



Title	連続離散混合領域におけるファジィ制約充足問題とその反復改善型解法
Author(s)	須藤, 康裕; 栗原, 正仁; 三田村, 保
Citation	知能と情報 : 日本知能情報ファジィ学会誌, 17(3), 367-375
Issue Date	2005-06
Doc URL	https://hdl.handle.net/2115/14590
Rights(URL)	http://www.j-soft.org/
Type	journal article
File Information	sudoy2005soft-final.pdf



論文

標題 : 連続離散混合領域におけるファジィ制約充足問題とその反復改善型解法

Hybrid Domain Fuzzy Constraint Satisfaction Problems and Their Iterative Improvement Algorithms

要約

ファジィ制約充足問題は人工知能の基礎戦術として知られる制約充足問題 (CSP:Constraint Satisfaction Problem) を拡張したモデルである。CSP およびファジィCSP において、変数の領域は有限離散集合として定義される。しかしながら数理計画法に代表されるように、現実には変数領域として実数値をとる場合も多く、現実世界の問題を CSP として定式化する場合に制限となることがある。

本論文では、より自由な問題表現を可能にするため、変数領域が連続領域および離散領域を併せ持つ混合領域ファジィCSP (HDFCSP:Hybrid Domain Fuzzy CSP) を提案し、反復改善に基づくその解法 Spread Repair アルゴリズムを提案する。また実験により、連続領域を単純に離散化して得られる従来のファジィCSP アルゴリズムと比較し、解の質および計算時間の面で HDFCSP および Spread Repair アルゴリズムが優れていることを示す。

キーワード

ファジィCSP, 最適化, 混合領域 FCSP, 連続領域, 反復改善, Spread Repair

Abstract

A Fuzzy Constraint Satisfaction Problem(FCSP) is an extension of the classical CSP, a powerful tool for modeling various problems based on constraints among variables. In traditional CSPs and FCSPs, values for the variables are chosen from discrete domains. However, this is often inconvenient when one wants to express real world problems. In this paper, we propose a new model that allows the variables to have a mixture of discrete and continuous domains. We show that this model, called HDFCSP(Hybrid Domain FCSP), can be solved by a new algorithm SpreadRepair, an extension of the well-known iterative improvement algorithms. Experimental results on some test problems show that the algorithm has an ability of finding practical approximate solutions with high probability in a computation time much shorter than the traditional algorithms for the discrete-domain FCSPs.

Keywords

Fuzzy CSP, optimization, Hybrid domain FCSP, iterative improvement, Spread Repair

1. はじめに

制約充足問題 (CSP : Constraint Satisfaction Problem)[1, 2] は変数間の制約を全て満たす変数への割当てを解とする組合せ問題であり、一般に NP 完全問題である。これまで、現実の世界における問題を CSP として定式化し、さまざまな場面で成功をおさめて来た。人工知能の分野においてはこのような問題の定式化と、その問題解決器の構築が研究対象の一つであるが、CSP はあまりにも単純なモデルのため現実問題の定式化において様々な制限が設けられることとなる。これに対し CSP を拡張したモデルについて多くの研究がある。ファジィ制約充足問題 (FCSP:Fuzzy CSP) も CSP の拡張モデルの一種であり、制約にファジィ関係を導入し、その充足度に曖昧さを持たせることにより、不完全に充足される解を求め、現実の問題解決への有用な情報提供を可能としている [3, 4, 5].

CSP およびファジィ CSP において、変数の領域は有限離散集合として定義される。このこと自体は欠点ではないが、離散領域から値をとる組合せ的要因により問題は格段に難しい。最近の人工知能の研究では、このような組合せ系における問題についてヒューリスティックを用いた反復改善手法 [6] を用いることが多い。ところが線形計画問題を含む数理計画法に代表されるように、現実には変域として実数値をとる場合も多く、両者は別の観点から研究が進められてきた。

本論文ではより自由な問題表現を可能にするため、ファジィ CSP の変数領域に連続領域および離散領域を併せ持つ、前述の両方の性質を持つ問題 (混合領域ファジィ CSP (HDFCSP:Hybrid Domain Fuzzy CSP)) および反復改善に基づくその解法を提案する。

本論文の構成は以下の通りである。2 章では制約充足問題について述べる。3 章ではファジィ制約充足問題を拡張した HDFCSP を提案する。4 章では提案されたモデルの解法として Spread Repair アルゴリズムを提案する。5 章では実験結果と考察を述べる。6 章では結論を述べる。

2. 制約充足問題

CSP は現実世界における様々な問題を定式化可能であり、これまでスケジューリング、プランニング等の応用に対して実際に効果を上げてきた。有名なものではハッブル宇宙望遠鏡の観測スケジュール [7] やフランスの資源探査衛星の撮影スケジュールの決定 [8] などがある。また、携帯電話の周波数割当て [9] においても成功をおさめている。

