The Genesis of Swarm Engineering

S. Kazadi
Jisan Research Institute
28 North Oak Avenue
Pasadena, CA 91107
USA
sanza@jisan.org

Abstract

This paper reviews the development of swarm based engineering through various examples drawn from the literature. The paper explores the current swarm intelligence paradigm and describes some of the problems with this paradigm. The paper describes a new method called *swarm engineering* which approaches the design and implementation of swarms in an entirely new way. The efficacy and limitations of this approach are discussed with examples from the literature.

1 Why the swarm?

Over the past several years, I have been working with swarms, both computationally, and for part of my professional life, with real rolling robots. The goal behind this work has been both pragmatic and intellectual. On the one hand, I've been working on swarms which have attempted to do specific things - track odor plumes to their source, complete work on large computational problems using multiple processors without centralized control. On the other hand, I've worked on swarms that have been very far removed from pragmatism in their application. These include the cluster-based construction swarms, which occupy the majority of my current time and effort. These swarms, though capable of demonstrating remarkable ability in the face of almost crippling ineptitude and potentially severe failure rates, will not be able to "earn their keep" for some years to come.

Yet despite some of the remarkable things that people have done and are doing with swarms, I believe that at least one major problem exists in the application of swarms. That is the idea, however even-minded, that swarms are or need to be intelligent. It is true that swarms carry out actions that may be construed as interesting and somewhat intelligent, but the idea that intelligence is the goal or outcome of the design of swarms is not only restrictive, but might seem to be altogether wrong. An ant colony does indeed carry out actions which might be characterized as intelligent, but to label the actions or the anthill itself as intelligent seems to be too much of a leap - an anthropomorphization without need or justification. The dynamics of the ant hill, we will argue, are all that are needed to be understood to fully characterize the anthill. Any intelligent or perceived intelligence is then in the eyes of the observer, and not in the reality of the system.

What this paper is, then, is an argument for a paradigm shift in the development and investigation of swarms. The idea that swarms should be made up of many different agents whose detailed individual behaviors are the central starting point for engineering work would seem to be a limiting factor in the design of swarms. This, of course, is motivated out of the need to take a simple behavior with limited intelligence and scale up to a complex behavior with more intelligence. The problem with this approach lies in the myriad of configurations a simple system can take on, and the apparent unpredicability of that system in the face of simple rules. Many of these systems, which are simple to articulate, boggle the mind in their complexities, and cannot be either predicted or controlled. This does not bode well for engineering systems.

What this paper is not is a detailed history of what has happened in swarm engineering over the past twenty or thirty years. (Yes it has actually been that long, and I'll explain why.) The paper is simply an attempt to pick up many of the salient points of swarms and to explore how these points lead to a new paradigm of swarm design. Of course, the most reasonable place to start is the beginning. So that's where we'll start.

2 The very first swarms and pseudoswarms

The first swarms on earth were not very sophisticated at all. In fact, from a certain point of view, these primitive swarms whose only communication was stigmergic, had both very simple designs and rules. The first swarms on earth were evolutionary swarms in which the swarming action was obtained by the entire population. Let's have a look at how this works.

Of course in evolutionary systems, the ultimate goal is to survive and reproduce. It would seem to be an individual effort. In actuality, the only individual effort is in evaluating the effectiveness of the particular set of genes one is given. The swarm becomes effective in the search for more effective genes in what becomes an ever-changing fitness field.

Let's look at this another way. At one point, I was very interested, as others were, in learning how to carry out olfactory search [15][18][20][24][28]. In olfactory search, the goal is to find a target by tracking the odor plume originating at the target to the target. The problem is difficult because the plume is typically in a turbulent fluid, and is, as a result, made up of lots of small packets of concentrated odorant. The plume tracker then needs to find the source having only the information found in these small packets of concentrated odorant. Of course, what makes this task possible is the fact that plumes exist only in wind, and the wind direction can be obtained by hairs or temperature differentials or other means.

