Amorphous Predictive Nets

Regina Estkowski, Michael Howard, David Payton,
HRL Laboratories, LLC
Malibu, CA 90265
{estkowski, howard, payton}@hrl.com

1. Introduction

This paper describes our approaches for coordinating the actions of extremely large numbers of distributed, loosely connected, embedded computing elements. In such networks, centralized control and information processing is impractical. If control and processing can be decentralized, the communications bottleneck is removed and the system becomes more robust. Since conventional computing paradigms provide limited insight into such decentralized control, we look to biology for inspiration.

Due to progress in the miniaturization of sensors and computing elements and in the development of necessary power sources, large arrays of networked wireless sensor elements may soon be realizable. The challenge is to develop software that enables such amorphous arrays to self-organize in ways that enable the sensing capabilities of the whole to exceed that of any individual sensor.

Our goal is to devise local rules of interaction that cause useful computational structures to emerge out of an array of distributed sensor nodes. These distributed logical structures appear in the form of local differences in sensor node state. These local state differences serve to form distributed circuits among nodes, allowing groups of nodes to perform cooperative sensing and computing functions that are not possible at any single node. Further, since the local differences emerge and are not pre-programmed, there is never a need to assign specific functions to specific nodes.

In this paper we describe two methods, each using only local interactions between nodes to detect the presence and heading of some local transient property of the environment (e.g., presence of a warm body). These methods provide a purely distributed means of computing the direction and likely destination of a sensed movement, with no need for centralized data analysis or explicit data fusion. Such a prediction could activate sensors ahead of the movement of the sensed object, turning on more expensive sensing functions that are normally dormant to save power. An active minefield could use the techniques to attract mines to the most likely avenue of approach. Streetlights could be turned on ahead of cars on a road less traveled.

2. Related Research

Our research focus is on future applications of sensor networks wherein the sensor nodes themselves will be extremely small, cheap, and simple, and will be deployed somewhat haphazardly or randomly. Examples of such networks can be found at UC Berkley, where the goal is to create sensor nodes the size of dust particles that can be released in large quantities from the air. Such sensor networks have very limited computational capabilities, and are unlikely to have sophisticated on-board position location capabilities such as GPS.

A number of other methods for monitoring object presence and movement have been developed, but many of these methods are limited in scope and related to a narrow application, or require sophisticated sensors and centralized processing. The most relevant is the work being done at MIT on amorphous computing and the work being done at USC on directed diffusion in sensor networks, although neither of the two encompasses our system.

MIT is making progress on pattern formation in amorphous networks in the context of Paintable Computing and "shape formation" via the use of origami mathematics [Nagpal 2001]. These patterns are not used in the context of object tracking. MIT uses some of the same basic primitives we use in pattern formation, but the overall methods are different.

The USC work [Intanagonwiwat 2000] uses directed diffusion for object tracking, but it assumes that each sensor knows its location, and can inform a user of an intruder's position via directed diffusion. We make no assumptions about node location.

Our use of a virtual pattern sets up a virtual heterogeneous network in which different sensors have different functions depending upon their position in the pattern. We are aware of no previous attempt to use the sensor distribution and network structure itself to track objects and predict movement direction.

3. Pheromone Messaging

We use a diffusion-based messaging paradigm called *virtual pheromones* [Payton, et al 2000, 2002]. Virtual pheromones provide a simple messaging scheme that establishes a gradient among a large number of distributed, locally communicating nodes. In earlier work, we have used a custom-made IR transceiver unit, as shown in Figure 1, to transmit and receive virtual pheromones in 8 distinct directions. However, in most of the sensor node applications described in this paper, we envision using RF communications between nodes, and therefore constrain ourselves to omni-directional transmission and reception of virtual pheromone signals. A virtual pheromone is encoded as a single modulated message packet consisting of a type field, a hop-count field, and a data field. The type is an integer that identifies a unique pheromone class for the message. The data field may be used to optionally transmit a few bytes of data. The hop-count is used to establish how far a pheromone may travel from its originating source and to establish simple pheromone gradients. The originating node sets the hop-count to an integral number of times the message is to be relayed, and sends the message to its local neighbors. Upon receipt, the hop-count field is decremented and the message retransmitted without any need for acknowledgement. If a node receives the same type of pheromone from multiple sources, only the message with the highest hop-count value is selected for re-transmission. This results in a uniform gradient leading away from the source. These rules for message propagation provide a distributed version of the wavefront propagation method used in Dijkstra's shortest-path algorithm [Dijkstra 1959].

