Swarm Intell (2010) 4: 199–220
DOI 10.1007/s11721-010-0042-8

Collective unary decision-making by decentralized multiple-robot systems applied to the task-sequencing problem

Chris A.C. Parker · Hong Zhang

Received: 9 June 2009 / Accepted: 16 April 2010 / Published online: 14 May 2010

Abstract When a complex mission must be undertaken, it often can be simplified by dividing it into a sequence of smaller subtasks, which are then completed in order. This strategy implicitly requires a system to recognize the completion of each subtask and make the decision to begin work on the next one. Decentralized multiple-robot systems can tackle many tasks, but their behavior is typified by continuous responses to stimuli. Task sequencing, however, demands a controlled, self-induced phase change in collective behavior—working on one task one moment and then on a different task the next—which is nontrivial for an emergent system. The main contribution of this study is a collective decision-making framework for decentralized multiple-robot systems that enables such a system to cooperatively decide that a current task has been completed and thus focus its attention on the next one in a sequence using only anonymous local communication. Central to the framework is the use of consensus, whereby task sequencing is delayed until a prespecified proportion of a system's robots agree that the current task is complete, reducing the likelihood of premature decisions. Two low-cost consensus estimation strategies are presented, both of which are practical for the extremely simple robots that are expected to compose large decentralized systems. Experiments in simulation and with real robots demonstrate that the proposed decision-making framework performs as predicted. Although the specific application of collective decision-making in this work is the cooperative task-sequencing problem, the proposed decision-making framework potentially has many additional applications.

Keywords Multiple-robot systems · Swarms · Decentralized control · Behavior-based control · Decision-making

C.A.C. Parker (✉) · H. Zhang
Dept. of Computing Science, University of Alberta, Edmonton, AB, Canada
e-mail: parkerca@ieee.org

Present address:
C.A.C. Parker
Dept. of Mechanical Engineering, University of British Columbia, Vancouver, BC, Canada

1 Introduction

System-level decision-making is an important cognitive operation for multiple-robot systems and swarms. A multiple-robot system (MRS) is a robotic team in which the individual robots cooperate to complete shared tasks. A particularly important class of MRS is the *decentralized* MRS (dec-MRS), in which none of the robots have authority over any of their teammates, and thus all of the robots are considered equally important. Dec-MRS are attractive because it is believed that they will be more immune to hazardous environments and less affected by individual robot failures than centralized or hierarchical systems (Cao et al. 1997). This is because none of the robots in a dec-MRS are critically important, so the failure of any particular robot would be unlikely to bring about the failure of an entire system. Furthermore, because the local, parallel operation of a decentralized system is less affected by the computational and communicative bottlenecks common in centralized systems, algorithms designed for them are likely to scale well to large robotic populations.

Dec-MRS design usually takes place at the microscopic level of the individual robots, since it is their local peer-to-peer interactions that give rise to a system's externally observed macroscopic behavior. However, when a system is deployed to carry out a mission, it is more convenient to consider the system macroscopically, as a single intelligent entity, as this point of view simplifies mission specification. Algorithms that enable designers to abstract away from the microscopic point of view provide the illusion of centralized control while simultaneously retaining the flexibility and resilience of the decentralized architecture.

This work focuses on decision-making by entire dec-MRS, an operation whereby a system cooperatively and unanimously makes an intelligent decision. Specifically, we describe decisions in which a single proposed alternative to the *status quo* is collectively accepted or rejected. Such *unary decisions* have many applications, including the sequencing of collective behaviors by dec-MRS. Decision-making, however, is a fundamental operation for autonomous systems, so the framework presented by this work is likely to find many uses beyond the specific focus of this study. Unlike previous decentralized collective decision-making strategies, the proposed approach yields collective decisions that are both unanimous and episodic (i.e., they have clear beginnings and endings), and the results of the collective decisions are known by the individual robots. Together, these characteristics mean that a dec-MRS could make a decision and then move on with each member of the system able to predicate its future actions on the collective decision's outcome.

1.1 The task-sequencing problem

Stepping a decentralized multiple-robot system through a sequence of tasks, although conceptually similar to sequencing multiple behavior-based controllers (Brooks 1991; Kube and Zhang 1996) in a single robot, is much more complicated. Not only must the individual robots step through a sequence of subtask controllers, but each robot also must synchronize its own task transitions with those of its teammates. If the robots simply were to base their individual decisions to continue working on the current task or begin the next one on the responses of their own unique sensors, they would be unlikely to synchronously transition through their collective mission's sequence of tasks. On the contrary, some robots would likely begin subsequent tasks prematurely, whereas others would continue to work on tasks for too long. Without some form of system-level coordination, the robots of a dec-MRS will tend to spread themselves across a sequence of tasks. Not only would this reduce the speed with which each task was completed, but also might lead to destructive interference between robots working on different tasks in a sequence, degrading the system's performance or leading to a complete mission failure (Parker and Zhang 2008). Instead, a dec-MRS should make a collective decision to begin work on its next task, and it is this kind of collective decision-making that we address in this work.

This work addresses the task-sequencing problem in decentralized systems by proposing a general-purpose decision-making framework that enables a dec-MRS to make collective decisions through the use of consensus estimation and quorum testing. Section 2 discusses related work. The framework is described in Sect. 3, as is the experimental domain used in this study: site preparation by blind bulldozing. A central component of our approach is the use of consensus to guide collective decisions, and Sect. 4 outlines two low-cost consensus estimation strategies. Experiments applying our decision-making framework to an instance of the task-sequencing problem are described in Sect. 5, and results of this empirical investigation are presented in Sects. 5.1 and 5.2. The work then closes with some summarizing conclusions in Sect. 6.

2 Related work

The decision-making framework presented in this work is based on the behaviors of honeybees and *Temnothorax* ants. Such biological inspiration has a long history in MRS research, dating back to Grey Walter's robotic tortoises (Holland 1997). More recently, the collective decision-making of natural social systems has inspired decision-making strategies for dec-MRS. Wessnitzer and Melhuish (2003) presented a biologically inspired behavior that enabled a swarm of robots to decide which of a pair of targets to collectively pursue. Garnier et al. (2005) applied cockroach social aggregation to dec-MRS. The robots were able to collectively choose a site at which to gather that was large enough to shelter all of them using simple local behaviors. Schmickl et al. (2009) described a different aggregation strategy inspired by honeybee behavior. Our previous works also have investigated collective decision-making in dec-MRS, focusing on decisions in which the single best alternative from a set of candidates is unanimously selected by an entire system (Parker and Zhang 2004, 2009). Those works, like the current one, were inspired by honeybee and *Temnothorax* ant behaviors. What sets our proposed decision-making behavior apart from those of other works is that the group decisions do not continually evolve but eventually terminate. Furthermore, following the completion of a collective decision using our proposed framework, each of the robots knows the decision's outcome, and thus it can be used to synchronize their individual behaviors with the collective state of their system.

Collective decision-making in large systems is constrained very differently than in smaller systems in which centralized/hierarchical architectures sometimes can be more practical. Decision-making by smaller MRS is exemplified by play selection in robotic soccer teams. In this high-speed non-episodic robotic team sport, it is important for the robots to agree on the particular play to implement (Bowling et al. 2004; Vlassis et al. 2004). The small sizes of robotic soccer teams allows them to employ centralized control strategies or high-bandwidth communication to optimize play selection. Computational and communicative bottlenecks make such approaches impractical for system-level decision-making in large dec-MRS. As a result, it is unlikely that advances made in the coordination of small MRS could be applied to the kinds of systems upon which we focus in this work.

Dynamic role-assignment within a dec-MRS is another form of collective decision-making that can be applied to large systems. Here, the collective task can be decomposed into a parallel set of simpler roles rather than a sequence. For example, the robots of Ijspeert et al. (2001) cooperatively removed long pegs from the floor holes of their environment.

Because the pegs were too long for a single robot to remove, another robot would have to provide a peg's initial lifter with assistance. The pulling and assisting roles were not statically assigned but adopted by the robots in response to the local needs of the collective task. A more sophisticated implementation of dynamic role assignment was described by Chaimowicz et al. (2002) in a collective foraging/transport domain. In that work, a team of robots searched their environment for objects to return to a goal area. However, similarly to the work of Ijspeert et al. (2001), the objects were too massive for an individual robot to move. By decomposing the individual robots' controllers into a series of subcontrollers that were switched on and off in response to their own sensory perception and messages from teammates, the system was able to tailor the allocation of robots to roles in response to the current state of the collective task. Nouyan et al. demonstrated that a system composed of many identical (i.e., homogeneous) robots could distribute its individual members amongst several heterogeneous roles to complete a complex collective mission (Nouyan et al. 2009). This work is particularly noteworthy as, like ours, their results were demonstrated with real robots. Other works have presented market-based strategies (e.g., auctions) to dynamically allocate roles amongst the robots (or other kinds of agents) in a cooperative system (Dias et al. 2006). Roles are "bought" and "sold" by the individual robots, who bid on them. Robots that have purchased a particular role, allocating a portion of their finite resources to its execution, are free to auction it off to other robots that can complete it at a lower cost. As a result, a matching of robots to roles will evolve that maximizes the performance of the individuals while minimizing their costs with respect to the economic rules set out by the system's designer.

