Complex Systems **10** (1996) 73–90

An Immune Network Approach to Sensor-based Diagnosis by Self-organization

Yoshiteru Ishida*
Graduate School of Information Science,
Nara Institute of Science and Technology, Ikoma,
Nara, 630-01 Japan

Abstract. Self-organizing diagnosis is studied by applying the dynamic propagation of active states derived from the concept of an immune network. The proposed model is a mutual vote network which is a modification of the majority net. The model implements network-level recognition by connecting information from local recognition agents by dynamical evaluation chains. The model has been further elaborated to address the engineering concerns of identifying not only sensor faults but also process faults. The sensor faults are identified by evaluating the reliability of data from a sensor. Process faults are identified by evaluating constraints that must be satisfied among these data. We demonstrate that the sensor network works against both sensor faults and process faults by an illustrative example.

1. Introduction

A massively parallel and distributed model has been proposed and studied based on a neural network metaphor that is elaborated for pattern recognition as found in backpropagation [1]. In contrast, several researchers proposed dynamic network models to study the information processing character of biological systems including immune systems [2–6, 8–12]. Some of them (e.g., [13–17]) try to extract the information processing mechanism from the immune systems for the purpose of developing information processing systems. Similar to this line, we also have proposed a dynamic network based on the immune network metaphor [14, 18–20]. The motivation of this work is to elaborate the immune network metaphor as a model for dynamic networks. More specifically, the objective of this paper is to demonstrate diagnostic significance, or recognition significance in general, of the network model by applying it to self-organizing diagnosis.

*Electronic mail address: `ishdia@is.aist-nara.ac.jp`.

Several dynamical networks have been examined for studying the information processing character of immune systems or biological systems in a more general context. The majority network, where the state of each node becomes the state of a majority of the nodes adjacent to it, has been studied extensively. The significance of the majority network is that the period of the state is at most two no matter what the structure of the network and no matter how many finite states each node may have [21]. The majority network, with structural symmetry such that each node arranged on a ring is connected to both r left nodes and r right nodes, has been studied to understand *one to many* and *many to one* relations found in a cytokine network [4, 6]. The majority network model with a cylindrical structure, where rings of majority networks are layered, has also been studied for the analysis of information transformation characteristics from genotype to phenotype [5].

A cellular automaton uses any boolean function, instead of majority rule, to determine the state of nodes. However, the boolean function is fixed for all nodes, and the structure of the network is symmetrical as in the Agur's network described above. Boolean networks, which may have any boolean function for any nodes, with the network being any structure, have been studied extensively with regards to their statistical nature using the ensemble model ([11] and references therein). Our model, proposed in section 2, can be considered a modification of the majority network where each vote is weighted by reliability.

Studies to establish information models by investigating the features of information processing done by them, and by reflecting these features into the models were performed [2, 4, 5, 10, 16]. The features of information processing done by immune systems may be summed up with the following: memory, recognition, learning, and diversity. As many other researchers do, we view the immune network as a complex network where not only dynamic interaction among the agents but also activation of agents, inhibition of interactions, and self-reproduction with mutation of agents are controlled. An adaptive mechanism of immune systems may be formalized as dynamic interaction among agents as follows.

1. Each agent has not only information but a recognizing mechanism also.
2. An agent activated by an encounter with the antigen will reproduce its clone in order to enhance ability for elimination of the antigen.
3. The reproduction above will be performed with mutation to increase affinity with the antigen.

We share this view of dynamic interaction with many other researchers, and applied it to adaptive diagnosis. However, in the sensor network application explored in this paper, we do not adopt the adaptive change of the population of a specific agent as mentioned in 2 and 3. An immune algorithm which uses 2, 3, and diversity generation by genetic recombination has been proposed elsewhere [22]. It may be worth mentioning that there are two basic approaches for implementing these features of immune systems:

top-down approach (or network-based approach) where network structure is determined by reflecting the structure of the problem as discussed here; and bottom-up approach (or agent-based approach) where a population of agents and communication among them are organized by reflecting interactions with the environment.

In this paper, rather than building a detailed mathematical model of immune systems, an attempt has been made to explore the dynamic network based on the immune network metaphor, as done in neural network models. It should be emphasized that the model explored in this paper may not correctly reflect real immune systems. Immune systems could be completely disregarded in this paper, since our model is just inspired by the concept of idiotypic networks as has been elaborated for engineering applications as discussed briefly in section 3 and in [18–20] in more detail.

Immune network (idiotypic network) theory was proposed in [23]. Although immune network theory is disappointing for use in immunology (e.g., [24]) and is not much mentioned recently, it does provide a network view with B-cells that are mutually and dynamically connected by antigen–antibody reaction. After the proposal of immune network theory in [23], the approach was used extensively to characterize immune response by dynamical models. A few examples of the many researches in this line are [7–9, 12, 25–29]. These dynamical models essentially describe population dynamics of antigen, antibody, and specific immune related cells in a manner similar to models describing population dynamics of species in mathematical ecology. Features of the immune network that we have tried to use may be summarized as follows.

- Recognition is done by distributed agents which dynamically interact with each other in parallel. The agents carry redundant information. (Sensor values carry the redundant information in our application.)
- Each agent reacts based only on its own knowledge, that is, each sensor refers to its own state to judge its consistency with others.
- Memory is realized as stable equilibrium points of the dynamical network. Recognition of the network is done by changing the state of the network from one stable equilibrium to another by disturbances on the network. (Diagnosis is done by obtaining the stable equilibrium points where a change in the consistency pattern is considered to be a disturbance.)

In this paper, we implemented these features in our application of a sensor network. However, we do not pursue the parallelism between our model and immune systems further.