Actually it seems almost ridiculous to use a swarm for this task. It's actually an easy task for a single agent to find its way to the source of an odor. Were this not the case, no insect would ever find its mate by following an emmitted odor - yet they do. In fact, it is so easy to do, that picnics are routinely visited by untold tens of bees, wasps, ants, and other unwelcome guests. So the obvious question then is why do we use a swarm?

Well it turns out that in the case of swarm based plume

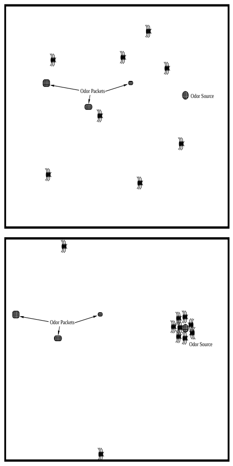

Figure 1: This swarm is involved in tracking a plume to its source in an unknown terrain. In (A), the swarm is initially randomly placed. In (B), it has found the target.

tracking, the swarm is actually capable of accurately tracking plumes that *individual elements of the swarm cannot* [18]. This is because, with simple communication, such as a noise or light, the entire swarm can be summoned to an area containing a single packet sensed by an individual in the swarm. The concentration of the entire swarm on the area where the plume fragment has been identified allows the swarm to reliably pick up more plume fragments and eventually locate the target [*Kazadi*, 2000]. In fact, the use of the swarm is so significant that it can cause an increase in sensitivity of two or more orders of magnitude, depending on the sensor and swarm.

Ironically, evolutionary systems behave in precisely the same way. However, they do this, not by recruitment, but rather by replacement. Consider the case that we have a limited population of any given animal in a closed environment (either by natural barriers or by man-made barriers). Natural limitations on the size of the population will be imposed by the system. These limitations will cause an evolutionary pressure towards subspecies which can more efficiently reproduce. Thus, those individuals which emerge with a competitive advantage in reproducing will come to dominate the population, both constraining the rest of the population and focusing the swarm's evolutionary activities. In the same way that the swarm of plume trackers focused the swarm's activities on a spatial region, the evolutionary swarms tended to focus the search through genome space in particular regions in which good structures have been found to be encoded.

The earliest work on this came from John Holland, Lawrence Fogel, and other colleagues. These works describe the first population-based evolutionary systems in which the earliest engineering work with swarms was done. Rather than simply having reproduction as the goal, as in the biological model, the goal was to create a swarm that satisfied a specific goal - construct a solution to numerical problems. The agents in the swarm carried out a dimensional search, with recruitment centering around competition for precious computational resources. This was the first time that engineering had been applied to swarm design, though the goal was the generation of designs that either yielded numerical optimization or artificially intelligent agents.

3 Swarms and artificial intelligence

It was Rodney Brooks [6] that brought to the forefront the idea that very simple agents can exhibit remarkably intelligent behavior despite their extremely simple design. Up to that time, artificial intelligence had been focused almost entirely on model-based investigations which had yet to exhibit dynamic decision-making behaviors of more than very specific applicability. However, Brooks' work illustrated that small robots could exhibit behaviors that had heretofore been attributed to intelligent agents. This led to a great deal of research dominated by the artificial life community and robotics community in which very simple agents were used to accomplish a large number of tasks of varying complexity, with very simple design.

About the same time, Jean-Louis Deneubourg [9-11] was carrying out various experiments using ants, demonstrating that these simple creatures could not only carry out sophisticated construction and reconstruction tasks, but they could make seemingly intelligent choices *as a group* [27]. These decisions exceeded the ants' individual capabilities, which made the research all the more interesting. In a landmark paper, Deneubourg illustrated the ability of an anthill to solve choice problems in which a shorter path between a food source and an anthill was chosen over a longer one , though no single ant ever knew how long either path was. Moreover, in other studies, Deneubourg and others demonstrated the ability of ants to build cemetary structures in which simple rules led to the completion of this high level task. These two simple examples, along with others drawn from the artificial life community, the robotics community, etc. became the basis for *swarm intelligence*.

