社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

局所情報によるアニーリングをつかった 大規模制約充足とその並列処理 — 創発的計算のためのモデル CCM の応用 —

金田　泰
新情報処理開発機構 (RWCP)　つくば研究センタ
〒305 茨城県つくば市竹園一- 6 - 1 つくば三井ビル
E-mail: *kanada@trc.rwcp.or.jp*, WWW: *http://www.rwcp.or.jp/people/yk/*

あらまし　この報告では，創発的計算のためのモデル CCM (化学的キャスティング・モデル) にもとづいて大規模な制約充足問題をとくための方法を提案する．また，その並列処理の方法をしめす．CCM にもとづく方法ではこれまで大規模な問題をとくことができなかったが，FAM (フラストレーション蓄積法) という一種のアニーリングを導入し，さらにパラメタをうまく調整することによって，大規模なグラフ彩色問題を GSAT やシミュレーテッド・アニーリングと同程度の計算時間で逐次処理でとくことができるようになった．さらに，かぎられた量の相互排斥だけをつかって，比較的容易に並列処理できることがわかった．ある条件のもとではほぼプロセッサ数に比例する性能がえられた．

キーワード　創発的計算，制約充足，並列処理，アニーリング，彩色問題，グラフ彩色

Large-scale Constraint Satisfaction Using Local-information-based Annealing and Its Parallel Processing — An Application of Emergent Computation Model CCM —

Yasusi Kanada
Tsukuba Research Center, Real-World Computing Partnership
Takezono 1-6-1 Tsukuba Mitsui Bldg., Tsukuba, Ibaraki 305, Japan
E-mail: *kanada@trc.rwcp.or.jp*, WWW: *http://www.rwcp.or.jp/people/yk/*

Abstract　A method for solving large-scale constraint satisfaction problems based on CCM (Chemical Casting Model), which is a model for emergent computation, is proposed in this report. A parallelized version of this method is also shown. Large-scale problems could not be solved using CCM. However, this report shows that, by introducing a method of annealing called FAM (Frustration Accumulation Method) and by adjusting the parameters appropriately, several large-scale graph coloring problems has become solvable with spending the same order of time as GSAT or simulated annealing by sequential processing using CCM. This report also shows that this method can easily be parallelized with restricted amount of mutual exclusion. The performance is almost proportional to the number of processors under certain conditions.

Key words　Emergent computation, Constraint satisfaction, Parallel processing, Annealing, Coloring Problem, Graph coloring.

1. まえがき

実世界の計算システムは刻々と変化する環境に対してひらかれた複雑なシステムである．そこではつねに予測不能な変化がおこりうるので，閉じた完全なシステム仕様は記述できない．また，複雑だということは問題が非線形である，またはうまくモジュール分解や分割統治ができないことを意味する．ところが，従来のシステム開発法では閉じた仕様の存在を仮定していて，かつ本来は線形 (単純) なシステムにだけ適用できるはずのモジュール分解を基本としている．したがって実世界のシステム開発には限界があるとかんがえられる [Kan 92, Kan 94a]．

このような状況のもとでは問題解決のために必要な情報をあらかじめ完全にあつめられないし，とじた完全な情報にもとづく従来型のシステムは環境の変化によわいであろう．そこでは局所的・部分的な情報による創発的計算 [For 91] が重要になるであろう．このようなかんがえにもとづいて，金田 [Kan 92, Kan 94a] は創発的計算のための計算モデルとして化学的キャスティング・モデル (Chemical Casting Model, CCM) を提案している．CCM は局所的・部分的な情報だけによるひらかれた計算をめざしている．

局所 (的な) 情報にもとづく計算といっても，個々のオブジェクト (エージェント) だけを見て周囲をまったく見なければ，計算の目的をはたせない．したがって局所性の程度を制御する必要がある．CCM においては比較的容易にそれを制御できる．ところが，より非局所的な計算をめざせば，定常状態にいたるまでの計算時間を短縮することができる一方で局所最適解にとらわれやすくなる．そこで，ある程度の非局所性をたもったまま，よりよい解をもとめるために一種のアニーリングを導入するのがよいとかんがえられる．通常はアニーリングは大域的な関数のうえで定義されるが，CCM においては局所情報による計算の原則をくずさないように定義する必要がある．そのための方法として金田 [Kan 94b] はフラストレーション蓄積法 (Frustration Accumulation Method, FAM) を導入し，他の方法と比較した．

この報告では FAM について，最近の研究結果を反映してよりくわしくのべる．また，FAM をともなう CCM の並列計算法についてものべる．局所情報による計算のひとつの利点は並列計算が容易だということである．この並列計算法を彩色問題に適用してえられた結果は，その容易さを実証している．

第 2 章で CCM とそれによる制約充足の例についてのべ，第 3 章で CCM におけるアニーリングの必要性と FAM について説明する．第 4 章で FAM つき CCM の並列処理の方法を説明する．第 5 章で FAM つき CCM による大規模グラフ彩色の逐次および並列処理の実測結果をしめし，第 6 章で結論をのべる．

