

チャンネル割当問題の解法

宮本 裕一郎[†] 松井 知己^{††}

本論文では、携帯電話の基地局に対するチャンネル割当問題を組合せ最適化問題として定式化した。そして厳密解法、近似解法、発見的解法を提案した。またそれぞれの解法について同心円グラフを入力とする計算実験を行い、考察を行った。厳密解法の章では、既存のパッケージソフトウェアを用いるため、整数線形計画問題へ帰着する定式化を提案した。近似解法の章では、特定のグラフに対して5-近似の精度を保証する解法を提案した。発見的解法の章では、2つの構築法と2つの改善法を提案し、それらを組み合わせたいくつかの発見的解法を提案した。

Algorithms for channel assignment problems

YUICHIRO MIYAMOTO[†] and TOMOMI MATSUI^{††}

In this paper, we present algorithms for channel (frequency) assignment problems. We formulate channel assignment problems as combinatorial optimization problems. We propose an exact method, an approximation algorithm and heuristic algorithms. We also report the results of computational experiences. We formulated the problem as an integer linear programming problem and applied a package software. We present a 5-approximation algorithm for particular graphs which are similar to real instances. We propose two construction methods and two improvement methods. Our heuristic algorithms are combinations of construction methods and improvement methods.

1. はじめに

近年、日本では携帯電話の普及率が急速に高まってきている。それに伴い携帯電話の基地局が街のあちこちに設置されている。携帯電話1つが1回の通話で使う周波数帯をチャンネルと呼ぶ。お互いの距離が近い携帯電話は、それぞれが同時に通話する際には、混線を避けるために、周波数の離れたチャンネルを使う必要がある。よって、狭い範囲でたくさんの携帯電話を使えるようにするためには、それぞれの基地局が使えるチャンネルを多くすればよい。1つのチャンネルで複数の携帯電話と通信する方法も考えられるが、本論文ではその方法は議論しない。本論文では1つの基地局が使えるチャンネルを多くする問題を扱う。

チャンネル割当問題は周波数割当問題 (Frequency Assignment Problem) とも呼ばれ、それに関する参考文献は多数存在する。以下では重要と思われる参考文献を紹介する。参考文献1)ではチャンネル割当問題を厳密に

解くための分枝限定法を中心に論じているが、定式化の工夫もまた施してある。参考文献2)では発見的解法を構築している。動的割当法により初期解を生成し、チャンネルの入れ換えによって改善を行う解法である。この解法は実際にドイツの移動通信網のチャンネル割当問題を解くために使われている。扱っている問題例として最大のもは頂点数約4,000、辺数約520,000である。

本論文の構成は以下の通りである。第2節では問題を定義する。そして問題の背景との関係を説明する。問題のモデル化についても言及する。第3節では厳密解法について説明する。この節では整数線形計画法への定式化を試みる。第4節では近似解法を紹介する。P≠NPの場合、チャンネル割当問題には多項式時間近似解法が存在しない。よってこの節では頂点間のユークリッド距離のみに依存して生成される問題例に限定して、いくつか近似解法を提案し、それぞれについて考察を与える。第5節では実用的な発見的解法を提案する。第6節はまとめである。

[†] システム計画研究所

Research institute of systems planning

^{††} 東京大学大学院

University of Tokyo

2. 問題の設定

本節では本論文で論じるチャンネル割当問題を定義しその問題例について言及する. 2.1節では問題の定義を行い, 2.2節では本論文で問題例として扱う同心円グラフについて述べる.

2.1 問題の定義

本論文で扱う問題は携帯電話の基地局にチャンネルを割り当てる問題である. 各基地局の対において, どれだけチャンネルを離せばよいかは, 基地局間の距離及び基地局の出力や指向性に依存するとみなす. 問題の目的は, 与えられたチャンネルの集合を使って, 基地局1つ1つがなるべくたくさんのチャンネルを使える割当を求めることである. 本論文では以下の, 組合せ最適化問題として定式化された問題を扱う. そしてその解法を論じる. 同様の問題は論文3)ですでに提起されており, 下記と同様に定式化されている.

問題 (CHAP)

枝重みつき単純無向グラフ $G_w = (V, E, w)$ が与えられている. 以下の条件の元で, 全ての頂点にチャンネルを割り当てる. チャンネルは自然数で表される. 隣接する頂点に割り当てられるチャンネルの差の絶対値は, 頂点对に接続する枝の重み以上でなければならない. 問題の目的は, 割当に使われているチャンネルの最大値が最小となるチャンネルの割当を求めることである.

割当に使われているチャンネルの最大値を m とすると, 上記の問題は

$$\begin{aligned} \min. \quad & m, \\ \text{s.t.} \quad & |x_i - x_j| \geq w(e) \quad (\forall e = \{i, j\} \in E), \\ & 1 \leq x_i \leq m \quad (\forall i \in V), \\ & x_i \in N \quad (\forall i \in V), \end{aligned}$$

と定式化される. 本論文ではこの問題をチャンネル割当問題 (CHannel Assignment Problem; CHAP) と呼ぶ. さらに辺の重みとして現れる自然数が高々 k である問題を k -CHAP と呼ぶ.

図1はCHAP(2-CHAP)の問題例である. 図1の細い辺は重み1の辺を, 太い辺は重み2の辺を表している. 図1中の数字は頂点へのチャンネル割当の例, すなわち可能解, を表している.

CHAPはNP-hardである. なぜならば, CHAPの特別な場合である1-CHAPは頂点彩色問題と同等であり, 頂点彩色問題はNP-hardだからである. よってCHAPは効率的に解くのは難しい問題であると予想される.