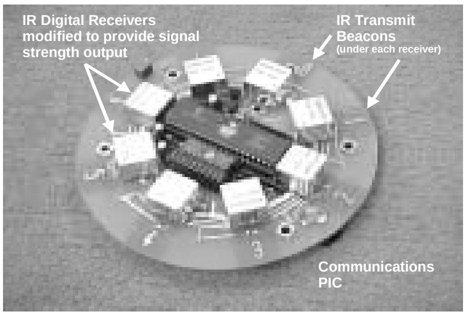

Figure 1: Transceiver for virtual pheromones

4. Motion Prediction Methods

In the following, we describe two different motion prediction methods. In the first method, a temporal differencing technique is used to obtain very coarse motion detection and prediction for objects moving across a sensor array. In the second method, patterns are formed within the sensor array to differentiate nodes. Interactions between such differentiated nodes produce more precise motion detection and prediction.

Both methods are applicable to a distributed network of locally connected sensor nodes, commonly called a sensor network. Each sensor measures some local transient property of the environment (e.g., presence of a warm body or an object), at a limited range, so the optimal distribution of sensor nodes would be at inter-node distances just less than double their maximum sensing range. In each of the methods below, sensors do not directly measure velocity - only the presence or absence of the object. The network connections are considered to be wireless, and nodes have no knowledge of their neighbors or even of their own location. All communication is by means of unreliable short range broadcast.

If there is a way to determine distance between nodes, e.g. signal strength, it is possible to select a more uniformly distributed subset of nodes in order to obtain a better gradient. One or more nodes emit a special distribution pheromone message, containing a minimum and maximum range parameter. Receiving nodes that are within the specified range will join the active subset of nodes. Those outside the specified limits become inactive. Only active nodes relay the distribution pheromone. Sometimes this type of pre-conditioning results in better pattern formation in the subsequent steps.

4.1. A Temporal Differencing Technique

The first technique creates a very simple pattern in the sensor array when each node compares its activation level from one moment to the next. In a two-state implementation, we will label the states ON and OFF. An ON node is either directly sensing the object or is experiencing an increasing activation level indicating it is potentially in the path of the object's motion. All other nodes are OFF.

Any node that senses the object turns ON and originates a pheromone with a high activation. It does not need to check for incoming messages. The activation message is diffused throughout the sensor node population creating a complete gradient of activation, using the algorithm described in Section 3. We assume that message diffusion is much faster than the movement of the objects the nodes sense. As an object moves, different nodes sense it and take over the job of initiating the pheromone gradient.

Nodes that do not directly sense the object base their state on the difference between the activation level of the last message they received and the level of the current message. As the object moves, different nodes sense it and become pheromone initiators, and others that no longer sense it stop sending their pheromones. This causes the gradient to slide across the [illegible] ion of each node from one moment to the next is positive in front of the movement, zero to the sides, and negative to the rear. In Figure 2, nodes that sense an object at geographic position A at time 1 create a gradient field. At time 2, other nodes at point B sense the object, resulting in the second gradient. When each node receives the gradient message at time 2, it subtracts the activation level from time 1 from the new time 2 activation, and gets a value that is either positive, negative or zero.

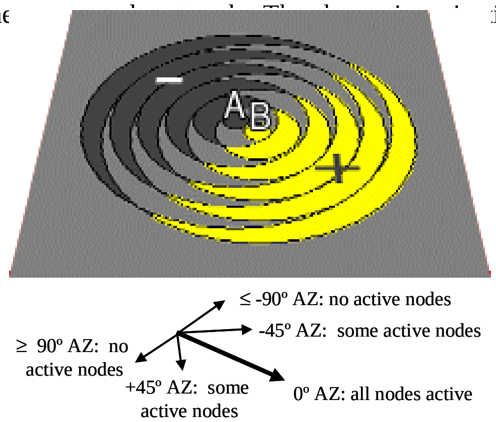

Figure 2. Temporal difference pattern in gradient after object moves from A to B, and % of active nodes as a function of azimuth

Therefore, the simple activation rule for nodes that do not directly sense the object is to turn ON if the temporal difference is positive, and turn OFF otherwise. The resulting activation pattern is useful for waking up nodes in a sensor net to track an object that may not continue to travel on a linear path. Imagine that the background of Figure 2 is a large number of randomly placed nodes. If the azimuth vectors are superimposed on the temporal difference pattern, it can be seen that all nodes in the center of the path of the object (at 0º AZ) will be turned on, and the percentage of activated nodes will drop as azimuth increases. In other words, the percent of activated nodes in a particular direction roughly corresponds to the likelihood that the object will move in that direction.