2. 計算モデル CCM と制約充足

この章では，CCM とそれにもとづく制約充足について最小限の説明をする．

CCM はエキスパート・システムの開発につかわれる前向き推論のプロダクション・システムにもとづいている．プロダクション・システムは if-then 型の規則の集合 (長期記憶) を作業記憶 (短期記憶) にふくまれるデータに適用することによって動作する．古典的なプロダクション・システムは決定論的に動作し，規則の条件部がみたされればそれだけで動作する．しかし CCM においてはマクロな動作は揺動的 (確率的) であり，局所情報 (少数のデータ) だけで計算される評価関数 (局所秩序度) で動作がきまる．

CCM は化学反応系とのアナロジーにもとづいている (図 1 参照)．したがって作業記憶内の単位データを原子とよぶ．原子は内部状態をもち，原子どうしを化学結合に似たリンクによって結合できる．システムの動作をきめるのは，化学反応式に相当する反応規則 (if-then 規則) である．局所秩序度は一種の評価関数であり，作業記憶の局所的な状態が "よりよい" ほどおおきな値をとるように定義される．局所秩序度は負号をつけた一種のエネルギー (原子間結合エネルギーのようなもの) とみなせる．

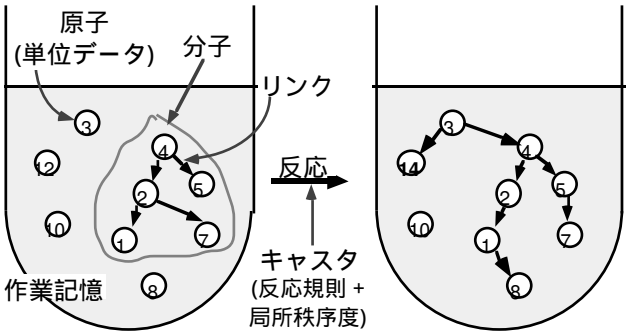

図 1　化学的キャスティング・モデルの構成要素

反応すなわち反応規則の適用は，それに関係する (規則両辺のパタンにマッチする) 原子の局所秩序度の和が反応によって減少しないときだけおこされる．反応しうる原子が存在しなくなると実行は中断する．反応しうる原子の組が複数あるときの反応順序は通常はランダムにきめられる．並列に反応させることもできる．したがって，システムはマクロにみれば局所秩序度の平均値すなわち平均秩序度 [Kan 94b]

の増加方向に非決定的に (ランダムに) 動作する[注1]．

CCM はプロダクション・システムにもとづいているが，規則を非決定的に適用する点と，規則の適用にあたって評価関数を使用する点が特徴的である．規則の非決定的な適用は，リミット・サイクルのようなのぞましくない状態をさけるやくわりをはたしている [Kan 93]．また，評価関数を使用することにより，環境が変化すると平衡点をめざしてふたたび動作するというような，ひらかれた環境のもとでの柔軟な計算が実現しやすいとかんがえられる [Kan 93, Kan 94b]．しかもこの評価関数には，ゲームのプログラムなどでつかわれる評価関数や，おなじく評価関数の一種といえるホップフィールドネットのエネルギー関数，遺伝的アルゴリズムの適応度関数などとちがって，局所情報 (少数のデータ) だけで計算されるという特徴がある[注2]．つまり CCM は創発的計算がみたすべき "局所情報による計算" という性質をみたしている．ただしこの性質をみたしているだけでは創発的計算とよべないことはもちろんである．

つづいて，CCM にもとづく静的な制約充足をとく方法を，グラフの頂点の彩色問題 [Kan 93] を例としてかんたんに説明する．この彩色問題は，グラフの頂点をあらかじめきめられた数の色にぬりわける問題である (以後，色数は 4 色に固定する)．グラフの隣接頂点は同色にならないようにぬる．なお，この問題は地図の彩色問題と対応づけることができる．

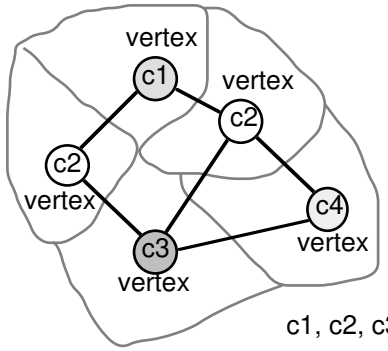

図 2　グラフ彩色問題の例

つぎに，グラフ彩色問題をとくためのシステムについて，つまり反応規則と局所秩序度について説明する．反応規則はただひとつあればよいが，その定義を視覚言語のかたちで 図 3 にしめす．この規則は，隣接する (リンクで結合された) ひとくみの 2 頂点をあらわす原子を (ランダムに) 選択して，そのうちの一方の頂点の色を 4 色のなかから選択してランダムにぬりかえる規則である[注3]．この反応規則のより詳細な意味については金田 [Kan 93] が説明している．

