

ユーザからの非明示的な要求の獲得手法の提案と その勤務表作成問題への応用

水戸将弥[†] 藤田 聡[†]

[†] 広島大学大学院 工学研究科 情報工学専攻
〒739-8527 広島県東広島市鏡山一丁目4-1
{mito,fujita}@se.hiroshima-u.ac.jp

あらまし: 近年の情報処理技術の飛躍的な進歩に伴い、コンピュータを介した人間同士のコミュニケーションが日常生活の様々な局面で用いられ始めている。本稿では、ユーザとシステムとの間の知的なインタフェースを実現するためのひとつのアプローチとして、ユーザからの非明示的な要求を自動的に獲得するための手法の提案をおこなう。要求の獲得は、ユーザとの対話の中で得られる選択の系列をもとにおこなわれる。また提案手法の応用例として勤務表作成問題を取りあげ、労働者の出勤パターンに関する嗜好の自動獲得を試みる。

A method for acquiring inexplicit requirements from users, and its application to the shift scheduling problem

Masaya Mito[†] Satoshi Fujita[†]

[†] Graduate School of Engineering, Hiroshima University
Kagamiyama 1 chome 4-1, Higashi-Hiroshima, 739-8527 Japan

Abstract: In this paper, we propose a method for acquiring inexplicit requirements from users via the interaction between users and the computer system. The proposed method is applied to the shift scheduling problem, that is the problem of assigning shift to each labor in such a way that as many requirements are satisfied as possible. An outline of the prototype system that is presently under construction is also given.

1 はじめに

人間の行動や選択は一般に不合理なものであり、時間や場所によって矛盾する振る舞いを見せることもしばしばである。またそれらの選択の中で意識的におこなわれるものはごくわずかであり、そのほとんどの部分は、過去の経験に基づく無意識的な判断やある種の嗜好などによって、自分でも意識しないままになされていると考えられる。したがって、たとえば数人で資源の配分をおこなう場合などには、お互いの履歴や嗜好などを把握しているほうがより早く適切な合意状態に達する場合が多い。またオンラインオークションのように不特定多数の人間が参加

するシステムなどでは、各ユーザが自分自身でも明確には意識していない嗜好を自動的にシステムに取り込むことによって、その使いやすさが大幅に向上することが期待される。

本稿では、コンピュータシステムを介した人間どうしのコミュニケーションの具体例として勤務表作成問題を取りあげ、この問題の解決過程におけるユーザからの非明示的な意図の自動抽出方法の提案をおこなう。勤務表作成問題とは、各労働者から提示される出欠勤希望日のリストと雇用者側が満たさなくてはならない制約のリストとから、すべての(あるいはできるだけ多くの)制約を満足するような勤務シフ

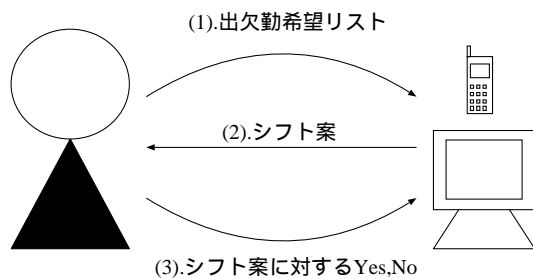


図 1: 労働者と計算機の対話。

トの割り当てを求める問題である¹。この問題は一般に多目的組合せ最適化問題や不完全制約充足問題として定式化され、これまでも遺伝的アルゴリズム [1] や制約論理プログラミング [2, 3]、タブーサーチ [4] などの解法が数多く提案されてきている。しかし従来の研究は、以下のような点で、本来解かれるべき問題の一部分しか取り扱っていないと考えられる:

- 各ユーザは一般に自分が満たしてほしいすべての要求を明示するわけではない。したがってアルゴリズムに対して提示されていない要求が満たされていないことによる不満がユーザ側に残ってしまう可能性が常に存在する。
- 従来の研究では、明示的に提示されない要求に対するモデル化が不十分であった。それらの要求の中には、休日の取り方の癖などある程度個々のユーザに関して不変な部分があるいっぽうで、外的な要因によって簡単に変化してしまうという側面もある。これらの要求を整理・分類し、状況の変化に対して柔軟に対応できる処理方式の枠組みを提案することが必要である。

