计算随机流量网络可靠度的算法综述*)

孙伟平　周敬利　余胜生
(华中科技大学计算机学院　武汉430074)

摘　要　随机流量网络比二态网络更适合用于描述现实生活中的许多系统。给定要求d,随机流量网络的可靠度定义为最大流不小于d的概率。这一领域的研究提供了许多算法来估计系统的可靠度。本文介绍了这些算法(特别是基于最小路径和最小割集的算法)的来源及思想。文章最后给出了将来研究工作的方向。

关键词　随机流量网络,可靠度,MPs,MCs

Survey of Algorithms on Reliability Evaluation of Stochastic-Flow Networks

SUN Wei-Ping　ZHOU Jing-Li　YU Sheng-Sheng
(Departmnt of Computer Science, Huazhong University of Science and Technology, Wuhan 430074)

Abstract　A stochastic-flow network in which each arc has several capacities is more suitable to describe real world systems than a binary-state network. Given the demand d, the system reliability is defined as the maximum flow of the network is not less than d. The researchers in the field have presented many algorithms on the reliability evaluation of stochastic-flow networks. In this paper, the authors introduce the properties of each algorithm, mainly those algorithms in terms of MPs and MCs. Finally the future directions of the work are presented.

Keywords　Stochastic-flow network, Reliability, MPs, MCs

1　引言

网络的可靠度是网络性能的一个重要指标,可靠性分析一直是各研究者研究的热点问题[1~14]。根据网络或其组成部分的状态的多少,网络可分为二态(binary state)网络和多态(multi-state)网络。

·二态(binary state)网络是指网络或网络的每个元素(节点或弧)只有两种状态:好/坏。网络的每条弧的容量的取值或为0,或为某一正整数。

·有的网络的组成部分具有多种状态或容量。在某些情况下,除了考虑网络状态的好与坏,还要研究不同的性能指标。这种网络我们称为多态网络。在本文中,又称为随机流量网络。

许多文献研究了二态系统,其中Lee[4]利用词典排序法计算系统的可靠度;Abraham[5],Aggarwal等[6]则通过MCs或MPs等方法来研究系统的可靠度问题。许多现实中存在的系统,比如计算机网络,电信网络,电力传输与分布网络,运输网络等等,不能用二态系统来描述。近些年来,对有着广泛应用的随机流量网络的可靠度的研究越来越多[8~14]。其中,文[10~12]采用的是最小路径的方法,文[13,14]采用的是最小割的方法。在本文第3节中我们将作详细介绍。

2　随机流量网络可靠度问题建模

设$G=(N,A,M)$是从源s到汇t的一个随机流量网络,其中N为节点集,$A=\{a_i|1\leqslant i\leqslant n\}$为弧集,$M=(M^1,M^2,\cdots,M^n)$,$M^i$为弧$a_i$的最大容量,$i=1,2,\cdots,n$。假设$G$还满足:

1. 每个节点均是可靠的。
2. 每条弧a_i的容量x_i为整数,$0\leqslant x_i\leqslant M^i$。
3. 不同弧的容量概率无关。
4. 网络G满足flow-conservation law。

记$X=(x_1,x_2,\cdots,x_n)$为系统的(当前)容量向量,其中x_i为弧a_i的(当前)容量。记$V(X)$为容量向量为X时的最大流。给定d,系统的可靠度R_d定义为最大流不小于d的概率,即$R_d=\Pr\{X|V(X)\geqslant d\}$。

称向量$X\geqslant Y$,如果$x_i\geqslant y_i$,$\forall\ 1\leqslant i\leqslant n+p$;称向量$X>Y$,如果$X\geqslant Y$,而且至少存在一个$i\in\{1,2,\cdots,n+p\}$,s.t. $x_i>y_i$。