The problem of integrating distributed sensors with fault tolerance concerns [30] and different types of sensors for biological adaptation [31] have been studied extensively with the progress of device technology such as intelligent sensors and their integrating techniques. Our approach aims at supporting self-organization of the sensor networks by enhancing the processing done in autonomous sensors using redundant information shared by many sensors. The sensor network we propose is designed to work especially in the dynamic and noisy environments found in industrial processing plants. The motivation for developing the sensor network comes from a practical reason of using redundant information in the sensors of instrumentation systems. As a result, it is possible to distribute the knowledge of a normally functioning process into all the sensors and the relationship among sensors. Thus, the sensor network is robust in the sense that it would work even with faulty sensors. It is also flexible in the sense that it can represent knowledge of relative relationship among sensor values as opposed to representing reference values specified by permissible maximum and minimum.

Section 2 presents the basic idea of a mutual vote network and discusses recognition significance by dynamical propagation of weighted votes. Simulations are also conducted for several types of propagation. Section 3 presents an application of the mutual vote network to a sensor network. Simulations are conducted by examples to explain the application.

2. Mutual voting by a continuous dynamic network

2.1 Mutual vote network and majority network

We consider a mutual vote network similar to the majority network mentioned in section 1. We modify the network step by step, to demonstrate significant features of our dynamic network. In the mutual vote network, each agent votes the other agents; whether or not they are reliable. A state variable (r_i and its normalization R_i) indicating the reliability of agents is assigned to each agent. An important difference between this mutual vote network and the majority network is that the two states (active or inactive) are asymmetrical in determination of the next state, while states in a majority network are symmetrical in the sense that they can be exchanged without changing behavior. In other words, states in the majority network are a matter of labeling. In a mutual vote network, the effect from active states and that from inactive states are qualitatively different; only votes from active states are considered while those from inactive states will be neglected in determining the state of agents.

Figure 1 shows an example of the evaluation chain of mutual voting. The $+/-$ pattern associated with the evaluation arc shows a case when agents 4 and 5 are unreliable. A positive arc from agent i to agent j indicates that agent i voted positively for agent j (i.e., considered reliable), and a negative arc negatively (i.e., considered unreliable). Formally, evaluation results are assumed to give the following pattern. (Similar assumptions have been made in self-diagnosable models for fault tolerance, e.g., [32].)

$$T_{ij} = \begin{cases} -1 & \text{if evaluating agent } i \text{ is reliable and} \\ & \quad \text{evaluated agent } j \text{ is unreliable} \\ 1 & \text{if both agents } u_i \text{ and } j \text{ are reliable} \\ -1/1 & \text{if evaluating agent } i \text{ is unreliable} \\ 0 & \text{if there is no evaluation from agent } i \text{ to agent } j. \end{cases}$$

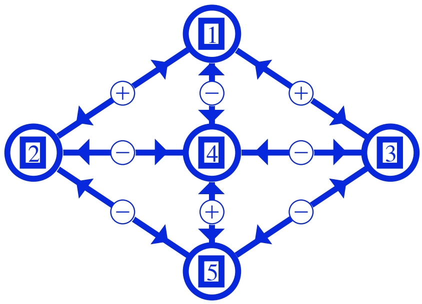

Figure 1: An example evaluation chain of mutual voting.

Simple voting at each agent does not work, since three agents (2, 3, and 5) are all evaluated as unreliable by two other agents and hence cannot be ranked in terms of reliability. Since an unreliable agent may give unreliable results, these votes should be weighted. Next, let us introduce the binary weight for each agent: 0 (inactive or unreliable) when the sum of votes for the agent is negative, and 1 (active or reliable) when the sum of votes for the agent is zero or positive. Starting with all agents active, evaluating the weight would synchronously result in the following sequence of state vector $(R_1 R_2 R_3 R_4 R_5)$

$$\begin{array}{l} (1 \quad 1 \quad 1 \quad 1 \quad 1) \\ (1 \quad 0 \quad 0 \quad 0 \quad 0) \\ (1 \quad 1 \quad 1 \quad 0 \quad 0). \end{array}$$

Thus, weighting the vote and propagating the information identifies the unreliable agents correctly. Let us map the above discrete model to the continuous dynamical model. A continuous dynamic network is constructed by associating the time derivative of the state variable with state variables of other agents connected by the evaluation chain. A possible association corresponding to the evaluation chain of Figure 1 would be the following continuous dynamic network:

$$\begin{aligned} dr_1(t)/dt &= R_2(t) + R_3(t) - R_4(t) \\ dr_2(t)/dt &= R_1(t) - R_4(t) - R_5(t) \\ dr_3(t)/dt &= R_1(t) - R_4(t) - R_5(t) \\ dr_4(t)/dt &= -R_1(t) - R_2(t) - R_3(t) + R_5(t) \\ dr_5(t)/dt &= -R_2(t) - R_3(t) + R_4(t) \end{aligned}$$

where reliability $R_i(t) \in [0, 1]$ is a normalization of $r_i(t) \in (-\infty, \infty)$ (we use a sigmoid function for this normalization as shown later).

