〔公　開〕

ＴＲ－Ｃ－０１５４

危険な情報フロー削除手法

沖　也寸志
Yasushi OKI

１９９６　３．１５

ＡＴＲ通信システム研究所

危険な情報フロー削除手法

沖　也寸志

１９９６年３月

目次

あらまし

本報告では，コンピュータシステムにおいて機密漏洩の原因になりうる情報フロー，すなわち危険な情報フローを取り除く技術を提案する．

あるコンピュータシステムで機密漏洩が起きる場合には，機密情報が到達すべきでないエンティティに到達可能となる，機密情報の情報フローが存在しているはずである．特に，システムに厳密な機密性が要求される場合には，起こりうるすべての情報フローを想定し，もし危険な情報フローが生じる可能性があるならば，その発生を防止する必要がある．情報フローはデータへのアクセスに伴って起きるので，危険な情報フローができないようにするためには，危険な情報フローに関係するアクセスの一部を禁止せざるを得ない．しかしながら，このアクセス禁止という措置は一般的にシステムの可用性を下げることにつながるので，必要最小限に抑えるべきである．

本報告では，まず，危険な情報フローを取り除く問題は，データベースシステムのみならずネットワークシステムをも包含するコンピュータシステムの機密性を守るために重要，かつ基本的な問題であることを述べる．次にこの問題はＮＰ困難であることを述べ，その問題を解くための手法として集合カバー問題に対して研究された技術が利用できることを述べる．さらに，もっと効率良く解くための試みとして問題サイズ削減手法を提案し，その手法の評価実験結果について述べる．最後に，まとめとして研究の成果と今後の課題について述べる．

１．はじめに

近年，通信技術やネットワーク技術の進歩が急速に社会の情報化を押し進めてきており，情報の価値が高く評価されるようになっている．その結果，広く共有されるべき情報に対しては，インターネット[平原95]に代表されるような，情報の流通を促進する技術がますます重要になってきている．一方では，流通させてはならない情報も増加しており，情報を流通させないための技術，すなわち情報の機密性を実現するセキュリティ技術も重要になってきている．しかしながら，情報はしばしば上記の両方の側面を持つことを考えると，情報化社会では両方の技術の均衡が取れている必要がある．本報告では，データベースシステムだけでなく，ネットワークシステムをも視野に入れたコンピュータシステムを対象とし，アクセス制御というアプローチから情報の機密性を実現する手法について述べる．

あるシステムで機密漏洩が起きる場合には，機密情報が到達すべきでないエンティティに到達可能となる機密情報の情報フローが存在しているはずである[Denning 83]．厳密な機密性が要求される情報を扱うシステムでは，起こりうるすべての危険な情報フローを防止するために，直接的情報フローだけでなく間接的情報フローにも注目し，もし危険な情報フローパスが生じる可能性があるならば，そのパスの発生を防止する必要がある．情報フローはデータへのアクセスに伴って起きるので，システムに対するアクセス要求をすべて許可したとき機密要求に反する危険な情報フローができる恐れがある場合には，アクセス要求と機密要求は矛盾している．矛盾解消のためには，その情報フローが切断されるように，情報フローに関係するアクセスのうち一部を禁止せざるを得ない．ところが，アクセス禁止という措置はシステムを制限し，システムの可用性が下がることにつながるので，禁止するアクセスの決定は慎重に行われる必要がある．形成される恐れのあるすべての危険な情報フローの形成を防ぐだけでなく，可用性の低下を最小に抑えることが望まれている．このセキュリティ設計は，システムの規模と複雑さの増加につれて困難な問題になるので，人手で設計できるものではなく，システムによる自動的な設計が望まれる．

これまでに特定のデータモデル（オブジェクト指向データベース）に対して，人が介在する矛盾解消手法[荒木 93, Oki 95]が報告されているが，本報告は汎用的に使える技術をめざしており，一般的なデータモデルに対して可用性を考慮した矛盾解消手法について提案する[Oki 96]．この手法は，たとえばデータベースシステムやネットワークシステムのうち，厳密な機密性が要求されるシステムのアクセス制御の設計に利用可能であり，設計工数の削減が期待できる．

以下，２節では上記で述べたセキュリティ設計を形式的な問題として定義し，その問題がＮＰ困難な組み合わせ最適化問題であることを述べる．３節では，その問題を解くための手法として，集合カバー問題のための成果[Balas 80][茨木 83]が有効であることを述べる．４節では，さらに効率良く解くための試みとして問題サイズ削減手法を提案し，５節でその手法の評価実験結果について述べる．最後に６節でまとめとして研究成果と今後の課題を述べる．

2．危険な情報フローパス除去問題

この節では，危険な情報フローを防止するためのセキュリティ設計を形式的な問題として整理し，その問題の性質を明らかにする．

2．1　定義

[定義1] システムモデル：システムの構成要素を抽象的にエンティティと呼び，その集合をEで表わす．能動的エンティティを主体，受動的エンティティを客体と呼ぶ．両方の役割を持つエンティティも存在する．

□

[定義2] 情報フローグラフ IFG：有向グラフ IFG=(E, DIF) である．
ここでEは，システムのエンティティの集合である．
DIFは E×E の部分集合であり，エンティティ間の直接的情報フローの集合である．(e1, e2)∈DIF が存在するとき，情報フローの向きは e1→e2 である．

□

情報フローグラフは，エンティティが頂点に，直接的情報フローが弧に対応している有向グラフである．

[定義3] 情報フローパス：IFG において，(e1, e2), (e2, e3), ..., (en-1, en) という弧の列を情報フローパスと呼ぶ．

□

[定義4] アクセス要求：主体が客体に対して行いたい要求である．各要求は情報フロー (e1, e2) で表わされ，情報フローの向きは e1 から e2 である．A はアクセス要求の集合を表わす．

□

[定義5] 機密要求 ：ある主体 s と客体 o に関して，o の情報が s に到達してはならないというセキュリティ上の要求であり，二項組 (o, s) で表わす．C は機密要求の集合を表わす．

□

[定義 6] 危険な情報フローパス：すべてのアクセス要求が許可された場合の IFG 上の情報フローパスのうち，始点 o と終点 s の組 (o, s) が機密要求に一致しているものを危険な情報フローパスという．
危険な情報フローパスが存在するなら，アクセス要求と機密要求は矛盾しているという．また，危険な情報フローパスを矛盾パスという．

□

[定義 7] アクセス禁止コスト：各アクセス要求 a∈A に対して，a を禁止したときのシステムの可用性の低下量を示す，各a∈A に対してアクセス禁止コスト c(a)>0 が与えられている．

□

矛盾パスが存在するなら，そのパスを通じた情報フローが機密を漏洩するかもしれない．矛盾を解消するには，矛盾パスを構成しているアクセスの一部を禁止しなければならない．

しかしながら，アクセス禁止はシステムを制限し，可用性を下げることにつながるので，必要最小限であることが望ましい．

この矛盾解消問題は，以下のように定義される．

[定義 8] 矛盾パス除去(CPE)問題：矛盾パスの集合 P＝{pi | 1≦i≦m} が与えられたとき，パスを構成している弧の集合 A={aj | 1≦j≦n} であるとき，以下の条件を満たすアクセス要求 A の部分集合 Amin を求める．
（条件１） Amin に属すアクセス要求をrejectすれば，すべての危険な情報フローパスが切断される．
（条件２） すべての a∈Amin に対する c(a) の総和が最小

□

上記の定義は与えられる矛盾パスがその一部に閉路を持つか否かについて言及していないが，それらのパスの内，仮に閉路を持つパスが含まれているならば，それらを除外してよい．この性質については後述するが，与えられるパスは閉路を持たないと仮定して話を進める．したがって条件１は，下記の単純な条件１′と置き換えてかまわない．

（条件１′）パスは，少なくとも一つの Amin に属す弧を含む．

[問題例]　図２．１(a)はアクセス要求 A={a1, a2, ..., a7} から成る情報フローグラフと各アクセスの禁止コストを表わしている．図２．１(b)は，情報フローグラフ上の矛盾パス P={p1, p2, ... , p10} を表わしている．これらを入力としたCPE問題について考える．まず，c(a6)=c(a7)=∞ なので，a6 または a7 が最適解に含まれないことは明らかである．

他のアクセス禁止コストはすべて１なので，禁止するアクセス数が最小ならば，総コストが最小になる．ここで，ある弧 a がすべてのパス中に現われる回数を出現回数 N(a) とする．直観的に出現回数最大の弧を削除すると良い解を見つけることができそうである．なぜなら，ある弧の出現回数はその削除により切断されるパスの数に等しいからである．この例に出現回数最大を選ぶルールを適用すると，まず N(a5)=6 が最大なので，a5 が選択される．a5 を削除すると，残りの弧 a1 から a4 の出現回数はすべて２になる．ここで４本のパスが残っているので，少なくとも２つの弧を禁止しなければならないから，全部で３つの弧を禁止することになる．ところが，この例の最適解は {a1, a2} である．

□

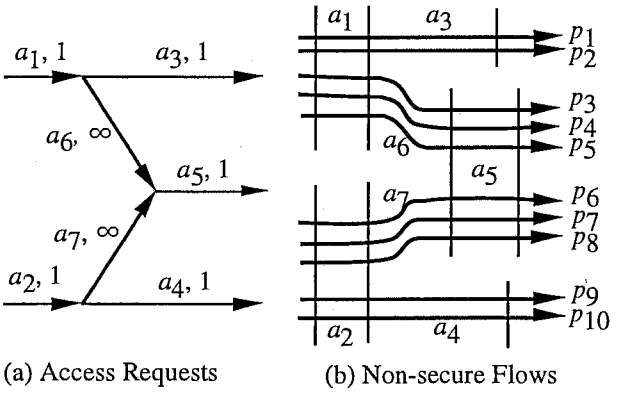


図２．１　問題例

２．２　問題の性質

定義８でCPE問題を最適化問題として定義した．今これを以下のように決定問題として扱った場合，この問題はＮＰ完全である．

[定義９]　CPE問題（決定問題）：矛盾パスの集合 $P=\{p_i \mid 1 \leqq i \leqq m\}$，パスを構成している弧の集合 $A=\{a_j \mid 1 \leqq j \leqq n\}$ と正整数 $k \leqq |A|$ が与えられたとき，A は k 個以下の A の部分集合で，かつすべての P を切断可能な A' が存在するか？すなわち，すべてのパス $p \in P$ が少なくとも一つの弧 $a \in A'$ を含むか？

□

[定理１]　CPE問題（決定問題）はＮＰ完全である．

CPE問題において，各アクセス禁止コストが同じであるように制限した問題は，クラス ＮＰ に属すことは明らかである．さらに，集合カバー問題はＮＰ完全であることが知られており，任意の集合カバー問題は上記の制限されたCPE問題に多項式的に帰着可能である．したがって，制限しないCPE問題もＮＰ完全である．

□

上記で言及した集合カバー問題は次のとおりである．

[Set Covering Problem (Mimimum Cover)] [GS 79]
Instance: Collection C of subsets a finite set S, positive integer $K \leqq |C|$.
QUESTION: Does C Contain a cover for S of size K or less, i.e., a subset $C' \subseteq C$ with $|C'| \leqq K$ such that every element of S belongs to at least one member of C.
Comment: Remains NP-complete even if all $c \in C$ have $|c| \leqq 3$.