3　解决随机流量网络可靠度问题的算法综述

3.1　Xue的算法

最小上界与最大下界是研究系统的可靠度时常常和到的两个概念,分别对应于图论中的最小路径(Minimal Paths)和最小割(Minimal Cuts)两个概念。文[8]提出了最小上界与最大下界的概念,但是他没有讨论怎样得到最小上界与最大下界。文[9]在离散函数理论中提出了一些算法来获得最小上界与最大下界。Xue[10]发现这些算法也可用来研究多态系统的水平为d的最小上界与最大下界。

为与其它算法进行比较,我们重新表述Xue的算法如下:将系统$G=(N,A)$进行模块分解,例如分解成MPs:$K_1,K_2,\cdots,K_m$,其结构函数取

$$V(X)=D(\mathrm{sum}(\cdot),(f_1,f_2,\cdots,f_m))=\sum_{j=1}^{m}f_j$$

其中$D(\cdot)$表示系统的分解,sum$(\cdot)$为通过MPs的所有流,$(f_1,f_2,\cdots,f_m)$为X下的可行流。记

$$\mathrm{MI}_{K_j}(i)=\{(q_{j1},q_{j2},\cdots,q_{jn})\};$$

$$\mathrm{MI}_{\mathrm{sum}}(i)=\{(f_1,f_2,\cdots,f_m)\mid\sum_{j=1}^{m}f_j=i,f_j\in\{0,1,2,\cdots,L_j\},j=1,2,\cdots,m\};$$

$$\mathrm{PMI}(i)=\{(\sum_{j=1}^{m}q_{j1},\sum_{j=1}^{m}q_{j2},\cdots,\sum_{j=1}^{m}q_{jn})\mid(q_{j1},q_{j2},\cdots,q_{jn})\in$$

*)本项目受国防预研基金资助。孙伟平　博士后,主要研究方向为存储网络及网络性能分析等。

$\mathrm{MI}_{K_j}(f_j), j=1,2,\cdots,m\}$

其中 $L_j=\min\{M^i \mid a_i\in K_j\}, q_{jk}=\begin{cases} i & a_k\in K_j; \\ 0 & a_k\notin k_j. \end{cases}$ $j=1,2,\cdots,m, i=0,1,2,\cdots,L_j$

其求系统的水平为 d 的最小上界的算法为：

Step 1. 对系统进行模块分解，列出 $\mathrm{MI}_{K_j}(i)$；

Step 2. 求 $\mathrm{MI}_{sum}(d)$；

Step 3. 求 $\mathrm{PMI}(d)$；

Step 4. 从 $\mathrm{PMI}(d)$ 中去掉不是最小上界的向量。

对那些不能直接分解成模块的系统，需要额外的工作扩展系统，使其能够分解成模块，然后再使用上面的算法。

3.2 最小路径法

系统可靠度

$R_d=\Pr\{X\mid V(X)\geqslant d\}$

$=\Pr\{X\mid X\geqslant X', X'$ 为水平为 d 的最小上界$\}$

故只要找到系统的所有水平为 d 的最小上界就可以计算出系统的可靠度了。文[11,12]等基于最小路径法设计算法来寻找系统的所有水平为 d 的最小上界。他们的主要做法如下：

设系统有 m 个 MPs：$K_1,K_2,\cdots,K_m$，记流向量 $F=(f_1,f_2,\cdots,f_m)$，其中 f_j 为 MP K_j 上的当前流，MP K_j 的最大容量 $L_j=\min\{M^i \mid a_i\in K_j\}$。$F$ 为 $M(X)$ 下的可行流，如果

$$\sum_{j=1}^{m}\{f_j \mid a_i\in K_j\}\leqslant M^i(x_i)$$

定义 $V(X)=\max\{\sum_{j=1}^{m}f_j \mid F$ 为 X 下的可行流$\}$。

给定 d，称系统状态 X 为一个 d-MP，如果下列两个条件成立：1. $V(X)\geqslant d$；2. $\forall Y<X, V(Y)<d$。

记 $\mathbf{F}=\{F=(f_1,f_2,\cdots,f_m) \mid \sum_{j=1}^{m}\{f_i \mid a_i\in K_j\}\leqslant x_i$，且 $\sum_{j=1}^{m}f_j=d\}$。文[12]证明了：

