes such as the distance from a population center (e.g., among two objects with identical features, the one closer to a population center will have higher utility). We are currently developing a predicting component (shown with dashed lines in Figure 1) that tracks meteorological phenomena (e.g., storm centroids) and predicts their future locations. New objects, corresponding to the predicted future locations of the phenomena, can then be added into the feature repository allowing these predicting modules to be easily integrated into the current architecture. The MC&C task-generation component takes objects from the feature repository and produces tasks, with an associated utility, for the optimization component. A task is basically a description that defines how one or multiple radars should scan a meteorological feature. For example, a task specifies that radar X should scan an area of high reflectivity with a 120° sector scan at 7 different elevations. End user needs are the basis for the utility values which are assigned at the task generation procedure. We use K-means clustering (Ramanathan et al. (2005)) to generate the tasks. The initial centroids of the clusters are chosen by sorting the objects by utility and using the K spatially-separated objects with the highest utility as our starting points. The task-generation algorithm uses task positional similarity between heartbeats during time-since-last-scanned calculation. Generic K-means clustering typically uses randomly seeded initial cluster centroids which often results in a lack of positional continuity for tasks between heartbeats. To achieve more consistency between cluster positions we select the initial centroids of the clusters using the position of tasks from the previous heartbeat. This simple pre-clustering step is designed to ensure good spatial coverage for our clusters. The K-means distance metric uses a 4-dimensional vector of parameters: an objects X, Y position, utility, and meteorological type. The relative weighting of these parameters can be adjusted to give differing emphasis to each parameter. After clustering is complete, a final filtering step removes tasks with utility below a given threshold. A result of this clustering process is shown in Figure 5 (b) where the polygons that enclose the detections define the tasks.

3.5 Optimization

The input to the radar targeting components is a list of tasks that can potentially be scanned and their associated utility. The optimization module determines, for each radar, the angular sector to be scanned (targeted) by that radar for the next 60-second cycle. The overall utility of a given configuration of the radars, will depend not only on the utility of the objects scanned, but also the size of the sector[4] and the number of radars targeting a given volume (since more radars illuminating a volume implies higher accuracy of measurement, especially for velocity). For our first testbed, a simple brute-force approach towards optimization is used that enumerates all possible configurations and computes the associated overall utility. In Section 5.3, we present measured execution times for this approach. Other, more scalable, approaches towards optimization are currently being investigated (Krainin et al. (2007)).

4 Utility Optimization

Our discussion above has focused on the overall CASA DCAS system architecture and the MC&C software architecture. We now turn our attention to the central aspect of DCAS control - determining *where* and *how* the radars should target their scans during a 60-second interval. This is the role of the task generation and optimization components of the MC&C and it is here that end-user needs have an impact on the overall system behavior.

In order to simplify our discussion below, we consider the simpler problem of determining only *where* the radars should focus their scanning, ignoring additional considerations such as the most appropriate radar waveform to be employed. This simple setting will nonetheless allow us to illustrate the key end-user related aspects of the MC&C.

We define a radar configuration to be the set of sectors to be scanned by the radars during a 60-second interval. Each radar will be assigned to scan one sector (defined by start and stop angle and number of elevations). The MC&C takes a utility-based approach towards determining the optimal radar configuration for a particular heartbeat, where (as we will see) utility is directly defined in terms of end-user requirements. Two sets of factors contribute to the utility of a particular radar configuration. The first set of factors is concerned with *how well* a particular portion of the atmosphere is sensed by a given radar configuration. The second set of factors is concerned with *how important* the scanned sectors are to the end-users. The overall utility of a particular configuration combines these two considerations of *how well* and *how important*.

4.1 Scan Quality

Before specifying how utility is computed, let us first qualitatively discuss the factors that will come into consideration. The first set of factors are related to the quality of the data from the scanned sectors (*how well*):

- *Scan sector size.* Each radar can be tasked to scan a sector with an arbitrary starting point and at any width from 60° to 360° during the 60-second heartbeat. In the current implementation of the system the scan rate (angular velocity) of each radar is kept constant. This results in a trade-off between sector width and number of elevations that can be scanned. For example, a scan in a 60-second heartbeat could be defined as follows. A standard 360° scan for the first 20 seconds at 2° elevation followed by 60° sector scans for elevations 1°, 3°, and 5°. In future work, we plan on enhancing the system by a scanning policy that would also allow for changes in angular velocity. That would allow for longer dwell times and more ac-

[4] Currently, there is a trade off between sector width and elevations that can be scanned in a 60 second interval. E.g., a radar can perform two 360° scans or 8, 90° sector scans at 8 different elevations.

curate estimates of the sensed values due to increased sensitivity (Donovan and McLaughlin (2005)).

