An Approach to Switching Control Beyond Nearest Neighbor Rules

Brian Shucker, Todd Murphey, and John K. Bennett
College of Engineering, 430 UCB
University of Colorado
Boulder, Colorado 80309
Email: {shucker, jkb, murphey}@colorado.edu

***Abstract*— Current approaches to distributed control involving many robots generally restrict interactions to pairs of robots within a threshold distance. While this allows for provable stability, there are performance costs associated with the lack of long-distance information. We introduce the *acute angle switching algorithm*, which allows a small number of long-range interactions in addition to interactions with nearby neighbors. We show that the acute angle switching algorithm provides an improvement in performance while retaining the quality of provable stability.**

I. Introduction

With recent advances in integration and wireless communication, there has been increasing interest in the control problem associated with large numbers of cooperating robots. We are particularly interested in the problem of fully distributed control (commonly referred to as *swarming*), in which useful formations are created without any centralized coordination. Limitations on communication bandwidth and range make effective swarming algorithms necessary when the number of robots is large.

Many robotic swarming algorithms are modeled after phenomena observed in nature, such as the flocking behavior of birds or the schooling behavior of fish. Others are based on simulated physical systems. Common to these approaches are simple local control laws implemented on each robot, and designed in such a way that desirable global behaviors emerge. The control laws are typically based on interactions between a given robot, the environment, and any nearby robots that are within a threshold distance.

One key drawback of this approach is that disconnected clusters of robots may never coalesce into a single formation. Disconnected clusters may form as a result of the initial deployment configuration, localized disturbances in the environment, or temporary communication failure, for example.

Our work extends the nearby-neighbors approach so that robots interact with selected neighboring robots at larger distances when possible, in addition to interacting with neighbors within a threshold distance. We have developed a nearest neighbor dynamics model paired with an *acute angle switching algorithm* that uses a small amount of global information to guarantee a planar and connected adjacency graph at all points in time. This allows robots to be deployed in an arbitrary starting configuration and still reach a single connected formation if their sensing is not limited.

We show that the underlying system is stable in terms of velocity; that is, all of the robots are guaranteed to come to rest. Further, we show that the addition of long-range interactions based on the acute angle switching algorithm do not destabilize the system. Thus, in environments where some long-range interactions are possible, we may attain both provable connectivity between all robots and provable stability of the entire system.

II. Related Work

There is a significant body of previous work dealing with coordination of small teams of robots, e.g.[1], [2], [3], [4], [5], [6]. More recently, there has been research into behavior-based and virtual-physics based control of large teams of robots[7], [8], [9], [10], [11], [12]. The work most closely related to our own is summarized below.

A. Behavior-based Control

Fully distributed control based upon simple local behaviors has been used in several contexts. Much of this research is based on the intuition gained from observing behaviors such as flocking in animals. In flocking situations, animals seem to draw most of their behavioral cues from the nearby flockmates. Using this observation as a basis, Brooks[7] has investigated behavior-based control extensively; Werger[8] later described the design principles of such systems. Balch and Hybinette[13] suggested the use of “attachment sites” that mimic the geometry of crystals; this is used to create formations with large numbers of robots. A variety of projects have made use of “swarm robotics,” e.g., [14] and [15], to carry out simple tasks such as light tracking. Gage[9] investigated the use of robot swarms to provide blanket, barrier, or sweep coverage of an area. Several researches have used models based on the interactions of ants within a colony[16], [15], [17]. These approaches generally seek to define simple local behaviors that lead to large-scale properties that are beneficial in a particular application.

Our work seeks to extend the intuition behind behavior-based control to include small amounts of non-local information. We hypothesize that while animals in a flock mostly follow their local neighbors, they may also make use of some larger-scale observations, especially when there are few neighbors in the immediate vicinity. This inspires us to use a switching function that occasionally allows interaction over longer distances.

B. Virtual Physics

Distributed control based on virtual physics (also called "artificial physics" or "physicomimetics") has also been investigated, although not in the manner described here. Howard, Mataric and Sukhatme[10] model robots as like electric charges in order to cause uniform deployment into an unknown enclosed area. Spears and Gordon[11], [12], [18] use a more sophisticated model analogous to the gravitational force, but make the force repulsive at close range. Both of these models use switching functions based on a threshold distance. McLurkin[19] used a partially-connected interaction graph with a physics model similar to that of compressed springs to produce uniform deployment within a limited indoor environment. These works provide useful heuristic algorithms, but unlike our work, they do not attempt to show any provable properties of the resulting formations.

