論　文

バブルソートグラフにおける素な経路選択アルゴリズム

鈴木　康斗†　　　金子　敬一†

An Algorithm for Disjoint Paths in Bubble-Sort Graphs

Yasuto SUZUKI† and Keiichi KANEKO†

あらまし　n-バブルソートグラフは，正則かつ対称なグラフであり，その頂点数，辺数，連結度，直径はそれぞれ $n!$，$(n-1)n!/2$，$n-1$，$n(n-1)/2$ である．単純で対称な構造をしていることから相互結合網の位相として注目を集めている．本論文では，n-バブルソートグラフを対象とし，頂点から頂点集合への互いに素な経路問題：k-連結グラフにおいて，頂点 $\boldsymbol{s}$ と頂点集合 $D = \{\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_k\}$ が与えられたとき，$\boldsymbol{s}$ を除いて互いに素な k 本の $\boldsymbol{s}$ から $\boldsymbol{d}_i$ $(1 \leq i \leq k)$ への経路を求めよ；を $O(n^5)$ の時間で解くアルゴリズムを提案する．$\boldsymbol{s}$ から D 中の k 個の頂点へ向けてマルチキャストを行う際，たかだか $k-1$ 個の停止型故障（辺または頂点）があっても，少なくとも一つの頂点は $\boldsymbol{s}$ からのメッセージを受信可能である，という耐故障性をこの問題の解決により達成できる．更に本論文では，提案されたアルゴリズムが与える経路の長さの総和が $O(n^3)$ であることも併せて示す．計算機実験の結果，平均時間計算量，経路長の総和の平均はそれぞれ $O(n^{4.7})$，$O(n^{3.0})$ となった．

キーワード　バブルソートグラフ，素な経路，マルチキャスト，耐故障性，多項式オーダアルゴリズム

1. まえがき

グラフにおいて互いに素な経路を求めることは，並列分散計算機システムの設計や実装において重要な問題である [2], [4], [5], [11]〜[13]．グラフにおける頂点から頂点集合への互いに素な経路問題：k-連結グラフにおいて，頂点 $\boldsymbol{s}$ と頂点集合 $D = \{\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_k\}$ が与えられたとき，$\boldsymbol{s}$ を除いて互いに素な k 本の $\boldsymbol{s}$ から $\boldsymbol{d}_i$ $(1 \leq i \leq k)$ への経路を求めよ；は，そのような問題の一つである．$\boldsymbol{s}$ から D 中の k 個の頂点へ向けてマルチキャストを行う際，たかだか $k-1$ 個の停止型故障（辺または頂点）があっても，少なくとも一つの頂点は $\boldsymbol{s}$ からのメッセージを受信可能である，という耐故障性をこの問題の解決により達成できる．この問題は，最大フローアルゴリズムによって頂点数の多項式時間で解くことができる [3]．しかし，頂点が多いグラフでは，このアルゴリズムは非効率的である．n-ハイパキューブや n-スターグラフ，n-ローテータグラフに対しては，この問題をそれぞれ n の多項式時間で解くアルゴリズムや，その耐故障性への応用が提案されている [4], [7], [12]．

n-バブルソートグラフは，$n!$ 個の頂点，$(n-1)n!/2$ 本の辺をもった $(n-1)$-連結の無向グラフである．また，グラフの直径は $n(n-1)/2$ である．バブルソートグラフは，その単純，対称，再帰的な構造から相互結合網の位相として一部で注目を集めている [1], [6], [8]〜[10]．

本論文では n-バブルソートグラフを対象とし，一つの頂点から $n-1$ 個の頂点集合への互いに素な経路問題を考え，n の多項式時間で終了するアルゴリズムを与えることを目的とする．

本論文の構成を以下に示す．**2.** では，本論文で使用する用語の定義を行う．**3.** では，本論文で提案するバブルソートグラフにおける頂点から頂点集合への互いに素な経路問題を解くアルゴリズムを示す．**4.** で提案アルゴリズムの正しさの証明と計算量を示す．**5.** に計算機実験の結果を示し，**6.** で本論文の結論と今後の課題を述べる．

2. 諸　定　義

本章では，本論文で使用する用語の定義を行う．

［定義 1］（隣接交換操作）$\boldsymbol{u} = u_1 u_2 \cdots u_n$ を n 個

† 東京農工大学大学院工学教育部，小金井市
Faculty of Technology, Tokyo University of Agriculture and Technology, 2–24–16 Naka-cho, Koganei-shi, 184–8588 Japan