□

以上のことから，問題サイズが大きくなるとCPE問題（最適化問題）を実用時間内に解くことはきわめて困難であると考えられる．

3．最適解へのアプローチ

CPE問題と集合カバー問題は単純な対応関係で相互に帰着できるので，集合カバー問題を実用時間内に解くための研究成果は，CPE問題を解くために適用可能である．

集合カバー問題に対しては盛んに研究がなされており[茨木 83]，ある程度の規模まで実用的に解ける解法が報告されている[Balas 80]．その基本技術は，分枝限定法，数理計画法の分野の線形計画法や整数計画法およびヒューリスティックである．これらの技術はCPE問題にも有効である．

一般に組み合わせ最適化問題を解くための所要時間は問題のサイズと深い関係がある．特に問題を解くための最悪計算量が指数関数的である場合に顕著である．与えられた問題に対してある処理を行うことにより問題サイズを削減でき，かつ削減された問題を解くことにより元の問題の最適解が得られるならば，与えられたCPE問題をさらに効率良く解くことができる可能性がある．

4．問題サイズ削減手法

本節では，問題サイズ削減について述べる．

4．1　パスの除外

与えられたCPE問題において，最適解を求めることに関して，不必要なパスが存在するかもしれない．この節で，不必要なパスの性質について述べる．

[定理2] あるCPE問題 P において，与えられた矛盾パス集合Pに属す2つのパスpaとpbに関して，パスpaの除去により必ずpbも除去されるならば，パス集合がP-{pb}である問題を $P1$ とするとき， P と $P1$ の最適値は等しい．

問題 P と $P1$ の最適値が異なると仮定したとき矛盾が生じる．

□

定理2から，以下の系が導き出される．

4．1．1　閉路

CPE問題の定義から，入力として閉路を持つ矛盾パスpが与えられているならば，pから閉路が除かれただけのパスp1も必ず存在している．さらに，p1が除去されたら必ずpも除去される．したがって，系2．1が成立する．

[系2．1] あるCPE問題 P において，与えられた矛盾パス集合Pに属すあるパスpがその一部に閉路を持っているならば，パス集合がP-{p}である問題を $P1$ とするとき， P と $P1$ の最適値は等しい．

□

系2．1から，CPE問題の入力である矛盾パスのうち閉路を持つパスがあった場合，そのパスを除いてもかまわない．もっと言えば，アクセス要求と機密要求から矛盾パスを求めるときに，閉路を持たないパスのみを求めれば十分である．

これ以降は，矛盾パスは閉路を持たないと仮定して議論する．

4．1．2　部分グラフ

矛盾パスpaがpbの部分グラフの関係にあれば，paの除去によりpbも除去されることから，系2．2が成立する．

[系2．2] あるCPE問題 P において，与えられた危険な情報フローパス集合Pに属すパスpaとpbに関して，パスpaがpbの部分グラフであるならば，与えられるパス集合がP-{pb}である問題を $P1$ とするとき， P と $P1$ の最適値は等しい．

□

系２．２から，CPE問題の入力として与えられる矛盾パス集合のうち，他のパスを包含するパスが存在する場合は，包含する側のパスを除くことにより，問題サイズを小さくできる．

なお，系２．１の閉路に関する性質は，実は系２．２の部分グラフに関する性質の特別な場合である．

４．２　弧の除外

最適値を与える弧の集合に必ず含まれる弧が明らかな場合，次の性質が成立する．

[定理３]　最適値がzであるCPE問題Pにおいて，弧 a は常に最適解に属すならば，P から弧 a を削除して得られる問題を $P1$，その最適値を z1 とするとき，z=c(a)+z1 が成り立つ．

z=z1+c(a) でないと仮定すると矛盾が生じる．

□

長さ１の矛盾パスがあれば，そのパスの除去は，パスを構成している唯一の弧 a を削除する以外にはない．したがって，任意の最適解は必ず a を含んでいるので，以下の系が成立する．

[系３．１]　最適値が z であるCPE問題 P において，長さ１の矛盾パス p が存在し，弧 a は p を構成する唯一の弧であるならば，P から弧 a を削除して得られる問題を $P1$，その最適値を z1 とするとき，z=c(a)+z1 が成り立つ．

□

4．3　問題分割

ある問題を独立ないくつかの問題に分割し，それらの個々の問題の最適解を求めることにより，元の問題の最適解を求める手法について述べる．

[定理4] あるCPE問題の矛盾パス集合Pとそれを分割したP1, P2に関して，最適値がそれぞれz, z1, z2, P1, P2に属すパスを構成する弧の和集合がそれぞれA1, A2であるとき，$A1 \cap A2 = \phi$ ならばz=z1+z2を満たす．

z=z1+z2でないと仮定すると矛盾が生じる．

□

定理4では与えられた矛盾パス集合が2つに分割できる場合の性質について述べている．しかしながら，問題サイズ削減の目的は，問題を2つに分割することではなく，問題を可能な限り小さくすることなので，分割は以下のとおり，分割ができなくなるまで続けるべきである．

矛盾パス集合の分割

入力：

矛盾パスの集合 $P=\{p \mid p\text{はパス}\}$

出力：

以下を満たす，Pの部分集合 P1, P2,…, Pm

(i) $P1 \cup P2 \cup \cdots \cup Pm = P$

(ii) $\forall Pi, Pj \subseteq P\ (Pi \neq Pj \Rightarrow \forall px \in Pi \forall py \in Pj\ (\forall aa \in px \forall ab \in py\ (aa \neq ab)))$

(iii) $\forall Pi \subseteq P\ (|Pi| > 1 \Rightarrow \forall px \in Pi \exists py \in Pi\ (px \neq py \wedge \exists aa \in px \exists ab \in py\ (aa = ab)))$

４．４　アルゴリズム

この節では，問題サイズ削減アルゴリズムについて述べる．処理の入力は，あるCPE問題の矛盾パス集合 P である．出力は，入力で与えられた矛盾パス集合のサイズを削減し，さらに分割することにより得られたパスの集合列 P1, P2, ..., Pm である．最終的に得られたパスの集合列 P1, P2, ..., Pm のうち，空でない集合が独立した部分問題であり，それぞれが一つのCPE問題である．

一般に組み合わせ最適化問題において最悪計算量が指数関数オーダーになる場合には，問題サイズの増加につれて組み合わせ数が爆発的に増加するので，問題サイズの増加はわずかでも計算時間は爆発的に大きくなる傾向がある．その結果，実用時間内に問題を解くことができなくなる．逆に言えば，問題サイズがわずかに減少することにより，計算時間が劇的に減少する可能性がある．前節までの議論により，部分問題を独立に解き，各部分問題の最適解および長さ１のパスを構成している弧をまとめることにより得られる解は，元のCPE問題の最適解と一致することが保証されている．さらに，問題サイズ削減処理の時間計算量（表４．１）は，時間的に負担が軽いので，問題サイズ削減を試みる価値がある．

表４．１　削減処理の時間計算量

ステップ1	O（パス数×パス数×弧数）
ステップ2	O（パス数）
ステップ3	O（パス数×弧数）

ここで，パス数とは入力として与えた矛盾パス集合中のパス数であり，弧数は矛盾パス集合のパスを構成している弧の種類数である．

以下は，問題サイズ削減処理のアルゴリズムを疑似C言語で記述したものである．

```
集合列　reduceProblemSize (/*Input*/ 矛盾パス集合P)
{
  P = reduceIncludingP (P);     /* ステップ１：包含しているパスを除去 */
  P = reducePwithLength1 (P);   /* ステップ２：長さ１のパスを除去 */
  return ( divideP (P) );       /* ステップ３：問題分割 */
}
```

```
void reduceIncludingP (P)
{
  P = Pを長さの小さい順にソーティング;
  for (pa = Pから順にpaを選ぶ)
    for (pb = Pから順にpbを選ぶ)
      if (paの長さ＜pbの長さ) then
        if (paがpbに含まれる) then        /* paのすべての弧がpbの弧 */
          Pからpbを除去;
  return (P);
}

void reducePwithLengthl (P)
{
  for (p = Pから順に選ぶ)
    if (pの長さが1) {
      pを構成している弧を最適解に加える;
      pを除去;
    }
}

集合列 divide( P )
{
/* 準備 */
  与えられたパス集合 P={p1,p2, ...,pm} に対して,
      Pの要素を一つずつ含むパス集合P1={p1}, P2={p2}, ..., Pm={pm} を作る;
  それぞれのパスp1からpmで使われている弧の和集合A={a1,a2, ..., an} を作る;

/* 本処理 */
  for (ai = Aから順にaiを選ぶ) {
    S = φ;
    for (Pj = P1からPmの中から順に選ぶ) {
      if (Pjに属すパスの中にaiを弧として含むパスが存在)
        S = S ∪ {Pj};
    }
    while ( |S| ＞ 1 ) {
      Sから異なる２つのPc1, Pc2を選ぶ;
      Pc1 = Pc1∪Pc2;  Pc2 = φ;
      S = S − {Pc2};
    }
  }
  return ( 集合列：P1からPm );
}
```

５．問題サイズ削減実験と評価

５．１　実験のねらい

既に述べたように，CPE問題と集合カバー問題は単純な対応関係で相互に帰着できるので，集合カバー問題を実用時間内に解くための研究成果はCPE問題を解くために適用可能であり，ある程度の規模の問題なら実用時間内に解ける．しかしながら，逆に言えば，解ける範囲を超えた問題を解くのはやはりむずかしいということである．そこで，別のアプローチとして問題サイズの削減に注目し，その手法を提案した．サイズ削減処理を行ってあらかじめ問題サイズを小さくしておくことにより，さらに効率良く解ける可能性がある．したがって，解ける範囲が広がる可能性がある．

