AI Technologies for Tactical Edge Networks *

Karen Zita Haigh
BBN Technologies
5775 Wayzata Blvd
St. Louis Park, MN 55416
khaigh@bbn.com

ABSTRACT

The field of adaptable communication networks is a rich application area for artificial intelligence technology. Recent developments in software defined radio technology have created the opportunity to develop networks that are, in principle, highly adaptable and effective under a much wider range of operating conditions than currently possible, but few researchers are addressing the issue of how to take advantage of this new flexibility. This paper briefly discusses some of the Artificial Intelligence techniques that can and have been be leveraged in this domain.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Network Communications*; C.2.1 [**Computer-Communication Networks**]: *Network Operations*; I.2.6 [**Artificial Intelligence**]: Learning; I.2.8 [**Artificial Intelligence**]: Problem Solving, Control Methods, and Search; I.2.8 [**Artificial Intelligence**]: Distributed Artificial Intelligence

Keywords

Cognitive networks, Distributed control, Distributed learning

General Terms

Design, Algorithms

Invited paper to ACM SIGMOBILE MobiHoc 2011, Workshop on Tactical Mobile Ad Hoc Networking, May 2011, Paris, France

1. INTRODUCTION

The demand is increasing for networking technologies that support robust communication and functionality under challenging operating conditions. In general, network configurations are hand-tuned and remain static during operations. However, since user needs and operating conditions both change over time, *cognitive networks* must be designed that are aware of their performance needs, determine if their needs are being met, and revise system configurations to better meet their needs.

Recent developments in software defined radio technology have opened up the opportunity to develop networks that are, in principle, highly adaptable and effective under a much wider range of operating conditions than currently possible [3, 9, 26]. However, while these tools provide new flexibility, few are addressing the issue of how to manage or control them. This paper briefly discusses some of the Artificial Intelligence techniques that can (and should) be leveraged in this domain, and highlights specific cases of successful implementations.

A Mobile Ad hoc NETwork (MANET) is a type of ad hoc network that consists of "mobile platforms... which are free to move about arbitrarily... At a given point in time... wireless connectivity in the form of a random, multi-hop graph or 'ad hoc' network exists between the nodes" [15]. MANETs are characterized by dynamic topologies, bandwidth-constrained, variable capacity links, energy-constrained operation, and limited physical security. A MANET is needed for self-forming, self-configuring, and self-healing operation where the media and communications channels undergo rapid changes (e.g., over free space optical, RF, and underwater acoustic links) and nodes freely enter and leave the network. MANETs are not needed when links are unchanging, e.g., GEO satellite links, LOS microwave tower links, fiber optics, Ethernet and wired infrastructure. For these reasons, MANETs have critical requirements for rapid and accurate adaptivity.

Artificial Intelligence (AI) techniques enable real-time, context-aware adaptivity that have the potential to meet the needs of networks in general, and MANETs in particular. To meet this grand vision, there are several key challenges that need to be addressed.

- The network must be able to *identify and forecast* network conditions, including communications environment and performance.
- The network must be able to *adapt* to constantly changing conditions—communication needs change, communication conditions change, and participants may join or depart.
- The network must *balance* the needs of many users—military, commercial, civilian, and government—while conforming to official regulations and Policies such as rules-of-engagement.

Intelligent cognitive radios require multiple interacting capabilities for situation assessment, planning and learning. Cognitive networks require those capabilities to operate cooperatively in a distributed, diverse environment.

2. A BRIEF INTRO TO ARTIFICIAL INTELLIGENCE

Given that the majority of readers of this paper come from the communications and networking community, it may be useful to provide a little context.

Artificial Intelligence (AI) is the branch of computer science concerned with the automation of intelligent behaviour [36], usually associated with human thinking such as decision making, problem solving and learning [2]. In 1950, Alan Turing proposed the *Turing Test* [62] which called for a human judge to interact through a terminal to both another human and a computer; if the judge cannot tell which is which, then the machine is said to pass the test and would be considered intelligent. The term *Artificial Intelligence* was coined in 1956 by notable researchers including Herb Simon, Allen Newell, John McCarthy and Marvin Minsky, at the Dartmouth Conference [40]. McCorduck [41] presents a comprehensive history of AI, while Russell and Norvig [56] describe AI techniques appropriate for building decision-making agents that make rational actions for their given context.

AI draws techniques from a broad variety of fields including mathematics, psychology, economics, and control theory. AI has a huge variety of subfields, including plannning and scheduling, machine learning, knowledge engineering and fusion, and constraint reasoning.