2.2 問題例と同心円グラフ

まず同心円グラフの定義を与える.

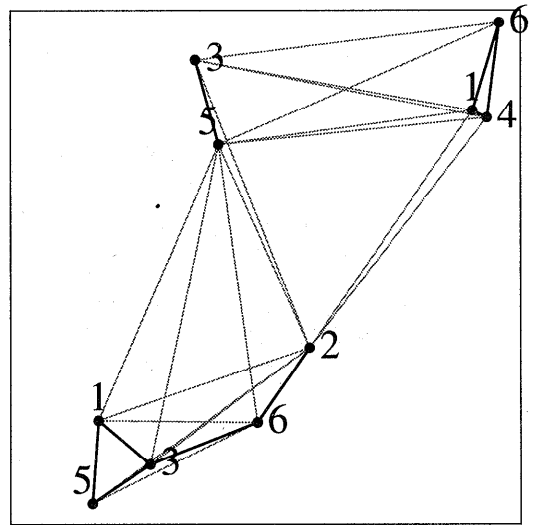


図1 CHAP(2-CHAP)の問題例と可能解.

Fig. 1 An instance and a feasible solution of CHAP(2-CHAP).

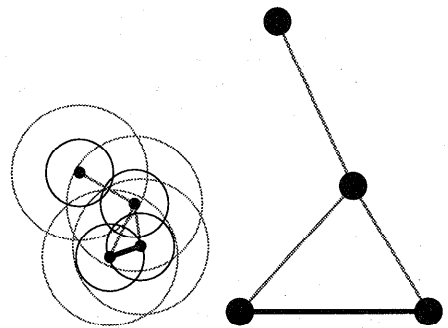


図2 同心円図と同心円グラフの例.

Fig. 2 An example of concentric disks and related concentric disk graph.

定義 (同心円グラフ)

$0 \leq r_2 \leq r_1$ を満たす実数定数 r_1, r_2 が与えられているとする. 2次元平面上の点集合が与えられた時, 各点対の間のユークリッド距離が r_1 以下ならば線分で結ぶ. 線分が r_2 より長いならば重み1を与え, 線分が r_2 以下の長さならば重み2を与える. こうして得られた図を同心円図と呼ぶ. 重みつき単純グラフのうち同心円図として平面に埋め込むことが出来るグラフを同心円グラフと呼ぶ.

図2は同心円図とそれに対応する同心円グラフの例である. 図の細い線は重み1の線分を, 太い線は重み2の線分を表している. 分かりやすくするために, 同心円図には同心円を書き加えてある.

CHAPは任意のグラフ $G = (V, E)$ と任意の重み関数 w を入力とする問題として定義された. 図1の問題

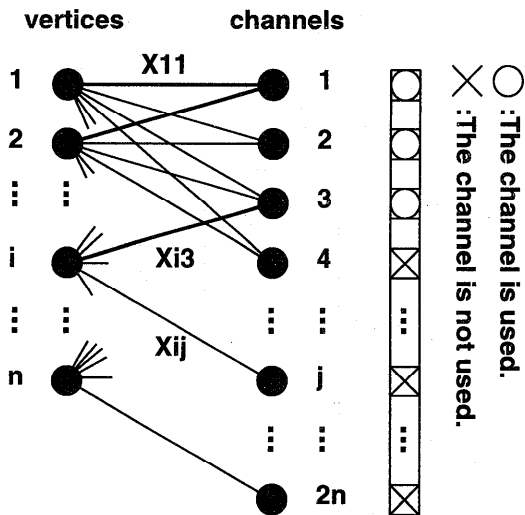


図3 マッチングに基づいた定式化の概念図。
Fig. 3 Matching based formulation.

例のグラフは同心円グラフである。これ以降も問題例として同心円グラフを使う。その理由を以下に述べる。基地局対から発せられる電波の干渉は基地局対の距離、発せられる電波の強度、基地局の指向性等に依存する。これらの要因の内一番重要なものは基地局対の距離とみなし、残りは無視する。電波干渉の強さは基地局対の距離が長いほど小さくなるとする⁴⁾。電波干渉の強さは連続値であるが、閾値を決めて{強い, 弱い, ない}の3つに分類する。そして強い干渉が起こる基地局対は同じチャンネルはもちろん隣のチャンネルも使ってはいけない。弱い干渉が起こる基地局対は同じチャンネルを使ってはいけない。干渉がない基地局対は同じチャンネルを使っても良い、とする。以上の手続きにより作られる問題例は同心円グラフになる。

3. 整数線形計画問題への定式化と厳密解法

本節では CHAP を整数線形計画問題に定式化して既存のソフトウェアを用いることを試みる。本節では 2-CHAP についてのみ説明するが、本節で説明する定式化は一般の k -CHAP の場合にも容易に拡張できる。

3.1 マッチングに基づいた定式化

本章で提案する定式化では、2-CHAP を頂点とチャンネルの間のマッチングとみなす。与えられたグラフ G の頂点数を $n(=|V|)$ とする。2-CHAP の問題例として与えられたグラフが完全グラフであり、その辺の重みが全て 2 の場合に備えて、チャンネルの値は $\{1, 2, \dots, 2n\}$ を用意する。整数線形計画問題として定式化するために、以下の記号を導入する。添字 i は 2-CHAP のグラフ G の頂点に対応し、添字 j は用意されたチャンネルの値に

