Pattern Formation in Wireless Sensor Networks

Thomas C. Henderson
University of Utah

Kyle Luthy and Edward Grant
North Carolina State University

UUCS-08-008

School of Computing
University of Utah
Salt Lake City, UT 84112 USA

6 August 2008

Abstract

Biological systems exhibit an amazing array of distributed sensor/actuator systems, and the exploitation of principles and practices found in nature will lead to more effective artificial systems. The retina is an example of a highly tuned sensing organ, and the human skin is comprised of a set of heterogeneous sensor and actuator elements. Moreover, the specific organization and architecture of these systems depends on contextual influences during the developmental stages of the organism. Comparable theoretical and technological methodologies need to be found for wireless sensor networks. We propose the study of reaction-diffusion systems from mathematical biology as a starting point for this endeavor. Algorithms and experiments are described here for a useful set of pattern formation methods in wireless sensor networks.

1 Introduction

Alan Turing introduced a revolutionary reaction-diffusion model as the chemical basis of morphogenesis [28], and this method lends itself particularly well to pattern synthesis in distributed systems. For more detailed explanations, see his original paper (which provides an exemplar of the scientific paper – theory, analysis and numerical solution on the Manchester machine which Turing helped design and build!), as well as the works of Murray [20], Meinhardt [18], and more recently, Maini and Othmer [17]. Turing's key insight was that diffusion of an inhibitory morphogen could lead to the formation of stable and variegated patterns. This is related to nonlinear far from equilibrium thermodynamics, and dissipative structures (e.g., see Prigogine [23, 25, 26] who received the Nobel prize in chemistry for work in this area). One goal of our work is to understand how these principles are at work in biological sensor systems and how they may be exploited in wireless sensor networks.

We have previously proposed to use Turing's reaction-diffusion mechanism to generate patterns in wireless sensor networks. [10]. The basis of this mechanism is a set of equations that capture the reaction and diffusion aspects of certain chemical kinetics:

$$\frac{\partial \mathbf{c}}{\partial t} = f(\mathbf{c}) + D\nabla^2\mathbf{c} \tag{1}$$

where $f(\mathbf{c})$ describes the reaction and $D\nabla^2 c$ expresses the diffusion component. The simplest such systems have two *morphogens* or variables; one of these acts as the activator and the other acts as the inhibitor. The two variable system can be modeled by:

$$\frac{\partial u}{\partial t} = \gamma f(u,v) + \nabla^2 u, \frac{\partial v}{\partial t} = \gamma g(u,v) + d\nabla^2 v \tag{2}$$

where u and v are the concentrations of the morphogens, d is the diffusion coefficient and γ is a constant measure of scale. The functions $f(u,v)$ and $g(u,v)$ represent the reaction kinetics. As an example, we have explored the generation of spatial patterns using the Turing system of equations:

$$f(u,v) = \beta - uv, g(u,v) = uv - v - \alpha$$

where u and v are the morphogen concentrations, α and β are the decay and growth rates, respectively, and γ sets the speed of the reaction. They define a domain in which Equation (2) becomes linearly unstable to certain spatial disturbances. This domain is referred to as *Turing space* where the concentrations of the two morphogens will become unstable and result in the patterns shown in Figure 1.

The pattern is the result of each cell running the equations locally while *diffusing* to its neighbors; a stable solution may be thresholded to produce a binary value at each sensor,

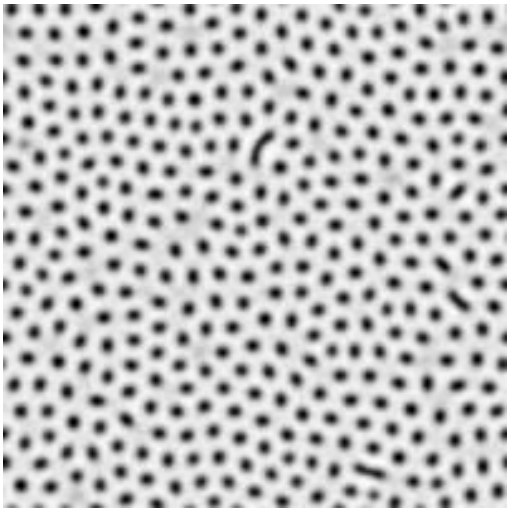

Figure 1: Turing Spot Pattern.

and the total of these gives the pattern. Note that the distribution of these spots is close to hexagonal.

We introduced the use of Turing's reaction-diffusion pattern formation to support high-level tasks in sensor networks (*S-Nets*). This has led us to explore various biologically motivated mechanisms. We address below some issues that arise in trying to get reliable, efficient patterns in irregular grids.

Much remains to be done at the higher level of information extraction, interpretation and exploitation of networked sensor systems. Our central thesis is that bio-based engineering will lead to strong solutions in this domain; that is, we propose to identify and ultimately incorporate effective computational strategies used by biological systems. The challenge is to identify mechanisms that lead to algorithms or paradigms that are reliable, inexpensive and ubiquitous in many applications.