Natural language processing, speech recognition, machine vision and robotics all had origins in AI. Practical AI successess are often pulled into their own domains, leaving AI researchers to deal with the unsolved problems. Larry Tesler is often misquoted as having said "AI is whatever hasn't been done yet" [27]. Tesler corrects the quote to "Intelligence is whatever machines haven't done yet" [60].

The Odd Paradox

> *Practical AI successes, computational programs that actually achieved intelligent behavior, were soon assimilated into whatever application domain they were found to be useful in, and became silent partners alongside other problem-solving approaches, which left AI researchers to deal only with the "failures," the tough nuts that couldn't yet be cracked.*
>
> *McCorduck, 2004 [41]*

3. NETWORKING PROBLEMS AMENABLE TO AI

Artificial Intelligence techniques could plausibly be used in any Networking problem that involves some form of situation assessment and/or decision making. The following list is a small sample of some of the specific domain problems that AI techniques may be able to help solve:

- Cyber Security[1]
- Network Configuration and Planning
- Network Control and Coordination
- Policy and Constraint Management
- Performance Analysis

These application areas appear in wired networks, wireless networks, and MANETs to different degrees. Because the cost of cabling is so high in wired networks, network configuration and planning is a critical requirement, but is unlikely to change rapidly; planning systems can therefore take their time to generate extremely high-quality plans. In a wireless network, access control, policy management and fairness, take on increasing importance. In a mobile ad hoc network, particularly those used by the military, connectivity is extremely dynamic, nodes may be more heterogeneous and applications may be supporting more time-critical needs. Tactical MANETs may further face adversarial conditions and limited knowledge.

AI systems are often described using the cognition loop of "Sense, Plan, Act, and Learn," similar to the OODA loop: "Observe, Orient, Decide, Act" [6, 7]. Joe Mitola proposed the OOPDAL cognition loop specifically for cognitive radio that effectively merges these: "Observe, Orient, Plan, Decide, Act and Learn" [44].

In the **Observe** step, the system must both *collect* the raw sensor data and cluster that data into hypothesized events, and then *validate* that data by paring the set of hypothesized events down to the set of likely events. Network modules need to expose internal state, including current values of controllable parameters, current values of monitored parameters, and current activity. AI modules need to integrate these values across the layers in the stack, and provide the modules with guidance on important element, including setting param-

[1]Because cyber security is a huge research area unto itself, we do not further address these issues in this paper.

eters and performance objectives. The AI also needs to collect patterns of activity and track performance trends.

In the **Orient** step, the system must *assess* the situation (situation assessment). This process involves inferring what else might be true if the event has happened to collate a common understanding of the overall situation. *Intent Inference* infers the goals of other agents based on observations of their actions. Finally, *impact analysis* examines the potential ramifications of the current situation, including predicting future states given possible dynamism in the domain and selected courses of action. The network modules need to expose any derived computations, including analyses of performance or network state. If network modules estimate future conditions, these estimates should also be exposed. The AI needs to interpret these observations and identify potential factors (or root causes) of situations, and compute progress toward performance goals. The AI then needs to estimate future conditions and the likelihood of achieving goals, so that it can decide on the urgency of responding to problems.

In the **Plan** step, the system must first *identify goals* to be achieved (and when). This step involves managing the multi-objective performance criteria for all current tasks and upcoming reservations. (Note that network-wide goals are often different from node-specific goals.) The system must then *generate plans* to achieve those goals. Planning involves causality reasoning, conditional planning (or "what-if" analyses), temporal reasoning, constraint reasoning, and resource management. Given the current state of the network, the planner must predict the effect of potential actions on the future state of the network. Plans can be generated at multiple timescales, handling immediate concerns at a fine-granularity, and longer-term issues at a coarse-granularity (potentially allowing negotiation with other nodes). Given that MANETs operate in a multi-objective space, a planner may generate several plans that tradeoff meeting one objective for another, ideally on a pareto-optimal curve. Finally, the system must *schedule* to allocate specific resources to specific activities over time. Planning & scheduling often operate iteratively, in the sense that tasks cannot be selected for a plan if no schedule exists. Traditionally, network modules contain significant scheduling capabilities, but the planning capabilities are implicit in the software, that is, the human network engineer performs the planning.

The **Decide** step *selects* among the candidate plans and schedules, and then *allocates* computational and radio resources. Given how quickly the domain changes, a potential approach is to select actions that are common at the beginning of "most" of the candidate plans, because the plan is likely to be revised as conditions change.

In the **Act** step, the system implements the chosen activities. This may include setting values of parameters or replacing running modules or waveforms. Note that if the radio or its software has actions that can be selected, these should be exposed to the plan/decide steps otherwise the system will be unable to use the full breadth of system capability.