It is difficult to extend the object's motion vector to nodes ahead of the object in a more focused way without requiring directional messaging. One strategy using directional messaging would be for each node to keep track of the gradient vector as the object moves toward it. If any node subsequently senses the object directly, it can conclude the object came from the direction of steepest ascent, which is still in memory. The sensing node sends activation pheromone as before, but the message that goes in the direction of the object's movement is annotated to tell nodes that receive it to turn ON. Nodes that receive the annotated message forward it in the same direction, while again sending unannotated messages to all other neighbors so they can track changes in the gradient over time.

4.2. Focused Predictions Using Pattern Formation

In the second technique, sensor nodes are differentiated into parallel spatial bands to provide motion detection along different axes. In this method, a virtual pattern of bands emerges through specially designed local interactions between nearby nodes. This results in a pattern state within individual nodes that either sensitizes or desensitizes them to particular activation/inhibition signals from neighboring nodes. Activation/inhibition rules are designed such that messages signaling the presence of an object are inhibited along bands of the same type, but are propagated into bands of a different type as shown in Figure 3. This, in effect, leads to a form of moving edge detection for objects moving across the sensor array from one spatial band to another.

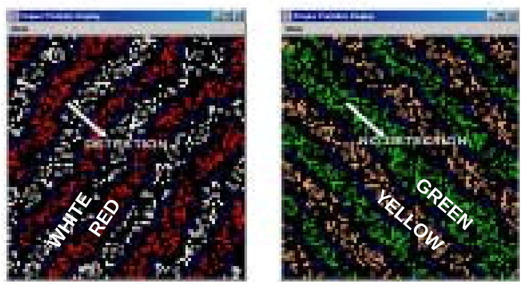

Figure 3. Orthogonal sets of parallel bands are used to detect movement along specific directions.

Band Creation

Bands are generated by first choosing two "anchor nodes" lying on opposite ends of a diameter of the sensor net. These anchor nodes determine the orientation of the initial set of bands. Starting with an identical pattern state in all nodes, one of the two anchor nodes initiates a pheromone signal that creates a gradient throughout the entire network. When this signal reaches the second anchor node, the recipient issues a second pheromone signal. This second signal propagates using rules comparable to directed diffusion [Intanagonwiwat 2000], wherein signals only advance along the axis of steepest ascent of the first gradient. We call this a "white" pheromone signal because all nodes that receive it will switch to a white pattern state, thereby forming a white band as shown in the leftmost frame in Figure 4.

After nodes have switched to a white pattern state, they transmit a limited-range "red" pheromone signal. This causes all nodes that receive it that are not already in the white pattern state to switch to the red pattern state, as shown in the middle frame of Figure 4. Likewise, nodes in the red pattern state transmit a white pheromone signal that switches all non-red nodes within range to the white pattern state. The net result is a sequence of parallel bands as shown in the rightmost frame of Figure 4.

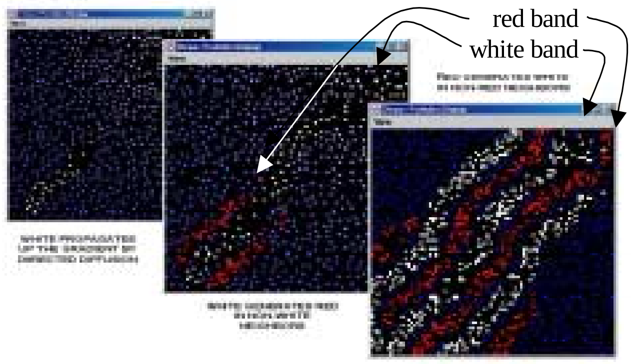

Figure 4. Formation of initial set of parallel bands.

A set of bands orthogonal to the first is formed using gradients initiated from both of the anchor nodes. Midway between the two anchor nodes, these gradients meet with equal hop counts, and the nodes in that region switch to a green pattern state. Just as before, nodes in the green pattern state send a pheromone that triggers neighboring nodes to switch to the yellow pattern state. Likewise, nodes in the yellow pattern state send messages to switch neighbors to the green pattern state. This results in another set of parallel yellow / green stripes that is orthogonal to the original red / white stripes. Because the green and yellow pattern states are independent of the red and white pattern states, each node can be a member of both the green/yellow and the red/white stripe patterns. The same process could be used to create a number of different band orientations to achieve any desired resolution of motion sensitivity.

