Cognitive Adaptive Radio Teams

Richard Lau, Stephanie Demers, Yibei Ling, Bruce Siegell (Telcordia Technologies, Inc.)
Einar Vollset, Ken Birman, Robbert vanRenesse, (Cornell University)
Howie Shrobe, Jonathan Bachrach (MIT CS & AI Laboratory)
Lester Foster (MultiSpectral Solutions, Inc.)

Abstract -- Cognitive Adaptive Radio Teams (CART) is a new platform developed by our group[1] in support of collaborative mapping of complex communications-challenged environments, for example in support of search and rescue operations in environments lacking an adequate communications infrastructure. Experience during the 9/11 terrorist attacks, Asian Tsunami, Kashmir Earthquake, and post-Katrina Gulf Coast make it clear that rescue workers cannot count upon computer networks or even cell telephone support in the immediate aftermath of such events. Similarly, military urban warfare operations must also be conducted in locations lacking communication infrastructure. CART combines state-of-the-art ad-hoc networking technology with machine learning and prediction algorithms to offer new capabilities under these very difficult conditions.

I. INTRODUCTION

First responders confronting a terrorist attack, a fire, or an environmental disaster must coordinate their actions despite a severely degraded communications environment. Consider the kind of search-and-rescue operation mounted in the wake of Hurricane Katrina. The Gulf Coast was flooded, trapping residents in isolated regions. The computer communications infrastructure was destroyed, and cellular telephone capabilities were erratic. The region was large hence most estimates for restoring communications were of weeks or months, and satellites could only offer sporadic connectivity and limited coverage. Yet poor communication can kill: In the case of the 9/11 terrorist attacks on the World Trade Center, studies revealed a pattern of communication breakdowns and showed that better technology could have saved lives. Soldiers engaged in urban warfare confront a similar problem. They, too, must cooperate, despite limited opportunity to communicate to any sort of base station or infrastructure network.

Today, many of us carry off-the-shelf portable devices that have been designed under the assumption that infrastructure support is common and ubiquitous. In the target settings for our effort, such devices become nearly useless: one can pre-load them with maps or other fixed data, but opportunities to upload revisions to infrastructure services or to download updated information are extremely rare.

The CART is a new platform designed specifically for use in such settings. It is being developed under the DARPA ACERT program [1] and exploits fundamental advances in cognitive systems, distributed computing, learning, and cross-layer optimization to overcome adverse wireless conditions. The design goals for our solution include efficient control of radio platform resources, dynamic teaming, secure distribution of information, and collaborative optimization of channel characteristics. In contrast to much of the prior work in this area, we focus on cooperation between nodes searching some region over extended periods of time. Moreover, power conservation is not a primary concern: we assume that searchers would be able to recharge their devices every day or two.

In summary, the contributions of our effort are:

1. Explore the benefits of teaming on achieving common goals.
2. Demonstration of cognitive capabilities to increase efficiency and accuracy of searches.
3. Security architecture for mobile collaboration.
4. Cognition-driven cross-layer optimization.

In the remainder of this paper we offer additional details on the CART architecture. At the time of this writing, we were just beginning to integrate system components with the goal of experimenting on a first version of the system late in 2006.

II. CART ENVIRONMENT

---

[1] This material is based upon work supported by the Defense Advanced Research Projects Agency (DARPA) under contract NBCHC050161. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the Defense Advanced Research Projects Agency (DARPA); or its contracting agent the US Department of the Interior, National Business Center, Acquisition and Property Management Division, Southwest Branch.

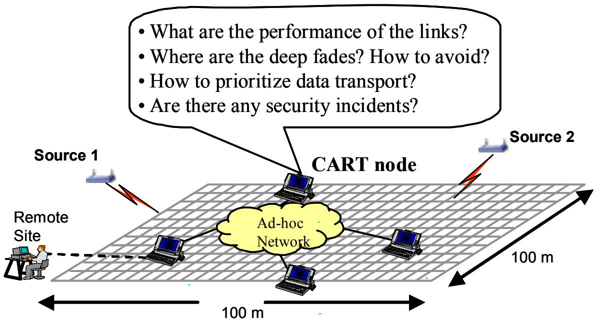

Figure 1 CART Environment

Figure 1 illustrates a simple application of CART, in which the system is used to search a 100m × 100m grid. Each searcher traverses a separate path in a designated time and captures various types of data, including the measured SNR, precise location, radio link performance, and potentially others (e.g. images). Goals in this case include disseminating all of this data within the full set of searcher nodes, and also transmitting the aggregated findings to a remote location.