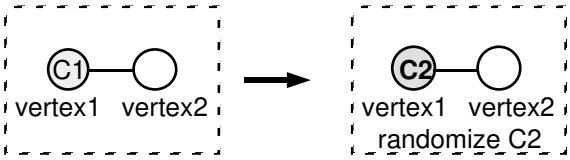

図 3　グラフ彩色問題のための反応規則の例

つぎに局所秩序度の定義をしめす．2 個の頂点 $v1$, $v2$ のあいだの局所秩序度をつぎのように定義する．

$$o_{vv}(v1, v2) =$$
$$1 \quad \textbf{if } connected(v1, v2) \textbf{ and } v1.color \neq v2.color$$
$$0 \quad \textbf{otherwise}$$

この定義は，頂点 $v2$ が $v1$ の隣接点であってかつ両者の色がひとしくなければ局所秩序度は 1，そうでなければ 0 という意味である．すなわち 2 個のオブジェクト間の局所秩序度は，その間の制約がみたされていれば 1，みたされていなければ 0 と定義する．このように定義すると平均秩序度は 0 と 1 のあいだの値をとり，全制約がみたされた状態で 1 になる．

図 3 の vertex1, vertex にマッチした頂点を $v1$, $v2$ とし，$v1$, $v2$ のフラストレーションを F_1, F_2 とすれば，反応がおこる条件はつぎのようになる (o_{vv} は反応前の局所秩序度，o'_{vv} は反応後のそれ)．

$$o_{vv}(v1, v2) \ < \ o'_{vv}(v1, v2)$$

図 3 とはことなる反応規則の例を図 4 にしめす．これは第 5 章で実験に使用する反応規則である．この規則は，ある頂点 vertex1 を (ランダムに) 選択して，その一方の頂点の色を 4 色のなかから選択してランダムにぬりかえるという点で図 3 の規則とひとしい動作をする．ただし，この規則のなかには vertex1 に隣接する (リンクで結合された) すべての頂点が記述されている (図では "…" によってそれらが表現されている)．それは，この規則を適用するかどうかをきめるときにこれらの頂点の局所秩序度を計算にとりいれるためである．

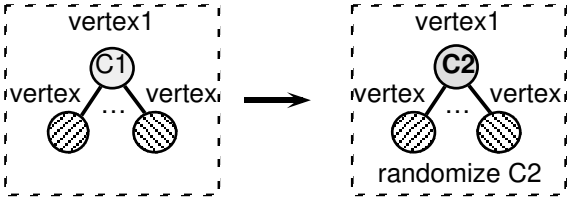

図 4　グラフ彩色の実験に使用した反応規則

[注1] システムの動作をきめるのに平均秩序度を計算するわけではない．平均秩序度を観測すれば増加しているという意味である．

[注2] "局所情報による計算" と "局所探索" の "局所" はまったく意味がちがうから注意が必要である [Kan 94b]．

[注3] 反応規則の右辺にあらわれる randomize C3 という表現が，反応規則の両辺にあらわれる頂点 vertex1 のあたらしい色の選択をランダムにおこなうことをあらわしている．

3. アニーリングの必要性と FAM

まえがきでのべたように，局所情報にもとづく計算といっても，完全に局所的な極限では意味のある計算はおこなえない．したがって周囲のオブジェクト (エージェント) を見る，つまりある程度の非局所性をもたせる必要がある．ところがそうすると局所最適解にとらわれる確率がたかくなる [Kan 94b]．それをさけるための一方法がアニーリングである．

通常，アニーリングは大域的な関数のうえで定義される．しかし CCM においては局所情報による計算の原則をくずさないように定義する必要がある．アニーリングの具体的な方法としてはさまざまなものがある．たとえばボルツマンマシンをまねたシミュレーテッド・アニーリング[注4]やフラストレーション蓄積法 (FAM) [Kan 94d] がある．ここでは実験でよりよい結果がえられた FAM だけを説明する．

FAM においては各原子がフラストレーションとよばれるエネルギーをもつ．各原子にはあらかじめ 0 をこえる一定値のフラストレーションがわりあてられる．反応規則の左辺が実行されたとき，まだ左辺にあらわれるパタンにマッチした原子に関してみたされていない制約が存在するならば，その原子のフラストレーションを増加させる．CCM においては反応させるかどうかを決定するときに反応前後の局所秩序度の和を比較するが，そのときに前者の値から各原子のフラストレーションの値をひく．これにより，フラストレーションがたまると平均秩序度を減少させるような反応しやすくなる．反応したときには，フラストレーションは初期値にもどす．

たとえば図 3 の反応規則においては，パタン vertex1, vertex にマッチした頂点を $v1, v2$ とし $v1$ のフラストレーションを f_1 とすれば，つぎの式がなりたつときに反応がおこる (o_{vv} は反応前の局所秩序度，o'_{vv} は反応後のそれ)[注5]．

$$o_{vv}(v1, v2) - f_1 \prec o'_{vv}(v1, v2)$$

フラストレーションを増加させる方法としては加法的な (線形に増加させる) 方法，乗法的な方法などがかんがえられる．これまで実験した範囲では乗法的な方法がよい (この点については第 5 章で説明する)．これは，頂点 i の更新前のフラストレーションを f_i，更新後のフラストレーションを f'_i とするとき，つぎのような式にしたがって更新する方法である．

$$f'_i = c f_i \quad (1 < c)$$

ここで係数 c は 1.01 ~ 4 程度の定数である．

図 4 の反応規則を使用するばあいにも，同様にして，つぎのような反応がおこる条件式があたえられる．ここでは $v2, v3, \ldots, vn$ がパタン vertex (複数個ある) にマッチした頂点をあらわす．

$$o_{vv}(v1, v2) + o_{vv}(v1, v3) + \ldots + o_{vv}(v1, vn) - f_1 \prec o'_{vv}(v1, v2) + o'_{vv}(v1, v3) + \ldots + o'_{vv}(v1, vn)$$

