3L-1

故障発生時の連結性判定問題を解く分散アルゴリズムについて

和田　幸一† 守谷　幸男† 川口　喜三男† 森下　正浩††
† 名古屋工業大学 ††立石電機(株)

1．まえがき

計算機網の複数台の計算機が各々の局所的な情報から，互いにメッセージを交換することにより，ある問題を解くアルゴリズムを分散アルゴリズムと呼ぶ．従来，種々の分散アルゴリズムが提案されているが，これらの多くは連結な計算機網を対象として考えられている．しかし，実際の計算機網では，リンクや計算機に故障が発生することが少なくなく，これらの故障により計算機網が非連結になることがある．計算機網が非連結であると，計算機網に通信不可能な計算機が存在することになり，従来の分散アルゴリズムがうまく動作しない場合がある．従って，計算機網におけるリンクや計算機で故障が発生した場合に計算機網が連結であるかどうかを判定することは重要である．

本稿では，まず故障をリンクに限定し，非同期式計算機網においてリンクに故障が発生した場合に計算機網が連結であるかどうかを判定する問題（リンク故障時の連結性判定問題（ＣＬＦと呼ぶ））を解く分散アルゴリズムを，以下の１～３の場合について考察する．

１．各計算機が基本情報（自分自身の識別子と接続しているリンク数）だけを保持している場合（基本情報モデル）．

２．各計算機が基本情報と隣接計算機の識別子を保持している場合（隣接計算機既知モデル）．

３．各計算機が基本情報と計算機網の計算機総数を保持している場合（サイズ既知モデル）．

分散アルゴリズムの効率の評価は通信複雑度と理想時間複雑度により行われる[3]．通信複雑度は，計算機網全体で分散アルゴリズムの実行中に交換されるメッセージ総数である．理想時間複雑度は，各計算機の処理時間を無視し，メッセージの伝播時間が単位時間であると考えた場合の分散アルゴリズムの実行時間である．ＣＬＦを解く分散アルゴリズムの通信複雑度，理想時間複雑度をそれぞれC，Tと表す．本稿では，基本情報モデルにおいては，ＣＬＦを解く分散アルゴリズムが存在しないことを示す．また，計算機網の計算機数，正常リンク（故障していないリンク）数，故障リンク数をそれぞれn，e_f，fと表すとき，隣接計算機既知モデルに対しては，$C=O(e_f+\min(fn, n^2))$，$T=O(n)$である分散アルゴリズムを示し，$C=\Omega(\max(n,\min(fn\frac{\log f}{\log n}, n^2)))$かつ$T=\Omega(n)$であることを示す．サイズ既知モデルに対しては，$C=O(e_f+n\log f)$，$T=O(n)$である分散アルゴリズムを示し，$C=\Omega(e_f)$かつ$T=\Omega(n)$であることを示す．更に，故障をより一般化し，計算機の故障も許す場合には，各計算機が如何なる情報を保持していても連結性判定問題を解く分散アルゴリズムは存在しないことを示す．

2．諸定義[3]

計算機網$N=(P, L)$は，計算機の(有限で空でない)集合Pと，リンクと呼ばれるPの相異なる要素の非順序対の集合によって定義され，計算機 $u, v(\in P)$に対して，$(u,v)\in L$であるとき，計算機u，v間に全２重通信路が存在し，uからvへとvからuへは互いに独立にメッセージを送ることができるものとする．このとき，uはvの(vはuの)隣接計算機という．また，計算機網Nの計算機総数$|P|$をNのサイズという．計算機網はグラフとみなせるので，以下ではグラフに関する用語や記法を計算機網に関しても用いるものとする．

各計算機uには，uが識別できる自然数の番号のついたポートが存在し，各ポートは送信用と受信用の２つの待行列(queue)から構成されている．uに接続する$\deg(u)$本のリンクは，１番から$\deg(u)$番までの$\deg(u)$個のポートにそれぞれ１本ずつ接続されているものとする．p番のポートをポートpと呼ぶ．計算機網Nにおいて，uのポートpにリンク(u,v)が接続しているとき，この接続関係を$\rho(u,p)=(u,v)$と表す．