III. ARCHITECTURAL OVERVIEW

The architecture design starts with hardware choices that maximize connectivity and information acquisition options. The design is oriented towards teaming applications that require nodes making contact with each other using ad-hoc communication to share information.

Figure 2 illustrates the overall architecture of the CART platform, focusing on a typical node. Location awareness is provided by a combination of sensor technologies including GPS receivers, and a ranging system based on Ultra-Wide Band. Connectivity among peered nodes is initially based on 802.11 systems but will later migrate to a software defined radio platform. Immediately above these is a layer that can be understood as the "device drivers" for the radios and other hardware components; these acquire data from the devices and present it to higher level applications, while also standardizing interfaces for parameter setting and other control of the hardware modules. An environment-aware enhanced MAC (EMAC) layer augments the lower level functions with flexible control logic that can implement new MAC-layer protocols to optimize performance.

Moving up to higher levels of abstractions, the Group Communications System (GCS) is a communication component that functions much like a store-and-forward router: messages ("objects") are handed to the CART Object Sore (COS) with security policies and destination information, and COS uses the lower-level hardware to move the data about. The Cognitive Controller (CC) is the "intelligence" of the CART node. It provides overall awareness and uses predictive information to make decisions to prioritize communications and on the choice of transport mode and wireless parameters. A key information source for CC is from the R*adio Environment Modeling Application*, or REMA. This module specialized in understanding the current radio environment and provides predictions of radio performance in the near future.

Finally, the *signal noise ratio (SNR) mapping application* is a "stand in" for a large class of applications that acquire some form of sensor data and use it to annotate "maps". In the future, we plan to support applications that track, for example, locations of individuals in need of medical assistance were found, concentrations of toxic chemicals or other contaminants, hazards (including, in the case of urban warfare applications, locations of possible bombs or other threats), and so forth. At the present stage of development we opted for a simple and convenient target: our SNR mapper just tracks the measured signal-to-noise ratio for a designated 802.11 channel (an idle channel, not used for communication). The initial demonstration of CART will involve forming SNR maps in various settings: a building, an outdoors picnic area, and so forth.

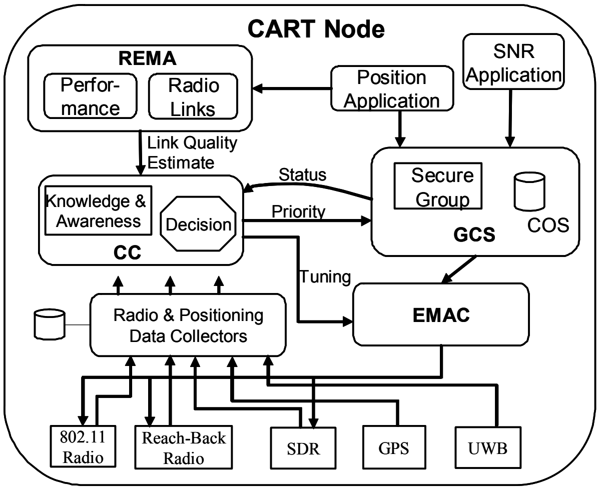

Figure 2 CART Architecture

IV. PREDICTION OF THE WIRELESS ENVIRONMENT (REMA)

The role of the Radio Environment Modeling Application (REMA) is to predict connectivity and communication performance so that the CC can prioritize and optimize communication. The CC needs this information so that it can avoid sending data that cannot be received by its intended recipients and also so that it can take advantage of the broadcast nature of wireless communication to maximize the impact of each transmitted message.