対応しているとする。変数 x_{ij} は頂点 i に値 j のチャンネルを割り当てた時に 1、そうでない時に 0 とする。値 j 以上のチャンネルを割り当てた頂点が存在する時、値 j のチャンネルは使われているという。変数 y_j は値 j のチャンネルが使われている時に 1 とし、そうでない時に 0 とする。頂点集合とチャンネル集合のマッチングを表したものが図 3 である。以上の変数を導入することにより、2-CHAP は、次の整数線形計画問題に定式化される。目的関数は、使われるチャンネルをなるべく少なくすることを表している。

$$\begin{aligned} \min. & \sum_{j=1}^{2n} y_j, \\ \text{s.t.} & \sum_{j=1}^{2n} x_{ij} = 1 \quad (\forall i \in V), \quad (1) \\ & x_{i_1 j} + x_{i_2 j} \leq 1 \\ & \quad (\forall j \in C, \forall (i_1, i_2) \in E_1), \quad (2) \\ & x_{i_1 j} + x_{i_2 j} + x_{i_1 j+1} + x_{i_2 j+1} \leq 1 \\ & \quad (\forall j \in C \setminus \{2n\}, \forall (i_1, i_2) \in E_2), \quad (3) \\ & x_{ij} \leq y_j \quad (\forall i \in V, \forall j \in C), \quad (4) \\ & 1 \geq y_1 \geq y_2 \geq \dots \geq y_{2n}, \quad (5) \\ & x_{ij}, y_j \in \{0, 1\} \quad (\forall i \in V, \forall j \in C). \quad (6) \end{aligned}$$

ただし $C = \{1, \dots, 2n\}$. $E_1(E_2)$ は G 中の重み 1(2) の枝集合を表している。 V は G の頂点集合を表している。この定式化を定式化 1 と名付ける。

3.2 クリーク不等式の導入による計算時間の短縮

定式化 1 によって作った整数線形計画問題を、汎用の整数線形計画法のソフトウェアに入力し実際に解くと、問題が大きくなるにしたがって計算時間が爆発的に増えると予想される。その理由として、式や変数の数が非常に多いことが考えられる。定式化 1 では、変数の数が $O(n^2)$ 、制約式の数が $O(n^3)$ である。

計算速度を上げるために、まず、変数を減らす工夫を行なった。チャンネル数が $2n$ 個必要なことは減多でない。そこで、簡便な方法で必要なチャンネル数の上界と下界を求め、変数 y_j の個数を減らす。上界は 2-CHAP の近似解法を用いて求めた値を用いた。下界は、問題として与えられたグラフの最大クリークを近似解法で求めて、その値を用いた。これにより、変数 y_j は上界の数だけ用意すれば十分であり、その添字が下界以下のものについては $y_j = 1$ に固定することができる。

次に、制約式を減らすことによって、さらに計算速度を上げることを試みた。特に多い制約式は (2) 式と (3) 式なのでこれらを減らす工夫を行なった。例えば重み 1 の辺に関して、クリークになっている頂点集合は同じチャンネルを使えない。よって、(2) 式と (3) 式は極大な

クリークに対する制約式に置き換えることができる。

以上の手続きにより次の整数線形計画問題を得る。ただし、 u は最適値の上界、 l は最適値の下界である。

$$\min. \quad l + \sum_{j=l+1}^u y_j,$$

$$\text{s.t.} \quad \sum_{j=1}^u x_{ij} = 1 \quad (\forall i \in V), \quad (7)$$

$$\sum \{x_{ij} | i \in S\} \leq 1 \\ (\forall j \in C, S \in Q_1), \quad (8)$$

$$\sum \{x_{ij} + x_{i,j+1} | i \in S\} \leq 1 \\ (\forall j \in C \setminus \{2n\}, S \in Q_2), \quad (9)$$

$$x_{ij} \leq y_j \quad (\forall i \in V, \forall j \in C), \quad (10)$$

$$1 \geq y_{l+1} \geq \dots \geq y_u, \quad (11)$$

$$x_{ij}, y_j \in \{0, 1\} \quad (\forall i \in V, \forall j \in C). \quad (12)$$

ただし Q_1, Q_2 は各々グラフ (V, E_1) と (V, E_2) の極大クリークの族である。この定式化を定式化2と名付ける。定式化2の前処理として、解の上界下界を求める他に、問題例のグラフ $G = (V, E)$ の極大クリークの族を求める必要がある。全ての極大クリークを列挙する方法は参考文献5)で提案されている。参考文献5)の方法は出力される解の個数の多項式時間で計算が終了するものである。

3.3 計算実験の結果

定式化1及び定式化2のもとで、既存のパッケージソフトウェアを用いた。問題例は、1辺の長さ1の正方形上に一様にばらまかれた点をもとにした同心円グラフ(2.2参照)を用いた。同心円グラフの閾値は $r_1 = 0.6, r_2 = 0.3$ とした。以下の表1は頂点数10の問題例であり、表2は頂点数25の問題例である。整数線形計画問題を解くためのソフトウェアはlp_solve2.0を用いた。計算機はSPARC station 20 (150M Hz)を用いた。定式化1では、実行時間内に解ける問題の大きさは頂点数10個程度である。同じ問題例ならば、定式化2を用いることにより、計算時間が大幅に短縮された。しかし定式化2を用いても実行時間内に解ける問題の大きさは頂点数25個程度であった。

4. 近似解法

本節では2-CHAPの近似解法を提案する。P≠NPの場合、頂点彩色問題には多項式時間近似解法は存在しないのでCHAPにも多項式時間近似解法は存在しない⁶⁾。本節では同心円グラフに問題例を限定する。

4.1 頂点除去法

参考文献7)には、単位円グラフに適用すると3-近似

表1 頂点数10の問題例の計算時間。

Table 1 Instances of size of 10.