Others have explored the use of both reaction-diffusion and more general diffusion methods in computer vision and robotics. For example, Fukuda et al. describe the use of reaction-diffusion techniques in robot motion[6]. Moreover, as described by Peronna et al.[24], multi-scale descriptions of images (i.e., scale-space) can be produced by embedding the original image in a family of images obtained by convolving the original image with a filter; Koenderink[15] showed that this is equivalent to finding the solution of the diffusion equation:

$$I_t = \bigtriangledown^2 I = I_{xx} + I_{yy}$$

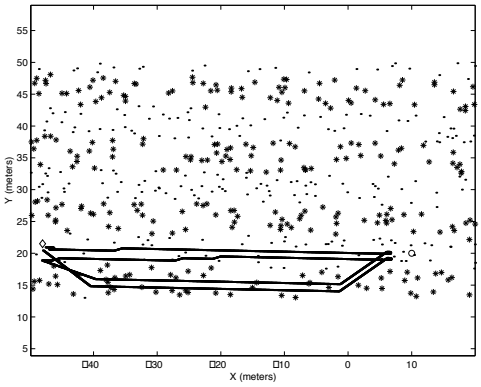

Figure 2: Robot Path in Reaction Diffusion Pattern ($\circ$ is the fire control point; $\diamond$ is the robot load point)

We believe that it will be quite useful for *S-Nets* to use similar methods to analyze sensed data of various sorts. Other proposed diffusion models include, for example, [11] who proposes directed diffusion - a datacentric communication coordination technique that "enables energy savings by selecting empirically good paths and by caching and processing data in- network." The focus of such work is more on the networking and operating systems aspects of the sensor network, whereas our work is more concerned with the sensor network as a computation engine itself. More closely related to our work is that of Justh and Krishnaprasad [12] who propose the active coordination of a large array of microactuators by means of diffusive coupling implemented as interconnection templates, and Nagpal [22] who describes methods to create patterns of diverse geometry. We believe that this style of research will reap great benefits in three aspects: (1) network *morphogenesis*, (2) sensed data analysis, and (3) display pattern synthesis.

As Meinhardt points out [18], "the control of development in a higher organism is one of the major unresolved problems in biology ... in a developmental system a signaling and signal-receiving mechanism must exist which enables the cell to communicate in a manner appropriate to its position ... [the] goal is to show which interactions of substances can lead to such signaling systems and how the cells then can respond to these signals in order that stable states of determination are attained." This matches our view of the core issues, and we see that their solution can heavily impact sensor network algorithms as well.

For example, consider a forest fire scenario: sensor devices are dropped into a wide geographic area, establish a network, compute coordinate frames, calculate gradients, and produce a stripe pattern of off-on signals that can be used by fire fighting agents to go to a fire control point by following *on* devices (pattern == 1) and return by following *off* devices (pattern == 0) (see Figure 2). Such patterns can be computed by very robust reaction-

diffusion systems derived from models of biological pattern formation.

Our general research program is to explore a small set of biological sensing and signaling mechanisms, and we hope to make significant contributions by providing (1) biologically realistic models and efficient computational counterparts, (2) fault tolerant frameworks in which to run them, and (3) demonstrations of their application in human interface and large-scale sensor networks. In addition, we are building *S-Net* simulation, emulation, and experimentation testbeds [9]. Here we describe some initial results in the first of these areas.

Patterns in the *S-Net* can be used to support many high-level algorithms or activities:

- stripe, spot or ring patterns can be used as encoders for physical or logical purposes; for example, a robot can keep track of how far it has traveled (physical), or communication packets can travel along certain stripes to minimize power cost or to avoid congestion (logical).
- certain sets of patterns form a basis set for 2D images (e.g., Haar or Hadamard basis sets); any map (topo, etc.) or image can then be encoded in terms of the coefficients associated with the respective basis images.
- the patterns can be used as a reference wave so that sensed data (or features derived from it) can be encoded as an interference pattern (i.e., a hologram)
- moving waves can also be computed, and thus the *S-Net* can serve as a signal carrier or modulator.

Understanding the precision and reliability of pattern formation is then of high importance.

The most common application of *S-Nets* is to serve as a sensory organ; e.g., to capture images, sounds, chemical concentrations, temperature, etc., over the region of interest. However, *S-Nets* may also be used as a display, either directly through LED's or by making values avaliable upon query. A biological analogy is the skin; a zebra's stripes provide coloration as well as a myriad of sensors embedded in the skin, and these serve to provide some ecological advantage. A combination of these is especially interesting where the display is influenced by the sensing; e.g., for camouflage effects like in the chameleon.

Given a set of *SEL*s, sensor elements, in the plane, it may be useful to store and exploit a pattern in the *S-Net*. For example, stripes may be useful for several purposes: (1) as pathways for physical or logical tasks, e.g., mobile agents trajectories or packet transmission,

(2) as distance encoders as mobile agents cross them, or (3) as boundary markers. The ability to store arbitrary patterns (e.g., maps) has been shown useful [8, 4] for calculating shortest paths for robots to follow through terrain with varying traversability properties.

