プロダクション規則と局所評価関数による最適化の方法とその計算過程におけるマクロなふるまい

プロダクション規則と局所評価関数による制約充足問題の解法

新情報処理開発機構

金田　泰

目次

まえおき

- ■ **計算モデル CCM (化学的キャスティング・モデル) を提案している．**
  - ◆ 現実世界の問題をとくための自己組織的計算をめざす．
  - ◆ 局所的に計算される**評価関数**が導入された**非決定的** (確率的) なプロダクション・システムにもとづくモデル．
- ■ **CCM にもとづく制約充足問題の解法をしめす．**
  - ◆ 多数の局所的な記号的制約を充足する問題の解法．
  - ◆ 最適化や制約充足は CCM が本来めざす計算ではないが，そこへの第 1 歩．
  - ◆ これまでに，CCM により TSP などの最適化問題も実験した．

計算モデル CCM

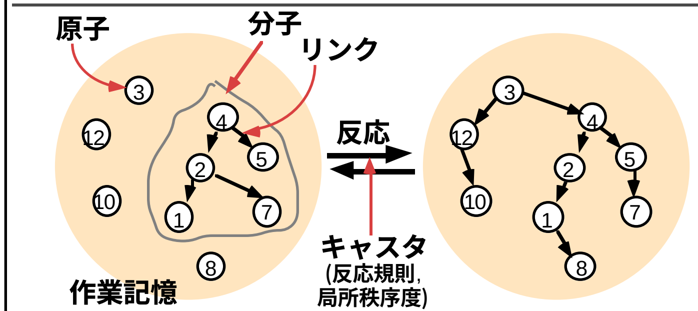
- ■ **計算モデル CCM (化学的キャスティング・モデル) を考案．**
  - ◆ CCM は化学反応系とのアナロジにもとづく計算モデルであり，プロダクション・システムにもとづいている．．
  - ◆ “キャスティング” は “プログラミング” や “計算” にかわることば．
  - ◆ 以前は CPM または MCP (化学的プログラミング・モデル) とよんでいた．
- ■ **CCM では計算を秩序化 (“自己組織化”) とみなす / をめざす．**
  - ◆ 局所的情報にもとづく計算によって大域的な “秩序的構造” をつくることをめざす．

プロダクション規則と局所評価関数による最適化の方法とその計算過程におけるマクロなふるまい

CCM におけるシステムの構成要素と動作 — 1

- **作業記憶 — オブジェクトのあつまり**
- **オブジェクト (WME)**
  - 原子：データの単位．原子は内部状態をもつ．
  - 分子：原子がリンクによって結合されたもの．
- **キャスタ (プログラム)**
  - 反応規則：系の局所的な変化のしかたをきめる規則 (前向き推論によるプロダクション規則)．
    - 化学反応式に相当し，双方向に動作しうる．
    - 反応順序は非決定的 (確率的) — 複数の反応がおこりうるとき．
  - 局所秩序度：局所的な情報から計算される評価関数 (秩序化の程度をあらわす)．原子 (対) ごとに定義される．
  - 局所秩序度 (の和) が増加する時だけ規則が適用される．

CCM におけるシステムの構成要素と動作 — 2



- **CCM の動作**
  - 反応規則が適用できる原子のくみが存在するかぎり反応がつづき，なくなると停止する．
  - 大域的な “秩序状態” (目的状態？) を探索する．