- *Coordination among radars.* It is often advantageous to have two or more radars focus their scans on overlapping regions in the atmosphere, a so-called Dual-Doppler region. One radar may be able to see that portion of the atmosphere better (e.g., due to less signal attenuation), and data from multiple radars can provide more accurate estimation of wind velocity vectors. Another benefit of radar coordination arises even when each meteorological feature is scanned by a single radar when a particularly high utility feature can be scanned by more than one radar, the system can scan that feature with the radar that allows remaining radars to scan other meteorological features of next highest utility.

4.2 End User Utility

The second set of factors is concerned with *how important* the scanned portions of the atmosphere are to the end-users:

- *Expressing the preferences of multiple end-users.* The initial end-users in our Oklahoma testbed are (i) the National Weather Service Forecast Office (NWS-FO) in Norman, OK, (ii) a group of emergency managers who have jurisdictional authority within and upstream of the test bed area, and (iii) CASA's science researchers themselves. Different end-users will derive different utility from a particular radar configuration. For example, researchers interested in tornado genesis may be more interested in wind events regardless of where they occur, while emergency response managers may be more interested in area of high reflectivity over particular areas of interest; emergency response managers may be mostly interested in the lower elevation scans, while NWS forecasters and researchers may be interested in full volume scans that include higher elevations. End users must be able to express their relative preferences among different radar configurations, and these preferences must then be incorporated into the overall system behavior. Our DCAS system summarizes end-user needs as rules which on one hand can be understood in natural language, and on the other directly translated into quantitative utility functions (Philips et al. (2007)).

- *Policy: mediating among the conflicting scanning requests of different end users.* If a given radar configuration is of particularly high utility to one group of users but of lower utility to another group of users, while a second configuration is of low utility to the first group of users but of high utility to the second group of users, the system must decide which of these two alternatives is preferable. That is, a policy mechanism must be defined for mediating among the conflicting demands of different end-users. Optimization of overall utility results into a scan configuration with the best trade-off.

4.3 MC&C Equation

The above considerations give rise to the following optimization problem (which we refer to as the MC&C equation) that must be solved by the MC&C:

$$J = \max_{configurations, C} \sum_{tasks, t} U(t,k)Q(t,C) \qquad (1)$$

where:

- t is the set of so-called tasks, i.e., meteorological features that are within the IP1 radars coverage areas and thus can potentially be scanned at time interval k. Examples of tasks include areas of wind rotation, areas of high reflectivity, and areas of wind shear. These tasks are created by the task-generation, as described in Section 3.4. An example for tasks that have been created by this MC&C module are shown in Figure 5(b).

- $U(t,k)$ is the aggregate end-user utility of task t to the set of end-users at heartbeat k, and captures the *how important* aspect of utility. $U(t,k)$ in turn is defined as:

$$U(t,k) = \sum_{groups, g} w_g U_g(t,k) \qquad (2)$$

  where g is the set of user groups, $U_g(t,k)$ is the utility of task t at heartbeat k to user group g, and w_g is the weight associated with user group g. Below, we discuss how $U_g(t,k)$ is computed. The user group weight w_g is a value between 0 and 1, reflecting the relative priority of user group g with respect to other user groups. The values of w_g are set by high-level system user policy. Note that the architecture itself is policy-neutral in that it does not prescribe values for w_g but instead provides a mechanism for weighting the relative importance of different user groups. In the future, w_g will be set by meteorological context. For example, in the case of a hazardous weather event NWS forecasters priority will be increased over, e.g, the researchers priority.

- $Q(t,C)$ is the expected quality if task t is scanned under radar configuration C, capturing the *how well* aspect of utility. $Q(t,C)$ is a value between 0 and 1, with a value of 1 representing the highest quality possible. The details of the computation of $Q(t,C)$ can be found in Pepyne et al. (2007). We note here that for a particular task, t, the value of $Q(t,C)$ *(i)* is zero unless the entire task is covered by the sector *(ii)* increases with decreasing sector width (smaller sectors imply that more elevations can be scanned), and *(iii)* increases as the physical location of the tasks becomes closer to the radar. (For rotation and velocity tasks $Q(t,C)$ also increases with the number of radars covering the task.)

Several comments are in order regarding the MC&C equation. First, note that the form of the MC&C equation involves a sum over all tasks, where the utility of each task has a quality component, $Q(t, C)$ (reflecting how well the task is sensed by the radars in a particular configuration) and an end-user utility component, $U(t, k)$ (reflecting how important that task is to the end-users). The overall utility of a given task is the product of these two components. An alternative to taking the product of these two components would have been to allow end-user utility itself to depend on the quality component, i.e., to define task utility under configuration C in the form $U(t, k, Q(t, C))$, rather than $U(t, k)Q(t, C)$ as in Equation 1. This alternative formulation would have resulted in a significantly more complex optimization problem, since end-user utility would have been a function of the scan quality of the configuration.

