数理解析研究所講究録
第1148巻 2000年 219-224

伸長係数2のコンパクトラウティングアルゴリズム

河内 亮周 (Akinori Kawachi)　岩間 一雄 (Kazuo Iwama)
京都大学
{kawachi,iwama}@kuis.kyoto-u.ac.jp

内容梗概

[Cow99] において単純で実際的なモデルの下で伸長係数3、テーブルサイズ $O(n^{2/3}\log^{4/3}n)$ のコンパクトラウティングアルゴリズムが示された。本稿では同じモデルを用いて伸長係数が3未満の場合に必要なテーブルサイズがどの程度大きくなるかを調べ、その結果、(i) 伸長係数2で各節点のテーブルサイズの上限 $(n-\sqrt{n}+2)\log n$ のアルゴリズムの存在、(ii) 伸長係数3未満のどんなアルゴリズムに対しても $(n-2\sqrt{n})\log n$ のテーブルサイズが必要となる節点を持つグラフの存在、が示される。テーブルサイズ $n\log n$ で自明に最短距離を保証するラウティングが可能であるから、それに比べて $\sqrt{n}\log n$ 程度しか減らすことができない。

1　序論

コンパクトラウティングとは近似アルゴリズムの一種である。コンパクトラウティングにおいて近似アルゴリズムでの近似度にあたるものは伸長係数 (stretch factor) と呼ばれ、与えられたグラフに対してアルゴリズムによってパケットが通る距離を最短距離で割ったときの任意の2節点間に対する最大値を表す。すなわち最悪の場合でも最短距離の伸長係数倍以内で目標節点に到着できるということを意味し、ラウティングの効率を見るための一つの基準となる。本稿では [Cow99] で用いられたモデルにおいて伸長係数3という値が必要なテーブルサイズについて大きな影響を与えることが示される。すなわち [Cow99] のモデルの下で伸長係数が3を下回るような要求を出した場合、テーブルサイズが非常に大きくなってしまうことが証明される。一方で、伸長係数3を許せば非常に効率の良いアルゴリズムが知られている [Cow99]。また異なったモデルの下でも、[GG97] で伸長係数3を下回る場合にテーブルサイズが大きくなることが証明されている。

本稿で扱われるコンパクトラウティングでは、各節点がエントリ $[v,p]$ からなるテーブルを持っており、このエントリはパケットの目標節点が v ならばポート p へ送るという意味である。各節点のテーブルで全節点に対してエントリがあれば、最短経路でラウティングできるのは明らかであるが、テーブルの大きさが $n\log n$ となる。(1エントリの大きさは $\log n$ とする)。コンパクトラウティングの目的は、ヘッダを利用しつつ伸長係数をある程度大きくする代わりにラウティングテーブルの大きさを減らすことである。

Fig.1のようなネットワークを考える。ここで A,B,C はそれぞれ v_i,v_j,v_k 間の最短距離である。最短距離の原理から $C\leq A+B$ である。ここである $\sigma\geq 1$ に対し $A+B\leq\sigma C$ が成立すると仮定する。伸長係数 σ を許すならば、v_i から v_k へ送られるパケットは v_j を通って v_k へ向かうことができる。このラウティングは例えば (v_k,v_j) といった (主) 目標と副目標の節点対で表されるヘッダで実現することができる。副目標はテーブルに v_k へのエントリがないときに用いられる。

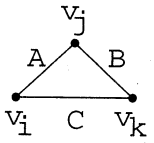

Fig.1

[Cow99] では伸長係数3でメモリ量 $O(n^{2/3}\log^{4/3}n)$ のアルゴリズムが示されている。その論文では伸長係数3を下回るとメモリ量を小さく押えることができないと推測されており、実際に本稿でその推測が正しいことが証明される。より詳しくは次の2つの結果を証明する。(i) [Cow99] のモデルの下で任意のグラフに対して、最短距離の2倍以内を保証するとき各節点のテーブルの大きさが $(n-\sqrt{n}+2)\log n$ で十分であるよう