本節では，提案した問題サイズ削減手法が有効か否かを明らかにするために，実験により手法の効果を調べる．具体的には，手法全体の効果だけでなく，手法中の各処理はそれぞれどの程度の効果があるのか，どのような性質を持った問題に効果があるのかを明らかにする．

５．２　実験概要

（１）パラメタ

問題のサイズや性質を表現するパラメタについて述べる．この実験で扱ったパラメタは，以下のとおり６つである．

問題を表現するパラメタ：

パス数	m
弧数（種類）	n
総パス長	ΣL
密度	d　＝　総パス長ΣL／（パス数m×弧数n）
平均パス長	Lave　＝　総パス長ΣL／パス数m
平均弧出現回数	Cave　＝　総パス長ΣL／弧数n

このうち，問題を規定するパラメタは，パス数，弧数（種類），総パス長の３つである．残りの３つのパラメタ，密度，平均パス長，平均弧出現回数は，いずれも問題を規定するパラメタから導くことが可能なので，従属的なパラメタである．

（２）　実験手順

実験手順は以下のとおりである．この手順を繰り返すことにより，CPE問題の性質と提案した手法の効果の関係を明らかにする．

実験手順：　与えるCPE問題を変化させながら以下の１から３を繰り返す．

１．入力データ（CPE問題）作成
　所望の性質を持ったCPE問題を作成する．

２．問題サイズ削減手法の適用
　作成されたCPE問題を提案された手法を適用してサイズ削減する．

３．削減効果の収集
　どの程度削減効果があったのか，様々なデータを取得する．

（３）実験システム

実験を行うため小規模な実験システムを作成したので，簡単に述べておく．

実験システムは，問題作成部，削減処理部，結果表示部から構成されている．問題作成部は，作成したい問題の性質がパラメタ表現で記述されているファイルを読み込み，所望の問題を作成する．削減処理部は，作成された問題に削減処理手法を適用し，削減処理を行う．最後に結果表示部は，削減結果のサマリを出力する．

実験システムの仕様

機能　　：　問題作成，削減処理，結果表示
開発言語：　C言語（gcc）
規模　　：　約1500行（コメント行含む）
動作環境：　SunOS 4.1.2

５．３　実験１－均等分布の場合

５．３．１　実験の説明

弧が均等分布している場合に，どのような効果があるかを実験した．均等分布とは，パスと弧（種類）がそれぞれ行列の行と列だと考えたとき，行列の各要素に弧が設定される確率が均等になる場合である．問題のパラメタとして，パス数，弧数（種類），総パス長が決まったとき，ランダムにパスと弧の組み合わせを求め，そこに弧を一つずつ設定し，予定している総パス長に到達するまで弧を発生させた．

以下は実験したパラメタの組み合わせである．

(a) パス数 100,　弧数（種類） 100，総パス長が変動
(b) パス数 100,　弧数（種類）1000，総パス長が変動

実験データの作成のとき，ランダムにパスと弧の組を発生させているので，たまたま既に弧を設定した組に再度設定しようとすることがある．重ねて設定してしまうと予定の総パス長より少なくなるので，それを防ぐためまだ設定されてない組だけに弧の設定をするようにしている．

次に，設定すべき弧が少ない（総パス長が短い）場合には，たとえランダムにデータを発生させても最終的にまったく利用されない弧ができてしまう．ところがCPE問題の意味を考えると，使用されていない弧を考慮してはならないので，弧数としては実際に使用されている弧だけを対象にしている．そのため，設定すべき弧が非常に少ないときは，弧数に関して予定した値と実際に使う値にかなり差がある．

さらに，たまたま複数のパスが全く同じパスになってしまうこともある．この場合は一つだけを残して残りのパスは除外する．これによりパス数も予定の値と実際の値がずれることがある．

なお同じパラメタの設定でも，得られる結果には結構ばらつきがあるので，ばらつきを減らすために一組のパラメタにつき１０回の実験を行い，その平均値を結果として採用した．

５．３．２　評価

実験１の結果は，付録２の図Ａ２．１(a) (b)，図Ａ２．２(a) (b)，図Ａ２．３(a) (b)の６つのグラフにまとめた．また，データ一覧は表Ａ２．１として添付した．

図Ａ２．１(a) (b)：密度と削減率
図Ａ２．２(a) (b)：平均パス長と削減率
図Ａ２．３(a) (b)：弧の平均出現回数と削減率
表Ａ２．１　　：データ一覧

以下に，各削減処理ステップと全体の効果の分析を述べる．

（１）　ステップ１

－密度に関して：密度が小さいほど効果が大きくなっている．
－平均パス長に関して：平均パス長が短いときに効果が大きくなるが，ある値より小さくなると効果が小さくなっている．実験結果からは平均パス長が２から３程度のときに効果が最大になっている．
－平均弧出現回数に関して：平均弧出現回数が小さいときに効果が大きくなるが，ある値より小さくなると効果が小さくなっている．効果が最大になる値は，与えられた問題の傾向によりかなり異なる．

（２）　ステップ２

－密度に関して：密度とは無関係．
－平均パス長に関して：平均パス長が短いほど，効果が大きくなっている．
－平均弧出現回数に関して：平均弧出現回数が小さいほど，効果が大きくなっている．

（３）　ステップ３

－密度に関して：密度が小さいほど効果がある．密度が大きいと効果が小さくなる．ステップ１と同じ傾向．
－平均パス長に関して：平均パス長が短いときに効果が大きくなるが，ある値より小さくなると効果が小さくなっている．ステップ１と同じ傾向．
－平均弧出現回数に関して：平均弧出現回数が小さいときに効果が大きくなるが，ある値より小さくなると効果が小さくなっている．ステップ１と同じ傾向．

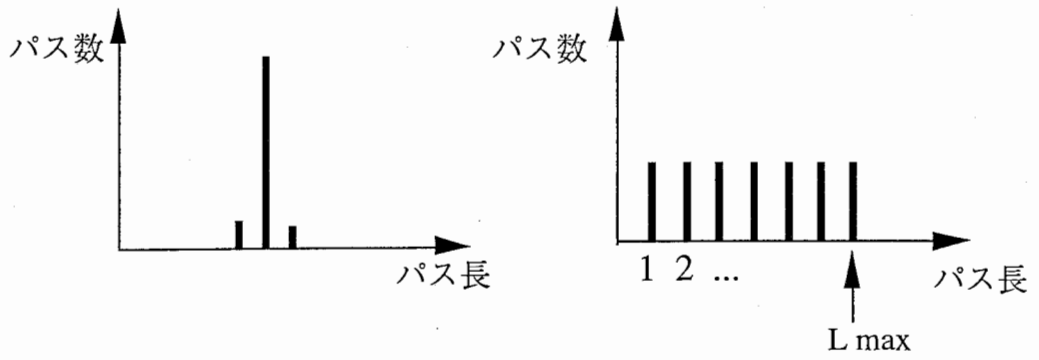
（４）　全体

－密度，平均パス長，平均弧出現回数　のすべてに関して：
パス数に比べて弧数が非常に多い場合はステップ３の効果が出やすいが，パス数に比べて弧数がそれほど多くない（同程度）場合はステップ１の効果が大きい．しかしステップ２の効果が大きくなると，全体の削減効果はほとんどステップ２に依存している．

５．４　実験２ー各パス長同数分布の場合

５．４．１　実験の説明

前節では，均等分布について実験した．均等分布の場合は，各パスの長さがほぼ平均長に一致する．しかしながら，実際のシステムでは，想定される各矛盾パスの長さがほぼ同じという状況は考えにくい．実際にはさまざまな長さのパス長が存在しているはずである．そこで実験２として，さまざまなパス長が存在している「各パス長同数分布」について実験した（図５．１参照）．



(a) 均等分布（実験１）　　(b) 各パス長同数分布（実験２）

図５．１　実験データのパス分布

各パス長同数分布とは，パス長が１からある指定した最大長（Ｌｍ）まで存在しており，各パス長のパス数が同数になっているものと定義する．たとえば，最大長を１０，全パス数が１００としたとき，各パス長のパス数は１０ずつである．すなわち，長さ１のパス１０本，長さ２のパス１０本，…，長さ１０のパス１０本になる．

実験の入力データの作成は，各長さごとに前節で述べた均等分布を使って作成し，最後にまとめる方法で行った．まず，各パス長ごとのパス数（パス数ｍ／最大長Ｌｍ）を求めた後，各パス長ごとに均等分布による弧の設定を繰り返した．さらに話は最初に戻るが，パス数，最大長と整数の関係についても考える必要があった．各パス長ごとのパス数（パス数ｍ／最大長Ｌｍ）が整数にならないときは端数を切り捨てたが，（整数化した数×Ｌｍ）が元のパス数とずいぶんずれる場合は，その最大長（Ｌｍ）について実験しなかった．

以下は実験したパラメタの組み合わせである．

(a) パス数 100，弧数（種類） 100，最大長が変動
(b) パス数 100，弧数（種類）1000，最大長が変動

５．４．２　評価

実験２の結果は，付録２の図Ａ２．４(a) (b)，図Ａ２．５(a) (b)，図Ａ２．６(a) (b)の６つのグラフにまとめた．また，データ一覧は表Ａ２．２として添付した．

図Ａ２．４(a) (b)：密度と削減率
図Ａ２．５(a) (b)：平均パス長と削減率
図Ａ２．６(a) (b)：弧の平均出現回数と削減率
表Ａ２．２　　：データ一覧

以下に，各削減処理ステップと全体の効果の分析を述べる．

（１）　ステップ１

－密度に関して：密度にあまり関係せず，ほぼ一定の効果がある．
特に，パス数に対して弧数（種類）が小さい場合の方が大きい場合よりも効果が大きい．
－平均パス長に関して：平均パス長にあまり関係せず，一定の効果がある．
パス数に対して弧数（種類）が小さい場合の方が大きい場合よりも効果が大きい．
－平均弧出現回数に関して：平均弧出現回数にあまり関係せず一定の効果がある．
パス数に対して弧数（種類）が小さい場合の方が大きい場合よりも効果が大きい．