Our separation of how well and how important into two independent considerations represents an architectural decision to separate lower level, radar-specific, sensing considerations from higher-level end-user considerations. This also illustrates one way in which we have avoided building a stovepipe system. If a new set of sensors[5], with new operating characteristics were to be used, we need only change $Q(t, C)$. On the other hand, with a task utility of the form $U(t, k, Q(t, C))$, we would have to redefine end-user utility as well, a difficult and time-consuming task that requires numerous interactions with the end-users, described in Philips et al. (2007). We emphasize, however, that both task quality and task importance are taken into account in the end-user utility calculation (through their product).

Second, we note that while it appears that the optimization is considering each 60-second heartbeat as an independent optimization problem, this is actually not the case. The system does maintain some history of the tasks it has scanned in the past. Users have specified not only what tasks they want scanned (Pepyne et al. (2007), Philips et al. (2007)), but also how often different types of tasks need to be scanned. A typical end-user rule has the form: If a meteorological phenomena of type X is detected, then scan it with Y radars (when possible) at least once every Z heartbeats. To implement this capability we keep track of the time-since-last-scanned for each task. If the time-since-last-scanned is less than the user-defined interval between scans (Z heartbeats in the previous sentence), then the task has a base utility value. If the time-since-last-scanned is greater than the user-defined interval, we scale (increase) the utility value $U_g(t, k)$ of task t at each heartbeat until the task is eventually scanned. At this point, the task utility value is reset to its base value, and the scaling process begins again. With this simple scheme, we implicitly solve what is otherwise a complex multistage optimization problem. A prediction and tracking approach that could be integrated into the MC&C is presented by Manfredi et al. (2005).

5 2007 Spring Experiment and Results

5.1 Description and Goals

The overarching goal of this experiment was to demonstrate the value-added by a DCAS system relative to the state-of-the-art (sit and spin, long-range, open-loop, non-networked radars) and its impact on the severe weather warning processes. To achieve this goal it had to be shown that the IP1 system is able to satisfy the needs of multiple end-users by demonstrating closed-loop detections and adaptive scanning in real-time. From the perspective of the MC&C software architecture that means it had to be demonstrated that the system has the capability to provide real-time data processing and optimization for the control of the radars. In addition, it had to be shown that meteorological data observed by a DCAS system, which is of high temporal and spatial resolution, provides the capability to observe storm morphological features not routinely observed by the current operational systems like NEXRAD. In the following an overview of the experiment will be given before results of a performance evaluation of the MC&C during this experiment will be presented.

The Spring Experiment lasted for the duration of a total of 9 weeks (from April 9th to June 10th). According to Oklahoma climatology (Brotzge (2006)), the majority of tornados (85%) occur in that time period. During weather events the DCAS network was operated 24/7 in closed-loop mode as described in Section 2.

In addition to the members of the CASA team, operational end-users were part of this experiment. These operational end-users were members of the Hazardous Weather Testbed[6] forecast team.

5.2 Meteorological Events

In the following, we give a short description of two major weather events that occurred in the testbed during the Spring Experiment. In Section 5.3, we present an evaluation of the MC&C for both of these events. IP1 Radar Data (2007) contains movies that show the radar data (reflectivity) and the scanning behavior of the radars.

5.2.1 April, 24^{th} 2007

Elevated, non-severe convection occurred on the morning of April 24 from 14-16 UTC. Several isolated cells developed just south and west of the testbed, and moved NE directly through the center of the testbed. Complete 360° scans were collected at an elevation of 2.0°, and sector scans were collected at elevations of 1, 3, 5, 7, 9, 11, and 14 degrees (where the actual number of elevations depends on the sector size).

A Tornado Watch was issued in anticipation of a significant severe weather outbreak. A severe squall line developed just west of the testbed, and moved through the testbed from 17-20 UTC. Large hail was prevalent along

[5] This is not limited to radars, since the architecture can accommodate heterogeneous sensors

[6] http://www.nssl.noaa.gov/hwt/

the line. Data were collected at three sites (KSAO, KCYR, and KLWE) in DCAS mode using the 60-second heartbeat.

5.2.2 May, 8th and 9th 2007

A large area of convection with embedded supercells moved north from Texas into the testbed. Several areas of significant circulation were detected by KLWE and KCYR between 00 UTC and 03 UTC (May 9). A tornado was reported north of the KSAO radar around 11PM. Several circulation signatures appeared in the KSAO radar data in the same time period. Data were collected with all 4 radars in DCAS mode with scanning patterns as describes in 5.2.1

using the 60 second heartbeat. Complete 360° scans were collected at 2.0°, and sector scans were collected at 1, 3, 5, 7, 9, 11, and 14 degrees, depending on the sector size.

