

# 自己組織型事例ベース

生 天 目 章<sup>†</sup> 小 谷 琢 磨<sup>†</sup>

事例に基づく問題解決システムにおいては、事例ベースの自己組織化機能や知的探索機能が不可欠である。本論文では、動的スキーマモデルとしてハイパーフレームモデルについて提案する。ハイパーフレームモデルに基づく自己組織型事例ベースの構築法について提案する。情報が、ある構造をもって体系的に記憶されるほど、後で想起されやすくなる。人間のそのような情報構造は記憶の自己組織化機能によって作り出されている。記憶の自己組織化とは、新しく記憶される内容が既存の記憶内容と自動的に関連づけられ、記憶ベースとして新しく組織化されることをいう。個々の事例を特徴づける記号情報を、その概念意味構造をビットコード化した意味コードを用いてインデックス化する。意味コード化されたインデックス情報と自己組織化機能や知的探索機能を手続きとしてもつオブジェクトとして個々の事例を定式化する。オブジェクト指向プログラミング手法により開発した自己組織型事例ベースのプロトタイプモデルを紹介する。

## A Self-Organizing Case Base

AKIRA NAMATAME<sup>†</sup> and TAKUMA KOTANI<sup>†</sup>

This paper discusses a new technology of building a self-organizing case base. We present the hyperframe data model and demonstrate how this new representation model will support for building a self-organizing case base. Each hyperframe encapsulates a specific set of a self-organizing mechanism and inference rules. We also address the issues involved with the intelligent case retrieval. The semantic indexing scheme is introduced in order to transform symbolic representation into bit codes. We demonstrate a tool of building a self-organizing case base based on the object-oriented programming approach.

### 1. はじめに

過去の事例を知識ベースとしてもつ知的システムの構築法に関する研究が盛んに行われている<sup>1)~4)</sup>。これらの研究の基本概念は、事例ベースとして過去の問題解決手順や方法論を蓄積しておき、その中から直面する状況や問題に類似した事例を抽出して、問題解決に役立てようとすることである。事例に基づく問題解決法は、人間の行う問題解決モデルに基づいたきわめて自然な問題解決のためのアプローチである。このような事例に基づく問題解決システムの構築においては、対象領域に関する具体的な問題解決手順や方法論を蓄積した事例ベースの内容の充実と併せて、事例ベースを構築するための枠組みとしての自己組織化機能や知的探索機能が不可欠である。

情報が、ある構造をもって体系的に記憶されるほど、想起されやすくなる。人間のそのような情報構造

は記憶の自己組織化機能によって作り出されている<sup>5)</sup>。記憶の自己組織化とは、新しく記憶される内容が既存の記憶内容と自動的に関連づけられ、記憶ベースとして新しく組織化されることをいう<sup>6)</sup>。構造化、組織化された知識の単位に対して、スキーマという名称が使用されるが、スキーマ構造は本来静的なものではなく、その状況に応じて情報構造の再編を逐次行うといった動的なものである。本論文では、動的スキーマモデルとしてハイパーフレームモデルを提案する。ハイパーフレームモデルを用いて個々の事例を記述することにより、自らの組織構造を自律的に変化させるための自己組織化機能や自己成長機能を具備した自律的な事例オブジェクトを定義することができる。自己組織型事例ベースをそのような自律型事例オブジェクトの集合として実現をする。

### 2. 事例のデータモデル

#### 2.1 ハイパーフレームモデル

フレームモデルは、しかしながら、人間の記憶に似たさまざまな機能を有している。しかしながら、現在

<sup>†</sup> 防衛大学校情報工学科  
Department of Computer Science, National Defence Academy

のフレームモデルは、フレームの集合体の組織構造を自ら獲得したり、フレームの内部データ構造を適応するための自己組織化機能が不十分である。従来のフレームモデルにこれらの機能を付加したのがハイパーフレームモデルである。記号(シンボル)表現された情報をビット表現に変換することにより処理がより効率的になる<sup>7)</sup>。個々のハイパーフレームには、記号情報以外にそれらの記号情報をビット変換したビットリストがインデックスとして付加される。ハイパーフレームの自己組織化機能や推論機能をこれらのビットコード情報を処理するための手続きとして定式化する。

ハイパーフレームモデルの概念図を図1に示す。ハイパーフレームの構成要素であるビットコードでインデックス化された組織情報と属性情報をそれぞれ組織コードおよび属性コードと呼ぶことにする。組織情報および属性情報を固有の変化しない静的記憶情報と動的に変化する動的記憶情報の2つに分類する。静的記憶とは、そのハイパーフレームが生成されたときに付与される記憶情報であり、その後の変化には一切の影響を受けない。それに対し動的記憶は、ハイパーフレーム間の関係が変更されることによって動的な変化を受ける。組織情報とは、ハイパーフレーム間に定義される概念階層上の関係情報を記述するもので、それによってハイパーフレームの集合の組織構造が規定される。概念階層上の関係情報として、一般に多様な関係を記述することができるがここでは、その中で推移則が成立するような二項関係情報を対象にする。推移則が成立するような二項関係情報を「>」または「<」で表現し、以後そのような二項関係を上位/下位関係

と呼ぶことにする。ハイパーフレームの組織情報に基づく自己組織化とは、個々のハイパーフレームの近傍に定義される関係情報から、ハイパーフレームの集合全体の組織構造を自律的に獲得することと定義する。固有組織情報とはそのハイパーフレームが生成されたときに付与され、変化を受けない組織情報である、個々のハイパーフレームは組織情報と属性情報をビットコード化した情報以外にそれらのビット情報を操作するための手続きとして定式化された自己組織化ルールと推論ルールをもつ。自己組織化ルールは、個々のハイパーフレームが自己の固有組織情報とその近傍に定義される関係情報から、ハイパーフレームの集合全体の組織構造を反映した動的組織情報を自律的な手続きによって獲得するための手順である。推論ルールは、記号表現されたハイパーフレームの組織情報や属性情報の意味に基づき関係するハイパーフレームを並列的に検索するための手順である。

2.2 組織情報のインデックス化

個々の要素間に上位/下位の関係「<」が定義されたハイパーフレームの集合  $W = \{O_i : 1 \leq i \leq N\}$  において、ハイパーフレームの組織コードを求める手順を以下に示す。

<Step 1>  $W$  の要素であるそれぞれのハイパーフレームを局所表現した  $N$  個ビットリストを構成する。すなわち、 $i$  番目のハイパーフレーム  $O_i$  に対し、 $i$  番目のビットが1で他は0となる  $N$  ビットリストで定義される固有組織コード  $d_i$  を付与する。