If the *SEL*s are situated in a coördinate frame, i.e., each *SEL* knows its (x, y) location, then given a specification of the pattern as a function $f(x, y)$, each *SEL* can determine its own value. Useful binary patterns may also be represented as a bit stored at each *SEL*, which requires a thresholding function $t(x, y)$, as well; this works well for paths or checkerboards. Moreover, combinations of stripe and checkerboard patterns may serve as a set of 2D basis functions to allow the representation of any arbitrary pattern as a linear combination of them.

2 Regular Geometric Figures

Equi-spaced straight line segments, stripes, with orientation θ, may be easily computed by means of the following function ([21]):

$$f(x, y) = cos(xcos(\theta) + ysin(\theta))$$

Figure 3 shows the result with $\theta = 0$ and $\theta = \pi/4$.

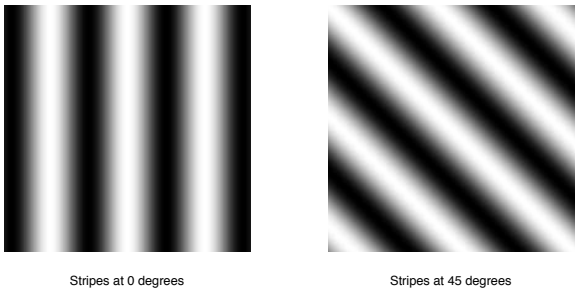

Figure 3: Vertical and 45^o Stripes.

If the range of $f(x, y)$ is $[-1, 1]$, then a mobile agent can determine how far it is from the center of the nearest stripe. Consider now the case of a set of *SEL*s randomly distributed in a square area (each x and y coördinate is sampled from a uniform distribution). Figure 4 shows an instance of 1,000 *SEL*s. Each *SEL* can determine its own value using the formula given above, and mobile agents can request this information in order to stay near the stripe of interest; in this way, the mobile agent does not need to be incorporated into the *S-Net*

coördinate frame. Figure 5 shows the set of *SEL*s with values near 1 shown as a circle ('o') and values near -1 shown as a dot ('.') for both stripe orientation 0 and $\pi/4$.

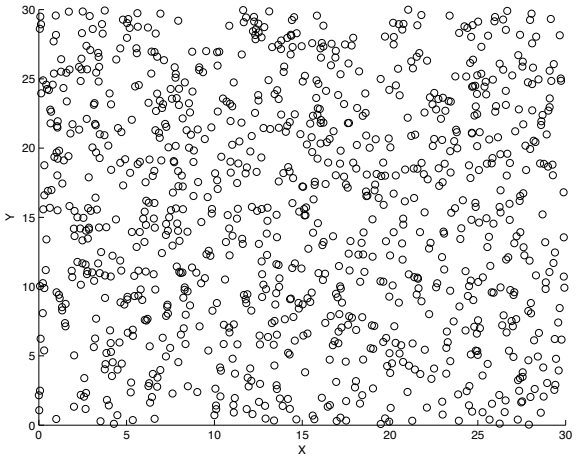

Figure 4: Sample of 1,000 *SEL*s in a Square Area.

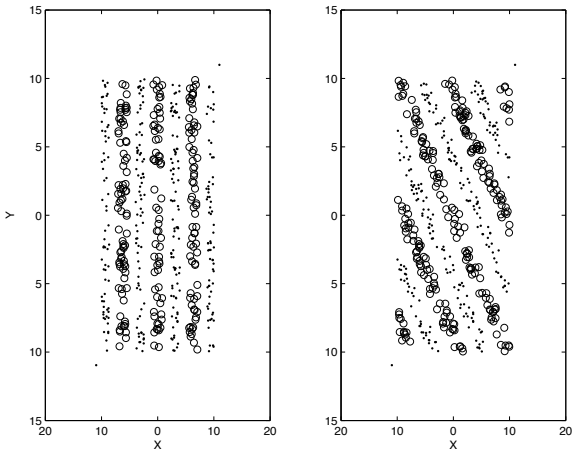

Figure 5: Vertical and 45^o Stripes in *S-Net*.

Another useful pattern is the checkerboard. The function:

$$f(x, y) = cos(x)cos(y)$$

thresholded by $0 < f(x, y)$ yields a checkerboard oriented along the x and y axes (see Figure 6 for both the original and thresholded versions). Using 2,000 *SEL*s, it is possible to approximate the squares in an *S-Net* (thresholded at $0.3 < f(x, y)$ and $f(x, y) < -0.3$ as shown in Figure 7).

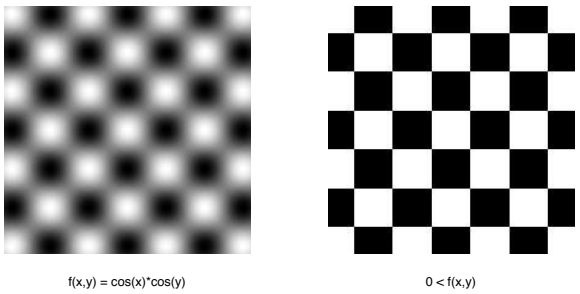

Figure 6: Checkerboard in the Plane.