Considering not only the effect from evaluating agents, but also that from evaluated agents leads to the following dynamic network:

$$dr_i(t)/dt = \sum_j T_{ji} R_j + \sum_j T_{ij} R_j - \frac{1}{2} \sum_{j \in \{k : T_{ik} \neq 0\}} (T_{ij} + 1) \qquad (1)$$

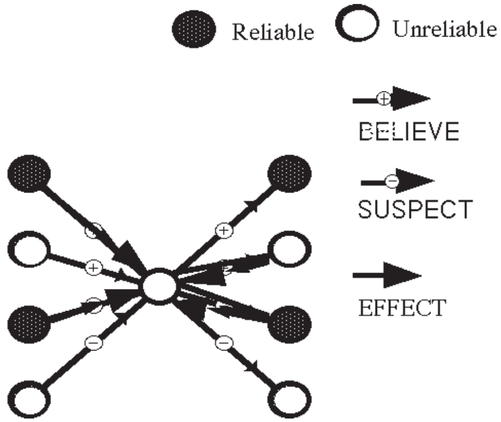

Figure 2: Effect of reliability evaluation.

where

$$R_i(t) = \frac{1}{1 + \exp(-r_i(t))}.$$

We call this model the *black and white model*, meaning that the network tries to separate an unreliable agent clearly from a reliable agent; that is, the reliability of an agent tends to be 1 or 0, not an intermediate value as shown later by simulation. In [18] we proposed several different variants of this dynamic network such as the *skeptical model* and the *gray model* for different engineering needs. In this black and white model, it should be noted that the reliability of one agent is evaluated not only from the opinions of other agents evaluating the agent, but from the opinions of what the agent said to the other evaluated agents. The former corresponds to the first term of the right-hand side of equation (1) and the latter to those of the second and third term. We call the latter *reflection effect*. The reflection effect is somewhat similar to the situation that if you criticize a highly respected person, it affects your own credit, not theirs. Figure 2 illustrates the effect from evaluating agents and that from evaluated agents.

Let $J_i(T, R)$ denote the right-hand side of equation (1). $J_i(R, T)$ will be positive (negative) if $R_i = 0(1)$ makes an inconsistency with the data T_{ij} and R_j. The dynamical model will change to the direction of canceling the inconsistency, that is, if $J_i(R, T)$ is positive (negative), then r_i hence R_i will increase. There is a *competitive interaction* among agents. When $T_{ij} = -1$ then, $R_i = 1$ inhibits R_j from being 1, and *vice versa*. When $T_{ij} = 1$, however, the interaction is not symmetric; $R_i = 1$ activates R_j to be 1, while $R_j = 0$ inactivates R_i to be 0.

Equation (1) is known to converge by the following Liapunov function [33] in the same manner as that of the Hopfield Net Algorithm [34]:

$$V(t) = -\sum_{i=1}^{n} \int_{0}^{R_i} \left\{ -\frac{1}{2} \sum_{j \in \{k: T_{ik} \neq 0\}} (T_{ij} + 1) \right\} R_i'(y_i) dy_i - 1/2 \sum_{i,j=1}^{n} T_{ij}^* R_i R_j$$

where

$$dV/dt = -\sum_{i=1}^{n} R_i'(r_i) \left\{ -\frac{1}{2} \sum_{j \in \{k: T_{ik} \neq 0\}} (T_{ij} + 1) + \sum_{j=1}^{n} T_{ij}^* R_j \right\}^2 \leq 0$$

and $T_{ij}^* = T_{ij} + T_{ji}$.

Since there are several local minimal points in the energy function, it may lead to a wrong diagnosis if starting from an inappropriate assignment of initial reliabilities. Starting with all the agents reliable may be a good strategy when a relatively small number of unreliable agents exist. In diagnosis, minimal diagnosis is often favored, where the number of $R_i = 0$ is minimal among consistent diagnoses. In order to make such a diagnosis, the energy function must be modified by adding the energy E_2:

$$E = E_1 + E_2$$

where E_1 is the same energy function as $V(t)$ defined previously, and $E_2 = -\alpha \sum_{i=1}^{x} nR_i \ \ (\alpha > 0)$. E_1 is added for making the diagnosis consistent with the given evaluation pattern and E_2 is for making the number of unreliable agents as small as possible in the diagnosis. This new energy modifies the evaluation function J_i of equation (1) as follows:

$$J_i(T, R) + \alpha.$$

Then, the local minimal points are biased to those including more 1s. If both $R_i = 1$ and $R_i = 0$ are consistent diagnoses, then $R_i = 1$ is always selected in this modified model. The parameter α is chosen so that the minimum point of the total energy $E_1 + E_2$ corresponds to the correct diagnosis, however, the modified model may possibly be trapped in the minimal point, not the minimum one.

2.2 Mutual vote networks of several propagation types

We have studied the following dynamic network where we proposed several types of the form of the right-hand side function depending upon its engineering use:

$$dr_i(t)/dt = f(\{T_{ij}\}, \{R_i(t)\}). \tag{2}$$

The skeptical model has been developed [18] for the case when evaluations, even from reliable agents, may be unreliable. That is, the evaluation results are defined as follows:

$$T'_{ij} = \begin{cases} 1 & \text{if both agents } i \text{ and } j \text{ are reliable} \\ -1/1 & \text{if either agent } i \text{ or } j \text{ is unreliable} \\ 0 & \text{if there is no evaluation from agent } i \text{ to } j. \end{cases}$$

For this evaluation result, the skeptical model is formalized as follows:

$$dr_i(t)/dt = \sum_j T^+_{ji} R_j(t) \tag{3}$$

where

$$T^+_{ij} = \begin{cases} T'_{ij} + T'_{ji} - 2 & \text{if both evaluation from } i \text{ to } j \text{ and } j \text{ to } i \text{ exist} \\ T'_{ij} + T'_{ji} - 1 & \text{if one of the evaluations from } i \text{ to } j \text{ or } j \text{ to } i \text{ exists} \\ 0 & \text{if neither evaluation from } i \text{ to } j \text{ nor } j \text{ to } i \text{ exists.} \end{cases}$$

Further, (2) is modified to become the *gray model*, allowing ambiguous states of reliabilities rather than forcibly separating into the binary states reliable and unreliable. This comes from the engineering requirement that operators want to know the reliability as it is. The gray model is formalized as follows:

$$dr_i(t)/dt = \sum_j T^+_{ji} R_j(t) - r_i(t). \tag{4}$$

The second term of the right-hand side of the gray model is the inhibition term that keeps ambiguous states of reliabilities.