The matching of robots to roles that emerges out of a market-based strategy is a kind of collective decision, but the individual robots cannot predicate their own decisions on the result of the entire collective decision since they have only incomplete knowledge of it. Furthermore, the available tasks are executed simultaneously in the task allocation domain. In this work, collective decisions serve to synchronize the execution of a particular task (or set thereof) by an entire dec-MRS. In the task-sequencing domain, tasks are executed one at a time, and *all* of the robots work on the same task simultaneously. Of course, it need not be a question of task sequencing *or* task allocation, since both are important and might be present in a single system. Tasks in a mission's sequence could be decomposed further into several parallel roles, necessitating dynamic role assignment to allocate the individual robots appropriately. In this scenario, a collective task transition would signal to the individual robots a change in the set of tasks to dynamically allocate.

3 Collective unary decision-making applied to the task-sequencing problem

Our focus in this work is on collective decisions to reject aspects of the *status quo*, decisions that we refer to as *unary decisions*. Collective unary decision-making allows the individual robots of a dec-MRS to update their shared knowledge in a coordinated fashion, enabling them to proceed as a cohesive entity in a dynamic environment. For example, rejecting the belief that the Sun is above the horizon might trigger the individual robots to begin nighttime operations. Rejecting the hypothesis that the current group task is incomplete signals that work on the next task should commence. The decision-making behavior proposed by this work makes collective unary decisions by pooling many robots' opinions, resulting in a more stable collective behavior.

All inter-robot communication by the individual robots over the course of a collective unary decision is assumed to be very short range, and we further assume that the robots are

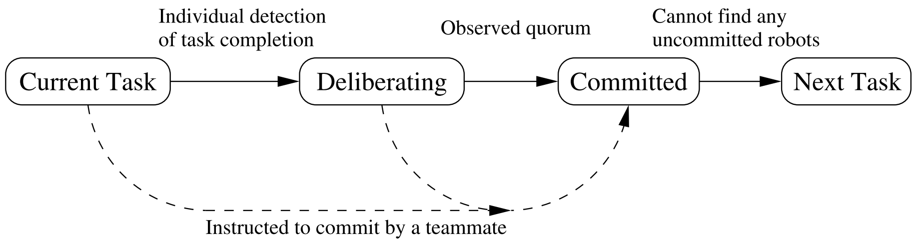


Fig. 1 The cooperative task sequencing framework proposed by this paper inserts two additional behavioral states between each task in a sequence that has been assigned to a dec-MRS: *deliberating* and *committed*. When a robot working on the current task decides that the task is complete, it enters the deliberating state. Deliberating robots query their teammates to estimate the proportion of them that also believe that the current task is complete. Once a deliberating robot believes that a quorum of its teammates agree that the current task is complete, it enters the committed state. Committed robots instruct encountered teammates to enter the committed state. Once a committed robot no longer can find uncommitted teammates, it exits the committed state and begins to work on the next task

free to wander about amongst each other. By using very local transmissions, the communication graph of a dec-MRS will be very disconnected, with single-hop peer-to-peer links between random pairs of robots stochastically being made and then broken as robots encounter each other and then move apart. In the unlikely event that a robot simultaneously receives messages from two or more of its teammates, we assume that the messages collide destructively and that neither will be heard, although assumptions about the nature of message collisions do not significantly alter the overall behavior of the proposed decision-making framework.

Unary decisions are made with the addition of two robot behavioral states: *deliberating* and *committed*. A decision begins once one of the robots in a dec-MRS decides that some component of the shared state of its system is no longer true. In the context of collective task-sequencing, the piece of shared knowledge in question is the assumption that the current task is incomplete. Refer to Fig. 1. As the individual robots work on the current task, they continually estimate its progress. As individual robots come to believe that the current task is complete,[1] they enter the deliberating state. Because each robot monitors the progress of the current task independently, they will tend to enter the deliberating state at different times.

Deliberating robots anonymously communicate with the teammates that they encounter one-by-one, asking them whether or not they also believe that the current task is complete. Queried robots respond to these messages either affirmatively or negatively. Each deliberating robot uses the responses to its queries to estimate the proportion of its teammates that also agree that the current task is complete. Once a deliberating robot believes that a quorum of its teammates believe that the current task is complete, it will enter the second of the two decision-making states: committed.[2]

[1] That is, they adopt the belief that the collective assumption that the current task is not yet complete (the *status quo*) is *false*.

[2] Note that a second quorum test could be employed to make the transition from deliberation to commitment more robust, by preventing a deliberating robot from committing until it believes both that a quorum of its teammates agrees that the current task is complete *and* that a quorum of its deliberating teammates agree that the first quorum has been satisfied. Such *compound quorum testing* is discussed in greater detail in Parker (2009).

Committed robots do not query their teammates for their opinions like deliberating robots, since they are *committed* to the belief that the current task indeed is complete and that work on the next task should commence. Their goal is to ensure that all of their teammates know that the current task is complete and commit before they themselves begin to work on the next one. They do this by sending commit-messages to each teammate that they encounter. Robots still working on the current task or in the deliberating state that receive a commit-message immediately enter the committed state and respond to the commit-message with an acknowledgement. Robots already in the committed state ignore altogether any commit-messages that they receive. Once one robot enters the committed state, commit-messages will flood system-wide via peer-to-peer spatial gossip (Kempe et al. 2001).

As more and more robots enter the committed state, fewer acknowledgements to commit-messages will be sent, ceasing altogether once all of the robots have committed. Committed robots measure the amount of time that elapses from the reception of one acknowledgement to the next. A committed robot will exit the committed state and begin to work on the next task once the time since it last received an acknowledgement to a commit-message exceeds a preset limit called the *commitment timeout*, denoted T_C. The commitment phase of a unary decision serves to synchronize a dec-MRS's transition from the current task to the next, whereas the consensus estimation and quorum testing of the deliberation phase delay the collective transition until a sufficiently large proportion of the robots agree that the current task is complete.

This decentralized approach to collective decision-making, like our earlier work on best-of-N decision-making described in Parker and Zhang (2009), was inspired by the colony-level decision-making of *Temnothorax* ants and honeybees (Franks et al. 2002). However, in a unary decision, there are no alternatives to compare as there are in a best-of-N scenario. Instead, the goal is to collectively decide "whether or not" a proposed alternative to the *status quo* should be adopted by the entire system, an operation that necessitates a modification to the deliberating behavior. We also more closely examine the performance of the commitment phase of a collective decision in this work than we did in our previous studies.

3.1 Site preparation by blind bulldozing

Site preparation, a collective task described by Huntsberger et al. (2000), consists of clearing an area of debris to allow other operations to take place. Because it is a relatively coarse task, site preparation work should end before work on subsequent tasks commences. Blind bulldozing (Franks and Deneubourg 1997) is a site preparation algorithm for dec-MRS that was inspired by the nest construction behavior of *Temnothorax* ants (Parker and Zhang 2006). An initial clearing is expanded by a dec-MRS as the individual robots repeatedly push back the debris surrounding the clearing, reorienting to a random heading whenever the force of debris on their plows exceeds a preset threshold or a teammate is encountered. Each push against the debris at the edge of the clearing expands the prepared site somewhat, and the subsequent random reorientations of the robots uniformly distribute their plowing around the clearing's perimeter. As a result, the cleared area swells, adopting a circular shape.

The task is complete once the prepared site achieves a desired size, after which blind bulldozing should cease and subsequent work begin to prevent cross-task interference. Site preparation and secondary construction thus are *mutually exclusive* tasks (Parker and Zhang 2008). The individual blind bulldozing robots can estimate the size of the cleared area by measuring the distance that they travel between each reorientation (Parker and Zhang 2006). For example, if a robot can travel a given distance in a straight line without encountering an obstacle, it can conclude that the diameter of the site is at least as long as its path. However, the robots' paths are random, and so each one will tend to conclude that the site preparation task is complete at a different time.[3] A collective decision to halt site preparation and begin work on the subsequent task would solve this problem.