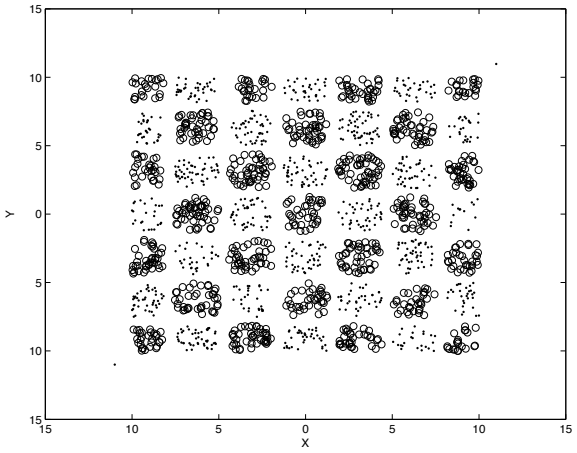

Figure 7: Checkerboard in *S-Net*.

As a final example, consider the hexagonal structure defined by:

$$f(x,y) = cos(\frac{y\sqrt{3}+x}{2}) + cos(\frac{y\sqrt{3}-x}{2}) + cos(x)$$

shown in Figure 8 with both original values and thresholded at $0 < f(x,y)$. Figure 9 shows how this would be displayed by an *S-Net*.

Of course, if coördinates are available, any pattern which can be expressed as a 2D function can be approximated by an *S-Net*. The quality of the approximation must be evaluated in probabilistic terms when the distribution of *SEL*s is random.

These regular patterns have been implemented on a set of Moteiv Tmote Sky motes, and the results are shown in Figures 10 through 14. Figure 10 shows the set of the 42 motes as laid out for these experiments. Figures 11 and 12 show the vertical and 45^o lines, while Figures 13 and 14 show the checkerboard and hexagonal layouts, respectively.

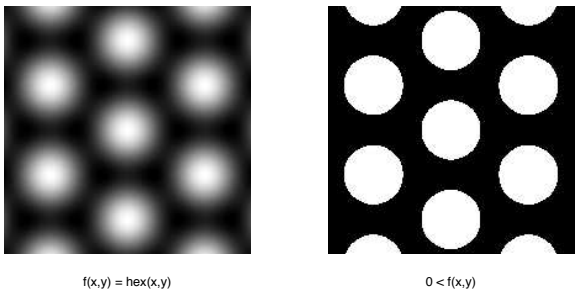

Figure 8: Hexagonal Structure.

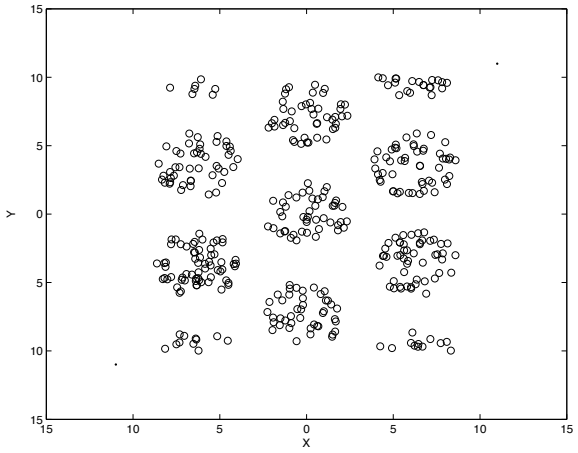

Figure 9: Hexagonal Structure in *S-Net*.

3 Reaction-Diffusion Patterns

Some work has already been done to determine the range and type of patterns possible with the Turing pattern formation approach. Theoretical aspects have been studied and regions of the parameter space characterized as they relate to pattern formation (i.e., the parameters are the coefficients in the PDEs) [2, 7, 16]. Others have investigated how pattern formation is influenced by number of cells, time scale, and initial condition variation. In particular, Bard and Lauder [3] showed that "stable repeating peaks of chemical concentration of periodicity 2-20 cells can be obtained in embryos in periods of time less than an hour. We do find however that these patterns are not reliable. Small variations in initial conditions give small but significant changes in the number and positions of observed peaks." They showed that this method has difficulty producing exact patterns reliably. We have found other difficulties in producing the patterns necessary to support higher-level tasks. We describe these here and propose some solutions.

Figure 10: Layout of 42 X-bow T-Sky Motes.

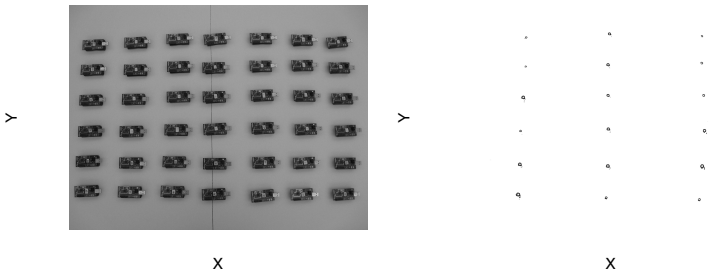

Figure 11: Vertical Lines in T-Sky Motes.