<Step 2> 二つのハイパーフレーム  $O_i$  および  $O_j$  に上位/下位関係 ( $O_i < O_j$ ) が成立する場合、下位関係にあるハイパーフレームの組織コードを (2.1) 式で定義する。

$$\begin{aligned} & \text{(下位ハイパーフレーム組織コード)} \\ & = \{ \text{(上位ハイパーフレームの組織コード)} \oplus \\ & \quad \text{(下位ハイパーフレーム固有組織コード)} \} \end{aligned} \tag{2.1}$$

すなわち、 $O_i$  の組織コードを上位ハイパーフレーム  $O_j$  の組織コードを  $O_i$  の固有組織コードに継承させたコードとして定義する。そのような組織コードの継承によって求まる  $O_i$  の組織コードは、上位ハイパーフレーム  $O_j$  の組織コード  $r_j$  と  $O_i$  の固有組織コード  $d_i$  との論理和

$$r_i = d_i \oplus r_j \tag{2.2}$$

によって与えられる。ここで、 $\oplus$  は各要素の論理和を表す。上位関係にあるハイパーフレームが複数存在す

Object

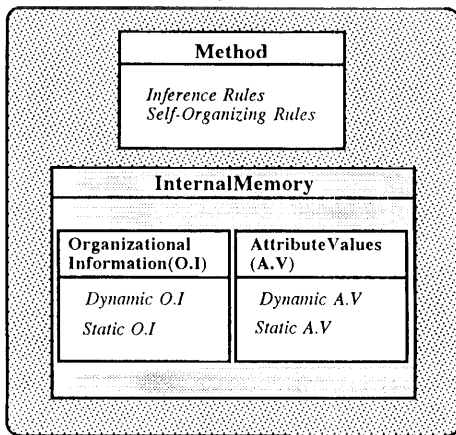


図1 ハイパーフレームデータモデル  
Fig. 1 A hyperframe data model.

の場合には、それぞれ複数の上位ハイパーフレームの組織コードの論理和を求め、それを (2.2) 式における上位ハイパーフレームの組織コード  $r_j$  とすることにより多重継承の問題にも対処することができる。このようにして求められる組織コードは、次のような性質をもつ<sup>9)</sup>。

[性質 1]

$$O_i \supset O_j \text{ ならば } r_i \otimes r_j = r_i \\ (\text{または } O_i \subset O_j \text{ ならば } r_i \otimes r_j = r_j) \quad (2.3)$$

ここで、 $\otimes$  はベクトルの各要素の論理積を表す。

自己固有の組織コードは、ハイパーフレームの集合  $W$  における固有の識別コードである。固有の組織コードは、ハイパーフレームが生成されると付与され、ハイパーフレームの集合  $W$  の組織構造の変化に対して変化を受けない。一方、(2.2) 式によって求められる組織コードは、ハイパーフレームの集合  $W$  の組織構造の変化によって影響を受ける動的な組織コードである。

### 2.3 属性情報のインデックス化

ハイパーフレームの集合  $W$  において、より下位のハイパーフレームになるにつれ、その属性情報の集合はより大きなものになる。しかしながら、その属性情報の大部分は上位ハイパーフレームの属性情報と同じである。そこで、ハイパーフレームの集合  $W$  を定義する場合には、上位ハイパーフレームと重複する属性情報はその上位ハイパーフレームに吸収させ、ハイパーフレームの集合  $W$  における検索時には下位ハイパーフレームに属性情報を継承させる方法を用いるのが効率的である。そのような属性情報の継承ルールを属性情報をビットコード化した属性コードの操作手続きとして定式化する。以下に属性コードの求め方を示す。

<Step 1> ハイパーフレームを特徴づけるそれぞれの属性の属性値を局所表現する。すなわち、 $k$  個の要素を持つ属性  $A$  の  $i$  番目の属性値に対し  $i$  番目のビットを 1 とする  $k$  ビットリストで定義される固有属性コード  $e_i$  を付与する。

<Step 2>  $W$  の要素であるハイパーフレーム  $O_i, O_j$  に上位/下位関係 ( $O_i \subset O_j$ ) が成立する場合、下位ハイパーフレームの属性コードを (2.4) 式で定義する。

$$\begin{aligned} & (\text{下位ハイパーフレーム属性コード}) \\ & = (\text{下位ハイパーフレーム固有属性コード}) \ominus \\ & \quad (\text{上位ハイパーフレームの属性コード}) \end{aligned} \quad (2.4)$$

ここで、 $\ominus$  は属性コードの各要素のビット差演算を表

す。固有属性コードは、ハイパーフレームが生成されるときに付与され、ハイパーフレームの集合  $W$  の組織構造の変化に対して影響されない。上位関係にある複数のハイパーフレームを持つ場合の属性コードは、それぞれの上位ハイパーフレームの属性コードの論理和を (2.4) 式における上位ハイパーフレームの属性コードとする。以上の式に基づき属性情報の継承を個々のハイパーフレームが自律的に行う機能をハイパーフレームの属性情報に基づく自己組織化機能と定義する。

個々のハイパーフレームを、外延的情報 (組織情報) と内包的情報 (属性情報) を用いて表現した。組織情報は、他のハイパーフレームとの関係を記述したもので、対象とするハイパーフレームの集合  $W$  の組織構造変化に伴い、外延的關係も異なるのが自然である。外延的關係の変化によって影響を受けるのが動的組織情報 (組織コード:  $r_i$ ) であり、固有組織情報 (固有組織コード:  $d_i$ ) は影響されない。また、属性情報はハイパーフレームの特徴に関する情報記述で、上位ハイパーフレームと同じ属性情報を共有することが多い。属性情報の継承によって変化を受けるのが動的属性情報 (属性コード:  $b_i$ ) であり、固有属性情報 (固有属性コード:  $e_i$ ) はハイパーフレームの集合の組織構造の変化によって影響を受けない。ビット情報に基づくハイパーフレームの自己組織化ルールは以下のように定式化することができる。

ハイパーフレームの集合  $W$  の個々の要素  $O_i$  には、(2.5) 式のビット情報がインデックス情報として付与される。

$$O_i = (r_i, d_i, b_i) \quad (2.5)$$

ハイパーフレーム  $O_i$  および  $O_j$  の組織コード  $r_i$  および  $r_j$  の論理積において

$$r_i \otimes r_j = r_i \quad (2.6)$$

が成立すれば、 $O_i$  は  $O_j$  の上位のハイパーフレーム

$$O_j \subset O_i \quad (2.7)$$

であることがわかる。(2.6) 式が成立する場合の上位ハイパーフレーム  $O_i$  からその下位ハイパーフレーム  $O_j$  への属性の継承は、

[属性継承ルール]

$$e_j = b_i \oplus b_j \quad (2.8)$$

で与えられる。さらに、すでに組織化されているハイパーフレームの集合  $W$  にハイパーフレーム  $O_{new}$  を新しく組織化する場合には、 $O_{new}$  に対し直接の上位にあるハイパーフレーム  $O_i$  の持つ属性情報と重複

