A Formal Analysis of Potential Energy in a Multiagent System

William M. Spears
Diana F. Spears
Rodney Heil
Computer Science Department
University of Wyoming, Laramie, WY 82071
Email: wspears@cs.uwyo.edu

I. Introduction

The focus of our research is to build sensor network systems, specifically, to design rapidly deployable, scalable, adaptive, cost-effective, and robust networks (i.e., swarms, or large arrays) of autonomous distributed mobile sensing agents (e.g., robots). This combines sensing, computation and networking with mobility, thereby enabling deployment, assembly, reconfiguration, and disassembly of the multi-agent collective. Our objective is to provide a scientific, yet practical, approach to the design and analysis (behavioral assurance) of aggregate sensor systems.

Agent vehicles could vary widely in type, as well as size, e.g., from nanobots to micro-air vehicles (MAVs) and micro-satellites. Agents are assumed to have sensors and effectors. An agent's sensors perceive the world, including other agents, and an agent's effectors make changes to that agent and/or the world, including other agents. It is assumed that agents can only sense and affect nearby agents; thus, control rules must be "local." Desired global behavior emerges from local agent interactions.

This paper summarizes our *physicomimetics* framework for robot control. A theoretical analysis of potential energy is then provided, allowing us to properly set system parameters a priori. Finally, results of a multi-robot implementation are presented.

II. Relation to Alternative Approaches

System analysis enables both system design and behavioral assurance. Here, we adopt a physics-based approach to analysis. We consider this approach to fit under the category of "formal methods," not in the traditional sense of the term but rather in the broader sense, i.e., a formal method is a mathematical technique for designing and/or analyzing a system. The two main *traditional* formal methods used for this purpose are theorem proving and model checking. Why do we use a physics-based method instead of these more traditional methods? The gist of theorem proving (model checking) is to begin with a theorem (property) and prove (show) that it holds for the target system. But what if you don't know how to express the theorem or property in the first place? For example, suppose you visually observe a system behavior that you want to control, but you have no idea what causes it or how to express your property in concrete, logic-based or system-based terms? In particular, there may be a property/law relating various system parameters that enables you to predict or control the observed phenomenon, but you do not understand the system well enough to write down this law.

For such a situation, the traditional, logic-based formal methods are not directly applicable. One potentially applicable approach is empirical, e.g., machine discovery. We have chosen a theoretical (formal) physics-based approach because:

- Empirical techniques can tell you *what* happens, but not *why* it happens. Causal explanations are easier to understand, apply, build upon, and generalize.
- If a physics-based analysis technique is predictive of a system built on physics-based principles, then this analysis provides formal verification of the correctness of the system implementation. No such claims can be made

for empirical results.

- *Finally, and most importantly, it is possible to go directly from theory to a successful robot demo, without the usual extensive parameter tweaking! We have already demonstrated such successes with our theories [1].*

III. The Physicomimetics Framework

In our physicomimetics framework, virtual physics forces drive a multi-agent system to a desired configuration or state. The desired configuration (state) is the one that minimizes overall system potential energy. We also refer to our framework as "artificial physics" or "AP".

At an abstract level, physicomimetics treats agents as physical particles. This enables the framework to be embodied in vehicles ranging in size all the way from nanobots to satellites. Particles exist in two or three dimensions and are considered to be point-masses. Each particle i has position $p = (x, y, z)$ and velocity $v = (v_x, v_y, v_z)$. We use a discrete-time approximation to the continuous behavior of the particles, with time-step Δt. At each time step, the position of each particle undergoes a perturbation Δp. The perturbation depends on the current velocity, i.e., $\Delta p = v\Delta t$. The velocity of each particle at each time step also changes by Δv. The change in velocity is controlled by the force on the particle, i.e., $\Delta v = F\Delta t/m$, where m is the mass of that particle and F is the force on that particle. A frictional force is included, for self-stabilization. This force is modeled as a *viscous friction* term, i.e., the product of a viscosity coefficient and the agent's velocity (independently modeled in the same fashion by [2]).

The time step Δt is proportional to the amount of time the robots take to perform their sensor readings. A parameter F_{max} is added, which restricts the amount of acceleration a robot can achieve. A parameter V_{max} restricts the velocity of the particles. Collisions are not modeled, because AP repulsive forces tend to avoid collisions. Also, we do not model the low-level dynamics of the actual robot. We consider AP to be an algorithm that will determine "way points" for the actual physical platforms. Lower-level software can steer between way points.

Given a set of initial conditions and some desired global behavior, we define what sensors, effectors, and force F laws are required such that the desired behavior emerges.

