3T−1

グラフの色ぬり分け問題からSATへの効率の良い変換方法とその評価

宮崎 修一　岩間 一雄
九州大学工学部

1. はじめに

NP完全問題は,問題のサイズに対して多項式時間で解くアルゴリズムが知られていない問題の代表例である.これらの問題は多項式程度の違いを無視すれば,同じ計算時間で解けるという点で,同じクラスに属する.即ち,ある問題を多項式時間で別の問題に変換することができる.これらの変換は,問題のNP完全性を証明するのに用いられてきた.多項式時間であればどのような変換であってもかまわないという大雑把なものであったが,問題を変換して解くという実用性を考えれば,変換の効率を良くすることが大切になってくる.

例えば,CNF論理式の充足可能性問題(SAT)に対する効率の良いアルゴリズムに局所探索法[1][2][3]と呼ばれるものがある.ハミルトン閉路問題をSATに変換し,局所探索法で解く場合には,変換によって得られる論理式のの変数の数によって計算時間に格段の差があることが分かっている[4].

本研究では,別のNP完全問題である,グラフの頂点彩色可能性問題をSATに変換する方法を考察してみた.本稿では,まず,頂点数N,色数Kとして,$N\log K$変数での比較的自然な変換の方法を述べる.次に,$N\log K$よりも少ない変数の数で変換する方法を2つ述べる. 1つは,グラフ中に枝が少ないときに有効であり.もう1つは,枝の数が多いときに有効であることが分かった.

NP完全問題は"手に負えない"という形で統一的に論じられることが多く,個々の問題の難しさの違いはあまり論じられていないようにみえる.本論文で述べている手法,つまりNP完全問題PをSATに変換するのに何変数必要であるかは,ある意味でPの難しさのメジャーになりうると考えられる.計算ステップ数,領域量等の従来のメジャーとの大きな違いは,定数係数の違い,あるいは定数の差さえも十分に議論でき,それが重要とみなされる点である.

2. 頂点彩色可能性問題

頂点彩色可能性問題とは,N頂点のグラフ$G=(V,E)$と,K色が与えられたとき,グラフの頂点を,隣接した頂点どうしが同じ色で塗られないようにK色以内で塗り分けることができるかどうかを問う問題である.

2.1 アルゴリズム VC_to_SAT

ここでは,$N\log K$変数での変換について述べる.用意する変数は$x_{1,1},x_{1,2},\cdots,x_{1,\log K},x_{2,1},x_{2,2},\cdots,x_{2,\log K},\cdots,x_{N,1},x_{N,2},\cdots,x_{N,\log K}$である.色は$0\le l\le K-1$の$l$で表現することにし,それを2進数で表したものを$C_l=(a_{l,1},a_{l,2},\cdots,a_{l,\log K})$とする.つまり,頂点$i$の塗られる色を$x_{i,1},x_{i,2},\cdots,x_{i,\log K}$で表す.ここで,$x_{i,\log K}$が高い位を表し,$x_{i,1}$が低い位を表す. 例えば$(x_{5,1},x_{5,2},x_{5,3},\cdots,x_{5,\log K})$=(1,1,0,$\cdots$,0)であれば,頂点5は色3で塗られることになる.ここで$X_i=(x_{i,1},x_{i,2},\cdots,x_{i,\log K})$とする.

[アルゴリズム VC_to_SAT]

ステップ1: すべての頂点iに対して,$X_i\le C_K$のとき0となる節をつくる.

ステップ2: すべての枝(i,j)とすべての色C_lに対して,$(X_i=C_l)\wedge(X_j=C_l)$のとき0となる節をつくる.具体的には,すべての枝に対して次の操作を行なう.

枝の両端の頂点をi,jとする.$0\le l\le K-1$の各lに対して,節を1つつくる.つくり方は,以下の通りである.

$1\le s\le\log K$に対して,$a_{l,s}$を見る.$a_{l,s}=1$ならば,$\overline{x_{i,s}},\overline{x_{j,s}}$を節中に加える. $a_{l,s}=0$ならば,$x_{i,s},x_{j,s}$を節中に加える.

ステップ1では,塗られる色番号の最大値を超えないようにしている.ステップ2では,隣り合う頂点どうしが同じ色で塗られないようにしている.これは,すべての枝に対して,その両端にある頂点どうしが同じ色で塗られたとき0となる節を各枝に対してk個つくっている.

これは,すべての頂点にすべての色を割り当てる状態を表し得るだけの変数を用いた変換である.したがって変数の数をこれよりも少なくすることができれば,すべての状態を表さなくても変換が可能であるということが言える.次節からは,そういう変換について議論する.

2.2 アルゴリズム VC_to_SAT II

まわりに色が与えられていれば,その中心にある頂点には色を表す情報を与えなくても良い.すなわち,K色で塗られる場合,まわりの頂点を塗るのにK色全てが使わ

Efficient polynomial-time reduction from Vertex Coloring Problem to Satisfiability and its evaluation
Shuichi Miyazaki and Kazuo Iwama
Faculty of Engineering, Kyushu University

れてしまったなら中心の頂点はどの色でも塗ることができない. そこで, 中心の頂点の変数を省き, 中心の頂点の情報はまわりの頂点を用いて表すというのが基本戦略である. ここで, 頂点を分けるための2つの集合を用意する, 集合 $NOCOL$ には中心となる頂点を, 集合 COL にはそれ以外の頂点を入れるものとする. 使用する変数は色を与える頂点の数を N_1 個として $N_1 \log K$ であり, その使い方はアルゴリズム VC_to_SAT と同じである.