する属性情報を  $O_{new}$  から削除した方法により属性情報の自己組織化をする。ハイパーフレーム  $O_i$  および  $O_{new}$  の属性コード  $e_i$  および  $e_{new}$  論理積  $e_i \otimes e_{new}$  は、 $O_i$  および  $O_{new}$  が共通して保有する属性情報を表すので、属性情報の簡素化ルールは次式で与えられる。

[属性情報簡素化ルール]

$$b_{new} = e_{new} \ominus (e_i \otimes e_{new}) \tag{2.9}$$

### 3. ハイパーフレームと意味空間

#### 3.1 意味空間の定義

記号表現された属性情報を概念意味空間上の点として表現する。すなわち、属性情報の集合（ハイパーフレームを特徴づける各属性の定義域である属性値の集合）にそれぞれの要素間の意味関係によって定義される概念意味空間を構成する。概念意味空間において上位/下位の概念関係を表す属性情報を上位/下位の属性情報と呼ぶことにする。概念階層として表現されるそのような概念意味空間（シソーラス）をビットコード化し、そのような概念意味空間のビットコードを記号表現された属性情報に付加する。記号情報された属性情報を (2.1) 式の組織コードを求める手順と同じ方法により、(3.1) 式によりビットコード化したインデックスを意味コードと定義する。図 2 にハイパーフレームの属性情報に意味コードによるインデックス情報を付加するための手順を示す。

$$\begin{aligned} (\text{意味コード}) &= ((\text{上位属性情報の意味コード}) \\ &\oplus (\text{自己属性情報の意味コード})) \end{aligned} \tag{3.1}$$

(3.1) 式で求めた属性情報の意味コードは次のような性質をもつ。すなわち、ハイパーフレームを特徴づけ

るある属性の属性値（属性情報） $a_i$  および  $a_j$  間に、概念(意味)上の上位関係「 $\succ$ 」または下位関係「 $\prec$ 」が成立するならば、それぞれの属性情報の意味コードを  $u_i$  および  $u_j$  とすると以下の式の関係が成立する。

$$a_i \succ a_j \text{ ならば } u_i \otimes u_j = u_i \tag{3.2}$$

#### 3.2 意味空間の定義とその性質

記号表現された属性情報の各要素に意味コードに基づくインデックスを付加することにより形成される意味コードの集合を、ハイパーフレームの属性情報の意味空間と定義する<sup>9)</sup>。意味空間を定義することにより、(3.2) 式を利用して記号表現された属性情報の類似度に基づくハイパーフレームの検索が可能となる。ハイパーフレーム間の類似度は、意味空間上におけるハイパーフレーム間の類似測度によって与える。ハイパーフレームのデータ定義域として位相空間  $X$  を考える。この位相空間に類似測度  $Sim$  を与えた空間  $(X, Sim)$  内の点は、類似測度  $Sim$  によって他のフレームとの類似性を計量できる。類似測度  $Sim$  の与え方は、視点に応じていくつもありうる。あるハイパーフレーム集合に対応する  $X$  について、異なる類似測度  $Sim$  を与えることによって、ハイパーフレームの見方（視点）を変化させることができる。いくつかの意味空間が組み合わせられて、新しい空間が構成されるときに、この新たに構成された空間を積空間と定義する。ハイパーフレームの意味空間は、ハイパーフレームの記号表現された属性情報の意味コード意味コード間に定義される類似測度によって定義する。

$$\begin{aligned} (\text{属性情報の意味空間}) \\ = ((\text{属性情報の意味コード}) (\text{類似測度})) \end{aligned} \tag{3.3}$$

ここで、意味コード  $u_i$  および  $u_j$  間の距離  $Dis(u_i, u_j)$  を次式で定義する。

$$Dis(u_i, u_j) = \begin{cases} \min(\|u_i\|, \|u_j\|) - \|u_i \otimes u_j\| & (a_j \succ a_i, a_j \prec a_i) \\ \max(\|u_i\|, \|u_j\|) - \|u_i \otimes u_j\| & (\text{otherwise}) \end{cases} \tag{3.4}$$

また、意味コード  $u_i$  および  $u_j$  間に類似度  $Sim(u_i, u_j)$  を次式で定義する。

$$Sim(u_i, u_j) = \frac{1}{1 + Dis(u_i, u_j)} \tag{3.5}$$

ここで  $\|\cdot\|$  は、ビットベクトルのノルムを表す。このとき (3.5) 式に対して次のような関係が成立する。[性質 2]

(ア)  $a_j \succ a_i$  ( $a_j$  が  $a_i$  の上位概念) または、 $a_i \prec$

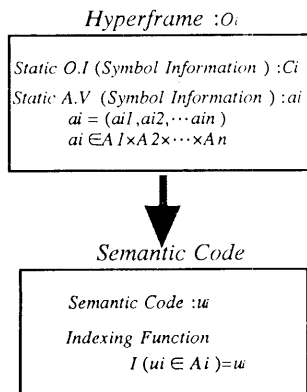


図 2 ハイパーフレームのインデックス化法  
Fig. 2 An indexing method of hyperframe.

$a_j$  ( $a_j$  が  $a_i$  の下位概念) のとき

$$Sim(u_i, u_j) = 1$$

(イ) それ以外の場合

$$Sim(u_i, u_j) = \frac{1}{\|u_j\| - \|u_i\| + 1} = 1 / (a_i \text{ と } a_j \text{ の概念階層の深さ})$$

で与えられる。ここで、概念階層の深さとは、属性情報の集合上に定義される概念階層構造上のノード間の距離と定義する<sup>11)</sup>。

次に、意味空間の性質を考える。記号表現されたハイパーフレーム  $O_i$  の属性情報の集合 (複合属性情報と呼ぶ) を  $J_i = \{a_{i1}, a_{i2}, \dots, a_{ik}\} \in A_1 \times A_2 \times \dots \times A_k$  で表し、 $u_{ij}$  で個々の属性情報の意味コードを表す。複合属性情報の個々の要素の意味コードの論理和として次式で定義される  $T_i$  をハイパーフレーム  $O_i$  複合意味コードと呼ぶ。

$$T_i = \sum_{j=1}^k u_{ij} \quad (3.6)$$

ここで「 $\sum$ 」は、ビットコード  $u_{ij}$  の論理和を表す。そのような複合意味コード  $T_i$  は次のような性質を持つ。

【性質3】 複合属性情報  $J_i$  のある構成要素  $a_{ij}$  に対して意味的に上位概念の属性情報  $a$  ( $a_{ij} \prec a$ ) の意味コード  $u$  と複合意味コード  $T_i$  の論理積は

$$T_i \otimes u = u \quad (3.7)$$

【性質4】 複合属性情報  $J_i$  の構成要素  $a_{ij}$  に対し意味的に下位概念の属性情報  $a$  ( $a_{ij} \prec a$ ) の意味コード  $u$  と複合意味コード  $T_i$  の論理積は

$$T_i \otimes u = u_{ij} \quad (3.8)$$

【性質5】 複合属性情報  $J_i$  とその構成要素である属性情報  $a$  および  $b$  の意味コード  $u$  および  $v$  の間に

$$\begin{aligned} T_i \otimes (u \otimes v) &= (T_i \otimes u) \otimes (T_i \otimes v) \\ T_i \otimes (u \oplus v) &= (T_i \otimes u) \oplus (T_i \otimes v) \end{aligned} \quad (3.9)$$

