Vol. 22 No. 3　　情報処理学会論文誌　　May 1981

ショートノート

双端キューの並列結合によるソーティングについて†

野　崎　昭　弘††

双端キューの並列ネットワークによる順列のソーティングを考える．ソートすべき順列のひとつの要素の処理方法の決定にあたって，それよりあとの未処理要素は参照できないとすると，長さ n の任意の順列がソートできるためには，$\lceil (n-1)/3 \rceil$ 個の双端キューが必要かつ十分である．

1. まえがき

スタック，キューあるいは**双端キュー**（両端で入出力が可能なリスト，double-ended queue, *Deque*）の直列または並列結合によるソーティングについては，Knuth[1] や今宮[2] らの研究がある．そして長さ n の任意の順列がソートできるためには

　並列結合では　$O(\sqrt{n})$ 個，

　直列結合では　$O(\log n)$ 個

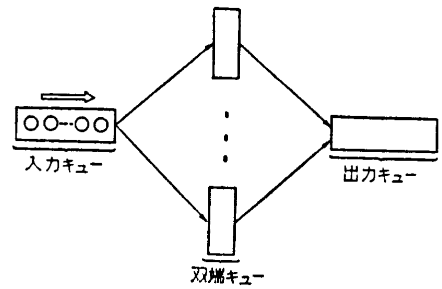
の双端キューが必要十分であることが知られている[3]．

ところでこれらの研究においては，ソーティングの手順をきめる以前に，ソートすべき順列が完全にわかっていると仮定していた．これに対して，渋谷政昭氏（日本 IBM）が次のような問題を提出された．

ソートすべき順列は最初入力キューに貯えられていて，この中のデータを双端キューのネットワークを通じて，小さい順にソートして出力キューに送り出すことを考える（**図1**）．従来は入力キューの中のデータは完全にわかっているものと仮定していたが，次の前提のもとでは，どうなるだろうか？

前提（＊）　データの移動方法を決定するために参照できるのは，入力キューの先頭の要素と，ネットワーク内に記憶されている要素とに限る．なお処理システムがフリップ・フロップやカウンタを使用することはさしつかえない．

直列結合によれば，この前提のもとでも，$O(\log n)$ 個のスタックで長さ n の任意の順列がソートできる．しかし並列結合では事情が一変して，$O(n)$ 個の双端キューが必要になる．我々は幸い，長さ n の任意の順列が（前提（＊）のもとで）ソートできるために必要十分な双端キューの個数を正確に決定することができたので，報告したい．



入力キュー　input queue
出力キュー　output queue
双端キュー　deque

図 1　双端キューの並列ネットワークによるソーティング
Fig. 1　Sorting by a parallel network of deques.

2. k 個の双端キューの能力

補題 1　k 個の双端キューの並列結合によって，前提（＊）のもとで，長さ $3k+1$ の任意の順列がソートできる．

［証明］ 長さ3の任意の順列はひとつの双端キューの中に，**図2**のような配置でおさめることができる（3角形はひとつの単調列をあらわす）．そこで $3k$ 個の要素を k 個の双端キューに3個ずつ，図2のような形に配分した上で，最後の1個と多重併合（multi-way merging）を行えば，全体のソーティングができる．

［証終］

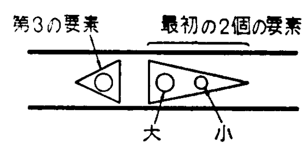


図 2　3個の要素の配置
Fig. 2　Disposition of three elements in a deque.

† Sorting by a Parallel Network of Deques by AKIHIRO NOZAKI (Natural Science Department, College of Liberal Arts, International Christian University).

†† 国際基督教大学教養学部理学科

3. k 個の双端キューの能力の限界

はじめにソーティングの手順がかなり制限されていることを注意しておこう．ソートすべき順列を $c_1, c_2, \cdots, c_n$ であらわし，これらを双端キューの並列ネットワークでソートすることを考える．

事実 1 $c_1, \cdots, c_{n-1}$ までは全部ネットワークの中に記憶しなければならない．

これは c_n が最小要素かもしれないからである．

事実 2 双端キュー間のデータ移動はできない．

これは並列ネットワークだから当然である．結局 $c_t (1 \leqq t < n)$ に対しては，「どの双端キューのどちら側から入れるか」という選択しかできない．そして最後の c_n と，多重併合を行うのである．

事実 3 双端キューの中のデータは，ひとつの単調列か，または小さい側が両端を向いているふたつの単調列かのどちらかである（**図 3**）．

たとえばある双端キューの中に 4, 2, 5 のような配列が生じると，中間の 2 をさきに取りだすことができないので，ソーティングは失敗する．

補題 2 前提（＊）のもとで，k 個の双端キューの並列結合によっては，長さ $3k+2$ の順列を全部ソートすることができない．

［証明］ いわゆる“競技法”(adversary approach) による[4]．すなわち，順列の各要素を指定する競技者 A と，指定された要素を「どの双端キューのどちら側に入れるか」を決定する競技者 B とが交互に着手する．我々は，競技者 B がどんな手を打っても，結局ソートできない場合があることを証明する．以下各要素 c_t は実数とし，簡単のためどれも負でないとする．

まず準備として，各双端キュー $D_i (1 \leqq i \leqq k)$ に対して，次のようなパラメータを定義しておく．

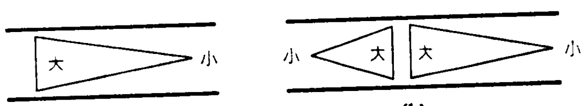
1） D_i が空のとき；$f_i=3,\ p_i=0,\ q_i=+\infty$.