IV. Designing Lattice Formations

The example considered in this section was inspired by an application which required a swarm of MAVs to form a hexagonal lattice, thus creating an effective antenna [3].

Since MAVs (or other small agents such as nanobots) have simple sensors and primitive CPUs, our goal was to provide the simplest possible control rules requiring minimal sensors and effectors. Creating hexagons appears to be rather complicated, requiring sensors that can calculate range, the number of neighbors, their angles, etc. However, it turns out that only range and bearing information are required. To see this, recall that six circles of radius R can be drawn on the perimeter of a central circle of radius R. Figure 1 illustrates this construction. If the particles (shown as small circular spots) are deposited at the intersections of the circles, they form a hexagon with a particle in the middle.

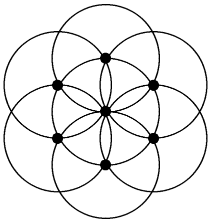

Fig. 1. How circles can create hexagons.

We see that hexagons can be created via overlapping circles of radius R. To map this into a force law, each particle repels other particles that are closer than R, while attracting particles that are further than R in distance. Thus each particle can be considered to have a circular "potential well" around itself at radius R – neighboring particles will want to be at distance R from each other. The intersection of these wells is a form of constructive interference that creates "nodes" of very low potential energy where the particles will be likely to reside. The particles serve to create the very potential energy surface to which they are responding![1]

[1] The potential energy surface is never actually computed by the robots. It is only computed in the simulation for visualization/analysis.

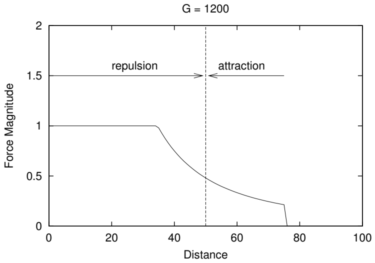

Fig. 2. The force law, when R = 50, G = 1200, p = 2 and F_{max} = 1.

With this in mind we defined a force law $F = Gm_im_j/r^p$, where $F \leq F_{max}$ is the magnitude of the force between two particles i and j, and r is the range between the two particles. The variable p represents a user-defined power, which can range from -5.0 to 5.0. When $p = 0.0$ the force law is constant for all ranges. Unless stated otherwise, we assume $p = 2.0$ and $F_{max} = 1$ in this paper. The "gravitational constant" G is set at initialization. The force is repulsive if $r < R$ and attractive if $r > R$. Each particle has one sensor that can detect the range and bearing to nearby particles. The only effector is to be able to move with velocity $v \leq V_{max}$. To ensure that the force laws are local, particles have a visual range of $1.5R$.

Figure 2 shows the magnitude of the force, when $R = 50$, $G = 1200$, $p = 2$, and $F_{max} = 1$ (the system defaults). There are three discontinuities in the force law. The first occurs where the force law transitions from F_{max} to $F = Gm_im_j/r^p$. The second occurs when the force law switches from repulsive to attractive at R. The third occurs when the force goes to 0.

The initial conditions are a tight cluster of robots, that propel outward (due to repulsive forces) until the desired geometric configuration is obtained. This is simulated by using a two dimensional Gaussian random variable to initialize the positions of all particles. Velocities of all particles are initialized to be 0.0, and masses are all 1.0 (although the framework does not require this).

Using this force law, AP successfully forms hexagonal lattices, with a small number of agents or hundreds. Square lattices are also easily obtained [1], [4]. For a radius R of 50, a gravitational constant of approximately $G = 1200$ provides good results. The issue of how to set G, given other system parameters, is the focus of the analysis in this paper.

V. Energy Analysis

Because our force law is conservative (in the physics sense), the AP system should obey conservation of energy – if it is implemented correctly. Furthermore, as we shall see, the initial potential energy of the system in the starting configuration yields important information concerning the dynamics of the system.

First, we measured the potential energy (PE) of the system at every time step, using the path integral $V = -\int_s \vec{F} \bullet d\vec{s}$.[2] This can be thought of as the amount of work required to push each particle into position, one after another, for the current configuration of particles. Because the force is conservative, the order in which the particles are chosen is not relevant. Then we also measured the kinetic energy (KE) of the particles ($mv^2/2$). Finally, since there is friction we also must take into account that energy as well, which we can consider to be heat energy. If there is no friction, the heat energy is zero.

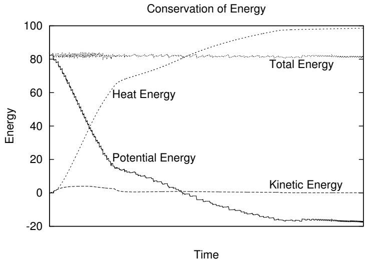

Fig. 3. Conservation of energy, showing how the total energy remains constant, although the amount of different forms of energy change over time.