C. Switched Systems

Jadbabaie and colleagues used algebraic graph theory to show stability for switched networks using nearby-neighbor rules[20], [21], [22]. Hespanha and Morse used dwell-time analysis to show stability in linear systems with arbitrary switching that is slow on the average[23], [24], [25]. Bullo and colleagues showed stability in a switched system using Voronoi neighbors[26]. These results all differ from our work in that we use a switching function that is designed first to create specific geometric properties.

III. Control Algorithm

Our algorithm is based on virtual springs created between specific pairs of robots. As with real springs, each virtual spring in the mesh has a natural length and a spring constant.

For a given set of springs, the control law for each robot is

$$\ddot{\mathbf{x}} = \mathbf{u} \tag{1}$$

$$\mathbf{u} = \big[\sum_{i \in S} k_s(l_i - l_0)\hat{\mathbf{v}}_\mathbf{i}\big] - k_d\dot{\mathbf{x}} \tag{2}$$

where $\mathbf{x}$ represents the Cartesian coordinates describing the robot's position, $\ddot{\mathbf{x}}$ is the robot's acceleration, $\dot{\mathbf{x}}$ is the robot's velocity, S is the set of springs connected to this robot, l_i is the length of the i'th spring, and $\hat{\mathbf{v}}_\mathbf{i}$ is the unit vector from this robot to the robot on the other end of the i'th spring. Control constants are the natural spring length (l_0), the spring stiffness (k_s), and the damping coefficient (k_d).

IV. Network Topology Switching

Deciding which adjacent robots with which to form springs is a nontrivial problem. After investigating many options[27], we have developed a new algorithm that is based upon an *acute-angle test*, described in more detail in [27] and [28]. This algorithm retains the connections typical to a nearby-neighbors approach, but adds additional long-range connections in order to maintain connectivity among all robots.

Consider a graph in which the vertices represent robots and the edges represent virtual spring connections. Each vertex has a location equivalent to the estimated location of the robot it represents. Under the Acute-Angle Test algorithm, there is an edge between vertices A and B if and only if for all other vertices C, the interior angle $\angle ACB$ is acute[1]. This creates a mesh of acute triangles. The acute-angle test is equivalent to a test for the presence of any vertex C inside the circle with diameter $\overline{AB}$, which is more efficient to compute.

Figure 1 shows two examples of the acute-angle test. In Figure 1(a), an edge exists between A and B, since all interior angles $\angle ACB$ are acute. In Figure 1(b), the edge does not exist because the acute-angle test fails with robot $C4$. The circle with diameter $\overline{AB}$ is also shown; it is equivalent to say that the edge does not exist because $C4$ is inside the circle.

The acute angle test is symmetric, so it does not require communication between the robots. It results in a planar and connected graph, regardless of the initial distribution of robots. Formal proofs of these properties are given in a companion paper[31]. This provable connectivity is a significant advantage over the standard threshold distance algorithm, since it prevents the formation of separated clusters of robots. The acute angle algorithm is also parameter–free (that is, there is no threshold value that needs to be determined), does not require a global reference frame, and does not put any constraints on the global shape of the mesh.

V. Stability

A. Static Stability

Any spring mesh with fixed topology and $k_d > 0$ will eventually converge to a stationary state, where all robots have velocity approaching zero. Intuitively, this is because the dynamics of a virtual spring are analogous to those of a real spring, in which energy is conserved. Since we ensure $k_d > 0$, there is always a damping effect acting against the motion of each robot. This forces a reduction in kinetic energy. Kinetic energy may be gained by converting potential energy stored in springs, but since springs are dissipative, the total energy (potential + kinetic) in the mesh cannot increase. Since the existence of kinetic energy results in a decrease in total energy, and this energy cannot be replenished, kinetic energy must eventually approach zero.