CSP は、変数の集合 $X = \{x_i\}_{i=1}^n$ とその値域となる領域の集合 $D = \{D_i\}_{i=1}^n$ および任意の変数間に設けられる制約の集合 $C = \{C_k\}_{k=1}^r$ の組として定義される。制約 C_k は X の部分集合 $S_k (S_k \subseteq X)$ 上の関係 R_k であり、 S_k に含まれる各変数が同時に取り得る値の組合せを表している。すなわち、 $S_k = \{x_{k_1}, \dots, x_{k_w}\}$ ならば $R_k \subseteq D_{k_1} \times \dots \times D_{k_w}$ である。 S_k を R_k の範囲 (scope) という。また、ここで $w = 2$ であるとき制約は 2 項関係であるといひ、 $w = 1$ の場合には単項関係、 $w \geq 3$ の場合をとくに多項関係という。例えば変数 x_1 は常に x_2 よりも大きい値でなければならないという制約 C_k があれば、 C_k の範囲は $S_k = \{x_1, x_2\}$ となり、それぞれの変数は (x_1, x_2) が

$$R_k = \{(x_1, x_2) | x_1 > x_2, x_1 \in D_1, x_2 \in D_2\}$$

の元となるように選ぶ。

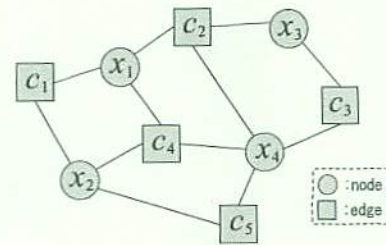


図 1: An example of a hypergraph

CSP は無向グラフによって表現することが可能であり、変数がノード、制約がエッジにそれぞれ対応付けられる。このとき、通常のエッジでは 2 項関係しか表現できないが、ハイパーエッジ [1] を使うことにより多項制約を表すことが可能であ

る (図 1). ハイパーエッジとは 2 個または 3 個以上のノードを結ぶエッジであり, このようなエッジを持つグラフをハイパーグラフという. このグラフを用いることにより多項制約における CSP を表現することができ, さらにグラフ理論における隣接や近傍などの用語を CSP の変数および制約などに対応付けることが可能であり, 以下本論文でも使用する.

このような離散系の問題においては効率的で完全な解法は一般に存在せず, もし, しらみつぶしに解を探索すれば, 計算時間が問題の規模に応じて指数的に増大することは避けられない. しかしながら様々な方法を駆使し, 多くの場合に現実的な時間で近似解を得られるよう研究が進められてきた. CSP の解法として大きく 2 つ, 部分的な変数割当ての逐次拡張に基づく厳密解法と, 完全な変数割当ての逐次修復に基づく近似解法とがある. 近年ではヒューリスティックな評価関数を用い, 変数を逐次変更する手法 [6] について研究が行われ, これにより 100 万クイーン問題を数十ステップで解を得る [10] などの成果が上げられている. 本論文で提案するモデルにおいても, この手法を用いて近似解を高速に探索する.

3. ファジィ制約充足問題の拡張

本論文では離散連続領域の導入により, FCSP を拡張する. 制約の表現として連続なメンバーシップ関数を用いることにより, 部分的に連続的最適化手法を導入することが可能になる.

3.1 ファジィ制約充足問題

FCSP は, 通常の CSP の制約にファジィ関係を導入したモデルであり, 最適化問題の一種である. ファジィ制約 C_k はファジィ関係 R_k に対応付けられ, そのメンバーシップ関数 μR_k は

$$\mu R_k : \prod_{x_i \in S_k} D_i \rightarrow [0, 1] \quad \dots (1)$$

の形式で与えられる. すなわち, 制約 C_k の範囲 S_k の割当て v_{S_k} に対するメンバーシップ値 $\mu R_k(v_{S_k})$ は, $[0, 1]$ の実数値をとり, それが制約の充足度を示す.