本稿で提案する方式では、月ごとに定期的に各ユーザから提示される(明示的な)要求の履歴と、勤務シフト作成時にシステムによって提示されるシフト案に対するユーザからの応答の履歴とから、ユーザの非明示的な意図の獲得をおこなう。システムの問題図を図 1 に示す。この図に示されるように、勤務シフトは、最初に提示された出欠勤希望リストをたたき台として労働者とスケジューラとの間で対話的に構成されていくが、それと同時に、対話から得られる選択の系列をもとにした嗜好の獲得がなされる。獲得された嗜好は次回の勤務表作成時に、より

¹ ナーススケジューリングやパイロットスケジューリングのような特殊な勤務形態に特化した問題設定もよくおこなわれている。

適切なシフト案を提示するために用いられる。なお本稿では「嗜好とはシフト集合の分割方法のことである」という立場をとる。すなわち、可能なすべてのシフトの集合の中で労働者にとって受け入れ可能なシフトたちからなる部分集合が、その労働者の嗜好であるとする。ただし一般に、受け入れ可能なシフト集合とその補集合との境界は非常に複雑な形になっていることが予想されるため、提案手法ではシフト集合空間上での直接的な分割はおこなわず、履歴から得られたサンプルシフトから適切な特徴ベクトル空間への写像をおこない、その上で特徴ベクトル空間のボロノイ分割をおこなうというアプローチをとる。

本稿の構成は以下の通りである。まず最初に 2 節では問題の分析をおこない、そのモデル化をおこなう。3 節では予備実験の結果に基づき、特徴ベクトルやその選択基準を具体的に定義する。4 節では、提案する手法に基づいて現在作成中のプロトタイプシステムの設計方針とその概観について説明する。最後に 5 節では今後の方針について述べる。

2 モデル

2.1 シフトの表現

本稿では勤務表作成は 4 週間ごとにおこなうものとする。この 4 週間のことを以下では期間と呼び、期間ごとの各労働者のシフトを長さ 28 のビットベクトルであらわすことにする²。ここで、ある期間のシフトに対応するベクトルの第 i ビット目の値が 0 のとき、その期間の第 i 日目が欠勤であることを意味し、1 のとき出勤であることを意味している。可能なすべてのシフトからなる集合を U と記す。定義よりあきらかに $U = \{0, 1\}^{28}$ である。各労働者は、勤務表作成開始時に、対応する期間における出欠勤に関する要求をシステムに対して明示する。ここで出欠勤に関する要求は期間中のすべての日に対して与える必要はなく、未定をあらわす記号*を用いることができるものとする。労働者によって提示される可能性のあるすべての要求の集合を V と記す。以下では V の各要素を出欠勤希望リスト(あるいは単に希望リスト)と呼ぶことがある。定義より $V = \{0, 1, *\}^{28}$ であることに注意されたい。

² 3 交替制の勤務をおこなう職場などに対しては、84 ($= 3 \times 28$) ビットのベクトルを使うことで同様の議論をおこなうことができる。

労働者によって提示された希望リストを $x \in \mathcal{V}$ とする。システムは労働者に対して U 中の適切な³ 要素 y を提示し、提示されたシフト案に対する労働者の判断をあおぐ。労働者は提示されたシフト案 y に対して受理 (1)・非受理 (0) のいずれかの応答をシステムに対して返し、もし応答が非受理の場合は、システムはあらたなシフト案の作成とその提示を、労働者が受理応答を返すまで繰り返しおこなう。

以上の一連の流れから、システムはユーザ (労働者) の嗜好に関する情報を、以下の 2 つの系列から得ることができることがわかる:

1. ユーザによって提示された希望リスト ($\in \mathcal{V}$) の系列 x_1, x_2, \dots, x_n
2. システムの提示したシフト案 ($\in U$) とそれに対する応答、およびシーケンス番号 t_i からなる組の系列 $(y_1, b_1, t_1), (y_2, b_2, t_2), \dots, (y_m, b_m, t_m)$ 。ここで $y_i \in U$ かつ $b_i \in \{0, 1\}$ である。またシーケンス番号 t_i はあたらしい期間に入るたびにひとつずつ増加する整数とし、 $t_1 = 0$ のように初期化されるものとする。