Formally, for n robots in a spring mesh, define X as the vector of Cartesian positions of the robots. Define the n x n spring matrix as S such that $S(i,j) = 1$ iff a spring exists between robots i and j and $S(i,j) = 0$ otherwise. Define the n x n displacement matrix D such that:

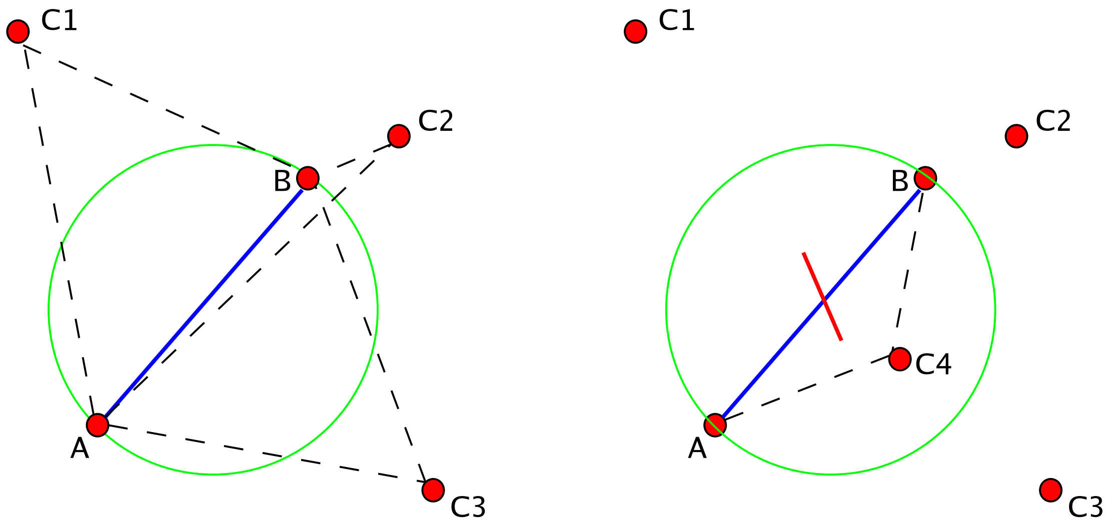
$$D(i,j) = (dist(i,j) - l_0)^2 \tag{3}$$

where $dist(i,j)$ is the distance between robots i and j. D represents the displacement from natural length of every spring that may exist, and is time-dependent.

Lemma 5.1: In a spring mesh with fixed topology and $k_d > 0$, all robots eventually reach zero velocity.

Proof: Our proof will take advantage of Barbalat's lemma, which states that if $f(t)$ is finite and if $\dot{f}(t)$ is

[1] This test is equivalent to that used to generate a Gabriel graph[29], [30].



(a) Satisfied acute-angle test

(b) Unsatisfied acute-angle test

Fig. 1. Illustration of acute-angle test

uniformly continuous (or equivalently, $\ddot{f}(t)$ is finite), then $\dot{f}(t)$ approaches zero as t approaches infinity. We will apply Barbalat's lemma to an energy function $\mathbf{V}$, thereby showing that $\dot{\mathbf{V}}$ goes to zero, which together with our definition will imply that all robots reach zero velocity.

Consider the following energy function:

$$\mathbf{V} = 1^{nT}\frac{1}{2}k_s(S.*D)1^n + \frac{1}{2}(\dot{X}^T\dot{X}) \tag{4}$$

where $k_s > 0$ is the spring stiffness, $.*$ represents an element-by-element multiply and 1^n is the vector of n ones.

As D and S are positive-definite and $k_s > 0$, $\mathbf{V}$ is positive-definite.

While we omit the derivation (which is nontrivial but fairly straightforward) for brevity, we claim that the derivative of the energy function is the following:

$$\dot{\mathbf{V}} = -k_d(\dot{X}^T\dot{X}) \tag{5}$$

which is obtained by differentiating $\mathbf{V}$ and using Equation 1 to substitute for $\ddot{X}$. As intended by our choice of control laws, all of the spring potential terms cancel out and leave only the damping terms.

Differentiating $\dot{\mathbf{V}}$, we see:

$$\ddot{\mathbf{V}} = -2k_d(\dot{X}^T\ddot{X}) \tag{6}$$

We defined $\ddot{\mathbf{X}}$ in Equation 1, and it is clearly finite as long as the distances between the robots are finite. We also know that $\dot{\mathbf{X}}$ is finite, because it is a term of $\mathbf{V}$ ($\mathbf{V}$ contains only positive terms and is bounded above by its initial condition, as its derivative is negative semi-definite). Thus, all terms in $\ddot{\mathbf{V}}$ are finite.