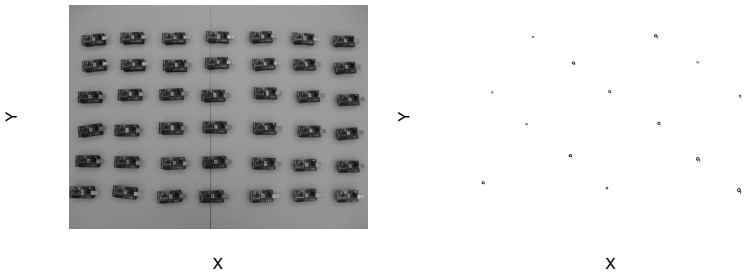

Figure 12: 45^o Lines in T-Sky Motes.

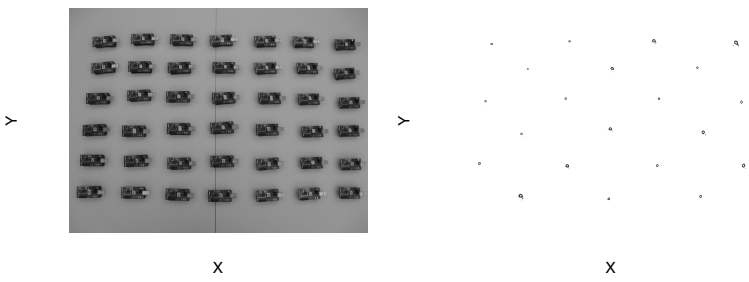

Figure 13: Checkerboard Layout in T-Sky Motes.

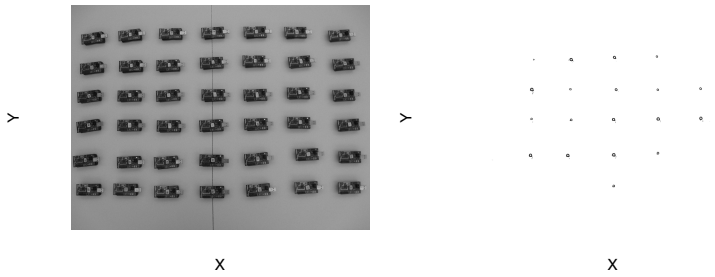

Figure 14: Hexagonal Layouts in T-Sky Motes.

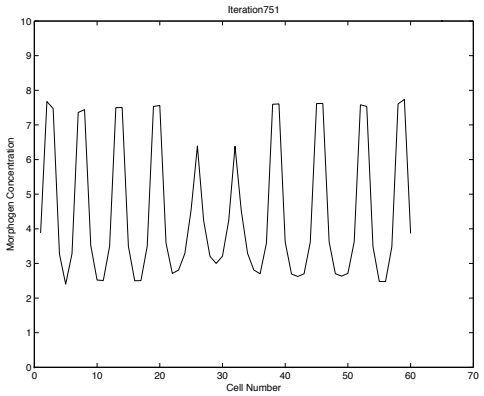

Figure 15: Typical 1-D Turing Pattern.

A more significant issue for us is that the reaction-diffusion pattern formation equations assume that the inter-cell distance is uniform (and usually equal to 1). Our *S-Nets*, however, do not form a uniformly spaced grid in 1D or 2D; in fact, we generally assume that the sensor devices are randomly dropped in the environment. In addition, the diffusion part of the equations uses the inter-node distances in the computation of the second derivative. Two concerns are:

- these distances are not uniform, and
- in an actual implementation, there will be some amount of error in the inter-node distance determination.

This has led us to investigate the impact of non-uniform spacing on the pattern computation.

The basic 1D Turing reaction-diffusion mechanism produces a pattern as shown in Figure 15, and takes about 1,040 iterations to converge for a set of 60 cells. A set of simulation experiments were run with from 10 to 100 cells in steps of 10. Table 1 gives the results for

number of cells	mean steps	mean stripes
10	720	1.35
20	883	3.14
30	938	4.83
40	955	6.62
50	998	8.30
60	1,041	9.89
70	1,047	11.54
80	1,086	13.16
90	1,102	14.80
100	1,113	16.49

Table 1: Mean steps and stripes

max x/y offset	≤ 0.3	0.35	0.40	0.45	0.50	0.55	0.60	$0.65 \geq$
avg failures per trial	0.0	0.06	0.11	0.58	0.85	0.94	0.96	1.0

Table 2: Average Number of Convergence Failures as a Function of x/y Offset from Grid Location.

the mean number of steps to converge (total change in morphogen a is less than 0.0001), and the mean number of stripes formed.

Next consider what happens when non-unit distances are introduced as the inter-device distances. The point locations are determined as follows:

- start with n equi-spaced points, 1 unit length apart,
- add uniform noise to the location with max x and y distances from the grid positions varying from 0 to 0.65.

Table 2 gives the average number of failures to converge for the Turing pattern.

This result, as well as similar results with 2D patterns, argues against using actual inter-SEL distances for the reaction-diffusion computation. However, this turns out to be an advantage since the inter-SEL distances may be difficult to determine. Moreover, the patterns will form based purely on the topological nature of the SEL interconnectivity graph. This can be very advantageous in 2D pattern formation in the *S-Net*.