（２）　ステップ２

－密度に関して：密度の減少（＝最大長の減少）とともに，効果が大きくなっている．
長さ１のパスの比率が大きくなるので，当然の結果である．
－平均パス長に関して：平均パス長の減少（＝最大長の減少）とともに，効果が大きくなっている．
長さ１のパスの比率が大きくなるので，当然の結果である．
－平均弧出現回数に関して：平均弧出現回数の減少（＝最大長の減少）とともに，効果が大きくなっている．
長さ１のパスの比率が大きくなるので，当然の結果である．

（３）　ステップ３

－密度に関して：密度が小さくなるにつれて効果が大きくなっている．
パス数に対して弧数（種類）が大きい場合の方が小さい場合よりも若干効果が大きい．
－平均パス長に関して：平均パス長が小さくなるにつれて効果が大きくなっている．
パス数に対して弧数（種類）が大きい場合の方が小さい場合よりも若干効果が大きい．
－平均弧出現回数に関して：平均弧出現回数が小さくなるにつれて効果が大きくな

っている．
パス数に対して弧数（種類）が大きい場合の方が小さい場合よりも若干効果が大きい．

（４）　全体

－密度，平均パス長，平均弧出現回数に関して：
全体的には，常にステップ１の効果が大きい．
密度，平均パス長，平均弧出現回数の減少とともにステップ２と３の効果が大きくなってくる．
パス数に比べて弧数が多くない（同程度の）場合はステップ１の効果が非常に大きいが，パス数に比べて弧数が多い場合はステップ１の効果は小さくなる．

５．５　効果のまとめ

（１）　ステップ１

ステップ１の効果は包含関係の数に関係するので，以下のことが考えられる．

－ある程度の高い密度（平均パス長が長い）のときに効果的．密度が低ければパス同士の関連が少ないので，パスの包含関係が少ない．密度が高い方がパスの包含関係が生じやすい．しかし密度が高すぎるときは，関連が強すぎて包含関係にならず，効果的でない．
－さまざまなパス長が存在する（パス長が一様でない）．
－同じ始点のパスが多い，あるいは同じ終点のパスが多い．そうであれば同じ弧で始まるパスが多い，あるいは同じ弧で終わるパスが多いと考えられる．

（２）　ステップ２

ステップ２の効果はパス長１のパス数に関係するので，以下のことが考えられる．

－平均パス長が短い場合，長さ１のパスの割合が大きいと考えられる．
－平均パス長が長い場合でも，パス長が広く分布している場合には，長さ１のパスも多くなる．

（３）　ステップ３

ステップ３の効果はパス同士の弧の共有に関係するので，以下のことが考えられる．

－密度が小さい場合，あるいは平均パス長が短い場合には，一般に弧の共有が少ないので，それぞれのパスが独立したグループになる可能性は高い．
－密度が大きい場合，あるいは平均パス長が長い場合には，一般に弧の共有が多いので，それぞれのパスが独立したグループになる可能性は低い．しかしながら，いくつかの弧の共有がないグループに分割できる可能性がないわけではない．幸運にもほぼ同サイズの２つのグループに分割できるかもしれない．最適解を求めるために指数関数的時間がかかることを考えると，その効果は大きい．

（４）　全体

－密度が低い，平均パス長が短い，平均弧出現回数が小さい
処理１　×．パス間の包含関係は少ない
処理２　○．長さ１のパスが現われやすい
処理３　◎．問題分割できる可能性が高い．

－密度が高い，平均パス長が長い，平均弧出現回数が大きい
処理１　○．パス間の包含関係が存在する可能性が高くなる．
処理２　×．長さ１のパスは少ないであろう
処理３　×．問題分割できる可能性は低くなる

6．まとめ

（到達点：何が得られたのか？）

本報告は，厳密な機密性が要求されるコンピュータシステムを実現するために，機密性の本質にかかわる情報フローに注目し，アクセス制御の面から危険な情報フローの削除手法を提案した．採用したシステムモデルは，特定のシステムモデルではなく汎用的でかつシンプルなモデルなので，対象となるコンピュータシステムは，データを管理するデータベースシステムはもちろん，データの授受を行うネットワークシステムをも包含しており，適用範囲が広い．

本報告ではまず，コンピュータシステムにおけるアクセスの設計問題を抽象度の高いレベルで矛盾パス削除問題として定義した．次に，その問題の持つ性質がＮＰ困難な組み合わせ最適化問題であることを述べ，最適解を求めるために，既に知られている集合カバー問題に対する研究成果が適用できることを述べた．さらに，矛盾パス削除問題をより効率良く解くための方法として，問題サイズの削減に関する性質を明らかにし，そのアルゴリズムについて述べた．最後に，問題サイズ削減アルゴリズムの効果を調べるために実験を行い，問題の性質とサイズ削減効果の関係を明らかにし，提案した手法が有効であることを示した．

（限界）

問題サイズ削減手法の有効性は，与えられる問題例に依存する．かなり問題サイズ削減できる場合がある一方で，まったく削減できない場合もある．実験により問題の性質と削減効果の関係がある程度明らかになったが，個々の問題例に対してどのようになるかという正確な予測は困難である．提案したサイズ削減処理は時間的に効率良く実行できるので，個々の問題例に対する効果の予測に重点を置く意味はあまり存在しない．明らかに効果がない場合を除いて，実行してみればよい．

（今後の課題：何が残っているのか？）

危険な情報フロー削除問題の解としてさらに効率良く最適解を見つけるためには，盛んに研究されている分野である最適化の研究を進めることになる．具体的な研究分野としては，整数計画法，分枝限定法，発見的手法（ヒューリスティック）がある．比較的新しいアプローチとしては，遺伝的アルゴリズムの適用も徐々に研究が盛んになっている．

上記は最適解について述べたが，工学的には近似解でかまわない場合も多いので，効率の良い，かつ最適解に近い近似解を求める手法の研究が必要である．近似解を求める場合であっても，一般的に問題サイズが小さいことは有利と考えられるので，問題サイズ削減は行うべきである．

また，本報告の成果を具体的なデータベースシステムやネットワークシステムに対して適用するための応用研究が必要である．

参考文献

[Denning 83] Denning, D. E., "Cryptography and Data Security", Addison-Wisley, 1983.

[茨木83] 茨木俊秀，“組み合わせ最適化　－分枝限定法を中心として－”，産業図書，1983.

[Balas 80] Balas, E. and Ho, A. , "Set covering algorithms using cutting planes, heuristics, and subgradient optimization: A computational study", Mathmatical Programming Study, 12, 37-60, 1980.

[荒木93] 荒木禎史，力石徹也，Hardjono, T.，太田理，“オブジェクト指向データベースのセキュリティ設計支援手法”，信学会 1993年暗号と情報セキュリティシンポジウム SCIS93-14A, Jan. 1993.

[Oki 95] Oki, Y., Chikaraishi, T., Shimomura, T. and Ohta, T. , "A Design Method for Data Integrity in Object-Oriented Database Systems", IEEE SICON/ICIE '95, pp.204-209, Jul. 1995.

[GJ 79] Garey, M. R. and Johnson, D. S., "Computer and Intractability: A Guide to the Theory of NP-completeness", W. H. Freeman and Company, San Fransisco, 1979.

[平原95] 平原正樹，“インタネット”，信学詩，78，4，pp.406-410　(1995-4)

[Oki 96] Oki, Y., Shimomura, T. and Ohta, T. , "An Access Determination Algorithms for Preventing Non-secure Information Flows", IASTED NETWORKS '96, pp.1-4, Jan. 1996.

付録１　定理の証明

矛盾パス除去問題に関する性質を証明する．

[矛盾パス除去(CPE)問題]　矛盾パスの集合 P＝{pi｜1≦i≦m} が与えられたとき，パスを構成している弧の集合 A={aj｜1≦j≦n} であるとき，以下の条件を満たすアクセス要求 A の部分集合 Amin を求める．

（条件１） Amin に属すアクセス要求をrejectすれば，すべての危険な情報フローパスが切断される．

（条件２） すべての a∈Amin に対する c(a) の総和が最小

[定理１]

CPE問題（決定問題）はＮＰ完全である．

[証明]

この定理を証明するために，以下のYES/NOを出力する判定問題を扱う．

[集合カバー問題]

入力：

集合　　S={si｜1≦i≦m}

集合列　C={Sj｜1≦j≦n, SjはS上の部分集合}

正の整数　k≦|C|

出力：

C から k 個以下の要素を選び，それらの和集合が S のすべての要素をカバーするか？

すなわち

I⊆{1, 2, ..., n}，|I|≦k が存在し，$\cup_{j\in I} Sj = S$ が成立するか？

[制限CPE問題]　（アクセス禁止コストはすべて同じ）

入力：

矛盾パスの集合　P={pi｜1≦i≦m, piはパス}，

正の整数k≦|A|，Aは各パスを構成している弧の和集合$A=\cup_{i=1 \text{ to } m} pi = \{aj｜1≦j≦n\}$

出力：

k 個以下の弧を削除することにより，すべてのパス p∈P を切断可能か？

すなわち，

I⊆{1, 2, ..., n}，|I|≦k が存在し，

すべてのj∈I について弧ajを削除すれば，すべてのパスを切断するか？

（１）制限CPE問題　∈NP

集合 {1, 2, ..., n} の部分集合I を非決定的に推測し，

(i)　|I|≦k？

(ii) {aj | j∈I} を取り除けば，すべてのパス pi∈P を切断できるか？

（各pについて少なくとも１つの弧が取り除かれるか？）

を調べることにより解の存在を判定できるので，制限CPE問題は NP で解くことができる．

（２）集合カバー問題は制限CPE問題に多項式的に帰着可能

（２．１）　集合カバー　→　制限CPE

集合カバー問題の入力が

S={si | 1≦i≦m}，C={Sj | 1≦j≦n, Sj は S 上の部分集合}，k ≦|C|

であるとする．

ここで S の要素 si (1≦i≦m) が有向グラフにおけるパスであり，各パス si を構成する弧を

Ci = {Sj | si∈Sj}

の要素と定める．

集合カバー問題に解が存在すると仮定する．すなわち

I ⊆ {1, 2, ..., n} が存在し，

|I| ≦k　かつ　$\cup_{j\in I}Sj = S$

が成立する．

ここで有向グラフにおいて，弧の組 Sj(j∈I) を削除したとき，切断されないパス si が存在したとすると，パス si を構成している弧のいずれも削除されなかったことになる．パスsiを構成する弧を Ci = {Sj | si∈Sj} と定めたので，パス si が残っているならば，解 Sj(j∈I)のいずれも要素 si を含まないはずである．したがって，要素 si が Sj(j∈I) によりカバーされないことになってしまうため，矛盾する．よって制限CPE問題にも解が存在する．