目次

- **まえおき**
- **計算モデル CCM**
- **CCM による制約充足問題の解法**
- **制約充足問題の計算過程と結果**
- **むすび**

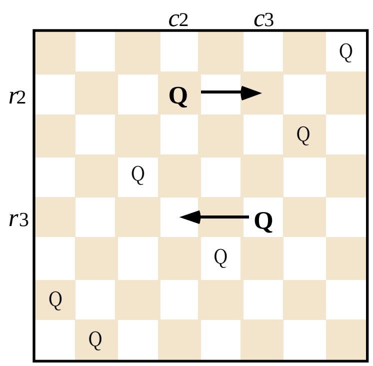
制約充足問題をとく手順

- **反応規則の仮決定**
  - 解探索空間を移動するための適当な操作をみつける．
- **データ構造の記述**
  - なにを原子としてどのように表現するかをきめる．
- **局所秩序度の記述**
  - 原子 (対) が解にふくまれるための必要条件からきめる．
- **反応規則の記述**
  - 反応により秩序度が増加するように規則を補正・記述．
- **初期状態の設定**
  - 探索空間内の適当な状態を初期状態として設定する．
- **計算**
  - システムを動作させる．

プロダクション規則と局所評価関数による最適化の方法とその計算過程におけるマクロなふるまい

反応規則の仮決定

- **探索空間を移動するための適当な操作 — 反応規則となるべきものをみつける．**
- **これらの操作は，反復適用によって探索空間をおおいつくすようにきめる．**
- **この段階では記述する必要はかならずしもない．**
- ***N* クウィーン問題のばあい：2 個のクウィーンの列を交換する操作がつかえる．**



データ構造の記述

- **なにを原子として表現するか，どのように表現するかをきめる．**
  - リンクをつかうか，内部状態として表現するかなどをきめる．
- **計算言語 SOOC (Self-Organization-Oriented Computing) によって記述する．**
- ***N* クウィーン問題におけるデータ構造の SOOC による表現**
  (defelement queen

  …… Lisp の defstruct にちかい (element = 元素)

  row …… 行番号

  col) …… 列番号

局所秩序度の記述 — 1

- **CCM においては局所秩序度の定義がもっとも重要．**
- **原子 (対) が解にふくまれるための必要条件をつかって局所秩序度を定義する．**
- **自己秩序度または相互秩序度のかたちで定義する．**
  - 自己秩序度 *o*(*x*)
    - 各原子 *x* について定義される．
    - 解にふくまれるための必要条件が 1 個の原子について定義できるばあい．
  - 相互秩序度 *o*(*x*, *y*)
    - 2 個の原子 *x*, *y* のあいだに定義される．
    - 解にふくまれるための必要条件が 2 個の原子のあいだに定義できるばあい．

局所秩序度の記述 — 2

- **自己秩序度の定義法の説明は省略する (相互秩序度に準じる)．**
- **相互秩序度のばあい**
  - 解は原子対によって構成されるとかんがえる．
  - 定義：「解にふくまれることとなる原子からなるすべての対 <a1, a2> (a1 ≠ a2) について C(<a1, a2>) がなりたち，かつ探索空間内の任意の状態 S について作業記憶がふくむ任意の原子の対 *p* について C(*p*) がなりたてば S が解である」という命題がなりたつような条件 C(*p*) がみつかれば，
    if C(<a1, a2>) then *o*(a1, a2) = 1 else *o*(a1, a2) = 0
    のように定義する．
  - *o* は C をみたす集合のメンバシップ関数．
- ***o* が $0 \le o \le 1$ の連続値をとるとすると，ファジィ制約充足になる (?)**

プロダクション規則と局所評価関数による最適化の方法とその計算過程におけるマクロなふるまい

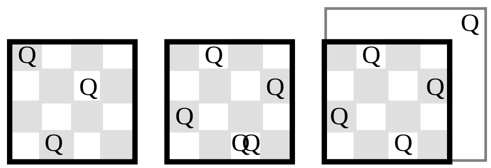
局所秩序度の記述 — 3

■ **命題後半部が必要な理由**

◆ 命題後半部：「探索空間内の任意の状態 S について作業記憶がふくむ任意の原子の対 *p* について C(*p*) がなりたてば S が解である」

◆ 理由は「この条件がないとシステムが不正に停止する可能性があるから」

◆ *N* クウィーン問題における，命題後半部をみたさない例



クウィーン数がすくなすぎる　　同位置に複数のクウィーンがある　　盤面をはみだすクウィーンがある

局所秩序度の記述 — 4

N クウィーン問題のばあい

■ **相互秩序度の定義**

$o(x, y) =$ 0 **if** *x.column* − *y.column* = *x.row* − *y.row* **or**
x.column − *y.column* = *y.row* − *x.row* ,
1 **otherwise** .