Since $\mathbf{V}$ is lower bounded by zero, $\dot{\mathbf{V}}$ is negative semi-definite, and $\ddot{\mathbf{V}}$ is finite (equivalently, $\dot{\mathbf{V}}$ is unifromly continuous), Barbalat's lemma states that $\dot{\mathbf{V}} \to 0$ as $t \to \infty$. This can only occur when all terms in $\dot{\mathbf{X}}$ are zero, so all robots eventually reach zero velocity.

■

Notice that while $\dot{\mathbf{V}}$ must approach zero, $\mathbf{V}$ may not. It is possible for some potential energy to exist even in a fixed spring mesh in its stationary state. In the general case, it is of course possible to add energy by changing the mesh topology.

Also note that the state with zero potential energy does not necessarily exist in a given environment. This is most easily seen in the case where many robots are placed in a small room (perhaps smaller across than the natural spring length). In this environment, there is no reachable configuration with zero potential energy.

These properties imply that in the absence of switching, all the robots come to rest, but that the exact final formation is not uniquely determined.

B. *Dynamic Stability*

In order to show stability in the presence of time-varying topology (the dynamic case), we modify the switching algorithm in a manner inspired by dwell-time analysis. Hespanha and Morse[24] proved that if all members of a given class of linear systems are stable, then arbitrary switching among those systems results in a stable hybrid system, provided that the switching rate is "slow-on-the-average". They further show how to compute the average time between switches (the *dwell time*) that guarantees stability. Essentially, the proof shows that the rate of decrease of the Lyapunov function due to the dissipation is greater than the rate of increase of the Lyapunov function due to switching.

In our approach, instead of computing a limit on the switching frequency explicitly, we use a notion of a global "energy reserve" to create the same limiting effect on the switching rate. We find this approach intuitive and more straightforward to implement in our distributed system, in which switching events are detected locally.

Let the global energy reserve E be denoted as E and define constant k_e such that $0 < k_e < 1$. The quantity E is defined to be the solution to a differential equation. It starts with some nonnegative initial value E_0 and evolves according to the following:

$$\begin{aligned} \frac{\Delta E}{\Delta t} = \quad & k_e k_d((\tfrac{\mathbf{\Delta X}}{\Delta t})^T(\tfrac{\mathbf{\Delta X}}{\Delta t})) & (7) \\ & -1^{nT}\tfrac{1}{2}k_s(\Delta S.*D_t)1^n & (8) \end{aligned}$$

where the $.*$ operator indicates an element-by-element multiply and 1^n indicates the column vector of n ones. Note that S is no longer constant; it changes whenever there is a change in topology.

The first term in the above equation is positive-definite as before. The second term only comes into play when a topology change occurs (that is, when there is a change in S), and it is exactly the opposite of the instantaneous change in $\mathbf{V}$ due to the topology change.

To ensure stability in the dynamic case, our algorithm must forbid topology changes in any case where the result would cause $E_t < 0$. This is easy to enforce, since the effect on E of forming each spring is precisely defined. This restriction, combined with the original switching algorithm, defines the *modified acute-angle switching algorithm*.

Algorithm 1 $\mathbf{u}$ = Update($\mathbf{X}$, $\mathbf{\Delta x}$, E, priorSprings)

```
currentSprings ← AcuteAngleTest(X)
deltaE ← Potential(currentSprings - priorSprings, X)
if (E - deltaE) < 0 then currentSprings ← priorSprings
else E ← E - deltaE
u ← ControlLaw(currentSprings, X, Δx)
E ← E + k_e k_d(Δx/Δt)^T(Δx/Δt)
E ← AvgWithNeighbors(E)
priorSprings ← currentSprings
```

Algorithm 1 shows pseudocode including the modified switching algorithm. The update function executes once per time step on each robot. The spring sets indicated only include springs connected to the robot on which the code is executing. $\mathbf{X}$ represents the vector of positions of all visible robots, and $\mathbf{\Delta x}$ represents the executing robot's change in position since the last time step.

At each time step, each robot recomputes its spring connections using the acute angle test. The change in potential caused by the new spring connections is computed, and that quantity is subtracted from E if it would not bring E below zero (this implements the second term of Equation 8). Otherwise, the old set of spring connections are retained. The control law is then applied to update u, and the first term of Equation 8 is applied to complete the update of E. Changes in E are propagated through the mesh through averaging with neighbors, which is described more below.