なお、本稿で提案する方式では、後者の系列から得られる情報のみに着目する⁴。

2.2 分割の基本方針

前述のように本稿では、「嗜好とはシフトの全体集合 U の分割の仕方のことである」という立場をとる。ユーザが (ある時点で) 受け入れ可能なシフトの集合を $\mathcal{A} \subseteq U$ であらわし、その補集合を $\bar{\mathcal{A}} = U - \mathcal{A}$ で定義する。集合 U の部分集合 \mathcal{A} と $\bar{\mathcal{A}}$ への分割の境界線は、一般に複雑な形状をしていることが予想される。具体的には以下のような状況が考えられる:

1. 期間中に出勤可能な日数には上限と下限とがあり、ビットベクトル中の 1 の個数は、通常ある一定の範囲内におさまってはいなくてはならない
2. 特定の条件を満たした日 (たとえば第 2 水曜日や第 2 土曜日など) の出欠勤に関する条件が付与される場合がある
3. 連続休暇や連続出勤に関する希望がだされる場合がある

³ ただしベクトル y は、最初に労働者によって提示された希望リスト x と、以下のような意味で無矛盾なベクトルでなくてはならない: x の第 i 要素が*以外ならば、 y の第 i 要素は x の第 i 要素と一致する。

⁴ ポジティブな例題のみからの学習が非常に困難であることがそのおもな理由である。

また境界線には、時間の経過や周囲の状況の変化に伴って動的に変化するという性質がある。さらに本システムの性格上、ユーザとの対話によって得られるサンプルの数はあまり多くすることができない⁵。

以上のような考察から本稿では、次のような方法でシフトの分割方法を獲得することを試みる:

- シフトを複数の特徴ベクトル空間へ写像し、各特徴ベクトル空間内での分割 (クラスタリング) をおこなう。
- 嗜好の動的な変化に対応するため、クラスタリングではより新しいサンプルがより重視されるようなメカニズムを用いる。
- 全体的な分割は特徴ベクトル空間での分割を統合することによって得られる。ただし、各特徴ベクトルがすべてのユーザにとって等しく意味のある選択基準であるとは限らないことから、ユーザの行動履歴とベクトルの特殊性とからそのユーザにとって意味のあるベクトルのみを選択し、それらを統合するというアプローチをとる。

なお本稿では、特徴ベクトルは次のような 3 つの観点から定義することにする: 1) 月や週など特定の期間にわたる出勤・欠勤のバランスや分布 (大域的側面)、2) 連続出勤や連続欠勤など、局所的な出勤・欠勤のあらわれかた (局所的側面)、3) 特定の曜日での出勤・欠勤状況 (周期的側面)。特徴ベクトルの具体例については 3 節を参照されたい。

2.3 特徴ベクトル空間の分割

さてシステムが提示したシフト案 y_1 に対してユーザが 0 を返し、その後で提示されたシフト案 y_2 に対して 1 を返したとしよう。このとき、以下の言明が成立する:

シフトベクトル空間上で y_1 と y_2 を結ぶ直線上に \mathcal{A} と $\bar{\mathcal{A}}$ の境界が存在する。

この境界の (直線上での) 具体的な位置については明確にいうことはできないが、 y_1 と y_2 を結ぶ線分上では、 y_2 に近ければ近いほど 1 を返す (つまり \mathcal{A} に属する) 可能性が高いといえる。また y_2 を通って y_1 と

⁵ ユーザから提示された希望リスト ($\in U$) はそれまでに勤務した月数分しか存在しないし、システムの基本的な設計方針としてはできるだけ早く満足のいくシフト案に合意することが望ましいため、ネガティブな応答の数はできるだけ少なくする必要がある。

は逆方向にのびていくような半直線上では、(もしほかに何の情報もなければ) y_2 より遠ざかれば遠ざかるほど1を返す可能性が高いと考えられる。同様の議論は、特徴ベクトルに写像された後も成立する。すなわち、シフト案 y_1 から写像された特徴ベクトル $f(y_1)$ に対する応答とシフト案 y_2 から写像された特徴ベクトル $f(y_2)$ に対する応答とが異なるとき、シフトの(特徴ベクトル空間における)境界は、 $f(y_1)$ と $f(y_2)$ とを結ぶ直線上に存在する。