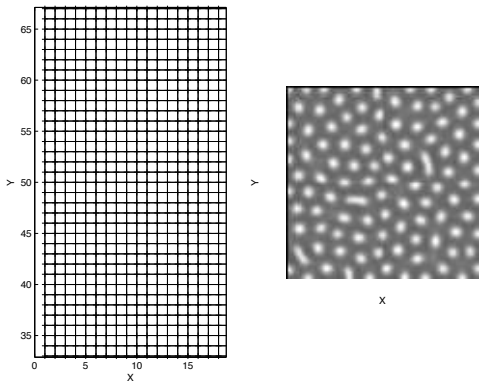

Figure 16: *S-Net* Regular Grid 2-D Turing Pattern.

Figure 1 shows the Turing pattern, but it is formed in an $n \times n$ array, where every array element has two vertical neighbors and two horizontal neighbors (boundary elements wrap around to the opposite boundary elements), and all neighbors are at unit distance. This makes the Laplacian calculation reasonably accurate.

As a first approximation to an arbitrary *S-Net*, we produce a set of *SEL*s whose locations correspond to these grid locations. However, if the grid connectivity is kept, and unit distances are used in the Laplacian, then convergence occurs. Figure 16 shows a close up of part of the grid, as well as the concentrations of morphogen a resulting from an execution of the reaction-diffusion code at each *SEL*. The concentration of morphogen a in the *S-Net* is displayed by producing an image array of specified size and assigning the appropriate gray level at each pixel according to the amount of morphogen a at each *SEL* located in the corresponding pixel. Also, note that in the *S-Net*, there is no wrap-around diffusion at the boundaries.

Similar to the 1D case, we find that if the locations of the *SEL*s are perturbed off the grid locations, and we use the actual inter-*SEL* distances to compute the reaction-diffusion equations, then the failure rate of convergence increases with distance from grid locations. Figure 17 shows a set of *SEL*s perturbed by up to 2 pixels in x and 2 pixels in y from the grid positions, but maintaining the same connectivity as the grid, and the corresponding pattern computed by the reaction-diffusion process. This is the pattern after 3,000 iterations, but it has not yet stabilized and the spots have not yet developed.

Next, we consider the case of *SEL*s randomly distributed in the square area. It turns out that if all neighbors within a certain distance (e.g., broadcast range) are used in the reaction-diffusion calculation, and the distances are used to compute the Laplacian, then the process generally fails to converge. However, if each *SEL* randomly selects four of its neighbors

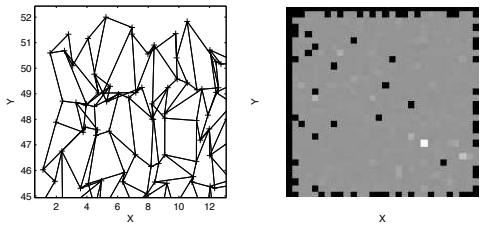

Figure 17: *S-Net* of Perturbed Regular Grid 2-D Turing Pattern.

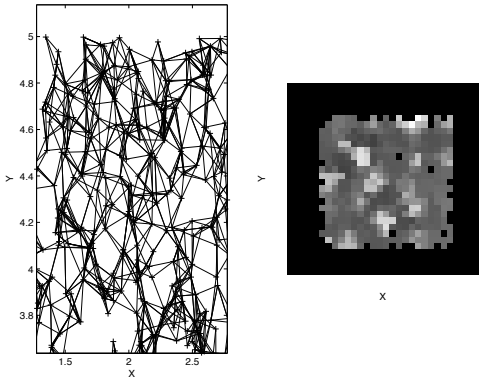

Figure 18: *S-Net* of Randomly Placed *SEL*s and the Resulting 2-D Turing Pattern using 4 Randomly Selected Neighbors.

(e.g., from the broadcast connectivity graph), then the reaction-diffusion process converges. Figure 18 shows an example of this on a 5×5 square area using 2,000 *SEL*s placed by sampling the x and y coordinates from the uniform distribution. Figure 19 shows the layout of a set of 42 X-Bow T-Sky motes; with the small number of motes, the reaction-diffusion process leads to the formation of spots (i.e., the morphogen concentration $a > 5$). In one instance, a spot appeared after 1,250 iterations; however, the spot pattern is not stable, and eventually disappears if allowed to continue. We believe that several issues are at play: the number of nodes, the asynchronous nature of the mote transmissions, as well as the locality and bi-directionality of the connectivity.

One further example of the application of the spot formation reaction-diffusion process is to create straight lines in *S-Nets*. This can be done without knowing the distance between motes, and without a coördinate frame. Suppose the goal is to create stripes as shown in Figure 2. At each mote, determine its neighbors with lowest and highest sensor measurements. These are assumed to lie in the gradient direction. If the diffusion is restricted to take place only through these neighbors, then the spots blend in this direction and become straight lines. Figure 20 shows this result on a simulated set of motes.

Figure 19: Layout of a set of 42 X-Bow T-Sky Motes.

