数理解析研究所講究録
1054 巻 1998 年 40-53

$(4,7)$–、$(5,6)$–マージングネットワークの最小比較器数のコンピュータによる計算

電気通信大学　丹野 岳久 (Takehisa Tanno)
電気通信大学　岩田 茂樹 (Shigeki Iwata)

(m,n)–マージングネットワークの最小比較器数を $M(m,n)$ で表す。本稿では $M(4,7)=16$ 、$M(5,6)=16$ を示す。いずれもマージングネットワークが 15 比較器からなると仮定し矛盾を導く。考えられうるすべての 15 比較器からなるネットワークを構成し、そのすべてのネットワークがマージングネットワークでないことをコンピュータで検証した。コンピュータの計算において分散処理の手法を用い複数の計算機を使用した。

1　まえがき

m 個からなる要素 $x_1 \le x_2 \le \cdots \le x_m$ と n 個からなる要素 $y_1 \le y_2 \le \cdots \le y_n$ を入力とし $m+n$ 個の要素を出力するネットワークを考える。出力が $m+n$ 個の要素 $z_1 \le z_2 \le \cdots \le z_{m+n}$ のとき、このネットワークを (m,n)–**マージングネットワーク**あるいは単に**マージングネットワーク**という。マージングネットワークはいくつかの**比較器**からなっている。比較器は図 1 の形をしており、それ自身が $(1,1)$–マージングネットワークである。(m,n)–マージングネットワークを構成するのに必要な最小比較器数を $M(m,n)$ とする。

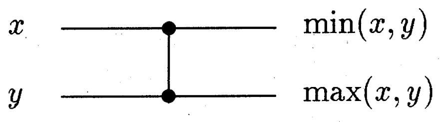


図 1: 比較器

Batcher [1] は odd-even 法により (m,n)–マージングネットワークの最小比較器数の上界を示した。また Yao and Yao [2] は $M(2,n)=\lceil 3n/2 \rceil$ を、Aigner and Schwarzkopf [3] は $M(3,n)=\lceil (7n+3)/4 \rceil$ を示した。最近、増田、岩田 [4] は $M(4,5)=12$ を証明し、さらにコンピュータを用いた計算により $M(4,6)=14, M(4,8)=17$ を示した。コンピュータの計算による下界の証明は、9 要素のソーティングネットワークで深さ 6 のネットワークが存在しないことを求めた Parberry の計算 [5] が知られている。

本稿では $M(4,7)=16$, $M(5,6)=16$ を示す。$15 \le M(4,7) \le 16$, $15 \le M(5,6) \le 16$ であることが知られていた。本稿ではどちらのネットワークも 15 比較器で構成できると仮定し矛盾を導き出す。いずれの場合もコンピュータの計算により下界を求めた。

このような種類の計算は基本的には木の探索を行うことである。しかし木の大きさが膨大になり、単純な計算方法では計算時間がかかりすぎて、合理的な時間内に計算が終了しない。そこで枝刈りや計算を早めるための手法をどのように用いるかが重要となり、それ無しには合理的時間内に計算が完了できないことになる。

(m,n)–マージングネットワークにおいて、入力要素が $z_i \in \{x_1, x_2, \cdots, x_m, y_1, y_2, \cdots, y_n\}$ である水平の線を**入力行** z_i と呼ぶ。マージングネットワークにおいては、入力行 x_i の添字の値が大きい入力線ほどネットワークの下位に位置し入力行 y_i についても同様である。任意の入力行の並びをもつマージングネットワークから他の入力行の並びをもつマージングネットワークに比較器の数を保存する変換が存在する ([6, p.239, Exercise 16] を参照)。比較器を表すのに $[z_i : z_j]$ と書く。これは比較器の上端が入力行 z_i であり、下端が入力行 z_j であることを示す。また $\uparrow z_i$ や $\downarrow z_i$ も入力行を表し$\uparrow z_i$ は比較器の上端あるいは下端が入力行 z_i かそれより上側にあること、また$\downarrow z_i$ は比較器の上端あるいは下端が入力行 z_i かそれより下側にあることを表す。

z_i, z_j, z_k, z_ℓ をそれぞれ上から i, j, k, ℓ 番目の入力線を表すとする。そのとき 2 比較器 $\alpha = [z_i : z_j], \beta = [z_k : z_\ell]$ に対して$\alpha \le \beta$ が成り立つとは、$(i < k) \vee [(i = k) \wedge (j < \ell)]$ が満たされるときである。つまり、α の上端が β の上端よりもネットワーク上の上位にあるか、α と β の上端が同じ入力行でかつ α の下端が β の下端よりもネットワーク上の上位にあるときである。同じ入力行に連結する 2 比較器 α, β に対して、α が β より入力側に近い方にあるとき $\alpha \prec \beta$ とかく。

マージングネットワーク中で入力行 x_i, y_j を含む隣接する複数の入力行とそれら複数の入力行中に含まれる比較器で作られたネットワークを**サブネットワーク**という。このとき次の補題が成り立つ。

補題 1　サブネットワークはマージングネットワークである。

(証明)　マージングネットワークを N、N のサブネットワークを N' とする。いま N' がマージングネットワークでないとする。ある入力 I が存在して、I は N'でソートされない。いま N の N' より上側に位置する入力行には I 中の最小の要素よりも小さい値を、N' より下側に位置する入力行には I 中の最大の要素よりも大きい値を入力しまた N' には I を入力とする。このような入力のもと N においては N' への入力と N' 以外の入力線の間に交換が起きないことに注意する。仮定より N' はマージングネットワークではないので N' の中はソートされない。すなわちこの入力のもとでは N は入力をソートせず、したがって Nはマージングネットワークではない。これは最初の仮定と矛盾する。よって N' はマージングネットワークである。□

隣接する 2 入力行を結ぶ比較器を**隣接比較器**と呼ぶ。隣接する 2 入力行 x_i, y_j が存在するならその 2 入力行からなるサブネットワークは $(1,1)$–マージングネットワークであるのでその 2 入力行間には隣接比較器が存在する。隣接比較器以外の比較器を**非隣接比較器**と

呼ぶ。どのような入力に対しても入力要素の入れ替えが起きない比較器を**冗長比較器**と呼ぶ。$M(m,n)$ 個の比較器で構成される (m,n)–マージングネットワークを (m,n)–**最適マージングネットワーク**よとぶ。明らかに次の補題が成り立つ。

補題 2　(m,n)–最適マージングネットワークは冗長比較器を含まない。

補題 3　上 3 行の入力行が $x_1, y_1, z,\ z \in \{x_2, y_2\}$ である (m,n)–最適マージングネットワークにおいて

(1) 隣接比較器 $[x_1 : y_1]$ はただ 1 個存在する。

(2) $[x_1 :\downarrow z]$ がネットワーク中に存在するなら $[x_1 :\downarrow z] \prec [x_1 : y_1]$ が成り立つ。

