Shared Focus of Attention for Heterogeneous Agents (Short Paper)

Jacob Beal
MIT CSAIL
77 Massachusetts Ave
Cambridge, MA, USA
jakebeal@mit.edu

ABSTRACT

A network of cooperating agents must be able to reach rough consensus on a set of topics for cooperation. With highly heterogeneous agents, however, incommensurable measures and imprecise translation render ordinary consensus algorithms inappropriate. I present a distributed mechanism for shared focus of attention that begins to address these problems, using an engineered emergence approach inspired by recent results on the dynamics of evolution in systems with spatial extent. Simulation shows that the algorithm converges in time proportional to the diameter of the network and gives a range of reasonable settings for the parameters.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems*

General Terms

Algorithms

Keywords

Shared Focus of Attention, Heterogeneous Agents

1. CONTEXT

If agents are to cooperate effectively, they must first agree on a set of topics for cooperation. In many systems, this is trivial to accomplish—for example, in blackboard systems there is no disagreement on topics because the topics are dictated by the contents of the blackboard, which is shared by all of the agents.

With heterogeneous agents, maintaining agreement on a set of topics is more problematic because there may be no clear standard for comparison of topics proposed by different agents. Consider, for example, a set of specialist agents in a model of natural intelligence. When faced with an everyday task like crossing the street, the vision specialist may suggest the traffic light as a topic, the hearing specialist may suggest the sound of an unseen approaching car, the tactile specialist an uncomfortable pebble underfoot, and so on.

Cite as: Shared Focus of Attention for Heterogeneous Agents (Short Paper), Jacob Beal, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)*, Padgham, Parkes, Müller and Parsons (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 1627-1630

These differing categories of topic proposals are effectively incommensurable: although we might assign them all an abstract quantity such as "utility" or "salience," the lack of a clear standard will mean that their values will be effectively arbitrary. Other examples of heterogeneous systems where this situation may hold include:

- Exploration robots with a mix of capabilities
- Cognitive architectures without a "master attention"
- A long-deployment sensor network, where nodes may develop significant differences in resources due to partial failures or upgrades

If each agent maintains its own list of topics, then agreement on topics is a consensus problem. Consensus is subject to many impossibility results showing that perfect consensus can only occur when speed or robustness is sacrificed. For example, it is impossible to simultaneously provide consistency, availability, and partition tolerance[7]. Standard consensus algorithms choose perfection over speed and robustness, but in systems like the examples above, where topics may need to shift quickly on the basis of ongoing sensor input, speed and robustness may be worth a small degree of inconsistency. Note that for heterogeneous agents formal consensus may even be ill-defined, since the mapping of topics between agents may not be one-to-one, but that problem is beyond the scope of this paper.

This paper explores the problem of shared focus using an engineered emergence approach. We begin by defining the problem, then use an analogy between consensus and extinction to relate it to emergent dynamics recently discovered in the evolution of systems with spatial extent. I then present a distributed shared focus of attention mechanism derived from this analogy, and refine the mechanism with a series of behavioral surveys to determine reasonable parameter settings, producing a prescription for a robust best-effort agreement mechanism that may solve the shared focus of attention problem for heterogeneous agents.

1.1 Related Work

There is a long history of work on distributed consensus algorithms: a good survey can be found in [10]. Recent work has produced relatively light-weight reconfigurable quorum consensus algorithms (e.g. [11], [9], [13]), but these algorithms guarantee correctness at the expense of progress under adverse circumstances.

Less strict models of consensus are common in economics (see, for example [17]), but consider agents with homoge-

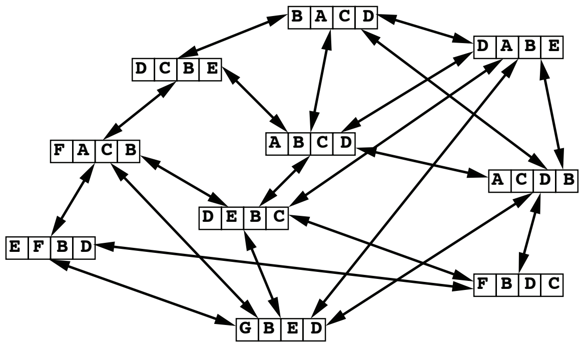


Figure 1: An example focus network of $n = 10$ devices with diameter 3, where each device has $f = 4$ foci. The network contains seven topics, A through G, of which only B and D are dominant.

neous structure and representation (though their information and preferences may differ). Likewise, the many studies of joint attention generally concern interactions between people or robots using essentially similar capabilities (e.g. [5], [12], [16], [8]).