In the **Learn** step, the system uses experience to update models so that the other steps can make more accurate forecasts. The system can learn human and application-level behaviour, including node mobility and data-access patterns (which applications or humans or roles are accessing the network, and what each of them needs and when). The system can learn environmental conditions, including connectivity patterns and geographical factors. It can also learn node capabilities, including capacity, reliability, and functionality. For example, if the original description of the node's capabilities are incorrect, then the plan/decide steps may choose actions that cannot be implemented; the learner should update these models. The learner can use both explicit human feedback (e.g., QoS is below par), or empirical performance data (e.g., statistics mapping parameter settings to QoS).

4. AI TECHNIQUES IN NETWORKING

While almost any AI technique could potentially prove useful in a networking environment, certain techniques are more promising and/or have already produced interesting results. These include Knowledge Engineering, Planning and Scheduling, Machine Learning, Distributed AI and Multi-agent systems, including biologically-inspired approaches, and Game Theory.

Knowledge Engineering aims to capture knowledge so that a computer system can solve complex problems [19]. Different knowledge representation approaches are used for different types of knowledge, and the different ways that it will be used. Much knowledge engineering work is concerned with constructing Ontologies. In the networking domain, this knowledge would include models of physics and signal propagation, constraints on the system, analysis of interactions, and rules of thumb (e.g., about how to configure the system). A formal ontology may help a cognitive system reason about how and when capabilities are interchangeable, e.g., recognizing that either of two metrics for computing Quality of Information may be used and that a metric for Quality of Service may be an appropriate replacement under some conditions. Semantics and representations are important considerations for cognitive networks [22, 30]. Several researchers have developed knowledge bases and heuristic rules to optimize the network [22, 33, 52].

Planning and Scheduling techniques are appropriate for decision-making situations, where tasks need to be organized and coordinated to meet performance objectives, under resource constraints. In dynamic environments, the plan needs to be monitored because predictions about performance may have been inaccurate or the conditions have changed such that previously-selected actions are no be appropriate. In these cases the strategy needs to be revised online. Multi-agent planning, dynamic programming, constraint satisfaction, and distributed or combinatorial optimization algorithms are common techniques. Planning and scheduling techniques in networks can decide what content to move, where, when, and how, including power-aware computing, node activity and task scheduling, and network management. Scheduling packets and admission control may also benefit from these approaches, but strictly AI-based approaches may find the rapid-decision cycle challenging. Chadha [10, 11] created a self-organizing network management hierarchy that dynamically updates itself based on changes in connectivity or domain requirements. As an example task-allocation scheme, mobile ad hoc networks can benefit from pre-pulling or pre-pushing data towards the nodes at the edge of the network. Intelligent search mechanisms can similarly decide which nodes to use as resources for information [28, 67]. Chadha et al use machine learning, planning and domain expertise to dynamically select and place servers in MANETs [12]; Tapiador and Clark [59] combine genetic algorithms with policies for the same problem. Lau et al [32] use AI techniques for planning under uncertainty to estimate the best opportunities for communicating with other nodes. PNUTS [64] contains an adaptive scheduler for handling server queries.

Machine Learning (ML) techniques aim to improve the performance of a system by observing the environment and updating models that describe the interactions of observables [31, 42, 43]. ML techniques are appropriate in every domain that is imperfectly modelled. Most complex domains (including networking) fall into this category. Because the set of all possible behaviors is too large to be covered by observed examples, the learner must generalize so that the learned model is useful for new (previously unseen) cases. ML techniques include artificial neural networks, support vector machines, clustering, explanation-based learning, induction, reinforcement learning, genetic algorithms, nearest-neighbour methods, and case-based learning. *Data Mining* techniques are a subset (or close cousin) to ML techniques, in that they identify patterns in large datastores. Data Mining results can be used in a ML system to improve its models.

Dietterich and Langley [17] provide a good overview of ML techniques and how they could be applied to Cognitive Networks, but cite only one concrete example of a realized system in communications networking. Possibly the earliest use of ML in networking, Littman and Boyan [5] introduced a reinforcement-learning approach to routing in networks. Other researchers have extended this work to a wireless environment, to handle dynamic load, to manage energy and to plan node mobility [13, 14, 34, 55, 58]. Another rich area for ML is learning how parameters interact with each other and with the domain. Rieser [53] and Rondeau [54] used genetic algorithms to tune parameters and design waveforms. The experiments show no data about how fast it works and moreover the learning appears to operate offline; Rieser states explicitly that it "may not be well suited for the dynamic environment where rapidly deployable communications systems are used." All demos involve one receiver, one transmitter, and one jammer, although in theory the approach should not be limited. Newman et al [48, 47] similarly use genetic algorithms to optimize parameters in a simulated network; they also show no time results. Montana et al [46] used a genetic algorithms approach for parameter configuration in a wireline network that can find the 95% optimal solution in "under 10 minutes." Haigh et al [24, 25, 61] were the first to demonstrate ML in an on-line (real-time) real-world (not simulation) MANET. Their distributed-learning approach met the requirements for speed, low-communication, heterogeneous mobile ad hoc networks, and dramatically improved the overall performance of the MANET.