が成立する。(3.7)式から(3.9)式までの複合意味コードの性質を用いることにより、以下のようなハイパーフレームの属性情報に基づく知的検索が可能となる。

【Case 1】 ある属性情報 ( $A_i: a_{ij}$ ) の意味コード  $u_{ij}$  およびその意味的に上位概念の属性情報 ( $A_i: a$ ) (すなわち、 $(A_i: a) \succ (A_i: a_{ij})$ ) に対して、

$$\begin{aligned} T_i \otimes u &= u \\ \text{となるので} \\ Sim(T, u) &= 1 \end{aligned} \quad (3.10)$$

となる。すなわち、ある特定の属性情報をもつハイパーフレームをその属性情報の意味的に上位の概念か

らも概念間の類似度に基づき推論することが可能である。また概念間の類似度は、意味コードのビット演算により容易に求めることができる。

【Case 2】 ある属性情報 ( $A_i: a_{ij}$ ) の意味コード  $u_{ij}$  および意味的に下位概念の属性情報 ( $A_i: a$ ) (すなわち、 $(A_i: a) \prec (A_i: a_{ij})$ ) に対して、

$$T \otimes u = u_{ij}$$

となるので

$$Sim(T, u_i) = 1 / (a_{ij} \text{ と } a \text{ との概念意味構造の深さ}) \quad (3.11)$$

を得る。したがって、意味的に下位概念からも概念間の類似度に基づき推論することが可能である。

## 4. 自己組織型事例ベース

### 4.1 事例の表現

事例は、一般に事例の記述 (状況や問題に関する記述) とそれに対する解法や結果に関する記述で構成される。事例の記述においては、問題の特徴が必要かつ十分な方法で表現される必要がある。多くの事例ベースでは、状況や問題に関する普遍的な属性を用い、また属性間に従属関係がある場合には、属性に関する概念階層体系を構成し、フレーム形式で表現されることが多い。これらの方法は、事例そのものをインデックス化し、事例に含まれる数多くの情報をインデックスの中に集約化することを目的としており、事例そのもの (1次情報) に直接アクセスする必要がない場合には有効な手段である。しかしながら、多くの実際的な問題においては、事例そのものに直接アクセスする必要がある。すなわち、事例に直接アクセスすることを容易にするような事例の表現法や、事例の知的な探索法について確立することが重要である。特に、事例の表現においては、以下のことを考慮する必要がある。第1に個々の事例は、さまざまな角度から観察された多元表現として記述される。第2に事例は、それを構成する要素的な事例の複合化として表現される。第3に事例は、さまざまなメディア表現を持つマルチメディアとして表現される<sup>9),10)</sup>。さらに、与えられた状況に適合した事例を探索する場合、完全に適合することではなく部分的にしか適合しないことが普通であり、記述事例との類似性尺度 (類似性) の導入が必要である。また、事例の記述内容と表面的な特徴との適合による検索だけではなく、記述されている記号情報と意味的に適合した意味に基づく知的検索機能が必要である。

#### 4.2 事例の合成・複合化

事例に基づく問題解決法は、新しく発生した問題の解を過去の経験または発生した事例の集合の中から類似したものを求め、過去における解決方法を参考にして解く方法で、何か新しく問題に直面した場合に過去と同じような問題を連想して、そのときの解決方法から現在の問題の解決方法を類推したり、それからヒントを得て新しい解決方法を導出するといった人間の問題解決方法を踏襲している<sup>12),13)</sup>。その場合、単一的事例からヒントを得ることはまれで部分的に類似した複数事例を合成することが多い。このとき、問題をいくつかの部分問題（副問題）に分割し、別問題に展開してそれに類似した事例を探索し、それらの事例を合成することにより最初の問題に対処する方法が考えられる。このように事例を、その対象によって小さなユニットまたはカードと呼ばれる単位に分割する。これらのユニットまたはカードをハイパーテキスト<sup>14),15)</sup>を利用することによって、テキスト情報以外にグラフィックスや音声、アニメーションをも含めることが可能になる。

#### 4.3 事例の多元表現

同じ事例を複数の観点から分類し、参照することを可能にするため、複数の観点を反映した多元表現を用いる。特に、事例間の外延的関係を記述した組織情報、および事例の特徴を記述した属性情報により事例を記述する。組織情報には、事例に共通する概念階層として表現される対象領域の知識を表現するための概念階層の体系による事例の分類構造や、分析者の視点による事例の分類構造等の情報を記述することができる。属性情報にはその事例の内容、特徴についての情報を含めることができ、その事例の背景、環境、状況等を記述する。

#### 4.4 事例の組織化

事例を断片化してモジュール化しておくことにより、部分的にしか適合しない事例でも、それを他のモジュールと組み合わせ合成することによって適合することがある。しかし、断片化することにより、事例の特徴付けや事例の表現が困難になる。事例ベースの探索の効率化のためには、事例ベースの構造化・組織化が重要な問題になる。そのために、事例の重要な属性項目にしたがって組織化が必要になる。大規模な事例ベースには、複数の事例間での特徴記述の共通性に着目して事例の概念階層の体系を形成することによって、事例ベースを組織化することができ、それによ

て事例の検索の高速化や事例ベースの維持が容易になる。しかしながら、事例の組織構造をどのように獲得するか、事例の組織構造が与えられていたとしても、それらの組織構造の動的な適応機能等が問題になる。ハイパーフレームモデルを用いることにより、個々の事例は、さまざまな角度から観察された多元表現として記述され、また、事例は、それを構成する要素事例の複合事例として表現することが可能になる。また、ハイパーフレームモデルの自己組織化により新しい事例の追加や既存事例の削除が容易になる。

### 5. 自己組織型事例ベースの構築

自己組織型事例ベースとは、通常的事例ベースに自己組織化能力を付与したもので、事例の追加、削除のためのルールや、事例の類似性を判断するためのルールを各事例オブジェクトが持っており、それによって自らの構造を柔軟に変化させる機能を有している。

#### 5.1 事例のデータモデル

事例のデータモデルとしてハイパーフレームモデルを用いる。それにより、新しい事例の追加や不必要になった事例の削除など、事例ベースの環境の変化に対する適応能力（自己組織能力）を付加することができ、事例ベースの柔軟性を高め、全体として高い能力を持つ事例ベースシステムを構築することができる。事例ベースの基本構成要素である事例オブジェクトとは、個々の事例をハイパーフレームデータモデルとして記述し、それぞれのインデックスベクトルとそれらを操作するための手続きとを合わせ持つものであると定義する。ここで、事例を特徴づけるデータとは、