（２．２）　集合カバー　←　制限CPE

逆に，制限CPE問題の入力が

パスの集合　P = {pi | 1≦i≦m, piはパス}，

k≦|A|，A は各パスを構成している弧の和集合　$A=\cup_{i=1\ to\ m}pi$ = {aj | 1≦j≦n}

であるとする．

ここで A の要素 aj (1≦j≦n) が集合であり，各 aj (1≦j≦n) の要素は

Pj = {pi | aj がパスpiの弧}

の要素であると定める．

制限CPE問題に解が存在すると仮定する．すなわち

$I \subseteq \{1, 2, ..., n\}$ が存在し，

$|I| \leq k$, すべてのパスは少なくとも一つの $aj(j \in I)$ から構成される，

が成立する．

ここで集合 $aj(j \in I)$ がある要素piをカバーしていないとする．各 ajの要素を $Pj = \{pi \mid$ ajがパスpiの弧$\}$ と定めたので，解が要素 pi をカバーしないということは，各 $aj(j \in I)$ はパス pi の弧ではないはずである．そうであれば，パス pi を構成する弧が削除されないので，すべてのパスが切断されることに矛盾する．よって集合カバー問題にも解が存在する．

上記で示した問題の変換は明らかに多項式ステップで可能である．以上の議論により，集合カバー問題は制限CPE問題に多項式的に帰着可能であることが示せた．

（３）NP完全

（１）から，各アクセス禁止コストが同じであるように制限したCPE問題は，クラスＮＰに属すことを示した．

さらに（２）から，集合カバー問題はＮＰ完全であることが知られており，任意の集合カバー問題は上記の制限されたCPE問題に多項式的に帰着可能であることを示した．

したがって，制限しないCPE問題もＮＰ完全である．

□

[定理２]

あるCPE問題 *P* において，与えられた矛盾パス集合Ｐに属す２つのパス pa と pb に関して，パス pa の除去により必ず pb も除去されるならば，パス集合が P-{pb} である問題を *P 1* とするとき， *P* と *P 1* の最適値は等しい．

[証明]

問題 *P 1* の最適値 z1 を与える弧の集合を Az1 とする．すなわち，Az1 に属すすべての弧を削除することによりパス集合 P-{pb} に属すすべてのパスが除去される．Ｐに対して Az1 の弧をすべて削除すれば，pa を含んでいる P-{pb} のパスがすべて除去されるので，pb も除去されているはずである．

ここで，z>z1 と仮定すると，問題 *P* に関して Az1 の弧をすべて削除すればｚより小さい最適値 z1 が存在することになりｚが最適値であったことに矛盾する．

次に，z<z1 と仮定する．問題 *P* の最適値ｚを与える解を Az とする．問題 *P 1* に対して Az を削除すれば，すべての弧が削除されるので，z1 より小さい最適値ｚが存在することから，z1 が最適値であったことに矛盾する．

したがって，z=z1 である．

□

[系２．１]

あるCPE問題 P において，与えられた矛盾パス集合Pに属すあるパスpがその一部に閉路を持っているならば，パス集合がP-{p}である問題を $P1$ とするとき，P と $P1$ の最適値は等しい．

[証明]

CPE問題の定義から，入力として閉路を持つ矛盾パスpが存在しているならば，pから閉路が除かれただけのパスp1も必ず存在している．このとき，p1が除去されたら必ずpも除去される．したがって，定理2から系２．１が成立する．

□

[系２．２]

あるCPE問題 P において，閉路を持たない矛盾パスの集合Pに属すパスpaとpbに関して，パスpaがpbの部分グラフであるならば，与えられるパス集合がP-{pb}である問題を $P1$ とするとき，P と $P1$ の最適値は等しい．

[証明]

閉路を持たない矛盾パスpaがpbの部分グラフの関係にあれば，paの除去によりpbも除去される．したがって，定理2から系２．２が成立する．

□

[定理３]

最適値がzであるCPE問題Pにおいて，弧aは常に最適解に属すならば，P から弧aを削除して得られる問題を $P1$，その最適値をz1とするとき，z=c(a)+z1が成り立つ．

[証明]

問題 P のパス集合をP，弧aの削除により除去されるパス集合をPa，問題 $P1$ のパス集合をP1とすると，P=Pa∪P1，Pa∩P1=ϕ が成立している．また，Az1は問題 $P1$ の最適値を与えるから，Az1に属す弧をすべて削除すればパス集合P1のすべてのパスが除去される．したがって，{a}∪Az1の弧をすべて削除すると，Pa∪P1=Pのパスが除去される．

ここでz>c(a)+z1と仮定すると，問題 P に関して{a}∪Az1に属す弧を削除すればzより小さい最適値c(a)+z1が得られることになり，zが最適値であることに矛盾する．

次にz<c(a)+z1，すなわちz-c(a)<z1と仮定する．問題 P に対して弧aを削除すると，残ってパス集合はP1である．同じ問題 $P1$ に関してAz-{a}に属す弧を削除すると，Azを削除したときの最適値z1より小さい最適値z-c(a)が得られることになり，z1が最適解であることに矛盾する．

したがって，z=c(a)+z1が成立する．

□

[系３．１]

最適値がzであるCPE問題 P において，長さ１の矛盾パスｐが存在し，弧ａはｐを構成する唯一の弧であるならば，P から弧ａを削除して得られる問題を $P1$，その最適値を z1 とするとき，z=c(a)+z1 が成り立つ．

[証明]

長さ１の矛盾パスがあれば，そのパスの除去は，パスを構成している唯一の弧ａを削除する以外にはない．したがって，任意の最適解は必ずａを含んでいるので，以下の系が成立する．

□

[定理４]

あるCPE問題の矛盾パス集合Pとそれを分割したP1, P2 に関して，最適値がそれぞれ z, z1, z2，P1, P2 に属すパスを構成する弧の和集合がそれぞれ A1, A2 であるとき，A1∩A2 = ϕ ならば z=z1+z2 を満たす．

[証明]

問題 P，$P1$，$P2$ のパス集合をそれぞれ P, P1, P2，最適値を与える弧の集合をそれぞれ Az, Az1, Az2 とする．Az1 は問題 $P1$ の最適値を与えるから，Az1 に属す弧をすべて削除すればパス集合 P1 のすべてのパスが除去される．同様に，Az2 に属す弧をすべて削除すればパス集合 P2 のすべてのパスが除去される．問題 P は $P1$ と $P2$ に分割されていることから，P=P1∪P2 が成立しているので，Az1∪Az2 に属す弧をすべて削除すればパス集合 P1∪P2=P のすべてのパスが除去される．

ここで z>z1+z2 と仮定すると，問題 P に関して Az1∪Az2 に属すすべての弧を削除すればｚより小さい最適値 z1+z2 が得られることになり，ｚが最適解であることに矛盾する．

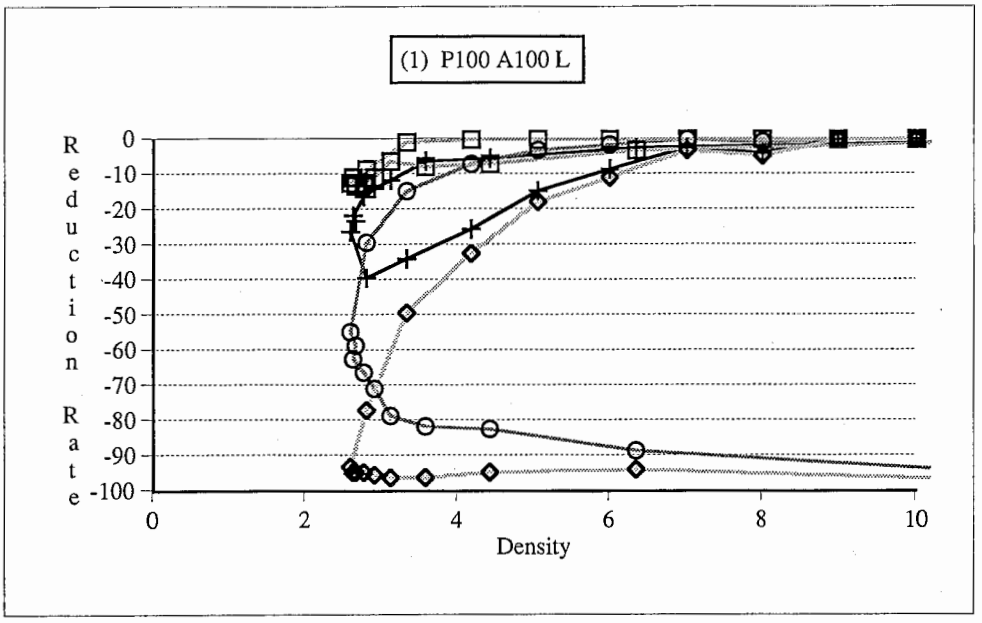
次に z<z1+z2，すなわち z-z1<z2 と仮定する．問題 P に対して，Az1 の弧をすべて削除すれば，P1 に属すパスが除去される．A1∩A2= ϕ から Az1∩A2= ϕ が成立するので，Az1 を除去しても P2 のパスが除去されることはない．したがって残っているパス集合は P2 なので，問題 $P2$ と同じである．同じ問題 $P2$ に関して Az-Az1 に属すすべての弧を削除すれば，z2 より小さい最適解 z-z1 が得られることになり，z2 が最適値であることに矛盾する．