MANET networks are often organized into cluster hierarchies to achieve performance guarantees [68]; ML techniques could be leveraged here. ML techniques could also be used to build patterns of users in forward-deployed enclaves: to understand the relationship between task (or role) and topics of interest, and when those files will be needed [70].

Distributed AI and *Multi-agent Systems* are concerned with finding distributed solutions for AI problems [21, 50, 66]. Techniques address domains that have the following characteristics:

Discrete: Local goals and constraints
Deprived: Locally resource constrained
Distributed: Embedded in a physical world
Decentralized: Local decisions and local views of the environment (i.e., no centralized decision maker)
Diverse: Different capabilities and different roles
Dynamic: Changing task/mission and domain

DAI and MAS approaches generally decompose centralized techniques to make them appropriate for the decentralized environment, often with some calculation of the tradeoff between optimality and latency. While conceptually appropriate for the communications networking environment [20], these traditional techniques have to-date not acknowledge or address a key requirement for communications networks, namely that *the task being negotiated is the communications itself.* In other words, traditional AI has always assumed that that communication is "safe," negotiating and coordinating only the application-level tasks [39, 45, 69]; moreover they generally require massive communications with non-neighbours, universally do not support mobility (changing connections or constraints between the nodes), and universally do not support a changing objective function. These drawbacks are so significant that extensive research and redesign are required to make them applicable in this domain.

Biologically-inspired computing approaches are lightweight coordination mechanisms [63], and have been used for a variety of networking problems. AntHocNet [16] uses both proactive and reactive schemes to update the routing tables, and outperforms AODV. Konak et al [29] use particle swarm optimization and agents to improve network connectivity. Sesum-Cavic and Kühn use swarm intelligence for dymanic load balancing. Parunak and Brueckner use a stigmergic approach to decide where to locate services on a MANET [51]. Biologically-inspired methods are often slower in reaction than conventional control systems, and may lose optimality, but can offer greater resilience.

Game theory is a branch of applied mathematics that is used for analyzing the interaction among agents whose decisions affect each other. Game theory is becoming a common formalism for studying strategic and cooperative interaction in multi-agent systems [18]. Applications of game theory to wireless communications have also received significant interest by the research community; Nisan et al [49] and Liu and Wang [34] present good introductions. Previous research includes enforcing fairness and thwarting selfish behavior in shared medium [37], multi-hop packet relaying [1, 8], multicarrier (OFDM) systems [4], MIMO [57], interactions between communicating nodes [35], and overlay networks [65].

5. AI CHALLENGES IN NETWORKING

There are many interesting challenges for AI in networking. Characteristics that make this an interesting domain for AI include:

Dynamic: Very few things in a MANET environment are static. Military missions change, user requirements change, users join or leave the network, hardware fails, and mobility causes continuous fluctuations in connectivity.

Resource constrained: Most notably, nodes are bandwidth-constrained: it would overwhelm the network for nodes to share all knowledge with other nodes. In fact, as noted above, most MAS negotiation techniques are unsuitable for this reason. Power management is a critical requirement for remote operations.

Partially-observable: Many factors that affect communication cannot be observed. Few radios, for example, have a "fog" sensor.

Ambiguous observations: Detection and understanding of a change in situation is not always simple. For example, how does the system automatically tell the difference between short-term fade versus entering a building?

Diverse: Nodes in a MANET have a wide variety of capabilities, from small hand-held radios to large radios with satellite communications (satcom); these vary both in communications and compute power. This heterogeneity requires different solutions on different nodes.

Massive scale: There are roughly 600 observable parameters and 400 controllable parameters (possibly continuous-valued) to configure *per node*[2]. We thus have a distributed, heterogeneous, low-communication, partially-observable, high-latency optimization problem of approximately μ^{PN} choices per timestep[3]; one second would be a large timestep.

Complex interactions: Networking parameters have deep, poorly-understood interactions with each other and with system performance. In many cases, specific pair-wise interactions can be identified, such as increased power reduces battery life. However, most of these pair-wise interactions are carefully caveatted by the networking community, with conditionals that are rarely observable or computable. Cognitive control in the general case is therefore seldom simple: the level at which symptoms appear may not be the level at which changes to the node configuration must be made; symptoms may be ambiguous at one level or at a given time and require more context; changes at one layer may impact other layers and may cause new issues; and the timing of changes may be critical.