制約 C_k と C_l のファジィ論理積 $C_k \wedge C_l$ は, $S_k \cup S_l$ を範囲とするファジィ関係 $R_k \cap R_l$ であり, そのメンバーシップ関数を

$$\mu R_k \cap R_l(v) = \min(\mu R_k(v[S_k]), \mu R_l(v[S_l])) \quad \dots (2)$$

で定義する. ただし, $v[S]$ は v の割当てを範囲 S に含まれる変数に制限したもの (射影) である. 同様に全ての制約のファジィ論理積すなわち CSP 全体の充足度を以下のように定義する.

$$\mu \bigcap_{k=1}^r R_k(v) = \min_{1 \leq k \leq r} (\mu R_k(v[S_k])) \quad \dots (3)$$

つまり, 全ての制約の中で, 最も制約を満たしていない制約の充足の度合いを CSP の充足度 C_{min} とする.

$$C_{min}(v) = \min_{1 \leq k \leq r} (\mu R_k(v[S_k])) \quad \dots (4)$$

ここで $C_{min}(v) > 0$ のとき, v をファジィ CSP の解という. また, 最適解とは最大の充足度を与える解をいう. したがって, FCSP を解くことは, 充足度が最低の制約 (C^* とする) の充足度を最大化するような変数への値の割り当て v を求める最適化問題となり, 以下の式がその最適解の目的関数値となる.

$$\max_v (\min_{1 \leq k \leq r} (\mu R_k(v[S_k]))) \quad \dots (5)$$

ただし, (5) 式のみを考慮した場合, C^* 以外の制約は改善されることがない. 現実の応用においては, C^* 以外の制約違反もできるだけ改善すべきであり, 4 節でその手法について述べる.

3.2 混合領域ファジィ制約充足問題

通常, CSP の変数領域は有限離散集合であり, その要素を列挙することができるので, 全ての組み合わせを探索する木を生成することができる. しかしながら変数領域が,

- i. 要素の多い集合である
- ii. 連続値である
- iii. 連続値と離散値をとる場合がある

というような状況においては、従来の CSP および FCSP では解が得難かった。本論文では上記のような場合でも CSP として処理可能にするため、変数領域に離散および連続領域を導入した混合領域ファジィ CSP (HDFCSP) を提案する。

HDFCSP は FCSP を拡張したモデルであり、領域の定義以外は FCSP と同じである。HDFCSP では、変数の各領域を閉区間の和集合として、

$$D_i = \bigcup_{j=1}^m [l_j, u_j] \quad \dots (6)$$

であるとする。\$[l_j, u_j]\$ は下限を \$l_j\$, 上限を \$u_j\$ とする閉区間、すなわち \$\{x | l_j \le x \le u_j\}\$ という領域を表す。また、メンバーシップ関数は各領域で連続であるとする。ここで \$l_j = u_j\$ である場合、その区間は単一の実数値を取るの、離散的な領域とみなすことができる。全ての \$j\$ において、\$l_j = u_j\$ となる場合、それは通常の CSP もしくは FCSP の領域と同じであるので、HDFCSP は FCSP を包含している。また、全ての変数領域が唯一の連続領域を持ち、なおかつ制約が連続なメンバーシップ関数によって表現されている場合は、通常の非線形計画問題とみなすことができる。

(6) 式の定義により、制約充足問題および数理計画問題の両方の性質をもつ問題を扱うことが可能になり、これまでとは異なった、より柔軟な定式化が可能になる。

4. HDFCSP の解法

FCSP に関する研究では、分枝限定法 [4] やニューラルネット [11, 12] を用いて解が得られることが示されている。HDFCSP における変数領域は連続領域を含むので、探索木の生成は不可能であるが、一つの方法として、領域を離散化することによって組み合わせを有限通りにすることは可能である。しかしながら領域が広大である場合に、必要以上の精度で離散化することは探索空間を広げることになり、一方精度が足りなければより良い解を発見できない。また数理計画的な手法は、制約を表現するメンバーシップ関数の形状を限定することによって実用的な解法を得ているが、表現

の自由度をなるべく犠牲にたくないという人工知能的な価値観にとっては負の側面がある。これに対し、本論文では単純に領域の離散化を行うのではなく、有望な候補のみを高速に採択する手法を提案する。

4.1 Spread Repair アルゴリズム

HDFCSP の解法としてアルゴリズム Spread Repair を示す。反復改善を行う近似解法であるため、得られる解が大域的最適解である保証はない。局所改善的に CSP を解く場合、何らかのヒューリスティックを用いた評価関数を使用する。本手法では (5) 式を単純な評価関数として用い、一度に 1 変数の値のみ変更するものとする。Spread Repair アルゴリズムは、近傍状態への推移を反復的に行う (図 2)。ここで、\$C^*\$ が一度の変更によって改善できない状態を準局所最適ということにする。また、すべての制約が平均的な充足度

$$C_{ave} = \frac{1}{r} \sum_{k=1}^r \mu R_k(v[S_k]) \quad \dots (7)$$

を下げることなく一度の変更によって改善できない状態を局所最適とし、その時の割り当てを局所最適解とする。アルゴリズムは主に以下の 3 つの部分から構成され、メインプログラムから SpreadRepair() を呼出す。