の相違なるシンボル 1，2，...，n の任意の順列とする．この $\boldsymbol{u}$ と $1 \leq i \leq n-1$ なる i に対し，隣接交換操作 $b_i(\cdot)$ を

$$b_i(\boldsymbol{u}) = u_1 \cdots u_{i-1} u_{i+1} u_i u_{i+2} \cdots u_n$$

で定義する．

［定義 2］（バブルソートグラフ）無向グラフ $G = (V, E)$ に対し，

$$V = \{u_1 u_2 \cdots u_n \mid 1, 2, \ldots, n \text{ の順列 }\},$$

$$E = \{(\boldsymbol{u}, b_i(\boldsymbol{u})) \mid \boldsymbol{u} \in V, 1 \leq i \leq n-1\}$$

のとき，G を n-バブルソートグラフと呼ぶ．

以下では，n-バブルソートグラフを B_n と書く．4-バブルソートグラフ B_4 を図 1 に示す．

B_n 内の頂点のうち，その頂点を表す順列において，右端のシンボルが i であるすべての頂点から導出される部分グラフを $B_{n-1}i$ と書く．部分グラフ $B_{n-1}i$ はそれ自身 $(n-1)$-バブルソートグラフ，B_{n-1}，である（図 1 に示した B_4 において，右端のシンボルが 1 であるすべての頂点から導出されるグラフは B_3 となる）．すなわち，B_n は頂点を共有しない n 個の B_{n-1} を含む．また，B_n は対称グラフ，すなわち任意の 2 頂点に対して一方を他方に写す頂点間の自己同型写像が存在し，なおかつ任意の 2 辺に対して一方を他方に写す辺間の自己同型写像が存在するグラフであり，したがって正則グラフである．B_n の次数，連結度はともに $n-1$ である．

図 2 に，バブルソートグラフにおける最短経路選択アルゴリズムを示す．このアルゴリズムは，n-バブルソートグラフにおける任意の二つの頂点 $\boldsymbol{u}$，$\boldsymbol{v}$ 間の最短経路を与える．以下，各頂点を線形配列で表現し，1 語で n 以下の数を格納可能であると仮定すると，このアルゴリズムの計算量は $O(n^3)$，及び与えられる経路の長さはたかだか $n(n-1)/2$ となる．

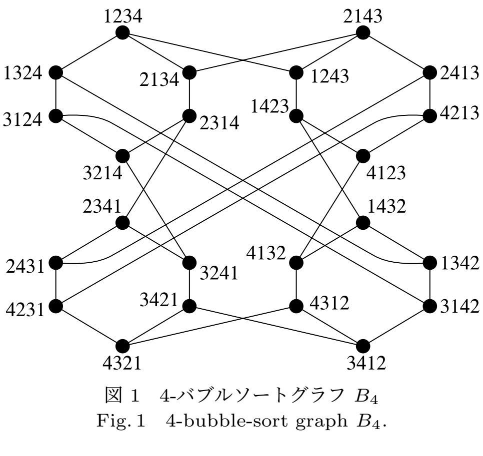

図 1　4-バブルソートグラフ B_4
Fig. 1　4-bubble-sort graph B_4.

［定義 3］（類）B_n の任意の 2 頂点 $\boldsymbol{u}$，$\boldsymbol{v}$ を表す順列において，その順列からシンボル n を除いた順列が同じとき，かつそのときに限り，頂点 $\boldsymbol{u}$，$\boldsymbol{v}$ は同じ類に属するとする．

頂点 $\boldsymbol{v}$ の属する類を $C(\boldsymbol{v})$ で表す．類の性質を以下に示す．

類の性質

(1)　B_n 内の各類は長さ $n-1$ の単純な経路を構成する．

(2)　B_n 内の各頂点はちょうど一つの類に属している．

(3)　B_n 内で右端または左端のシンボルが n である頂点は，その隣接頂点のうち，$n-2$ 個がそれぞれ異なる類に属している．それ以外の頂点は，その隣接頂点のうち，$n-3$ 個がそれぞれ異なる類に属している．

次数 3 のバブルソートグラフ B_4 は，以下のような六つの類をもち，それぞれの類は 4 個の頂点からなる．

{1234, 1243, 1423, 4123},
{1324, 1342, 1432, 4132},
{2134, 2143, 2413, 4213},
{2314, 2341, 2431, 4231},
{3124, 3142, 3412, 4312},
{3214, 3241, 3421, 4321}.