本稿では、特徴ベクトル空間に時間軸を加えた拡張特徴ベクトル空間を考え、その上でサンプル点によるベクトル空間のボロノイ分割をおこなう。サンプル (y_i, b_i, t_i) に対する(拡張)特徴ベクトルを $f(y_i, t_i)$ とする。点集合 $\{f(y_1, t_1), f(y_2, t_2), \dots, f(y_p, t_p)\}$ が与えられたとき、拡張特徴ベクトル空間を以下のような p 個の領域 A_1, A_2, \dots, A_p に分割することができる: A_i 中の任意の点 (x, t) について、 $|f(y_i, t_i) - (x, t)| \leq |f(y_j, t_j) - (x, t)|$ 、ただし $|\cdot|$ はユークリッドノルムをあらわす(このような分割のことをボロノイ分割と呼ぶ[5])。サンプルの中で応答1とともに保存されているものの集合を(一般性を失うことなく) $\{(y_1, b_1, t_1), (y_2, b_2, t_2), \dots, (y_q, b_q, t_q)\}$ とすると、ユーザに受け入れ可能なシフトに対応する特徴ベクトルの集合 B は、

$$B = A_1 \cup A_2 \cup \dots \cup A_q$$

によって近似的に表現されることになる。現在のシークエンス番号を t' とする。以下では、あるシフト案 y がユーザにとって受け入れ可能であるかどうかを判定する際、 $t = t'$ となるような射影空間上で、そのシフト案が B の像に属しているかどうかをチェックするという手法をとる。これにより、新しいサンプル点が判定に対して与える影響をより大きくすることが可能となる。

2.4 特徴ベクトルの取捨選択

特徴ベクトルの有効性は、すべてのユーザにとって均一的なものであるとは限らない。あるユーザにとっては非常によく特徴をとらえることのできる特徴ベクトルであっても、ほかのユーザにとってはほとんど意味のない場合もありうる(たとえばあるユーザは特定の曜日の欠勤にこだわるが、別のユーザはそのようなこだわりをもたないなど)。

各ユーザにとって意味のある特徴ベクトルを自動

的に検出するための方法として、本稿では以下の2点に着目する:

1. 特徴ベクトルの値が複数の労働者の平均的な値と比べて特異であるとき、その特徴ベクトルをそのユーザにとって意味のあるものとみなす(空間的特異性)。
2. 特徴ベクトルの値が数期間にわたって同じような傾向を示しているとき、その特徴ベクトルがそのユーザにとって意味のあるものであるとみなす(時間的継続性)。

2.5 特徴ベクトルの統合

特徴ベクトル空間ごとにおこなった判定結果の統合は以下のようにおこなわれる。あるユーザにとって意味のある特徴ベクトルを f_1, f_2, \dots, f_k とする。シフト案 y は、すべての f_i ($1 \leq i \leq k$)に関して y が受け入れ可能であると判断されたとき、ユーザにとって受け入れ可能なシフトであるとする。

3 具体例

本節では、前節で提案した手法の実現方法を具体例によって示す。

3.1 特徴ベクトル

前述のように、各シフトは長さ28のビットベクトルであらわされる。シフト x の第 i 要素を $x[i]$ であらわす。以下では、シフト x の特徴の大域的側面に着目した特徴ベクトル $f_1(x)$ 、周期的側面に着目した特徴ベクトル $f_2(x)$ 、局所的側面に着目した特徴ベクトル $f_3(x), f_4(x)$ の合計4つを定義する。ベクトル $f_1(x)$ は長さ4のベクトルであり、 x 中の各週ごとの出勤日の数を要素としてもつ。またベクトル $f_2(x)$ は長さ7のベクトルであり、 x 中の各曜日ごとの出勤日の数を要素としてもつ。 $f_3(x)$ と $f_4(x)$ はそれぞれ x 中で連続出勤・連続欠勤がどのくらいあったのかを示すベクトルである。可能性としては最大で28日の連続出勤や連続欠勤がおこることもありうるが、ここでは適切なパラメータ ℓ, m を用いて、その長さがそれぞれ ℓ, m におさえられるようにしてある。各ベクトルの定義は以下の通りである:

$$f_1(x) \stackrel{\text{def}}{=} \sum_{j=0}^6 (x[j], x[j+7], \dots, x[j+21])$$

$$f_2(x) \stackrel{\text{def}}{=} \sum_{i=0}^3 (x[7i], x[7i+1], \dots, x[7i+6])$$

$$f_3(x) \stackrel{\text{def}}{=} (p_1, p_2, \dots, p_{\ell-1}, \sum_{i=\ell}^{27} p_i)$$

$$f_4(x) \stackrel{\text{def}}{=} (q_1, q_2, \dots, q_{m-1}, \sum_{i=m}^{27} q_i)$$