問題例	計算時間 [秒]	計算時間 [秒]
	(定式化1)	(定式化2)
その1	13.59	0.10
その2	146.18	0.15
その3	10.24	0.10
その4	10.35	0.10
その5	153.86	0.26
その6	68.30	0.24
その7	15.77	0.08
その8	79.73	0.12
その9	105.90	0.12
その10	156.96	0.18
平均	76.09	0.15

表2 頂点数25の問題例の計算時間。

Table 2 Instances of size of 25.

問題例	計算時間 (定式化2) [秒]
その1	3664.23
その2	747.09
その3	16245.87
その4	51.77
その5	1456.83
その6	72.09
その7	2152.99
その8	11699.66
その9	9133.02
その10	301.06
平均	4552.46

である頂点彩色法が紹介されている。以下で提案する頂点除去法は7)の方法の拡張になっている。

頂点除去法の概要を説明する。頂点除去法は頂点の順序を固定した後にチャンネルを割り当てる方法である。頂点の順序を定める際にはグラフにおける頂点除去を行なう。チャンネルは貪欲に割り当てる。

頂点除去法の入力は2-CHAP、すなわち $G_w = (V, E, w)$ であり、出力はチャンネル割当、すなわち関数 $a: V \rightarrow \mathbf{N}$ である。ただし \mathbf{N} は自然数の集合である。 $n = |V|$ である。 $delete(H, v)$ はグラフ H から頂点 v を除去したグラフである。 $ud(v)$ は v に隣接する頂点の中でチャンネルを割り当てられていない頂点の集合である。重みつきグラフ G_w 上で頂点 v に重み1の辺で隣接する頂点の集合を $D^1(v)$ とする。重みつきグラフ G_w 上で頂点 v に重み2の辺で隣接する頂点の集合を $D^2(v)$ とする。頂点集合 U に対して $d^1(v, U) = |D^1(v) \cap U|$, $d^2(v, U) = |D^2(v) \cap U|$, $free(a, v, U) = \{j \in \mathbf{N} | |j - a(u)| \geq 1, \forall u \in U \cap D^1(v), |j - a(u)| \geq 2, \forall u \in U \cap D^2(v)\}$, とする。すなわち $free(a, v, U)$ は重みつきグラフ G_w の頂点部分集合 U にチャンネル割当 a が与えられている状

態で、頂点 v に対して制約条件を満たすチャンネルの集合である。

$taboo(a, v, U) = \mathbf{N} \setminus free(a, v, U)$ とする。

```

(a : V → N) ≡ 頂点除去法 (G_w = (V, E, w)) {
  (V_0, E_0) = G;
  /* 順列の生成 */
  for(i = 0; i < n; i = i + 1) {
    v_{i+1} を min{d^1(v, V_i) + 3d^2(v, V_i) | v ∈ V_i}
    を達成する頂点の中で
    番号の最も小さな頂点とする;
    (V_{i+1}, E_{i+1}) = delete((V_i, E_i), v_{i+1});
  }
  /* 貪欲な割当 */
  for(i = n; i > 0; i = i - 1) {
    a(v_i) = min{free(a, v_i, V_i)};
  }
  return(a);
}

```

上記の頂点除去法において、2行目～6行目は頂点除去であり、7行目～9行目はチャンネル割当である。頂点除去法は以下の定理を満たす。

定理 (頂点除去法の近似率)

頂点除去法は、 $r_1 \geq 2r_2$ なる同心円グラフを問題例とする 2-CHAP に対して 5-近似である。

(証明)

$\{d^1(v_{i+1}, V_i) + 3d^2(v_{i+1}, V_i) | i = 0, \dots, n-1\}$ 中の最大値を達成する i のうち最も小さい i を i^* とする。頂点除去法で得られた解の値を m_A とすると

$$m_A \leq d^1(v_{i^*+1}, V_{i^*}) + 3d^2(v_{i^*+1}, V_{i^*}) + 1 \quad (13)$$

である。なぜならば、 $d^1(v_{i^*+1}, V_{i^*}) + 3d^2(v_{i^*+1}, V_{i^*})$ は頂点除去法において v_i にチャンネルを割り当てる直前の時点において割当に使えないチャンネルの数の上限になっているからである。

グラフ (V_{i^*}, E_{i^*}) を同心円図に描いたとき、その同心円図上で最も x 座標の小さい頂点を v_l とする。明らかに

$$\begin{aligned} d^1(v_{i^*+1}, V_{i^*}) + 3d^2(v_{i^*+1}, V_{i^*}) \\ \leq d^1(v_l, V_{i^*}) + 3d^2(v_l, V_{i^*}) \end{aligned} \quad (14)$$

である。グラフ G に対する 2-CHAP の最適値を m_{OPT} とする。最適値においては

$$m_{OPT} \geq \frac{1}{3}(d^1(v_l, V_{i^*}) + d^2(v_l, V_{i^*})) + 1 \quad (15)$$

$$m_{OPT} \geq d^2(v_l, V_{i^*}) + 1 \quad (16)$$

が成り立つことを示す。これを説明するために記号を定義する。同心円図における v_l の x 座標を x_l 、 y 座標を y_l とする。 (x_l, y_l) を中心とする半径 r_1 の円の周及び

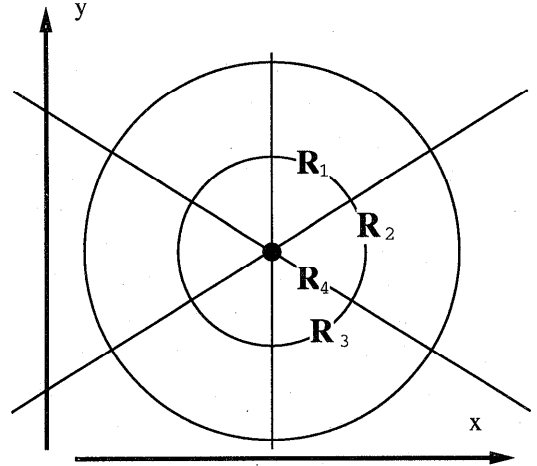


図4 証明の概念図。

Fig. 4 Area R_1, R_2, R_3 and R_4 .