SpreadRepair(): 常に \$C^*\$ に対して改善を試みる。最初 \$Y\$ は \$X\$ と同じ全変数のリストであり、\$V\$ は \$C^*\$ の範囲である。

RepairOneOf(V): 変数の集合 \$V\$ のうち、最も \$C^*\$ を改善できる変数を選択し、準局所最適に陥った場合に変更の対象となる制約を隣接する制約に移行し、再帰呼び出しを行う。\$Y\$ は未調査の変数の集合であり、\$Y\$ が空集合であれば局所最適。

Repair(i): \$x_i\$ に対し、最良の割り当てを計算する。getMaximalValue() および getIntersection() で極大値および交点を与える割り当てを全て求め、\$T\$ に加える。

Spread Repair アルゴリズムでは \$C^*\$ に対して影響力が強い周辺の変数への割り当てが変更されることにより準局所最適状態から抜け出す可能性を生じ

Spread Repair algorithm

global variables
input
 C_1, C_2, \dots, C_r : constraints
output
 x_1, x_2, \dots, x_n : assignments to variables
 C^* : constraint with minimum satisfaction degree
 Q : degree of satisfaction of C^*
 X : the set $\{1, 2, \dots, n\}$ of all the variables
 Y : a subset of X

procedure SpreadRepair()

```

initialize  $x_1, \dots, x_n$  to random values
do
  compute  $C^*$  and  $Q$ 
   $Y \leftarrow X$ 
   $V \leftarrow$  the scope of  $C^*$ 
  while RepairOneOf( $V$ )

```

function RepairOneOf(V)

```

//return true if a variable is repaired.
 $Y \leftarrow Y \setminus V$ 
for each variable  $i \in V$  do
  ( $v_i, r_i$ )  $\leftarrow$  Repair( $i$ )
let  $r_k = \max_i \{r_i | i \in V\}$ 
if  $r_k > 0$  then
   $x_k \leftarrow v_k$ 
  return true
else if  $Y = \phi$  then
  return false
else //spread
   $V' \leftarrow \phi$ 
  for each variable  $i \in V$  do
     $V' \leftarrow V' \cup \{ \text{variable } j \in Y | j \text{ is adjacent to } i \}$ 
  return RepairOneOf( $V'$ )
end if

```

function Repair(i)

```

//return the locally optimal assignment to  $x_i$ 
 $T \leftarrow \phi$ 
 $C \leftarrow$  a set of constraints relating to  $x_i$ 
for each constraint  $C_k \in C$ 
   $T \leftarrow T \cup (\text{getMaximalValue}(i, C_k) \cap D_i)$ 
  for each constraint  $C_l \in C$ 
     $T \leftarrow T \cup (\text{getIntersection}(i, C_k, C_l) \cap D_i)$ 
end for
 $T \leftarrow T \cup$  (the lower and upper bounds of  $D_i$ )
return ( $v_i, r_i$ ) where  $v_i$  is the element of  $T$  giving the
maximum satisfaction degree and  $r_i$  is the amount of
improvements

```

図 2: Spread Repair algorithm

させる。また、再帰的に改善の対象となる制約を周辺に広げていくので、 C^* 以外の制約を改善することになり、平均的な制約充足度の値も同時に増

大する(ただし山登りの手法であるため、このことは保証されないが、経験的には増大することがわかっている)。現実問題として平均的な充足度を高めることは重要であり [3, 4, 5], この利点は Spread Repair アルゴリズムの大きな特徴の一つである。

”Spread Repair”は、改善の対象が周囲に広がっていく様子进行している(図3)。アルゴリズムは全ての変数に対して改善が行えなくなった時点で停止し、そのときの割当て V を局所最適解として得る。局所最適状態から抜け出す方法については様々な研究がされており、Simulated Annealing(焼きなまし法)や、遺伝的アルゴリズムにおける突然変異がこれに該当する。また、制約の優先度を変化させる様々な手法が研究されている。

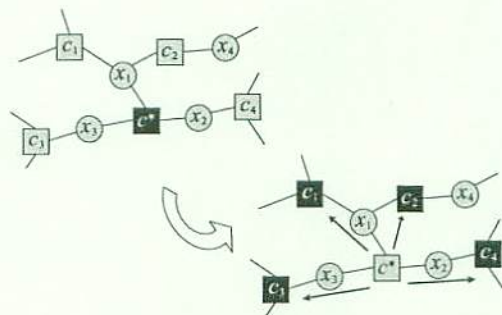


図 3: Spreading target

4.2 変更候補値の算出および選択

C^* を改善する場合、 C^* が w 項制約であるとすると、その範囲 S^* は w 個の要素を持ち、その割当てが C^* の充足度に影響をもつ。Spread Repair アルゴリズムでは唯一の変数への割当てを変更するが、このとき最も C^* が改善されるように、変数とその割当てを選択する。