■ **SOOC による表現**

```
(deforder ((x queen) (y queen))
  (if (or (eql (– (queen-col x) (queen-col y))
               (– (queen-row x) (queen-row y)))
          (eql (– (queen-col x) (queen-col y))
               (– (queen-row y) (queen-row x))))
      0
      1))
```

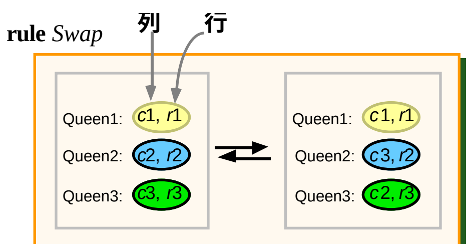

反応規則の記述

■ **必要ならば反応により秩序度が増加するように規則を補正して記述する．**

■ ***N* クウィーン問題のばあい**

◆ 規則にあらわれるクウィーンが 2 個だけだと，反応によって秩序度は変化しない．

◆ 第 3 のクウィーン (触媒) を規則の両辺にくわえる．



さらなる補正が必要ないことは，計算してみることによってわかる．

初期状態の設定

■ **探索空間内の適当な状態を初期状態として設定する．**

■ ***N* クウィーン問題のばあい**

◆ クウィーンはすべて盤面にある．

◆ クウィーンは各行各列にただ 1 個とする (対角線方向には 2 個以上あってもよい)．

計算

- **システムを起動する．**
- ***N* クウィーン問題のばあい**
  - 前記の規則と局所秩序度とでほぼ 100% 解をもとめることができる．
  - 規則の再補正などの必要はない．

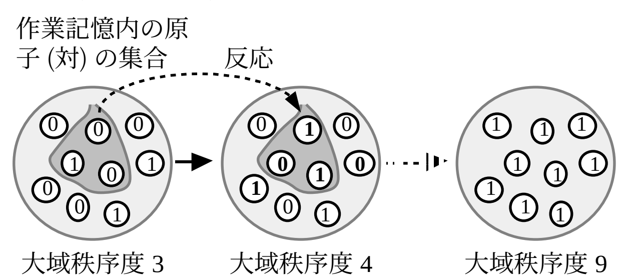
CCM による制約充足問題解法の特徴

- **バックトラックをつかった制約伝搬によらない．**
  - ランダムに部分的な制約充足をくりかえす．
- **充足度に関するやまのぼり法とはことなる．**
  - マクロには大域的な充足度をたかめるように動作するが，充足度は計算しない．
- **確率的な方法である．**
  - 解の正当性は保証できるが，計算の停止性は (確率的にしか) 保証できない．
- **“プログラム” が非常に単純である．**
  - *N* クウィーン問題や地図彩色問題のようなかんたんな問題のばあいは．

目次

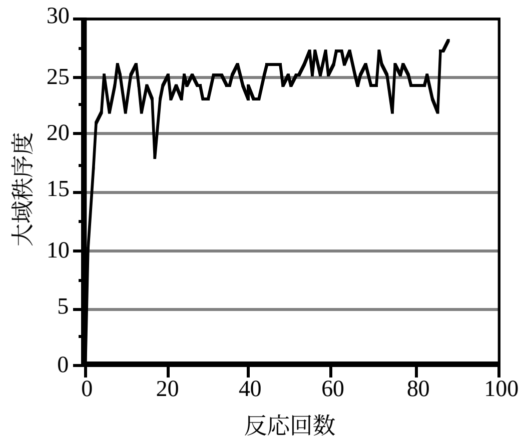
計算過程

- **大域秩序度の定義**
  - 局所秩序度の作業記憶全体にわたる和．
  - *N* クウィーン問題のばあいは，全クウィーン対に関する局所秩序度の和．
- **局所秩序度と大域秩序度との変化の例**