4 Consensus estimation and quorum testing

Central to our proposed task sequencing framework is the use of consensus[4] as a *social cue* to guide collective task transitions. The individual deliberating robots estimate the proportion of their teammates that agree that the current task is complete and compare their estimates to a threshold called the *quorum*. Without knowing how many teammates it has, a robot cannot include its own opinion in its estimate of consensus (Parker and Zhang 2009). Consequently, the deliberating robots compute estimates of what we call the *apparent consensus*. This is defined by (1), in which C_a is the apparent consensus, N is the population size of the dec-MRS, and N_d is the number of robots that believe that the current task is complete:

$$C_a \equiv \frac{N_d - 1}{N - 1}. \tag{1}$$

Note that apparent consensus is strictly less than *true consensus*, $C_t \equiv \frac{N_d}{N}$, when $N_d < N$, but that both quantities vary over the same range, [0, 100%]. In general, the difference between C_a and C_t is insignificant for all but the smallest systems. Deliberating robots test quorum by computing estimates of C_a and then comparing their estimates to the *quorum threshold*, $Q \in [0, 100\%]$. Each individual robot's estimate of apparent consensus is independent of its teammates' estimates. We use $\tilde{C}_a$ to denote a particular estimate of apparent consensus in order to differentiate it from C_a as defined by (1). The remainder of this section describes two low-cost strategies to compute $\tilde{C}_a$ using anonymous local communication and closes with a comparison of the two strategies with respect to their performance. A precursor to analog consensus estimation as presented here was described in Parker and Zhang (2008), whereas digital consensus estimation was introduced in Parker and Zhang (2009).

Quorum testing to regulate collective behavior is very common in natural decentralized systems, and so its use by behavior-based robots makes good sense. Honeybees and at least two species of *Temnothorax* ant employ quorum tests to guide the collective decisions that their colonies use to select a new nest site (Franks et al. 2002; Pratt 2005; Seeley and Visscher 2003). Indeed, it is the reported behavior of these insects that inspired our decision-making research detailed in Parker (2009). Even simple bacteria are now known to employ quorum tests to regulate their collective behaviors (Miller and Bassler 2001; Waters and Bassler 2005). In *Temnothorax* ants, it appears that the individual ants estimate consensus via their rate of encounter with their teammates (Pratt 2005), and it is this strategy that we propose here with some modifications to better suit it to the somewhat different strengths and weaknesses of the simple mobile robots that often compose dec-MRS. Biologically inspired quorum testing has also been applied to ad-hoc networking problems by Peysakhov et al. (2006).

[3] Factors such as wheel slippage and irregularities in the shape of the cleared area introduce additional sources of error into the individual robots' evaluation of task completion.

[4] Note that our use of the term *consensus* differs somewhat from the manner in which it is used in works such as Olfati-Saber et al. (2007).

4.1 Well-stirred systems

It is assumed in this work that a dec-MRS implementing the proposed decision-making framework will be *well-stirred*, meaning that the individual robots that compose the system will be equally likely to encounter any of their teammates as they wander about and that each teammate encounter is a statistically independent event (Parker and Zhang 2009). This is a simplifying assumption made for the purpose of analysis. Strictly speaking, most real systems are not perfectly well-stirred, as the probability of a robot encountering a particular teammate in the near future will be higher if that teammate recently was encountered by the robot. However, the interactions of a poorly stirred system can be made more uniformly random by instructing the individual robots to ignore teammates that are encountered too soon after the time of their last teammate encounter, thereby allowing more time for the robots to relocate between those encounters that are counted. In other situations, a system might be well-stirred only at a specific location, such as a rendezvous point to and from which the robots asynchronously arrive and depart. The goal location in the collective foraging domain of Goldberg and Matarić (1997) is one example of such a rendezvous point.

Many natural decentralized systems stir themselves. This is observed in many social insect species, and stirring is commonplace in schools of fish (Krause and Ruxton 2002). Stirring can even take place in more rigid formations. For example, consider the familiar vee of geese migrating for the winter. Flying in a vee increases the energy efficiency of the group, but the lead goose encounters greater drag than its followers. In response, the geese take turns flying in the lead position, thereby sharing the load (Anderson and Franks 2001; Badgerow 1988). As a result, the formation stirs itself. The noise injected into a group by this stirring could be used to power cognitive operations, including the collective decision-making described in this paper. The use of physical movement to power collective cognitive operations makes particularly good sense for swarms of very small robots, for which Campbell et al. (2005) showed that moving can be more energy-efficient than explicit computation.

4.2 Analog consensus estimation

The first of the two consensus estimation strategies described in this work is referred to as *analog consensus estimation* (ACE). An analog implementation might at first seem quaint or outdated, but note that analog circuitry often can be made more compact and energy efficient than computationally equivalent digital circuitry, enabling even the simplest and smallest of robots (e.g., micro- or nano-robots) to take advantage of the contributions of this study.

We begin with a pair of exponentially decaying signals maintained within each robot: $q(t) = q(t_o)e^{-t/\tau}$ and $k(t) = k(t_o)e^{-t/\tau}$. These signals are referred to respectively as the *quorum index* and *kin index*. Unperturbed, both will decay to zero at a rate proportional to their instantaneous amplitudes and their shared time constant τ. If they are incremented at a regular interval by some constant, Δ, then their amplitudes will adopt the form of sawtoothed waves. The amplitudes of the indices will approach equilibria, the peak values of which depend only on the decay rate of the indices, the amount by which the indices are incremented, and the frequency with which these increments are made. Because $k(t)$ and $q(t)$ decay at the same rate and are incremented by the same amount, they can differ only in response to the rates at which they are incremented. As implied by its name, a robot will increment its kin index every time that it encounters a teammate (i.e., one of its kin). The quorum index is incremented only by deliberating robots, and only after they encounter a teammate that agrees that the current task is complete.

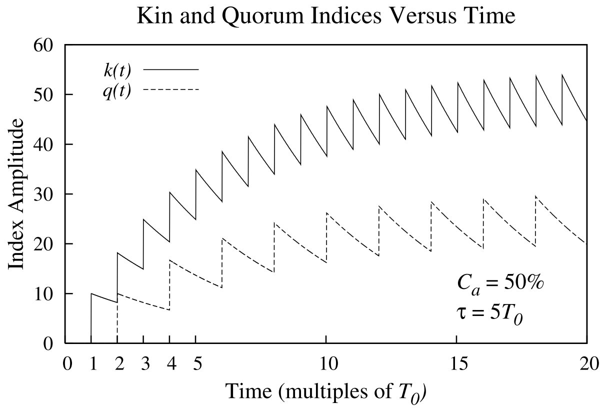


Fig. 2 This figure illustrates the principle behind analog consensus estimation (ACE). Each robot maintains its own quorum and kin index, both of which decay exponentially at the same rate. A robot increments its kin index every time that a teammate is encountered. The quorum index is incremented only by deliberating robots, after they have encountered a teammate that agrees that the current task is complete. The ratio of the peak equilibria of a robot's quorum and kin indices approximates the apparent consensus

Assume that each robot in a well-stirred dec-MRS will encounter one of its teammates every T_0 seconds on average. Each robot therefore will increment its kin index approximately every T_0 seconds. Each deliberating robot will tend to encounter an agreeing teammate every T_0/C_a seconds on average, and so they will increment their quorum indices at this rate. It was shown in Parker (2009) that the peak equilibria of the quorum and kin indices (q_{equ} and k_{equ}, respectively) are given by

$$\begin{aligned} q_{\text{equ}} &= \frac{\Delta}{1 - e^{\frac{-T_0}{\tau C_a}}} \approx \Delta \frac{\tau C_a}{T_0}, \\ k_{\text{equ}} &= \frac{\Delta}{1 - e^{\frac{-T_0}{\tau}}} \approx \Delta \frac{\tau}{T_0}. \end{aligned} \tag{2}$$

The ratio of a deliberating robot's two indices' peak values at equilibria, $\frac{q_{\text{equ}}}{k_{\text{equ}}}$, approximates the apparent consensus, C_a. In Fig. 2, the quorum index is incremented half as often as the kin index; its peak value at equilibrium is approximately 50% of that of the latter. The accuracies of the rightmost terms of (2) increase with τ, making the deliberating robots' estimates of C_a more precise as the indices are made to decay more slowly relative to the average rate at which teammates are encountered. The price of the improved accuracy achieved by increasing τ is that indices will take longer to reach their equilibria, increasing the amount of time required to compute $\tilde{C}_a$. Note that Fig. 2 is intended to illustrate the principle behind ACE. A robot's two indices would not in practice be incremented at perfectly regular intervals. In practice, the kin index should be allowed to reach equilibrium before beginning to collect teammate opinions. Otherwise, if the first teammate encountered by a robot also was deliberating, the robot's first estimate of apparent consensus would be 100%,[5] guaranteeing commitment regardless of the actual value C_a.

[5]Both $k(t)$ and $q(t)$ would have been zero prior to the reception of the opinion, and then both would be incremented by Δ, yielding $\tilde{C}_a = \frac{\Delta}{\Delta} = 100\%$.

All that remains for a deliberating robot to test quorum is to compare $\frac{q(t)}{k(t)}$ to the quorum threshold, Q. Equivalently, the quorum index could be compared to the kin index scaled by the quorum threshold (e.g., to test a quorum of 50%, $q(t)$ would be compared to $0.5k(t)$), an operation easily implemented in an analog system with a single operational amplifier in the form of a comparator. $q(t)$ and $k(t)$ could be represented by the voltages across the capacitors in a pair of identical RC circuits, and the voltages across the capacitors could be increased by Δ following appropriate teammate encounters by connecting them to a constant current source for a fixed period of time. The scaling of $k(t)$ by the quorum index is trivial, requiring only a resistive voltage divider. The elegant simplicity of analog consensus estimation and quorum testing is within the means of any mobile robot.