High-latency: Many actions cause a delayed effect. For example, data transmissions from one node may only affect downstream nodes; the result takes time to propagate back to the first transmitter.

Complex temporal feedback loops: Within a node, certain activities occur at very rapid speeds (e.g., between the Medium Access Control (MAC) and Physical layers) requiring very a very tight feedback loop to support cognitive control. Other activities (e.g., at the Routing layer) occur on a longer time-scale and cognitive control algorithms may need to take into account a wider range of factors

[2]We include the ability to dynamically reconfigure the IP stack as control parameters; we model alternate configurations by creating a control parameter x for each available network module, where $x = 1$ when the module has been invoked, and $x = 0$ when the module is not operating [25]. No current system exposes all of these parameters; the highest known is about 100 parameters, of which 30 are controllable.

[3]P = number of parameters, N = number of nodes, and μ is the average number of values that a parameter can take.

in a slow feedback loop. Between nodes, there is yet a longer feedback loop between changes that are made and the effects that are observed in network-level performance. The variety of temporal loops and their dramatic speed differences means that correlating cause and effect of actions is particularly challenging.

Discrete: As a result of the limited communication and frequent disconnections, nodes have to make decisions locally, considering local requirements and constraints.

Heterogeneous Intercommunication: There is a very strong norm in the networking community that all nodes must be designed and (statically) configured to interoperate; typical ad hoc networks build a group of homogenous nodes. Cognitive networks break this assumption: each node can have an independent cognitive controller, and thus network nodes *may be heterogeneous, and may fall into in non-interoperable configurations.*[4] Meanwhile traditional AI has always assumed that that communication is "safe," negotiating and coordinating only the application-level tasks [39, 45, 69]; moreover they also generally require very high communications overhead. Ensuring that multiple nodes are coordinated enough to maintain basic communications is a key research area for cognitive networking.

Complex Access Policies: Due to the heterogeneous nature of the data and the nodes, access policies may restrict the set of nodes that are permitted to hold or transmit specific data. This issue is especially true in military MANETs.

Complex multi-objective performance requirements: Multiple users have interacting requirements and policies, thus creating a complex multi-objective function that captures mission, situational and social standpoints [24]. It can include a wide variety of issues including bandwidth, application-level quality of service, energy, network connectivity, and security.

In other domains, AI techniques are capable of addressing the full richness of most of these challenges. In the networking domain, AI techniques are just beginning to scratch the surface. We need to bring these techniques into the networking domain, and address them in depth. Moreover, AI techniques have addressed many of these challenges in the same system (e.g., robotics), with the notable exception of automatic heterogeneous intercommunication. This challenge has never been addressed by the AI community *or* the Networking community. ADROIT [61], by giving each node its own learning system, represented a *radical* departure from the traditional networking stance that requires homogeneous configurations. ADROIT was the first system to demonstrate a effective heterogeneous MANET. As AI techniques are slowly given greater access to network configuration, this challenge will be critical to solve.

An important consideration is that Networking software is not typically designed to support AI-based control. The networking software architectures do not expose the states and controls needed to effectively adapt networking operations. Tight coupling between the networking software architecture and AI-based control has been demonstrated in ADROIT [61], a communications network based on an architecture proposed by Haigh et al [23].

[4]The alternative is to have one cognitive controller for several nodes; while coordination issues are reduced, communication overhead increases dramatically and intelligent control is vulnerable to network partitions.

6. CONCLUSIONS

By dynamically changing their communications patterns based on the current conditions, cognitive networks can communicate (interoperate) with both cognitive and conventional radios, adapt to changes in infrastructure, and modify behavior to avoid or mitigate threats.

There are many powerful AI techniques that address knowledge engineering, situation assessment, planning, scheduling, and learning in distributed environments. AI techniques are ready to be challenged with this complex real-world domain, just as Networking requirements are reaching the limits of what can be done without AI. We are at a nexus from which interesting ideas and capabilities will develop.

The challenges outlined above are all technical. There is a social-engineering challenge to address as well: the human-to-human interaction of the AI community differs dramatically from that of the networking community; we must find ways to address these cultural differences [23].

7. REFERENCES

[1] L. Anderegg and S. Eidenbenz. Ad hoc-VCG: a truthful and cost-efficient routing protocol for mobile ad hoc networks with selfish agents. In *Proc. of ACM annual International Conference on Mobile Computing and Networking (MobiCom)*, pages 245–259, 2003.

[2] R. E. Bellman. *An Introduction to Artificial Intelligence: Can Computers Think?* (San Francisco, CA: Boyd & Fraser Publishing Company), 1978.

[3] E. Blossom. GNU radio: tools for exploring the radio frequency spectrum. *Linux Journal*, 122, June 2004.