計算機網に関して以下のことを仮定する．

① 計算機網は計算機やリンクに故障が発生しない限り連結である．

② 計算機網には共有メモリは存在せず，計算機間の通信はリンクを介するメッセージの送受信のみで行われる．

③ ポートを構成する待行列はサイズに制限のないFIFO(First-In First-Out)待行列とする．

④ 任意の計算機uとその任意の隣接計算機vにおいて，それぞれのあるポートp，qに対して，$\rho(u,p)=\rho(v,q)$であるとき，uのポートpの送信用待行列の先頭にあるメッセージは，リンク(u,v)のuからvへの通信路を伝って有限時間内にvのポートqの受信用待行列の最後尾に入る．但し，この時間の上限は仮定できない．

⑤ 計算機網を構成する各計算機の処理速度は一般に異なる．

④の後半及び⑤は計算機網が非同期であることを意味する．

分散アルゴリズム（以下，アルゴリズムと呼ぶ）は，各計算機が実行するプログラムによって表現される．各計算機は同一のプログラムを持ち実行するが，各計算機がプログラム実行前に保持している計算機網に関する情報（計算機網情報と呼ぶ）はそれぞれ異なる．

各計算機uが保持している計算機網情報として次のような情報を考える．

- uの識別子 id(u)
- uの次数 deg(u)
- uの隣接計算機の識別子 adid(p) $(1\leq p\leq \deg(u))$
  但し，$\rho(u,p)=(u,v)$のとき，adid(p) = id(v)
- 計算機網のサイズ size

このうち，id(u)とdeg(u)を基本情報という．

計算機網の各計算機uのもつ計算機網情報によって，計算機網を次のように分類する．

(1)基本情報モデル　　　：基本情報のみ
(2)隣接計算機既知モデル：基本情報とadidのみ
(3)サイズ既知モデル　　：基本情報とsizeのみ

自発的に実行を開始する計算機を始動計算機と呼ぶ．始動計算機以外の計算機は隣接計算機からメッセージを受信することによって実行を開始する．

メッセージに関して次の仮定をおく．

⑥ １つのメッセージによって送信できる情報のデータ量は，$O(\log n)$ビットとする．ここでnは計算機網のサイズを表す．

故障に関して，以下のことを仮定する．

Distributed Algorithms for the Connectivity of Networks with Faulty Elements
Koichi WADA[1], Yukio MORITANI[1], Kimio KAWAGUCHI-IZAWA[1], Masahiro MORISHITA[2]
1: Nagoya Institute of Technology 2: Tateishi Electoric Corp.

⑦ 計算機の故障は発生しない.
⑧ リンクの故障が発生すると，そのリンクに接続している計算機はその故障を自動的に検知することができる.
⑨ リンクが故障していると，そのリンクを用いた通信は両方向ともできなくなる.
⑩ アルゴリズム実行中の計算機に接続しているリンクが新たに故障することはない.
⑪ アルゴリズム実行中に故障リンクが復旧することはない.

故障リンクの集合をＦで表す．また，リンクに故障の発生している計算機網Nに対して，Ｎからすべての故障リンクを除去した計算機網をＮの正常計算機網と呼び，Ｎ－Ｆで表す.

計算機網Nにおいて，リンクに故障が発生したとき，Ｎの正常計算機網が連結であるかどうかを判定する問題をリンク故障時の連結性判定問題（ＣＬＦ）という.

３．ＣＬＦを解く分散アルゴリズム

(1) 基本情報モデル

［定理１］ 基本情報モデルの計算機網において，ＣＬＦを解くアルゴリズムは存在しない. ▌

(2) 隣接計算機既知モデル

アルゴリズムＣＦＬＮの概略

（第一段階） 故障を検知した計算機が始動計算機として，文献(1)の方法で，正常計算機網Ｎ－Ｆの各連結成分Ｃごとに，根つき生成木T_Cを構成する.

（第二段階） Ｃ内で故障を検知した各計算機は，T_Cを介して自分の識別子を送り，根である計算機にそれらを集める．根である計算機はそれを故障を検知したすべての計算機に放送する．それを受け取った各計算機は，次のような条件で連結であるか（《連結》）または非連結であるか（《非連結》）をT_Cにおける親に送る.

・自分に接続する故障リンクの反対側の計算機の識別子がすべて存在し，自分のすべての子から《連結》を受け取るならば《連結》を送る.
・その他の場合は《非連結》を送る.

最終的に，根である計算機が《連結》を得たならば《連結》をすべての計算機に放送し，《非連結》を得たならば《非連結》を放送する.