Figure 3 illustrates an example of the energy dynamics of the AP system. As expected, the total energy remains constant over time. The system starts with only PE. Note that the graph illustrates one of the foundational principles of the AP system, namely, that the system continually evolves to lower PE, until a minimum is reached. This reflects a form of stability of the final aggregate system, requiring work to move the system away from desired configurations (thus increasing PE).

[2] V is the traditional notation for potential energy.

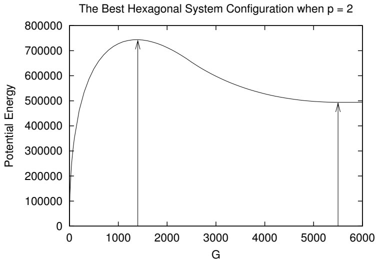

Fig. 4. The amount of potential energy of the initial configuration of the system is maximized for the G value that empirically yields the best results, which is roughly 1300. In this example $p = 2$. The arrows show the values of G_{opt} and G_{max}, respectively.

As the system evolves, the PE is converted into KE and heat, and the particles exhibit maximum motion, which is not very large (see Figure 3). Finally, however, the particles slow, and only heat remains. Note also that PE is negative after a certain point. This illustrates stability of individual particles (as well as the collective) – it would require work to push individual particles out of these configurations. Hence this graph shows how the system would be resilient to moderate amounts of force acting to disrupt it, once stable configurations are achieved.

We have found that the initial configuration PE indicates important properties of the final evolved system, namely how well it evolves and the size of the formation. Intuitively, higher initial PE indicates that more work can be done by the system – and the creation of bigger formations requires more work. We have also observed that higher initial PE is correlated with better formations. Apparently there is more energy via momentum to push through local optima to global optima.

For example, consider Figure 4, which shows the PE of the initial configuration of a 200 particle system (when $p = 2$), for different values of G. In the figures, G_{opt} is the value of G at which the PE is maximized, and G_{max} is the largest useful setting of G (i.e., above G_{max} all forces are equal to F_{max}). Interestingly, PE is maximized almost exactly at the range of values of G (around 1200 to 1400) that we have found empirically to yield the best structures.

We now compute a general expression for when PE is maximized. To find this expression for G_{opt}, we first need to calculate the potential energy, V. For simplicity, we begin by calculating the potential energy of a two particle system where the two particles are very close to each other.

It will be necessary to consider three different situations, depending on the radial extent to which F_{max} dominates the force law $F = G/r^p$. Recall that agents use F_{max} when $F \geq F_{max}$. This occurs when $G/r^p \geq F_{max}$ or, equivalently, when $r \leq (G/F_{max})^{1/p} \equiv R'$. The first situation occurs when F_{max} is used only when the other particle is at close range, i.e., when $0 \leq R' \leq R$. The second situation occurs when $R \leq R' \leq 1.5R$. The third situation occurs when F_{max} is always used, i.e., when $R' > 1.5R$. In this situation the force law is constant (F_{max}) and V remains constant with increasing G.

Let us now compute the PE for the first situation. It will be necessary to calculate three separate integrals for this situation. The first will represent the attractive force felt by one particle as it approaches the other, from a range of $1.5R$ to R. The second is the repulsive force of $F = G/r^p$ when $r < R$ and $F < F_{max}$. The third represents the range where the repulsive force is simply F_{max}.[3] Then:

$$V = -\int_R^{1.5R} \frac{G}{r^p}\,dr + \int_{R'}^{R} \frac{G}{r^p}\,dr + \int_0^{R'} F_{max}\,dr$$

Note that the first term is negative because it deals with attraction, whereas the latter two terms are positive due to repulsion. Solving and substituting for R' yields:

$$V = \frac{(2R^{1-p} - (1.5R)^{1-p})G}{(1-p)} - \frac{pG^{1/p}}{(1-p){F_{max}}^{(1-p)/p}}$$

The derivation of the second and third situations is similar. The first situation occurs with low G, when $G \leq F_{max}R^p$. The second situation occurs with higher levels of G, when $F_{max}R^p \leq G \leq F_{max}(1.5R)^p$. The third situation occurs when $G \geq F_{max}(1.5R)^p$. In the third situation the PE of the system remains constant as G increases even further. Thus the maximum useful setting of G is $G_{max} = F_{max}(1.5R)^p$. We can see this in Figure 4 (which represent the full curves over all three situations), where $G_{max} = 5625$.