本論文では変更後の割当てについて、単純な離散化を行った後逐一充足度の計算を行うよりも、高速かつ高精度に求める手法を提案する。変更する変数については、範囲 S^* の要素である w 個のそれぞれに対して変更後の充足度を調べればよいので、計算コストは w に対して線形である。しかしなが

ら、その時点での変数への最適な割当てを求めることは、メンバーシップ関数に対して最大値を与えるような割当てを探索することになり、関数の形状によっては多くの計算時間を要し、さらには最大値を得られない場合すらある。また S^* の要素である変数 $\{x_{ij}\}_{j=1}^w$ も、他の制約の範囲に属する場合があるので、 x_{ij} の割当て v_{ij} を変更するときは複数のメンバーシップ関数を同時に考慮する必要がある (図4)。ここで x_{ij} を範囲に含む制約の集合を $\{C_{k_h}\}_{h=1}^t$ ($\exists C^*$) とすると、そのメンバーシップ関数 $\mu R_{k_h}(v_{S_{k_h}})$ は範囲 S_{k_h} への割当て $v_{S_{k_h}}$ の関数であるが、 S_{k_h} の要素のうち唯一 x_{ij} のみを変更されることを仮定するので、メンバーシップ値は v_{ij} に対して一意に定まり、

$$\mu R_{k_h}(v_{ij}) = \mu R_{k_h}(v_{S_{k_h}})$$

と略記することにする。

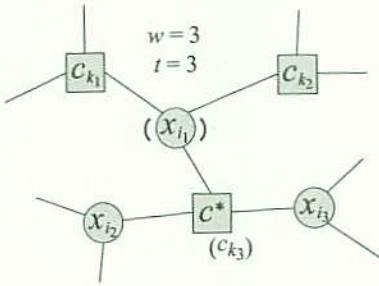


図4: example : Changing x_{i1}

任意のメンバーシップ関数同士のファジイ論理積による、新しいメンバーシップ関数の全体の形状を把握するのは困難である。しかし、必要なのは最大値 $\max(\mu R_{k_h}(v_{ij}))$ を与える割当て v_{ij} だけで、関数全体の形状を厳密に求める必要はない。ここで $\max(\mu R_{k_h}(v_{ij}))$ の出現する可能性のある位置を検討する。閉区間 $[l, u]$ において、連続な任意の関数 f の最大値は $f(l)$, $f(u)$, または f の極大値のいずれかである。したがって、それぞれのメンバーシップ関数がどのような形状であったとしても、最大値は l か u , それぞれのメンバーシップ関数の極大値、あるいはある2つのメンバーシップ

関数の交点に存在する (図5)。Spread Repair アルゴリズムでは単純に離散化するのではなく、このような候補点から次の割当てを探索する。

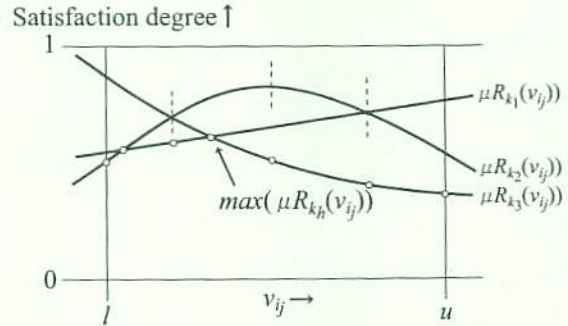


図5: $\max(\mu R_{k_h}(v_{ij}))$

このような評価値を用いた反復改善では、変更後の充足度が同一となる変更が複数存在する可能性がある。本論文ではこれを単に「非決定的に選択する」ととどめ、具体的には指定しないことにする。

4.3 交点と極大値を求める算法

交点を求めることは、1変数の方程式 (一般には非線形) の数値解を求めることに対応する。4次以下の単純な代数方程式であれば公式を用いて瞬時に数値解を計算することが可能であるが、高次代数方程式や指数関数などを含む超越方程式の数値解を求めるには、反復を用いて近似解を求めることになる。導関数が利用できるのであれば、ニュートン法などの高速な方法もあるが、制約を表現するメンバーシップ関数を限定することは、問題表現の柔軟さを欠くこととなる。これらの操作における最新の研究では、Brent のアルゴリズム [13] が知られている。この手法は収束の速さと確実性の両方を持ち、超1次収束する手法を監視しながら用い、超1次収束しないようであれば1次収束に切り替える。逆2次補間を用いたアルゴリズムで、導関数が使用できない場合に非常に有効であり、しかも収束が保証されている。