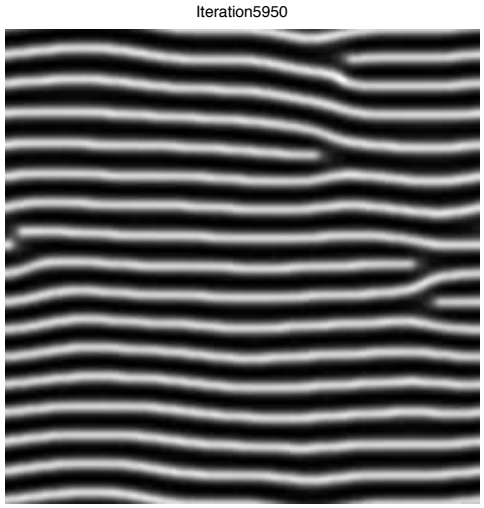

Figure 20: Lines produced by 2D Reaction-Diffusion Process.

4 Conclusions and Future Work

We have described techniques for forming patterns in sensor networks based on trigonometric functions and reaction diffusion equations. Such methods may be used to process data, although that is not discussed here, and to carry sensed signal information. The next focus of our work is on the production of patterns based on sense data analysis (e.g., camouflage synthesis). Such methods may also find application in sensor network security; in this scenario, a deformed pattern will emerge from a distributed computation if there are any nodes which have fallen victim to attack, or if external nodes have managed to get themselves incorporated into the *S-Net*. Of course, resource allocation and exploitation may also be based on patterns, and given the random nature of the patterns, may help conserve resources (e.g., energy) overall.

We are also exploring mesh generation in wireless sensor networks. As was pointed out by Adamatzky et al. [1], physical reaction-diffusion computers can calculate Voronoi graphs. Thus, a basis exists in the reaction-diffusion computation to produce good triangulations for mote connectivity. The triangle are also important for computations on irregular meshes; e.g., for finite element methods. We are exploring this in the context of a larger set of motes.

Another area of research is the calculation of level sets [27] in the *S-Net*. These can be used for shortest path computation where an arbitrary speed function may be defined. We have shown how mobile robots can use this approach to find the lowest time path to traverse variable speed terrain [5]. However, the Eikonal equation used there may be set up as a reaction-diffusion system, and piggy-backed on the approach defined here.

Finally, stripes and spots may be of use for various purposes by mobile agents or the *S-Net*, but a more direct combination of the *S-Net* as both a sensing and display device is to be found in the creation of active camouflage. Consider the truck shown in Figure 21. Although it has a standard camouflage tarpaulin, it does not blend well with the forest behind it. Several problems exist with coloration and blobs versus stripes (e.g., tree trunks and branches), leaf texture, etc.

One approach to overcome this mismatch is to pair a modeling process of the natural scene behind the truck with a display synthesis component in front of the truck. The technical basis for such a mechanism can be found in the work of Zhu and Mumford [29]. They propose (1) a theory for discovering the statistics of a set of natural images, and (2) a framework which allows the definition of reaction-diffusion equations to produce similar natural images, and in particular, they show how to remove conspicuously dissimilar segments from a scene. Specifically, they show that given a learned set of prior models that reproduce the

Figure 21: A Camouflaged Truck in the Forest.

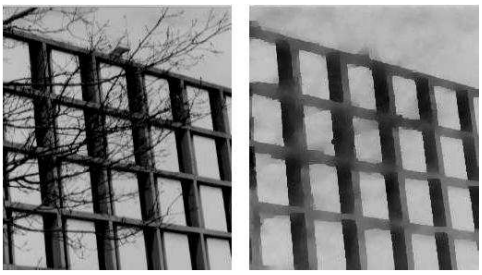

Figure 22: Zhu and Mumford [29] Clutter Removal Example.

observed statistics, the potentials of the resulting Gibbs distributions have potentials of the form:

$$U(I;\Lambda,S)=\sum_{\alpha=1}^{K}\sum_{x,y}\lambda^{\alpha}((F^{(\alpha)}\times I)(x,y))$$

where $S=\{F^{(1)},F^{(2)},\ldots,F^{(K)}\}$ is a set of filters and $\Lambda=\{\lambda^{(1)},\lambda^{(2)},\ldots,\lambda^{(K)}\}$ is the set of potential functions.

Reaction-diffusion equations are found as the gradient descent partial differential equations on $U(I;\Lambda,S)$; diffusion arises from the energy terms while pattern formation reactions are related to the inverted energy terms. These are then used to remove clutter in a scene and to denoise images. One example of this process given by Zhu and Mumford is shown in Figure 22. We propose that the resulting images can be displayed by LEDs distributed throughout the material of the camouflage canvas, and based upon our previous work in e-textiles [13, 14, 19], we believe that technical solutions exist for the realization of this goal.

References

[1] Amdy Adamatzky, Ben De Lacy Cotello, and Tetsuya Asai. *Reaction-Diffusion Computers*. Elsevier, Amsterdam, The Netherlands, 2005.

[2] A. Babloyantz and J. Hiernaux. Models for cell differentiation and generation of polarity in diffusion-governed morphogenetic fields. *Bulletin of Mathematical Biology*, 37:637–657, 1975.