フラストレーションの更新式は図 3 のばあいとおなじである．

システムの動作をよりくわしくみるために，例として図 5 の上図のような状態をかんがえる．フラストレーションの初期値は $f_0 = 0.1$ だが，それが $v3$ についてはすでに 1 回増加されて 1.03 になっているとする．そして，c3 という値をもっていた変数 C3 に反応時に代入するためにランダムに選択された色が c1 であるとする．すると，Case 1 (中図) にしめすようになる．局所秩序度の和は反応により減少するので反応はおこらない．したがって頂点 $v3$ のフラストレーションは 1.06 ($=1.03^2$) に増加する．

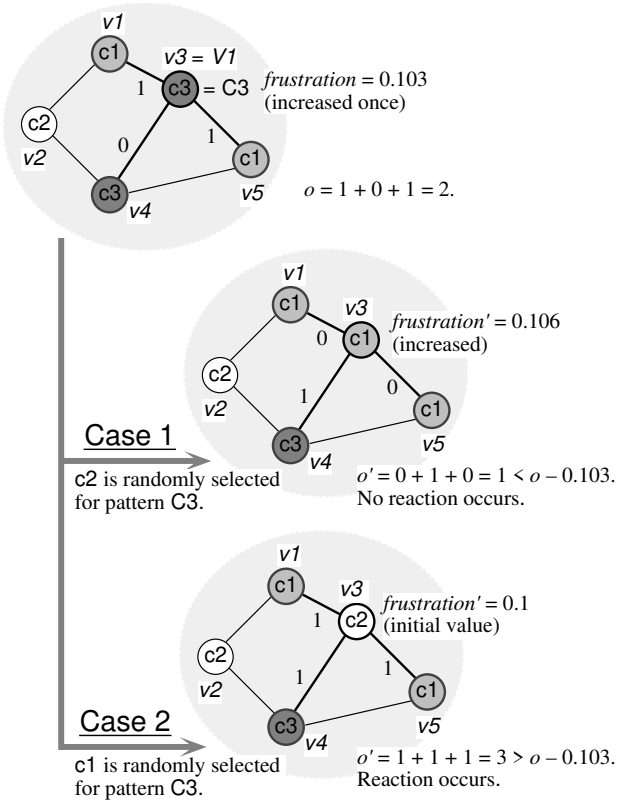

図 5　FAM の基本動作

もし C3 のために選択された色が c1 でなくて c2 であるとすれば，今度は Case 2 (下図) のようにな

[注4] ボルツマンマシンのばあいはシミュレーテッド・アニーリングによって 1 個だけのニューロンの動作がきめられるから，局所情報による計算の原則に合致している．

[注5] 金田 [Kan 94b] におけるのとおなじ方法をとるならば左辺から $v2$ のフラストレーションをひくことになる．しかしここでは触媒 (反応によって変化しない原子) のフラストレーションは計算にいれない方法をとっている．

る．このばあいは局所秩序度の和が増加するので反応がおこる．したがって，*v*3 のフラストレーションは初期値 $f_0 = 0.1$ にもどされる．

FAM をつかえば N クウィーン問題やグラフ彩色が経験的にはほとんどかならずとける．しかも，パラメタをうまくえらべば FAM をつかわずにとけたばあいにくらべた計算時間の増加を比較的ちいさくおさえられる．金田 [Kan 94b] は Johnson ら [Joh 91] があたえている大規模のグラフ彩色問題を FAM つきの CCM でとくと長時間かかるとのべているが，パラメタをうまく調整することによってこの問題はほぼ解決されることがわかった (c の値がおおきすぎたことが，時間がかかっていたおもな原因である)．Johnson らの問題の実験結果は第 5 章でしめす．

Selman ら [Sel 93a] や Morris [Mor 93] が提案している方法によれば，大域的な評価関数にもとづく充足可能性問題をとくときに各節 (FAM では原子に相当) がみたされるかどうかにしたがって各節におもみづけをする．それによって局所最小点にとらわれずに解がもとまるようになる．Selman らはかれらの方法が局所最小点において谷をうめるはたらきをするとのべているが，FAM も (正確な理解とはいえないが) 図 6 のように山 (上下が反転された谷) をけずるはたらきをしていて，前記の係数 c は山をけずる加速度だとかんがえられる．Selman らや Morris の方法と同様に FAM も制約充足問題 / 充足可能性問題のための方法であり，最適状態が局所的に定義できない大域最適化問題には基本的には適用できない．