なヘッダとテーブルの構成を与えるアルゴリズムが存在する。(ii) 伸長係数 3 未満を保証するときテーブルサイズが $(n-2\sqrt{n})\log n$ であるような節点を持つグラフが存在する。

本稿で扱う [Cow99] のモデルは次のような特徴を持つ。(i) ヘッダは目標節点に対して一意に決まり、目標節点に到着するまでの途中では変更されない。(ii) ヘッダの大きさは $O(\log n)$ である。(正確には $3\log n$) (iii) 初期節点でのヘッダの生成のためのメモリは必要メモリ量に入れない。特に (iii) について、このことよりヘッダは節点に付けられるラベルとみなすことができ、ラベルはラウティングプロセスではなくプリプロセスで計算されるという仮定が置かれる。このモデルは最初に [EGP98a, EGP98b] で導入され、伸長係数 5 で各節点のテーブルの大きさが次数 d の節点で $O(n^{1/2}\log^{3/2} n + d\log n)$、全体では $O(n^{3/2}\log^{3/2} n)$ という結果が得られている。

一方で多くの論文 [FG95, GG97, GP96, PU89] において異なったモデルの下で議論が行なわれている。主な違いとして (i) 目標節点に到着するまでの途中の節点でヘッダを書き換えることができる。(ii) ヘッダの初期化、書き換えのためのメモリも必要なメモリとして計算する。が挙げられる。(i) より明らかにモデルは強力なものとなっているが、2. より必要なメモリ量を小さくすることが難しくなっている。このことはこのモデルを使った論文においてラウティングアルゴリズムが非常に複雑であり、また必要なメモリ量の下限を示すことがより人気のある議論となっている理由である。その中で [GG97] では n 節点のネットワークにおいて伸長係数が 3 未満のどんなアルゴリズムに対しても必ず各節点で $\Omega(n)$ 必要であることを示しており、しかし伸長係数が 3 ならば [ABLP90] で $O(n^{3/2}\log n)$ の上限が示されている。

伸長係数が 3 か 3 未満で大きく異なってくる直観的理由は以下の様に説明できる。今 Fig.1 において $A \leq B$ であると仮定する。このとき $C \leq A+B$ より $A+C \leq A+A+B \leq 3B$。これが意味するのは v_j から v_k へ向かうパケットは v_i を経由しても伸長係数が 3 より小さいということである。また $B \leq A$ ならば同様に v_j から v_i へ向かうパケットは v_k を経由してもよい。このことは [Cow99] のアルゴリズムにおいて非常に重要な意味を持つ。しかし伸長係数が 3 を下回る場合、この性質をそのまま利用することはできない。

2 モデル

本稿では [Cow99] と同じモデルを扱うが、[Cow99] で「ラベル (label)」と呼ばれていたものを本稿では「ヘッダ (header)」として扱っている。(i) 各節点には 1 つのテーブルがあり、テーブルは [v,p] なる形式のエントリを含んでいる。本稿ではテーブルサイズの評価をエントリ数で行なう。(ii) 全てのパケットは目標節点に対し一意に決まるヘッダを保持し、$H(v) = (v, u, (x, p))$ の形で表現される。v はパケットの目標節点であり、u, x は節点、p はポートを表す。(iii) 各節点はパケットに対して次のような操作を行なう。まずパケットの目標節点がどうかをチェックする。今パケットがいる節点が目標節点でないのならば、目標節点 v に対してテーブルが $[v, p]$ なるエントリを含んでいればポート p へパケットを送る。もしそのようなエントリがなければ u を調べ、$[u, p']$ なるエントリを含んでいればポート p' へパケットを送る。この u を副目標と呼ぶ。それでも行き先が決まらない場合で現在いる節点が x ならばポート p_x へパケットを送る。