We next generalize to N particles for V (denoted V_N). Note that we can build our N particle system

[3] Throughout our theoretical results we assume that $p \neq 1.0$, which is reasonable since we typically do not run AP with that setting.

one particle at a time, in any order (because forces are conservative), resulting in an expression for the total initial PE.

$$V_N = \sum_{i=0}^{N-1} iV = \frac{VN(N-1)}{2}$$

with V defined above for the 2-particle system.

Now that we have a general expression for the potential energy, V_N, to find the expression for G_{opt} we need to find the value of G that maximizes V_N. First, we need to determine whether the maximum occurs in the first or second situation. It is easy to show that the slope of the PE equation for the second situation is strictly negative; thus the maximum must occur in the first situation. To find the maximum, we take the derivative of the V_N for the first situation with respect to G, set it to zero, and solve for G. The resulting maximum is at:

$$G_{opt} = F_{max} R^p [2 - 1.5^{1-p}]^{p/(1-p)}$$

Note that the value of G_{opt} does not depend on the number of particles, which is a nice result. This simple formula is surprisingly predictive of the dynamics of a 200 particle system. For example, when $F_{max} = 1$, $R = 50$, and $p = 2$, $G_{opt} = 1406$, which is only about 7% higher than the value shown in Figure 4. Similarly, when $p = 3$, $G_{opt} = 64,429$, which is very close to observed values. The difference in values stems from the fact that in our simulation we have initial conditions specified by a two-dimensional Gaussian random variable with a small variance σ^2, whereas our mathematical analysis assumes a variance of zero. Despite this difference, the equation for G_{opt} works quite well.

As described in [4], we have also had success in creating square lattices. Performing a similar potential energy analysis yields a G_{opt} of:

$$F_{max} R^p \left[\frac{\sqrt{2}(N-1)[2-1.3^{1-p}] + N[2-1.7^{1-p}]}{\sqrt{2}(N-1)+N} \right]^{p/(1-p)} \quad (1)$$

Note that G_{opt} actually depends on the number of particles N, which is the first time we have seen such a dependency. It occurs because we use two "species" of particles to create square lattices, which have different sensor ranges. However, because this difference is not large, the dependency on N is also not large. For example, with $R = 50$ and $F_{max} = 1$, then when $p = 2$ and there are 200 total particles, $G_{opt} = 1466$. With only 20 particles $G_{opt} = 1456$. Similarly, when $p = 3$ we obtain values of $G_{opt} = 67,330$ and $G_{opt} = 66,960$ respectively (for 200 and 20 particle systems).

VI. Experiments With a Team of Robots

For our experiments, we used seven robots from the KIPR Company. For detecting neighbor robots, Sharp GP2D12 IR sensors are mounted, providing a 360 degree field of view, from which object detection is performed. The output is a list that gives the bearing and range to all neighboring robots. Once sensing and object detection are complete, the AP algorithm computes the virtual force felt by that robot. In response, the robot will turn and move to some position. This cycle of sensing, computation and motion continues until we shut down the robots or they lose power. The AP code is simple to implement. It takes a robot neighbor list as input, and outputs a turn and distance to move.

The goal of the first experiment was to form a hexagon with seven robots. Each robot ran the same software. The desired distance R between robots was 23 inches. Using the theory we chose a G of 270 ($p = 2$ and $F_{max} = 1$). The beginning configuration was random. The final configuration was a hexagon. The results are consistent, achieving the same formation ten times in a row with the same starting conditions and taking approximately seven cycles on average. For all runs the robots were separated by 20.5 to 26 inches in the final formation, which is only slightly more error than the sensor error.

For our second experiment we placed four photodiode light sensors on each robot, one per side. These produced an additional force vector, moving the robots towards a light source (a window).[4] The results are shown in Figure 5, and were consistent over ten runs, achieving an accuracy comparable to the formation experiment above. The robots moved about one foot in 13 cycles of the AP algorithm.

[4] The reflection of the window on the floor is not noticed by the robots and is not the light source.

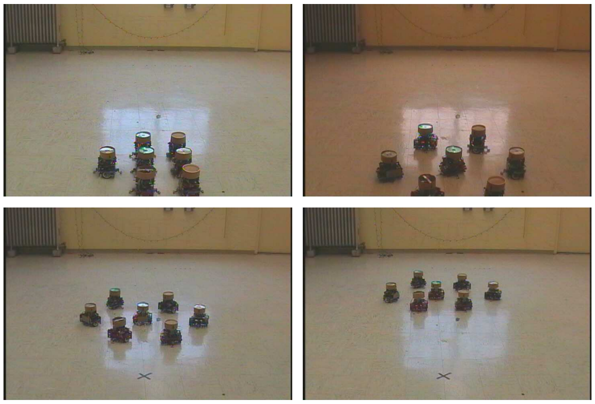

Fig. 5. Seven robots get into formation, and move toward the light.