Swarm intelligence is all about getting swarms to carry out tasks that we typically expect only intelligent agents to be able to carry out. These include tasks such as the construction of simple structures [7][8], the building of complex structures such as a bee hive or ant nest [14][17], the solving of numerical problems such as the choice of a minimal path, and other tasks such as that described in Section 2. In each case, the central focus of the research was the generation of high level behaviors using the coordinated efforts of simple agents using very simple behaviors. This was such a powerful design constraint that many very interesting projects struggled to fit into the mold of interesting high level behavior based on very simple initial behaviors [25].

One of these research foci centered around the ant cemetary phenomenon [4]. In this task, ants would move dead carcasses of other ants around an enclosed area, dropping them randomly in one place or another. After a long period of time, however, all of the ant carcasses would be in a single, central pile. The question of how this was accomplished was examined both experimentally, with real robots, and in simulation. The resulting algorithms were remarkably simple, as Deneubourg had demonstrated with ants. Simple push-and-back-away algorithms allowed robots to do the same work, though in these cases, the randomness of the system came from different sources than from the ants [1][16][17][23]. Thus, this minimally complex construction task could be solved by using extremely simple robots and algorithms.

The main difficulty with a swarm intelligence-based examination of swarms and their usefulness is that many of the designs of swarms are marginally capable of producing interesting behavior. A great deal of work on swarms has been done in which swarms were deliberately designed to do specific things, which on the face of it were interesting and exciting, but did not seem to lead to any specific real generalizable outcome [12-13][21-23]. That is, while the specific task was both interesting and satisfying, it was incapable of generating a method of generalizing to other tasks. This difficulty would seem to be the result of the lack of theoretical results that yeild new and interesting ways of utilizing swarms so as to complete high level tasks. The complexity of swarms is potentially so great that their design can be a daunting task.

Suppose that we have a two dimensional universe containing a swarm of independent autonomous agents. Each agent is able to move freely in two dimensions, and has no limitation on where or how it may move. Now, suppose that the agents are designed so that they will naturally align themselves to one another and move in a common direction. The direction they choose to move depends, let us suppose, on the initial orientation of the group of agents. What will happen with such a system? The goal of this swarm might be to exhibit *flocking* in which a number of agents go off in one direction together, perhaps in formation. While this may be one of the outcomes of the example system dynamics, it is dependant on a number of things, which need to be controlled for. First, the sensors in use by the robots can be a number different sensors, some of which are directional, and others of which are not. The accuracy with which the orientation of the robots in the swarm is ascertained, as well as the range of those robots. Depending on these parameters alone, a group of agents randomly oriented may break into multiple groups, all going off in different directions with respect to one-another. This is hardly the correct view of flocking, though it is a valid outcome of the low level rules introduced above. (It is also what happens in practice with real flocking organisms on occasion.)

The most common approach to handling different behaviors in the swarm that emerge unintended is a constant incremental tinkering with the system so as to yield a system that conforms to design specifications more correctly than that produced. This kind of tinkering can take weeks and even months to produce new interesting behaviors that work well, since the design of the system is essentially put under a directed random walk through design space. The experience and insight of the engineer can reduce the time required for redesign, but the need for the redesign still exists. Moreover, the final system is limited to those systems that can be generated *entirely* by such a random walk, and so excludes those systems that are both unlikely to be randomly arrived at or are unintuitive. To be fair, some of these systems are both facinating and ingenious. The problem is that progress is typically slow and piecemeal. A more powerful and general technique would seem to be needed for great leaps to be made.

4 The Emergence of Swarm Engineering

It was precisely this need that led me to what has become my current passion and which defines this paper. As a graduate student, I studied swarm-based plume tracking [18-20], puck clustering, and ant-based algorithms for the traveling salesman problem (TSP) [21]. All of these problems had their complexities and, indeed, idiosyncracies. The approaches of my colleagues and myself to these problems were both varied and myopic. Each solution centered around the individual problem, having little to do with one another. The clear problem with these solutions was that they had become so particular, and our method of arriving at each of them so specific to the problem, that no single effort seemed to benefit any other effort. Put simply, it was impossible for us to make one effort easier because of our experience with any of the others.