We now briefly describe a simulation of the typical behavior. Figures 3, 4, and 5 show the evolution of reliabilities of agents in the example shown in Figure 1 by the black and white, skeptical, and gray models, respectively. Agents 4 and 5 are unreliable agents. For each agent, its corresponding sensor value (upper part) and its reliability (lower part) are shown in these figures. All the sensor values $N1(t)$, $N2(t)$, $N3(t)$, $N4(t)$, and $N5(t)$ are assumed to produce 1.0, however, sensors corresponding to unreliable agents $N4(t)$ and $N5(t)$ produced 2.0 instead. It can be seen that both the black and white model and the skeptical model give a clear separation of reliable agents (whose reliabilities converged on 1) from unreliable agents (whose reliabilities converged on 0).

Since the skeptical model is more skeptical than the black and white model, even reliable agents 2 and 3 were dragged a little towards zero. The reliabilities of the reliable agents 1, 2, and 3 eventually converges on 1 in the black and white and skeptical models, while those in the gray model remain ambiguous states between 0 and 1.

3. A sensor network

In this section, the mutual vote network described is applied to a sensor network that reacts to on-line data from sensors. The sensor network not only self-eliminates sensor faults but also identifies process faults by monitoring consistency between sensor data and constraints of the target system. Only an illustrative example is presented to show the basic idea. Industrial applications and engineering concerns for the application are discussed elsewhere [18, 19, 22].

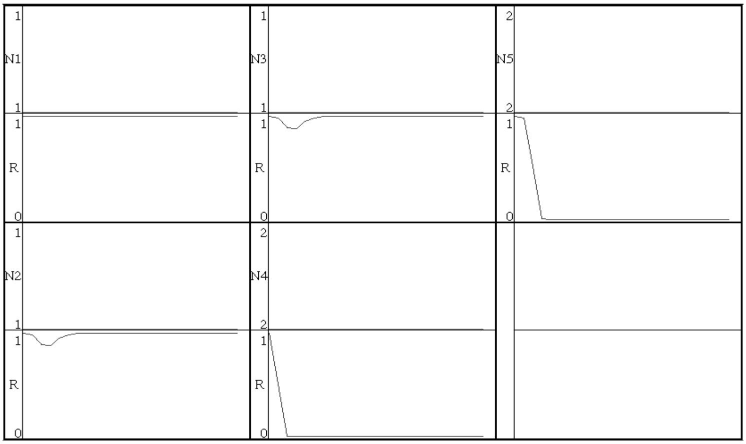

Figure 3: Simulation results of Figure 1 using the black and white model:

$$\begin{aligned}
dr_1(t)/dt &= 2R_2 + 2R_3 - 2R_4 - 3\\
dr_2(t)/dt &= 2R_1 - 2R_4 - 2R_5 - 1\\
dr_3(t)/dt &= 2R_1 - 2R_4 - 2R_5 - 1\\
dr_4(t)/dt &= -2R_1 - 2R_2 - 2R_3 + 2R_5 - 1\\
dr_5(t)/dt &= -2R_2 - 2R_3 + 2R_4 - 1.
\end{aligned}$$

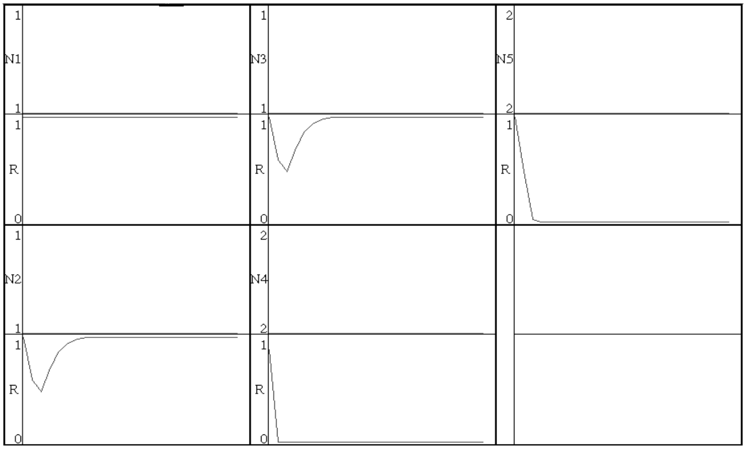

Figure 4: Simulation results of Figure 1 using the skeptical model:

$$\begin{aligned}
dr_1(t)/dt &= R_2 + R_3 - 3R_4\\
dr_2(t)/dt &= R_1 - 3R_4 - 3R_5\\
dr_3(t)/dt &= R_1 - 3R_4 - 3R_5\\
dr_4(t)/dt &= -3R_1 - 3R_2 - 3R_3 + R_5\\
dr_5(t)/dt &= -3R_2 - 3R_3 + R_4.
\end{aligned}$$

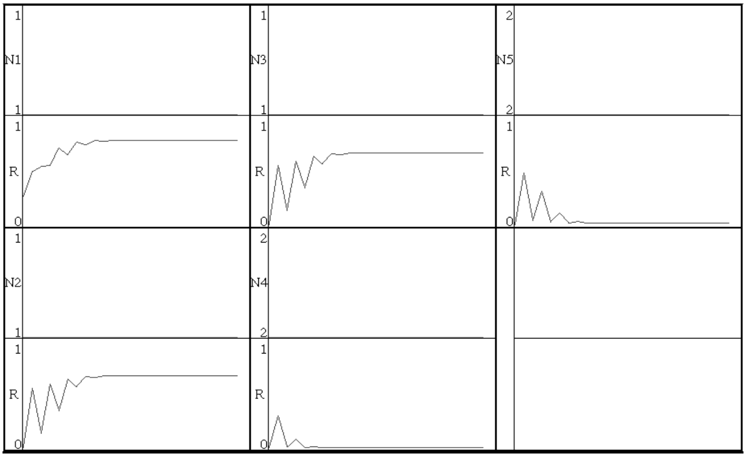

Figure 5: Simulation results of Figure 1 using the gray model:

$$\begin{aligned} dr_1(t)/dt &= R_1 + R_2 + R_3 - 3R_4 - r_1(t) \\ dr_2(t)/dt &= R_1 - 3R_4 - 3R_5 - r_2(t) \\ dr_3(t)/dt &= R_1 - 3R_4 - 3R_5 - r_3(t) \\ dr_4(t)/dt &= -3R_1 - 3R_2 - 3R_3 + R_5 - r_4(t) \\ dr_5(t)/dt &= -3R_2 - 3R_3 + R_4 - r_5(t). \end{aligned}$$