Emergent behavior in natural systems has often been used as an inspiration for multi-agent engineering systems, though reliable approaches to doing so are only beginning to emerge. Ant-Colony Optimization[6] and other insect-based metaphors have been a subject of much interest (see, for example [3]). Another example is amorphous computing[1], which is significantly inspired by the coordinated behavior of cells during morphogenesis and repair of organisms.

Investigation into distributed attention also has some insectile roots[15]. More recently, Chu et al. [4] apply factor graphs to distributed task allocation in sensor networks.

2. PROBLEM DEFINITION

Distributed focus operates on a network of n devices, each containing a set of f foci (Figure 1). Each of the foci contains a *topic* that is currently a subject of attention, where a topic is simply a unique name local to a particular device. There is no constraint on the structure of the network. Each device has a stream of local requests for attention arriving, which it may service or discard.

Devices update periodically at roughly the same frequency as one another, and transmit the contents of their foci to their neighbors after each update. Each device contains a set of relations indicating which pairs of neighbor topic and local topic match. In this paper, I will consider only topics with one-to-one translations to all devices.

A distributed focus mechanism must balance four competing goals: dominance, fairness, agility, and longevity.

- **Dominance** means that a few topics occupy almost all of the foci on almost all of the agents in the network. Without dominance, agents are not participating in the same "conversation."
- **Fairness** means that any agent in the network has an equal chance to propose a new topic and have it become dominant. Without fairness, the system cannot respond reliably to unanticipated demands.
- **Agility** means that when the dominant topic shifts, it shifts quickly to a new dominant topic. Without agility, the network of agents cannot respond quickly to surprises.
- **Longevity** means that when a topic has become dominant, it is likely to stay there for a long time. Without some longevity, the "conversation" between agents is unlikely to stick to a topic long enough to be useful.

It is not possible to satisfy all of these goals perfectly, since some are fundamentally in conflict with one another: shifting between topics, for example, requires an intermediate state with lower dominance. We must instead seek an understanding of the ways in which goals go unsatisfied, and seek good metrics to characterize the trade-offs between them.

3. CONSENSUS IS LIKE EXTINCTION

To become dominant, a topic needs to invade devices throughout the network. Fair competition between topics, however, suffers from a symmetry breaking problem: if topics can invade one another, they are likely to thrash, but if they cannot invade one another, they are likely to end in deadlock. In both cases, dominance is unlikely.

We can gain insight into this problem from Rauch's work on spatial separation and group evolution[14]. Rauch uses a cellular automata host/disease model where each cell has three states: live, dead, and infected. There is a fixed death probability that an infected cell will die, a fixed growth probability that a live cell will spread to adjoining dead cells, and an evolvable infection probability that an infected cell will spread to adjoining live cells. Rauch discovered that, across a large range of parameters, the probability of infection evolves to sustain a stable but shifting population of all three states: local perturbations are damped out by the action of other areas, and changes in the global parameters change only the characteristic diameter of regions of cells with each state (See Figure 2).

Mapping this back to the shared focus problem, we can consider devices where a competing topic is invading others to be in the infected state, devices where it is subject to invasion to be in the dead state, and devices lacking the topic to be in the live state. I interpret Rauch's work as suggesting that thrashing may be a durable behavior across a wide range of algorithms. I also extract a suggestion for how to prevent thrashing: extinction events occur when the characteristic diameter of any state's region approaches that of the network, causing stability to break down.

I will use this observation to break the stalemate between competing topics. On a low probability coin flip, a spreading topic will become privileged for a short time, during which it cannot be invaded and usually succeeds in invading. This breaks symmetry, quickly spreading the topic throughout the network. Paradoxically, once the privilege ends, the topic will typically no longer be spreading, and can be quickly driven to extinction by its competitors.