The infuriating thing about each effort was indeed the same, however. The complexity of behavior of our swarms was often times difficult to predict, and we spent alot of time tuning our working solutions so that they might work in increasingly difficult situations. However, these great efforts seemed to provide no clue about how to carry out any other efforts. Most importantly, however, despite calling our efforts swarm intelligence, we were moving further and further away from the intelligence analogy, and simply generating engineered swarms with particular goals in mind. What we were doing was engineering swarms, not investigating artificial intelligence.

Thus it became clear that a paradigm shift was needed. This paradigm shift centers around the generation of a new way of thinking about building up swarms. First, and perhaps most importantly, is the understanding that what we are doing is engineering swarms for particular purposes. These purposes are widely varied, and can take many different forms. Already, many of these have appeared in telecommunications, in path planning, in distributed sensing, and in other fields. Without the need to connect what we are doing to artificial intelligence, it is possible to create dynamic swarms whose properties have nothing to do with creating intelligent systems. Second, we are interested in generating swarms that are provable *before* construction. That is, we center our approach to swarm engineering on the development of what we call *swarm conditions* which lead to a high order global outcome when satisfied. The engineering work, then, is centered around building devices which satisfy these swarm conditions, after which the swarm will properly function. These conditions must take into account the dynamics of the swarm, making the transition from viewing the swarm's capability to handle complex situations in the way a rational agent might to viewing the same as an instance of the underlying system dynamics.

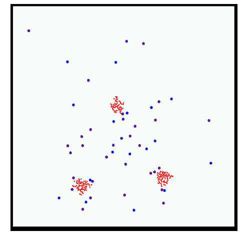

Figure 2: This figure illustrates one possible use of the swarm engineering paradigm. This illustrates the extensions available to the classic puck clustering problem once the swarm engineering paradigm has been applied. In this case, the pucks have been arranged in three equal sized clusters equally far from one another. In this image, the large circles represen t agents, and the smaller circles represent pucks.

> Let us consider an example, taken from [19]. Consider the cemetary ants discussed above. These ants, are capable of carrying out the simple clustering construction task without central control or any apparent knowledge of the task they're doing. In order to engineer a similar behavior, we need to determine under which conditions clusters of larger size will tend to absorb pucks from clusters of smaller size. This tacitly assumes that no individual cluster or single location is preferred over any other cluster or location. It has already been shown that what is needed for this to occur is that the smaller clusters must have a higher percentage of pucks coming off of them than the larger clusters, and there must be a continued circulation of pucks between the different clusters. *Any robotic system* satisfying these conditions will cause clustering to occur.

The power of this method is not immediately apparent, as it accomplishes the same things that many of the swarm intelligence approaches have accomplished. In fact, the early applications were directed solely at accomplishing these things. However, many of things done in this way are much more useful than swarm intelligence-based approaches. As an example, the same approach can be used to make predictions about the efficiency of certain clustering algorithms [18][20], allowing different robotic systems to be compared on a cost/efficiency basis as well as whether or not they accomplish their stated task.

Thus, I propose the following methodology for the generation of swarm designs for engineering systems.

- First, generate a clear statement of the goal of the swarm. What is the mission, and how will the mission be accomplished and its accomplishment validated.

- Generate a minimal condition or set of conditions required for the completion of the task at hand. This condition should be able to be validated independently of the swarm.
- Translate the minimal set of conditions to minimal agent requirements.
- Construct agents satisfying the minimal agent requirements

These steps will generate swarms that have the desired properties [20][26][29]. Moreover the use of the methods will allow comparison of the swarms before construction, saving the engineer time and effort. These methods are heretofore referred to as *swarm engineering techniques*, and would seem to be a set of simple and general guidelines for the generation of a good many dynamic and useful swarms.