3 テーブルサイズの上限

定理 1 n 節点の任意のグラフ G に対して、伸長係数 2 を保証する場合、各節点のテーブルサイズの上限は $(n-\sqrt{n}+2)\log n$ である。

証明 最初に基本的な方針を述べた後、詳細を追うことにする。

3.1 基本方針

まず与えられた節点集合 V ($|V| = n$) を半分づつの節点数をもつ集合 U, W に分ける。つまり $V = U \cup V$, $|U| = |W| = n/2$, $U \cap W = \phi$ である。(以下の議論を簡単にするため n を偶数と仮定する。一般化は容易にできる)。この U, W に対して $U \times W$ で最も 2 点間の距離が小さい対 (u_1, w_1) ($u_1 \in U, w_1 \in W$) を取る。次に $\{U - \{u_1\}\} \times \{W - \{w_1\}\}$ で最も 2 点間の距離が小さい対 (u_2, w_2) を取る。以下同様に $\{U - \{u_1, ..., u_{i-1}\}\} \times \{W - \{w_1, ..., w_{i-1}\}$ から対 (u_i, w_i) を選ぶ。

理由は後で述べるが、3 つの節点 u_i, u_j, w_j ($j < i$) について (i)$A + C \leq 2B$ (ii)$B + C \leq 2A$ のいずれか一方が必ず成立する。(Fig.2) (i) のときは u_j を副目標とすることで w_j のエントリを省略し、(ii) のときは

w_j を副目標とすることで u_j のエントリを省略する。全ての $j < i$ についてこのことは言えるので番号 i の節点では $i-1$ エントリが省略できることになる。しかし、i が小さいと省略できるエントリ数も少なくなってしまう。例えば u_1, w_1 では全く省略できない。そこである値 m に対して、新たに節点集合を

$$X = \{u_i\}_{1 \le i \le m} \cup \{w_i\}_{1 \le i \le m}, \quad Y = V - X$$

で定義し、Y の節点では上記の性質を利用して X の節点へのエントリのうち半分を省略し、X の節点では自分より節点番号が小さい節点に対しては Y の場合と同様の省略を行うが、その他に Y の節点へのエントリに対してヘッダを利用した省略を行う。例えば u_m において上記の省略だけでは $m-1$ エントリだが、u_{m+1} のヘッダを $H(u_{m+1}) = (u_{m+1}, u_{m+1}, (u_m, p_{u_{m+1}}))$ とすることで u_m で u_{m+1} へのエントリが省略できることが分かる。ここで $p_{u_{m+1}}$ は u_m における u_{m+1} への最短経路へ向かうポートとする。m の値は後で最適なものを求める。

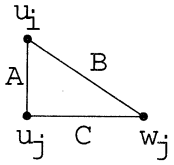

Fig.2

3.2 ヘッダの構成

X の節点に対して次のようにヘッダを構成する。

$$H(u_i) = (u_i, w_i, *), \quad H(w_i) = (w_i, u_i, *)$$

すなわち X の節点 u_i, w_i が目標、副目標の役割を交互に持つように構成する。ここで $*$ の部分は特に必要としない。このヘッダの構成は前小節で述べた副目標を経由させる省略に必要である。

また前小節で述べたように X から Y へ向かうエントリをヘッダにより省略することを考える。副目標を経由させる省略方法によって u_i, w_i では少なくとも $i-1$ 個のエントリが省略されるので i に応じて傾斜的にヘッダを割り当てることにする。例えば u_m, w_m のテーブルにおいて副目標を経由させる省略方法では $m-1$ エントリが省略されるので

$$H(u_{m+1}) = (u_{m+1}, u_{m+1}, (u_m, p_{u_{m+1}})), \quad H(w_{m+1}) = (w_{m+1}, w_{m+1}, (w_m, p_{w_{m+1}}))$$