(証明) (1) 入力行 x_1, y_1 からなる N のサブネットワークは $(1,1)$–マージングネットワークであり、隣接比較器 $[x_1 : y_1]$ が少なくとも 1 個存在する。2 個以上の $[x_1 : y_1]$ の形の比較器 $\alpha, \beta_1, \cdots, \beta_n$ が N に存在し $\alpha \prec \beta_1 \prec \cdots \prec \beta_n$ と仮定する。このとき α より出力側に位置する比較器 $\beta_1, \cdots, \beta_n$ ではどのような入力に対しても入力要素の入れ換えは起きない。よって $\beta_1, \cdots, \beta_n$ は冗長比較器であり、補題2より N が (m,n)–最適マージングネットワークであることと矛盾する。

(2) N の中に $\alpha = [x_1 : y_1]$, $\beta = [x_1 :\downarrow z]$ が存在し $\alpha \prec \beta$ であると仮定する。すると x_1 と y_1 はどちらも x と y の最小の値であるので (1) と同様に β では入力要素の入れ替えが起きない。ゆえに β は冗長比較器である。よって $\beta \prec \alpha$ が成立する。□

(m,n)–マージングネットワークのすべての入力行に対して比較器が少なくとも 1 個は接続しているので次の補題が成り立つ。

補題 4　$\min(M(m+1,n), M(m,n+1)) \geq M(m,n) + 1$

補題 5　入力行が x_1, x_2, y_1 である $(1,2)$–最適マージングネットワークは図 2の 2 通りである。

(証明) 入力行 x_2, y_1 からなるサブネットワークは $(1,1)$–マージングネットワークであるので隣接比較器 $[x_2 : y_1]$ が存在する。$M(1,2) = 2$ よりもう一つの比較器は $[x_1 :\downarrow x_2]$ である。この時考えられる比較器の形と位置関係は次の 4 通りである。$[x_2 : y_1] \prec [x_1 : x_2], [x_1 : x_2] \prec [x_2 : y_1], [x_2 : y_1] \prec [x_1 : y_2], [x_1 : y_2] \prec [x_2 : y_1]$ 。しかし $[x_2 : y_1] \prec [x_1 : x_2]$ と $[x_1 : y_2] \prec [x_2 : y_1]$ の場合、入力要素 y_1 の値が最小のとき入力線 x_1に移動できない。よって $(1,2)$–最適マージングネットワークは図 2の 2 通りである。□

補題 6　入力線の並びが $x_1, x_2, y_1, x_3, y_2, x_4$である $(2,4)$–最適マージングネットワークにおいて $[y_1 : x_3]$ という形の比較器を 2 個持つネットワークは存在しない。

(証明) $M(2,4) = 6$ 個の比較器からなる $(2,4)$–最適マージングネットワーク N が $[y_1 : x_3]$ の形の比較器を 2 個有すると仮定する。補題 5と N の隣接比較器を考えると N は $[x_2 : y_1], [x_1 : z], (z = x_2$または$z = y_1), [x_3 : y_2], [y_2 : x_4]$ と 2 個の $[y_1 : x_3]$ からなる。このとき入力要素が移動するのに必要な比較器の位置関係について考える。入力要素 y_1 が入力線 y_2

に移動する場合を考えると $[y_1 : x_3] \prec [x_3 : y_2]$ である。また入力要素 y_2 が入力線 x_2 に移動する場合もあるので $[x_3 : y_2] \prec [y_1 : x_3] \prec [x_2 : y_1]$ である。入力要素 x_2 が入力線 x_3 に移動する場合を考えると $[x_2 : y_1] \prec [y_1 : x_3]$ である。しかしこれらの 3 条件を同時に満たすように N 中に 2 個の $[y_1 : x_3]$ を配置することができず、そのような N は存在しない。□

補題 7 [7]　$(3,3)$−最適マージングネットワークは図 3の 4 通りである。

補題 8 [4]　$(3,4)$−最適マージングネットワークは図 4の 18 通りとそれぞれの図の比較器の位置を入力行 y_2 に対して上下反転させた図 5の 18 通りの合計 36 通りである。また入力行を上から $x_1, x_2, y_1, x_3, y_2, x_4, y_3$ とした $(3,4)$−最適マージングネットワークは図 6とそれを上下反転した図 7の合計 36 通りである。

図 2: $(1,2)$−最適マージングネットワーク

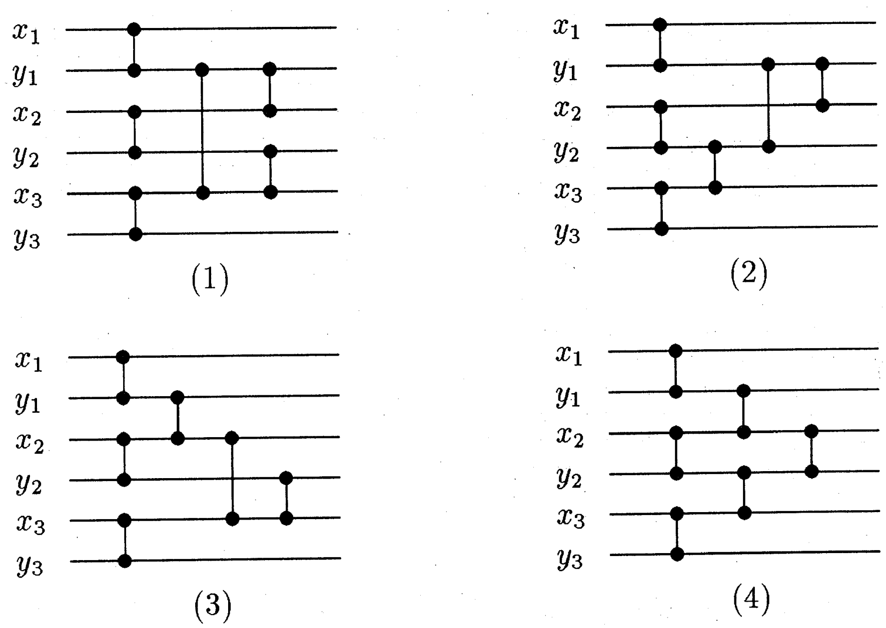
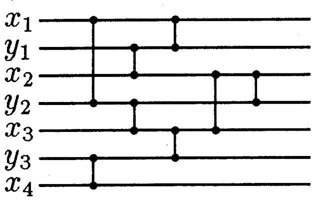
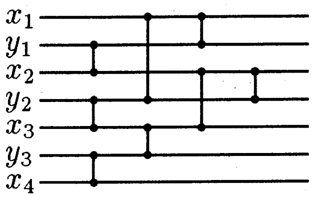