極大値を求める手法においても、Brent の手法が有効である。呼び方は同じであるが内容は異なる。

り、極大値を求めるアルゴリズムは2階導関数が連続のときに放物線補間を用い、そうでないときには黄金分割法を用いる。導関数を利用できない場合でも後者の方法で収束は約束されている。Spread Repair アルゴリズムにもこの手法を用いれば、最大および交点を求める操作に関して複雑なメンバーシップ関数であっても高速に処理することが期待できる。

5. 実験結果と考察

HDFCSP の有効性と、Spread Repair アルゴリズムの効率について実験を行う。CSP の例題としてグラフ彩色問題がある。平面上のどのような地図も4色で必ず塗り分けることが可能であることが証明されているが、グラフ彩色問題はその問題を平面グラフとは限らない一般の無向グラフに置き換えたものである。文献 [14] において、ファジィグラフ彩色問題を扱っている。変数領域として4色用意し、それぞれの色間に制約として主観的に決定した色相の異なりを用意し、ファジィ行列を用いて隣接した2色間の選好性を充足度として表現している。HDFCSP においては、連続領域から値を取る状況について実験するため、新たに問題を設定した。

実験に用いた問題は、グラフ彩色問題の一種とみなすことができる。ただし色空間を連続領域として表現するにあたり、問題を単純にするために、白から黒に至るグレイスケール値を用い、その明度の差を制約の充足度とし、それに伴うメンバーシップ関数を用意した。実験で構築したシステムでは、一般によく使用される形状のメンバーシップ関数をいくつか組み込み、ユーザがプログラムを書き換えなくても解を得ることができるようにした。また、ユーザが自由なメンバーシップ関数の記述を望む場合も考慮し、メンバーシップ関数のみを定義すれば交点および極大値を求めることが可能である。したがって、このようなシステムを用いれば様々なメンバーシップ関数を持つ問題に対しても、Spread Repair アルゴリズムを実行することが可能である。

実験では比較の対象として、Spread Repair アルゴリズム (SR) の他に、離散化を行った後単純な山登り法を用いたもの (DS)、山登り法に Spread の手法を取り入れたもの (DS+S) を用いた。それぞれの手法はできるだけ正確に比較するため、アルゴリズムの主となる部分を除いてほとんど同一のプログラムコードを使用し、全く同じ問題を同一の初期値から反復改善を行った。

5.1 グラフ彩色問題

一般に組合せ問題においては一部のパラメータで問題の性質が急激に変化する場合がある。とくに、グラフ彩色問題ではノード数とリンク数の関係によって突然問題が難しくなる領域が存在することが知られている [15]。それについて考慮するため、実験はノード数とリンク数の比率を変化させて行う。グラフはランダムに生成し、変数領域も連続領域および離散領域をランダムで混在させる。

同程度の計算時間で得られた解の質を比較するために、最初に得られた局所解の充足度 C_{min} およびそのときの平均充足度 C_{ave} の分布を示す (図 6)。

グラフはそれぞれ、ノード数 30 に対してリンク数を完全グラフの 10%, 20%, 30%, 40% と変化させた場合である。離散化精度は計算時間が実測で Spread Repair と近くなるよう、 2^{-10} に設定した。全ての結果において、 C_{min} および C_{ave} 共に SR は DS より解の質が高い。また、DS+S も、DS と比較して優れている。この比較から、Spread によって準局所解から脱出することが有効であると確認できる。SR と DS+S とでは、それほど差は見られない。これは無限の精度で離散化した手法と、SR が理論上解の質が同じであるためである。若干みられる差は、同一の充足度が得られる変更候補値が複数存在する場合に、いずれが選択されるか不定であるために生じる。

論文 [14] においてファジィグラフ彩色問題を、アメリカ合衆国本土の州の色分けでテストしている。本論文では連続領域を扱えるモデルを提唱しているため、前述の実験と同じ設定で合衆国のネッ