5.3 Experiment Results

In this section, we analyze the performance of the MC&C during the two meteorological events described in Section 5. The goal of this analysis is to show the extent to which the system, driven by the MC&C, is able to satisfy the needs of the end-users. To evaluate the satisfaction of the end-user needs the following metrics are used:

- *Tasks satisfied/unsatisfied*: As described in Sections 3 and 4 the generation of the tasks and the final scanning configuration for the radars is influenced by the end-user utility. Through this utility the different end user groups can express their preferences in terms of scanning configuration. To measure how well these preference are met we introduce the *tasks satisfied/unsatisfied* metric. This metric determines how many of the tasks that were created for one heartbeat were actually scanned by the radars in that heartbeat. Tasks for which $Q(k,t) >= 0.6 * Q_{max}(t)$ is considered *satisfied*. Otherwise it is considered *unsatisfied*. Note that a task that is determined to be unsatisfied does not have to stay that way; it can very well be that this task is satisfied in one of the following heartbeats (recall a task not satisfied at one heartbeat has its utility increased to raise the chance it will be scanned during subsequent heartbeats).

- *Processing Time*. Another critical factor in a closed-loop system as the IP1 radar network is the processing time of the MC&C components. To allow a timely operation of the system it has to be verified that the processing times of the individual components allow a timely operation of the control loop. In the specific case of IP1 is has to be assured that new commands to radars are issued every 60 seconds. As processing time, we measure the duration it takes an MC&C module (e.g., task generation and optimization) to finish its processing for a single heartbeat.

5.3.1 April 24th 2007

The results of the MC&C performance analysis for the April 24th case are presented in this section. First, the results for the *tasks satisfied/unsatisfied* metric are presented. Figure 6 shows the results for cumulative overall number of satisfied and unsatisfied tasks, while the individual numbers for the RT and SCIT detections are shown in Figure 6 (b) and (c), respectively. The variation in the gradient for the slopes representing the metric is caused by meteorological activity in the coverage area of the network. In fact, the steepness of the slope indicates the intensity of the weather, i.e., higher intensity implies more tasks which implies a steeper slope.

It is obvious that in cases of less meteorological activity the average number of newly created tasks will be lower. This can also be matched to the total number of tasks in the system at each heartbeat, as shown in Figure 7 (a). Figures 6 (b) and (c) show a similar behavior for tasks created individually by the RT or the SCIT detection algorithm. The major difference between the results shown in Figures 6 is the fact that during some periods (e.g., between hours 5 and 10) many RT-based tasks are created while almost no SCIT-based tasks are created. This behavior is caused by the fact that the RT algorithm has a relative threshold while the threshold for the SCIT algorithm is fixed. Thus, RT will generate more detections during initial storm formation, while SCIT will generate more detections once a storm has matured.

Only SCIT-based tasks stay unsatisfied in the April 24^{t}h case because of the fact that the SCIT algorithm creates tasks for multiple radars if the meteorological feature is located in a multiple-Doppler area. A SCIT-based task can require that multiple radars scan a meteorological feature simultaneously. Due to resource conflicts this cannot always be achieved and results in an unsatisfied tasks for that heartbeat. In many cases this tasks will be satisfied in one of the subsequent heartbeats.

In addition, we also analyzed the time between scans for a specific task. To measure this interval we register when a task is initially scanned and log the time at which this task is scanned again. Results from this analysis are shown in Figure 8. This analysis show that the system outperforms the specified goals for interscan periods. The user requirements mandate that RT-based tasks are scanned every 120 minutes; from Figure 8, we see that they are actually scanned every 60 seconds. In the SCIT-based tasks case we measured an interscan time of 78.46 seconds, while the required interval is 180 seconds.

Figure 9 shows the results for the *processing time* metric for the optimization and task generation modules of the MC&C. The results show that with a moderate number of tasks the run time for both modules are well below one second allowing the MC&C control loop to operate in a timely manner.