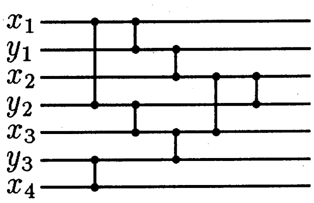
図 3: $(3,3)$−最適マージングネットワーク



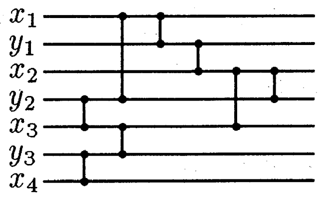
(1)



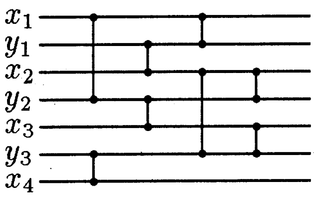
(2)



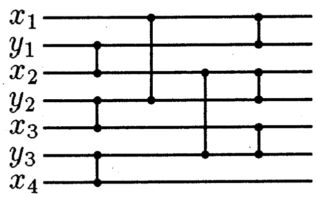
(3)



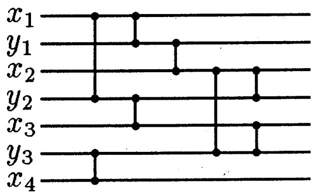
(4)



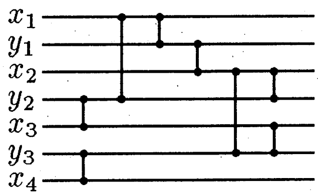
(5)



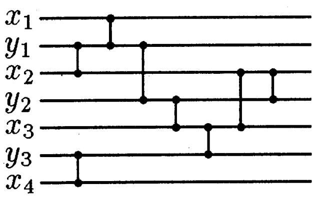
(6)



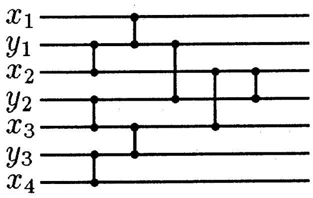
(7)



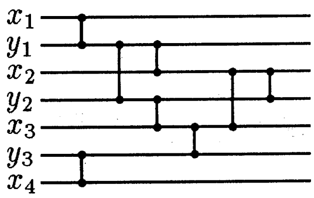
(8)



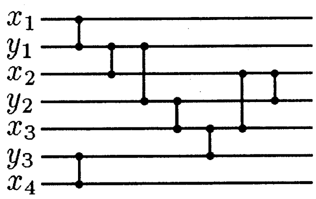
(9)



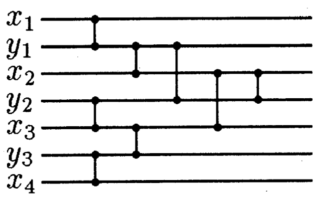
(10)



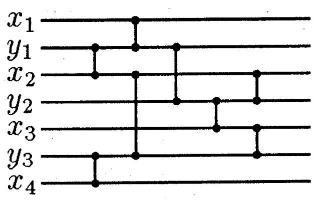
(11)



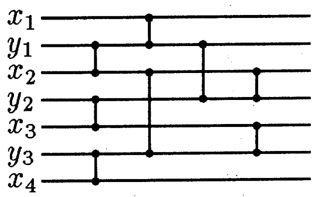
(12)



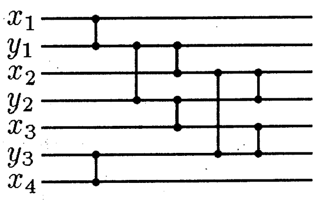
(13)



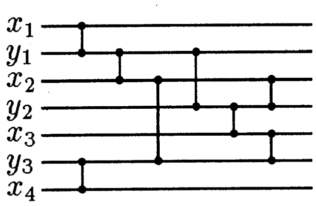
(14)



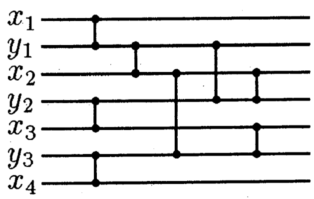
(15)



(16)



(17)



(18)

図 4: (3,4)–最適マージングネットワーク (その 1)

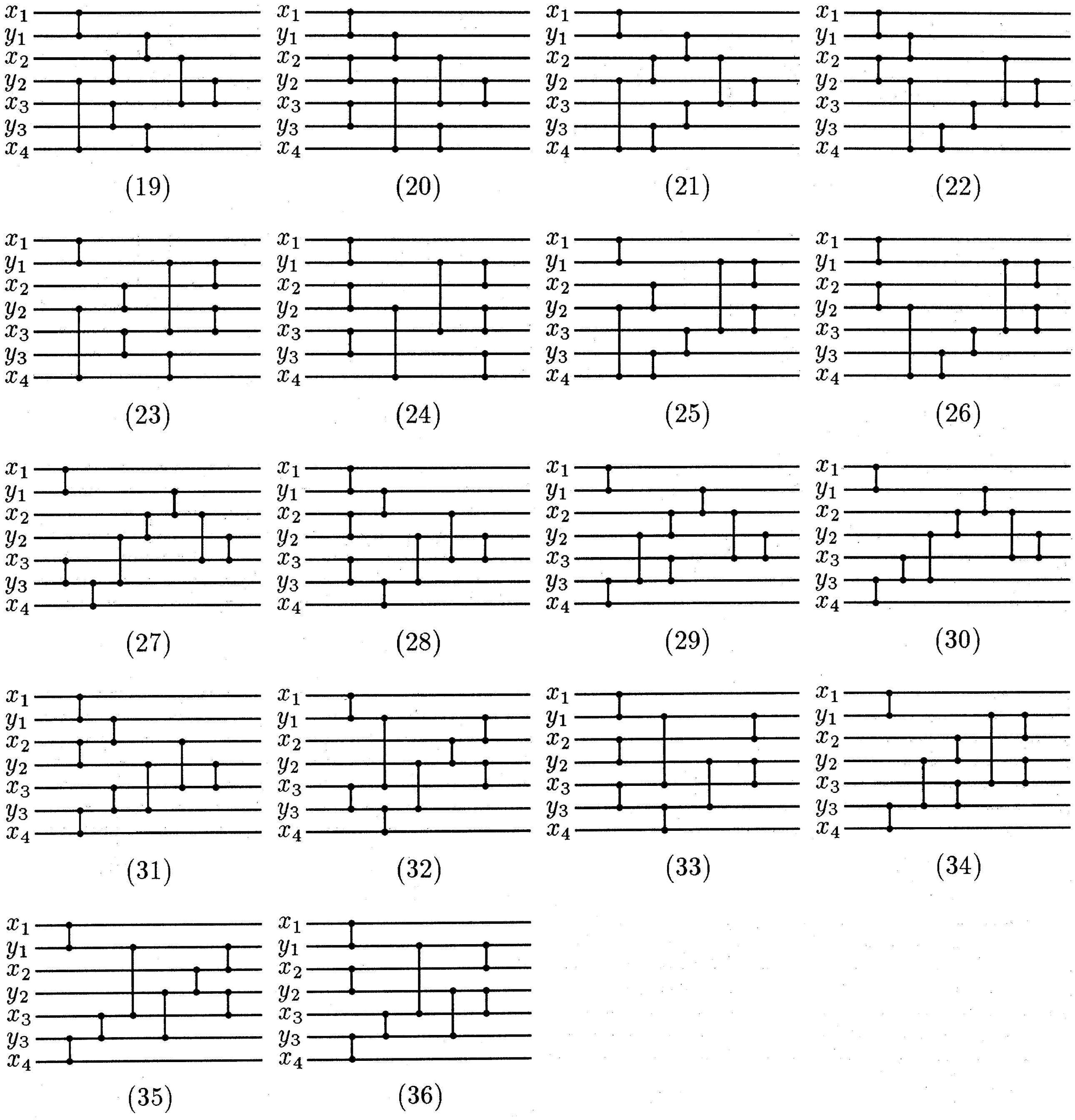


図 5: (3,4)–最適マージングネットワーク (その 2)