4.3 Digital consensus estimation

The second approach to consensus estimation described in this work is referred to as *digital consensus estimation* (DCE). The advantage of this strategy, beyond the fact that it is more readily implemented using a digital controller than ACE, is that the estimate of apparent consensus computed by it is not affected by variations in the rates at which the robots encounter each other. As was the case for ACE, the digital approach requires the individual deliberating robots to gather opinions from their teammates regarding the state of the current task (complete or not) as they are randomly encountered. Each teammate's opinion is treated as a binary variable: 1 if a queried teammate agrees that the current task is complete (e.g., it is in the deliberating or a later state), or 0 if the queried teammate is still working on the current task. Rather than being used to increment decaying indices, the opinions are inserted into a circular queue of size n, overwriting the oldest one. Each deliberating robot computes its own estimate of apparent consensus as the mean of the n values stored in its queue.

For example, consider a robot estimating C_a using the $n = 5$ most recently acquired teammate opinions. If the collected teammate opinions (from most to least recent) were 1, 0, 1, 0, 0, then the robot would compute $\tilde{C}_a = \frac{2}{5} = 40\%$. If the next teammate that this hypothetical robot were to encounter believed that the current group task was complete, a 1 would be inserted into the queue, and the new estimate of apparent consensus would be 60%. More teammate opinions will be averaged to estimate apparent consensus as n is increased, and thus the precision of $\tilde{C}_a$ will increase with n. However, increasing n also means that older opinions will be included in each estimate of C_a for longer, so each robot's $\tilde{C}_a$ will tend to lag the actual value of apparent consensus at any given moment. As in ACE, each deliberating robot independently tests quorum by comparing its own $\tilde{C}_a$ to the quorum threshold. A robot will conclude that quorum has been satisfied once it computes $\tilde{C}_a \geq Q$. The probability of a particular robot concluding that quorum has been satisfied based on n teammate opinions randomly sampled with replacement (uniformly distributed across all of the opinion-gathering robot's teammates) is given by

$$P(\tilde{C}_a \geq Q) = \sum_{i=\lceil Q\cdot n\rceil}^{n} \left[\binom{n}{i}(C_a)^i(1-C_a)^{n-i}\right]. \tag{3}$$

Refer to Parker and Zhang (2009) for a derivation of (3). It is important to remember that each of the N_d deliberating robots simultaneously estimates apparent consensus and tests quorum. The commitment phase of a collective decision will commence as soon as *one* of the deliberating robots computes $\tilde{C}_a \geq Q$. The likelihood of commitment occurring after each of the N_d robots collects n teammate opinions depends both on the population size of a

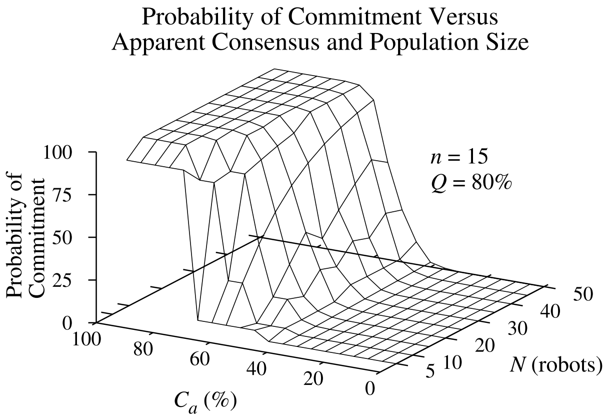


Fig. 3 Given the parameterization of a quorum test (n and Q), the likelihood of at least one robot committing increases with the population of a dec-MRS. This occurs because more robots estimate apparent consensus simultaneously as N increases (for a given C_a), and thus the probability of at least one robot overestimating C_a will increase. As this plot illustrates, $n = 15$ and $Q = 80\%$ would be a good parameterization to test a quorum of 40%, since the probability of commitment is low for $C_a < 40\%$

dec-MRS (N) and the apparent consensus. This overall probability of commitment is given by

$$\begin{aligned} P(\geq 1 \text{ robot commits}) &= 1 - \prod_{i \in \{N_d\}} \big(1 - P(\tilde{C}_{a_i} \geq Q)\big) \\ &= 1 - \left[1 - \sum_{i=\lceil Q \cdot n \rceil}^{n} \binom{n}{i} (C_a)^i (1 - C_a)^{n-i}\right]^{(\lfloor C_a(N-1) \rfloor + 1)}, \end{aligned} \tag{4}$$

where $\{N_d\}$ is the set of all deliberating robots and $\tilde{C}_{a_i}$ is the ith deliberating robot's estimate of apparent consensus (note that the various $\tilde{C}_{a_i}$ are independent and identically distributed random variables).

The relationship between the probability of commitment, N, and C_a is illustrated graphically by Fig. 3 for fixed values of n and Q. As N is increased, the probability of the commitment phase beginning at a lower value of apparent consensus increases. In this figure, $n = 15$ and $Q = 80\%$, yet the probability of commitment begins to rise once C_a exceeds approximately 40%. Increasing n or Q increases the effective quorum employed by a system (e.g., the apparent consensus at which a positive quorum test would be likely).

Given Fig. 3, an external observer of the deliberation phase likely would conclude that the quorum tested by the robots was less than the quorum threshold, Q. However, the discrepancy between the apparent consensus at which commitment becomes likely and the actual value of the quorum threshold is *not* a failure of the approach to consensus estimation and quorum testing presented by this work. Estimates of apparent consensus will always be somewhat noisy, with each of the N_d robots overestimating C_a approximately half of the time. The variance of a particular estimate of C_a inversely depends on n. Because a quorum test is a thresholding operation that induces a phase change in dec-MRS behavior (the beginning of the commitment phase of a collective decision), Q should be chosen to take the expected noise in $\tilde{C}_a$ into account. Therefore Q should be made larger than the desired quorum to make premature commitment sufficiently unlikely.

4.4 Performance of analog and digital consensus estimation

Sections 4.2 and 4.3 describe two different approaches to the measurement of apparent consensus, ACE and DCE. In this section, we show that the two approaches are largely equivalent in their performance when speed and accuracy are considered. The TeamBots simulator

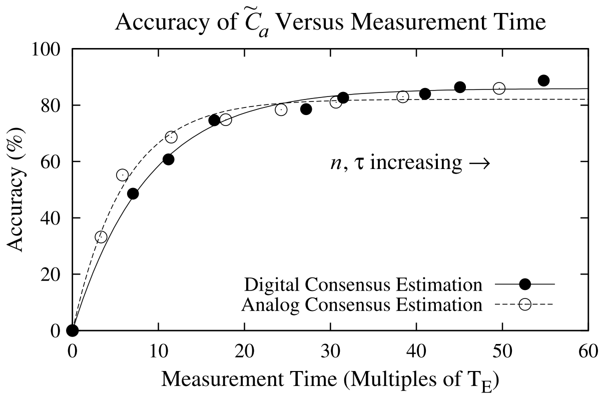


Fig. 4 This graph plots the tradeoff between the accuracy with which the robots estimate C_a and the time required by them to compute their estimates for different values of n or τ (increasing from *left* to *right*). This data was gathered from 40 simulations of a 15-robot system in which the apparent consensus was constant at 50%. DCE and ACE exhibit nearly the same accuracy for a given measurement time. Increasing n or τ increases the accuracy of $\tilde{C}_a$, but these increases incur the cost of an increasingly greater measurement time

(TeamBots 2005) was used to investigate the relationship between the time required for and the accuracy of apparent consensus estimation by robots in a dec-MRS. The simulation consisted of 15 robots randomly wandering about a circular environment. The deliberating population was fixed at eight robots (i.e., $N_d = 8$), thereby setting $C_a = 50\%$. Each of the deliberating robots' initial estimates of apparent consensus was 0%,[6] and these evolved as opinions were collected one-by-one from the teammates that were encountered. The simulation was repeated using several different values of n (for DCE) and τ (for ACE), and each configuration was repeated 40 times. For each configuration, the time required for each deliberating robot to estimate C_a and the accuracies of the robots' estimates were computed. Measurement time for each deliberating robot was computed as the amount of time required for its estimate of apparent consensus to reach the actual value of $C_a = 50\%$. The average accuracy of a robot's estimate of apparent consensus in the period after it first reached C_a was computed as

$$\tilde{C}_a \text{ Accuracy} = \left(1 - \frac{1}{m}\sum_{i=1}^{m}\frac{|\tilde{C}_a[i] - 0.5|}{0.5}\right) \times 100\%, \tag{5}$$

where $\tilde{C}_a[i]$ is the ith estimate of apparent consensus computed by the robot over this period. (5) assumes that the robot collected m teammate opinions in the period following the initial rise of its estimate of apparent consensus to 50%.[7] The average accuracy of apparent consensus estimation is plotted against the mean measurement time in Fig. 4 for both ACE and DCE.