定理1 如果系统状态 X 为一个 d-MP，则 $\exists F\in\mathbf{F}$，使得

$$x_i=\sum_{j=1}^{m}\{f_j \mid a_i\in K_j\}, i=1,2,\cdots,n \quad (1)$$

$\forall F\in\mathbf{F}$，记由(1)式生成的 X 为 X_F，记 $\Omega_d=\{X_F \mid F\in\mathbf{F}\}$，$\Omega_{d,\min}=\{X\in\Omega_d \mid \forall Y\in\Omega_d, Y\neq X, Y$ 不小于 $X\}$。则有：

定理2 $\Omega_{d,\min}=\{\text{d-MPs}\}$

由此得到算法找到系统的所有水平为 d 的最小上界：

Step 1. 确定系统所有的 MPs $=\{K_1,K_2,\cdots,K_m\}$。再由下列条件求 M 下的所有可行流 F：

$$\sum_{j=1}^{m}\{f_j \mid a_i\in K_j\}\leqslant M^i \quad (2)$$

$$\sum_{j=1}^{m}f_j=d \quad (3)$$

Step 2. 由上一步得到的所有 F 按(1)生成 Ω_d。

Step 3. 设由上一步得到的 $\Omega_d=\{X^1,X^2,\cdots,X^q\}$，由比较法 $\Omega_{d,\min}$：

3.1) $I=\emptyset$

3.2) FOR $i=1$ TO q and $i\notin I$

3.3) FOR $j=i+1$ TO q and $j\notin$ I

3.4) IF $X^j<X^i$ THEN X^i is not a d-MP. $I=I\cup\{i\}$ and goto step 3.7).

ELSEIF $X^j\geqslant X^i$ THEN X^j is not a d-MP. $I=I\cup\{j\}$.

3.5) $j=j+1$

3.6) X^i is a d-MP.

3.7) $i=i+1$

3.8) END.

文[12]有个小错误，其步骤

3.4) IF $X^j<X^i$ THEN X^i is not a d-MP. $I=I\cup\{i\}$ and goto step 3.7).

ELSEIF $X^j\geqslant X^i$ THEN X^j is not a d-MP. $I=I\cup\{j\}$ and goto step 3.7).

显然，第二个"goto step 3.7)"是错误的。

对 serial parallel（即 $K_i\cap K_j=\emptyset, \forall i\neq j$）的情形，Xue[10] 与文[11,12]的算法是一样的；但对 non-serial parallel 的情形，文[10]需先扩展系统，然后进行系统分解，再利用算法找到系统的所有的水平为 d 的最小上界，而文[11,12]的算法对 non-serial parallel 的情形同样适用，不需要任何额外的工作。

3.3 最小割法

还可以看到，系统可靠度

$R_d=\Pr(X\mid V(X)\geqslant d\}=1-\Pr\{X\mid V(X)\leqslant d-1\}$

$=1-\Pr\{X\leqslant X' \mid X'$ 为水平为 $d-1$ 的最大下界$\}$

故只要找到系统的所有水平为 d 的最大下界也可以计算出系统的可靠度。文[13,14]等用最大流最小割定理设计算法来寻找系统的所有水平为 d 的最大下界。

设系统有 m 个 MCs：$K_1,K_2,\cdots,K_m$。记 e_i 为 n 维向量，其第 i 个分量为1，其余分量均为0。不饱和弧集 $U_X=\{a_i \mid x_i<M^i\}$，敏感弧集 $S_X=\{a_i \mid a_i\in U_X, V(X+e_i)>V(X)\}$。MC K_j 的容量 $C_{K_j}(X)=\sum_{a_i\in K_j}x_i$，$\omega_d=\{X \mid V(X)=d\}$，$\omega_{d,\max}=\{X\in\omega_d \mid \forall Y\in\omega_d, Y$ 不大于 $X\}$。