したがって，z=z1+z2 が成立する．

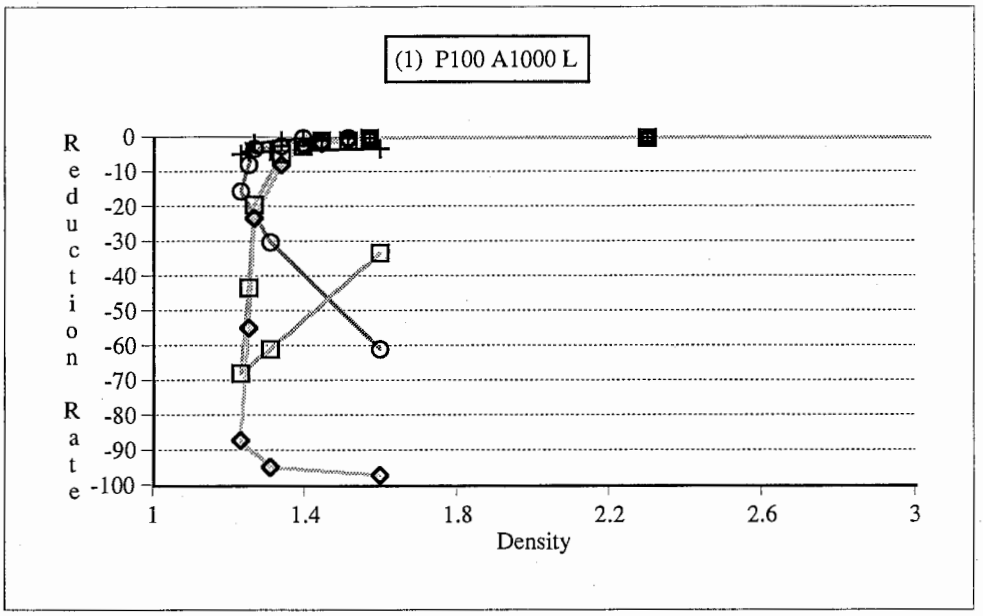
□

付録２　実験結果

＋：ステップ１　○：ステップ２　□：ステップ３　◇：全体



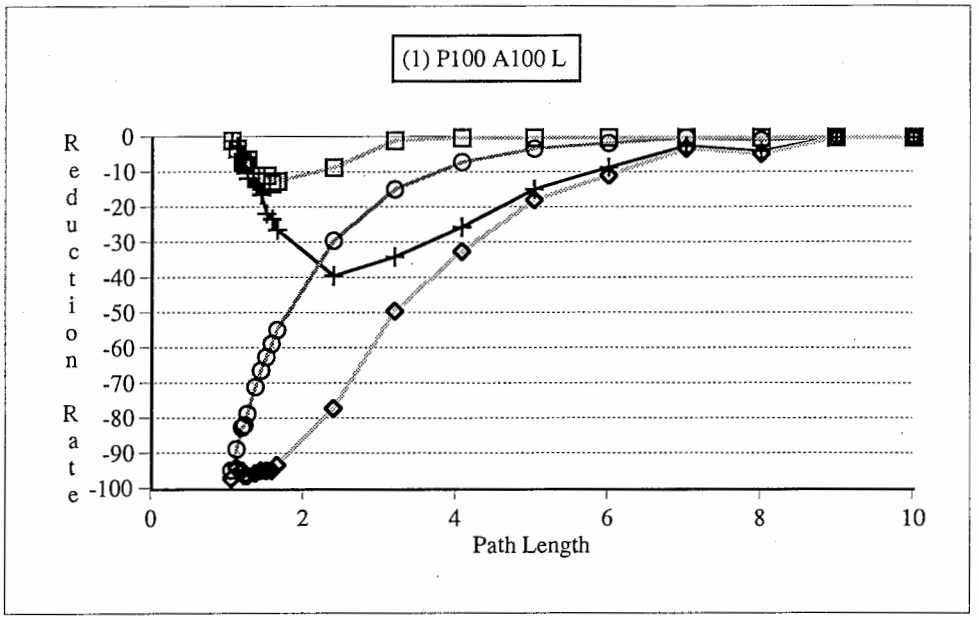
(a) パス数100, 弧数（種類）100



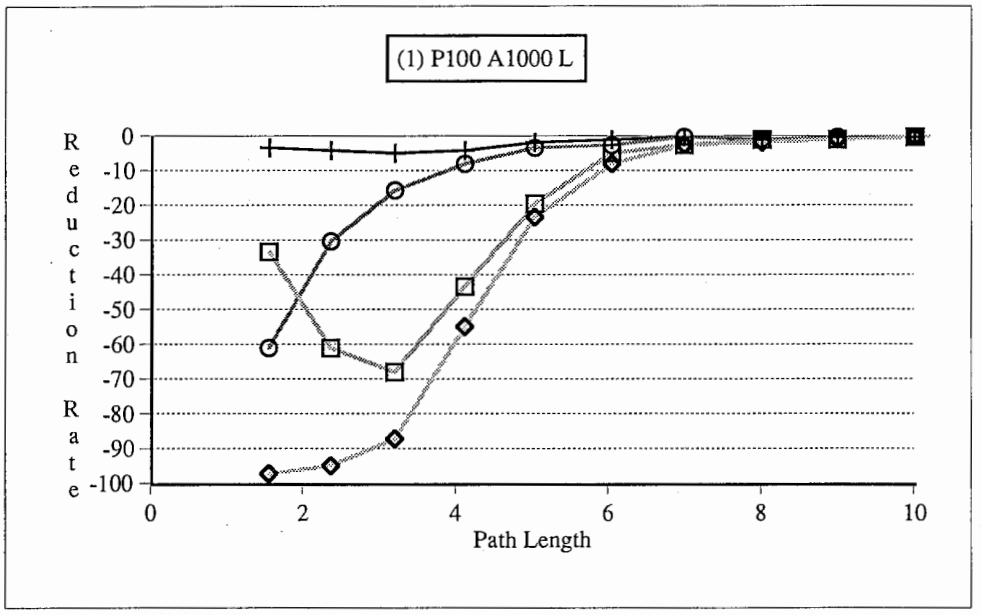
(b) パス数100, 弧数（種類）1000

図Ａ２．１　実験１－密度と削減率

＋：ステップ１　○：ステップ２　□：ステップ３　◇：全体



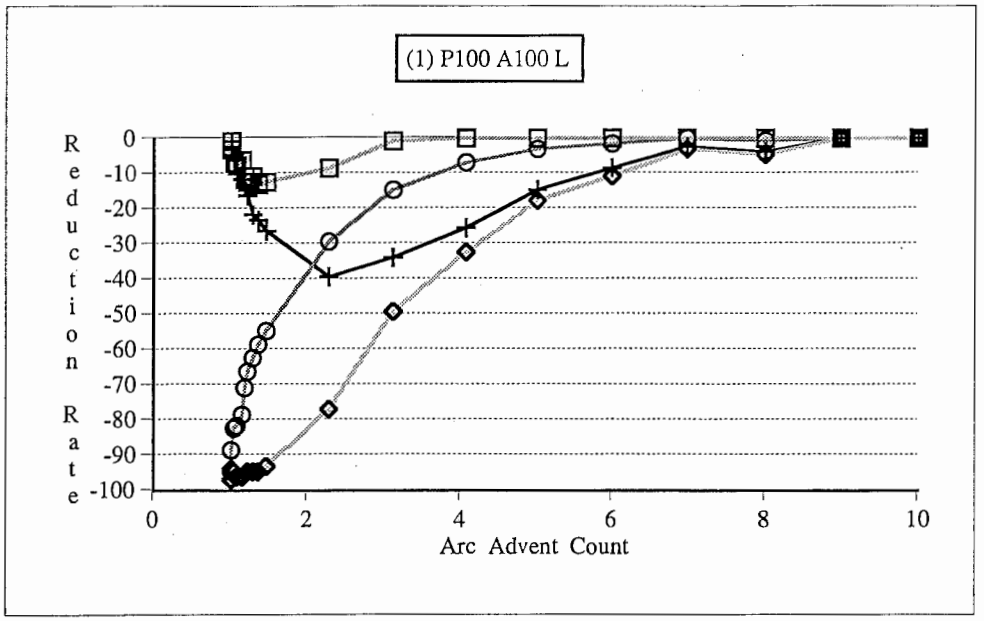
(a) パス数100, 弧数（種類）100



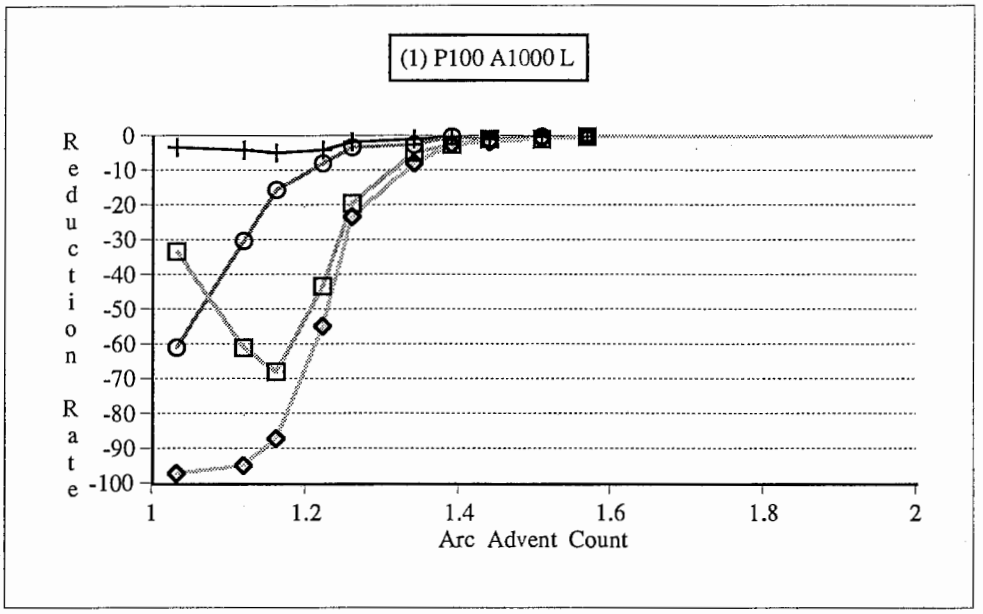
(b) パス数100, 弧数（種類）1000

図Ａ２．２　実験１－平均パス長と削減率

＋：ステップ１　○：ステップ２　□：ステップ３　◇：全体



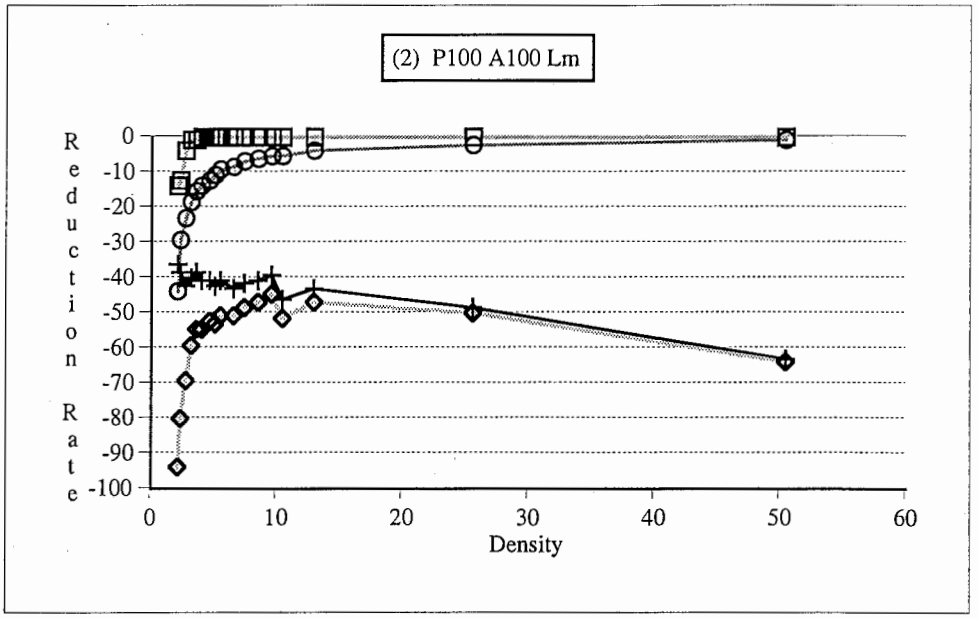
(a) パス数100, 弧数（種類）100



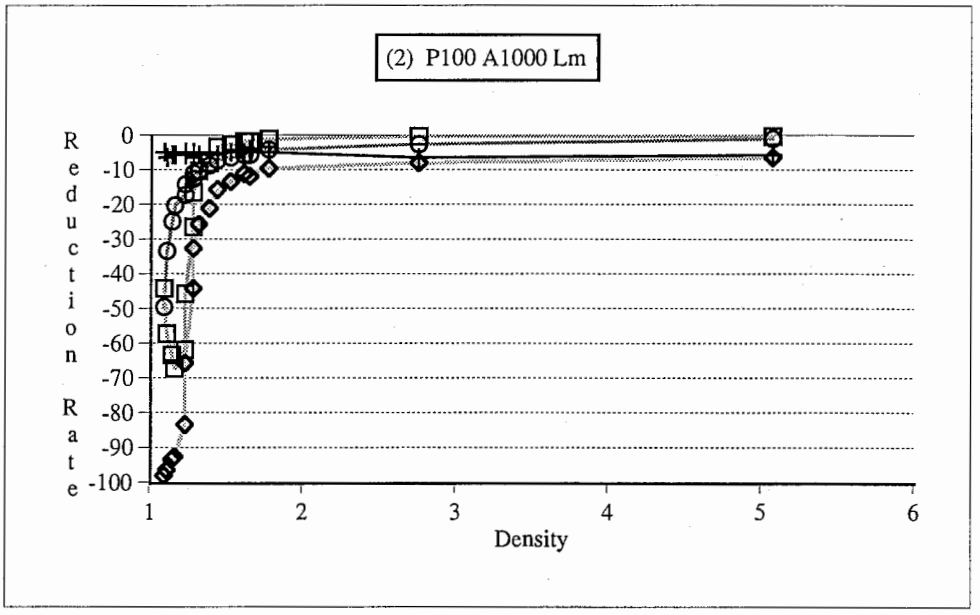
(b) パス数100, 弧数（種類）1000

図Ａ２．３　実験１－平均弧出現回数と削減率

＋：ステップ１　○：ステップ２　□：ステップ３　◇：全体



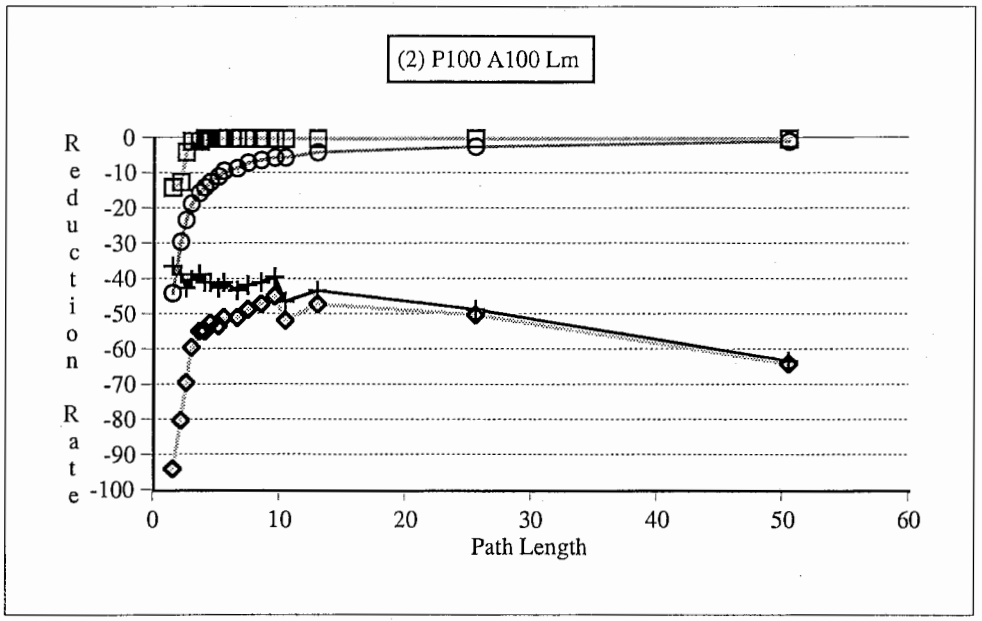
(a) パス数100, 弧数（種類）100



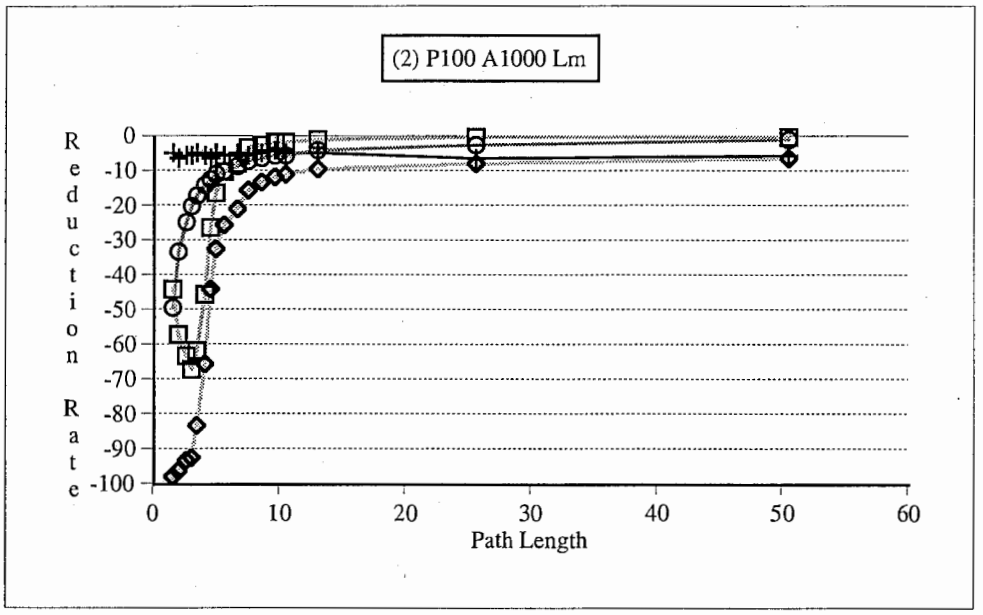
(b) パス数100, 弧数（種類）1000

図Ａ２．４　実験２－密度と削減率

＋：ステップ１　○：ステップ２　□：ステップ３　◇：全体



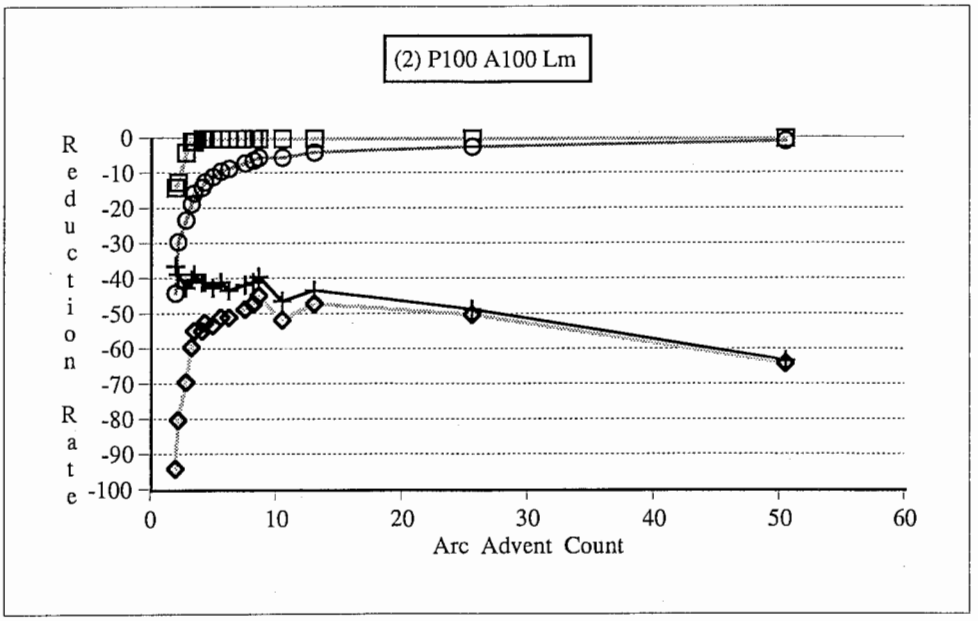
(a) パス数100, 弧数（種類）100



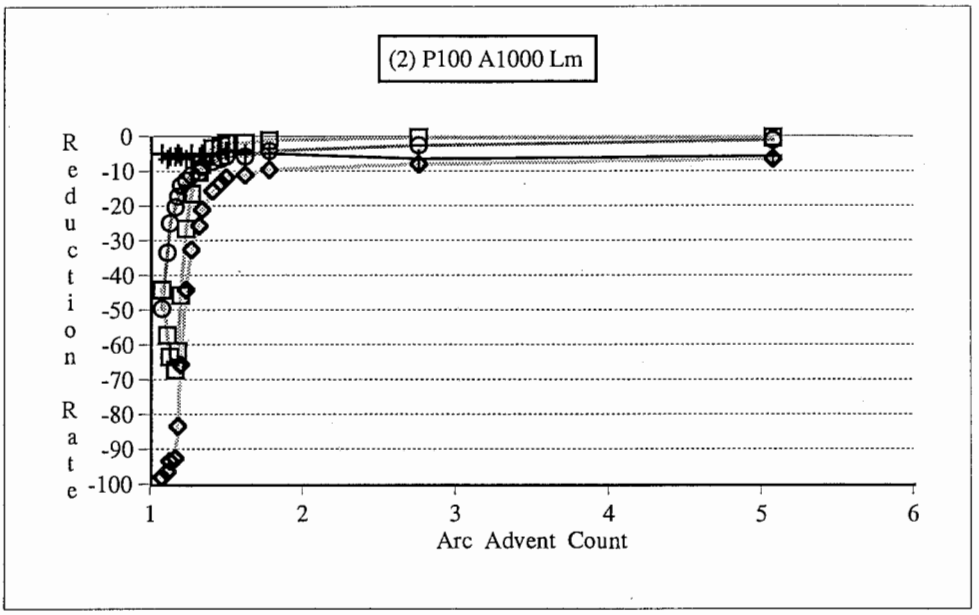
(b) パス数100, 弧数（種類）1000

図Ａ２．５　実験２－平均パス長と削減率

＋：ステップ1　○：ステップ2　□：ステップ3　◇：全体



(a) パス数100, 弧数（種類）100



(b) パス数100, 弧数（種類）1000

図Ａ２．６　実験２－平均弧出現回数と削減率

表Ａ２．１　実験１の結果

(a) パス数100, 弧数（種類）100　（各１０回平均）

Num of paths	Num of arcs	Total length	Density (%)	Average length	Average count	Step1 (%)	Step2 (%)	Step3 (%)	All (%)
100.0	100.0	5000.0	50.00	50.00	50.00	0.0	0.0	0.0	0.0
100.0	100.0	4000.0	40.00	40.00	40.00	0.0	0.0	0.0	0.0
100.0	100.0	3000.0	30.00	30.00	30.00	0.0	0.0	0.0	0.0
100.0	100.0	2000.0	20.00	20.00	20.00	0.0	0.0	0.0	0.0
100.0	100.0	1000.0	10.00	10.00	10.00	-0.1	0.0	0.0	-0.1
100.0	100.0	900.0	9.00	9.00	9.00	-0.1	0.0	0.0	-0.1
100.0	100.0	800.0	8.00	8.00	8.00	-4.2	-0.5	0.0	-4.7
99.8	100.0	700.0	7.01	7.01	7.00	-2.6	-0.2	0.0	-2.8
99.9	99.9	600.0	6.01	6.01	6.01	-8.8	-1.6	0.0	-10.4
99.6	99.4	500.0	5.05	5.02	5.03	-14.7	-2.8	0.0	-17.5
97.9	98.2	399.9	4.16	4.08	4.07	-25.2	-6.7	-0.1	-32.1