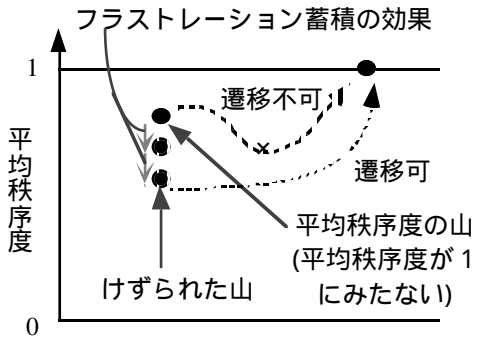

図 6　フラストレーション蓄積法の作用のイメージ

4. FAM つき CCM の並列処理法

CCM の並列処理法をおおきく分類すると，つぎの 2 種類がかんがえられる．

(1) 並列探索法：それぞれが独立の作業記憶をもつ複数のプロセスによって，独立に解を探索する方法である[注6]．この方法は Prolog でいえば OR 並列処理に相当する．

(2) 並列反応法：作業記憶を共有する複数のプロセスが，ひとつの解をもとめるために協調計算する方法である．この方法は Prolog でいえば AND 並列処理に相当する．

独立並列処理法 [Kan 94c] は確率的な問題分割にもとづく並列探索法の例である．FAM つきの計算にもこれを適用できるが，ここでは並列反応法をためした．

CCM は化学反応系とのアナロジーにもとづいている．化学反応が高速なのは並列に反応がおこるからにほかならない．CCM における計算は局所情報だけにもとづいているので，原理的には個々の場所で独立に計算ができるはずである．したがって，化学反応系と同様に並列処理ができるはずである．このような並列処理法が並列反応法である．並列反応法を実現するための機構を図 7 に図示する．

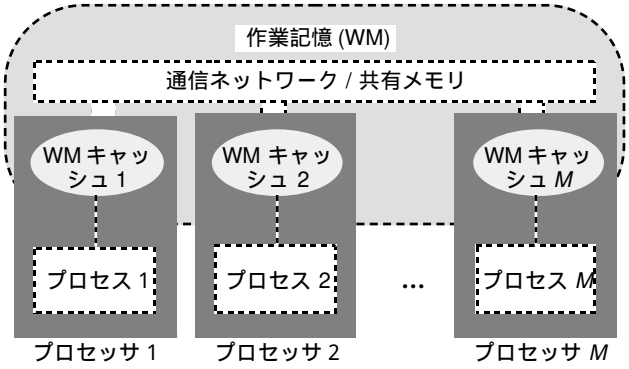

図 7　CCM にもとづく並列反応計算の方法

並列反応法においては複数の反応が並列におこる．複数の反応が同時にひとつの原子をかきかえようとすると，奇妙な結果がえられる可能性がある．しかし，ここでは化学反応系とのアナロジーが有用である．化学反応系においてはひとつの化学結合に影響をおよぼす複数の反応は同時にはおこらない．CCM において化学結合に相当するのはリンクだが，化学反応系とはちがって CCM においては原子の内部状態も変化するので，複数の反応は各原子を排他的にアクセスするということにすればよい．

しかし彩色問題においては，陽に排他制御をおこなわなくても並列反応によって不正な結果がえられることはない．なぜなら，図 3 ~ 4 の反応規則においては 1 回の反応でただひとつの原子をかきかえるが，そのかきかえがコンピュータ上で単位操作 (unit of operation) で実行されることが保証されていれば，並列におこる 2 つ以上の反応が競合してつじつまのあわないかきかえをおこなうことはないからである．

このばあい，並列処理によって生じる唯一のあら

[注6] 並列反応法においては概念的には作業記憶を共有するが，かならずしも物理的に共有記憶が必要なわけではない．

たな効果は，フラストレーションがちいさくても反応によって局所秩序度の和が減少しうることである．こういう現象がおこるのは，あるプロセスが反応条件をテストしてから実際に反応をおこすまでのあいだに，他のプロセスが他の反応をおこしてしまうことがありうるし，フラストレーションの更新が排他制御のもとでおこなわれないからである．同時におこる複数の反応が同一の原子に作用する確率が通常はひくいことをかんがえれば，この効果はちいさいとかんがえられる．したがって，彩色問題のばあいについては排他制御なしに効率的に解をもとめることができると予測される．このように排他制御をゆるめられるのは，CCM が決定的でなく揺動的 (stochastic) であって動作の厳密さが要求されないばあいがあるからだとかんがえられる．

並列処理性能を改善するためにはさらにつぎの 2 つの問題を解決する必要があった．第 1 の問題は停止条件，すなわち定常状態に達したかどうかの判定条件についてのものである．CCM にもとづく計算は停止する必然性はないが，工学的には定常状態を検出して停止することは重要だとかんがえられる．ところで各プロセスは基本的に独立に動作する．したがって，原則に忠実であるならば，各プロセスは他のプロセスの状態を参照することなく停止条件をきめるべきである．しかも CCM は揺動的に動作するので，いつ停止するかは確率的に (さいころの目で) きまる．しかしこれではいくつかのプロセスがはやく停止しすぎることになる．その結果，あるプロセスは他のプロセスが停止したあとも長時間走行することになり，並列度が低下して性能が低下する．この現象は実際に実験において観測されている．