4. DISTRIBUTED FOCUS MECHANISM

Using this insight, I have designed a distributed focus mechanism that allows a group of devices to balance between the goals set forth above. Figure 3 gives a LISP implementation for a device and its update function. I explain

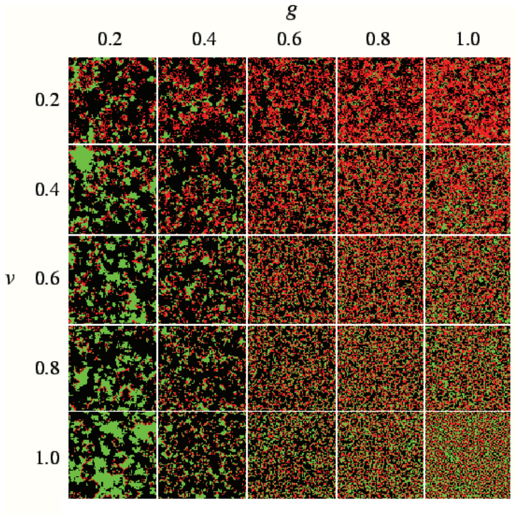


Figure 2: From [14], snapshots of evolutionarily stable behavior for a spatially extended host-pathogen system with various settings for pathogen virulence (v) and host reproduction (g). Parameter combinations produce a characteristic distribution of moving patches that do not stabilize unless patch size is close to the diameter of the network.

```
(defvar p_wait 1/2) (defvar policy 'follow)
(defvar k_p 2) (defvar p_ext 1)
(defvar t_priv 20) (defvar p_priv (/ k_p n t_priv))
(defclass foci ()
  ((foci :accessor foci)     ; array of foci
   (priv :accessor priv)     ; privileged focus countdowns
   (counter :initform -1)))  ; round-robin replacement pointer
(defmethod initialize-instance :after ((fc foci) &key f default)
  (setf (foci fc) (make-array f :initial-element default))
  (setf (priv fc) (make-array f :initial-element 0)))
(defun next-focus (f old) ; implement replacement policy
  (with-slots (foci counter) f
      (cond ((eq policy 'random) (random (length foci)))
            ((eq policy 'roundrobin)
             (mod (incf counter) (length foci)))
            ((eq policy 'follow)        ; change in synch
             (or (position old foci) (random (length foci)))))))
;; change foci, spike privilege w. low probability
(defun service-req (f ext new &optional old (p-req 0))
  (with-slots (foci priv) f
    (when (< (random 1.0) p_priv) (setf p-req t_priv))
    (let ((i (position new foci)))
      (if i
          (!= max (aref priv i) p-req)  ; possibly upgrade priv
        (when (or (not ext) (< (random 1.0) p_ext))
          (let* ((j (next-focus f old)) (pj (aref priv j)))
            (when (or (zerop pj) (< pj p-req)) ; don't override
             (setf (aref foci j) new (aref priv j) p-req))))))))
(defun update-foci (f) ; service first f requests randomly
  (with-set ((p (priv f))) (when (plusp p) (decf p))) ; leak
  (let* ((reqs (focus-requests f)) (oreq (randomize reqs))
         (clipreq (subseq (stable-sort oreq #'> :key #'fourth)
                    0 (min (length oreq) (length (foci f))))))
    (dolist (r clipreq) (apply #'service-req f r))))
```

Figure 3: Distributed focus implemented in LISP

only the key engineering decisions here; a full description is presented in Appendix C of technical report [2].

The mechanism assumes an arbitrary network of n devices, each with f foci, that update their foci at regular intervals. At each update, a device considers the set of topic requests that have appeared since the last update, either from local input or changes in a neighbor's foci, which are taken as implicit requests for a matching change.

Ordinarily, a device just serves as many topic requests as it can, so that topics spread rapidly through the network, thrashing when more than f collide.

Extinction events are induced by privileging topics. When a topic spreads, there is a chance p_{priv} that it will become privileged, leaking away its privilege over the next t_{priv} updates. During this time, the topic automatically wins over topics with less privilege, carrying its value as it spreads. Setting p_{priv} low and t_{priv} high ensures that privileged topics spread through the whole network but rarely conflict.