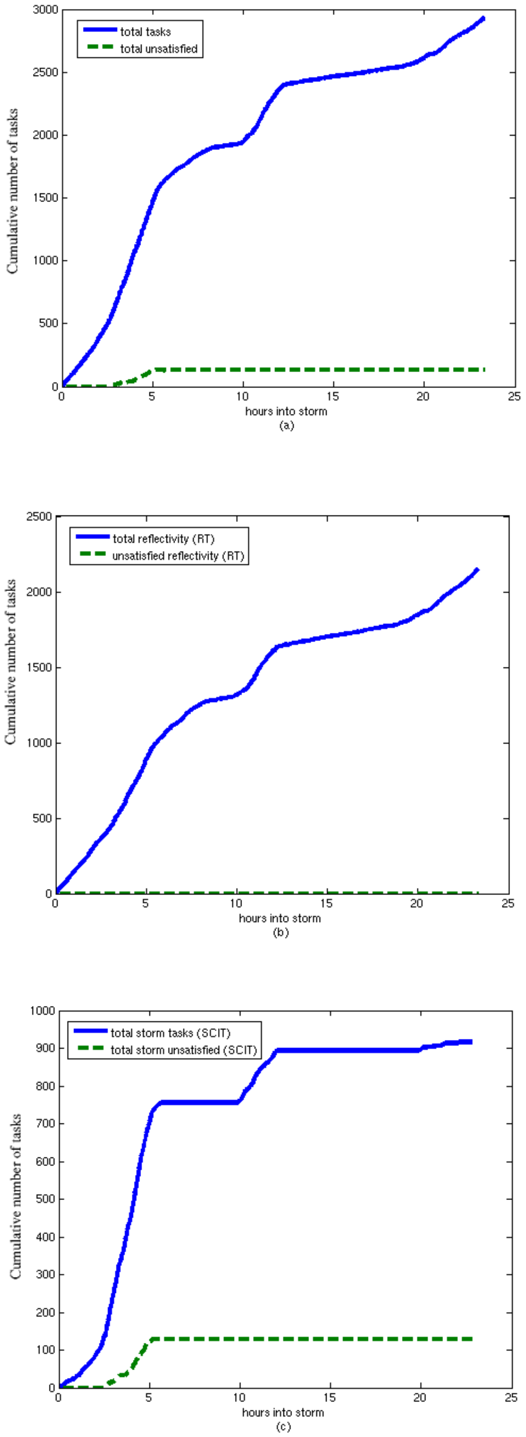

Figure 6: Cumulative number of tasks satisfied and unsatisfied during the April 24th event. (a) shows the total tasks generated by both the RT and SCIT detection algorithms. (b) shows the results for the RT-based tasks and (c) for the SCIT-based tasks.

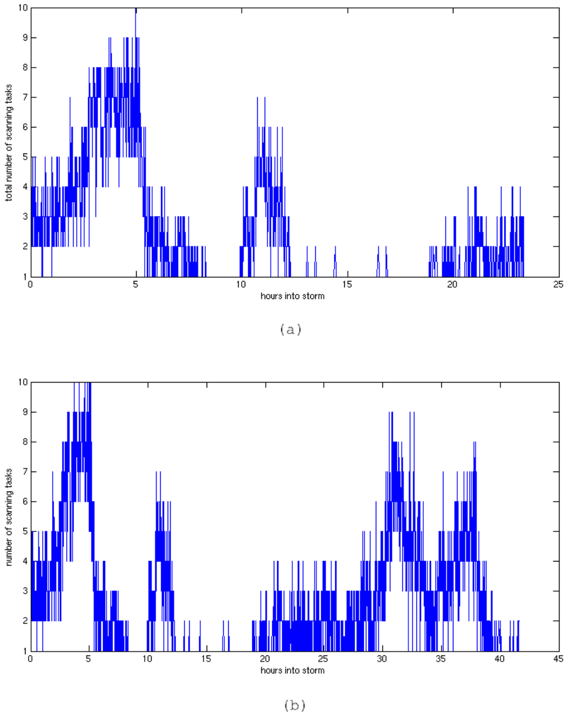

Figure 7: Total number of tasks in the system at each heartbeat the April 24th (a) and May 8th (b) event. A larger number of tasks reflects an increase in storm intensity.

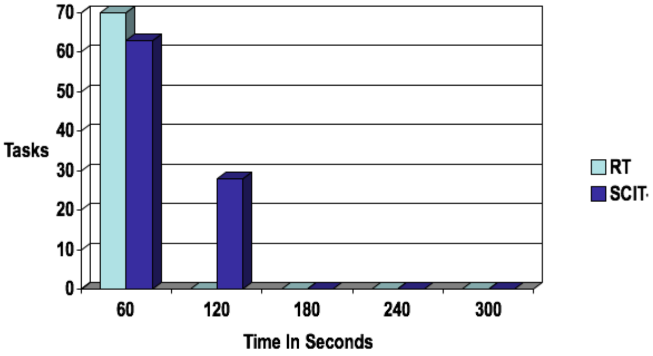

Figure 8: Time between tasks for the April 24th event. RT avg. interscan period = 60.0 (required period is 120 seconds). SCIT avg. interscan period = 78.46 (required period is 180 seconds).