5 Concluding remarks

Swarm design is now entering an entirely new phase, with significant engineering projects being undertaken for the first time with swarms. The benefits to the engineer are many. Despite the apparent complexity of the swarm, the design of the individual elements of the swarm are simple. The dynamics of the swarm can be both simple and efficient, or complex, depending on the design of the individual elements. Moreover, the swarm itself can be capable of much more than the individual agent is capable of. Finally, for physically instantiated swarms, the primary advantage is that the swarm's success does not need to be dependent on a single individual in the group, though some swarms may be designed in such a way that this is not the case.

With the growth of the use of swarms around the world, a hard look has to be taken at the paradigm we use in the development of swarm-based algorithms. While the previous methods have centered around the development of complex behavior, decision-making, and intelligence, the current needs seem to be quite different. Current needs seem to be very goal oriented, and are rooted in practicality, rather than simple academic curiosity. Thus, it is necessary to explore ways of building swarms in a way that does not get bogged down in the details of the dynamics of the swarm outside of what we intend for the dynamics. While it is generally impossible to foresee all of the potential dynamics of swarms that are built, it is important to be able to predict that at least those dynamics that we intend do come about. The swarm engineering approach would seem to solve all of these problems without limiting the applicability of the swarm paradigm.

Thus, I propose the swarm engineering paradigm as a general methodology to swarm generation. With such a methodology, each swarm can be come a simple result of the design specification of the swarm, with provable properties and a standard generation methodology. Not only is this already in practice today, but it would seem to be the method of choice for future efforts. While the understanding of intelligence and intelligent systems is an interesting pursuit in and of itself, it is perhaps a crippling restriction in practice for the general application of swarms. Moreover, the bottom up methodology is so replete with pitfalls, that it is unlikely to lead to swarms that defy human intuition, despite the likelihood that nonintuitive behaviors lead to desired global outcomes. For these reasons, it may be beneficial for engineers to use a top-down methodology in building swarms of computational, simulated physical, or real physical agents capable of solving complex computational and physical problems.

References

[1] R. Beckers, O. Holland, and J. Deneubourg. *From local actions to global tasks: stigmergy and collective robotics.* **Proceedings of the Fourth International Workshop on the Synthesis and Simulation of Living Systems**, 1994.

[2] E. Bonabeau, A. Sobkowski, G. Theraulaz, and J. Deneubourg. *Adaptive task allocation inspired by a model of division of labor in social insects.* **Bio Computation and Emergent Computing**. D. Lundh, B. Olsson, and A. Narayanan, eds. World Scientific: 36-45, 1997.

[3] E. Bonabeau, E. Henaux, S. Guerin, D. Snyers, P. Kuntz, and G. Theraulaz. *Routing in telecommunications networks with smart ant-like agents.* **Proceedings Intelligent Agents for telecommunications Applications**, Berlin, Springer-Verlag, 1998.

[4] E. Bonabeau, G. Theraulaz, V. Fourcassie, and J. Deneubourg. *Phase ordering kinetics of cemetary organization in ants.* **Physical Review E**. 57 (4), 1998.

[5] E. Bonabeau, G. Theraulaz, J. Deneubourg, N. Franks, O. Rafelsberger, J. Joly, and S. Blanco. *A model for the emergence of pillars, walls, and royal chambers in termite nests.* **Phil Trans. Roy. Soc. London B**, 1998.

[6] R. Brooks. *A robust layered control system for a mobile robot.* **IEEE J. Robotics and Automation**, RA-2 (1), 14-23, 1986.

[7] S. Camazine, J. Sneyd, M. Jenkins, and D. Murray. *A mathematical model of self organized pattern formation on the combs of honeybee colonies.* **Journal of Theoretical Biology**, 147, 553-571, 1990.

[8] Y. Cao, A. Fukanaga, and A. Kahng. *Cooperative mobile robotics: antecedents and directions.* **Autonomous Robots**: 4, 1-23, 1997.

[9] J. Deneubourg, J. Pasteels, and J. Verhaeghe. *Probabilistic behaviour in ants: a strategy of errors?* **Journal of Theoretical Biology**, 105: 259-271, 1983.