[4] H. Bogucka. Game theoretic model for the OFDM water-filling algorithm with imperfect channel

state information. In *Proc. IEEE International Conference on Communications*, pages 3814–3818, 2008.

[5] J. A. Boyan and M. L. Littman. Packet routing in dynamically changing networks: A reinforcement learning approach. In *Advances in Neural Information Processing Systems (NIPS)*, pages 671–678. (San Mateo, CA: Morgan Kaufmann), 1994.

[6] J. Boyd. Destruction and Creation, 1976. Unpublished Essay, available at http://www.goalsys.com/books/documents/DESTRUCTION_AND_CREATION.pdf.

[7] J. Boyd. Patterns of conflict, December 1986. http://www.ausairpower.net/JRB/patterns.ppt.

[8] L. Buttyan and J.-P. Hubaux. *Security and Cooperation in Wireless Networks: Thwarting Malicious and Selfish Behavior in the Age of Ubiquitous Computing*. (Cambridge, UK: Cambridge University Press), 2007.

[9] A. Casimiro, J. Kaiser, and P. Veríssimo. An architectural framework and a middleware for cooperating smart components. In *Proc. of the 1st Conference on Computing Frontiers*, pages 28–39, Ischia, Italy, 2004.

[10] R. Chadha and C.-Y. Chiang. Drama: Distributed policy management for manets. In *IEEE Workshop on Policies for Distributed Systems and Networks*, pages 235–237, June 2008.

[11] R. Chadha and L. Kant. *Policy-Driven Mobile Ad Hoc Network Management*. (Somerset, NJ: John Wiley and Sons), 2007.

[12] R. Chadha, A. Poylisher, B. Deb, M. Littman, and B. Sabata. Adaptive dynamic server placement in MANETs. In *IEEE Military Communications Conference (MILCOM 2005)*, pages 1083–1089, 2005.

[13] Y.-H. Chang, T. Ho, and L. P. Kaelbling. Mobilized ad-hoc networks: A reinforcement learning approach. In *Proc. International Conference on Autonomic Computing (ICAC)*, 2004.

[14] S. P. M. Choi and D.-Y. Yeung. Predictive q-routing: A memory-based reinforcement learning approach to adaptive traffic control. In *Advances in Neural Information Processing Systems (NIPS)*, 1996.

[15] S. Corson and J. Macker. Mobile ad hoc networking (manet): Routing protocol performance issues and evaluation considerations, January 1999. RFC 2501.

[16] G. Di Caro, F. Ducatelle, and L. M. Gambardella. AntHocNet: an adaptive nature-inspired algorithm for routing in mobile ad hoc networks. In *European Transactions on Telecommunications (Special Issue on Self-organiztion in Mobile Networking)*, volume 16, pages 443–455, 2005.

[17] T. G. Dietterich and P. Langley. Machine learning for cognitive networks: Technology assessment and research challenges. In Mahmoud [38].

[18] E. Elkind and K. Leyton-Brown. Introduction to the ai magazine special issue on algorithmic game theory. *AI Magazine*, Winter, 2010.

[19] E. A. Feigenbaum and P. McCorduck. *The fifth generation (1st ed.)*. (Reading, MA: Addison-Wesley), 1983.

[20] D. H. Friend, R. W. Thomas, A. B. MacKenzie, and L. A. DaSilva. Distributed learning and reasoning in cognitive networks: Methods and design decisions. In Mahmoud [38].

[21] L. Gasser and M. N. Huhns. *Distributed Artificial Intelligence, vol 2*. (San Mateo, CA: Morgan Kaufmann), 1990.

[22] A. Ginsberg, W. D. Horne, and J. D. Poston. The semantic side of cognitive radio. In Mahmoud [38].

[23] K. Z. Haigh, T. S. Hussain, C. Partridge, and G. D. Troxel. Rethinking networking architectures for cognitive control. In *Cognitive Wireless Networking Summit*. Snoqualmie, WA, 2008.

[24] K. Z. Haigh, O. Olofinboba, and C. Y. Tang. Designing an implementable user-oriented objective function for MANETs. In *IEEE International Conference On Networking, Sensing and Control*, London, U.K., April 2007.

[25] K. Z. Haigh, S. Varadarajan, and C. Y. Tang. Automatic learning-based MANET cross-layer parameter configuration. In *Workshop on Wireless Ad hoc and Sensor Networks (WWASN2006)*, Lisbon, Portugal, 2006.

[26] M. A. Hiltunen and R. D. Schlichting. The Cactus approach to building configurable middleware services. In *Proc. of the Workshop on Dependable System Middleware and Group Communication (DSMGC)*, Nuremberg, Germany, October 2000.