とすることで u_m, w_m からそれぞれ u_{m+1}, w_{m+1} へのエントリ、すなわち 1 エントリを省略できる。ここで p_{u_i}, p_{w_i} は u_m, w_m における u_i, w_i への最短経路へ向かうポートを表す。同様に u_{m-1}, w_{m-1} では $m-2$ エントリが省略されるので

$$H(u_{m+2}) = (u_{m+2}, u_{m+2}, (u_{m-1}, p_{u_{m+2}})), \quad H(w_{m+2}) = (w_{m+2}, w_{m+2}, (u_{m-1}, p_{w_{m+2}})),$$

$$H(u_{m+3}) = (u_{m+3}, u_{m+3}, (w_{m-1}, p_{u_{m+3}})), \quad H(w_{m+3}) = (w_{m+3}, w_{m+3}, (w_{m-1}, p_{w_{m+3}}))$$

としてヘッダで 2 エントリ省略できるように構成する。一般的には u_k, w_k $(k \le m)$ に対して $k-1$ エントリが省略されるので、k が偶数の場合

$$\begin{aligned}
H(u_{\frac{1}{2}k(k+1)+1}) &= (u_{\frac{1}{2}k(k+1)+1}, u_{\frac{1}{2}k(k+1)+1}, (u_{m-k}, p_{u_{\frac{1}{2}k(k+1)+1}})) \\
H(w_{\frac{1}{2}k(k+1)+1}) &= (w_{\frac{1}{2}k(k+1)+1}, w_{\frac{1}{2}k(k+1)+1}, (u_{m-k}, p_{w_{\frac{1}{2}k(k+1)+1}})) \\
&\vdots \\
H(u_{\frac{1}{2}k(k+2)+1}) &= (u_{\frac{1}{2}k(k+2)+1}, u_{\frac{1}{2}k(k+2)+1}, (u_{m-k}, p_{u_{\frac{1}{2}k(k+2)+1}})) \\
H(w_{\frac{1}{2}k(k+2)+1}) &= (w_{\frac{1}{2}k(k+2)+1}, w_{\frac{1}{2}k(k+2)+1}, (u_{m-k}, p_{w_{\frac{1}{2}k(k+2)+1}}))
\end{aligned}$$

とすることで $m-k+1$ エントリ省略できる。Y の節点のヘッダで重要なのは 3 番目の要素であり、副目標は特に利用しない。

3.3 テーブルの構成

最初は全てのテーブルで n エントリがある状態、すなわち任意の2節点間を最短距離でパケットをやりとりできる状態とする。任意の節点 v のテーブルにおいて、X の節点である u_i, w_i $(1 \leq i \leq m)$ のどちらかを省略することを考える。v と u_i、v と w_i、u_i と w_i の最短経路をそれぞれ A, B, C とする。(Fig.3)
v が u_i と w_i の最短経路上にある場合はいずれも省略しない。そうでなければ、$A \leq B$ のとき、$A + C \leq 2B$ が成立するならば v のテーブルから w_i へのエントリを省略し、$A > B$ のとき、$B + C \leq 2A$ ならば u_i のエントリを省略する。直観的には u_i, w_i のうち近い方を経由させることにより、遠い方の節点へのエントリを省略するという方法である。

次にこの省略方法で u_i, w_i $(1 \leq i \leq \frac{n}{2})$ のテーブルでどれだけ省略されているか考える。$j < i$ なる u_j, w_j に対し、まず u_i について u_i と u_j、u_i と w_j、u_j と w_j の最短経路をそれぞれ A, B, C とする (Fig.4)。