The relationship between consensus estimation time and accuracy essentially is the same for ACE and DCE. The accuracy of $\tilde{C}_a$ rises rapidly with τ and n when these parameters are small. However, each increase in consensus estimation accuracy comes at the cost of

[6] When C_a was estimated via ACE, the robots were allowed to wander about for an initial period during which $N_d = 0$ to allow their kin indices to equilibrate.

[7] One estimate of apparent consensus is computed following the reception of each teammate opinion.

an increasingly longer measurement time. There also appears to be an equivalence between ACE and DCE with regards to n and τ. The accuracy/time-cost tradeoff of ACE with a given value of τ is very close to that of DCE when $n = \tau/T_E$, where T_E is the median period of time between consecutive teammate encounters by a robot. This equivalence makes sense, since a robot would be able to collect n teammate opinions in τ/T_E seconds on average, and so both approaches will tend to base their estimates of C_a on an equal number of teammate opinions when equivalently calibrated in this way. As a result, the conclusions drawn from Fig. 3 regarding the effective quorum for a given parameterization of consensus estimation is likely to apply equally well to both ACE and DCE. For more information, refer to Parker (2009).

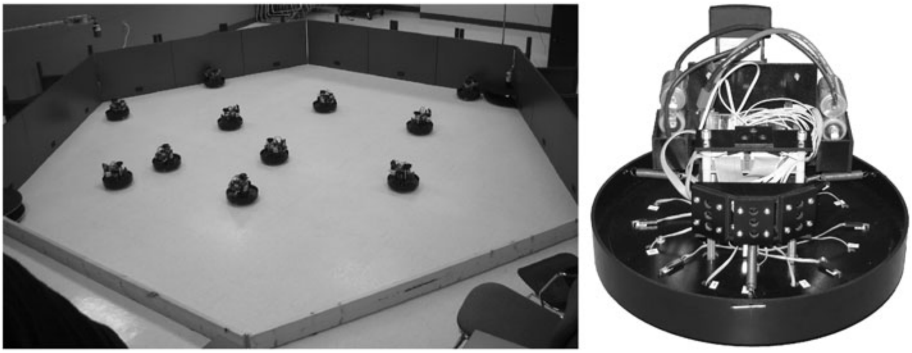
5 Experiments

A series of experiments were conducted both in simulation and using real robots to examine the performance of the proposed decision-making framework using blind bulldozing as the "current task." The experimental environments (both simulated and real) were convex arenas of a fixed size that resemble a blind bulldozing domain as the task neared its completion, the point at which its size would have ceased to expand. This static environment was chosen for our experimental trials because in it the robots' movements resemble those of a well-stirred system.

The simulated environment was a circular arena, 12 meters in diameter, whereas the physical environment was hexagonal with sides 2.75 meters in length. A photograph of the physical environment containing the robots is shown in Fig. 5(a). Throughout each collective decision, the robots moved according to the blind bulldozing behavior described earlier, driving in straight lines until they encountered an obstacle (either the arena's wall or another robot), at which point they would reorient to a new heading before resuming straight-line motion.

In the simulations, implemented using TeamBots, a robot (circular hulls, 0.5 meters in diameter) working on the blind bulldozing task would enter the deliberating state once it had traveled a path 11 meters in length without reorienting since its previous reorientation. The real robots (circular hulls, 0.3 meters in diameter) transitioned to the deliberating state after they had followed a 4-meter straightline path without reorienting. Note that our goal was not to duplicate the conditions of the simulated trials in physical domain. Indeed, the differences in the number of robots involved, their sizes relative to their environment, and the lengths of paths that initiate deliberation should alter the rate at which apparent consensus evolves, but not the overall success of the collective decisions. Both ACE and DCE were implemented in the simulated trials with n, τ, and Q varied over several values. The commitment timeout also was varied in a set of simulations to elucidate its impact on the unanimity of the commitment phase. Only DCE was used in the physical trials, since, as it has been and will again be seen, ACE and DCE behave similarly.

Teammate opinions were gathered and commitment spread by the individual robots during each decision using explicit local communication. Deliberating robots would send the robots that they encountered *query-messages*, the responses to which are called *vote-messages*: "yes" if the queried robot agreed that the current task was complete and "no" if it did not. Committed robots sent every robot that they encountered a *commit-message*, which would induce the recipient to enter the committed state unless it already had exited the collective



(a) (b)

Fig. 5 The experimental environment for the trials involving real robots was a hexagonal enclosure with sides 2.75 meters long. It is shown here (*left*) during a trial using 11 robots. The environment is static, convex form mimics a blind bulldozing domain near the completion of the task, when the expansion of the cleared area would have nearly halted. The simulated experiments were carried out in a similar environment, except that it was circular and contained 15 robots. The robots used to carry out the physical experiments described in this paper were custom built for our study of collective decision-making. Each robot (*right*) used its 360° bumper sensor to detect encounters with its teammates. At the rear of each robot is its 802.11B wireless interface, used for inter-robot communication

decision.[8] If an uncommitted robot (i.e., a robot in the current task or deliberating states) received a commit-message, it also would respond with an acknowledgment. Robots that had exited a decision (i.e., in the next task state) ignored any messages that they might receive.

Very local broadcast communication was used by the simulated robots, so they needed to be very close (0.05 meters, measured hull-to-hull) to each other to be able to communicate. Despite being anonymous and omnidirectional, the transmissions' extremely short range meant that most messages were received by a single teammate. Simultaneous responses to a message were discarded, simulating destructive collisions. The real robots, however, were restricted to using 802.11B wireless transceivers to communicate with each other. The range of the robots' radios was global in our experimental environment. Local communication therefore had to be simulated. Conceivably, a centralized server could have been used to coordinate the individual robots' communication so that a robot actually would communicate with the teammate that it had encountered, but this approach was deemed unnecessarily complex. Instead, each robot's program included a list of its teammates' IP addresses. When a robot encountered a teammate and communication was appropriate, it would select a teammate's IP at random (uniformly distributed across all of its teammates) and send it a message using TCP. The recipient of the message would respond, if necessary, to the original sender. As a result, two robots might not be in close proximity to each other when they communicated. Nonetheless, the overall pattern of the robots' random peer-to-peer exchanges (Parker and Zhang 2007) closely resembled those of the simulated robots that used short-range local communication.

[8] In a general task-sequencing context, it cannot be assumed that the robots that have begun to work on the next task will remain at the site of the current task, and thus in general they would not be present to respond to their deliberating or committed teammates' messages.

5.1 Simulation results

In this section, the results of the simulated experiments are presented and discussed. The real robots' performance is presented in Sect. 5.2.

5.1.1 *Consensus at time of commitment*

It is important to restate that the individual robots do not know the actual value of the apparent consensus, C_a. Each deliberating robot's behavior is predicated upon $\tilde{C}_a$, its estimate of it. Even though the individual deliberating robots compare their estimates of apparent consensus to the quorum threshold Q, the actual apparent consensus that coincides with the beginning of the commitment phase will tend to be less than Q. The consensus that coincides with the beginning of commitment is called the *observed quorum* to distinguish it from the value of the quorum threshold.

Figures 6 and 7 plot the observed quorum as a function of the quorum threshold for ACE and DCE, respectively, for several different values of τ and n. Regardless of the value of these parameters, increasing the quorum threshold increases the observed quorum. This demonstrates that the desired consensus necessary to trigger a collective task transition can

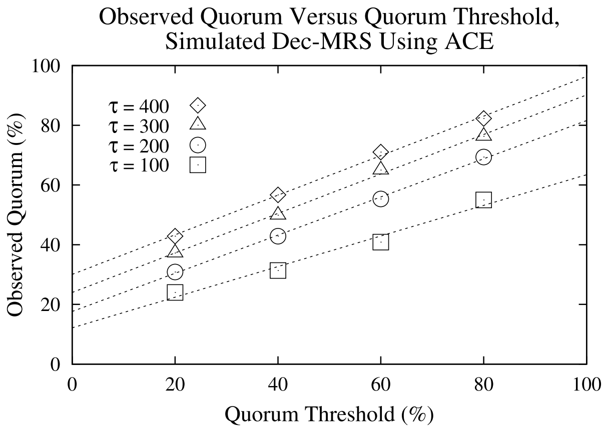


Fig. 6 The observed quorum for a collective decision is the true consensus that coincides with the onset of commitment. Increasing the quorum threshold or the value of τ increases the observed quorum of a collective decision when ACE is used to estimate the apparent consensus