そこで，停止するかどうかの判定をつぎのように改善した．大域的なカウンタ変数を 1 個用意し，システムの初期化時にそれにプロセス数 (固定されている) を代入しておく．各プロセスはその局所的な停止条件がみたされると，このカウンタの値を減少させる (カウンタ変数の排他アクセスのために相互排斥が必要になる)．しかしながら，そのプロセスはカウンタの値が 0 にならないかぎり，反応をおこしつづける．この方法によって局所性の原則は一部くずされるが，プロセスのはやすぎる停止をふせぐことができ，性能低下がさけられる．

第 2 の問題はフラストレーションを頻繁に増加させることによる性能低下である．逐次処理のときはフラストレーションは反応規則の左辺がテストされるごとに増加または減少される．しかし並列処理においてはフラストレーションという共有データに対するかきこみは高価なので，頻繁なかきこみは性能低下をまねく．そこで，反応ごとにフラストレーションを増加させるのをやめて，各プロセスが q 回のテストをおこなうごとにかきこむように変更し，同時に定数 c を c^q でおきかえた．これによって，ある原子のフラストレーションは増加すべきなのに増加しないままになったり，ある原子のフラストレーションは必要以上に増加されたりという現象がおこる．そのためにシステムのふるまいは変化するが，この効果は長期的にみれば各原子について公平に作用するので，一時的な不公平はたえられるであろう．

5. 逐次および並列処理の実測結果

この章では FAM つき CCM の逐次処理版と並列処理版の大規模なグラフ彩色への適用結果をしめす．

5.1 逐次処理の性能

性能測定用の例題は DIMACS ベンチマーク [Tri, DIM] から選択した．例題の一部について Johnson ら [Joh 91] によってシミュレーテッド・アニーリング (SA) による解法がためされ，Selman と Kautz [Sel 93a] によって GSAT による解法がためされている．

実測結果は表 1 にしめした．例題として使用したグラフは Johnson らによってランダムに生成されたものであり，125 ないし 250 の頂点をもつ．そして，辺の生成確率 p は 0.1 から 0.9 である．計算時間とはシステムが停止するまでの CPU 時間を意味する．この表から，使用しているコンピュータのちがいを考慮にいれても，いくつかのケースにおいては FAM つき CCM によって，GSAT や SA による解法と比較しうる性能がえられていることがわかる．ここで使用しているパラメタは実行中一定であり，その値は表の脚注にしるしてある．ただしその値はかならずしも最適ではない．各ケースはすくなくとも 20 回測定し，表にしめした結果は解をもとめるのに失敗したばあいをのぞいての平均値である[注7]．

CCM にもとづく SOOC-94 という計算言語とその処理系を開発してある [Kan 94d] が，C によって記述されたプログラムのほうが性能がよいので，ここでは C のプログラムによって反応規則や局所秩序度をコーディングして測定した．性能は Cray Superserver 6400 (CS6400) によって測定した．CS6400 は 60 MHz の SuperSPARC プロセッサを搭載した並列コンピュータだが，ここでは CPU を 1 個だけ使用している．

[注7] うまく停止条件をえらべば，解をもとめるのに失敗する回数が 20 回中 2 回をこえることはほぼなくせる．ただしい解がもとめられたかどうかは，検証用の反応規則を決定的に動作させることによって容易に検証できる．

表 1　大規模グラフ彩色の性能比較

グラフ			色数	計算時間 (sec)		
名称	頂点数	p		ACCM[*1]	Johnson etc.[*2]	GSAT [*3]
DSJC125.1	125	0.1	6	0.2 [1]	< 360	
DSJC125.1	125	0.1	5	323 [1]	720	
DSJC125.5	125	0.5	18	15 [1]	360	44
DSJC125.5	125	0.5	17	1536 [1]	6700	1800
DSJC125.9	125	0.9	44	119 [1]	1080	
DSJC125.9	125	0.9	43	とけていない	とけていない	
DSJC250.1	250	0.1	9	1.8 [2]	< 360	
DSJC250.1	250	0.1	8	*108 [2]	9360	
DSJC250.5	250	0.5	31	125 [3]	360	
DSJC250.5	250	0.5	30	548 [3]	2900	
DSJC250.5	250	0.5	29	11059 [3]	22000	4000

[*1] アニーリングつき CCM .マシンは Cray CS6400 (1 CPU).
[*2] Johnson ら [Joh 91] における最高速の方法．マシンは Sequent Balance 21000.
[*3] Selman ら[Sel 93a]における最高速の方法．マシンは MIPS R6000.
[1] この測定に使用したパラメタ値はつぎのとおり: $f_0 = 0.8$, $c^8 = 1.05$. 並列版のプログラムを使用.
[2] パラメタ値はつぎのとおり: $f_0 = 0.6$, $c = 1.08$.
[3] パラメタ値はつぎのとおり: $f_0 = 0.2$, $c = 1.023$ (この値はまだ最適化されていない).