ここで p_i はシフト x 中で i 日間連続出勤をした回数、 q_i はシフト中で i 日間連続欠勤した回数をそれぞれあらわす。

ある事業所で採集されたシフト集合から得られた特徴ベクトルの具体例を表 1 に示す(表 1(a) の距離は、特徴ベクトルの平均からのユークリッド距離を表す。詳細については 3.2 節を参照されたい)。この事業所ではたらいっているのは約 20 名のパート職員(大学生, 社会人)であり、採集は 2001 年 1 月 1 日から 7 月 15 日までの 28 週間おこなわれた。表では、社会人の労働者 A と大学生の労働者 B, C, D の計 4 名について計算された特徴ベクトルの一部が示されている。なお 2001 年 1 月 1 日は月曜日だったため、 $f_2(x)$ の第 1 要素は月曜日の出勤回数となっている。

(a) はベクトル $f_1(x)$ の具体例である。労働者 A が期間 1 の第 2 週と期間 4 の第 3 週を除いてほぼ全ての週で平均的に出勤しているのに対し、労働者 B の出勤の分布には偏りがあることがわかる(1 週間のうち 6 日出勤している週があるかと思えば、1 週間のうち 1 日も出勤していない週もある)。このことから、労働者 A が偏りがあることを好まず、労働者 B が出勤の大域的分布に偏りがあることを好む労働者ではないかと推測できる。(b) はベクトル $f_2(x)$ の具体例である。労働者 B がいろいろな曜日にまんべんなく出勤しているのに対し、労働者 A は特定の曜日に出勤する傾向が強いことがわかる(月の前後半に関する分布の偏りとまったく逆の傾向を示していることが興味深い)。(c) はベクトル $f_3(x)$ の具体例である。労働者 C が連続出勤を好んでいるのに対し、労働者 A は連続した出勤をほとんどおこなっていないことがわかる((b) からわかるように、労働者 A の出勤日は月曜日、水曜日、金曜日が 95% 以上を占めている)。(d) はベクトル $f_4(x)$ の具体例である。この表からは、労働者 D が 3 日間の連続欠勤を特に好み、労働者 A は 1 日~2 日程度の連続欠勤を好むのではないかと推測される。

表 1: 各ベクトルの具体例(各労働者ごとに採集された 28 週分のシフトからそれぞれの特徴ベクトルを生成したもの。ここで、たとえば期間 1 は、2001 年第 1 週から第 4 週までのシフトを表す。)

(a) ベクトル $f_1(x)$

期間	労働者 A	距離	労働者 B	距離
1	(4,1,2,3)	2.24	(0,6,3,4)	4.33
2	(3,3,3,3)	0.00	(4,1,2,3)	2.24
3	(2,2,3,3)	1.00	(3,1,4,1)	2.60
4	(2,3,1,3)	1.66	(5,4,3,2)	2.24
5	(2,3,3,4)	1.41	(2,2,4,4)	2.00
6	(2,3,2,2)	0.87	(4,1,2,4)	2.60
7	(3,3,3,3)	0.00	(2,4,2,3)	1.66

(b) ベクトル $f_2(x)$

期間	労働者 A	労働者 B
1	(2,1,2,1,4,0,0)	(2,1,3,3,0,3,1)
2	(4,0,4,0,4,0,0)	(1,2,1,1,0,2,3)
3	(3,0,3,0,4,0,0)	(2,1,3,2,1,0,0)
4	(3,0,3,0,3,0,0)	(2,2,2,0,4,1,3)
5	(3,0,4,0,4,0,1)	(4,1,3,1,1,1,1)
6	(3,0,3,0,3,0,0)	(1,3,2,2,0,0,3)
7	(4,0,4,0,4,0,0)	(1,3,1,2,0,1,3)

(c) ベクトル $f_3(x)$ ($\ell = 4$ の場合)

期間	労働者 A	労働者 C
1	(6,2,0,0)	(5,3,0,1)
2	(12,0,0,0)	(1,0,3,0)
3	(10,0,0,0)	(2,3,0,0)
4	(9,0,0,0)	(1,1,3,1)
5	(12,0,0,0)	(5,0,2,0)
6	(9,0,0,0)	(5,3,0,0)
7	(12,0,0,0)	(4,2,1,0)

(d) ベクトル $f_4(x)$ ($m = 5$ の場合)