最後に本論文で対象とする問題の定義を与える．

［定義 4］（頂点から頂点集合への互いに素な経路

```
procedure bubble-route(u, v);
begin
  for i := n to 2 step −1 do
    if u_i <> v_i then begin
      find j such that u_j = v_i;
      for l := j to i − 1 do begin
        select Edge (u, b_l(u));
        u := b_l(u)
      end
    end
end;
```

図 2　B_n における最短経路選択アルゴリズム
Fig. 2　Shortest routing algorithm in B_n.

問題）k-連結グラフにおいて，頂点 $\boldsymbol{s}$ と頂点集合 $D=\{\boldsymbol{d}_1,\boldsymbol{d}_2,\ldots,\boldsymbol{d}_k\}$ が任意に与えられたとき，$\boldsymbol{s}$ を除いて互いに素な k 本の $\boldsymbol{s}$ から $\boldsymbol{d}_i$ $(1\le i\le k)$ への経路を求めよ．

3. アルゴリズム

本章では，B_n における頂点から頂点集合への互いに素な経路問題を，再帰を用いて解くアルゴリズムを示す．ただし，$n\le 3$ のとき，この問題は自明であるため，以下では $n\ge 4$ と仮定する．

B_n の対称性より，一般性を失うことなく出発頂点 $\boldsymbol{s}$ を $12\cdots n$ とすることができる．目的頂点の集合を $D=\{\boldsymbol{d}_1,\boldsymbol{d}_2,\ldots,\boldsymbol{d}_{n-1}\}$，目的頂点が属する全類の集まりを $\mathcal{C}=\{C_1,C_2,\ldots,C_k\}$ $(k\le n-1)$ とする．

アルゴリズムを以下のような場合分けで記述する．

Case 1　$D\subset B_{n-1}n$ のとき

Case 2　$D\not\subset B_{n-1}n$ のとき

3.1 Case 1 $D\subset B_{n-1}n$ のとき

(1)　$B_{n-1}n$ において，アルゴリズムを再帰的に呼び出すことにより，$\boldsymbol{s}$ から $\boldsymbol{d}_i$ $(1\le i\le n-2)$ への互いに素な経路を構成する．このとき，$\boldsymbol{d}_{n-1}$ が構成された経路上にあれば（例えば $\boldsymbol{s}$ から $\boldsymbol{d}_h$ への経路上），$\boldsymbol{d}_h$ と $\boldsymbol{d}_{n-1}$ を入れ換え，$\boldsymbol{d}_{n-1}$ から $\boldsymbol{d}_h$ への部分経路を除去する．

(2)　$\boldsymbol{s}$ から $b_{n-1}(\boldsymbol{s})\in B_{n-1}(n-1)$ への辺を選ぶ．

(3)　$\boldsymbol{d}_{n-1}$ の右から2番目のシンボルが $n-1$，すなわち $b_{n-1}(\boldsymbol{d}_{n-1})\in B_{n-1}(n-1)$ ならば，$\boldsymbol{v}=b_{n-1}(\boldsymbol{s})$ とする．さもなくば，以下の (3a) から (3c) を行う．

(3a)　$B_{n-1}(n-1)$ において，右から 2 番目のシンボルが，$\boldsymbol{d}_{n-1}$ のそれと同じである頂点を一つ選び，それを $\boldsymbol{u}$ とする．

(3b)　$B_{n-1}(n-1)$ において，$b_{n-1}(\boldsymbol{s})$ から $\boldsymbol{u}$ への最短経路を選ぶ．

(3c)　$\boldsymbol{u}$ から $\boldsymbol{v}=b_{n-1}(\boldsymbol{u})$ への辺を選ぶ．

(4)　$\boldsymbol{v}$ から $b_{n-1}(\boldsymbol{d}_{n-1})$ への最短経路を選ぶ．

(5)　辺 $(b_{n-1}(\boldsymbol{d}_{n-1}),\boldsymbol{d}_{n-1})$ を選ぶ（図 3 参照）．

3.2 Case 2 $D\not\subset B_{n-1}n$ のとき

(1)　目的頂点を含む各類 C_i $(1\le i\le k)$ において，その類に属する目的頂点のうち，シンボル n が最も右側にあるものからなる集合を D_1 とする．