5.2 並列処理の性能

前節とおなじプログラムと CS6400 の 12 台のプロセッサとを使用して並列処理性能を測定した．比較的少数のプロセッサをつかっている理由は，使用したコンピュータが 12 台しかプロセッサを搭載していないことと，問題の規模が頂点数にして 125 程度なので数 100，数 1000 のプロセッサを使用しても性能が改善されないとかんがえられることとである．

4 題の問題，つまり DSJC125.1 というグラフの 5 色および 6 色による彩色と，DSJC125.5 というグラフの 17 色および 18 色による彩色の性能を図 8 にしめす．縦軸にプロセッサ数 1 のときとの性能 (CPU 時間の逆数) の比をとっている．FAM のパラメタ値はつぎのとおりである：$q = 8$, $c^8 = 1.05$, $f_0 = 0.8$．各ケースをやはり最低 20 回測定して平均をとった．

5 色および 17 色による彩色の性能は理想性能に比較的ちかい．つまり線形にちかい加速をしめした．しかし 6 色および 18 色による彩色の性能は，プロセッサ数 M が 12 台のときに理想性能の半分以下になっている．この性能低下の原因は不詳だが，パラメタが最適値からはずれるためだとかんがえられる．実際，パラメタの値をかえて測定するとよりよい結果がえられることがある．たとえば 18 色で $M = 12$, $c^8 = 1.2$, $f_0 = 0.2$ のときには加速率は 10.3 となった．

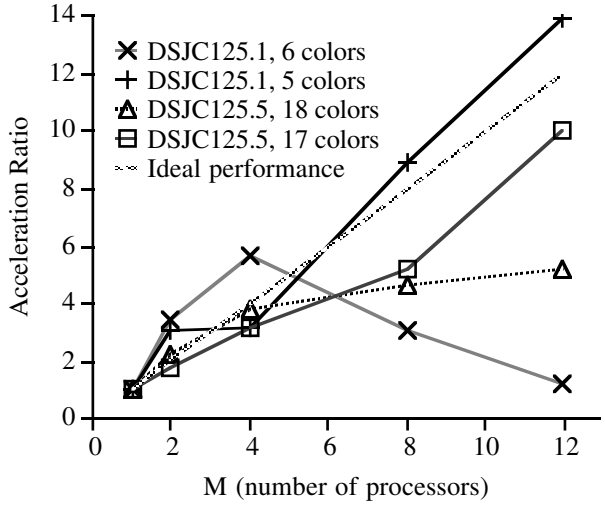

図 8　アニーリングつき並列反応 CCM による大規模グラフ彩色の性能

5.3 パラメタ値と性能との関係

DSJC125.5 の 18 色による彩色において $M = 12$ のばあいのパラメタ f_0 および c^8 を変化させたときの性能の変化を測定した結果を図 9 ~ 10 にしめす．図 9 には c^8 を固定して f_0 を変化させたときの計算時間の変化をえがいた．c^8 がおおきいほうがよい性能がえられる f_0 の範囲がひろいので，問題解決法としてあつかいやすいであろう．c^8 がちいさければフラストレーションの増加はほぼ線形になる．したがってこの結果は，第 3 章でのべたように，乗法的なフラストレーションが加法的なフラストレーションよりよいこともしめしている．

図 10 には f_0 を固定して c^8 を変化させたときの計算時間の平均値と標準偏差の変化をえがいた．この図からつぎのようなことが観察される．第 1 に，c^8 がおおきいところで，するどい計算時間の増加が観察される．たとえば $f_0 = 0.8$ なら $c^8 \approx 1.08$, $f_0 = 0.4$ なら $c^8 \approx 1.4$ のときにこの現象がみられる．この現象の原因は，おそらく c^8 がおおきいときにはフラストレーションと制約違反数とのあいだに正のフィードバックが存在するからだとかんがえられる．第 2 に，c^8 がちいさいところでは計算時間の標準偏差が比較的おおきいことがわかる．たとえば，$f_0 = 0.8$ ならば $c^8 \le 1.02$ のときに標準偏差がおおきい．おそらくこの現象が意味しているのは，もし c^8 がちいさければアニーリング効果が不十分なので，局所最大点からぬけだすのが困難になって，計算時間が非常におおきくなるばあいが生じるからだとかんがえられる．