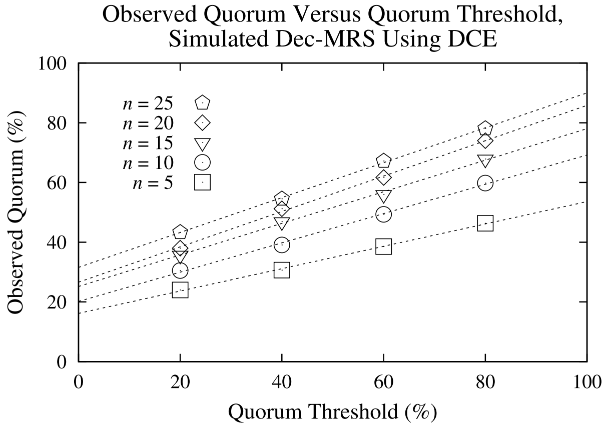


Fig. 7 This figure demonstrates that DCE is as effective as ACE (see Fig. 6). Not only does observed quorum increase with the quorum threshold, but also with n, the number of opinions used by each robot to compute $\tilde{C}_a$. Increasing the observed quorum has the effect of predicating a collective task transition on a greater number of independent assessments of the current task's state, making the collective decision more reliable

be varied via the quorum threshold. The slopes of the regression lines are less than unity, however. One explanation for this discrepancy is a subtle difference between the quorum threshold and the observed quorum. The individual robots test quorum by comparing their estimates of *apparent consensus* to the quorum threshold, whereas the observed quorum measured in terms of *true consensus*. The relationship between true and apparent consensus is given by

$$C_t = \frac{C_a(N-1)+1}{N}. \tag{6}$$

If the robots' estimates of apparent consensus were 100% accurate and could be computed instantaneously, Figs. 6 and 7 would be straight lines from (0, 6.7%)[9] to (100%, 100%). This clearly is not the case, and thus some other phenomenon is at work.

It is the interaction of the noise in the robots' estimates of apparent consensus, the time required by a robot to collect enough teammate opinions to compute its estimate of apparent consensus, and the rate at which the robots transition from blind bulldozing to the deliberating state that explains the appearances of Figs. 6 and 7.

The median period of time separating a robot's encounters with teammates in the simulations was $T_E = 21.5$ seconds. Each deliberating robot was able to query for a teammate opinion approximately every T_E seconds, regardless of the state of a decision. However, the rate at which robots entered the deliberating state was proportional to the number of robots still working on the current task, and thus the size of the deliberating population in our experimental domain prior to the onset of commitment is given by

$$N_d(t) = N\left(1 - e^{-t/\tau_d}\right). \tag{7}$$

τ_d for the simulated experiments ($N = 15$) was found to be 951.2 seconds using a least squares fit. N_d grew most quickly early in each trial, and so $\tilde{C}_a$ as computed by deliberating robots early in the trials tended to be less than C_a, lagging it significantly and thus elevating the observed quorum for low quorum thresholds. Increasing the accuracy of apparent consensus estimation by increasing τ or n makes this effect more pronounced. As a trial progressed, however, the growth of the deliberating population slowed relative to T_E, but more robots simultaneously were testing quorum. It therefore became more likely that the first robot to commit would do so prematurely due to noise in its estimate of C_a. Therefore, the relationship between the observed quorum and the quorum threshold illustrated by Figs. 6 and 7 is precisely what should be expected. Once the growth of the deliberating population had slowed relative to T_E, the observed quorum reported by Figs. 6 and 7 agrees with our earlier analysis. For example, a system composed of robots that compute $\tilde{C}_a$ using $n = 15$ and tests quorum by comparing $\tilde{C}_a$ to $Q = 80\%$ will be 50% likely to commit once the observed quorum reaches 63% (determined by interpolating the $N = 15$ curve of Fig. 3 and mapping to true consensus via 6). The mean observed quorum in the $n = 15$, $Q = 80\%$ trial was $67.9 \pm 8.5\%$, which agrees well with its predicted value.

5.1.2 Deliberation time

Increasing the quorum threshold, and as a result the observed quorum of a given collective decision, increases the consensus necessary within a dec-MRS to induce commitment, but it

[9] When $Q = 0$, the commitment phase will begin as soon as one of the robots detects the completion of the current task and enters the deliberating state. Therefore, the observed quorum for $Q = 0$ will be $\frac{1}{N} \times 100\%$, which is 6.7% for a 15-robot system.

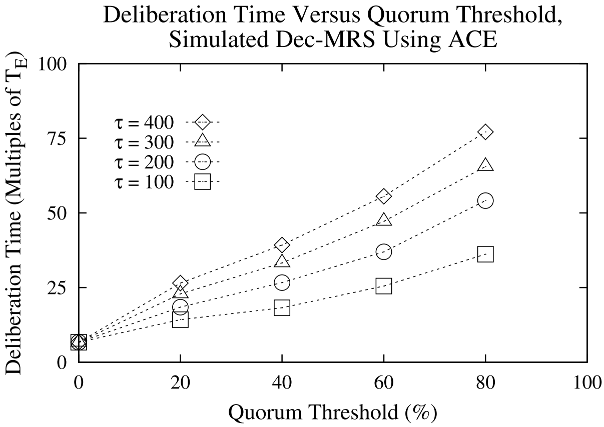


Fig. 8 Increasing the quorum threshold increases the observed quorum (Figs. 6 and 7), which delays the commitment phase until more robots agree that the current task is complete. Increasing the accuracy of consensus estimation (by increasing τ when using ACE) also increases the length of the deliberation phase because it makes the robots less likely to commit prematurely

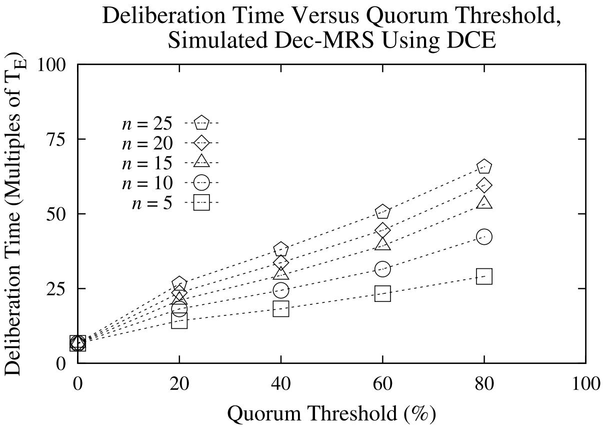


Fig. 9 As was the case for the ACE (see Fig. 8), deliberation time when using DCE increases with the quorum threshold, and the rate at which it increases with n (analogously to τ in ACE). The curves corresponding to the different values of n share a common y-intercept because quorum always is satisfied when $Q = 0$, regardless of the accuracy with which the individual robots estimate apparent consensus

also increases the amount of time that a system will take to deliberate prior to commitment. The deliberation time of a collective decision is defined as the amount of time that passes from the beginning of a trial until the beginning of the commitment phase. The mean deliberation times are plotted for several calibrations of ACE and DCE in Figs. 8 and 9. Variance is omitted for clarity, but standard deviations of 20% to 30% of the means were observed. The deliberation time when $Q = 0\%$ is constant regardless of the accuracy with which $\tilde{C}_a$ was computed, since a quorum of zero is always satisfied. In these figures, the deliberation time at the y-intercept is not zero as might be expected because the manner in which the deliberation time is defined includes the brief period at the beginning of each trial before any of the robots entered the deliberating state.

Increasing the quorum threshold Q increases the number of robots that must be in the deliberating state to induce commitment, and thus the length of the deliberation phase will increase as more time will be required for these additional robots to independently decide that the current task is complete. By increasing τ or n, the accuracy of consensus estimation is increased, which decreases the likelihood of one of the deliberating robots overestimating the apparent consensus and prematurely committing. It is because a more accurate quorum test reduces the probability of premature commitment that the deliberation time increases more rapidly with the quorum threshold for systems that employ a greater τ or n. It is not that the increased accuracy actually makes the deliberation more expensive per se. Instead, increasing these parameters increases the degree to which the observed quorum will resemble the quorum threshold employed by the individual robots.

5.1.3 Performance of the commitment phase

The final phase of a collective decision using the proposed framework is commitment. The role of commitment is to ensure the unanimity of a decision, so that all of the robots will begin to work on the next group task at approximately the same time. Earlier, mutual exclusivity of tasks was discussed: the constraint that certain pairs of adjacent tasks in a sequence cannot be worked on simultaneously. Indeed, it was the problem of mutual exclusivity that initially prompted our investigation of cohesive, decentralized task sequencing in dec-MRS. If it is assumed that robots halt their work on the current group task as soon as they enter the deliberating state and do not begin their work on the next group task until they have exited the committed state, then mutual exclusivity will be satisfied as long as all of the robots of a dec-MRS have entered the deliberating or committed states before any of the robots enter the next task state. Not all pairs of adjacent tasks are mutually exclusive. For adjacent tasks that are not mutually exclusive, the commitment phase would be said to succeed only if it was unanimous. By unanimous, we mean that once the commitment phase begins, the number of committed robots remains greater than zero until all of the robots have begun to work on the next task (i.e., the robots transition from the current task and deliberating states through commitment to the next task in a single wave). Unanimous commitment ensures that a system will not divide itself across adjacent tasks.