REMA on each node builds a model of the radio communication environment based on communication measurements taken locally as well as measurements shared by the other nodes using the GCS. The communication measurements include SNR and signal attenuation measurements between pairs of nodes as well as performance measurements (achieved throughput) between nodes. From this information, REMA characterizes the radio environment so that it can provide predictions of communication quality between points that have not yet been traversed by the nodes.

Based on node trajectory information provided by the CC, REMA uses its model to make predictions of radio communication capabilities for the next few (e.g., 5-10) seconds. To obtain a prediction of communication performance, the CC provides REMA with a prediction of the locations or regions where two radio nodes will be in the near future. Based on probabilities associated with the locations, REMA computes a weighted average of predicted radio performance and determines whether the two nodes will be able to communicate during the time period associated with the trajectories, for how long, and at what quality/performance.

REMA will also provide measures of confidence along with its predictions. At the beginning of the experiment, there is little data to base predictions on, so confidence will be very low; in this case, predictions might only be based on the most recent few measurements. As the radios move around the grid and communicate with each other, REMA's database of measurements grows; more advanced prediction approaches can be used, and predictions can be given with higher confidence.

V. COGNITIVE CONTROLLER

The CC is the software module that provides for learning and adaptation within each CART node. The CC combines information from all of the other CART modules in order to model the communications environment the trajectories being followed by each CART node and the knowledge state of each of the nodes within the systems. It uses this information to choose the best actions for the CART system to take including 1) what information to transmit (and to whom), and 2) how to configure the communications protocol stack and the physical hardware in order to optimize communications.

The CC accomplishes its tasks using artificial intelligence techniques for planning under uncertainty. It looks ahead in time, using REMA to estimate when there will be opportunities to communicate with other nodes and at with what level of link quality. It also makes estimates about which nodes are likely to know what information. In some cases, this knowledge is absolute, the CC may know that it had already successfully transmitted some information to a peer node; but in other cases the knowledge will be probabilistic based on estimates that two other nodes had some probability of communicating with one another and of exchanging specific information.

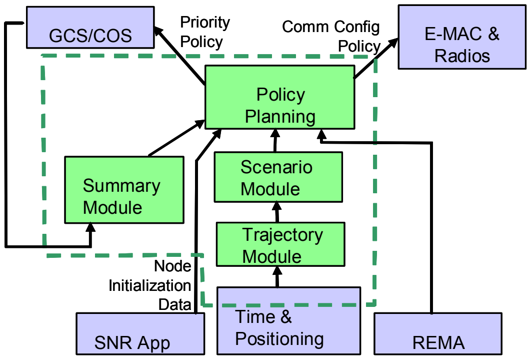

Figure 3 Cognitive Controller

The CC has four major components that use information gathered from the other modules to build a variety of models. The Trajectory component uses information from the Time-and-Positioning module to construct models of what trajectories the nodes are following. The Scenario component uses this information to construct models of when the nodes are likely to be able to communicate. The Summary module uses information gathered from GCS/COS (and the Trajectory component) to construct models of which nodes know what information. Using all of these, the Policy and Planning component constructs a number of “Communications Plans” each specifying a sequence of communication actions to be taken by its node as well as by its peer nodes (these plans include relays of information in which the original node sends information to a peer who in turn sends it to a third node when they meet).

To build these communications plans, the CC constructs a limited depth tree of future possibilities. First it projects the inferred trajectories of each node into the future, determining the sequence of communications configurations that might arise, where each configuration represents a set of nodes that are in communication with one another. Then in each configuration it creates branches for the different information exchanges that each node might be able to conduct. Thus, each path through this tree specifies a set of actions to be taken by its CART node, by peer nodes with which it is likely to be able to communicate, and by peer nodes that these other nodes are likely to be able to communicate with. In effect, the CC puts itself "into the shoes" of its neighbors, estimating what they are likely to do, given information that it can share with them, either directly or indirectly. Since other CART nodes have a CC and each of these is performing the same type of reasoning, this leads to a "collective intelligence" with emergent aggregate behavior that is more optimal than would be possible without such planning.