- 事例の各オブジェクトの内容（1次情報）
  - 各事例オブジェクトの上位関係にある事例オブジェクトの集合
  - 組織コード
  - 複合属性意味コード
- などが挙げられる。また、事例ベースの自己組織化、知的検索のための手続きとしては、
- 組織情報による事例オブジェクト組織化手続き
  - 組織情報および属性情報による事例オブジェクトの推論手続き
  - 事例オブジェクトの追加・削除のための手続き
- などがある。

このように、事例をそのデータと手続きと共にオブジェクトとしてカプセル化することによって、独立した処理単位として定義が可能となる。すなわち、事例

ベースへの新しい事例の追加を考えた場合、新たな事例オブジェクトを生成し、そのことによる他の事例オブジェクトへの影響は、各事例オブジェクトに定義されている手続きに従って自律的に処理される。環境の変化に適応するための機能についても各事例オブジェクトにカプセル化することができる。事例オブジェクトの並列化モデルの概念を図3に示す。

5.2 戦史事例の表現

自己組織型事例ベースをオブジェクト指向言語である SmallTalk/V を用いて Macintosh 上で実現した。具体的な事例としては、古代～現代までの海戦の戦闘事例 40 個について実施した。以下ではまず、具体的な戦史事例の表現について説明し、次に SmallTalk/V 上でのインプリメント方法について簡単に示す。戦史情報を伝えることとは、これらの行動を起こした人、場所、原因や、行動の対象物、行動の手法という情報を伝えることに等しい。一般的に戦史情報を扱う場合には、これらの情報を的確に伝えるために、5W1H (When, Where, Who, What, Why, How) で表されるイベント情報として記述する。

本モデルの事例として古代～現代までの 40 個の逆転海戦事例（兵力が劣っていたにもかかわらず勝利した海戦史<sup>16)</sup>）を組織情報および属性情報として

5W1H で記述されるイベント情報、およびキーワードを用いて表現した (図4(a))。ただし、What は「逆転海戦」であるので本応用例では考慮していない。また、キーワードは逆転要因となった 2~6 個の事項を各事例に付与した。事例のデータモデルを (5.1) 式に示す。

$$(逆転戦史) = [(クラス) (5W1H) (キーワード)] \quad (5.1)$$

さらに、イベント情報には意味コードを付与し、意味に基づく知的検索を可能にした。図4(b)に Why 情報に関する意味コードの例を示す。

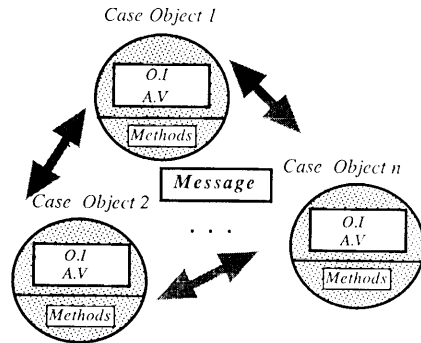


図3 事例オブジェクトモデル  
Fig. 3 A case object model.

<p>#1 リウム</p> <p>ako:ペロポネソス when:BC428 where:地中海 who1:アテネ who2:スバルタ why: how:戦意不足 戦力比:0.4</p> <p>key:パニック, 突破翼側, 戦意</p>	<p>#2 リウム</p> <p>ako:ペロポネソス when:BC428 where:地中海 who1:アテネ who2:スバルタ why:反転攻撃 how:攻撃不均一 戦力比:0.3</p> <p>key:奇襲・奇略パニック, 集中, 戦技, 戦意</p>	<p>シラクサ</p> <p>ako:ペロポネソス when:BC413 where:地中海 who1:シラクサ who2:アテネ why:火船 how:- 戦力比:0.9</p> <p>key:奇襲・奇略パニック, 戦意</p>	<p>ナウロクス</p> <p>ako:古代 when:BC36,10 where:地中海 who1:- who2:- why:衝角防材, 火船 how:- 戦力比:1.0</p> <p>key:装備位相差, 戦技</p>
---	---	---	---

図4(a) 海戦史事例  
Fig. 4(a) Case example of sea war.

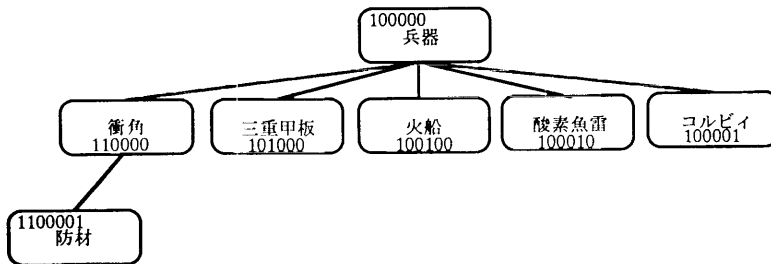


図4(b) Why 属性の意味コード  
Fig. 4(b) Semantic codes of Why attribute.

### 5.3 SmallTalk/V 上でのクラス

コンピュータ上に作成したクラスは、大きく分けて5つのクラスをもつ。システムがすでにもつトップクラス“Object”クラスのすぐ下に、自己組織型事例ベースの基本要素である事例オブジェクトのデータ・手続きをもつ“CaseModel”クラス、事例オブジェクト管理用インターフェイスをつかさどる“Case-Browser”クラス、検索を実施する“CaseSearcher”クラス、クラスの上位・下位関係を指定する“Case-Chooser”クラス、事例オブジェクトの生成・削除その他一般動作を管理する“Case”クラスである(図5参照)。

#### (1) 事例オブジェクトの基本クラス

##### ●CaseModel クラス

自己組織型事例ベースの基本要素となる事例オブジェクトの本体(データ、手続き)となる抽象クラスである。Method クラスや CaseObject クラスのスーパークラスとなる。

#### (ア) Method クラス

各事例オブジェクトの手続きを定義するクラスで、3つのサブクラスをもつ抽象クラスである。ハイパーフレームの自己組織化ルール、検索アルゴリズムを定義し、組織情報・属性情報を利用して各事例オブジェクトに対するメッセージ機能を提供する。

#### (イ) CaseObject クラス

各事例オブジェクトに属する情報に関するクラスで、3つのサブクラスをもつ抽象クラスである。“Method”クラスに対応した情報を内部にもつ。

#### (2) シミュレーション環境用クラス

##### ●Case Browser クラス

事例ベースに登録されている現存のすべての事例オブジェクトを表示し、各事例オブジェクトごとにオブジェクトの内部状態を観察するウィンドウを提供する。また新しく事例を登録するためのユーザインターフェイスウィンドウを提供するクラスである。入力された事例の情報を“Case”にメッセージとして送るメソッドを持つ。このウィンドウを通じて、事例オブジェクトの生成・削除、事例オブジェクトや1次情報のインスペクトウィンドウの起動が可能である。