The behavior of the commitment phase of a decision is controlled by the commitment timeout, the amount of time that a committed robot will remain in the committed state without receiving an acknowledgment to one of its commit-messages. The percentages of the simulated trials that satisfied either unanimity or mutual exclusivity as the commitment

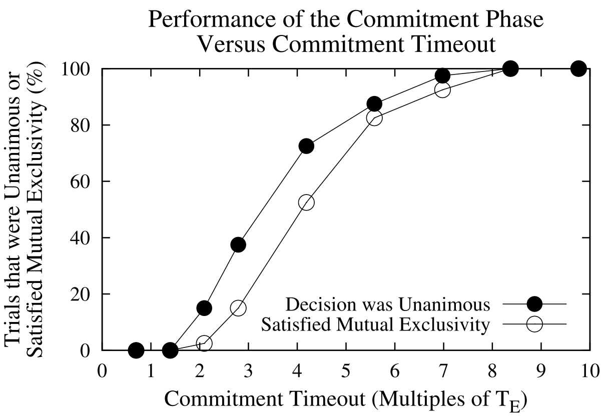


Fig. 10 Increasing the commitment timeout increases the likelihood that a collective decision to begin work on the next task will be unanimous. It also increases the probability of a decision satisfying mutual exclusivity, meaning that there will not be robots working on adjacent tasks simultaneously. Because committed robots send a commit-message to every teammate that they encounter, the amount of energy consumed by their transmissions will increase with the commitment timeout, as will the time that they spend wandering about before beginning work on the next task. The cost of increased reliability must be considered when choosing a value for the commitment timeout

timeout was varied (each data point was computed from 40 trials) are plotted in Fig. 10. Both curves exhibit the same behavior. Initially, increasing the commitment timeout yields an increasingly greater likelihood of these conditions being satisfied, with the satisfaction of mutual exclusivity lagging that of unanimity slightly. Diminishing returns are encountered, however, leading to the curves' S-shaped nature. Note that the additional cost of satisfying mutual exclusivity beyond that of unanimity is minor, requiring the commitment timeout to be increased by approximately T_E seconds to achieve the same reliability for this stricter criterion. Recall that T_E is the median period of time between robot's encounters with teammates, and so increasing the commitment timeout by this amount would allow each committed robot to send one more commit-message on average before beginning work on the next task.

The overall cost of the commitment phase in terms of the time delay between the satisfaction of quorum and the completion of the transition to the next task is roughly linear with the commitment timeout (Parker 2009). Robots remain in the committed state for a period of time that is linear with the commitment timeout, and each committed robot communicates with its teammates at a relatively constant rate. As a result, increasing the commitment timeout results in a linear increase in the energy expended by the robots (wandering about and sending commit-messages) for a decreasing gain in the reliability of the commitment phase. It would depend on the nature of the task at hand where the best trade-off between cost and reliability would be made.

5.2 Experiments with real robots

Because ACE and DCE behaved so similarly in the simulated trials, as was predicted earlier in Sect. 4.4, only DCE was implemented on the real robots. The robots that were used to carry out these experiments were custom built for the research that we have been carrying out investigating system-level decision-making in dec-MRS. A photograph of one of the robots can be seen in Fig. 5(b). Refer to Parker and Zhang (2009) and Parker (2009) for a more detailed description of this hardware. The remainder of this section presents the results of the decision-making experiments that were carried out with the real robots.

Figure 11 plots the mean observed quorum versus the quorum threshold for the real robots. Variance is omitted from this figure for clarity, but a nearly constant standard deviation of approximately 11% was present about each mean when $n = 5$, whereas the standard deviation in the $n = 15$ trials decreased steadily from 16% when $Q = 20\%$ down to 5% when

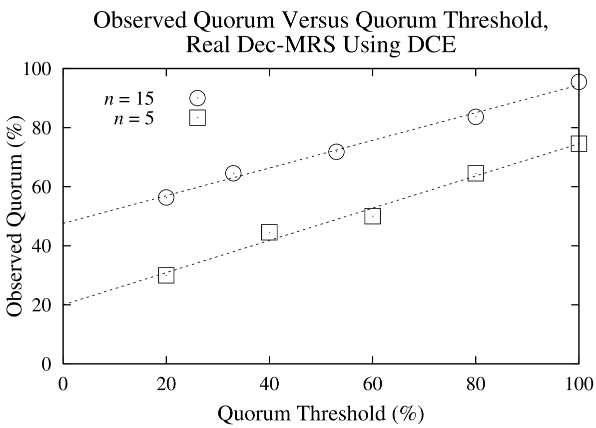


Fig. 11 This figure plots the relationship between the observed quorum and the quorum threshold observed in the experiments with real robots. As was the case in the simulations, the observed quorum increases both with the quorum threshold and the number of teammate opinions used to compute $\tilde{C}_a$

$Q = 100\%$. As was observed in the simulated trials, increasing the quorum threshold increased the observed quorum. Increasing the number of teammate opinions used by each robot to estimate C_a also increased the observed quorum in a given trial. In the physical experiments, the deliberating population tended to grow more rapidly relative to the rate at which the individual robots encountered their teammates than it did in the simulated trials. In the experiments with the real robots, $T_E = 9.0$ seconds and $\tau_d = 140.4$ seconds. Early in each of these trials, the deliberating population grew at twice the rate that it did in the simulations (0.70 robots$/T_E$ versus 0.34 robots$/T_E$), causing $\tilde{C}_a$ to lag C_a to a greater degree. In other words, while the real deliberating robots were gathering teammate opinions, the apparent consensus increased more rapidly than it did in the simulated trials, resulting in the greater observed quorums for given n and Q.

The specific nature of the tasks at hand and the environments in which they are carried out will influence the rate at which the robots enter the deliberating state (and whether or not it is appropriate for robots to revert from the deliberating state back to working on the current task, thereby decreasing C_a). Therefore the precise mapping of quorum threshold to observed quorum reported by our experimental results should not be taken as absolute. Rather, our results demonstrate that a decision's observed quorum will tend to increase with Q. This occurs because increasing Q causes the individual robots to delay commitment until a greater number of them have entered the deliberating state. Increasing n when using DCE (or τ in the case of ACE) to compute $\tilde{C}_a$ tends to increase the observed quorum of a decision, but for a subtly different reason. Increasing n (or τ) increases the precision of each robot's estimate of C_a. In domains in which C_a monotonically increases (such as the site preparation domain described in this work), an increased precision of consensus estimation reduces the probability of premature commitment due to overestimation of C_a, thereby increasing the observed quorum.

The deliberation time versus the quorum threshold of the real robots is plotted in Fig. 12, displaying the same relationship observed in the simulated trials. Increasing the quorum threshold increased the duration of the deliberation phase of the collective decision, and increasing n increased the slope of the deliberation time versus quorum threshold relationship. Once again, because the initial period in each trial before any of the robots had detected task completion is included in the measurement of the deliberation phase's length, the common y-intercept of the two curves is slightly greater than zero.

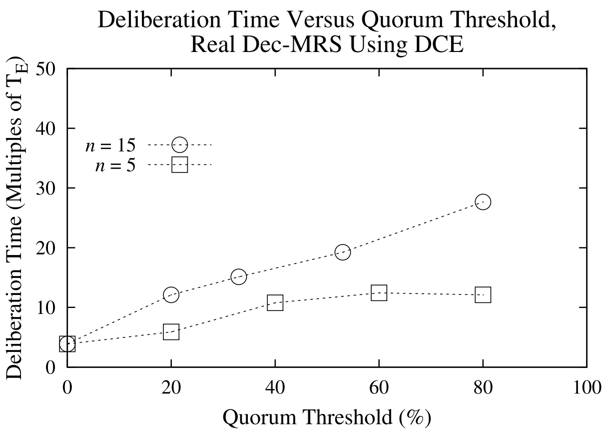


Fig. 12 The deliberation time for the real robots plotted here as a function of the quorum threshold is very similar to the relationship observed in the simulations. As the quorum threshold is increased, and with it the observed quorum, the deliberating robots delayed their commitment until more of their teammates agreed that the current task was complete

6 Conclusions

Behavior-based dec-MRS have been shown by many previous works to be capable of a variety of useful behaviors. However, designing behavior-based controllers for complex tasks can be very difficult, especially if robust collective behavior is required. By dividing complex tasks into sequences of more elementary behaviors, the controller design problem is greatly simplified. To take advantage of this attractive divide-and-conquer strategy, however, a system must be able to shift from one collective behavior to the next. In many scenarios, the task transitions must be synchronized, necessitating a collective coordination mechanism like the unary decision-making framework proposed by this work.