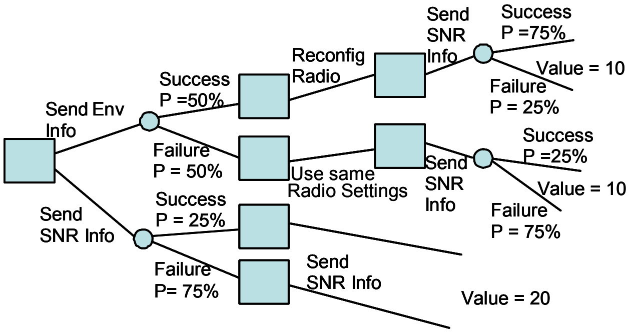

Figure 4 CC Decision Tree

The CC evaluates each of these plans (using standard decision theoretic techniques) based on their expected impact on the situational awareness of the entire CART team, identifies the best near term action to be taken and then sets priorities and policies for GCS/COS and the EMAC modules that support the selected communications plan.

As an example: Node A has not previously been in communication with node B; A also has not been in communication with any other node that has been in communication with B. The environment is not noisy. In this case A sends as much of its SNR map as possible to B followed by as much of A's summary of C and D's knowledge state as possible. The rationale is that, although the SNR is expensive to send, it is of higher value; summaries are cheap but of lesser value, however they may help B make better decisions in the future.

VI. GROUP COMMUNICATIONS SYSTEM

The key components of GCS are the CART Object Store (COS) and the security systems. CART objects can be large – files, publish-subscribe events, etc. For purposes of communication, these large objects are fragmented into smaller message-sized objects, which we call COS objects. COS objects associated with a single CART object and its access control list are managed as a set: although transmitted one at a time, they are delivered to destinations in an all-or-nothing manner.

COS operates like a store-and-forward network; objects have destination lists (individual processes or groups), reside within COS until delivered to all destinations, and are then garbage collected. Delivery is by upcall to the appropriate subsystem on the destination node or nodes. Like packets in a network, COS objects are *immutable*: once they are handed off to COS, they cannot be modified. The originating machine "signs" each COS object at time of generation, by encrypting a SHA-1 hash; an improperly signed object will be flagged as corrupted and discarded.

COS operates using a form of communication based on *gossip* [4], with modification to accommodate the CART environment. Each time a node, say A, enters into contact with some other node, B, each computes a *digest* summarizing the COS objects available upon it. Node A can then pull desired objects from node B and push objects that node B is missing, and vice-versa. This exchange of objects continues for as long as the communication link persists, with actual transmission respecting a prioritization ordering computed by the CC (CC actually prioritizes CART objects; these priorities are then inherited by the COS objects into which the CART object is fragmented). When an object reaches one of its destinations, the receiver sends a signed acknowledgment, which is used to inhibit further gossip and to trigger garbage collection. CART can thus be understood as supporting an intelligent and extremely robust broadcast-flood style of communication.

VII. SECURITY POLICY

As mentioned earlier, secure operation is an important objective for the CART project. We have adopted a security model that should be familiar to users of operating systems such as Windows and Linux. Objects are tagged with the identification of their creator, and have an associated *access control list* giving the identification of individuals and *groups* that have access rights (read, write, etc). A *group* is a set of individuals sharing rights.

When a CART node is first initialized, the user authenticates him or her to the system, establishing initial identification and also authenticating group membership. We associate asymmetric (public-private) keys with individuals and groups, and implement a protocol whereby a node that joins a group is able to obtain a shared copy of that group's private key (shared, that is, only among group members). As is usual for group keying mechanisms [5], we rekey groups each time membership changes by adding a new authenticated member, or by having some authenticated member removed from the group (e.g. a search team member who rotates out of the team to rest, or a node that has been lost, captured, or possibly compromised). We do not rekey a group merely because connectivity to a node is lost – we expect this to be a very common event in the CART environment, and it does not constitute "membership change" for the group.