5.3.2 May 8th 2007

In this section, we evaluate the performance of the MC&C for the May 8th case. Figure 10 shows the results for the *tasks satisfied/unsatisfied* metric. The overall results for this metric are shown in Figure 10 (a) while the individual results for the RT and SCIT detections are shown in Figure 10 (b) and (c), respectively. The results of this evaluation show that 97% of the generated tasks were satisfied

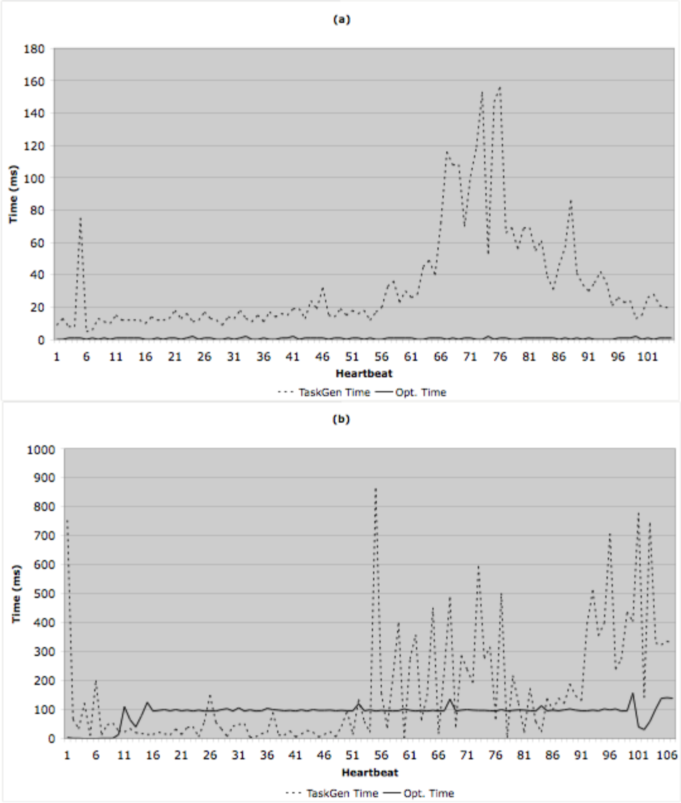

Figure 9: Processing times of the task generation and optimization modules for the April 24th (a) and May 8th and 9th (b) event.

(scanned by the radar) immediately. Only 3% of the generated tasks were not immediately satisfied. From these results we conjecture that the system satisfied the needs of the end-users in relation to the data it provided during the event. The number of total tasks per each heartbeat throughout the event is shown in Figure 7 (b). This figure reflects the evolution of the storm's intensity, which is shown by an increase in total tasks per heartbeat.

Results for the time between scans for this event are shown in Figure 11. As it is the case for the April 24th event the system performs better than the requirements since the measured interscan period is lower than the required one.

Results for the *processing time* metric are shown in Figure 9 (b). Here, we show results for the task generation and the optimization module. Compared to the results for the April 24th case (Figure 9 (a)), the processing times for both components are significantly higher. This is due to the fact that the average number of tasks in the May 8th case is higher than in the April 24th case. The increased number of tasks is caused by a higher number of isolated storm cells. The results show that the processing time for both is still well below one second, and this means, the control loop of the IP1 system also worked in a timely manner