(2)　目的頂点を含む各類 C_i $(1\le i\le k)$ において，その類に属する $D\backslash D_1$ 中の目的頂点のうち，シンボル n が最も左側にあるものからなる集合を D_2 とする．ただし，$X\backslash Y=\{x\mid x\in X, x\notin Y\}$ とする．

(3)　$D_3=D\backslash D_1\backslash D_2$ とする．

(4)　D_1 中の各目的頂点 $\boldsymbol{d}_i$ に対して，$\boldsymbol{u}_i=B_{n-1}n\cap C(\boldsymbol{d}_i)$ からの最短経路を構成する（図 4 参照）．

(5)　D_3 が空でない場合，D_3 中の各目的頂点 $\boldsymbol{d}_i$ に対して，その隣接頂点のうち，以下の 2 条件を満たす $\boldsymbol{c}_i$ を選び，辺 $(\boldsymbol{c}_i,\boldsymbol{d}_i)$ を選ぶ．$D_3\neq\emptyset$ であることから，$D_2\neq\emptyset$ となり，$|D_1\cup D_3|\le n-2$ が成り立つ．このことと，類の性質 (3) とを併せると，このような $\boldsymbol{c}_i$ を必ず選択することができる．

- $C(\boldsymbol{c}_i)\notin\mathcal{C}$,
- $C(\boldsymbol{c}_i)\neq C(\boldsymbol{c}_j)$ if $i\neq j$.

ただし，$\mathcal{C}$ は，目的頂点が属する全類の集まりであることに注意する．

(6)　(5) で選ばれた各 $\boldsymbol{c}_i$ に対して，$\boldsymbol{u}_i=B_{n-1}n\cap C(\boldsymbol{c}_i)$ からの最短経路を構成する（図 5 参照）．

(7)　D_2 中の各目的頂点 $\boldsymbol{d}_i$ に対して，$C(\boldsymbol{d}_i)$ に

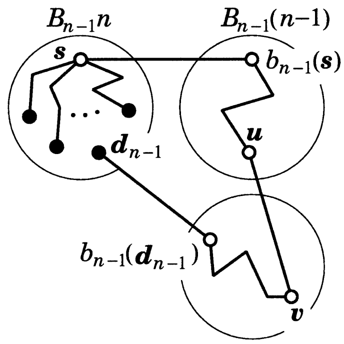

図 3　Case 1 $(b_{n-1}(\boldsymbol{d}_{n-1})\notin B_{n-1}(n-1))$

Fig. 3　Case 1 $(b_{n-1}(\boldsymbol{d}_{n-1})\notin B_{n-1}(n-1))$.

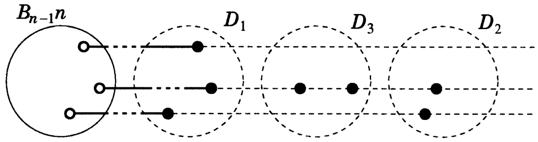

図 4　$B_{n-1}n$ 内の頂点から D_1 内の目的頂点への経路

Fig. 4　Selection of the paths from nodes in $B_{n-1}n$ to the destination nodes in D_1.

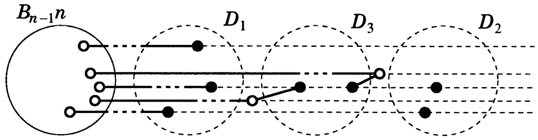

図 5　D_3 内の目的頂点に対する隣接頂点の選択

Fig. 5　Selection of the neighbour nodes for the destination nodes in D_3.

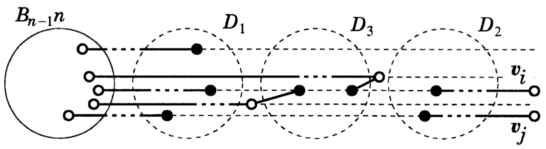

図 6　D_2 内の目的頂点から $\boldsymbol{v}_i$ への最短経路
Fig. 6　Selection of the shortest paths from the destination nodes in D_2 to $\boldsymbol{v}_i$'s.