内部でかつ $x \geq x_l$ の領域、すなわち半円を、 $\pi/3$ ラジアンずつ 3 等分する。そのそれぞれを領域 R_1, R_2, R_3 とする。すなわち

$$\begin{aligned} R_k = \{(x, y) | x = x_l + r \cos \theta, y = y_l + r \sin \theta, \\ \frac{1}{2}\pi - \frac{1}{3}\pi k \leq \theta \leq \frac{5}{6}\pi - \frac{1}{3}\pi k, \\ 0 \leq r \leq r_1\} \quad (k = 1, 2, 3), \end{aligned}$$

とする。 (x_l, y_l) を中心とする半径 r_2 の円の周及び内部でかつ $x \geq x_l$ の領域を領域 R_4 とする。すなわち

$$\begin{aligned} R_4 = \{(x, y) | x = x_l + r \cos \theta, y = y_l + r \sin \theta, \\ -\pi/2 \leq \theta \leq -\pi/2, 0 \leq r \leq r_2\}, \end{aligned}$$

とする (図4参照)。 (V_{i^*}, E_{i^*}) 上で v_l に隣接する頂点は領域 R_1, R_2, R_3 のいずれか 1 つに含まれる。領域 R_1, R_2, R_3 のどの領域においても距離が r_1 より大きい頂点対はないので、同じ領域内の全ての頂点には相異なるチャンネルを割り当てなければならない。そして領域 R_1, R_2, R_3 の少なくともいずれか 1 つは $(d^1(v_l, V_{i^*}) + d^2(v_l, V_{i^*}))/3$ 個以上の頂点を含み、その領域中の v_l 以外の頂点にチャンネルを割り当てるには $(d^1(v_l, V_{i^*}) + d^2(v_l, V_{i^*}))/3$ チャンネル以上が必要である。よって式 (15) が成り立つ。また (V_{i^*}, E_{i^*}) 上で v_l に隣接する頂点のうち重み 2 の辺で隣接する頂点は全て領域 R_4 に含まれる。 $r_1 \geq 2r_2$ より、領域 R_4 中にも距離が r_1 より大きい頂点対はないので、 R_4 に含まれる全ての頂点にも相異なるチャンネルを割り当てなければならない。そして v_l 以外の頂点にチャンネルを割り当てるには $d^2(v_l, V_{i^*})$ 以上が必要なので式 (16) が成り立つ。式 (13)～式 (16) を組み合わせると

$$\begin{aligned}
m_A &\leq d^1(v_{*+1}, V_{i^*}) + 3d^2(v_{*+1}, V_{i^*}) + 1 \\
&\leq d^1(v_i, V_{i^*}) + 3d^2(v_i, V_{i^*}) + 1 \\
&= (d^1(v_i, V_{i^*}) + d^2(v_i, V_{i^*})) + 2d^2(v_i, V_{i^*}) + 1 \\
&\leq 3(m_{OPT} - 1) + 2(m_{OPT} - 1) + 1 \\
&= 5m_{OPT} - 4 < 5m_{OPT},
\end{aligned}$$

が成り立つ。よって頂点除去法は5-近似である。(証明終り)

証明の式(13)が成り立つためには、頂点除去法の貪欲な割当において制約条件を満たす最も小さいチャンネルを割り当てることは必ずしも必要ではない。つまり頂点除去法の貪欲な割当は式(13)を満たす手続きのうちの1つでしかない。よって近似率を保ったまま頂点除去法の貪欲な割当に工夫を加えることができる。以下が貪欲な割当の改訂版である。

/* 貪欲な割当の改訂版 */

```

{
  c_max = 0;
  for(i = n; i > 0; i = i - 1){
    a* = min{free(a, v_i, V_i)};
    if(a* > c_max){
      c_max = a*;
      a(v_i) = a*;
    }
  }
  else{
    C = free(a, v_i, V_i);
    a(v_i) を min{f(a, v_i, V_i, c) | c ∈ C}
    を達成するチャンネル c の中で
    最も小さなチャンネル c とする;
  }
}
}

```

```

(整数 x) ≡ f(a, v_i, V_i, c){
  x_1 = ∑_{v ∈ ud(v_i) - V_i} |taboo(a, v, V_i)|;
  a(v_i) = c;
  x_2 = ∑_{v ∈ ud(v_i) - V_i} |taboo(a, v, V_{i-1})|;
  a(v_i) の割当をなくす;
  return(x_2 - x_1);
}

```

頂点除去法の貪欲な割当をその改訂版で置き換えた解法を頂点除去法 II と名付ける。頂点除去法 II は頂点除去法と同じ近似率を保証するだけであって、頂点除去法 II が近似率や計算速度に関して頂点除去法に勝るといふ裏

表3 疎な同心円グラフを入力とする計算実験。

Table 3 Instances of sparse graphs.

問題例	最適値 の 下界	頂点除去法 による 目的関数値	頂点除去法 II による 目的関数値
その1	30	35	34
その2	32	40	39
その3	27	37	36
その4	26	36	36
その5	32	39	37
その6	31	41	41
その7	27	39	34
その8	28	36	37
その9	27	39	39
その10	29	37	34
一番良かった回数	—	4	9

表4 密な同心円グラフを入力とする計算実験。

Table 4 Instances of dense graphs.