[10] J. Deneubourg and S. Goss. *Collective patterns and decision-making. Ethology, Ecology, and Evolution*, 1:295-311, 1989.

[11] J. Deneubourg, S. Goss, N. Franks, A. Sendova-Franks, C. Detrain, and L. Chretien. *The dynamics of collective sorting. Robot-like ants and ant-like robots.* **From Animals to Animats**. J. A. Meyer and S. W. Wilson, eds., Cambridge, MA: MIT Press/ Bradford Books, 1991.

[12] M. Dorigo, V. Maniezzo, and A. Colorni. *The ant system: optimization by a colony of cooperating agents.* **IEEE Transactions on Systems, Man, and Cybernetics B**, 26: 29-41, 1996.

[13] M. Dorigo and L. Gambardella. *Ant colony systems: a cooperative learning approach to the travelling salesman problem.* **IEEE Transactions on Evolutionary Computation**, 1(1): 53-66, 1997.

[14] N. Franks, A. Wilby, B. Silverman, and C. Tofts. *Self-organizing nest construction in ants: sophisticated building by blind bulldozing.* **Animal Behavior**, 44: 357-375, 1992.

[15] F. Grasso, T. Consi, D. Mountain, and J. Atema. *Behavior of purely chemotactic robot lobster reveals different odor dispersal patterns in the jet region and the patch field of a turbulent plume.* **Biol. Bull.**, 191: 312- 313, 1996.

[16] O. Holland and C. Melhuish. *Stigmergy, self organization, and sorting in collective robotics.* **Artificial Life**, 5:173-202, 1999.

[17] I. Karsai and Z. Penzes. *Comb building in social wasps: self-organization and stigmergy script.* **Journal of Theoretical Biology**, 161: 505-525, 1993.

[18] S. Kazadi. **Swarm Engineering**. Caltech Doctoral Thesis, 2000.

[19] S. Kazadi. *On the convergence of puck clustering systems.* **Robotics and Autonomous Systems**. 38 (2), 93-117, 2002.

[20] S. Kazadi, R. Goodman, D. Tsikata, D. Green, H. Lin. *An Autonomous Water Vapor Plume Tracking Robot Using Passive Resistive Polymer Sensors.* **Autonomous Robots**, 9(2): 175-188, 2000.

[21] D. Lee, S. Kazadi, R. Goodman *Swarm Engineering for TSP.* **Proceedings of ANTS'2000**.

[22] A. Martinoli, A. Ijspeert, and F. Mondada. *Understanding collective aggregation mechanisms: from probabilistic modelling to experiments with real robots.* **Robotics and Autonomous Systems**, 29: 51-63, 1999.

[23] C. Melhuish. **Strategies for Collective Minimalist Mobile Robots**, Ph.D. Thesis, University of West of England, 1999.

[24] M. Renou and P. Lucas. *Sex pheromone reception in* Mamestra brassicae *L. (lepidoptera): responses of olfactory receptor neurons to minor components of the pheromone blend.* **J. Insect Physiology**, 40(1): 75-85, 1994.

[25] R. Schoonderwoerd and O. Holland. *Ant-based load balancing in telecommunications networks.* **Adaptive Behavior**, 5(2): 169-207, 1997.

[26] W. Spears and D. Gordon. *Using Artificial Physics to Control Agents.* **Proceedings of IEEE International Conference on Information, Intelligence, and Systems**, November, 1999.

[27] G. Theraulaz, E. Bonabeau, and J. Deneubourg. *Response threshold reinforcement and division of labour in insect societies.* **Proceedings of the Royal Society of London, Series B**, 265: 327-332, 1998.

[28] P. Zanen and R. Carde. *Effects of host-odour plume altitude and changing wind velocity on upwind flight maneuvers of a specialist braconid parasitoid.* **Physiological Entomology**, 21:329-338, 1996.

[29] A. Zhang, M. Chung, B. Lee, R. Cho, S. Kazadi, and R. Vishwanath. *Variance in Converging Puck Cluster Sizes.* **Proceedings of AAMAS'02**, July 2002.