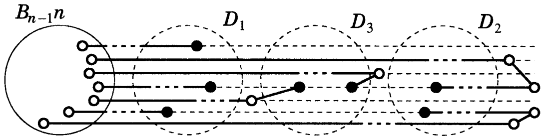

図 7　D_2 内の目的頂点への経路の選択
Fig. 7　Selection of the paths to the destination nodes in D_2.

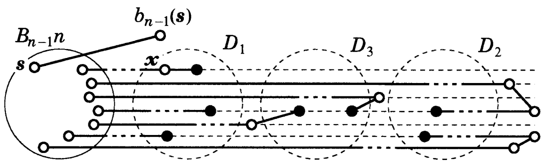

図 8　最近頂点 $\boldsymbol{x}$ の選択
Fig. 8　Selection of the nearest node $\boldsymbol{x}$.

属する頂点のうち，その頂点を表す順列においてシンボル n が左端にある頂点 $\boldsymbol{v}_i$ を選び，$\boldsymbol{v}_i$ から $\boldsymbol{d}_i$ への最短経路を構成する（図 6 参照）.

(8)　(7) で選ばれた各 $\boldsymbol{v}_i$ に対して，その隣接頂点のうち，以下の 2 条件を満たす $\boldsymbol{c}_i$ を選び，辺 $(\boldsymbol{c}_i, \boldsymbol{v}_i)$ を選ぶ．このような $\boldsymbol{c}_i$ は，類の性質 (3) と $|D| = n-1$ から必ず選択することができる．

- $C(\boldsymbol{c}_i) \notin \mathcal{C}$,
- $C(\boldsymbol{c}_i) \neq C(\boldsymbol{c}_j)$ if $i \neq j$.

ただし，$\mathcal{C}$ は，目的頂点が属する全類の集まりであることに注意する．

(9)　(8) で選ばれた各 $\boldsymbol{c}_i$ に対して，$\boldsymbol{u}_i = B_{n-1}n \cap C(\boldsymbol{c}_i)$ からの最短経路を構成する（図 7 参照）.

(10)　$\boldsymbol{s} \in \bigcup \boldsymbol{u}_i$ のとき，$\boldsymbol{s} = \boldsymbol{u}_{i^*}$ として (13) へ進む．さもなくば辺 $(\boldsymbol{s}, b_{n-1}(\boldsymbol{s}))$ を選ぶ．

(11)　上で構成した経路上の頂点のうち，$B_{n-1}n$ 外にあり，$b_{n-1}(\boldsymbol{s})$ に最も近いものを一つ選び，それを $\boldsymbol{x}$ とする（図 8 参照）．ここで，$\boldsymbol{x}$ は $\boldsymbol{u}_{i^*}$ から $\boldsymbol{d}_{i^*}$ への経路上の頂点と仮定する．

(12)　$\boldsymbol{u}_{i^*}$ から $\boldsymbol{x}$ への部分経路を捨て，$b_{n-1}(\boldsymbol{s})$ から $\boldsymbol{x}$ へ，`bubble-route` により最短経路を構成する（図 9 参照）.

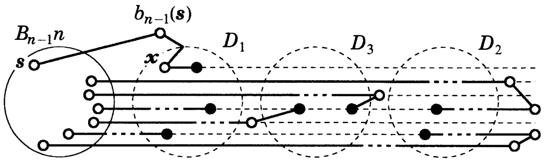

図 9　$b_{n-1}(\boldsymbol{s})$ から $\boldsymbol{x}$ への経路選択と $\boldsymbol{u}_{i^*}$ から $\boldsymbol{x}$ への部分経路の削除
Fig. 9　Selection of the path from $b_{n-1}(\boldsymbol{s})$ to $\boldsymbol{x}$ and deletion of the subpath from $\boldsymbol{u}_{i^*}$ to $\boldsymbol{x}$.

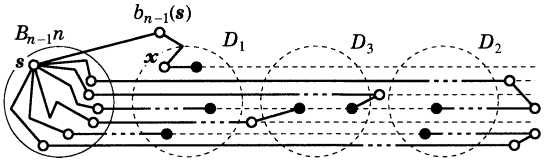

図 10　アルゴリズムの再帰呼出し
Fig. 10　Recursive call of the algorithm.