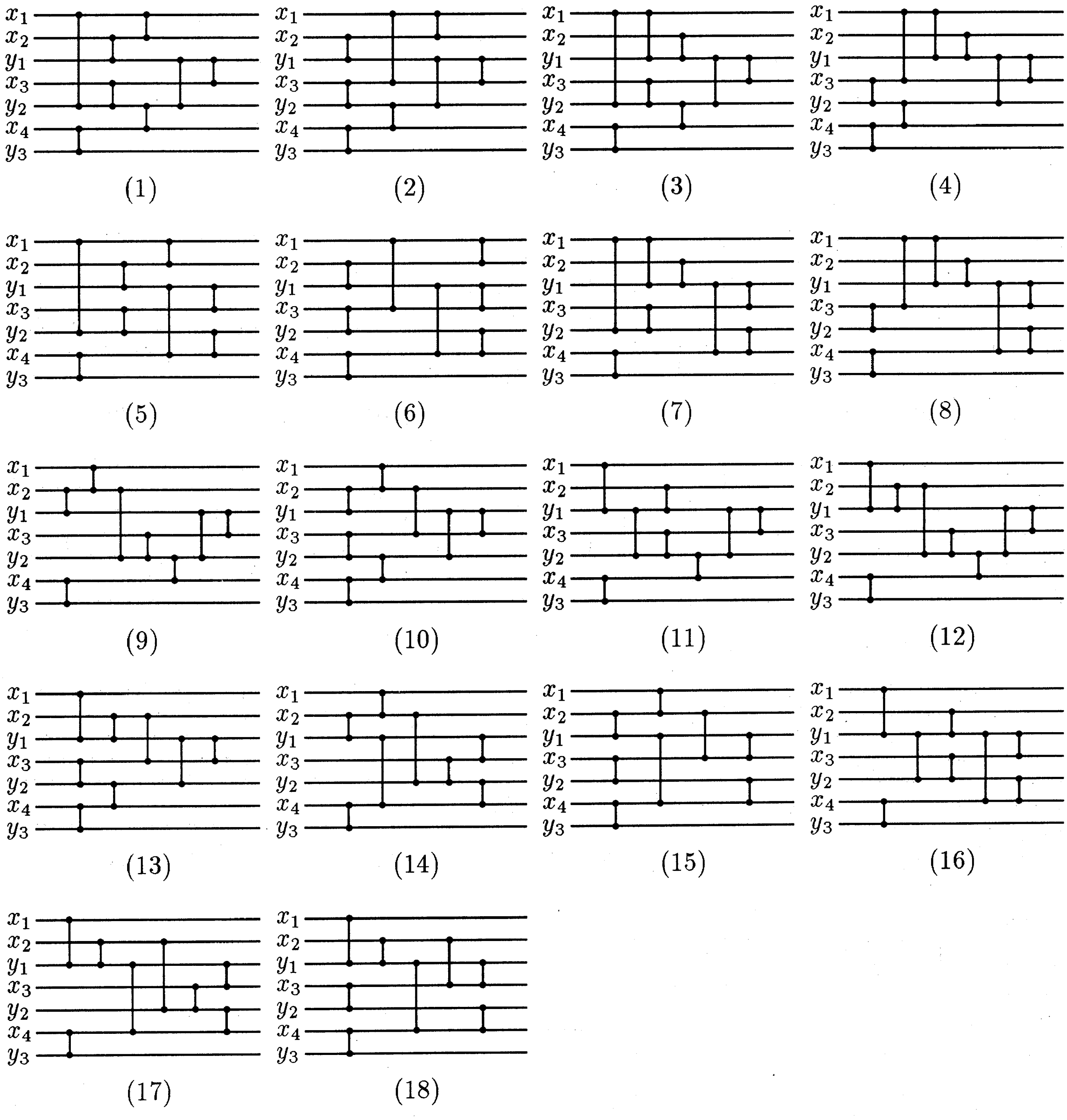


図 6: 入力行の並びを変えた (3, 4)–最適マージングネットワーク (その 1)