Theorem 5.2: In a spring mesh with topology determined by the modified acute-angle switching algorithm, all robots eventually reach zero velocity.

Proof: Our proof will invoke Barbalat's lemma in the same manner as before, but with a modified potential function:

$$\mathbf{V}' = \mathbf{V} + E \tag{9}$$

Recall that E was defined in such a way that it is non-decreasing in the absence of switching. Also, our modified switching algorithm forbids any switch that would cause E to become negative. This restriction is vital in that it ensures that $\mathbf{V}'$ never becomes negative.

Converting $\dot{\mathbf{V}}$ to discrete-time form, we have:

$$\begin{aligned} \frac{\Delta \mathbf{V}}{\Delta t} = \quad & -k_d(\tfrac{\mathbf{\Delta X}}{\Delta t})^T(\tfrac{\mathbf{\Delta X}}{\Delta t}) \\ & +1^{nT}\tfrac{1}{2}k_s(\Delta S.*D_t)1^n \end{aligned}$$

The first term is derived from Equation 5; the second term reflects the possible instantaneous change in $\mathbf{V}$ due to a topology change and is derived from Equation 4.

Substituting from previous equations, we have:

$$\frac{\Delta \mathbf{V}'}{\Delta t} = \frac{\Delta \mathbf{V}}{\Delta t} + \frac{\Delta E}{\Delta t} \tag{10}$$

$$\begin{aligned} \frac{\Delta \mathbf{V}'}{\Delta t} = \quad & \big[-k_d(\tfrac{\mathbf{\Delta X}}{\Delta t})^T(\tfrac{\mathbf{\Delta X}}{\Delta t}) + 1^{nT}\tfrac{1}{2}k_s(\Delta S.*D_t)1^n\big] \\ & +\big[k_e k_d(\tfrac{\mathbf{\Delta X}}{\Delta t})^T(\tfrac{\mathbf{\Delta X}}{\Delta t}) - 1^{nT}\tfrac{1}{2}k_s(\Delta S.*D_t)1^n\big] \end{aligned}$$

which simplifies to:

$$\frac{\Delta \mathbf{V}'}{\Delta t} = -k_d(1-k_e)((\frac{\mathbf{\Delta X}}{\Delta t})^T(\frac{\mathbf{\Delta X}}{\Delta t})) \tag{11}$$

It is convenient at this point to convert back to continuous-time notation:

$$\dot{\mathbf{V}}' = -k_d(1-k_e)(\dot{X}^T\dot{X}) \tag{12}$$

Now it is easy to see that, similarly to the static case,

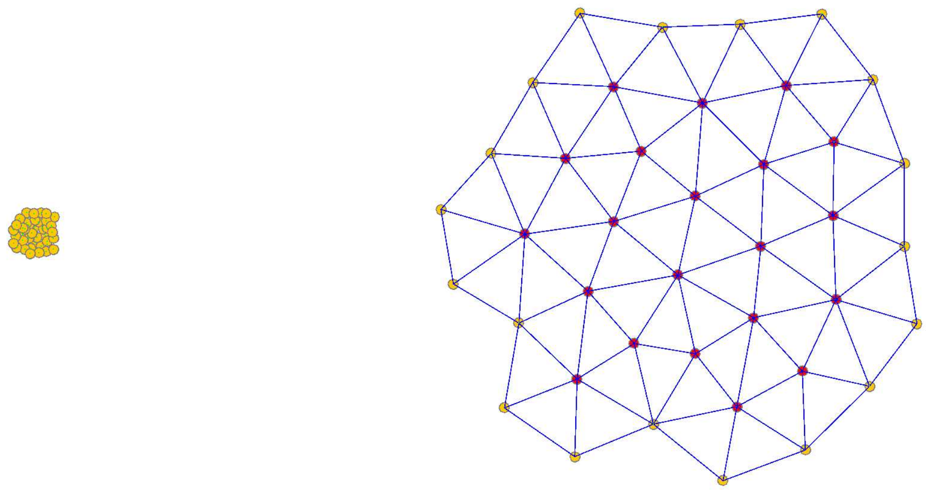
$$\ddot{\mathbf{V}}' = -2k_d(1-k_e)(\dot{X}^T\ddot{X}) \tag{13}$$

where $\ddot{\mathbf{X}}$ is now discontinuous, but is still finite.