Detection and Prediction

Once stripes have formed, the resulting pattern states can aid in detection of a moving object. When a sensor detects an object, it sends out a short-range priming pheromone labeled with its pattern state as shown in Figure 5. Nodes that receive

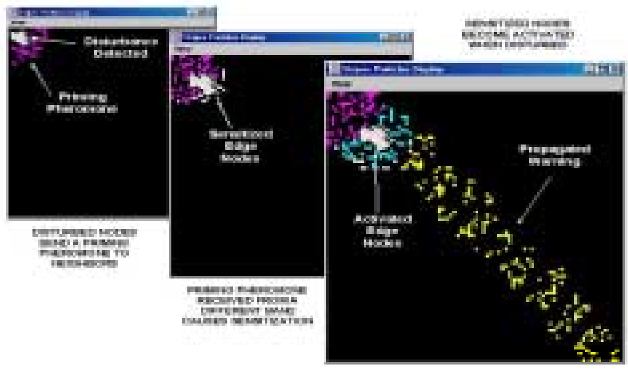

Figure 5. A warning signal (yellow) propagates through the network after persistent motion is detected.

this message and lie in a different stripe band become sensitized for a short time. If a sensitized node detects an object, it sends out a warning to be propagated. This warning message is accepted and propagated only by nodes that lie on the same band as both the priming sensor and the warning sensor. The initial warning message is weak and does not travel far. If the detected object continues to move in the same direction, the warning message is reinforced at receiving nodes and propagates further. If motion is no longer detected along the given direction, the warning pheromone at these nodes decays and the nodes revert to their original state. This provides a simple form of motion prediction whereby nodes far from the moving object register a warning if the object continues to move toward them.

5. Data Extraction

These methods provide a purely distributed means of computing the direction and likely destination of a sensed movement, with no need for centralized data analysis or explicit sensor data fusion. In the preceding discussion, we have used the results of the distributed computation only to change node state. However, it may be desirable to view the states of nodes, e.g., to follow the activation path to the sensed object. We would like the sensor array to act as a distributed display embedded in the environment. In effect, each node becomes a pixel, or an annotation, on the immediate environment. The node's position within the environment provides context to interpret the meaning of the transmitted information.

The easiest solution would be to put colored blinking lights on each node; but this limits the type of data that can be represented. Our approach, called the World-Embedded Display [Payton et al, 2002], is to visualize the distributed data in the collection of nodes directly, *in situ*. The user wears an augmented reality (AR) head-mounted display, shown in Figure 7. Each node transmits a character of data via infrared, and a camera mounted on the head mounted display with an IR filter reads the data. The system converts the character into a symbol that is drawn on the video shown in the display. Figure 7 illustrates the effect: two of our mobile robots as they

Figure 7. AR head-mounted display has pencil-cam to image IR data transmitted from individual sensor nodes. Gradient superimposed on nodes.

are viewed in the AR display, with the pheromone gradient arrows floating above them.

6. Conclusion

We have described two purely distributed methods for computing the direction and likely destination of an object of interest, with no need for centralized data analysis or explicit sensor data fusion. These algorithms apply to spatially distributed collections of simple sensing nodes with only local communication and rudimentary processing capabilities.

The temporal differencing scheme is very simple and has the nice property of waking up the most nodes in directions that are likeliest to be in the object's path. This results in a sort of heuristic search pattern for the future movements of the object, which is desirable in many applications. The prediction can be somewhat more focused using a more constrained type of messaging.

The second technique produces a tightly focused motion prediction without requiring directed messaging. It detects object movement that crosses an oriented pattern, and uses the pattern to activate nodes in the path of motion. The pattern must be oriented correctly for motion to be detected; it is possible for several different orientations to coexist in the network.

This paper is concerned with motion extrapolation, not the identification problem. However, without the ability to uniquely identify objects, these approaches can be fooled much like the eye is fooled into perceiving motion with a movie marquee. If sensors can accurately identify objects, pheromones could be labeled with a feature ID, making node state ID-specific. These issues are left for future work.

References

Dijkstra, E.W. 1959, A Note on Two Problems in Connection with Graph Theory, *Numerische Mathematik*, **1**:269-271.

Intanagonwiwat, C., Govindan, R., Estrin, D., 2000, Directed Diffusion: A Scalable and Robust Communication Paradigm for Sensor Networks, *MobiCOM 2000*

Nagpal, N., 2001, Programmable Self-Assembly: Constructing Global Shape using Biologically-inspired Local Interactions and Origami Mathematics, PhD Thesis, MIT Dept. of EE and CS

Payton, Daily, Estkowski, Howard, Lee, 2001, Pheromone Robots, *Autonomous Robots,* Kluwer Academic Publishers, Boston, MA, Vol. 11, No. 3.

Payton, Estkowski, Howard, 2002, Progress in Pheromone Robotics, *Proceedings 7th International Conference on Intelligent Autonomous Systems,* Marina del Rey, CA, 25-27.