[3] J. Bard and I. Lauder. How Well does Turing's Theory of Morphogenesis Work? *Jnl Theor Biology*, 45:501–531, 1974.

[4] Yu Chen. Snets: Smart sensor networks. Master's thesis, University of Utah, Salt Lake City, Utah, December 2000.

[5] Yu Chen. Snets: Smart sensor networks. Master's thesis, University of Utah, Salt Lake City, Utah, December 2000.

[6] Dario Floreano, Jean-Daniel Nicoud, and Francesco Mondada, editors. *Advances in Artificial Life, 5th European Conference, ECAL'99, Lausanne, Switzerland, September 13-17, 1999, Proceedings*, volume 1674 of *Lecture Notes in Computer Science*. Springer, 1999.

[7] M.I. Granero, A. Porati, and D. Zanacca. A bifurcation analysis of pattern formation in a diffusion goverened morphogenetic field. *Jnl of Mathematical Biology*, 4:21–27, 1977.

[8] Thomas C. Henderson, Mohamed Dekhil, Scott Morris, Yu Chen, and William B. Thompson. Smart sensor snow. *IEEE Conference on Intelligent Robots and Intelligent Systems*, pages 1377–1382, October 1998.

[9] Thomas C. Henderson, Jong-Chul Park, Nate Smith, and Richard Wright. From motes to java stamps: Smart sensor network testbeds. In *Proc International Conf on Intelligent Robots and Systems*, page to appear, Las Vegas, NV, October 2003.

[10] Thomas C. Henderson, Ramya Venkataraman, and Gyounghwa Choikim. Reaction-diffusion patterns in smart sensor networks. In *Proc International Conference on Robotics and Automation*, New Orleans, April 2004.

[11] C. Intanagonwiwat, R. Govindan, and D. Estrin. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In *Proc. of Mobicom 2000*, Boston, August 2000.

[12] E.W. Justh and P.S. Krishnaprasad. Pattern-forming systems for control of large arrays of actuators. *Jnl of Nonlinear Sci*, 11(4):239–277, 2001.

[13] T. Kang, C.R. Merritt, B. Karaguzel, J.M. Wilson, P.D. Franzon, B. Pourdeyhimi, E. Grant, and T. Nagle. Sensors on textile substrates for home-based healthcare monitoring. In *Conference on Distributed Diagnosis and Healthcare (D2H2)*, pages 5–7, Arlington, VA, April 2006.

[14] B. Karaguzel, C.R. Merritt, T.H. Kang, J. Wilson, P. Franzon, H.T. Nagle, E. Grant, and B. Pourdeyhimi. Using conductive inks and non-woven textiles for wearable computing. In *Proceedings of the 2005 Textile Institute Worlsd Conference*, Raleigh, NC, March 2005.

[15] J. Koenderink. The structure of images. *Biol. Cyber.*, 50:363–370, 1984.

[16] T.C. Lacalli and L.G. Harrison. Turing's conditions and the analysis of morphogenetic models. *Jnl of Theoretical Biology*, 76:419–436, 1979.

[17] PK Maini and HG Othmer. *Mathematical Models for Biological Pattern Formation*. Springer-Verlag, Berlin, 2001.

[18] H. Meinhardt. *Models of Biological Pattern Formation*. Academic Press, London, 1982.

[19] C.R. Merritt, B. Karaguzel, T.H. Kang, J. Wilson, P. Franzon, H.T. Nagle, B. Pourdeyhimi, and E. Grant. Electrical characterization of transmission lines on specific nonwoven textile substrates. In *Proceedings of the 2005 Textile Institute Worlsd Conference*, Raleigh, NC, March 2005.

[20] J. Murray. *Mathematical Biology*. Springer-Verlag, Berlin, 1993.

[21] J. Murray. *Mathematical Biology*. Springer-Verlag, Berlin, 1993.

[22] R Nagpal. Programmable pattern-formation and scale-independence. In *Proc International Confernce on Complex Systems (ICCS)*, Nashua, NH, June 2002.

[23] Gregoire Nicolis and Ilya Prigogine. *Exploring complexity: an Introduction*. W.H. Freeman and Co, New York, NY, 1989.

[24] P. Perona, T. Shiota, and J. Malik. Anisotropic diffusion. In B. Romeny, editor, *Geometry-Driven Diffusion in Computer Vision*. Kluwer, 1994.

[25] Ilya Prigogine. *Thermodynamics of Irreversible Processes*. Interscience Publishers, New York, NY, 1968.

[26] Ilya Prigogine. *From Being to Becoming: Time and Complexity in the Physical Sciences*. W.H. Freeman and Co, San Francisco, CA, 1980.

[27] J. A. Sethian. *Level Set Methods*. Cambridge University Press, New York, 1996.

[28] Alan Turing. The chemical basis of morphogenesis. *Philosophical Transactions of the Royal Society of London*, B237:37–72, 1952.

[29] Song Chun Zhu and David Mumford. Prior learning and gibbs reaction-diffusion. *IEEE-T on Pattern Analysis and Machine Intelligence*, 19(11):1236–1250, 1997.