##### ●Case Searcher クラス

事例検索のための事例検索ウィンドウを提供するクラスである。入力されたキーワード、属性および組織情報を事例オブジェクトにメッセージとして送る機能を持つ。

##### ●Category Chooser クラス

組織情報を入力する際のオブジェクトの上下関係を指定するユーザインターフェイスウィンドウを提供するクラスである。これにより容易にオブジェクトの上下関係を入力することが可能になる。

##### ●Case クラス

ハイパーフレームモデルに基づく事例オブジェクトを実現するクラスである。このクラスの重要な機能は、

- (ア) 各事例オブジェクトの生成
- (イ) 各事例オブジェクトごとの学習例の生成およびメッセージ通信
- (ウ) 検索における協調動作

といった事例オブジェクト間の処理を実際に行うクラスである。

### 5.4 シミュレーション

本小節では、例を用いて簡単にシミュレーションのイメージを示す。最初に CaseBrowser を生成する(図6参照)。次に Case を生成し、事例を追加する。メニューから [AddCase] を選択し、事例のタイトルを入力する(図7参照)。続いて事例のキーワード、属性およびクラス情報を入力する。事例は“Open-Case”によって参照することができる(図8参照)。[MakeExample] を選択すると入力された情報が Case に送られる。Case は、送られた事例オブジェクトから各事例オブジェクトごとの学習例を生成する。各事例オブジェクトが、自己に定義された手続き(自己組織化ルール、推論ルール)を用いて獲得したオブジェクト内部状態は、CaseBrowser で確認できる。各事例オブジェクトは外部からのメッセージに基づき内部状態を自律的に変化させることができる。事例を検索するときは、CaseSearcher に検索したい情報を入力しメニューから [CaseSearch] を選択することに

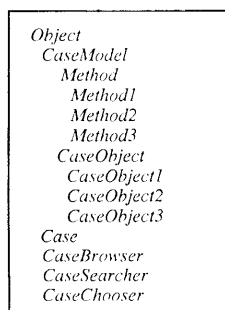


図5 SmallTalk/V 上におけるクラス  
Fig. 5 Classes of SmallTalk/V.



より、その情報が Case にメッセージとして送られる。Case は、その情報を解釈して各事例オブジェクトにそれぞれの入力メッセージを送り、その返答メッセージを CaseSearcher に検索結果として送る (図 9 参照)。

ハイパーフレームモデルを用いた自己組織型事例ベースの有効性を確かめるために以下の実験を行った。具体的な事例としては、古代～現代までの逆転海戦の戦闘事例 40 個について自己組織型事例ベースに

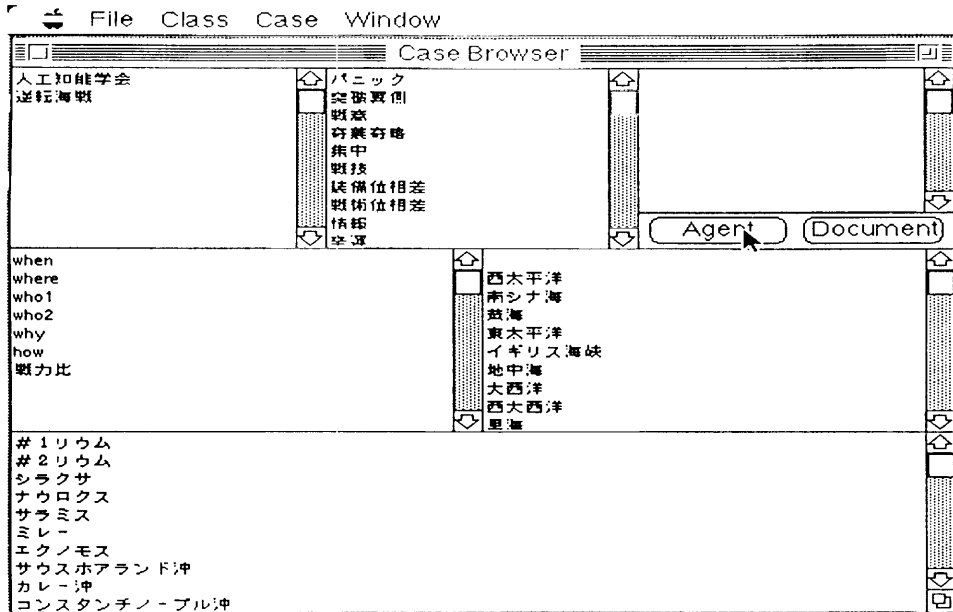


図 6 Case Browser  
Fig. 6 Case Browser.

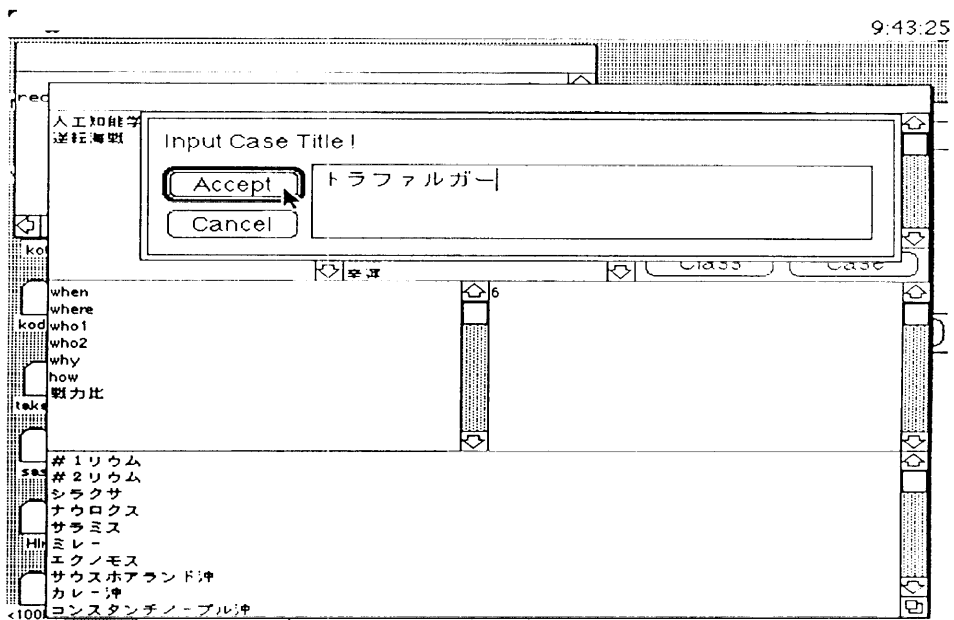


図 7 事例タイトル入力  
Fig. 7 Input of case title.