Three other parameters also influence the update mechanism: p_{wait} can break up local oscillation, p_{ext} can retard the spread of topics, and *policy* determines which topic is replaced. All potential topics are treated equally, deliberately avoiding use of salience measures in order to avoid systematically disregarding topics or worries about how to compare salience across heterogeneous agents. Salience might still be expressed via request frequency. Finally, mechanism costs are modest: parallel comparison gives $O(n^2f^2)$ hardware complexity, and serial update gives $O(f)$ time complexity.

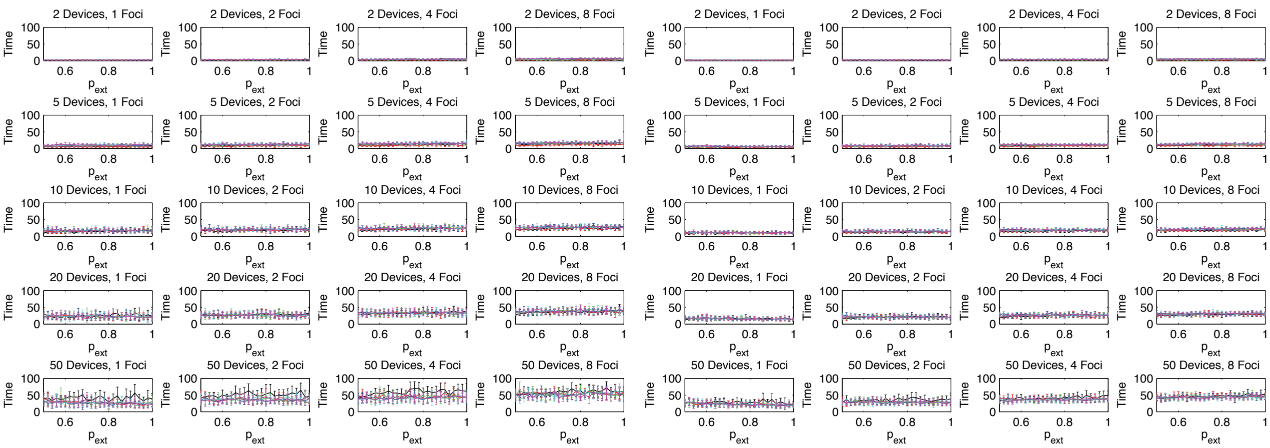
5. BEHAVIOR CHARACTERIZATION

The behavior of the distributed focus mechanism has been characterized with a mixture of analysis and experimental surveys of parameter space, detailed in Appendix C of technical report [2].

The surveys test parameter settings against most or all combinations of $n = 2$ to 50 devices with $f = 1$ to 8 foci each, updating synchronously or randomly, and arranged in a grid, complete network, or random network with connection probability of 0.2, 0.4, 0.6 or 0.8.

Figure 4 shows a tableau of summarizing the results of one such survey, a set of 6240 data points collected to determine the effect of p_{ext}, holding the other parameters fixed at $t_{priv} = 20$, $k_p = 2$ (which sets p_{priv}), $p_{wait} = 0.5$, and the follow policy. Each graph in the tableau plots p_{ext} from 0.5 to 1.0 against time to convergence from an impulse of nf topics, for a particular combination of n, f, and update, plotting the six network structures in different colors on each graph. In no case is there a significant effect, and since $p_{ext} < 1$ slows down propagation, reducing agility, this survey suggests that p_{ext} can be eliminated (setting it to 1).

Other surveys determine that the mechanism operates generally as desired (though it is unproved whether network structure significantly affects fairness) and behaves reasonably under a wide range of parameter settings. Some results of note: t_{priv} should be approximately equal to the diameter, as predicted from the analogy to extinction. Convergence is not delicate with respect to replacement policy, synchronous vs. random update, p_{wait} or k_p: reasonable values include $p_{wait} = 0.25$, $k_p = 2$ and the follow policy. When tuned well, convergence appears to be proportional to diameter, and most time before convergence is spent eliminating the last few excess competing topics, also consistent with the analogy to extinction. Finally, when local topic requests appear relatively infrequently, at least on third of the foci contain dominant topics (present in $\geq 90\%$ of devices) each round, and the number of rounds a topic is dominant appears to scale in proportion to the number of foci and the time between new topic requests.