3.1 Sensor fault diagnosis by evaluating reliability of data from sensors

A sensor network may be formed when the sensors s_i and s_j are related by any constraint (such as equality, inequality, etc.) that restricts the interval J of value s_j corresponding to the interval I of value s_i. In process diagnosis, for example, temperature, pressure, and flow measured independently often have such interrelation, and hence interval correspondence between, for example, high temperature and high pressure, is attained. The following composes an agent i which will produce the evaluation result T_{ij} by the sensory data of s_i and s_j

$$T_{ij} = \begin{cases} 1 & s_j \in J \text{ when } s_i \in I \\ -1 & \text{otherwise.} \end{cases}$$

Consider an example of a heat exchanger of the condenser type. Two flows, that is, the flow of the shell side and the flow of the tube side, exchange heat. Steam enters from the inlet of the shell side, then it is condensed, and is finally cooled by the flow of the tube side. The flowing object of the tube side then becomes heated by the flow of the shell side. Now, we consider the inequalities among temperatures of these two flows:

1. $T_{hi} > T_{ho}$

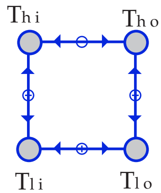

Figure 6: Diagram of a sensor network in a heat exchanger.

2. $T_{lo} > T_{li}$
3. $T_{hi} > T_{li}$
4. $T_{ho} > T_{lo}$

where

$$\begin{aligned} T_{hi} &= \text{temperature at inlet of shell side,} \\ T_{ho} &= \text{temperature at outlet of shell side,} \\ T_{li} &= \text{temperature at inlet of tube side, and} \\ T_{ho} &= \text{temperature at outlet of tube side.} \end{aligned}$$

Using these constraints among sensors, the sensor network will be constructed (see Figure 6). For example, when the relation 1 ($T_{hi} > T_{ho}$) does not hold, then either T_{hi} or T_{ho} can be unreliable. This means that the sensors measuring values T_{hi} and T_{ho} are evaluating each other through this relation.

We have applied this approach to industrial processing plants such as the firing section of a cement process with twenty thermometers and the blast furnace of a steel process with about five hundred thermometers [19]. The main characteristic of the mutual vote approach to process diagnosis is that it admits relative relation between process values, hence it does not suffer from the shifting of all the process values (which occurs depending on the load to the process or a change in the environment such as seasonal temperatures).

3.2 Process fault diagnosis by a sensor network

In order to extend the sensor network so that it can diagnose process faults as well, we have proposed two methods [20]: (1) simply preparing the agent corresponding to the process fault, and (2) introduction of a *virtual sensor* composed of multiple sensors. In this section, we present another natural extension based on the insight that the knowledge of a normal process is embedded in the constraints among sensors. When process faults occur, it amounts to a violation of the constraints. In fact, when a process fault occurs, reliability of many sensors related to the constraints become low simultaneously. Figure 7 illustrates the situation when a process fault corresponding to a violation of the constraint between s_i and s_j occurs.

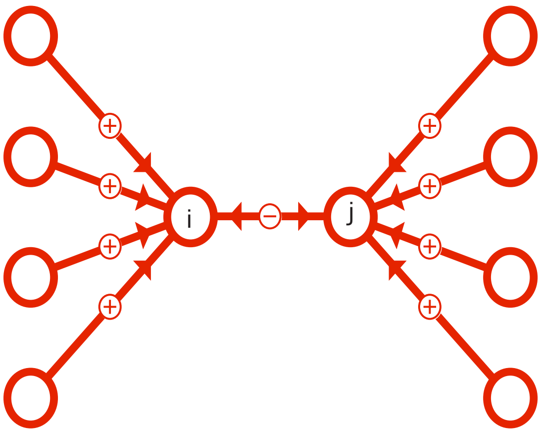

Figure 7: Situation when a process fault occurs.

Therefore, one natural way of detecting process faults by a sensor net is to introduce reliability for evaluating the relation. Let $R_{T_{ji}}$ denote the reliability of the evaluation T_{ji}. Then the gray model (equation (4)) becomes as follows:

$$dr_i(t)/dt = \sum_j T_{ji}^{+} R_j(t) R_{T_{ji}} - r_i(t) \quad (5)$$

$$dr_{T_{ji}}(t)/dt = T_{ji}^{+} R_j(t) R_i(t) - r_{T_{ji}}(t) \quad (6)$$

with

$$R_i(t) = \frac{1}{1+\exp(-r_i(t))}$$

$$R_{T_{ji}}(t) = \frac{1}{1+\exp(-r_{T_{ji}}(t))}.$$

Equation (5) is a modification naturally resulting from consideration of the effect of the reliability of the evaluation T_{ji}. The change rate of the agent i (i.e., $dr_i(t)/dt$) should reflect all the opinions of other agents weighted not only with the reliabilities of these agents but with those of their evaluation. Equation (6) comes from the fact that the evaluation relation is considered to be unreliable only when T_{ji}, $R_j(t)$, and $R_i(t)$ are contradictory; $T_{ji} = -1$, $R_j(t) = 1$, and $R_i(t) = 1$.