[27] D. R. Hofstadter. *Gödel, Escher, Bach: An Eternal Golden Braid*. (New York, NY: Vintage Books), 1979.

[28] V. Kalogeraki, D. Gunopulos, and D. Zeinalipour-Yazti. A local search mechanism for peer-to-peer networks. In *Proc. International Conference on Information and Knowledge Management (CIKM)*, pages 300–307. (New York, NY: ACM Press), 2002.

[29] A. Konak, O. Dengiz, and A. E. Smith. Improving network connectivity in ad hoc networks using particle swarm optimization and agents. In *Wireless Network Design: Optimization Models*. (New York, NY: Springer), 2011.

[30] V. J. Kovarik. Cognitive research: Knowledge representation and learning. In *Cognitive Radio*

Technology. 2008.

[31] P. Langley. *Elements of Machine Learning*. (San Mateo, CA: Morgan Kaufmann), 1995.

[32] R. Lau, S. Demers, Y. Ling, B. Siegell, E. Vollset, K. Birman, R. van Renesse, H. Shrobe, J. Bachrach, and L. Foster. Cognitive adaptive radio teams. In *IEEE Sensor and Ad Hoc Communications and Networks*, pages 842–847, 2006.

[33] H. Liu, A. Kershenbaum, and R. Van Slyke. Artificial intelligence applications to communication network design with bulk facilities. In *Proc. of the ACM annual conference on Communications (CSC)*, pages 345–350, Kansas City, Missouri, 1992. (New York, NY: ACM Press).

[34] K. J. R. Liu and B. Wang. *Cognitive Radio Networking and Security: A Game-theoretic View*. (Cambridge, UK: Cambridge University Press), 2011.

[35] X. Liu, G. Noubir, R. Sundaram, and S. Tan. SPREAD: Foiling smart jammers using multi-layer agility. In *Proc. of InfoCom Mini-Symposium*, 2007.

[36] G. F. Luger and W. A. Stubblefield. *Artificial Intelligence: Structures and Strategies for Complex Problem Solving*. (Palo Alto, CA: Benjamin Cummings), 1993. 2nd edition.

[37] A. B. MacKenzie and S. B. Wicker. Stability of multipacket slotted aloha with selfish users and perfect information. In *Proc. of IEEE International Conference on Computer Communication (InfoCom)*, 2003.

[38] Q. Mahmoud (ed). *Cognitive Networks: Towards Self-Aware Networks*. (Somerset, NJ: John Wiley and Sons), 2007.

[39] T. W. Malone and K. Crowston. The interdisciplinary study of coordination. *ACM Computing Surveys*, 26(1):87–119, March 1994.

[40] J. McCarthy, M. Minsky, N. Rochester, and C. Shannon. A proposal for the Dartmouth summer research project on Artificial Intelligence. See http://en.wikipedia.org/wiki/Dartmouth_Conferences, 1955.

[41] P. McCorduck. *Machines Who Think*. (Natick, MA: A K Peters, Ltd.), 2004.

[42] R. S. Michalski, I. Bratko, and M. Kubat. *Machine Learning and Data Mining*. (Somerset, NJ: John Wiley and Sons), 1998.

[43] T. Mitchell. *Machine Learning*. (New York, NY: McGraw-Hill), 1997. ISBN 0070428077.

[44] J. Mitola III. Cognitive radio for flexible multimedia communications. In *IEEE International Workshop on Mobile Multimedia Communications (MoMuC)*, pages 3–10, 1999.

[45] P. J. Modi, W.-M. Shen, M. Tambe, and M. Yokoo. ADOPT: Asynchronous distributed constraint optimization with quality guarantees. *Artificial Intelligence*, 161(1-2):149–180, 2005.

[46] D. Montana, T. Hussain, and T. Saxena. Adaptive reconfiguration of data networks using genetic algorithms. In *Proc. Genetic and Evolutionary Computation Conference (GECCO)*, 2002.

[47] T. Newman, J. Evans, and A. Wyglinski. Reconfiguration, adaptation and optimization. In A. M. Wyglinski, M. Nekovee, and T. Hou (eds), *Cognitive Radio Communications and Networks: Principles and Practice*. (New York, NY: Elsevier/North Holland), 2009.

[48] T. R. Newman, R. Rajbanshi, A. M. Wyglinski, J. B. Evans, and G. J. Minden. Population adaptation for genetic algorithm-based cognitive radios. In *Proc. of ICST Conference on Cognitive Radio Oriented Wireless Networks and Communications (CROWNCOM)*. (New York, NY: IEEE Press), May 2007.

[49] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani (eds). *Algorithmic Game Theory*. (Cambridge, UK: Cambridge University Press), 2007.