(a) Synchronous Update (b) Random Update

Figure 4: Tableau showing the results of a survey of distributed focus p_{ext} against convergence time.

6. CONTRIBUTIONS

I have introduced a distributed mechanism by which a group of heterogeneous agents can reach rough consensus on a shared focus of attention. This mechanism was created using engineered emergence techniques, beginning with an emergent phenomenon in spatially extended evolutionary models, mapping it to the problem at hand through an analogy between extinction and consensus, then refining the mechanism to find a broadly applicable set of parameters through a series of behavioral surveys. Finally, the smooth transitions found in behavior surveys indicate that the mechanism can be expect to degrade gracefully under adverse conditions.

Important questions remain unanswered, however, such as the degree to which fairness is affected by network topology, whether it is possible to prove the properties determined empirically via the behavioral surveys, and whether this mechanism can be adapted to agents which lack a one-to-one mapping between topics. This mechanism shows promise as an alternative to traditional consensus systems when agility is valuable and precision is not important, and might be applied immediately to problems in modelling intelligence, cognitive architectures, robotic teams, and sensor networks.

7. ACKNOWLEDGEMENTS

This work was partially funded by NSF Grant #6898853.

8. REFERENCES

[1] H. Abelson, D. Allen, D. Coore, C. Hanson, G. H. y, T. Knight, R. Nagpal, E. Rauch, G. Sussman, and R. Weiss. Amorphous computing. Technical Report AIM-1665, MIT, 1999.

[2] J. Beal. Learning by learning to communicate. Technical Report CSAIL-TR-2007-042, MIT, 2007.

[3] E. Bonabeau, M. Dorigo, and G. Theraulaz. *Swarm Intelligence: From Natural to Artificial Systems*. Oxford University Press, 1999.

[4] M. Chu, J. Reich, and F. Zhao. Distributed attention for large video sensor networks. In *IEE Intelligent Distributed Surveillance Systems*, 2004.

[5] M. Doniec, G. Sun, and B. Scassellati. Active learning of joint attention. In *IEEE/RSJ International Conference on Humanoid Robotics*, 2006.

[6] M. Dorigo. *Optimization, Learning and Natural Algorithms*. PhD thesis, Politecnico di Milano, Italy, 1992.

[7] S. Gilbert and N. Lynch. Brewer's conjecture and the feasability of consistent, available, partition-tolerant web services. *Sigact News*, 2002.

[8] F. Kaplan and V. Hafner. The challenges of joint attention. *Interaction Studies*, 7(2):135–169, 2006.

[9] M. Liu, D. Agrawal, and A. E. Abaddi. On the implementation of the quorum consensus protocol. In *Parallel and Distributed Computing Systems*, 1995.

[10] N. Lynch. *Distributed Algorithms*. Morgan Kaufman, 1996.

[11] N. Lynch and A. Shvartsman. RAMBO: A reconfigurable atomic memory service for dynamic networks. In *DISC*, pages 173–190, 2002.

[12] Y. Nagai, K. Hosoda, and M. Asada. How does an infant acquire the ability of joint attention?: A constructive approach. In *Third International Workshop on Epigenetic Robotics*, pages 91–98, 2003.

[13] D. Peleg and A. Wool. The availability of quorum systems. *Information and Computation*, 123(2):210–223, 1995.

[14] E. M. Rauch, H. Sayama, and Y. Bar-Yam. Dynamics and genealogy of strains in spatially extended host-pathogen models. *Journal of Theoretical Biology*, 221:655–664, 2003.

[15] S. Reed and V. Lesser. Division of labor in honey bees and distributed focus of attention. Technical Report 80-17, Dep. Comput. Inform. Sci., Univ. Massachusetts, Amherst, 1980.

[16] J. Triesch, C. Teuscher, G. Deak, and E. Carlson. Gaze following: why (not) learn it? *Developmental Science*, 9(2):125–147, 2006.

[17] A. Urbano and J. Vila. Computational complexity and communication: Coordination in two-player games. *Econometrica*, 70(5):1893–1927, 2002.