3.3 Sensor net for multiple evaluations

Constraints among process variables may be expressed by the vector equation: $\mathbf{f}(\mathbf{y}^*, t) = 0$ where $\mathbf{y}^*$ is the true value of process variables. The evaluation results for multiple evaluation can be defined as:

$$T_k \equiv \begin{cases} 1 & \text{if } |f_k(\mathbf{y}(t), t)| \leq \epsilon_k \\ -1 & \text{if } |f_k(\mathbf{y}(t), t)| > \epsilon_k \end{cases} \quad (7)$$

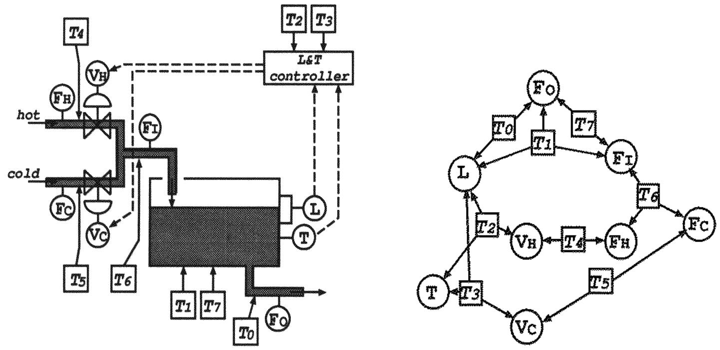

Figure 8: Tank with level and temperature controlled (left) and its sensor net (right).

where $\mathbf{y}$ is the measured value (i.e., sensor values) and ϵ_k are permissible residuals. Let $S = \{s_1, \ldots, s_n\}$ be a set of all the sensors and let S_k be the subset involved in the constraint f_k. When there are more than three variables in f_k, equations (5) and (6) can be modified as follows:

$$dr_i(t)/dt = \sum_{k, s_i \in S_k} \{T_k^+ R_{T_k}(t) \prod_{\substack{j, s_j \in S_k \\ j \neq i}} R_j(t)\} - r_i(t) \tag{8}$$

$$dr_{T_k}(t)/dt = \frac{1}{2}(T_k^+ - 1) \prod_{i, s_i \in S_k} R_i(t) - r_{T_{ji}}(t) \tag{9}$$

with

$$R_{T_k}(t) = \frac{2}{1 + \exp(-r_{T_k}(t))}$$

where $T_k^+ \equiv n_k(T_k - 1) + \lambda_k, n_k \equiv |S_k|$: the number of the sensors involved in the constraint f_k, $0 \leq \lambda_k \leq n_k$.

As an illustrative example, consider the process of keeping the level and temperature in a tank as shown in the left image of Figure 8. The right image of Figure 8 shows the sensor network of this process consisting of eight sensors and eight evaluations among the sensors. The model of this sensor network follows:

$$\begin{aligned}
dr_{F_0}(t)/dt &= T_0^+ R_L(t) R_{T_0}(t) + T_7^+ R_{F_I}(t) R_{T_7}(t) \\
&\quad + T_1^+ R_L(t) R_{F_I}(t) R_{T_1}(t) - r_{F_0}(t) \\
dr_{F_I}(t)/dt &= T_7^+ R_{F_0}(t) R_{T_7}(t) + T_1^+ R_L(t) R_{F_0}(t) R_{T_1}(t) \\
&\quad + T_6^+ R_{F_H}(t) R_{F_C}(t) R_{T_6}(t) - r_{F_I}(t) \\
dr_L(t)/dt &= T_0^+ R_{F_0}(t) R_{T_0}(t) + T_1^+ R_{F_I}(t) R_{F_0}(t) R_{T_1}(t) \\
&\quad + T_2^+ R_{V_H}(t) R_T(t) R_{T_2} + T_3^+ R_{V_C}(t) R_T(t) R_{T_3}(t) - r_L(t) \\
dr_{F_H}(t)/dt &= T_4^+ R_{V_H}(t) R_{T_4}(t) + T_6^+ R_{F_I}(t) R_{F_C}(t) R_{T_6}(t) - r_{F_H}(t) \\
dr_{F_C}(t)/dt &= T_5^+ R_{V_C}(t) R_{T_5}(t) + T_6^+ R_{F_I}(t) R_{F_H}(t) R_{T_6}(t) - r_{F_C}(t)
\end{aligned}$$

$$
\begin{aligned}
dr_{V_H}(t)/dt &= T_4^+ R_{F_H}(t) R_{T_4}(t) + T_2^+ R_L(t) R_T(t) R_{T_2}(t) - r_{V_H}(t) \\
dr_T(t)/dt &= T_2^+ R_{V_H}(t) R_L(t) R_{T_2} + T_3^+ R_{V_C}(t) R_L(t) R_{T_3}(t) - r_T(t) \\
dr_{V_C}(t)/dt &= T_5^+ R_{F_C}(t) R_{T_5}(t) + T_3^+ R_T(t) R_L(t) R_{T_3}(t) - r_{V_C}(t) \\
dr_{T_0}(t)/dt &= \frac{1}{2}(T_0^+ - 1) R_{F_0}(t) R_L(t) - r_{T_0}(t) \\
dr_{T_1}(t)/dt &= \frac{1}{2}(T_1^+ - 1) R_{F_0}(t) R_{F_I}(t) R_L(t) - r_{T_1}(t) \\
dr_{T_2}(t)/dt &= \frac{1}{2}(T_2^+ - 1) R_T(t) R_{V_H}(t) R_L(t) - r_{T_2}(t) \\
dr_{T_3}(t)/dt &= \frac{1}{2}(T_3^+ - 1) R_T(t) R_{V_C}(t) R_L(t) - r_{T_3}(t) \\
dr_{T_4}(t)/dt &= \frac{1}{2}(T_4^+ - 1) R_{F_H}(t) R_{V_H}(t) - r_{T_4}(t) \\
dr_{T_5}(t)/dt &= \frac{1}{2}(T_5^+ - 1) R_{F_C}(t) R_{V_C}(t) - r_{T_5}(t) \\
dr_{T_6}(t)/dt &= \frac{1}{2}(T_6^+ - 1) R_{F_H}(t) R_{F_C}(t) R_{F_I}(t) - r_{T_6}(t) \\
dr_{T_7}(t)/dt &= \frac{1}{2}(T_7^+ - 1) R_{F_C}(t) R_{F_I}(t) - r_{T_7}(t).
\end{aligned}
$$