問題例	最適値 の 下界	頂点除去法 による 目的関数値	頂点除去法 II による 目的関数値
その1	394	434	426
その2	395	426	412
その3	397	417	413
その4	413	444	433
その5	424	463	457
その6	411	421	419
その7	399	419	418
その8	389	441	414
その9	408	435	426
その10	400	432	422
一番良かった回数	—	0	10

付けはない。しかし頂点除去法はやみくもに小さいチャンネルを割り当てるのに対して、頂点除去法 II は既に割り当てられたチャンネルを考慮に入れているのでより良い解を期待できる。

4.2 計算実験の結果

前節で紹介した頂点除去法の近似率は5であったが、この近似率は最悪の場合に基づいており、最悪の場合は減多に起こらないと思われる。これを確かめるために計算実験を行なった。単位平面上に500個の点を一様にばらまき、その座標をもとにして同心円グラフを生成し、それを問題例とした。表3の問題例は $r_1 = 0.2, r_2 = 0.1$ とした問題例であり、表4の問題例は $r_1 = 1.0, r_2 = 0.5$ とした問題例である。平均計算時間は、表3の場合、頂点除去法は0.49秒、頂点除去法 II は0.66秒、であり表4の場合、頂点除去法は1.12秒、頂点除去法 II は1.73秒、であった。最適値の下界は貪欲法で求めた極大クリークの大きさとした。表3、表4より分かる通り、いずれの例においても近似率は2よりも小さい。また、ほとんどの例において頂点除去法 II は頂点除去法に勝る。

5. 発見的解法

本節では CHAP を解くための発見的解法を提案する。本節で提案する発見的解法はいくつかの構築法と改善法を組み合わせた解法である。5.1節では発見的解法の構成要素となる構築法と改善法をいくつか提案する。5.2節では CHAP に対する発見的解法とその計算実験の結果を示す。

5.1 いろいろな構築法と改善法

本節では次節以降で提案する発見的解法の構成要素となる構築法と改善法をいくつか提案する。本節で提案する解法はいずれも近似率の保証はない。

以下で提案する動的割当法は構築法の1つである。入力は CHAP, すなわち重みつきグラフ $G_w = (V, E, w)$ であり, 出力はチャンネル割当関数 $a: V \rightarrow \mathbf{N}$ である。

```
( $a: V \rightarrow \mathbf{N}$ )  $\equiv$  動的割当法 ( $G_w = (V, E, w)$ ) {
   $U = \emptyset$ ;
  for( $i = 1; i \leq n; i = i + 1$ ) {
     $v^* = \max\{g(a, v, U) | v \in V \setminus U\}$ 
    を達成する頂点の中で
    番号の最も小さな頂点とする;
    /* 貪欲な割当 */
     $a(v^*) = \min\{free(a, v^*, U)\}$ ;
     $U = U \cup v^*$ ;
  }
  return( $a$ );
}
```

```
(整数  $x$ )  $\equiv g(a, v, U)$  {
  return( $n \times |taboo(a, v, U)| + |ud(v)|$ );
}
```

頂点除去法と同様にして動的割当法の貪欲な割当を改訂した解法を動的割当法 II と名付ける。

動的割当法は, 制約条件を満たさないチャンネルの数が多き頂点を優先してチャンネルを割り当てる。結果として, 既にチャンネルを割り当てられた頂点に隣接する頂点が次々とチャンネル割当されてゆくと思われる。

次に提案する改善法 I は改善法の1つである。入力は CHAP とその可能解と繰り返しの回数, すなわち重みつきグラフ $G_w = (V, E, w)$ とチャンネル割当関数 $a: V \rightarrow \mathbf{N}$ と自然数 $ITERATE$ であり, 出力はチャンネル割当関数 $a: V \rightarrow \mathbf{N}$ である。

```
( $a: V \rightarrow \mathbf{N}$ )  $\equiv$  改善法 I ( $G_w = (V, E, w)$ ,
   $a: V \rightarrow \mathbf{N}, ITERATE$ ) {
```

```
   $c_{\max} = \max\{a(v) | v \in V\}$ ;
  while(真) {
     $a' = a$ ;
     $S = \{v | v \in V, a(v) == c_{\max}\}$ ;
     $c_{\max} = c_{\max} - 1$ ;
    for( $i = 1; i \leq ITERATE; i = i + 1$ ) {
       $[1, c_{\max}]$  中の整数を
      ランダムに選び  $c$  とする;
      for( $v \in S$ )  $a(v) = c$ ;
       $S_{\text{new}} = \{v \in V | \exists \{u, v\} \in E, u \in S$ 
        かつ  $a(v) \in taboo(a, v, V)\}$ ;
      if( $S_{\text{new}} == \emptyset$ ) break;
      if( $S_{\text{new}}$  は独立集合) {
         $S = S_{\text{new}}$ ;
         $i = 1$ ;
      }
    }
  }
  if( $i > ITERATE$ ) return( $a'$ );
}
```

改善法 I はチャンネルを選ぶ時に乱数を使っている。これは, 決定論的方法を避けることによって巡回を防ぐためである。繰り返しの回数 $ITERATE$ は許容できる計算時間にしがって決める。