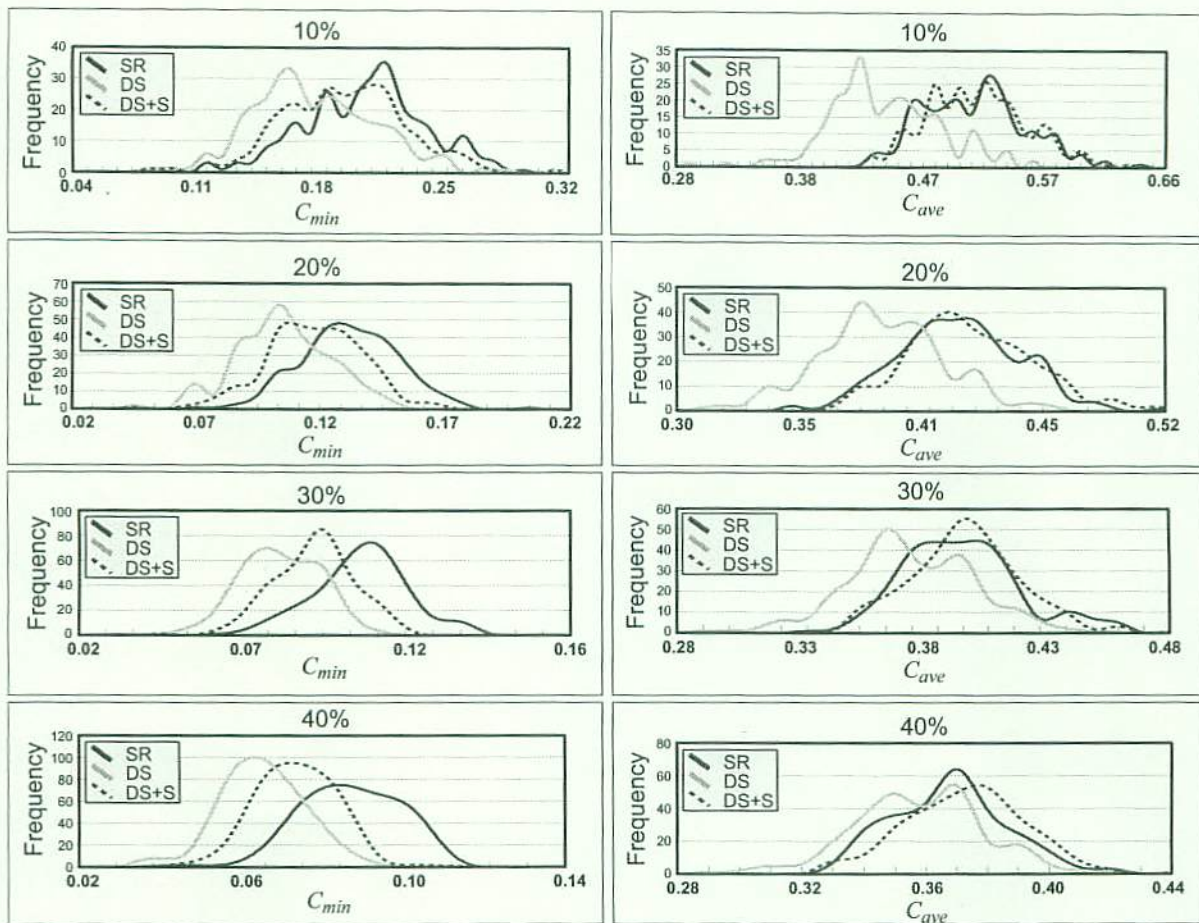


図 6: Results on random HDFCSP

トワークを作成し、HDFCSPとして定式化した。各州の初期値はランダムで決定し、Spread Repair アルゴリズムを用いて最適化を行った結果を図 7 に示す。

図 7 は一つの解であり、充足度 C_{min} は約 0.26 である。また、複数回試行した結果の平均もほぼ同様であった。理論上最適解の充足度は $1/3$ (4 色 0, $1/3$, $2/3$, 1 で塗り分け) であるので、やや最適値には及ばないものの約 79% の近似度が達成されている。また、 C_{ave} の平均も約 0.58 となり、全体的な改善も行われていることがわかる。

5.2 離散化による計算時間

図 8 は離散化の精度を変化させた場合の計算時間の推移である。グラフ彩色問題を離散化し、



図 7: Fuzzy USA map coloring

DS+S でのアルゴリズムが停止するまでの平均時間を SR と比較している。高精度で離散化すると組合せ爆発を引き起こし、指数的に計算時間が増

大することは避けられない。ところがSRでは計算時間は離散化精度と無関係である。問題の規模によって多少差が見られるが、 2^{-10} 程度より粗い離散化においてはSRよりもDS+Sの方が計算時間は短かく、それより高精度の離散化ではSRの方が効率が良いことがわかる(ただし、離散化が粗すぎると同じ計算時間でも解の質はSRの方が上である)。Spread Repairの精度は、実験においては倍精度の浮動小数点型を用いているので、単純に 2^{-52} 程度の精度が得られており、仮にこの精度で領域を離散化するとすると天文学的な計算時間を要する。