In the simulation that follows, the parameter λ_k is set to be 2. Figure 9 shows the time evolution of the reliability of sensors and those of evaluation relations when there is a leakage in the pipe between F_I and F_H or between F_I and F_C. In this case, the constraint $F_I = F_H + F_C$ (i.e., the evaluation relation T_6) will be violated ($T_6 = -1$). It can be seen that although there is some decrease in the reliability of the sensor F_H, there is a significant decrease in the reliability of the evaluation relation T_6. Thus, the process fault will be known. Figure 10 shows the time evolution of the reliability of sensors and those of evaluation relations when the sensor F_H is unreliable ($T_6 = -1$, $T_4 = -1$). Again, although there is some decrease in the reliability of evaluations related to the unreliable sensor F_H, there is a significant decrease in the reliability of the unreliable sensor F_H. Thus, sensor fault can be concluded without difficulty.

4. Conclusion

We developed a dynamical network architecture based on the idea of a chain of active state propagation found in immune (idiotypic) networks. Fault recognition features of the mutual vote network are also discussed. The dynamic network model is elaborated as a sensor network that can diagnose not only sensor faults by evaluating reliability of data from sensors but also process faults by evaluating reliability of constraints among data. The sensor network dynamically reacts to the on-line data from sensors. It can self-identify the unreliable sensor and unreliable constraint by moving from one equilibrium to another, reacting to the change of the relationship among data.

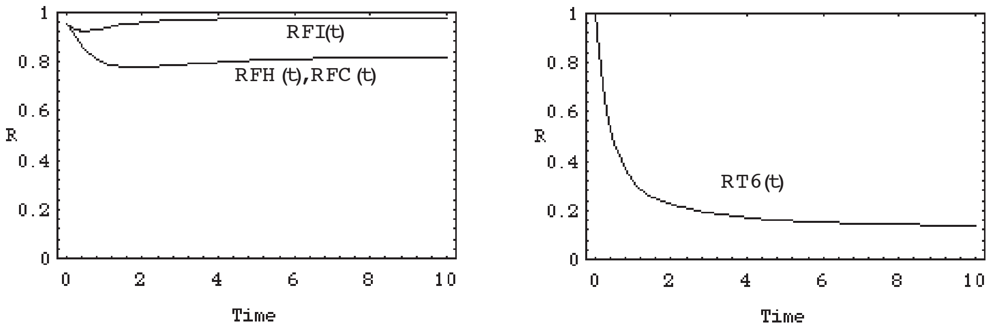

Figure 9: Time evolution of the reliability when there is a leakage in the pipe between F_I and F_H or between F_I and F_C.

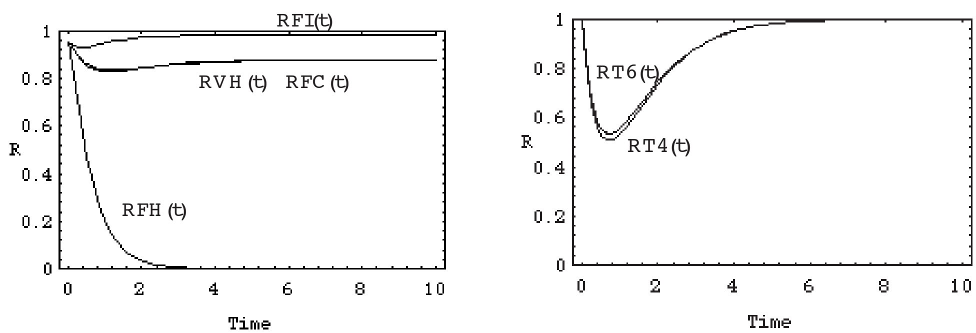

Figure 10: Time evolution of the reliability when the sensor F_H is unreliable.

Acknowledgments

This work has been supported in part by the Asahi Glass Foundation.

References

[1] D. E. Rummelhart and J. L. McClelland, *Parallel Distributed Processing* (MIT Press, 1986).

[2] M. Kaufman, J. Urbain, and R. Thomas, "Towards a Logical Analysis of the Immune Response," *J. Theor. Biol.* **114** (1985).

[3] G. W. Hoffman, "A Neural Network Model Based on the Analogy with Immune System," *J. Theor. Biol.*, **122** (1986) 33–67.

[4] Z. Agur, "Resilience and Variability in Pathogens and Hosts," *IMA Journal of Mathematics and Applied in Medicine & Biology*, **4** (1987) 295–307.

[5] Z. Agur, "Fixed Points of Majority Rule Cellular Automata with Application to Plasticity and Precision of the Immune System," *Complex Systems*, **2** (1988) 351–357.

[6] Z. Agur, "Optimizing the Immune Control of Parasitic Invasion," in *Theories of Immune Networks*, edited by H. Atlan and I. R. Cohen (Springer-Verlag, 1989, 107–116).