［補題１］ 計算機網において，リンクの故障が起きたとき，正常計算機網が連結である必要十分条件は，故障を検知した計算機と故障リンクの反対側の計算機がすべて同じ連結成分内に存在することである. ▌

［定理２］ 隣接計算機既知モデルの計算機網において，アルゴリズムCFLNはＣＬＦを正しく解き，$C=O(e_f+\min(fn, n^2))$，$T=O(n)$である. ▌

［定理３］ 隣接計算機既知モデルの計算機網において，ＣＬＦを解く任意のアルゴリズムは$C=\Omega(\max(n, \min(fn\frac{\log f}{\log n}, n^2)))$かつ$T=\Omega(n)$である.

（証明） $T=\Omega(n)$であることは明らかである.

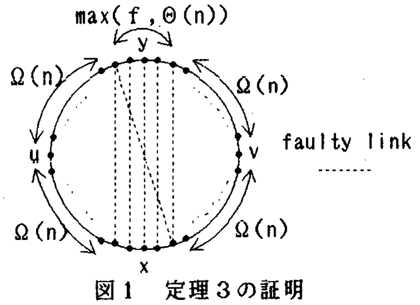

図１ 定理３の証明

通信複雑度の下界は，図１を用いて示される.

［補題２］ ある計算機ｘ，ｙ（$(x,y)\in F$）が存在して，それらの計算機を識別するのに必要な情報（必ずしも識別子であるとは限らない）がいずれも，ｕとｖのどちらの計算機にも受信されないならば，どの計算機も連結性を判定できない. ▌

補題２より，故障を検知した任意の計算機ｘ，ｙ（$(x,y)\in F$）に対して，それらの計算機を他と区別するのに必要な情報のいずれかが，ｕまたはｖのどちらかに受信されなければならない．このとき，故障を検知した計算機数が，$f=O(n)$ならば２ｆ個，$f=\Omega(n)$ならば$\Theta(n)$個となるような計算機網が構成できる．従って，この情報の１つを表すには$\Omega(\min(\log f, \log n))$ビットが必要となり，このような情報が$\Omega(\min(f, n))$個必要なので，全体で$\Omega(\min(f\log f, n\log n))$ビットを$\Omega(n)$以上送る必要がある．ここで，メッセージ長は$O(\log n)$ビットと仮定しているので，メッセージ数は，$\Omega(\min(fn\frac{\log f}{\log n}, n^2))$となる.

また，明らかに$\Omega(n)$メッセージを必要とする計算機網及び故障が存在するので，上の議論と合わせると$C=\Omega(\max(n, \min(fn\frac{\log f}{\log n}, n^2)))$となる. ▌

(3) サイズ既知モデル

アルゴリズムＣＦＬＳの概略

（第一段階） ＣＦＬＮの第一段階と同様.

（第二段階） T_Cの葉である計算機から根となる計算機へと順に親の計算機に自分の子孫の数を送ることによって，Ｃ内の計算機数n_Cを根に持たせて，根がそれと計算機総数ｎを比較することにより連結性を判定し，結果をT_Cを用いて放送する.

$n_C < n$ ⇒ 非連結
$n_C = n$ ⇒ 連結

［補題３］ ｎ個の計算機を持つ計算機網において，リンクの故障が起きたとき，正常計算機網が連結であるための必要十分条件は，正常計算機網がｎ個の計算機を持つ連結成分を持つことである. ▌

［定理４］ サイズ既知モデルの計算機網において，アルゴリズムCFLSはＣＬＦを正しく解き，$C=O(e_f+n\log f)$，$T=O(n)$である. ▌

［定理５］ サイズ既知モデルの計算機網において，ＣＬＦを解く任意のアルゴリズムは$C=\Omega(e_f)$かつ$T=\Omega(n)$である.

（略証） $T=\Omega(n)$であることは明らかである．$C=\Omega(e_f)$は，文献(2)の手法により得られる. ▌

４．故障の一般化

故障に関する仮定⑦を緩め，計算機の故障も許した場合には以下の定理が成り立つ.

［定理６］ 各計算機が如何なる計算機網情報を保持していても計算機が故障する場合には，連結性判定問題を解くアルゴリズムは存在しない. ▌

参考文献

(1) B. Awerbuch :"Optimal distributed algorithms for minimum weight spanning tree, counting, leader election and related problems", Proc. 19th Symp. Th. Comp., 230-240 (1987).
(2) 朴,増澤,萩原,都倉 :"ネットワークの連結性関連問題を解く効率のよい分散アルゴリズム",信学論 J72-D-I, 5, 343-356 (1989-5).
(3) 萩原,都倉,増澤 :"分散アルゴリズムの複雑度について",セル構造に基づく高度並列情報処理システムに関する総合的研究, 104-113 (1988-03).