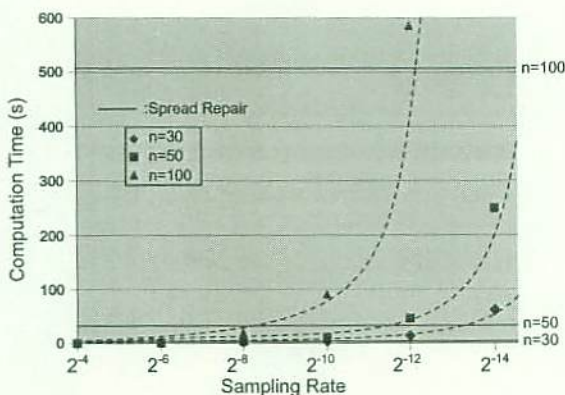


図 8: Computation time

この実験においてはメンバーシップ関数を線形近似することによって交点および極大値を与える割当てを代数的に高速かつ高精度で求めているので、このような結果が得られた。ただしメンバーシップ関数が複雑な場合には4.3節で述べたBrentのアルゴリズムを使用するなどして交点を求めることになり、効率がやや落ちることが予想される。それでも、計算時間が指数的に増大することはないので、離散化を行うよりは遥かに効率が良いといえる。また、問題の性質による若干の増減は予想できるが、それは今後の課題とする。

6. おわりに

本論文ではファジィCSPの変数領域を連続化したHDFCSPと、その解法について述べ、典型的

な問題を用いた実験によりアルゴリズムの性能を検証した。制約違反の改善において、得られた解の質は C_{min} および C_{ave} 共に単純な離散化を行って得られたFCSPを、山登り法を用いて計算した場合と比較して優れていることが示された。また特に計算時間において、高精度の離散化によって計算した通常のFCSPと比較して効率が良いことが示された。実験に用いた例題の場合、約 2^{-10} 以上の精度が必要な場合にはSpread Repairアルゴリズムを用い、それ以下の精度でよい場合には単純な離散化+Spreadの組合せアルゴリズムがよいという結論が得られた。

参考文献

- [1] Dechter, R. : Constraint Processing, Morgan Kaufmann(2003)
- [2] 西田豊明 : 人工知能の基礎, 丸善株式会社 (1999)
- [3] Ruttkay, Zs. : Fuzzy constraint satisfaction, *Proceedings of 3rd IEEE Intern. Conf. on Fuzzy Systems*, Vol3, 1263-1268(1994)
- [4] Meseguer, P. and Larrosa, J. : Solving fuzzy constraint satisfaction problems, *Proceedings of 6th IEEE Intern. Conf. on Fuzzy Systems*, Vol3, 1233-1238(1997)
- [5] Kanada, Y. : Fuzzy Constraint Satisfaction Using CCM - A Local Information Based Computation Model, *Proceedings of 4th IEEE Intern. Conf. on Fuzzy Systems*, Vol4, 2319-2326(1995)
- [6] Minton, S., Johnston, M.D., Philips, A.B., Laird, P. : Minimizing conflicts: a heuristic repair method for constraint satisfaction and scheduling problems, *Artificial Intelligence* 58, 161-205(1992)

- [7] Fox, M.S. : Constraint-Directed Search: A Case Study of Job-Shop Scheduling, Morgan Kaufmann, San Mateo, CA(1987)
- [8] Badie, C., Verfile, G. : OSCAR ou Comment Planifier Intelligemment des Missions Spatiales, *Proceedings of the 9th International Avignon Workshop* (1989)
- [9] Marathe, M.V., Breu, h., Hunt, B., Ravi, S.S., Rosenkrantz, D.J. : Simple Heuristics for Unit Disk Graph, *networks*, Vol.25, pp.59-68(1995)
- [10] Minton, S., Johnston, M.D., Philips, A.B., Laird, P. : Solving Large-Scale Constraint Satisfaction and Scheduling Problems Using a Heuristic Repair Method, *Proceedings of AAAI-90*, Vol.1, pp.17-24(1990)
- [11] Wong, J.H.Y. and Leung, H. : Extending GENET to Solve Fuzzy Constraint Satisfaction Problems, *AAAI Constraint Satisfaction Problems*, 380-385(1998)
- [12] Adorf, H.M., Johnston, M.D. : A discrete stochastic neural networks algorithm for constraint satisfaction problems, *Proceedings International Joint Conference on Neural Networks*, CA(1990)
- [13] William, H.P., Brain, P.F., Saul, A.T., William, T.V.: NUMERICAL RECIPES in C, Cambridge University Press(1988)
- [14] Takefuji, Y. : *Neural Network Parallel Processing*, Kluwer Academic Publishers (1992)
- [15] Hogg, E., Bernardo, A., Huberman, C., Williams. : Phase transitions and search problems, *Artificial Intelligence*, Vol. 81, Issues1-2, pp. 1-15(1996)

[問い合わせ先]

〒 060-0814 札幌市北区北 14 条西 9 丁目
 北海道大学 大学院情報科学研究科 コンピュータサイエンス専攻

栗原 正仁

TEL : 011-706-6855