Thus, we again have a nonnegative potential function $\mathbf{V}'$ with $\dot{\mathbf{V}}'$ negative semi-definite and $\ddot{\mathbf{V}}'$ finite. Barbalat's lemma still applies, implying $\dot{\mathbf{V}}' \to 0$ as $t \to \infty$, so all velocities must approach zero.

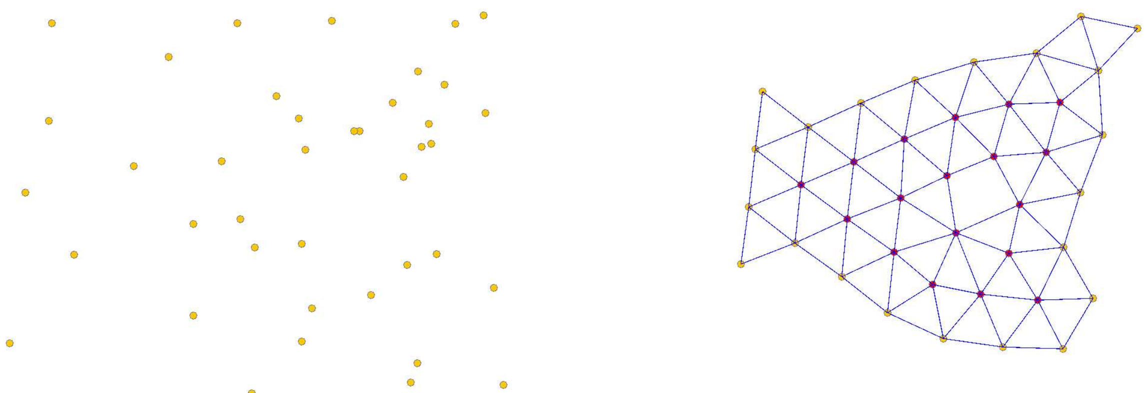
■

By design, switching does not affect the value of $\mathbf{V}'$ because the changes in $\mathbf{V}$ caused by switching are countered exactly by opposite changes in E. Intuitively, the change in E at each time step indicates the amount of energy that is damped out of the system (reduced by the constant factor k_e), minus the energy created or destroyed by spring switching. By forbidding spring switching when $E < 0$ and using $0 < k_e < 1$, we ensure that, on the average, the energy introduced by switching is less than the energy removed from the system by damping. This is intuitively similar to the result obtained through dwell-time analysis.



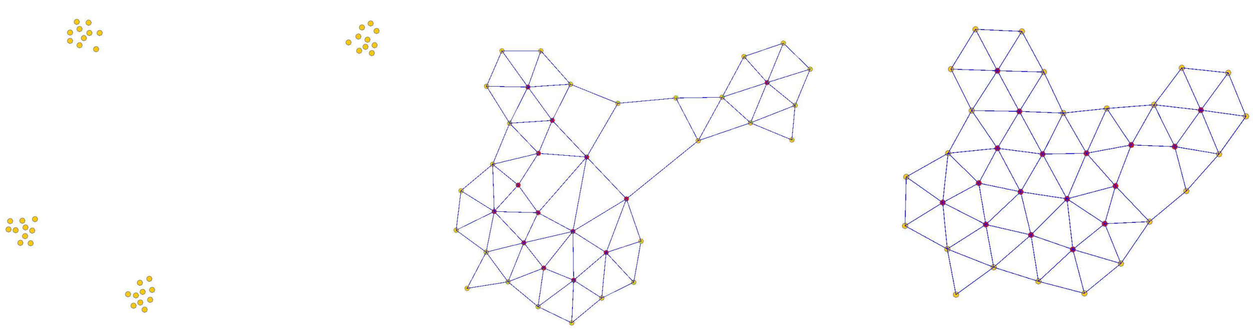
(a) Initial state (b) Final state

Fig. 2. Cluster deployment scenario: robots deploy from a clustered initial configuration



(a) Initial state (b) Final state

Fig. 3. Distributed deployment scenario: robots deploy from a scattered initial configuration



(a) Initial state (b) Intermediate state (c) Final state

Fig. 4. Multiple cluster deployment scenario: robots from initially separated clusters deploy into a single mesh