［アルゴリズム **VC_to_SAT II**］

ステップ1： V の要素の中で $NOCOL$ の要素とつながっていないものをランダムに選び,$NOCOL$ に入れる操作を続け, できなくなったら残りの頂点を COL に入れる.

ステップ2： すべての COL の頂点 i に対して,$X_i \leq C_K$ のとき0となる節をつくる.

ステップ3： COL の頂点どうしをつなぐ枝 (i,j) とすべての色 C_l に対して, $(X_i = C_l) \wedge (X_j = C_l)$ のとき0となる節をつくる.

ステップ4： $NOCOL$ の要素である頂点を a とし,a の次数を $d(a)$ とする. $d(a) < K$ ならば何もしない.$d(a) \leq K$ ならば次のように節をつくる.
a とつながる $d(a)$ 個の頂点の中から K 個取り出し, 全てが異なった色を与えられたとき0となる節をつくる.

ステップ1では頂点を2つに分けている. ここで, 集合 $NOCOL$ の要素となる頂点は互いに継っていないので, 集合 $NOCOL$ は独立頂点集合であることが分かる. この最大を求める問題はNP完全であるが, ここで採用した方法は多項式時間の近似解法である. また, 色数 K は定数と考えているので, 式の長さは多項式で収まる.

2.3 アルゴリズム VC_to_SAT III

ここでは,4色塗り分けについて考えてみる. グラフの中に三角形を構成する部分があればそれらの3頂点は全て違う色で塗られなければならない. ここでは三角形を構成する3頂点に対して5変数を割り当てることにする. これは,2つの頂点には従来通り2変数を割り当て, 残りの頂点に1つの変数を割り当てることにするものである.2つの頂点の色を表した段階で, 残りの頂点には残り2色のうちの一つしか割り当てられないから, その色を表すために1変数を用い, 残った色のうち小さい番号の色を表す場合には0を, 大きい番号の色を表す場合には1を使うことにする. つまり,1変数与えられた頂点の色を表すために5変数を用いるわけである. また, 三角形に使われなかった頂点にも2変数を与える. ここで,2変数与える頂点を入れる集合 $BASE$ と1変数与える頂点を入れる集合 TRI を用意する. ここでは, 三角形の選び方についてのみ述べる.

［アルゴリズム **VC_to_SAT III**］

ステップ1： TRI にも $BASE$ にもつながっていない枝を1つ選び, その両端の頂点を集合 $BASE$ に入れる. 集合 $BASE$ のどちらともつながっている頂点を選べるだけ選んで, 集合 TRI に入れる.

ステップ2： ステップ1ができなくなるまで繰り返す.

ステップ3： 残った頂点を集合 $BASE$ に入れる.

この方法ではステップ1で見つけた TRI の頂点どうしがつながっていた場合には4完全グラフとなるため, さらにどちらかの頂点の変数を減らすことができる.

ここでは4色塗り分けについて論じたが, 一般の K 色に拡張できる.

3. 変換効率の解析

ここでは, 提案したアルゴリズムによってどれだけ変数の数が減らせるかを計算する. 与えられるグラフの頂点数を N, 頂点間に枝の存在する確率を p とする.

アルゴリズム VC_to_SAT II の方法では, $\frac{\log N}{\log N + pN} N \log K$ 変数を減らせることが分かった. つまり, $p = \frac{\log N}{N}$ 以下のときは $\frac{1}{2} N \log K$ （すなわち半分）ぐらい減らせることを意味する. また, アルゴリズム VC_to_SAT III では, $\frac{N - \frac{1}{p^2}}{3}$ 変数を減らせることが分かった. この場合も $p = \sqrt{\frac{\log N}{N}}$ 以上なら, $\frac{N}{3}$ 変数ぐらい減らせることが分かる.

直観的にも計算結果からも分かるように,p の値が大きいときはアルゴリズム VC_to_SAT III によって, また,p の値が小さいときはアルゴリズム VC_to_SAT II によって変数を減らすことができる. したがって, この2つのアルゴリズムを兼ね合わせて使うことが考えられる.

4. おわりに

本稿では頂点ぬり分け問題からSATへの効率の良い（少ない変数での）変換方法を示した. これからは, 他のNP完全問題からSATへの変換についても変換方法を考えるとともに, 変換に必要な変数の下限を求めたい.

参考文献

[1] E. Koutsoupias and C. Papadimitriou. On the greedy algorithm for satisfiability. *Information Processing Letters 43*, pp.53-55,1992.

[2] D. Mitchell, B. Selman and H. Levesque. Hard and easy distributions of SAT problems. In *Proc. Tenth National Conference on Artificial Intellegence.* pp.459-465,1992.

[3] 宮崎, 岩間. CNF論理式の充足解に対する改良されたランダム探索法. 平成4年度九州支部連合大会（平成4年10月）.

[4] 宮崎, 岩間. CNF論理式に対する局所探索法の評価. 平成4年度情報処理学会アルゴリズム研究会（平成5年3月）.