93.6	95.8	297.9	3.32	3.18	3.11	-33.5	-14.9	-0.7	-49.1
82.0	85.8	196.1	2.79	2.39	2.29	-39.0	-29.6	-8.2	-76.8
56.9	63.5	93.2	2.58	1.64	1.47	-26.2	-54.8	-12.1	-93.1
52.1	59.2	81.6	2.65	1.57	1.38	-22.8	-58.5	-13.1	-94.4
49.8	56.8	74.6	2.64	1.50	1.31	-21.9	-62.0	-10.8	-94.8
44.8	51.7	64.3	2.78	1.44	1.24	-16.5	-66.1	-12.1	-94.6
41.0	46.7	56.0	2.92	1.37	1.20	-13.7	-70.5	-11.0	-95.1
36.7	39.9	45.9	3.13	1.25	1.15	-11.4	-78.2	-6.5	-96.2
30.2	34.0	36.7	3.57	1.22	1.08	-6.3	-81.8	-7.9	-96.0
24.0	27.2	28.8	4.41	1.20	1.06	-5.4	-82.1	-7.1	-94.6
16.2	17.9	18.4	6.35	1.14	1.03	-2.5	-88.3	-3.1	-93.8
9.3	9.6	9.8	10.98	1.05	1.02	-1.1	-94.6	-1.1	-96.8

(b) パス数100, 弧数（種類）1000　（各１０回平均）

Num of paths	Num of arcs	Total length	Density (%)	Average length	Average count	Step1 (%)	Step2 (%)	Step3 (%)	All (%)
100.0	993.5	5000.0	5.03	50.00	5.03	0.0	0.0	0.0	0.0
100.0	983.6	4000.0	4.07	40.00	4.07	0.0	0.0	0.0	0.0
100.0	948.9	3000.0	3.16	30.00	3.16	0.0	0.0	0.0	0.0
100.0	871.1	2000.0	2.30	20.00	2.30	0.0	0.0	0.0	0.0
100.0	637.7	1000.0	1.57	10.00	1.57	0.0	0.0	-0.1	-0.1
100.0	593.6	899.0	1.51	8.99	1.51	0.0	-0.1	-0.4	-0.5
100.0	554.3	800.0	1.44	8.00	1.44	-0.5	-0.5	-0.8	-1.8
100.0	503.5	700.0	1.39	7.00	1.39	0.0	-0.2	-2.0	-2.2
99.7	447.8	600.0	1.34	6.02	1.34	-0.8	-2.1	-4.6	-7.5
99.6	396.0	500.0	1.27	5.02	1.26	-1.5	-2.8	-19.0	-23.3
97.7	328.4	400.0	1.25	4.09	1.22	-4.0	-7.5	-43.4	-54.9
94.4	257.9	300.0	1.23	3.18	1.16	-4.3	-15.3	-67.5	-87.1
85.0	178.9	199.6	1.31	2.35	1.12	-3.9	-30.0	-61.1	-94.9
64.8	95.9	99.2	1.60	1.53	1.03	-3.1	-61.0	-33.2	-97.2

表Ａ２．２　実験２の結果

(a) パス数100, 弧数（種類）100　（各１０回平均）

Num of paths	Num of arcs	Total length	Density (%)	Average length	Average count	Step1 (%)	Step2 (%)	Step3 (%)	All (%)
100.0	100.0	5050.0	50.50	50.50	50.50	-63.0	-1.0	0.0	-64.0
100.0	100.0	2550.0	25.50	25.50	25.50	-48.3	-2.0	0.0	-50.3
100.0	100.0	1300.0	13.00	13.00	13.00	-43.3	-4.0	0.0	-47.3