期間	労働者 A	労働者 D
1	(4,2,0,1,1)	(4,2,2,0,0)
2	(8,4,0,0,0)	(1,2,0,0,1)
3	(5,5,1,0,0)	(2,0,3,0,1)
4	(5,4,0,0,1)	(2,2,2,0,0)
5	(8,4,0,0,0)	(0,0,4,0,1)
6	(4,4,1,1,0)	(4,2,3,0,0)
7	(8,4,0,0,0)	(2,4,2,0,0)

表 2: ベクトル $f_3(x)$ と $f_4(x)$ の平均値 (それぞれ 10000 個のベクトルをランダムに生成し、その平均をとったもの)

(a) ベクトル $f_3(x)$

k=9	(4.37, 1.34, 0.39, 0.19)
k=10	(4.33, 1.51, 0.49, 0.28)
k=11	(4.29, 1.66, 0.56, 0.39)
k=12	(4.17, 1.75, 0.65, 0.53)
k=13	(3.98, 1.81, 0.77, 0.68)
k=14	(3.80, 1.85, 0.81, 0.87)

(b) ベクトル $f_4(x)$

k=9	(2.28, 1.50, 0.80, 0.81, 1.25)
k=10	(2.58, 1.64, 0.84, 0.79, 1.03)
k=11	(2.93, 1.76, 0.86, 0.74, 0.83)
k=12	(3.23, 1.84, 0.87, 0.67, 0.65)
k=13	(3.50, 1.87, 0.86, 0.60, 0.49)
k=14	(3.81, 1.85, 0.80, 0.51, 0.36)

3.2 特徴ベクトルの取捨選択

前述のように、ある特徴ベクトルがユーザにとって意味のあるものであるかどうかを判断するための方法として、ここでは以下の 2 点に着目する: 1) ほかの労働者と比べて特殊かどうか、2) ある一定期間以上同じ傾向を示しつつづけているかどうか。

まず前者の方針の実現方法について説明する。あるユーザによって受理されたシフト x を考える。シフト x の中の 1 の個数を k とし、集合 $U (= \{0, 1, \dots\}^{28})$ の中で 1 を k 個含むすべてのベクトルの集合を Z と記す (定義より $|Z| = {}_{28}C_k$ である)。シフト x から得られる特徴ベクトル $f_1(x), f_2(x), f_3(x), f_4(x)$ のそれぞれが他の労働者のベクトルと比べて特殊であるかどうかを判断するための基準として、ここでは各特徴ベクトルの Z に関する平均値とそこからのユークリッド距離を用いる。

たとえば $k = 12$ のとき、ベクトル $f_1(x)$ の平均値 f_1^* は $f_1^* = (3, 3, 3, 3)$ となり、ベクトル $f_2(x)$ の平均値 f_2^* は $f_2^* = (\frac{12}{7}, \frac{12}{7}, \frac{12}{7}, \frac{12}{7}, \frac{12}{7}, \frac{12}{7}, \frac{12}{7})$ となる。また、 $k = 9, 10, 11, 12, 13, 14$ に対する $f_3(x)$ と $f_4(x)$ の平均値は、それぞれ表 2 の (a) と (b) にまとめられている通りである。

提案方式では各特徴ベクトル $f_i(x)$ について、そ

表 3: 表 1(a) の労働者 A の特徴ベクトル間の距離 (表 3 の期間 05 は表 1(a) の労働者 A の期間 05 の特徴ベクトルに対応)

特徴ベクトル間の距離

期間	1	2	3	4	5	6	7
1	—	2.45	2.45	3.00	3.16	3.00	2.45
2	2.45	—	1.41	2.24	1.41	1.73	0.00
3	2.45	1.41	—	2.24	1.41	1.41	1.41
4	3.00	2.24	2.24	—	2.24	1.41	2.24
5	3.16	1.41	1.41	2.24	—	2.24	1.41
6	3.00	1.73	1.41	1.41	2.24	—	1.73
7	2.45	0.00	1.41	2.24	1.41	1.73	—

の平均 f_i^* からある一定の距離 θ_i 以内のものを標準的なベクトルとし、それ以外のものを特殊なベクトルとみなす。たとえば $\theta_1 = 2.5$ とすると、表 1(a) では労働者 B の期間 1, 3, 6 の各ベクトルがそれぞれ特殊であるとみなされる。このことから $f_1(x)$ は、空間的特異性の観点から、労働者 A にとってあまり意味がないが、B にとっては意味のある特徴ベクトルであると考えることができる。