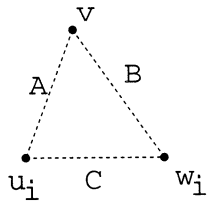

Fig.3

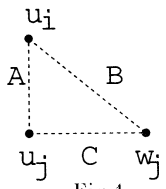

Fig.4

補題1 u_i, w_i $(i = 1, ..., \frac{n}{2})$ は u_j, w_j $(1 \leq j < i)$ の最短経路上には存在しない。
証明 もし u_i, w_i の中のある節点 v が u_j, w_j の間の最短経路上に存在すると仮定すると、u_j, w_j の取り方より $v \in U$ ならば u_j, w_j 間の最短距離より v, w_j 間の最短距離の方が小さいので v が u_j として選ばれなければならず、また $v \in W$ ならば同様に v が w_j として選ばれなければならないので矛盾。■

この補題より u_i, w_i は u_j, w_j 間の最短経路上には存在しない。$j < i$ なので節点対の取り方より $C \leq B$ が成立しなければならない。従って $A \leq B$ のとき $A+C \leq B+C \leq 2B$。よって u_i では w_j へのエントリが省略されている。$A > B$ のとき $B + C \leq 2B < 2A$。よって u_i では u_j へのエントリが省略されている。w_i でも同様に u_j, w_j のいずれかへのエントリが省略されている。よって $j < i$ より u_i, w_i では少なくとも $i - 1$ 個のエントリが省略されている。したがってまず $u_i, w_i \in Y$ のテーブルでは X の節点へのエントリのうちの半分は省略できているので、すでに m エントリは省略されている。$u_i, w_i \in X$ に対しては $j < i$ を満たす u_j, w_j へのエントリのうち $i - 1$ 個を少なくとも省略できている。この省略に更にヘッダによる省略を加えれば合わせて m エントリとなる。

ここで m の値を評価する。X の節点数は $2m$ であり、Y の節点数は $n - 2m$ である。$u_i, w_i \in X$ において、ヘッダによる省略エントリ数は前小節より $m - i + 1$ である。ここで X の節点で利用されたヘッダの数は Y の節点数 (= 利用できるヘッダ数) 以下なので

$$2\sum_{i=1}^{m} m - i + 1 \leq n - 2m$$

という不等式が成立する。したがって $m(m+1) \leq n - 2m$ より $m = \lceil -2 + \sqrt{n} \rceil$ は上の不等式を満たすので、このとき少なくとも $\sqrt{n} - 2$ エントリが各節点で省略されていることが分かる。すなわちテーブルサイズの上限は $(n - \sqrt{n} + 2)\log n$ である。

3.4 伸長係数 ≤ 2 の証明

次に前述の省略方法で伸長係数が2以下であることを示す。

1. まず目標節点が Y の節点である場合、目標節点のエントリが省略されていたとしてもヘッダによる省略のみである。したがって明らかに最短経路で目標節点に到着できる。

2. 初期節点を x として、$u_i, w_i \in X$ のうち w_i を目標節点とする。x と w_i の最短経路上の任意の節点を y として、x と u_i、y と u_i、u_i と w_i、x と y、y と w_i の最短経路をそれぞれ A, B, C, D, E とする (Fig.5)。

2-1. $A > D+E$ のとき x にいるパケットはヘッダもしくはテーブルを用いて w_i への最短経路に沿って進もうとする。このとき節点 y に対して $D+B \geq A > D+E$ より $B > E$ が成り立つ。したがって y でも w_i への最短経路に沿って進むことがわかる。すなわちこの場合は x から最短距離で目標節点 w_i に到達できる。

$A \leq D+E$ のとき、x で u_i を経由させようとする省略が行われているかどうかでさらに場合分けをする。

2-2. $A+C > 2D+2E$ のとき x では w_i へのエントリは省略されていない。すなわちパケットは w_i への最短経路に沿って進もうとする。このとき y では $D+B+C \geq A+C > 2D+2E$ より $B+C > 2E$ が成り立つ。よって y でも w_i への最短経路に沿って進むことがわかる。すなわちこの場合は x から最短距離で目標節点 w_i に到達できる。

2-3. $A+C \leq 2D+2E$ のとき x では w_i へのエントリは省略されており、x にいるパケットは u_i への最短経路に沿って進もうとする。