次に提案する 2-OPT は, 巡回セールスマン問題に対する改善法である 2-OPT にヒントを得たものである。入力は CHAP とその可能解と繰り返しの回数, すなわち重みつきグラフ $G_w = (V, E, w)$ とチャンネル割当関数 $a: V \rightarrow \mathbf{N}$ と自然数 $ITERATE$ であり, 出力はチャンネル割当関数 $a: V \rightarrow \mathbf{N}$ である。

```
( $a: V \rightarrow \mathbf{N}$ )  $\equiv$  2-OPT ( $G_w = (V, E, w)$ ,
   $a: V \rightarrow \mathbf{N}, ITERATE$ ) {
   $c_{\max} = \max\{a(v) | v \in V\}$ ;
  for( $i = 1; i \leq ITERATE; i = i + 1$ ) {
     $[1, m]$  中の整数をランダムに選び  $c_1$  とする;
     $[1, m]$  中の整数をランダムに選び  $c_2$  とする;
    if( $c_1 \geq c_2$ ) continue;
     $a = flip(a, c_1, c_2)$ ;
    if( $\exists v \in V, c_1 \leq a(v) \leq c_2$  かつ
       $a(v) \in taboo(a, v, V)$ ) {
       $a = flip(a, c_1, c_2)$ ;
      continue;
    }
  }
  if( $\forall v \in V, a(v) == c_2$ 
```

表5 疎な同心円グラフを入力とする計算実験.
Table 5 Instances of sparse graphs.

問題例	最適値	頂点 除去法	頂点 除去法 II	動的 割当法	動的 割当法 II	頂点除去法 +	頂点除去法 II +	動的割当法 +	動的割当法 II +
	の	+	+	+	+	2-OPT	2-OPT	2-OPT	2-OPT
	下界	改善法 I	改善法 I	改善法 I	改善法 I	+	+	+	+
						改善法 I	改善法 I	改善法 I	改善法 I
その 1	30	35-34	34-34	39-37	35-35	35-35-34	34-34-34	39-38-38	35-35-35
その 2	32	40-39	39-39	39-38	39-39	40-40-39	39-39-39	39-39-39	39-39-39
その 3	27	37-36	36-36	35-34	40-35	37-37-37	36-36-36	35-35-34	40-38-37
その 4	26	36-36	36-36	37-35	37-37	36-36-36	36-36-36	37-37-35	37-37-37
その 5	32	39-38	37-37	38-38	38-38	39-39-38	37-37-37	38-38-38	38-38-38
その 6	31	41-41	41-41	38-38	42-41	41-41-40	41-41-41	38-38-38	42-42-41
その 7	27	39-38	34-34	36-36	36-35	39-38-38	34-34-34	36-36-36	36-36-34
その 8	28	36-35	37-36	35-34	34-34	36-36-35	37-37-37	35-35-35	34-34-34
その 9	27	39-38	38-38	35-35	35-35	39-38-37	39-39-39	35-35-35	35-35-35
その 10	29	37-37	34-34	36-36	39-39	37-37-37	34-34-34	36-36-36	39-39-39
一番 良かった 回数	—	1	4	6	2	1	4	4	3

```

    ならば  $a(v) - 1 \in \text{free}(a, v, V)$  {
       $a = \text{sift}(a, c_2, -1)$ ;
       $c_{\max} = c_{\max} - 1$ ;
    }
  }
  if( $\forall v \in V, a(v) == c_1$ )
    ならば  $a(v) - 1 \in \text{free}(a, v, V)$  {
       $a = \text{sift}(a, c_1, -1)$ ;
       $c_{\max} = c_{\max} - 1$ ;
    }
  }
}
(a :  $V \rightarrow \mathbf{N}$ )  $\equiv \text{flip}(a, c_1, c_2)$  {
  for( $v \in V$ ) {
    if( $c_1 \leq a(v) \leq c_2$ )  $a(v) = c_1 + c_2 - a(v)$ ;
  }
  return( $a$ );
}

(a :  $V \rightarrow \mathbf{N}$ )  $\equiv \text{sift}(a, c, x)$  {
  for( $i = c; i \leq c_{\max}; i = i + 1$ ) {
    for( $v \in V$ ) {
      if( $a(v) == c$ )  $a(v) = a(v) - x$ ;
    }
  }
  return( $a$ );
}

```

2-OPT もチャンネルを選ぶ時に乱数を用いている。理由は改善法 I と同じである。初期解において同じチャンネルを割り当てられた頂点集合は、2-OPT で改善された解

においても同じチャンネルを割り当てられている。よって 2-OPT は初期解の特徴をある程度保存する改善法であるといえる。繰り返しの回数 *ITERATE* は許容できる計算時間によって決める。

5.2 発見的解法

本節で提案する発見的解法は、大きく分けて初期解の生成とその改善の 2 段階に分かれる。改善法を用いる場合、初期解の値が最適値に近い方が良いとは限らない。よって初期解の生成法は頂点除去法、頂点除去法 II、動的割当法、動的割当法 II の 4 つを、改善は改善法 I だけを用いたものと改善法 I + 2-OPT の 2 つを試す。以下の表 5、表 6 が計算実験結果である。問題例は 4.2 節の表 3、表 4 と全く同じである。改善法 I の繰り返し回数は疎なグラフの場合 100 回、密なグラフの場合 1,000 回にした。これは初期解の目的関数値の約 2 倍である。2-OPT の繰り返し回数は 10,000 回にした。それぞれの表において目的関数値は、初期解 - 改善された解 (- さらに改善された解) と表記してある。表 5 より疎なグラフでは動的割当法 + 改善法 I が効果的であることが分かる。表 6 より密なグラフでは頂点除去法 II + 2-OPT + 改善法 I が効果的であることが分かる。

次に、より大きな問題例を解いた時にかかる時間を調べた。問題例は頂点数 10,000 である。手法は疎なグラフに効果的と思われる、動的割当法 + 改善法 I を用いた。改善法 I の繰り返し数は 300 とした。表 7 が計算実験結果である。

6. まとめ

本論文の成果をここにまとめる。本論文の成果は主に第 3 節の厳密解法、第 4 節の近似解法、第 5 節の発見的

表6 密な同心円グラフを入力とする計算実験.