登録し検索を行った(図4(a)参照)。海戦の年代、その海戦が起こった戦争名に基づいて与えた事例間の2項関係情報から図10に示した戦史事例の組織構造が獲得されることを確認をした。すなわち、それぞれの事例オブジェクトは、図10に示した事例オブジェクト全体の組織構造における自己の位置づけを反映したビット情報をその動的な組織コードとして獲得できた

ことを確認した。また、事例オブジェクトを特徴づけるために当初記述された属性情報も図10に示した事例オブジェクト組織構造に応じて上位の事例オブジェクトと重複するものは下位の事例オブジェクトの属性情報の集合から自動的に削除されることを確認した。CaseSearcherを用いて、不完全な情報を事例オブジェクトに与え、事例の検索をシミュレーションとし

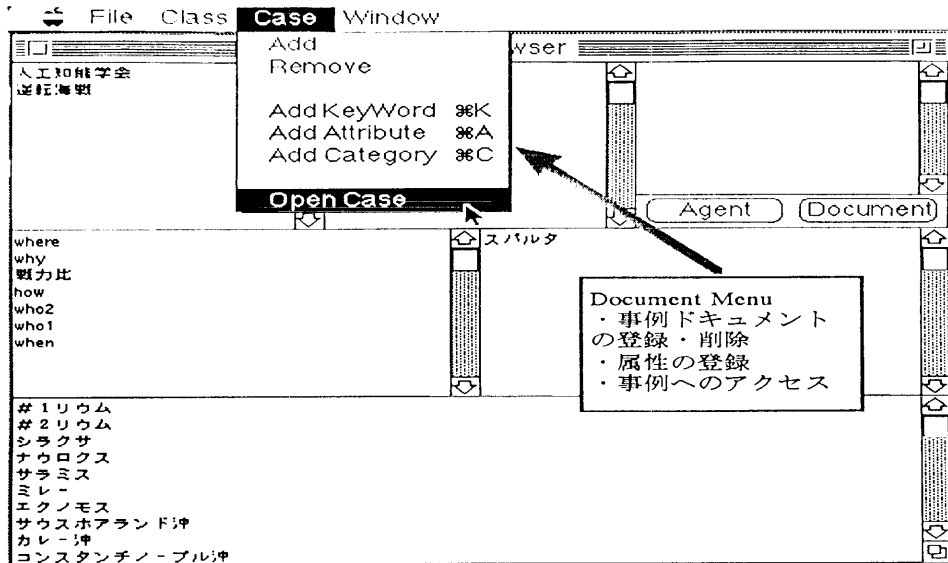


図 8(a) Open Case  
Fig. 8(a) Open Case.

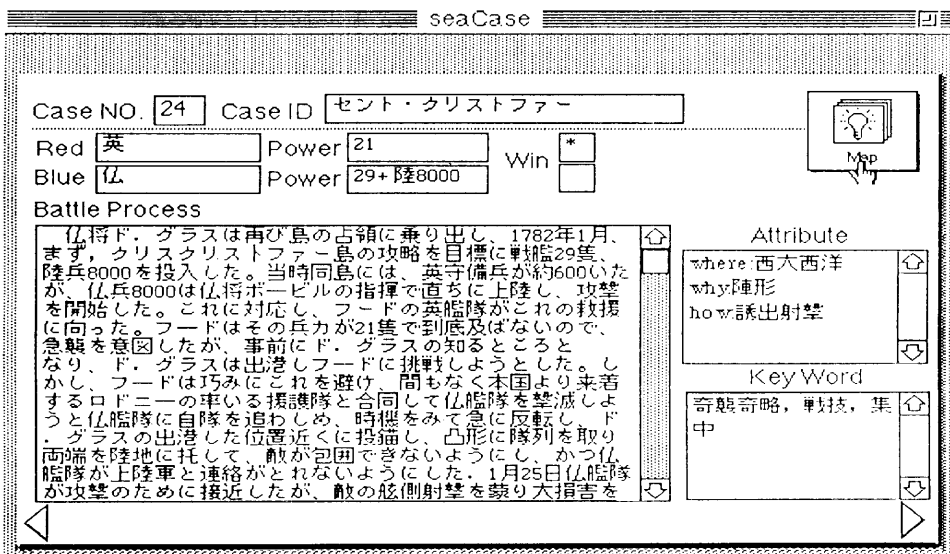


図 8(b) 事例カード  
Fig. 8(b) Case Card.

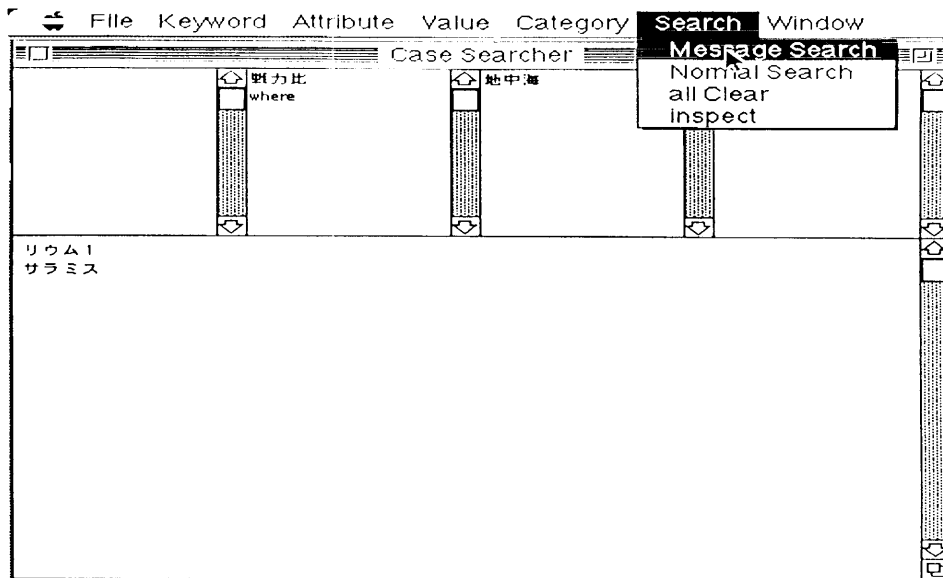


図 9 Case Searcher  
Fig. 9 Case Searcher.