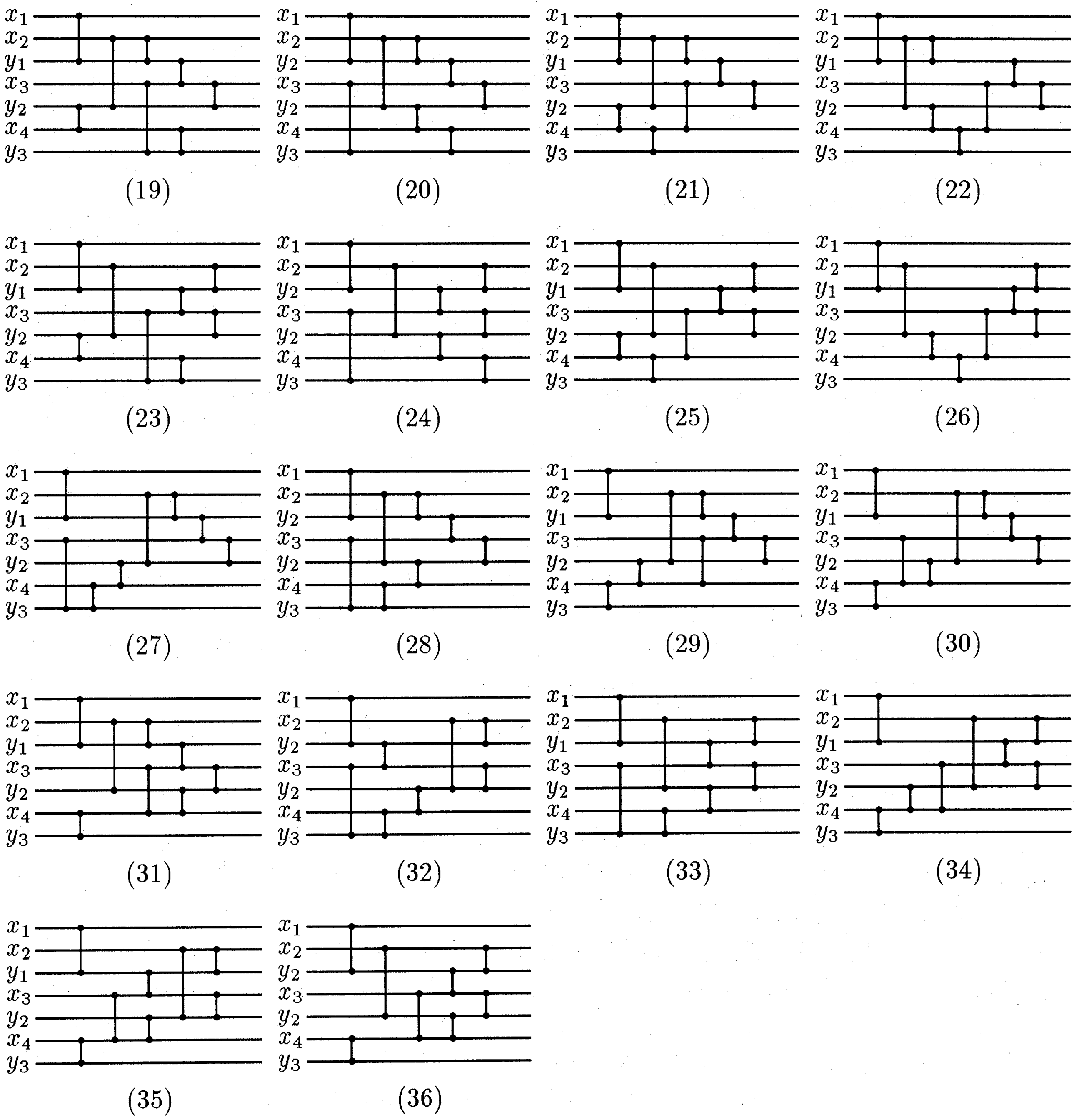


図 7: 入力行の並びを変えた (3,4)–最適マージングネットワーク (その 2)

2 $M(4,7)$–マージングネットワーク

命題 1 $M(4,7) = 16$

（証明の概略）Batcher [1] より $M(4,7) \leq 16$ なので $M(4,7) \geq 16$ を示せば十分である。増田、岩田 [4] より $M(4,6) = 14$ なので、補題 4より $M(4,7) \geq 15$ である。そこで以下では $M(4,7) = 15$ と仮定する。15 比較器で構成可能なすべてのネットワークを生成し、生成したネットワークが $(4,7)$–マージングネットワークでなければ $M(4,7) \neq 15$ すなわち $M(4,7) = 16$ であることが示せる。

15 個の比較器からなる任意のマージングネットワークを Nとする。Nの各行の並びを上から $x_1, x_2, y_1, x_3, y_2, x_4, y_3, x_5, y_4, x_6, x_7$ とする。N の下 10 入力行からなるサブネットワーク N_1 は $(4,6)$–マージングネットワークである。$M(4,6) = 14$ なので N_1 には 14 個以上の比較器がある。x_1 を上端とする比較器が 1 個存在し、かつ N 全体の比較器数は 15 なので N_1 は 14 比較器からなる $(4,6)$–最適マージングネットワークである。よって補題 3(1) より比較器 $[x_2 : y_1]$ がただ 1 個存在する。また N の上 3 入力行からなるサブネットワーク N_2 は $(1,2)$–マージングネットワークである。以上から N_2 中にはちょうど 2 個の比較器が存在し、N_2 は $(1,2)$–最適マージングネットワークである。補題 5より N_2 中には 2 比較器 $[x_2 : y_1], [x_1 : z], (z = x_2$または$z = y_1)$ が存在し、$[x_2 : y_1] \prec [x_1 : x_2]$ かまたは $[x_1 : y_1] \prec [x_2 : y_1]$ が成り立つ。$[x_2 : y_1] \prec [x_1 : x_2]$ の場合、入力要素 x_1 が N 中で入力線 y_1 よりもさらに下に移動するとき $[x_1 : x_2] \prec [x_2 :\downarrow y_1]$ となる比較器 $[x_2 :\downarrow y_1]$ が N_1 中に存在する。しかし N_1 は $(4,6)$–最適マージングネットワークなので補題 3(2) と矛盾する。よって N_2 中には $[x_1 : y_1]$,$[x_2 : y_1]$ の比較器が存在し、$[x_1 : y_1] \prec [x_2 : y_1]$ が成立する。同様にして、Nの下 3 入力行 y_4, x_6, x_7 からなるサブネットワークは比較器 $[y_4 : x_7]$,$[y_4 : x_6]$ からなり $[y_4 : x_7] \prec [y_4 : x_6]$ を満たす。

N 中に 8 個の隣接比較器 $[x_2 : y_1], [y_1 : x_3], [x_3 : y_2], \cdots, [y_4 : x_6]$ が存在するので、$[x_1 : y_1], [y_4 : x_7]$ をあわせた 10 個の比較器の上端と下端の位置が確定した。この 10 個の比較器の集合を S_1 とする。N は 15 比較器からなるので S_1 以外の 5 個の比較器を $\alpha, \beta, \gamma, \delta, \epsilon$ とし、この 5 個の比較器の集合を S_2 とする。

次に S_2 の比較器の上端と下端の位置のとりうる組み合わせを求める。x_1 を上端とする比較器はただ 1 個 $[x_1 : y_1]$ が存在することはすでに述べた。同様に入力行 x_7 を下端とする比較器もただ 1 個 $[y_4 : x_7]$ であることが示せる。よって $\alpha, \beta, \gamma, \delta, \epsilon$ の比較器はいずれも $[\downarrow x_2 :\uparrow x_6]$ の形をしている。N の上 5 入力行からなるサブネットワークは $(2,3)$–マージングネットワークである。$M(2,3) = 5$ なので、S_2 の比較器のうち少なくとも 1 個はこのサブネットワークに存在する。そこで α を $[\downarrow x_2 :\uparrow y_2]$ とする。N の上 7 入力行からなるサブネットワークは $(3,4)$–マージングネットワークである。$M(3,4) = 8$ なので、S_2 の比較器のうち少なくとも 2 個の比較器がこのサブネットワークに存在する。これらの比較器のうち α 以外の比較器を β とする。β は $[\downarrow x_2 :\uparrow y_3]$ の形である。N の上 8 入力行からなるサブネットワークは $(3,5)$–マージングネットワークである。$M(3,5) = 10$ なので、S_2 の比較器のうち少なくとも 3 個の比較器がこのサブネットワークに存在する。α, β 以外の比較器 γ の形を $[\downarrow x_2 :\uparrow x_5]$ とする。N の上 9 入力行からなるサブネットワークは $(4,5)$–マー