(13)　$B_{n-1}n$ において，アルゴリズムを再帰的に呼び出すことにより，$\boldsymbol{s}$ から $\bigcup \boldsymbol{u}_i \backslash \boldsymbol{u}_{i^*}$ への互いに素な経路を構成し，終了する（図 10 参照）.

4.　正しさの証明と計算量

本章では，提案アルゴリズムの正しさの証明と計算量について述べる．以下では，n-バブルソートグラフに対する提案アルゴリズムの時間計算量を $T(n)$，得られる経路長の総和の上界を $L(n)$ とする．

［補題 1］　Case 1 のときに構成される $n-1$ 本の経路は $\boldsymbol{s}$ を除いて互いに素である．これを構成するための時間計算量は $T(n) = T(n-1) + L(n-1) \times O(n) + O(n^3)$ であり，経路長の総和はたかだか $L(n) = L(n-1) + n^2 - 3n + 5$ である．

（証明）　(1) で構成される $n-2$ 本の経路は，帰納法の仮定により $\boldsymbol{s}$ を除いて互いに素である．(2) から (5) で構成される経路上の頂点は，$\boldsymbol{s}$ と $\boldsymbol{d}_{n-1}$ を除きすべて $B_{n-1}n$ 外にある．したがって，(1) で構成される経路と (2) から (5) で構成される経路は $\boldsymbol{s}$ を除いて互いに素である．

(1) から (5) の時間計算量は，それぞれ $T(n-1) + L(n-1) \times O(n)$，$O(n)$，$O(n^3)$，$O(n^3)$，$O(n)$ である．したがって，Case 1 全体の時間計算量は $T(n) = T(n-1) + L(n-1) \times O(n) + O(n^3)$ となる．

(1) から (5) で構成される経路の長さの総和は，それぞれたかだか $L(n-1)$，1，$(n-1)(n-2)/2 + 1$，$(n-1)(n-2)/2$，1 である．したがって，Case 1 で構成される経路の長さの総和はたかだか $L(n) =$

$L(n-1)+n^2-3n+5$ となる． □

［補題 2］ Case 2 のときに構成される $n-1$ 本の経路は $\boldsymbol{s}$ を除いて互いに素である．これを構成するための時間計算量は $T(n)=T(n-1)+O(n^4)$ であり，経路長の総和はたかだか $L(n)=L(n-1)+(3n-2)(n-1)/2$ である．

（証明） (4) で構成される最短経路は，同じ類の頂点からなる経路となるため，i が異なれば互いに素となる．(5) で選ばれる各辺は，$\boldsymbol{c}_i$ の条件により (4) で選ばれる経路とは互いに素である．(6) で選ばれる経路は，(4) で選ばれる経路と同じ理由により互いに素である．(4) から (6) で選ばれる経路は，(5) 及び (6) で選ばれる経路の接続点 $\boldsymbol{c}_i$ を除いて互いに素である．(7) で選ばれる経路は (4) で選ばれる経路と同様の理由で互いに素であり，D_2 の定義より，(4) から (6) で選ばれた経路とも互いに素である．(8) 及び (9) で選ばれる経路は，(5) 及び (6) で選ばれる経路と同様の理由により接続点 $\boldsymbol{c}_i$ を除いて互いに素である．(8) 及び (9) で選ばれる経路と (4) から (7) で選ばれる経路とは，(7) で選ばれる経路との接続点 $\boldsymbol{v}_i$ を除いて互いに素である．(10) で $\boldsymbol{s} \in \bigcup \boldsymbol{u}_i$ のときに選ばれる辺 $(\boldsymbol{s}, b_{n-1}(\boldsymbol{s}))$ が，(4) から (9) で選んだ経路と互いに素であることは明らかである．(11) で選ばれる頂点 $\boldsymbol{x}$ は，(4) から (9) で選んだ経路上の頂点の中で $b_{n-1}(\boldsymbol{s})$ に最も近いものなので，(11) で選ばれる経路と (4) から (9) で選ばれる経路は $\boldsymbol{x}$ を除いて互いに素である．また，(10) で選ばれる辺と (11) で選ばれる経路はその接続点 $b_{n-1}(\boldsymbol{s})$ を除いて互いに素である．(13) で選ばれる経路は，帰納法の仮定より $\boldsymbol{s}$ を除いて互いに素である．(4) から (12) までで選ばれる経路上の頂点は，$\boldsymbol{s}$ と $\boldsymbol{u}_i$ を除いて $B_{n-1}n$ の外にあり，(13) で選ばれる経路上の頂点はすべて $B_{n-1}n$ 内にある．したがって，(4) から (12) で選ばれる経路と，(13) で選ばれる経路は $\boldsymbol{s}$ とその接続点 $\boldsymbol{u}_i$ を除いて互いに素である．