Each CART object is maintained in the clear only on nodes that have permission to read the contents. Although our architecture does permit copies of an object to be placed on nodes that lack read permission, such an object is first encrypted, as follows. We create a per-object private symmetric key (a DES key), and encrypt the object with the key. We then encrypt the key itself once per entry in the access control list, in this manner generating an object consisting of a vector of encrypted key material. An encrypted CART object is accompanied by its access list; in this manner, any node with legitimate access rights on an object will be able to decrypt the corresponding ACL entry, obtain the per-object key, and then decrypt the object.

Once an object has reached a destination, that destination creates a very small COS object acknowledging receipt, signed in the usual manner. These unforgeable delivery receipts are used to trigger the garbage collection mechanism.

VIII. ENHANCED MAC (EMAC)

The EMAC is a software module that resides on each CART node between the hardware radios and GCS. It is used to relay data traffic between GCS and lower layers, provide optional enhancements to conventional 802.11 unicast and broadcast transfers, control radio parameters to optimize the communication environment, send status report to the CC module on its current communication medium, and record signal strength from peer CART members.

Conventional 802.11 unicast or broadcast mechanism can be used to relay data traffic between GCS and the 802.11 communication radio. The EMAC module also offers optional enhancements to these conventional transfer mechanisms that can be turned on or off by higher layers when needed. These algorithms are meant to improve the performance of the transfer between CART members by reducing the time to acquire a message and the bandwidth required to do so and by increasing the probability of a successful transfer.

One such EMAC enhancement provides a way to reliably transfer broadcast messages. The conventional 802.11 broadcast transfer is unreliable. With this enhancement algorithm enabled, a subset of nodes receiving the broadcast message will acknowledge its receipt. The transmitting node will have the option to retransmit a broadcast message when no acknowledgement is received.

Another enhancement of the EMAC module is to provide wider transmission range and increased delivery ratio by allowing nodes to rebroadcast a received broadcast message when required. The goal of a broadcast transfer is to reach all nodes. The conventional 802.11 broadcast transfer only reaches nodes within hearing range of the transmitting node. Therefore, all nodes more than 1-hop away from the transmitting node will not receive the broadcast message. In a harsh wireless environment with deep fades caused by terrain and interference, only a subset of nodes will be reached with a broadcast transfer. The rebroadcast enhancement allows each node to independently decide if a message should be rebroadcast to quickly disseminate it to as many nodes as possible. The goal of the rebroadcast algorithm is to pick the smallest number of node to rebroadcast a message to reach to largest amount of nodes. However, rebroadcast raises a security concern (a faulty node might use rebroadcast to initiate a DDoS attack in which correct nodes inadvertently cooperate in carrying out the attack). The CART security system is expected handle this carefully.

IV. LOCATION AWARENESS

In the CART system architecture, position establishment is achieved via two independent but somewhat complementary sources: GPS receiver and local UWB (ultra wideband) ranging measurements based on MultiSpectral technology[6]. Each approach has its advantages as well as its inherent limitations, and is thus appropriate in certain circumstances.

Position establishment via GPS receiver is very popular due to widely available commercial GPS receiver devices. However, GPS signal could be blocked by obstacles, therefore, it generally is unavailable indoors. In addition, the positional accuracy error can be large (~10 meter) for a commercially available GPS receiver, and the sampling interval of GPS receivers is normally one or two seconds. Accordingly, CART uses UWB radios as an alternative technology for indoor positioning and tracking (eventually we will also use UWB for data transfer). UWB radio signals can penetrate a variety of obstacles including walls, offering a way to measure all pair-wise distances between nodes with centimeter level accuracy, for up to 30 nodes several times per second.

The Position Data Collector Adapter (PDCA)-GPS interface is used to initialize a GPS receiver and to register the application with the GPS receiver driver. Once the CART system is registered with the GPS receiver driver, it can receive a periodic callback from the GPS receiver. The callback function contains the information about the current position (GPS coordinate), bearing, speed, and GPS time. The time interval of callback could be 0.25~2 second, and varies with GPS receiver being used. We will say that a node with accurate GPS location data is a potential *anchor node*.

The PDCA-UWB interface is used to retrieve pair-wise UWB distance measurements. By combining these measurements with positioning data from some set of anchor nodes, a CART node can establish its position via triangulation.