2-3-1. x と u_i の最短経路上の任意の節点で w_i への行き先が見つからなかった場合、パケットは u_i に到着してから w_i へ向かう。このとき u_i から w_i への最短経路上の節点では w_i へのエントリは省略されていないので省略が行われた条件より最短距離の 2 倍以内で w_i に到達できる。

2-3-2. パケットが u_i に着く前に w_i へのエントリが見つかり、その節点が u_i, w_i の最短経路上に存在しない場合、x と u_i の最短経路上で初めて w_i へのエントリが見つかる節点を z、z と w_i の最短経路上の節点を z' とし、改めて x と z、z と u_i、z と u_i、x と w_i、z と z'、z' と u_i、z' と w_i、u_i と w_i の最短経路をそれぞれ A,B,C,D,E
,F,G とする (Fig.6)。z が G 上の節点ではない場合、仮定より w_i のエントリが省略されていない。また $A+B \leq C \leq A+D+F$ より $B \leq D+F$ なので $B+G > 2(D+F)$ が成立する。よって $D+E+G \geq B+G > 2(D+F)$ より $E+G > 2F$ なので z' でも w_i のエントリは省略されていない。したがってパケットは ADF という経路を通ることになり、仮定より $A+D+F \leq A+B+G \leq 2C$ であるので最短距離の 2 倍以内で到達可能である。

2-3-3. z が u_i, w_i 最短経路上にあった場合 (Fig.7)、このとき G 上の節点では w_i へのエントリは省略していないのでパケットが到着した G 上の節点からは必ず最短距離で行ける。パケットが通る距離は $A+D$ であり、$A+D \leq A+B+E \leq 2C$ なので最短距離の 2 倍以内で到着できる。

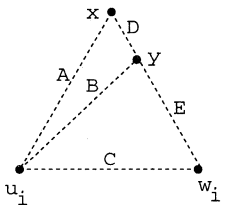

Fig.5

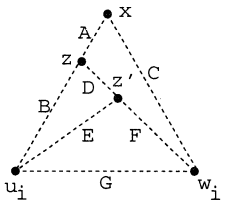

Fig.6

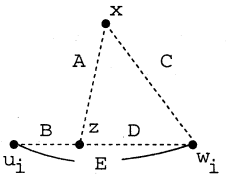

Fig.7

ここでループができないことを保証する。

補題 2 パケットは同一節点を二度通ることはない。

証明 パケットがまっすぐ目標節点 w_i を目指す場合は明らかである。x から u_i(副目標) もしくは z(初めて w_i) へ到着してから w_i を目指す場合、z もしくは u_i に到着するまで w_i へのエントリが存在しない節点を通っているが z, u_i より先では全ての節点で w_i へのエントリが存在する。すなわちパケットが通る x から u_i, z への経路と u_i, z から w_i への経路は別の節点で構成されている。■

3. 目標節点が u_i の場合でもほぼ同様にして最短距離の 2 倍以内で到達可能であることが証明できる。

以上より最短距離の 2 倍以内でパケットは目標節点に到達できる。■

4 テーブルサイズの下限

定理 4 第 2 節で述べたモデルの下で伸長係数 3 未満を保証するどんなアルゴリズムでもテーブルサイズが $(n-2\sqrt{n})\log n$ は必要な節点を持つ n 節点のグラフ G が存在する。