次に後者の方針の実現方法について説明する。あるユーザに関して観測された特徴ベクトルの集合

$$\{f_i(x_1), f_i(x_2), \dots, f_i(x_k)\}$$

を考える。この集合中のそれぞれのベクトルに対して、ユークリッド距離が λ_i 以下であるような (同じ集合中の) ベクトルの個数を求める。もしそのような近傍ベクトルが n_i 個以上存在すれば、その特徴ベクトルはそのユーザにとって同じ傾向を示しているとみなす。またどのベクトルに対しても近傍ベクトルが $(n_i - 1)$ 個以下であれば、その特徴ベクトルはそのユーザにとって非定常的であるとみなす。表 1(a) の労働者 A の特徴ベクトル集合を例に考える。各ベクトル間の距離は表 3 のようになる。例えば $\lambda_1 = 1.5$, $n_2 = 2$ とすると、表 1(a) の労働者 A の特徴ベクトル集合では期間 2, 3, 5, 6, 7 の特徴ベクトルが同じ傾向を示しているとみなされる。このことから $f_1(x)$ は、時間的継続性の観点から、労働者 A にとって意味のある特徴ベクトルであると考えることができる。

4 システムの設計方針

本節では、前述の方針にしたがって現在構築中のプロトタイプシステムの概要について述べる。

4.1 モデル

構築中のシステムはマルチエージェントモデルに基づいて構成されている。前述のように勤務表作成問題は不完全制約充足問題として定式化することが可能である。よって本システムでは与えられた制約をその性質によっていくつかに分類し、そのそれぞれを異なるエージェントに割り当てるという手法をとる。各エージェントはそれぞれ自分に割り当てられた制約の充足を試み、その結果、すべての(あるいはできるだけ多くの)制約を満足するシフト(案)が構成されていくことになる。

さて勤務表作成問題で取り扱われる制約には、大きくわけて労働者に関するものと組織に関するものがある。労働者に関する制約は各労働者にとって固有のものであり、それぞれの労働者に関して独立に取り扱うことができるが、組織に関する制約も各部門の制約に細分化して考えることができる。なおここでいう部門とは、いわゆる組織全体を縦割りして得られる部署のことではなく、特定の労働者集合のことを指している。したがって組織全体もひとつの部門として取り扱われ、また場合によっては、ある労働者が複数の部門に所属することもありうるものとする。

各労働者(laborer)に割り当てられたエージェントを L -エージェント、各部門(department)に割り当てられたエージェントを D -エージェントとそれぞれ呼ぶ。 L -エージェントの集合を $\mathcal{L} = \{L_1, L_2, \dots, L_n\}$ 、 D -エージェントの集合を $\mathcal{D} = \{D_1, D_2, \dots, D_m\}$ であらわす。集合 \mathcal{D} の中には各部門間の競争を解消するための(組織全体に対応する) D -エージェントが含まれているものとする。

システムの概観を図2に示す。各エージェントはそれぞれ個別のデータベースをもつ。 L -エージェントのもつ希望リスト履歴には x_i の系列が保存され、嗜好データベースには (y_i, b_i, t_i) の系列が蓄えられる。また D -エージェントの定常制約リストには、法に基づく制約等、比較的定常的な制約が論理式の形で蓄えられる⁶。

⁶ 例えば、労働基準法第35条(1)「使用者は、労働者に対して、毎週少なくとも1回の休日を与えなければならない。」は

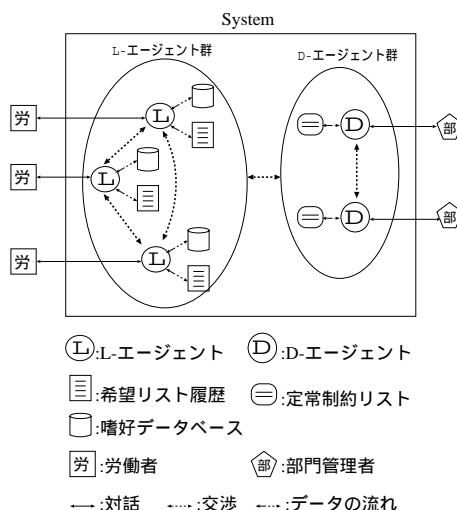


図 2: システムの概観.