(1) から (13) の時間計算量は，それぞれ $O(n^3)$, $O(n^2)$, $O(n)$, $O(n^4)$, $O(n^4)$, $O(n^4)$, $O(n^4)$, $O(n^4)$, $O(n^4)$, $O(n^2)$, $O(n^4)$, $O(n^3)$, $T(n-1)$ である．したがって，Case 2 全体の時間計算量は $T(n)=T(n-1)+O(n^4)$ となる．

(4) から (9) において構成される経路長の総和は，たかだか $(n-1)^2$ である．(10)，(12)，(13) で選ばれる経路長の総和は，それぞれたかだか 1，$n(n+1)/2$，$L(n-1)$ である．また，(12) においては (4) から (9) で構成した経路を少なくとも 1 だけ短くする．したがって，Case 2 で構成される経路の長さの総和はたかだか $L(n)=L(n-1)+(3n-2)(n-1)/2$ となる． □

補題 1 及び 2 より，以下の定理が直ちに導かれる．

［定理 1］ 提案アルゴリズムが与える経路は $\boldsymbol{s}$ を除いて互いに素である．また，計算量は $O(n^5)$，経路長の総和はたかだか $(n^3-n^2-8)/2$ である．

バブルソートグラフの平均距離は $O(n^2)$ であるため，定理 1 より，提案アルゴリズムによって得られる経路長の総和のオーダ $O(n^3)$ は最適となる．

5. 計算機実験

アルゴリズムの平均性能を評価するために，以下のような手順で計算機実験を行った．

・n を 2 から 50 まで変化させて，次の 1，2 を 1,000 回繰り返す．

(1) 出発頂点 $12\cdots n$ を除いて目的頂点集合 D

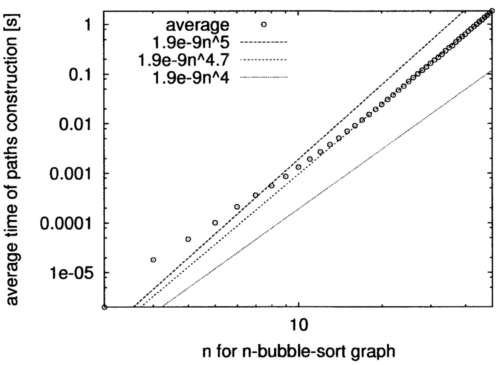

図 11 平均時間計算量
Fig. 11 Average time complexity.

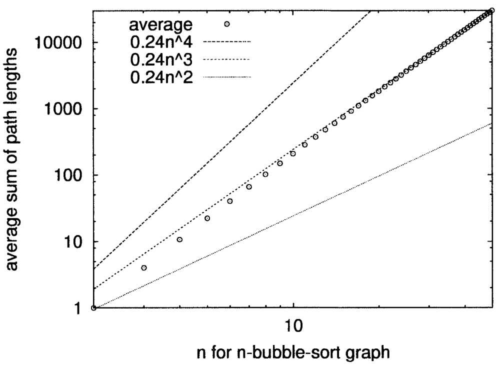

図 12 経路長の総和の平均
Fig. 12 Average sum of path lengths.

を無作為に選ぶ．

(2) アルゴリズムを適用し，実行時間と得られた経路長を測定する．

プログラムは手続き型言語 C で書き，コンパイラには gcc を使用した（オプションは-O2）．実験は，CPU が Intel Celeron 700 MHz，メモリが 256 MByte の計算機上で行った．図 11 及び図 12 に，平均時間計算量と経路長の総和の平均を示す．これらの図より，平均時間と経路長の総和の平均が n の多項式オーダであり，それぞれ $O(n^{4.7})$，$O(n^{3.0})$ 程度であることが分かる．