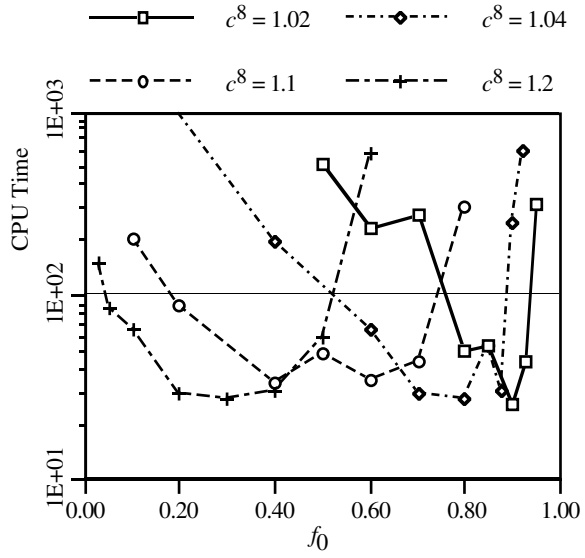

図 9　パラメタ値 f_0 と性能との関係

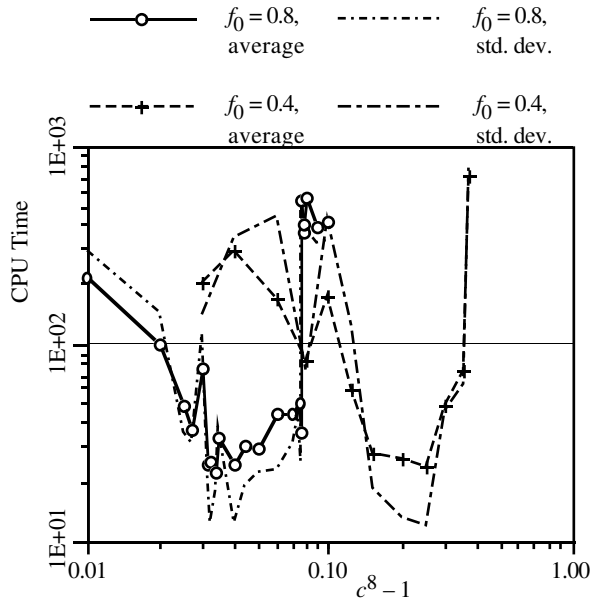

図 10　パラメタ値 $c^8 - 1$ と性能との関係

6. 結論

FAM によって CCM にもとづく大規模な制約充足が可能になった．FAM つきの CCM はかぎられた相互排斥のもとで容易に並列処理をおこなうことができる．このような並列処理は CCM が局所情報だけを使用し揺動的な計算をおこなうから可能になった．この方法で DIMACS ベンチマークのグラフ彩色のような問題については，ある条件のもとでプロセッサ台数にほぼ比例する性能をえることができた．

今後の課題を 2 点あげる．第 1 に，ここで実現されたのは比較的少数のプロセッサによる並列処理であり，いわゆる超並列処理は実現できていない．化学反応は超並列 (massively parallel) だとかんがえられるから，今後の研究によって CCM においても超並列処理が可能になるとかんがえられる．第 2 に，局所情報による計算はひらかれた変化する環境のもとでの問題解決を可能にするとかんがえられる．しかし，CCM においてこの言明を証明するには，今後，応用研究が必要である．

謝辞

CS6400 の使用法とプログラミング・テクニックについておしえていただいた RWC つくば研の関進氏と高橋裕信氏に感謝する．

参考文献

[DIM] Center for Discrete Mathematics and Theoretical Computer Science, *http://dimacs.rutgers.edu/*.

[For 91] Forrest, S., ed.: *Emergent Computation*, MIT Press, 1991.

[Joh 91] Johnson, D. S., Aragon, C. R., McGeoch, L. A., and Schevon, C.: Optimization by Simulated Annealing: An Experimental Evaluation; Part II, Graph Coloring and Number Partitioning, *Operations Research*, 39:3, 378–406, 1991.

[Kan 92] 金田 泰：コンピュータによる自己組織系のモデルをめざして，*第 33 回プログラミング・シンポジウム報告集*，1992.

[Kan 93] 金田 泰：プロダクション規則と局所評価関数にもとづく計算モデル CCM による問題解決法の特徴，*SWoPP '93*，93-AI-89-2, 11–20, 1993.

[Kan 94a] Kanada, Y., and Hirokawa, M.: Stochastic Problem Solving by Local Computation based on Self-organization Paradigm, *27th Hawaii International Conference on System Sciences*, 82–91, 1994.

[Kan 94b] 金田 泰：創発的計算のためのモデル CCM による問題解決法における局所性の制御法，*SWoPP '94*，94-AI-95-4, 29–38, 1994.

[Kan 94c] 金田 泰：創発的計算のためのモデル CCMによる制約充足問題などの独立並列処理法，*情報処理学会第 49回全国大会*, 4-321 - 322, 1994.

[Kan 94d] 金田 泰：創発的計算のための言語 SOOC: その特徴と実装 — 魔方陣を例題として —，情報処理学会研究報告，94-SYM-75-5, 31-38, 1994.

[Mor 93] Morris, P.: The Breakout Method For Escaping From Local Minima, *AAAI 93*, 40–45, 1993.

[Sel 93a] Selman, B., and Kautz, H.: Domain-Independent Extensions to GSAT: Solving Large Structured Satisfiability Problems, *IJCAI 93*, 290–295, 1993.

[Sel 93b] Selman, B., and Kautz, H. A.: An Empirical Study of Greedy Local Search for Satisfiability Testing, *AAAI 93*, 46–51, 1993.

[Tri] Tricks, M.: The Second DIMACS Challenge, *http://mat.gsia.cmu.edu/challenge.html*.