ジングネットワークである。$M(4,5)=12$ なので、S_2 の比較器のうち少なくとも 4 個の比較器がこのサブネットワークに存在する。α,β,γ 以外の比較器 δ の形を $[\downarrow x_2 :\uparrow y_4]$ とする。また ϵ を $[\downarrow x_2 :\uparrow x_6]$ とする。

同様にして N の下 5 入力行、下 7 入力行、下 8 入力行、下 9 入力行からなるサブネットワークを考えるとそれぞれ $(2,3)-$、$(3,4)-$、$(3,5)-$、$(4,5)-$マージングネットワークになっているので各サブネットワーク中に少なくとも S_2 の比較器はそれぞれ 1,2,3,4 個存在する。

補題 9 N 中の 2 入力線 y_2,x_4 間と 2 入力線 x_4,y_3 間の少なくともいずれか一方は S_2 の 2 比較器が通過する。

(証明) 2 入力線 y_2,x_4 間と 2 入力線 x_4,y_3 間がともに S_2 の 1 個の比較器が通過すると仮定する。y_2,x_4 間、x_4,y_3 間は 2 個の入力要素が上下どちらからも移動する場合がある。このとき入力要素の移動において 1 個は S_2の比較器を使いもう 1 個は S_1の隣接比較器 $[y_2 : x_4],[x_4 : y_3]$ を使う。しかし S_1の比較器では上下どちらか一方向にしか入力要素を移動することができない。よって y_2,x_4 間、x_4,y_3 間のいずれか一方には S_2 の比較器が少なくとも 2 個は存在しなければならない。□

隣接する入力行間を移動する入力要素の個数を考える。入力行 y_1,x_3 間では最大 2 個の入力要素が移動する場合がある。よって y_1,x_3 間には少なくとも 2 個の比較器が通過している。同様に考えると隣接する入力行の間にはそれぞれ上から順に 1,2,2,3,2,2,3,2,2,1 個の比較器が通過している。S_2 の比較器について考えるなら入力行 x_3,y_2 間、y_3,x_5 間にそれぞれ少なくとも 2 個づつ、y_1,x_3 間、y_2,x_4 間、x_4,y_3 間、x_5,y_4 間にそれぞれ少なくとも 1 個づつの比較器が存在する。また補題 9より y_2,x_4 間、x_4,y_3 間のいずれか一方には 2 比較器が存在する。

同一の組み合わせを避けるため、$c\in\{\beta,\gamma,\delta,\epsilon\}$ に対し c が $[\downarrow x_2 :\uparrow y_2]$ の形なら $\alpha \leq c$、$d\in\{\gamma,\delta,\epsilon\}$ に対し d が $[\downarrow x_2 :\uparrow y_3]$ の形なら $\beta\leq d$、また $e\in\{\delta,\epsilon\}$ に対し e が $[\downarrow x_2 :\uparrow x_5]$ の形なら $\gamma\leq e$ とし、ϵ が $[\downarrow x_2 :\uparrow y_4]$ の形なら $\delta\leq\epsilon$ として $\alpha,\beta,\gamma,\delta,\epsilon$ の上端下端の位置を考える。

さらに次の (1) $\sim$ (4) が成り立つ。

(1) N 中の上 5 入力行からなるサブネットワーク中に S_2 の比較器が k $(1\leq k\leq 5)$ 個存在するならその k 個のすべての比較器が同時に $[x_3 : y_2]$ の形をしている、ということはない。同様に N 中の下 5 入力行からなるサブネットワーク中に k $(1\leq k\leq 5)$ 個の S_2 の比較器が存在するなら k個の比較器すべてが $[y_3 : x_5]$ ということはない。

(2) N 中の上 6 入力行からなるサブネットワークを考える。このサブネットワーク中に 1 個の S_2 の比較器が存在するのならこのサブネットワークは $(2,4)-$最適マージングネットワークになっているので補題 6よりその比較器は $[y_1 : x_3]$ ではない。同様に N 中の下 6 入力行からなるサブネットワークを考える。このサブネットワーク中に 1 個の S_2 の比較器が存在するのならその比較器は $[x_5 : y_4]$ ではない。

(3) N 中の上 8 入力行からなるサブネットワークを考える。このサブネットワーク中に 3 個の S_2 の比較器が存在するならこのサブネットワークは $(3,5)-$最適マージングネット

ワークになっているので γ は $[y_3 : x_5]$ ではない。同様に N 中の下 8 入力行からなるサブネットワークを考える。このサブネットワーク中に 3 個の S_2 の比較器が存在するならこのサブネットワークは $(3,5)-$最適マージングネットワークになっているのでそれらの比較器はいずれも $[x_3 : y_2]$ ではない。

(4) N 中の上 9 入力行からなるサブネットワークを考える。このサブネットワーク中に 4 個の S_2 の比較器が存在するならこのサブネットワークは $(4,5)-$最適マージングネットワークになっているので δ は $[x_5 : y_4]$ ではない。同様に N 中の下 9 入力行からなるサブネットワークを考える。このサブネットワーク中に 4 個の S_2 の比較器が存在するならこのサブネットワークは $(4,5)-$最適マージングネットワークになっているのでそれらの比較器はいずれも $[y_1 : x_3]$ ではない。

S_2 中の比較器として $[x_2 : x_6]$,$[x_2 : y_1]$,$[y_4 : x_6]$ はいずれも冗長なので、S_2 の比較器はこれら 3 種類のいずれの比較器も含まない。

N 中の上 7 入力行からなるサブネットワーク N_3 を考える。N_3 中に S_2 の比較器が $[x_4 : y_3]$ を数えずに 2 個しかなければ、N_3 は S_2 中の $[x_4 : y_3]$ を除いて $(3,4)-$最適マージングネットワークであり、補題8より N_3 は図6,7のいずれかの形をしている。このとき比較器 $[x_1 : y_1]$,$[x_2 : y_1]$ が存在し $[x_1 : y_1] \prec [x_2 : y_1]$ を満たすこと、補題3(2) より比較器 $[x_2 :\downarrow y_1]$ が存在するなら $[x_2 :\downarrow y_1] \prec [x_2 : y_1]$ を満たすこと、上端が入力行 x_1 の比較器はただ一つしか存在しないことを考えると S_2 の 2 個の比較器は次の 4 通りのいずれかになる。

(1) $[y_1 : y_2], [y_1 : y_2]$, (2) $[y_1 : y_2], [y_1 : x_4]$, (3) $[x_2 : y_2], [x_3 : y_3]$, (4) $[x_2 : y_2], [x_3 : x_4]$

同様に N 中の下 7 入力行からなるサブネットワークを考えると、このサブネットワーク中に S_2 の比較器が $[y_2, x_4]$ を数えずに 2 個しかなければその 2 個の比較器は次の 4 通りのいずれかである。

(1) $[y_3 : y_4], [y_3 : y_4]$, (2) $[x_4 : y_4], [y_3 : y_4]$, (3) $[y_2 : x_5], [y_3 : x_6]$, (4) $[x_4 : x_5], [y_3 : x_6]$