証明 (Fig.8) のようなグラフ $G=(V,E)$ を考える。ここで $|(u_i, v_j)| = 1\ (i \leq m,\ j \leq s)$, $|(u_i, u_j)| = 2\ (i, j \leq$

m) である。なおこのグラフは三角不等式を満たす。u_i にいるパケットが v_j を目標とするとき、別の任意の節点を経由させても必ず 3 以上かかり、最短経路は明らかに 1 である。つまり最短距離の 3 倍を切るためには直接 v_i に行かなければならない。ただし上限の証明のときに使ったヘッダによる省略と同様の手法を用いることで一部省略可能である。すなわちヘッダを用いて 1 つの節点あたり 1 個の節点からはエントリがなくても直接向わせることが可能である。各 u_i から均等にヘッダによりエントリを減らしていくと、u_i のテーブルから減らせるエントリはせいぜい $\frac{s}{m}$ 個である。したがって u_i のテーブルには少なくとも $s - \frac{s}{m}$ 個のエントリが必要となる。$m = \lceil \sqrt{s} \rceil$、$n = m + s$ とおくことにより $s - \frac{s}{m} = s - \sqrt{s} > n - 2\sqrt{n}$。したがって u_i のテーブルには少なくとも $n - 2\sqrt{n}$ エントリが必要である。■

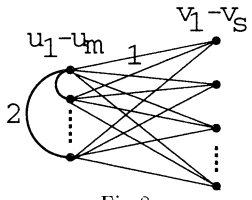

Fig.8

5 結論

以上より [Cow99] のモデルの下で最短距離の 3 倍未満のラウティングを保証するとき、テーブルサイズの上下限は $(1 - o(1))n \log n$ であることが分かった。本稿の結果より、妥当な拡張により伸長係数 3 を下回りながら本質的に良いメモリ効率を持つアルゴリズムが設計できるような別のモデルが存在するだろうか、という疑問が浮かび上がってくる。このモデルを少し拡張してヘッダの 2 番目、3 番目の要素が定数個を持つことを許すと仮定する。すなわち $H(v) = (v, v_1, ..., v_k, (x_1, p_1), ..., (x_l, p_l))$ というヘッダを用いる (k, l は定数)。このときヘッダの長さが定数倍になったところで本質的に結果が良くならないことが分かっている。すなわち同じサイズのヘッダでもその構成から根本的に変えなければ、良いメモリ効率を持つアルゴリズムはできないと考えられる。あるいは妥当なモデル、例えばヘッダ長が $O(\log n)$ であるといったものでは効率の良いアルゴリズムが存在しないことも考えられる。

参考文献

[ABLP90] B. Awerbuch, A. Bar-noy, N. Lininal and D. Peleg, "Improved routing strategies with succinct tables", *Journal of Algorithm*, 307-341, 1990.

[Cow99] L. Cowen, "Compact Routing with Minimum Stretch", *Proc. SODA '99*, 255-260, 1999.

[EGP98a] T. Eilam, C. Gavoille and D. Peleg, "Compact Routing Schemes With Low Stretch Factor", *in Annual 17th ACM Symposium on Principles of Distributed Computing(PODC)*, 11-20, 1998.

[EGP98b] T. Eilam, C. Gavoille and D. Peleg, "Compact Routing Schemes With Low Stretch Factor", *Laboratoire Bordelais de Recherche en Informatique Research Report*, RR-1195-98, 1998.

[FG95] P. Fraigniaud and C. Gavoille, "Memory Requirement for Universal Routing Schemes", *in Annual 14th ACM Symposium on Principles of Distributed Computing(PODC)*, 223-230, 1995.

[GG97] C. Gavoille and M. Gengler, "Space-Efficiency for Routing Schemes of Stretch Factor Three", *4th International Colloquium on Structural Information and Communication Complexity*, 162-175, 1997.

[GP96] C. Gavoille and S. Pérennès, "Memory requirement for Routing in Distributed Networks", *in Annual 15th ACM Symposium on Principles of Distributed Computing(PODC)*,125-133 , 1996.

[LT86] J. van Leeuwen and R.B.Tan, "Computer networks with compact routing tables", *The book of L*, Springer-Verlag, 259-273, 1986.

[PU89] D. Peleg and E. Upfal, " A Trade-Off between Space and Efficiency for Routing Tables", *Journal of the ACM*, 36, 510-530, 1989.