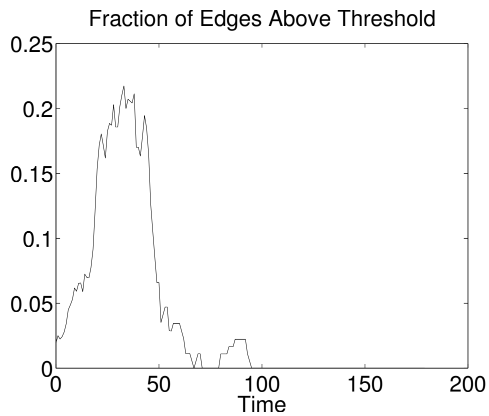


Fig. 5. Fraction of edges that are not within threshold distance, for the situation shown in Figure 4

Because the effect of each switch on the global potential can be locally determined, no global communication is necessary. Although E is defined as a global value, each robot may have its own estimate of the current value of E. Changes in E may be propagated by causing each robot to periodically average its estimate of E with its neighbors' estimates. Averaging is a conservative operation (it does not change the sum of all estimates of E), and in a connected mesh all estimates will eventually converge to the same value. Such convergence is provable and is known as the *consensus problem*[32]. Thus, every estimate will be driven to the same value, which has the same behavior (neglecting scaling by the number of robots) as the global E, since no energy was created or destroyed by the averaging process.

VI. Simulation Results

Figures 2, 3, and 4 show the results of three simulated deployment scenarios. In Figure 2, forty robots deploy from a tight cluster. In Figure 3, forty robots deploy from a scattered configuration. In Figure 4, four distinct clusters of ten robots each deploy simultaneously. In all cases, the potential in the final configuration is near zero, but not equal to zero. This is a typical result.

The result illustrated in Figure 4 is significantly different from what would be obtained with a nearby neighbors algorithm. The addition of the acute angle switching function guarantees that the clusters all connect into a single mesh; without this, there would be no assurance that the clusters would merge.

Figure 5 shows the fraction of the edges that are longer than the nearby-neighbors threshold distance as a function of simulation time. This quantity represents the degree to which the acute angle switching algorithm is behaving differently than the threshold distance algorithm. There are a significant number of long edges during the period when the clusters are combining, but the number of long edges decreases rapidly to zero when the combination is complete. This is not surprising—the dynamics of the system tend to drive all edge lengths to the natural length. Thus, the acute angle mesh becomes equivalent to the threshold distance mesh over time.

VII. Conclusion

We have demonstrated an alternative to nearby-neighbor switching algorithms. The acute angle switching algorithm creates a switched system that features a provably connected adjacency graph. While providing connectivity, the added long-distance links created by the acute angle switching algorithm do not destabilize the system. This allows an increase in performance when compared to standard switching algorithms based on simple distance thresholds.

References

[1] T. Balch and L. E. Parker, *Robot Teams: From Diversity to Polymorphism.* A K Peters Ltd, 2002.

[2] T. Balch and R. C. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, pp. 926–939, 1998.

[3] T. Balch and R. C. Arkin, "Motor schema-based formation control for multiagent robot teams," in *First International Conference on Multi-Agen Systems (ICMAS)*, 1995.

[4] M. Roth, D. Vail, and M. Veloso, "A world model for multi-robot teams with communication," in *IROS-2003*, 2003. (under submission).

[5] W. Burgard, D. Fox, M. Moors, R. Simmons, and S. Thrun, "Collaborative multi-robot exploration," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2000.

[6] B. P. Gerkey, R. T. Vaughan, K. Stoy, A. Howard, G. S. Sukhatme, and M. J. Mataric, "Most valuable player: A robot device server for distributed control," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2001.

[7] R. A. Brooks, "Integrated systems based on behaviors," *SIGART Bull.*, vol. 2, no. 4, pp. 46–50, 1991.

[8] B. B. Werger, "Cooperation without deliberation: A minimal behavior-based approach to multi-robot teams," *Artificial Intelligence*, vol. 110, pp. 293–320, 1999.

[9] D. W. Gage, "Command control for many–robot systems," in *Proceedings of Nineteenth Annual AUVS Technical Symposium*, June 1992.