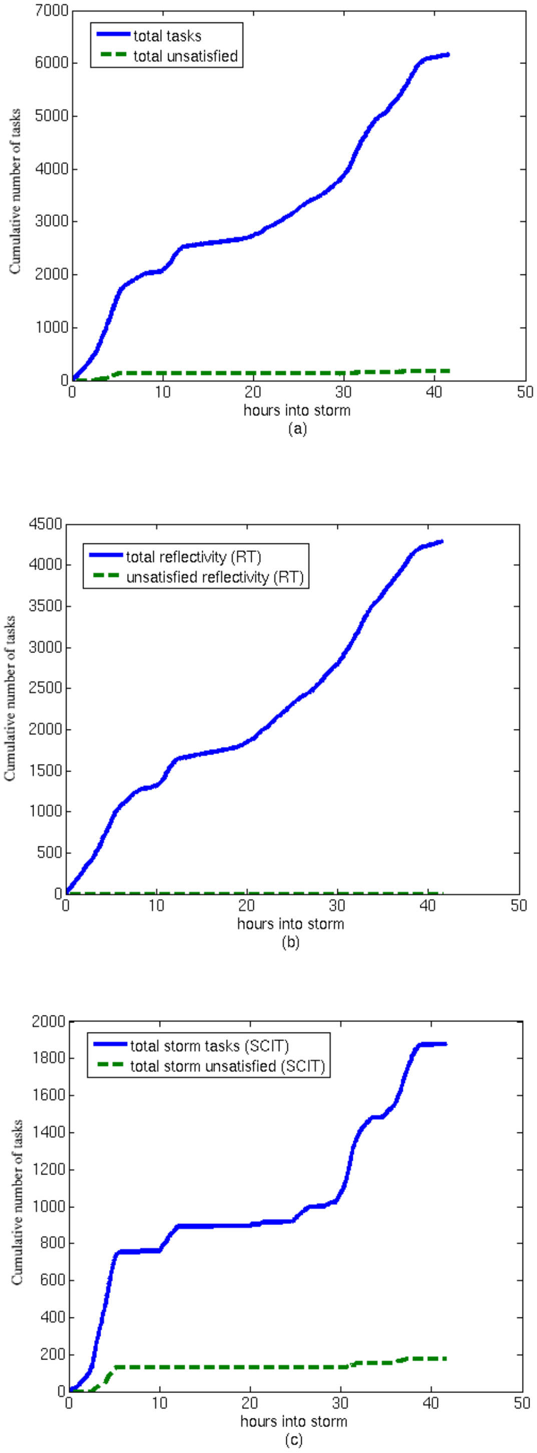

Figure 10: Cumulative number of tasks satisfied and unsatisfied during the May 8th and 9th event. (a) shows the total tasks generated by both the RT and SCIT detection algorithms. (b) shows the results for the RT-based tasks and (c) for the SCIT-based tasks.

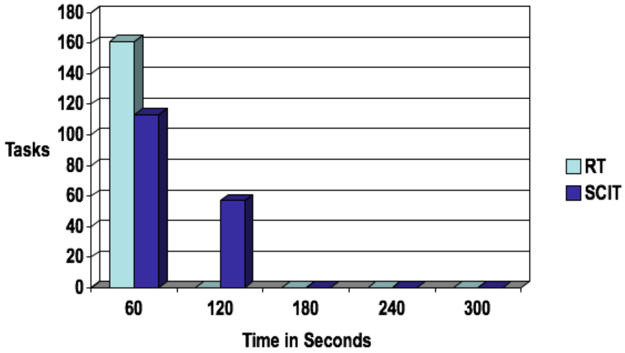

Figure 11: Time between tasks for the May 8^{th} and 9^{th} event. RT avg. interscan period = 60.0 (required period is 120 seconds). SCIT avg. interscan period = 80.12 (required period is 180 seconds).

during the May 8^{th} event.

6 Conclusion and Future Work

The CASA Engineering Research Center has designed, built, and operated a 4 radar DCAS prototype testbed in southwestern Oklahoma. This testbed has been put in full operational mode, with researchers and end-users using its data in real-time, during the spring 2007 experiment. One of the major components of the DCAS architecture is the control loop that is responsible for control and thus adaptive, end-user needs based scanning of the radars.

In this article, we present the essential part of this control loop, the MC&C in detail. The MC&C is essentially build as a closed-loop system that operates in soft real-time. A black board architecture is used to decouple timing dependencies between data ingest and and the generation of radar control commands. We also analyze the performance of the MC&C during two significant severe weather events in the spring of 2007. The results from this analysis reveal that we have designed and built a system that scans the atmosphere according to the need of the end-users. This is achieved by a control loop that performs real-time data processing and an optimization that determines the scanning of the radars based on a utility function.

Besides the numerical values that were obtained through the analysis of the MC&C performance we also got direct feedback from the actual end-users stating that the IP1 system provided value beyond the current state of the art.

We have recently started the design and implementation of a distributed MC&C architecture. This approach is based on a hierarchical resource-allocation mechanism. Initial results on a simulation-based investigation are reported in Krainin et al. (2007) and we recently started running experiment with this distributed architecture in the IP1 testbed.

ACKNOWLEDGMENT

This work is supported primarily by the Engineering Research Centers Program of the National Science Foundation under NSF award number 0313747. Any opinions, findings, conclusions, or recommendations expressed in this material are those of the authors and do not reflect those of the National Science Foundation.

REFERENCES

Banka, T., Lee, P. H., Jayasumana, A., and Kurose, J. (2007). An Architecture and a Programming Interface for Application-Aware Data Dissemination Using Overlay Networks. In *Proceedings of the 2nd Intl. Conference on Communication Systems Software and Middleware (COMSWARE), Bangalore, India.*

Bharadwaj, N. and Chandrasekar, V. (2005). Waveform Design for CASA X-Band Radars. In *Proceedings of AMS Radar Meteorology 2005, Abuquerque, NM, USA.*

Bringi, V. and Chandrasekar, V. (2001). *Polarimetric Doppler Weather Radar*. Cambridge University Press, 1st edition.