By measuring the consensus amongst their teammates and testing quorum, the robots that compose a dec-MRS can pool their individual decisions regarding the state of the current task to make a more reliable collective decision to begin the next one. The deliberating phase of the proposed decision-making framework delays collective task transitions until a quorum of robots agree that the current task is complete. Once the deliberating population has satisfied the quorum, the commitment phase induces the entire dec-MRS to transition to the next group task, preventing those robots that had yet to detect the completion of the current task from continuing to work on it after the rest of the robots had moved on. This collective decision-making strategy, inspired by the collective decision-making of ants and honeybees, provides the illusion of centralized control by making a dec-MRS behave as though is a single intelligent individual (a *superorganism*) while retaining all of the scalability and robustness of a decentralized architecture.

Two low-cost consensus estimation strategies were presented in this work, and both were shown to perform well. Consensus estimation and its use as a social cue have potential applications in other scenarios beyond task sequencing and collective decision-making, and we believe that consensus estimation and quorum testing are fundamental operations within dec-MRS and swarms. Experiments conducted in simulation and with real robots demonstrated that the proposed decision-making framework indeed enables a dec-MRS to make system-level decisions cooperatively.

References

Anderson, C., & Franks, N. R. (2001). Teams in animal societies. *Behavioral Ecology*, *12*(5), 534–540.

Badgerow, J. P. (1988). An analysis of function in the formation flight of geese. *The Auk*, *105*, 749–755.

Bowling, M., Browning, B., & Veloso, M. (2004). Plays as effective multiagent plans enabling opponent-adaptive play selection. In *Proceedings of the fourteenth international conference on automated planning and scheduling* (pp. 376–383). Menlo Park: AAAI Press.

Brooks, R. A. (1991). Intelligence without reason. In J. Myopoulos, R. Reiter (Eds.), *Proceedings of the 12th international joint conference on artificial intelligence (IJCAI-91)* (pp. 569–595). San Mateo: Morgan Kaufmann.

Campbell, J. D., Pillai, P., & Goldstein, S. C. (2005). The robot is the tether: Active, adaptive power routing for modular robots with unary inter-robot connectors. In *IEEE/RSJ international conference on intelligent robots and systems (IROS 2005)* (pp. 4108–4115). Piscataway: IEEE Press.

Cao, YU, Fukunaga, A. S., & Kahng, A. B. (1997). Cooperative mobile robotics: Antecedents and directions. *Autonomous Robots*, *4*, 1–23.

Chaimowicz, L., Campos, M. F. M., & Kumar, V. (2002). Dynamic role assignment for cooperative robots. In *Proc. of the IEEE intl. conf. on robotics and automation (ICRA 2002)* (pp. 293–298). Piscataway: IEEE Press.

Dias, M. B., Zlot, R., Kalra, N., & Stentz, A. (2006). Market-based multirobot coordination: A survey and analysis. *Proceedings of the IEEE*, *94*(7), 1257–1270.

Franks, N. R., & Deneubourg, J. L. (1997). Self-organising nest construction in ants: Individual worker behaviour and the nest's dynamics. *Animal Behaviour*, *54*, 779–796.

Franks, N. R., Pratt, S. C., Mallon, B. Eamonn, Britton, N. F., & Sumpter, D. J. T. (2002). Information flow, opinion polling and collective intelligence in house-hunting social insects. *Philosophical Transactions of the Royal Society of London Series B*, *357*, 1567–1583.

Garnier, S., Jost, C., Jeanson, R., Gautrais, J., Asadpour, M., Caprari, G., & Theraulaz, G. (2005). Collective decision-making by a group of cockroach-like robots. In *Proceedings of the 2nd IEEE swarm intelligence symposium* (pp. 233–240). Piscataway: IEEE Press.

Goldberg, D., & Matarić, M. J. (1997). Interference as a tool for designing and evaluating multi-robot controllers. In *Proceedings of AAAI 1997* (pp. 637–642). Menlo Park: AAAI Press.

Holland, O. E. (1997). Grey Walter: The pioneery of real artificial life. In C. Langton (Ed.), *Proceedings of the 5th international workshop on artificial life* (pp. 34–44). Cambridge: MIT Press.

Huntsberger, T., Rodriguez, G., & Schenker, P. S. (2000). Robotics challenges for robotic and human mars exploration. In *Proceedings of the fourth international conference and exposition on robotics for challenging situations and environment (ROBOTICS 2000)* (pp. 340–346). Reston: ASCE Publications.

Ijspeert, A. J., Martinoli, A., & Billard, A. (2001). Collaboration through the exploitation of local interactions in autonomous collective robotics: the stick pulling experiment. *Autonomous Robots*, *11*(2), 149–171.

Kempe, D., Kleinberg, J., & Demers, A. (2001). Spatial gossip and resource location protocols. In *Proceedings of the thirty-third annual ACM symposium on theory of computing* (pp. 163–172). New York: ACM.

Krause, J., & Ruxton, G. D. (2002). *Living in groups*. New York: Oxford University Press. Chap. Spatial heterogeneity of costs and benefits within groups (pp. 73–86).

Kube, C. R., & Zhang, H. (1996). The use of perceptual cues in multi-robot box pushing. In *Proceedings of the 1996 IEEE international conference on robotics and automation* (pp. 2085–2090). Piscataway: IEEE Press.

Miller, M. B., & Bassler, B. L. (2001). Quorum sensing in bacteria. *Annual Review of Microbiology*, *55*, 165–199.

Nouyan, S., Groß, R., Bonani, M., Mondada, F., & Dorigo, M. (2009). Teamwork in self-organized robot colonies. *IEEE Transactions on Evolutionary Computation*, *13*(4), 695–711.

Olfati-Saber, R., Fax, J. A., & Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE*, *95*(1), 215–233.

Parker, C. A. C. (2009). Collective decision-making in decentralized multiple-robot systems: A biologically inspired approach to making up all of your minds. Ph.D. thesis, University of Alberta, Canada.

Parker, C. A. C., & Zhang, H. (2004). Biologically inspired decision making for collective robotic systems. In *Proceedings of the 2004 IEEE/RSJ international conference on intelligent robots and systems (IROS 2004)* (pp. 375–380). Piscataway: IEEE Press.

Parker, C. A. C., & Zhang, H. (2006). Collective robotic site preparation. *Adaptive Behavior*, *14*(1), 5–19.

Parker, C. A. C., & Zhang, H. (2007). A practical implementation of random peer-to-peer communication for a multiple-robot system. In *Proceedings of the 2007 IEEE/RSJ international conference on intelligent robots and systems (IROS 2007)* (pp. 3730–3735). Piscataway: IEEE Press.

Parker, C. A. C., & Zhang, H. (2008). Consensus-based task sequencing in decentralized multiple-robot systems using local communication. In *Proceedings of the 2008 IEEE/RSJ international conference on intelligent robots and systems (IROS 2008)* (pp. 1421–1426). Piscataway: IEEE Press.

Parker, C. A. C., & Zhang, H. (2009). Cooperative decision-making in decentralized multiple-robot systems: the best-of-n problem. *IEEE/ASME Transactions on Mechatronics*, *14*(2), 240–251.

Peysakhov, M., Dugan, C., Jodi, P. J., & Regli, W. (2006). Quorum sensing on mobile ad-hoc networks. In *Proceedings of the 5th international joint conference on autonomous agents and multiagent systems (AAMAS 2006)* (pp. 1104–1106). New York: ACM.

Pratt, S. C. (2005). Quorum sensing by encounter rates in the ant *Temnothorax albipennis*. *Behavioral Ecology*, *16*(2), 488–496.

Schmickl, T., Thenius, R., Moeslinger, C., Radspieler, G., Kernbach, S., Szymanski, M., & Crailsheim (2009). Get in touch: cooperative decision making based on robot-to-robot collisions. *Autonomous Agents and Multi-Agent Systems*, *18*(1), 133–155.

Seeley, T. D., & Visscher, P. K. (2003). Choosing a home: How the scouts in a honey bee swarm perceive the completion of their group decision making. *Behavioral Ecology and Sociobiology*, *54*, 511–520.

TeamBots (2005). TeamBots home page. http://www.teambots.org.

Vlassis, N., Elhorst, R., & Kok, J. R. (2004). Anytime algorithms for multiagent decision making using coordination graphs. In *Proceedings of international conference on systems, man and cybernetics* (pp. 953–957). Piscataway: IEEE Press.

Waters, C. M., & Bassler, B. L. (2005). Quorum sensing: Cell-to-cell communication in bacteria. *Annual Review of Cell and Developmental Biology*, *21*, 319–346.

Wessnitzer, J., & Melhuish, C. (2003). Collective decision-making and behaviour transitions in distributed ad hoc wireless networks of mobile robots: Target hunting. In *Lecture notes in computer science (LNCS)* (Vol. 2801, pp. 893–902). Berlin: Springer.