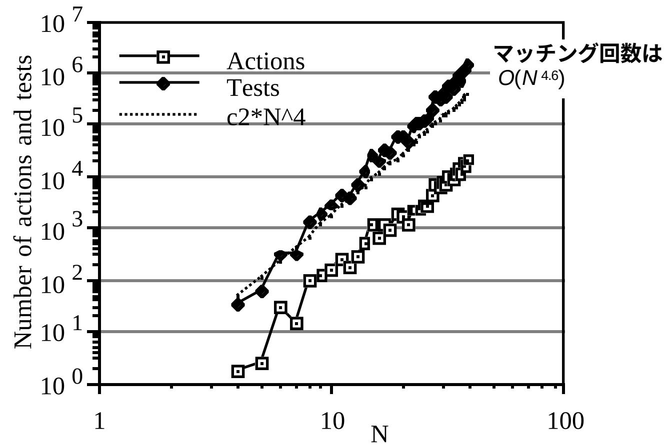


プロダクション規則と局所評価関数による最適化の方法とその計算過程におけるマクロなふるまい

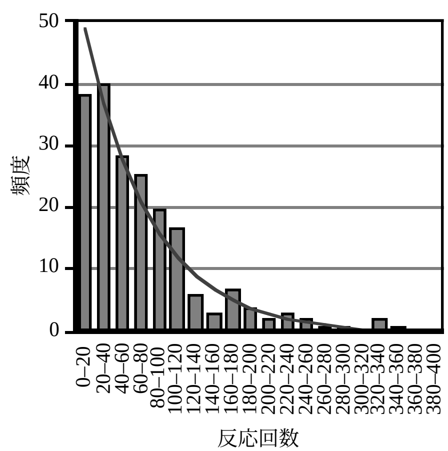
計算過程 — *N* クウィーン問題のばあい



平均計算時間 — *N* クウィーン問題のばあい

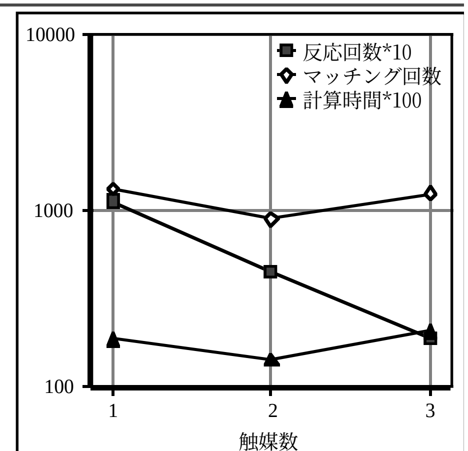


計算時間の分布 — *N* クウィーン問題のばあい



触媒数と計算時間

- **触媒数 2 のときが計算時間は最小．**
  - 触媒数 1 〜 3 の範囲で 20 回測定した結果．
  - *N* = 8 ．



むすび

- **結論**
  - 多数の局所的な記号的制約を充足する問題を CCM にもとづいてとく方法をしめした．
  - この方法によれば，制約伝搬によらない確率的な方法によって，単純な "プログラム" によってある種の制約充足問題をとくことができる．
- **今後の課題**
  - よりひろい範囲の制約充足問題に適用できるようにする．
    - システムがうまく動作しなかったときの補正法の洗練．
    - 大域的な制約もとけるようにする．
  - 従来の手法が適用しにくい動的な問題に適用すること．
    - この研究の本来の目標は仕様が時間とともに変化する，あるいは仕様を記述できない問題．