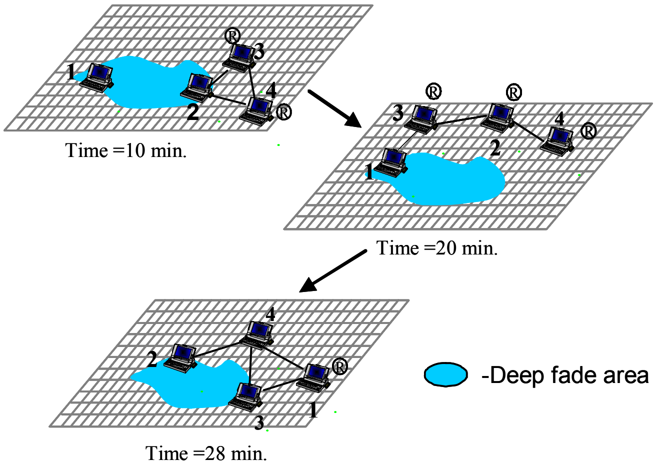

Figure 5 CART Scenarios

X. CART IN ACTION

As an example, we describe a CART experiment in which 4 nodes traverse paths each covering approximately 25% of the grid. Each node will need to perform local measurements and disseminate data to all the group members within a time frame of 30 minutes. At the end of this duration, the data will be sent to a remote location. Referring to Figure 5, at time=10 minutes, node 1 has a high percentage of data that are not received by other nodes since it is isolated due to the large deep fade area. Nodes 3 and 4 have access to the remote site as indicated by the "R" symbol. Node 2,3,4 are within each other's radio range and are able to exchange most of the data. Each of them is aware that there are no links to node 1 and thus concentrates on exchanging data among themselves, using a broadcast mode.

At time=20 minutes, all nodes are connected. However, there is "hidden node" problem, as node 3 and 4 are hidden from each other and their transmission to node 2 will not be protected by CSMA/CA and may collide. Moreover, since that there is no direct route from nodes 2 and 4 to node 1, CC is responsible for selecting the best option for communication to node 1: it could potentially use a relayed re-broadcast if the distances and signal strengths permit, or could rely upon the gossip mechanism in COS to relay data point-to-point.

At time=28 minutes, all nodes are connected, but only node 1 has an uplink to the remote site. CC will now bias communication so that all nodes will attempt sending node 1 the relevant data, possibly through nodes 3 and 4.

XI. RELATED WORK

We are aware of very little prior work on the use of purely ad-hoc communication to support teams that collaborate in the manner enabled by CART. However, the basic idea of sharing and annotating maps has become a major paradigm among web service systems, notably through the dramatic success of the Google "mash-up" architecture. We believe that this style of maps with overlays would be a natural fit for CART.

In addition, DARPA has sponsored an ACERT Savane server (http://acert.ir.bbn.com) as the central point for development, distribution, and maintenance of ACERT software projects.

XII. ACKNOWLEDGEMENT

Many people contribute to the ACERT idea and the CART platform. The authors would like to acknowledge DARPA Program Manager Jonathan Smith for his vision, and the rest of the CART team including Scott Alexander, Arturo Cisneros, Mark Garrett, Vikram Kaul, HeeChang Kim, Isil Sebuktekin and David Wu.

REFERENCES

[1] DARPA BAA 05-37: ACERT, Adaptive Cognition-Enhanced Radio Teams, April 2005.

[2] K. L. Tan and B. C. Ooi, *Data Dissemination in Wireless Computing Environments*. Boston: Kluwer Academic Publishers, 2000.

[3] A. Demers, *et. al.* Epidemic Algorithms for Replicated Database Management. ACM PODC 1987.

[4] K.P. Birman. *Reliable Distributed Systems.* Springer-Verlag (May 2005).

[5] M. Reiter *et. al.* A security architecture for fault-tolerant systems. ACM TOCS 12(4), Nov. 1994

[6] Robert J. Fontana. *et. al.* Commercialization of an Ultra Wideband Precision Asset Location System. Proc IEEE Conf. on UWB Systems and Technologies, November, 2003, Reston, VA