[50] G. M. P. O'Hare and N. R. Jennings (eds). *Foundations of Distributed Artificial Intelligence (Sixth Generation Computer Technologies)*. (Somerset, NJ: John Wiley and Sons), 1996.

[51] H. Parunak and S. Brueckner. Stigmergic learning for self-organizing mobile ad-hoc networks (manets). In *Proc. International Joint Conference on Autonomous Agents and MultiAgent Aystems (AAMAS)*. 2004.

[52] S. Pierre and H. Hoang. An Artificial Intelligence approach to improving computer communications network topologies. *Journal of the Operational Research Society*, 41(5):405–418, 1990.

[53] C. J. Rieser. *Biologically Inspired Cognitive Radio Engine Model Utilizing Distributed Genetic Algorithms for Secure and Robust Wireless Communications and Networking*. PhD thesis, Virginia Tech, Blacksburg, VA, 2004.

[54] T. W. Rondeau. *Application of Artificial Intelligence to Wireless Communications*. PhD thesis, Department of Electrical Engineering, Virginia Polytechnic Institute, Blacksburg, VA, September 2007.

[55] B. K. Russell. *Learning-Based Route Management in Wireless Ad Hoc Networks*. PhD thesis, Department of Computer Science, Rutgers University, New Brunswick, NJ, USA, August 2008.

[56] S. J. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. (Upper Saddle River, NJ:

Prentice-Hall), 1995.

[57] G. Scutari, D. Palomar, and S. Barbarossa. Competitive design of multiuser MIMO systems based on game theory: A unified view. *IEEE Journal on Selected Areas in Communications*, 26(7):1089–1103, 2008.

[58] D. Subramanian, P. Druschel, and J. Chen. Ants and reinforcement learning: A case study in routing in dynamic networks. In *Proc. International Joint Conference on Artificial Intelligence (IJCAI)*, pages 832–838. (San Mateo, CA: Morgan Kaufmann), 1997.

[59] J. Tapiador and J. Clark. Learning autonomic security reconfiguration policies. In *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*, pages 902–909, 2010.

[60] L. Tesler. Adages & Coinages. http://www.nomodes.com/Larry_Tesler_Consulting/Adages_and_Coinages.html, cited 29 March 2011.

[61] G. D. Troxel, E. Blossom, S. Boswell, A. Caro, I. Castineyra, A. Colvin, T. Dreier, J. B. Evans, N. Goffee, K. Z. Haigh, T. Hussain, V. Kawadia, D. Lapsley, C. Livadas, A. Medina, J. Mikkelson, G. J. Minden, R. Morris, C. Partridge, V. Raghunathan, R. Ramanathan, P. G. Rubel, C. Santivanez, T. Schmid, D. Sumorok, M. Srivastava, R. S. Vincent, D. Wiggins, A. M. Wyglinski, and S. Zahedi. Enabling open-source cognitively-controlled collaboration among software-defined radio nodes. *Computer Networks*, 52(4):898–911, March 2008.

[62] A. Turing. Computing machinery and intelligence. *Mind*, LIX(236):433–460, October 1950.

[63] P. Valckenaers, K. Hadeli, B. Saint Germain, P. Verstraete, and H. Van Brussel. MAS coordination and control based on stigmergy. *Computers in Industry*, 58(7):621–629, September 2007.

[64] Y. Vigfusson, A. Silberstein, B. F. Cooper, and R. Fonseca. Adaptively parallelizing distributed range queries. *Proc. VLDB Endowment*, 2:682–693, August 2009.

[65] Y. Wang and A. Nakao. On socially-inspired cooperative and efficient overlay network evolution based on group selection pattern. In *Proc. International Conference on Bio-Inspired Models of Network, Information and Computing Systems*, pages 21:1–21:8. (Brussels, Belgium: ICST), 2008.

[66] G. Weiss (ed). *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence.* (Cambridge, MA: MIT Press), 1999.

[67] S. J. H. Yang, J. Zhang, L. Lin, and J. J. P. Tsai. Improving peer-to-peer search performance through intelligent social search. *Expert Systems with Applications*, 36(7):10312–10324, September 2009.

[68] J. Yu and P. Chong. A survey of clustering schemes for mobile ad hoc networks. *Communications Surveys Tutorials, IEEE*, 7(1):32–48, 2005.

[69] X. Zhang, V. Lesser, and S. Abdallah. Efficient Management of Multi-Linked Negotiation Based on a Formalized Model. *Autonomous Agents and Multi-Agent Systems*, 10(2):165–205, 2005.

[70] I. Zukerman, D. Albrecht, and A. Nicholson. Predicting users' requests on the WWW. In *Proc. of the International Conference on User Modeling (UM)*, 1999.