2） D_i がただひとつの要素 a を含むとき；

$$f_i=2, \quad p_i=a, \quad q_i=+\infty.$$

3） D_i が 2 個以上要素を含むひとつの単調列（最小値を a，最大値を b とする）を含むとき；

$$f_i=1, \quad p_i=1, \quad q_i=b.$$



(a)　(b)

図 3 双端キュー内の単調列

Fig. 3 Monotone sequences in deques.

4） それ以外の場合；$f_i=0$，$p_i=$ 両端の要素の大きい方，$q_i=+\infty$

ソーティングが可能であるためには，4）の場合は図 3（b）の配置でなければならない．また

$$N=f_1+\cdots+f_k,$$
$$P=\mathrm{Max}\{p_1, \cdots, p_k\},$$
$$Q=\mathrm{Min}\{q_1, \cdots, q_k\}$$

とおく．最初はすべての双端キューが空なので

$$N=3k, \quad P=0, \quad Q=+\infty$$

である．これらの値は競技者 B の着手によって変化するが，競技者 A が次のようにすれば，B の着手ごとに N の値が 1 ずつ減ること，しかも不等式 $P<Q$ が保存されることが確かめられる．

競技者 A の着手：$1 \leqq t < n (=3k+2)$ のときは，$P < c_t < Q$ となるように c_t を選ぶ．

この c_t を，競技者 B が双端キュー D_j に入れたとしよう．

1） $f_j=3$ だったとき；f_j は 2 に変わるので，N の値は 1 だけ減る（他の f_i は変わらない）．また p_j は c_t になり，q_j に変わらないので，

$$(\text{新})P=c_t<(\text{旧})Q=(\text{新})Q.$$

2） $f_j=2$ だったとき；$p_j \leqq P < c_t$ であるから，p_j は変わらず，q_j は c_t になる．f_j は 2 から 1 に変わるので，N の値は 1 だけ減る．また

$$(\text{新})P=(\text{旧})P<c_t=(\text{新})Q.$$

3） $f_j=1$ の場合；D_j の中の単調列の最小値を a，最大値を b とすると，$a \leqq P < c_t$ であるから，c_t を a の側から入れることはできない．そこで c_t を b の側から入れると $a \leqq P < Q \leqq b > c_t$ なので，D_j 内はひとつの単調列ではなくなる．ゆえに f_j の値は 0 になり，$p_j=c_t$，$q_j=+\infty$ となる．

$$(\text{新})P=c_t<(\text{旧})Q \leqq (\text{新})Q.$$

4） $f_j=0$ の場合；この場合は D_j の両端 $\leqq P < c_t$ であるから，考えなくてよい（ここに c_t を入れるとソーティングは失敗する）．

結局いつも $P<Q$ が保存されるので，競技者 A は同じ方針で次の要素を指定し続けることができる．そして B の着手ごとに N の値が 1 ずつ減るので，c_{3k} をネットワークに取りこんだ時点で $N=0$ となる（すべての i について $f_i=0$）．そこで

$$c_{n-1}>P, \qquad c_n=0$$

とすると，競技者 B は c_{n-1} をどこにも入れられず，結局ソーティングは失敗する．［証終］

補題 1, 2 から次の定理が得られる．

定理 前提（∗）のもとで，長さnの任意の順列が双端キューの並列結合でソートできるためには，$\lceil (n-1)/3 \rceil$ 個の双端キューが必要十分である．

4. むすび

前提（∗）はもっと強く，次の形にしてもよい．

前提（∗∗） データの移動方法を決定するために参照できるのは，取出し可能な要素に限る．

また，双端キューに制限をつけて，たとえば入力が一方の端からしかできない**入力制限双端キュー** IRDQ におきかえたときには，$\lceil (n-1)/2 \rceil$ 個が必要十分になる（同じような手法で証明できる）．いずれにしても $O(\sqrt{n})$ では間にあわず，$O(n)$ だけ必要になるのは注意すべきことと思われる．

参考文献

1) Knuth, D. E.: Fundamental Algorithms, Addison-Wesley (1968).

2) 今宮淳美，野崎昭弘：制限つき Deques による順列のソーティング，情報処理，Vol. 17, No. 12, pp. 1128-1134 (1976).

3) Nozaki, A.: Sorting using networks of deques, J. of Computer and System Sciences, Vol. 19, No. 3, pp. 309-315 (1979).

4) Blum, M. et al.: Time bounds for selection, ibid, Vol. 7, No. 4, pp. 448-461 (1972).

(昭和55年9月11日受付)

(昭和55年11月20日採録)