Brotzge, J. (2006). Severe weather climatology of ip1. Technical Report CASA Technical Report, Oklahoma University.

Donovan, B. and McLaughlin, D. (2005). Improved Radar Sensitivity Through Limited Sector Scanning: The DCAS Approach. In *Proceedings of AMS Radar Meteorology 2005, Abuquerque, NM, USA.*

Hondl, K. (2003). Capabilities and Components of the Warning Decision Support System - Integrated Information (WDSS-II). In *Proceedings of American Meteorological Society Annual Meeting, Long Beach, CA, USA.*

IP1 Radar Data (2007). http://server.casa.umass.edu/movies/ip1.

Junyent, F. (2007). *Networked Weather Radar System Using Coherent-On-Receive Technology*. PhD thesis, University of Massachusetts Amherst.

Junyent, F. and Chrandrasekar, V. (2007). Radar Network Characterization. In *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium, Barcelona, Spain.*

Khasgiwale, R., Krnan, L., Perinkulam, A., and Tessier, R. (2005). Reconfigurable Data Acquisition System for Weather Radar Applications. In *Proceedings of IEEE Midwest Symposium on Circuits and Systems, Cincinati, OH, USA*, pages 484–488.

Krainin, M., An, B., and Lesser, V. (2007). An Application of Automated Negotiation to Distributed Task Allocation. In *Proceedings of International Conference on Intelligent Agent Technology (IAT07), Silicon Valley, CA, USA*.

Kurose, J., Lyons, E., McLaughlin, D., Pepyne, D., Philips, B., Westbrook, D., and Zink, M. (2006). An End-User-Responsive Sensor Network Architecture for Hazardous Weather Detection, Prediction, and Response. In *Proceedings of Asian Internet Conference (AINTEC), Pathumthani, Thailand*.

Lashmanan, V., Smith, T., Stumpf, G., and Hondl, K. (2007). A real-time, three-dimensional, rapidly updating, heterogenous radar meger technique for reflectivity, velocity, and derived products. *Weather and Forecasting*, 22(3):596–612.

Lu, Y., Wang, Y., Willie, D., Chandrasekar, V., and Bringi, V. N. (2007). Operational Evaluation of the Real-Time Attenuation Correction System for CASA IP1 Testbed. In *Proceedings of the 33rd AMS Conference on Radar Meteorology, Cairns, Australia*.

Manfredi, V., Mahadevan, S., and Kurose, J. (2005). Switching Kalman Filters for Prediction and Tracking in an Adaptive Meteorological Sensing Network. In *Proceedings of the IEEE SECON conference, Santa Clara, CA, USA*.

McLaughlin, D., Chandrasekar, V., Droegemeier, K., Frasier, S., Kurose, J., Junyent, F., Philips, B., Cruz-Pol, S., and Colom, J. (2005). Distributive Collaborative Adaptive Sensing (DCAS) for Improved Detection, Understanding, and Predicting of Atmospheric Hazards. In *Proceedings of the 9th Symp. Integrated Obs. Assim. Systems - Atmos. Oceans, Land Surface (IOAS-AOLS), Amer. Meteorological Society, San Diego, CA, USA*.

NEXRAD (2007). http://www.roc.noaa.gov.

Pepyne, D., Westbrook, D., Philips, B., Lyons, E., Zink, M., and Kurose, J. (2007). A design for distributed collaborative adaptive sensing of the atmosphere. Technical Report CMPSCI-2007-018, University of Massachusetts.

Philips, B., Pepyne, D., Westbrook, D., Bass, E., Brotzge, J., Diaz, W., Kloesel, K., Kurose, J., McLaughlin, D., Rodriguez, H., and Zink, M. (2007). Integrating End User Needs into System Design and Operation: The Center for Collaborative Adaptive Sensing of the Atmosphere (CASA). In *Proceedings of 16th Conf. Applied Climatol., American Meteorological Society Annual Meeting, San Antonio, TX, USA*.

Ramanathan, R., Redi, J., Santivanez, C., Wiggins, D., and Polit, S. (2005). Ad Hoc Networking With Directional Antennas: A Complete System Solution. *IEEE Journal on Selected Areas in Communications*, 23(3):496–506.

Zink, M., Westbrook, D., Abdallah, S., Horling, B., Lakamraju, V., Lyons, E., Manfredi, V., Kurose, J., and Hondl, K. (2005). Meteorological Command and Control: An End-to-End Architecture for a Hazardous Weather Detection Sensor Network. In *Workshop on End-to-End, Sense-and-Respond Systems, Applications, and Services, Seattle, WA, USA*.