VII. Summary, Related and Future Work

This paper presents a novel analysis of AP, focusing on potential energy. This analysis provides us with a predictive technique for setting important parameters in the system, thus enabling a system user to create (with good assurance) large formations. This static analysis combines many important parameters of the system, such as G, R, p, F_{max}, and sensor range. It also includes the geometry of the formations in a natural fashion. The parameter N was included as well, but it turns out to be of little relevance for our most important results. This is a nice feature, since our original motivation for the AP approach was that we wished it to scale easily to large numbers of agents. To include the other relevant dynamic parameters such as Δt, V_{max} and friction will require a more dynamic analysis.

The work that is most related consists of other theoretical analyses of swarm systems. Our comparisons are in terms of the goal and method of analysis. There are generally two goals: stability and convergence/correctness. Under stability is the work by [5]–[7]. Convergence/correctness work includes [5]. Other goals of theoretical analyses include time complexity [8], synthesis [9], prediction of movement cohesion [5], coalition size [6], number of instigators to switch strategies [10], and collision frequency [11].

Methods of analysis are also diverse. Here we focus only on physics-based analyses of physics-based swarm robotics systems. We know of four methods. The first is the Lyapunov analysis by [7]. The second is the kinetic gas theory by [11]. The third is the minimum energy analysis by [9]. The fourth develops macro-level equations describing flocking as a fluid-like movement [12].

The capability of being able to set system parameters based on theoretical laws has enormous practical value. To the best of our knowledge, the only analyses mentioned above that can be used to set system parameters are those of [6], [10], [12]. The first two analyses are of behavior-based systems, while the latter is of a "velocity matching" particle system.

In the long run, we'd like to design and analyze virtual worlds based on AP. The theoretical results being developed here would formalize the multi-robot motions in such a virtual world, which would then influence the coordination of actual robots.

References

[1] W. Spears, D. Gordon-Spears, and R. Heil, "Distributed, physics-based control of swarms of vehicles in sensor network," *Autonomous Robots, Issue on Swarm Robotics*, submitted.

[2] A. Howard, M. Matarić, and G. Sukhatme, "Mobile sensor network deployment using potential fields: A distributed, scalable solution to the area coverage problem," in *Sixth Int'l Symposium on Distributed Autonomous Robotics Systems*, 2002.

[3] J. Kellogg, C. Bovais, R. Foch, H. McFarlane, C. Sullivan, J. Dahlburg, J. Gardner, R. Ramamurti, D. Gordon-Spears, R. Hartley, B. Kamgar-Parsi, F. Pipitone, W. Spears, A. Sciambi, and D. Srull, "The NRL micro tactical expendable (MITE) air vehicle," *The Aeronautical Journal*, vol. 106, no. 1062, 2002.

[4] W. Spears and D. Gordon, "Using artificial physics to control agents," in *IEEE International Conference on Information, Intelligence, and Systems*, 1999, pp. 281–288.

[5] Y. Liu, K. Passino, and M. Polycarpou, "Stability analysis of m-dimensional asynchronous swarms with a fixed communication topology," in *IEEE Transactions on Automatic Control*, vol. 48, no. 1, 2003, pp. 76–95.

[6] K. Lerman and A. Galstyan, "A general methodology for mathematical analysis of multi-agent systems," USC Information Sciences, Tech. Rep. ISI-TR-529, 2001.

[7] R. Olfati-Saber and R. Murray, "Distributed cooperative control of multiple vehicle formations using structural potential functions," in *IFAC World Congress*, 2002.

[8] S. K. O. Shehory and O. Yadgar, "Emergent cooperative goal-satisfaction in large-scale automated-agent systems," *Artificial Intelligence*, vol. 110, pp. 1–55, 1999.

[9] J. Reif and H. Wang, "Social potential fields: A distributed behavioral control for autonomous robots," in *Workshop on the Algorithmic Foundations of Robotics*, 1998.

[10] C. Numaoka, "Phase transitions in instigated collective decision making," *Adaptive Behavior*, vol. 3, no. 2, pp. 185–222, 1995.

[11] S. Jantz, K. Doty, J. Bagnell, and I. Zapata, "Kinetics of robotics: The development of universal metrics in robotic swarms," in *Florida Conference on Recent Advances in Robotics*, 1997.

[12] J. Toner and Y. Tu, "Flocks, herds, and schools: A quantitative theory of flocking," *Physical Review E*, vol. 58, no. 4, pp. 4828–4858, 1998.