100.0	100.0	1050.0	10.50	10.50	10.50	-46.5	-5.0	0.0	-51.5
89.8	100.0	854.8	9.52	9.52	8.55	-39.1	-5.3	0.0	-44.4
95.8	100.0	815.8	8.52	8.52	8.16	-40.7	-6.1	0.0	-46.8
97.7	100.0	734.7	7.52	7.52	7.35	-41.5	-6.9	0.0	-48.3
95.8	100.0	623.8	6.51	6.51	6.24	-42.8	-8.1	0.0	-50.9
99.3	99.6	549.3	5.55	5.53	5.52	-41.1	-9.4	0.0	-50.5
98.4	99.7	494.4	5.04	5.02	4.96	-42.6	-10.6	-0.1	-53.3
95.3	98.8	431.2	4.58	4.52	4.36	-40.5	-12.0	-0.1	-52.6
97.4	98.4	391.4	4.08	4.02	3.98	-40.9	-13.8	-0.2	-54.8
94.8	96.9	334.8	3.64	3.53	3.46	-38.7	-15.6	-0.4	-54.7
98.0	94.9	297.9	3.20	3.04	3.14	-40.1	-18.5	-0.8	-59.4
97.6	91.8	247.5	2.76	2.54	2.70	-42.0	-23.3	-3.7	-69.0
93.0	86.8	191.8	2.38	2.06	2.21	-38.3	-29.2	-12.6	-80.1
88.4	74.5	138.1	2.10	1.56	1.85	-36.4	-43.8	-14.0	-94.2

(b) パス数100, 弧数（種類）1000　（各１０回平均）

Num of paths	Num of arcs	Total length	Density (%)	Average length	Average count	Step1 (%)	Step2 (%)	Step3 (%)	All (%)
100.0	995.3	5050.0	5.07	50.50	5.07	-5.3	-1.0	0.0	-6.3
100.0	923.7	2550.0	2.76	25.50	2.76	-5.8	-2.0	0.0	-7.8
100.0	729.7	1300.0	1.78	13.00	1.78	-4.3	-4.0	-0.9	-9.2
100.0	653.8	1050.0	1.61	10.50	1.61	-4.3	-5.0	-1.2	-10.5
90.0	574.2	855.0	1.65	9.50	1.49	-3.9	-5.6	-1.8	-11.2
96.0	559.7	816.0	1.52	8.50	1.46	-4.7	-6.2	-2.0	-12.9
98.0	525.4	735.0	1.43	7.50	1.40	-5.1	-7.1	-3.4	-15.6
96.0	468.4	624.0	1.39	6.50	1.33	-4.7	-8.3	-7.8	-20.8
100.0	420.5	550.0	1.31	5.50	1.31	-5.5	-10.0	-10.2	-25.7
99.0	389.6	495.0	1.28	5.00	1.27	-4.9	-11.1	-16.0	-32.0
95.8	352.0	431.8	1.28	4.51	1.23	-5.9	-12.3	-25.8	-44.1
97.9	326.9	391.9	1.22	4.00	1.20	-5.3	-14.2	-45.5	-65.0
95.9	286.9	335.9	1.22	3.50	1.17	-4.5	-16.6	-61.6	-82.7
99.9	260.3	299.9	1.15	3.00	1.15	-5.1	-19.9	-67.0	-92.0
99.5	220.4	249.5	1.14	2.51	1.13	-5.0	-24.6	-63.3	-93.0
98.8	180.6	197.8	1.11	2.00	1.10	-6.2	-33.2	-57.1	-96.5
98.2	139.0	148.2	1.09	1.51	1.07	-4.5	-49.1	-44.2	-97.8