N の入力線の並びは入力行 x_4 を中心に上下対称に配置されている。S_2の組合せで例えれば$\alpha = [x_2 : x_3], \beta = [y_1 : y_2], \gamma = [x_3 : x_4], \delta = [y_2 : x_5], \epsilon = [y_3 : x_6]$ と $\alpha = [x_2 : y_2], \beta = [x_3 : y_3], \gamma = [x_4 : x_5], \delta = [y_3 : y_4], \epsilon = [x_5 : x_6]$ は入力行 x_4に対して上下対称であり一方のみを考慮する。

これらの条件をもとに考えられる S_2 の組み合わせはプログラムを作り実行したところ 311 通りになった。

次に S_1 の 10 比較器だけで構成可能なネットワークについて考えよう。入力行 y_1 を考えると入力行 x_1, x_2, x_3 と比較器でつながっている。すでにみたように 2 個の比較器は $[x_1 : y_1] \prec [x_2 : y_1]$ と定まっているで、比較器 $[y_1, x_3]$ の位置は $[y_1 : x_3] \prec [x_1 : y_1]$、$[x_1 : y_1] \prec [y_1 : x_3] \prec [x_2 : y_1]$、$[x_2 : y_1] \prec [y_1 : x_3]$ の 3 通りが考えられる。入力行 y_4 については 3 通りの場合が考えられ、また入力行 x_3 から x_5 迄の各入力行間にはそれぞれ 2 比較器が接続しているのでそれらの比較器間のとりうる位置関係は 2 通りの場合がある。これらをすべて組み合わせると S_1 の 10 比較器のみによって作られるネットワークは $3 \times 2 \times 2 \times 2 \times 2 \times 2 \times 3 = 288$ 通りになる。

これを S_2 の比較器の組合せ 311 通りと組み合わせて構成できるすべてのネットワーク

を生成した。

このとき、$[x_2 : y_1] \prec [x_2 :\downarrow y_1]$ のような比較器 $[x_2 :\downarrow y_1]$ を含むネットワークは、補題3(2)より N_1 が $(4,6)-$最適マージングネットワークであることと矛盾するので考慮しない。また、$[y_4 : x_6] \prec [\uparrow y_4 : x_6]$ のような比較器 $[\uparrow y_4 : x_6]$ を持つネットワークも考慮しない。

そして構成したネットワークがマージングネットワークか調べた。構成可能なネットワークは3,086,249,058通りであった。この全てがマージングネットワークにならなかったことを確認した。プログラムはC言語でかき、JU1/170 (Ultra SPARC, 167MHz, SPECint92:252) を2台とJS5/110 (microSPARC-II, 110MHz, SPECint92:78.6) を4台同時に使い実行した。プログラムは422行で6台の平均実行時間は4時間27分であった。

プログラム検証のため本稿の2著者がそれぞれ独立にプログラムを書き、S_2 の5比較器の組合せの311通りの各々の場合について構成したネットワークの数を数え上げ、その数が一致することを確認し、2種類のプログラムにより同じ結果が得られたことを確認した。

以上により $M(4,7) = 16$ が示された。□

3 $M(5,6)-$マージングネットワーク

命題 2 $M(5,6) = 16$

（証明の概要）Batcher [1] より $M(5,6) \leq 16$ なので $M(5,6) \geq 16$ を示せば十分である。増田、岩田 [4] より $M(4,6) = 14$ なので、補題4より $M(5,6) \geq 15$ である。そこで、以下では $M(5,6) = 15$ と仮定する。15比較器で構成可能なすべてのネットワークを生成し、生成したネットワークが $(5,6)-$マージングネットワークでなければ $M(5,6) \neq 15$ すなわち $M(5,6) = 16$ であることが示せる。

15個の比較器からなる任意のマージングネットワークを N とする。N の各行の並びを上から $x_1, y_1, x_2, y_2, x_3, y_3, x_4, y_4, x_5, y_5, x_6$ とする。N 中に10個の隣接比較器 $[x_1 : y_1], [y_1 : x_2], [x_2 : y_2], \cdots, [y_5 : x_6]$ が存在するので10個の比較器の上端と下端の位置が確定する。この10個の比較器の集合を S_1 とする。また N は15比較器からなるので S_1 以外の5個の比較器を $\alpha, \beta, \gamma, \delta, \epsilon$ とし、この5個の比較器の集合を S_2 とする。

次に S_2 の比較器の上端と下端の位置のとりうる組み合わせを求める。N の上5入力行からなるサブネットワークは $(2,3)-$マージングネットワークである。$M(2,3) = 5$ なので、S_2 の比較器のうち少なくとも1個はこのサブネットワークに存在する。そこで α を $[\downarrow x_1 :\uparrow x_3]$ とする。N の上7入力行からなるサブネットワークは $(3,4)-$マージングネットワークである。$M(3,4) = 8$ なので、S_2 の比較器のうち少なくとも2個がこのサブネットワークに存在する。これらの比較器のうち α 以外の比較器を β とする。β は $[\downarrow x_1 :\uparrow x_4]$ の形である。Nの上9入力行からなるサブネットワークは $(4,5)-$マージングネットワークである。$M(4,5) = 12$ なので、S_2 の比較器のうち少なくとも4個がこのサブネットワークに存在する。α と β 以外の比較器を γ と δ としその形を $[\downarrow x_1 :\uparrow x_5]$ とする。また ϵ を $[\downarrow x_1 :\uparrow x_6]$ とする。

同様にして N の下5入力行、下7入力行、下9入力行からなるサブネットワークを考えるとそれぞれ $(2,3)-$, $(3,4)-$, $(4,5)-$マージングネットワークになっているので各サブ

ネットワーク中に少なくとも S_2 の比較器がそれぞれ 1,2,4 個存在する。

隣接する入力行間を移動する入力要素の個数を考える。入力行 x_2, y_2 間では最大 2 個の入力要素が移動する場合がある。よって x_2, y_2 間には少なくとも 2 個の比較器が通過している。同様に考えると隣接する入力行の間にはそれぞれ上から順に 1,1,2,2,3,3,2,2,1,1 個の比較器が通過している。S_2 の比較器について考えるなら入力行 x_3, y_3 間、y_3, x_4 間にそれぞれ少なくとも 2 個づつ、x_2, y_2 間、y_2, x_3 間、x_4, y_4 間、y_4, x_5 間にそれぞれ少なくとも 1 個づつの比較器が存在する。