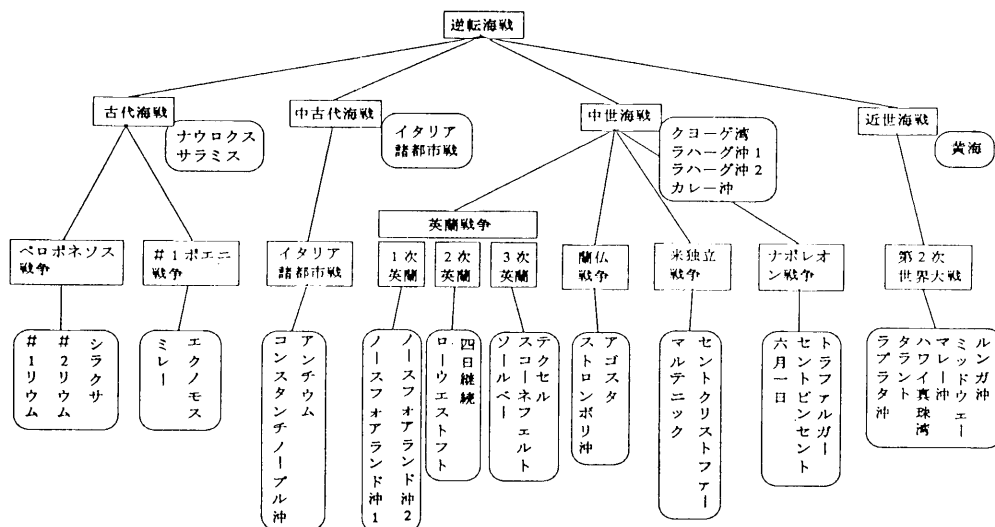


図 10 事例の組織構造  
Fig. 10 Case structure.

て行い、事例オブジェクトの属性情報の意味に基づく知的検索機能を確認した。また、組織情報および属性情報の2つの情報を融合した検索も可能である。例えば、キーワード検索においては半数以上キーワードが欠けると検索不能であったが、属性検索と組織情報による検索を併用することにより適切な事例オブジェクトが検索されることが確認できた。

## 6. むすび

事例に基づく問題解決システムの核となる事例ベースに求められる要件は、新しい事例の追加や既存の事例の変更に対する事例ベースの自己組織化機能や意味に基づく事例の知的探索機能である。本論文では、動的スキーマモデルとしてハイパーフレームモデルにつ

いて提案した。ハイパーフレームモデルを用いた自己成長機能を具備した自己組織型事例ベースについて提案した。ハイパーフレームモデルは推移則が成立する関係情報を組織情報としてもつ。本論文では一種類の関係情報しか扱っていないが、組織情報として多元表現をすることにより複数の種類の関係情報を同時に扱うことも可能である。また記号(シンボル)で表現された情報をビット表現に変換したインデックス情報を用いてハイパーフレームの自己組織化モデルや記号情報された属性情報の意味に基づく知的推論機能を実現した。ビットコード化において大規模なモデルにも対処できるよう工夫をこらしているが、複合ビットコード化等により大規模な事例ベースの構築がより効率的にするためのインデックス化技術の確立が今後の課題である。さらに、事例に基づく問題解決法といった実際的な問題解決システムの構築においては、ビットコードによるインデックス化以前に事例オブジェクトの関係情報や属性情報を多元的な観点から記号表現するためのオントロジーを整備する必要がある<sup>17)</sup>。

### 参 考 文 献

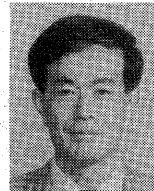
- 1) Riesbeck, C. and Schank, R.: *Inside Case-Based Reasoning*, Lawrence Erlbaum (1989).
- 2) 松原 仁: 推論技術の観点からみた事例に基づく推論, 人工知能学会誌, Vol. 7, No. 4, pp. 567-574 (1992).
- 3) Creecy, R., Msard, B., Smith, S. and Waltz, D.J.: A Trading Mips and Memory for Knowledge Engineering, *Comm. of ACM*, Vol. 35, No. 8, pp. 48-63 (1992).
- 4) Stanfill, C. and Waltz, D.: Toward Memory-Based Reasoning, *Comm. of ACM*, Vol. 29, No. 12, pp. 1213-1228 (1986).
- 5) Kohonen, T.: *Self-Organization and Associative Memory*, Springer-Verlag (1988).
- 6) 鈴木 寿, 有本 卓: 連想データベース自己組織化と自律移動知能ロボットへの応用, 日本ロボット学会誌, Vol. 5, No. 6, pp. 431-441 (1987).
- 7) Bertino, E. and Kim, W.: Indexing Techniques for Queries on Nested Objects, *IEEE Trans. on Knowledge and Data Engineering*, Vol. 1, No. 2, pp. 196-214 (1989).
- 8) 生天目章, 西村和幸: 自己組織型ハイパーネットワーク, システム制御情報学会論文誌, Vol. 6, No. 3, pp. 123-136 (1993).
- 9) 田淵仁浩, 村岡洋一: 距離空間データモデル MeSOD におけるオブジェクト概念について, 電

子情報通信学会論文誌, D-1, Vol. J74-D-1, No. 4, pp. 296-307 (1991).

- 10) Meghini, C. and Rabitti, F.: Conceptual Modeling of Multimedia Documents, *IEEE Computer*, Vol. 24, No. 10, pp. 23-29 (1991).
- 11) 長尾 真: 言語工学, 昭晃堂 (1980).
- 12) Klein, G. and Calderwood, Y.G.: How Do People Use Analogues to Make Decisions?, *Proceedings of the DARPA Workshop on Case-Based Reasoning*, pp. 209-223, Morgan Kaufmann (1988).
- 13) Kolodner, J.L.: Improving Human Decision Making through Case-Based Decision Aiding, *AI Magazine*, Vol. 12, No. 2, pp. 52-68 (1991).
- 14) Conklin, J.: Hypertext; An Introduction and Survey, *IEEE Computer*, Vol. 12, No. 9, pp. 17-41 (1987).
- 15) Smith, J.B. and Weiss, S.F.: HYPERTEXT, *Comm. ACM*, Vol. 31, No. 7, pp. 816-819 (1988).
- 16) 菊池 宏: 戦略基礎理論, 内外出版 (1982).
- 17) Schank, R. and Osgood, R.: A Content Theory of Memory Indexing, TR-2, Northwestern University (1990).

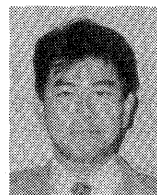
(平成 5 年 4 月 9 日受付)

(平成 6 年 4 月 21 日採録)



生天目 章 (正会員)

1950年10月4日生。1973年防衛大学校卒業(応用物理学専攻)。1977年および1979年スタンフォード大学院修士及び博士課程修了(Ph.D)。同年航空幕僚監部勤務。1987~1988年ジョージメイソン大学客員助教授。現在、防衛大学校情報工学教室助教授。人工知能、ニューラルネットワーク、意思決定工学等の研究に従事。人工知能学会、ソフトウェア科学会、神経回路学会、AAAI、ACM、IEEE 学会各会員。



小谷 琢磨 (正会員)

1965年10月10日生。1988年防衛大学校卒業(応用物理学専攻)1993年防衛大学校研究科(オペレーションズリサーチ専攻)修了。現在、陸上自衛隊勤務。人工知能学会会員。