6. むすび

本論文では，n-バブルソートグラフにおける頂点から頂点集合への互いに素な経路問題を，n の多項式時間 $O(n^5)$ で解くアルゴリズムを与え，そのアルゴリズムによって得られる経路の長さの総和が $O(n^3)$ であることを示した．また，計算機実験の結果，提案アルゴリズムの平均時間計算量，及び経路長の総和の平均はそれぞれ $O(n^{4.7})$，$O(n^{3.0})$ となった．

Case 2 の (5)，(8) において，D_2，D_3 中の各目的頂点に対して隣接頂点 $\boldsymbol{c}_i$ を求める際に条件 ‘$C(\boldsymbol{c}_i) \in \mathcal{C}$’ の判定を工夫し，更に経路の構成及び出力の方法を頂点列挙から，隣接交換操作の添字の列挙に変更するなどの方法により，計算量を $O(n^4 \log n)$ へ減らすことが今後の課題である．

謝辞 本論文に対して貴重な意見を頂いた査読者の方々に深謝する．本研究の一部は，日本学術振興会科学研究費補助金（基盤研究 (C)(2)）課題番号 16500015 及び同（特別研究員奨励費）による．

文献

[1] S.B. Akers and B. Krishnamurthy, “A group-theoretic model for symmetric interconnection networks,” IEEE Trans. Comput., vol.38, no.4, pp.555–566, April 1989.

[2] M. Dietzfelbinger, S. Madhavapeddy, and I.H. Sudborough, “Three disjoint path paradigms in star networks,” Proc. 3rd IEEE Symposium on Parallel and Distributed Processing, pp.400–406, 1991.

[3] J.T. Gross and J. Yellen, Graph theory and its applications, CRC Press, 1998.

[4] Q. Gu and S. Peng, “Node-to-set disjoint paths problem in star graphs,” Inf. Process. Lett., vol.62, no.4, pp.201–207, 1997.

[5] Y. Hamada, F. Bao, A. Mei, and Y. Igarashi, “Non-adaptive fault-tolerant file transmission in rotator graphs,” IEICE Trans. Fundamentals, vol.E79-A, no.4, pp.477–482, April 1996.

[6] J.S. Jwo, “Properties of star graph, bubble-sort graph, prefix-reversal graph and complete-transposition graph,” J. Inf. Sci. Eng., vol.12 no.4, pp.603–617, 1996.

[7] K. Kaneko and Y. Suzuki, “An algorithm for node-to-set disjoint paths problem in rotator graphs,” IEICE Trans. Inf. & Syst., vol.E84-D, no.9, pp.1155–1163, Sept. 2001.

[8] K. Kaneko and Y. Suzuki, “Node-to-node internally disjoint paths problem in bubble-sort graphs,” Proc. 2004 Pacific Rim International Symposium on Dependable Computing, pp.173–182, 2004.

[9] S. Lakshmaivarahan, J.S. Jwo, and S.K. Dhall, “Symmetry in interconnection networks based on Cayley graphs of permutation groups: A survey,” Parallel Comput., vol.19, pp.361–407, 1993.

[10] S. Latifi and P.K. Srimani, “Transposition networks as a class of fault-tolerant robust networks,” IEEE Trans. Comput., vol.45, no.2, pp.230–238, 1996.

[11] S. Madhavapeddy and I.H. Sudborough, “A topological property of hypercubes: Node disjoint paths,” Proc. 2nd IEEE Symposium on Parallel and Distributed Processing, pp.532–539, 1990.

[12] M.O. Rabin, “Efficient dispersal of information for security, load balancing, and fault tolerance,” J. Association for Computing Machinery, vol.36, no.2, pp.335–348, 1989.

[13] P.D. Seymour, “Disjoint paths in graphs,” Discrete Mathematics, vol.29, pp.293–309, 1980.

（平成 16 年 9 月 7 日受付，10 月 22 日再受付）

鈴木　康斗　（学生員）

平 13 東京農工大・工・電子情報卒．現在，同大大学院博士後期課程在学中．相互結合網の経路選択に関する研究に従事．

金子　敬一　（正員）

昭 60 東大・工・計数卒．昭 62 同大大学院修士課程了．同年同大・工・計数助手．平 8 千葉大・工・情報講師．平 11 東京農工大・工・情報コミュニケーション助教授．工博．関数プログラミング言語，並列分散計算，部分計算，ディペンダブルコンピューティング等の研究に従事．IEEE，ACM，情報処理学会，日本ソフトウェア科学会各会員．