[10] A. Howard, M. J. Mataric, and G. S. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *6th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, June 2002.

[11] W. M. Spears and D. F. Gordon, "Using artificial physics to control agents," in *Proceedings of IEEE International Conference on Information, Intelligence, and Systems*, 1999.

[12] D. F. Gordon, W. M. Spears, O. Sokolsky, and I. Lee, "Distributed spatial control, global monitoring and steering of mobile agents," in *Proceedings of IEEE International Conference on Information, Intelligence, and Systems*, 1999.

[13] T. Balch and M. Hybinette, "Behavior-based coordination of large-scale robot formations," in *Proceedings of the Fourth International Conference on Multiagent Systems (ICMAS*, pp. 363–364, July 2000.

[14] G. Baldassarre, S. Nolfi, and D. Parisi, "Evolving mobile robots able to display collective behaviors," in *Proceedings of the International Workshop on Self-Organization and Evolution of Social Behaviors*, pp. 11–22, September 2002.

[15] E. Şahin and N. Franks, "Measurement of space: From ants to robots," in *Proceedings of WGW 2002: EPSRC/BBSRC International Workshop Biologically-Inspired Robotics: The Legacy of W. Grey Walter*, (Bristol, UK), pp. 241–247, Aug. 14-16, 2002.

[16] S. Koenig and Y. Liu, "Terrain coverage with ant robots: A simulation study," in *Proceedings of the International Conference on Autonomous Agents*, pp. 600–607, 2001.

[17] B. B. Werger and M. J. Mataric, "From insect to internet: Situated control for networked robot teams," in *Annals of Mathematics and Aritificial Intelligence*, pp. 173–197, 2001.

[18] D. F. Gordon-Spears and W. M. Spears, "Analysis of a phase transition in a physics-based multiagent system," in *Proceedings of NASA–Goddard/IEEE Workshop on Formal Approaches to Agent-Based Systems*, 2002.

[19] J. McLurkin and J. Smith, "Distributed algorithms for dispersion in indoor environments using a swarm of autonomous mobile robots," in *7th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, June 2004.

[20] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, “Flocking in fixed and switching networks,” *IEEE Transactions on Automatic Control (submitted)*, 2005.

[21] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, “Stable flocking of mobile agents, part i: Fixed topology,” in *IEEE Conf. on Decision and Control*, 2003.

[22] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, “Stable flocking of mobile agents, part ii: Dynamic topology,” in *IEEE Conf. on Decision and Control*, 2003.

[23] J. P. Hespanha, “Extending lasalle’s invariance principle to switched linear systems,” in *IEEE Conf. on Decision and Control*, 2001.

[24] J. P. Hespanha and A. S. Morse, “Stability of switched systems with average dwell-time,” in *IEEE Conf. on Decision and Control*, 1999.

[25] J. P. Hespanha, D. Liberzon, A. S. Morse, B. D. O. Anderson, T. S. Brinsmead, and F. D. Bruyne, “Multiple model adaptive control, part 2: Switching,” *International Journal of Robust and Nonlinear Control*, 2001.

[26] J. Cortes, S. Martinez, T. Karatas, and F. Bullo, “Coverage control for mobile sensing networks,” *IEEE Transactions on Robotics and Automation*, 2004.

[27] B. Shucker and J. K. Bennett, “Virtual spring mesh algorithms for control of distributed robotic macrosensors,” Tech. Rep. CU-CS-996-05, University of Colorado, May 2005.

[28] B. Shucker and J. K. Bennett, “Scalable control of distributed robotic macrosensors,” in *7th International Symposium on Distributed Autonomous Robotic Systems (DARS)*, June 2004.

[29] K. R. Gabriel and R. R. Sokal, “A new statistical approach to geographic variation analysis,” *Systematic Zoology*, vol. 18, pp. 259–278, 1969.

[30] D. W. Matula and R. R. Sokal, “Properties of Gabriel graphs relevant to geographic variation research and the clustering of points in the plane,” *Geographical Analysis*, vol. 12, pp. 205–222, 1980.

[31] B. Shucker, T. Murphey, and J. K. Bennett, “Cooperative control using occasional non-local interactions,” in *IEEE Conference on Robotics and Automation*, 2006.

[32] R. O. Saber and R. M. Murray, “Consensus protocols for networks of dynamic agents,” in *American Control Conference*, 2003.