同一の組み合わせを避けるため、$c \in \{\beta, \gamma, \delta, \epsilon\}$ に対し c が $[\downarrow x_1 :\uparrow x_3]$ の形なら $\alpha \leq c$、$d \in \{\gamma, \delta, \epsilon\}$ に対し d が $[\downarrow x_1 :\uparrow x_4]$ の形なら $\beta \leq d$、$e \in \{\delta, \epsilon\}$ に対し e が $[\downarrow x_1 :\uparrow x_5]$ の形なら $\gamma \leq e$、ϵ が $[\downarrow x_1 :\uparrow x_5]$ の形なら $\delta \leq \epsilon$ として $\alpha, \beta, \gamma, \delta, \epsilon$ の上端下端の位置を考える。

N 中の任意の隣接する 5 入力行からなるサブネットワーク N_1 を考える。このとき入力行を上から z_1, z_2, z_3, z_4, z_5 とする。N_1 中に S_2 の比較器が 1 個しか存在しなければ N_1 は最適 $(2,3)$–マージングネットワークである。このとき S_2 の比較器は $[z_1 : z_2], [z_1 : z_5], [z_4 : z_5]$ でない。次に N 中の任意の隣接する 9 入力行からなるサブネットワーク N_2 を考える。このとき入力行を上から $z_1, z_2, z_3, z_4, z_5, z_6, z_7, z_8, z_9$ とする。N_2 中に S_2 の比較器が 1 個しか存在しなければ N_2 は最適 $(4,5)$–マージングネットワークである。このとき S_2 の比較器は $[z_1 : z_2], [z_1 : z_9], [z_8 : z_9]$ でない。また N 中の比較器として $[x_1 : y_1]$,$[x_1 : x_6]$,$[y_5 : x_6]$ はいずれも冗長なので、S_2 の比較器はこれら 3 種類のいずれの比較器も含まない。

N 中の任意の隣接する 7 入力行からなるサブネットワークに S_2 の比較器が 2 個しかなければそのサブネットワークは $(3,4)$–最適マージングネットワークであり、補題 8より図 4,5のいずれかの形をしている。この場合 S_2 の 2 比較器の組合わせは 8 通りである。また N 中の任意の隣接する 6 入力行からなるサブネットワークに S_2 の比較器が 1 個しかなければそのサブネットワークは $(3,3)$–最適マージングネットワークであり、補題 7より図 3 のいずれかの形をしている。このとき S_2 の比較器の形は 4 通りである。

N の入力線の並びは入力行 y_3 を中心に上下対称に配置されている。S_2の組合せで例えれば$\alpha = [x_1 : y_2], \beta = [x_2 : y_3], \gamma = [x_3 : x_4], \delta = [x_3 : y_4], \epsilon = [x_4 : y_5]$ と $\alpha = [y_1 : x_3], \beta = [y_2 : x_4], \gamma = [x_3 : x_4], \delta = [y_3 : x_5], \epsilon = [y_4 : x_6]$ は入力行 y_3に対して上下対称であり一方のみを考慮する。

これらの条件をもとに考えられる S_2 の組み合わせはプログラムを作り実行したところ 1061 通りになった。与えられた 5 個の比較器の情報をもとに既に決まっている 10 個の比較器と組み合わせてネットワークを構成する。15 比較器からなるすべてのネットワークを構成するのに必要な計算時間は $M(4,7)$ の計算よりも数倍時間がかかることが予想された。そこでサブネットワークに注目し全体を 3 通りの場合に分けて計算に改良を加えた。

(1) 隣接する 7 入力行からなるサブネットワークを考える。サブネットワーク中に S_2 の比較器が 2 個しかない場合、このサブネットワークは $(3,4)$–最適マージングネットワークである。このサブネットワークは補題 8の 36 通りのネットワークのいずれかの形をしている。実際にはサブネットワーク中の S_2 の 2 個の比較器の形から調べると 4 〜 5 通りが該当する。これを元に残りの比較器を付け加えていくことによりネットワーク全体を構成する。

(2) 隣接する 6 入力行からなるサブネットワークを考える。サブネットワーク中に S_2 の比較器が 1 個しかない場合このサブネットワークは $(3,3)$–最適マージングネットワークで

ある。このサブネットワークは補題7の4通りのネットワークのいずれかの形をしている。サブネットワーク中にある S_2 の比較器の形からただ1通りの形を特定できる。これを元に残りの比較器を付け加えていくことによりネットワーク全体を構成する。

(3) (1)、(2) のどちらの場合にも当てはまらないとき。$M(4,7)$ の計算のように S_1 の10個の隣接比較器からなるネットワークを作る。このネットワークは $2^9 = 512$ 通りある。これを S_2 の比較器の組合せと組み合わせて構成できるすべてのネットワークを生成する。

1061通りの S_2 の組合せのうち (1) の場合が536通り、(2) の場合が466通り、(3) の場合が59通りであった。

以上の3通りの方法を用いネットワークを構成した。このとき、$[x_1 : y_1] \prec [x_1 :\downarrow y_1]$ のような比較器 $[x_1 :\downarrow y_1]$ を含むネットワークは補題3(2)より $(5,6)$–最適マージングネットワークでない。また、$[y_5 : x_6] \prec [\uparrow y_5 : x_6]$ のような比較器 $[\uparrow y_5 : x_6]$ を持つネットワークも同様に $(5,6)$–最適マージングネットワークでない。この様な $(5,6)$–マージングネットワークにならない比較器の組み合わせは取り除いた。そして構成したネットワークが $(5,6)$–マージングネットワークかを調べた。構成可能なネットワークは926,032,127通りであった。この全てがマージングネットワークでないことを確認した。プログラムはC言語でかき、JU1/170を2台とJS5/110を4台同時に使い実行した。プログラムは1275行で6台の平均実行時間は1時間34分であった。

プログラム検証のため本稿の2著者がそれぞれ独立に上と異なるプログラムを作り、すべての15比較器からなるネットワークが $(5,6)$–マージングネットワークでないことを確認した。計算方法は $M(4,7)$ の計算に用いたものと同じでありJU1/170 1台で実行したときの計算時間はそれぞれ4日、14日であった。

以上により $M(5,6) = 16$ が示された。□

参考文献

[1] K.E.Batcher, Sorting networks and their applications, *Proc. AFIPS* 1968 *SJCC* 32 AFIPS Press (1968) 307–314.

[2] A.C.Yao and F.F.Yao, Lower bounds on merging networks, *J. Assoc. Comput. Mach.* **23** (1976) 566–571.

[3] M.Aigner and O.Schwarzkopf, Bounds on the size of merging networks, *Discrete Applied Math.* **61** (1995) 187–194.

[4] 増田一寿, 岩田茂樹, マージングネットワークの下界について, 信学論 D-I **J80-D-I**, (1997) 665–673.

[5] I.Parberry, A computer-assisted optimal depth lower bound for nine-input sorting networks, *Math. Systems Theory* **24** (1991) 101–116.

[6] D.E.Knuth, *The Art of Computer Programming Vol.3: Sorting and Searching*, Addison-Wesley (1973).

[7] K.Yamazaki, H.Mizuno, K.Masuda, S.Iwata, Minimum number of comparatur in (6,6)-merging network, 投稿中