4.2 アルゴリズム

勤務表の作成は以下の手順ですすめられる:

Step 1 (勤務表案の提示): 労働者は自分の L -エージェントに対して出欠勤希望リスト x を提示する。 L -エージェントは x を希望リスト履歴に登録したのち、 x に基づいて労働者の勤務シフト案を作成し、それを D -エージェントたちに提示する。

Step 2 (中間勤務表の提示): D -エージェントは、シフト案に対して必要な修正を加えたのち、それを中間勤務表として L -エージェントたちに提示する(修正は、雇用者側から提示される必要人員リストと定常制約リストを参照することでおこなわれる)。

Step 3: (中間勤務表の修正): 各 L -エージェントは中間勤務表に対して、あらかじめ獲得した嗜好に沿ったシフトの修正を試みる(このとき必要に応じてエージェント間で交渉がおこなわれる)。交渉終了後、 L -エージェントは修正後の中間勤務表を勤務シフト案として各労働者に提示する。

Step 4: (労働者による確認): 労働者は提示された勤務表案を確認し、もし修正したい箇所があれば L -エージェントに対して修正要求をだす。もしどの労働者からも修正要求がだされなければ勤務表を確定し、処理を終了する。そうでなければ、修正要求を受けた L -エージェントはそれを出勤希望リストに反映させるとともに、その労働者の嗜好を保存し

$$\bigwedge_{i=0}^3 \bigvee_{j=0}^6 x[j+7i] \text{ と定式化される。}$$



図 3: 出欠勤希望リスト入力インタフェース.

である嗜好データベースにも反映させ、Step 2 にもどる。

4.3 ユーザインタフェース

システムのインタフェースはブラウザを用いて設計されており、インターネットに接続可能な PC や携帯電話による入力が可能となっている。出欠勤希望リストのための入力インタフェースを図 3 に示す。各ユーザからの入力は出欠勤の希望をチェックボックスをクリックすることによりおこなわれる。入力終了後、システムにより処理がおこなわれ L -エージェントから図 4 のような画面でシフト案が提示される。ユーザはシフト案を確認し、応答をおこなう。応答にはシフト案の受け入れと修正要求の 2 種類があり、修正要求は修正したい日のチェックボックスをクリックすることにより、システムに対して送られる。

5 おわりに

本稿では、ユーザからの非明示的な要求の自動獲得手法の提案を行った。提案手法はビットベクトルから特徴ベクトルへの射影を行い、特徴ベクトル空間上で非明示的な要求を獲得するというものであった。本稿では提案する自動獲得手法の具体例として勤務表作成問題を取りあげ、各段階での具体的な手法の検討を行った。しかしユーザの意図しない要求

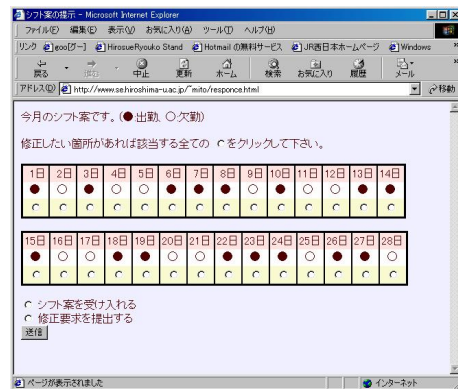


図 4: シフト案提示インタフェース.

をあつかっているため、手法の客観的な評価が難しく、システム全体の評価はまだ完了していない。また今回検討したのはシステムのインタフェース部分のみであり、エージェント間の交渉アルゴリズムやそのためのアーキテクチャについては、今後の検討が必要である。

参考文献

- [1] A. Jan, M. Yamamoto, and A. Ohuchi. Evolutionary algorithms for nurse scheduling problem. In *2000 Congress on Evolutionary Computation*, 1:pp.196–203, 2000.
- [2] S. Abdennadher and H. Schlenker. Interdependent interactive constraint based nurse scheduler. *The Practical Application of Constraint Technologies and Logic Programming, PACLP99*, 1999.
- [3] B.M.W. Cheng, J.H.M. Lee, and J.C.K. Wu. A nurse rostering system using constraint programming and redundant modeling, 1997.
- [4] E. Burke, P.D. Causmaecker, and G.V. Berghe. A hybrid tabu search algorithm for the nurse rostering problem. *SEAL'98, LNCS 1585*:pp.187–194, 1999.
- [5] F.P. プレパラータ/M.I. シェーモス 著 and 浅野孝夫/浅野 哲夫 訳. 計算幾何学入門. 総研出版, 1992.