Table 6 Instances of dense graphs.

問題例	最適値	頂点 除去法	頂点 除去法 II	動的 割当法	動的 割当法 II	頂点除去法	頂点除去法 II	動的割当法	動的割当法 II
	の	+	+	+	+	+	+	+	+
	下界	改善法 I	改善法 I	改善法 I	改善法 I	改善法 I	改善法 I	改善法 I	改善法 I
その1	394	434-426	426-421	435-419	425-417	434-422-422	426-415-415	435-414-414	425-415-415
その2	395	426-417	412-410	421-419	432-426	426-416-414	412-409-409	421-417-416	432-419-418
その3	397	417-412	413-410	420-416	422-415	417-411-411	413-407-406	420-417-416	422-416-416
その4	413	444-431	433-428	438-429	446-433	444-429-428	433-425-425	438-426-426	446-428-426
その5	424	463-445	457-449	440-436	440-437	463-441-439	457-442-442	440-435-435	440-434-434
その6	411	421-420	419-418	437-433	439-430	421-421-421	419-419-419	437-427-425	439-426-426
その7	399	419-418	418-416	428-423	426-423	419-417-417	418-413-413	428-420-419	426-422-422
その8	389	441-430	414-414	422-413	404-404	441-421-418	414-412-412	422-414-413	404-402-402
その9	408	435-428	426-422	438-427	431-422	435-420-420	426-418-418	438-427-425	431-420-420
その10	400	432-429	422-422	440-420	443-430	432-428-428	422-421-421	440-418-418	443-420-417
一番 良かった 回数	—	0	1	0	0	0	5	1	3

表7 頂点数10,000の同心円グラフを入力とする計算実験.

Table 7 Instances of size of 10,000.

問題例	目的関数値	計算時間 [秒]
その1	162-157	116.270
その2	146-144	117.080
その3	148-146	111.730
その4	163-153	114.690
その5	149-147	116.100
その6	151-151	113.570
その7	148-146	113.720
その8	145-145	114.450
その9	146-145	116.170
その10	149-149	111.770

解法である。

3節では2-CHAPを頂点とチャンネルのマッチングをしてとらえ、整数線形計画問題として定式化し、パッケージソフトウェアを用いて計算実験を行なって計算時間を計った。3節の定式化の手法は極めてオーソドックスなものである。この定式化を用いて解くことの出来る問題の大きさは、頂点数25個程度であった。

4節では2-CIAPに対する頂点除去法を提案し、頂点除去法が同心円グラフに対して5-近似であることを証明した。また、計算実験により得られる目的関数値は、ほとんどの場合、最適値の1.5倍以内であった。

5節ではCHAPに対する発見的解法を提案し、計算実験を行なった。5節で提案した発見的解法は計算時間が短いのが特徴である。問題例の頂点が10,000個程度ならば1~2分で解を得られることを確かめた。

参考文献

1) Ardal, K.I., Hipolito, A., Van Hoesel, C.P.M. and Hansen, B.: A Branch-and-Cut Algorithm

for the Frequency Assignment, Research Memo 96/001, <http://www.unimaas.nl/~oprres/>.

- 2) Borndörfer, R., Eisenblätter, A., Grötschel, M. and Martin, A.: Frequency Assignment in Cellular Phone Networks, Konrad-Zuse Zentrum für Informationstechnik Berlin (ZIB), Preprint SC 97-35 (1997).
- 3) Sengoku, M., Tamura, H., Shinoda, S., Abe, T., and Kajitani, Y.: Graph Theoretical Considerations of Channel Offset Systems in a Cellular Mobile System, *IEEE Transactions on Vehicular Technology*, Vol. 40, pp. 405-412 (1991).
- 4) Mizuike, T. and Ito, Y.: Optimization of Frequency Assignment, *IEEE Transactions on Communications*, Vol. 37, pp. 1031-1041 (1989).
- 5) Johnson, D.S., Yanakakis, M. and Papadimitriou, C.H.: On Generating all Maximal Independent Sets, *Information Processing Letters*, Vol. 27, pp. 119-123 (1988).
- 6) Lund, C. and Yannakakis, M.: On the Hardness of Approximating Minimization Problems, *Proc. 25th ACM Symposium on Theory of Computing*, pp. 286-293 (1993).
- 7) Marathe, M.V., Breu, H., Hunt, H.B., Ravi, S.S. and Rosenkrantz, D.J.: Simple Heuristics for Unit Disk Graphs, *Networks*, Vol. 25, pp. 59-68 (1995).

(平成10年6月29日受付)

(平成10年8月8日再受付)

(平成10年9月8日採録)



宮本裕一郎

平成8年東京大学工学部計数工学科卒業，平成10年東京大学大学院工学系研究科計数工学専攻修士課程修了，平成10年(株)システム計画研究所入社，現在に至る。組合せ最適化の研究に興味を持つ。日本オペレーションズ・リサーチ学会会員。URL:<http://www.misojiro.t.u-tokyo.ac.jp/~miyamoto/>



松井 知己 (正会員)

昭和62年東京工業大学大学院理工学研究科経営工学専攻修士課程修了，平成2年東京工業大学大学院総合理工学研究科システム科学専攻博士後期課程修了，平成2年東京理科大学理工学部経営工学科助手，平成4年東京大学工学部計数工学科講師，平成8年東京大学大学院工学系研究科計数工学専攻助教授，現在に至る。組合せ最適化の研究に従事。日本オペレーションズ・リサーチ学会会員。URL:<http://www.misojiro.t.u-tokyo.ac.jp/~tomomi/>