[7] M. Kaufman and J. Urbain, "A Model for the Dynamics of the Immune Response Based on Idiotypic Regulation," in *Cell to Cell Signaling: From Experiments to Theoretical Models*, edited by A. Goldberter (Academic Press, London, 1989, 303–314).

[8] R. J. de Boer "Information Processing in Immune Systems: Clonal Selection versus Idiotypic Network Models," in *Cell to Cell Signaling: From Experiments to Theoretical Models*, edited by A. Goldberter (Academic Press, London, 1989, 285–302).

[9] A. S. Perelson, "Immune Networks: A Topological View," in *Cell to Cell Signaling: From Experiments to Theoretical Models*, edited by A. Goldberter (Academic Press, London, 1989, 261–272).

[10] Z. Agur, G. Mazor, and I. Meilijson "Mimicking the Strategy of the Immune System: Insight Gained From Mathematics," in *Theoretical and Experimental Insight into Immunology*, edited by A. S. Perelson and G. Weisbuch (1992).

[11] S. A. Kauffman, *The Origin of Order* (Oxford University Press, 1993).

[12] F. J. Varela, A. Coutinho, and J. Stewart, "What is the Immne Network For?" in *Thinking about Biology*, edited by W. Stein and F. J. Varela (Addison Wesley, Reading, MA, 1993, 215–230).

[13] J. D. Farmer, N. H. Packard, and A. S. Perelson, "The Immune Systems, Adaptation, and Machine Learning," *Physica*, **22D** (1986) 187–204.

[14] Y. Ishida, "Fully Distributed Diagnosis by PDP Learning Algorithm: Towards Immune Network PDP Model," *Proc. of International Joint Conference on Neural Networks*, San Diego, (1990) 777–782.

[15] H. Bersisni and F. J. Varela, "The Immune Recruitment Mechanism: A Selective Evolutionary Strategy," *Proc. International Conference on Genetic Algorithms 91*, (1991).

[16] S. Forrest and A. S. Perelson, "Genetic Algorithm and the Immune System," in *Parallel Problem Solving from Nature*, edited by H. Schwefel and R. Maenner (Springer-Verlag, 1991).

[17] S. Forrest, A. S. Perelson, L. Allen, and R. Cherukuri, "Self-Nonself Discrimination in a Computer," in *Proceedings of 1994 IEEE Symposium on Research in Security and Privacy* (1994).

[18] Y. Ishida and F. Mizessyn, "Learning Algorithms on Immune Network Model: Application to Sensor Diagnosis," *Proc. International Joint Conference on Neural Networks*, Bei Jing, (1992) 33–38.

[19] F. Mizessyn and Y. Ishida, "Immune Networks for Cement Plants," *Proc. of International Symposium on Autonomous Decentralized Systems*, (1993) 282–288.

[20] Y. Ishida, "An Immune Network Model and Its Applications to Process Diagnosis," *Systems and Computers in Japan*, **24** (1993) 38–46.

[21] S. Poljak and M. Sura, "On Periodical Behaviour in Societies with Symmetric Influences," *Combinatorica*, **3** (1983) 119–121.

[22] Y. Ishida and N. Adachi, "Active Noise Control by an Immune Algorithm," *Proc. International Conference on Evolutionary Computation 96*, Nagoya, 1996, to appear.

[23] N. K. Jerne, "The Immune System," *Sci. Am.*, **1** (1973) 52–60.

[24] Z. Grossman, "The Concept of Idiotypic Network: Deficient or Premature?" in *Theories of Immune Networks*, edited by H. Atlan and I. R. Cohen (Springer-Verlag, 1989, 38–52).

[25] P. H. Richter, "Complexity and Regulation of the Immune System: The Network Approach," in *Proc. Working Conf. on System Theory in Immunology* (Springer-Verlag, New York, 1978, 219–227).

[26] R. J. de Boer, "Symmetric Idiotypic Networks: Connectance and Switching, Stability and Supression," in *Theoretical Immunology Part II*, edited by A. S. Perelson (Addison Wesley, 1988, 265–290).

[27] M. Kaufman, "Regulation of the Immune Response: A Discrete Mapping Approach," in *Theories of Immune Networks*, edited by H. Atlan and I. R. Cohen (Springer-Verlag, 1989, 38–52).

[28] L. A. Segel and A.S. Perelson, "Shape Space Analysis of Immune Networks," in *Cell to Cell Signaling: From Experiments to Theoretical Models*, edited by A. Goldberter (Academic Press, London, 1989, 273–284).

[29] R. J. de Boer, J. D. van der Laan, and P. Hogenweg, "Randomness and Pattern Scale in the Immune Network: A Cellular Automata Approach," in *Thinking about Biology*, edited by W. Stein and F. J. Varela (Addison Wesley, Reading, MA, 1993, 231–252).

[30] S. S. Iyengar, D. N. Jayasimha, and D. Nadig, "A Versatile Architecture for the Distributed Sensor Integration Problem," *IEEE Trans. on Comput.*, **2** (1994) 175–185.

[31] R. A. Brooks, "A Robust Layered Control System for a Mobile Robot," *IEEE J. Robotics and Automation*, **RA-2** (1986) 14–23.

[32] F. P. Preparata, G. Metze, and R. T. Chien, "On the Connection Assignment Problem of Diagnosable Systems," *IEEE Trans. Comp.*, **EC-16** (1967) 848–854.

[33] M. A. Cohen, and S. Grossberg, "Absolute Stability of Global Pattern Formation and Parallel Memory Storage by Competitive Neural Networks," *IEEE Trans Sys. Man Cybern.*, **SMC-13** (1983) 815–826.

[34] J. J. Hopfield, "Neural Network and Physical Systems with Emergent Collective Computational Abilities," *Proc. Natl. Acad. Sci.,* **81** (1982) 3088–3092.