相应地，给定 d，称系统状态 X 为一个 d-MC，如果

1. $V(X)=d$；2. $U_X=S_X$。

文[13]用反证法证明了：

定理3 如果系统状态 X 为一个 d-MC，则至少存在一个 MC $K_r=\{a_{r1},a_{r2},\cdots,a_{rn_r}\}$，使得

$$x_{r1}+x_{r2}+\cdots+x_{rn_r}=d \quad (4)$$

$$(x_{r1},x_{r2},\cdots,x_{rn_r})\leqslant(M^{r1},M^{r2},\cdots,M^{rn_r}) \quad (5)$$

$$\forall a_i\notin K_r, x_i=M^i \quad (6)$$

使(4)(5)(6)式均成立的所有 X 的集合记为 ρ_d，显然，$\omega_{d,\max}\subseteq\rho_d$；反之，若 $X\in\rho_d$ 且

$$V(X)=d, U_X\subseteq\cap\{K_i \mid C_{K_i}(X)=d\} \quad (7)$$

则 X 为一个 d-MC。基于定理3，文[13]设计如下算法寻找系统所有的水平为 d 的最大下界：

Step 1. 确定系统所有的 MCs $=\{K_1,K_2,\cdots,K_m\}$。

Step 2. 由(4)(5)(6)式求 ρ_d。

Step 3. 检查条件(7)是否满足：

3.1) IF $\exists i\neq r$, s.t. $C_{K_i}(X)<d$, THEN X is not a d-MC and goto step 3.4)

3.2) $I=\{i \mid C_{K_i}(X)=d\}$

3.3) IF $\exists a_i\in A\backslash\cap_{i\in I}K_i$, s.t. $x_i\neq M^i$, THEN X is not a d-MC, ELSE X is a d-MC.

3.4) NEXT $X\in\rho_d$

注意到条件 $U_X\subseteq\cap\{K_i \mid C_{K_i}(X)=d\}$ 的验算比较复杂，特别是当系统较大时。若记

$\rho_{d,\max}=\{X\in\rho_d \mid \forall Y\in\rho_d, Y$ 不大于 $X\}$

文[14]更证明了：

定理4 $\rho_{d,\max}=\omega_{d,\max}$

这样，就可以用简单的比较法来代替文[13]算法中的验算：

Step 3′ 设由 Step 2得到的 $\rho_d=\{X^1,X^2,\cdots,X^u\}$：

3.1′) $I=\emptyset$

3.2′) FOR $i=1$ TO u and $i\notin I$

3.3′) FOR $j=i+1$ TO u and $j\notin I$

3.4′) IF $X^i<X^j$ THEN X^i is not a d-MC. $I=I\cup\{i\}$ and goto step 3.7′).

ELSEIF $X^i\geqslant X^j$ THEN X^j is not a d-MC. $I=I\cup\{j\}$.

3.5′) $j=j+1$

3.6′) X^i is a d-MC.

3.7′) $i=i+1$

3.8′) END.

可以看到，MPs，MCs 是解决这类问题的简单而有效的方法。但是也有其局限性，这些方法的前提是系统的所有 MPs，MCs 已知，而求解 MPs，MCs 有时候并不是件容易的

事。

3.4 其它算法

除了上面提到的算法，还有一些论文也讨论了随机流网络的可靠度问题，例如：Doulliez，Jamoulle[1]采用Ford-Falkerson流分析方法，将系统的状态空间分解为三种集合：accepted状态，unaccepted状态，unspecified状态的不相交集合。每个unspecified状态用递归法继续分解，直至再也没有unspecified状态集了。accepted状态的不相交集合的概率可以直接计算出来，它们的和就是系统的水平为d的可靠度。Fishman[2]用Monte Carlo抽样法从有限个样本中分离出点估计和区间估计。Clancy等[3]则用逼近法估计系统的可靠度。先用Doulliez，Jamoulle的分解方法逐层分解系统的状态空间直到某一指定的水平，然后采用Monte Carlo拟合，用unspecified状态的不相交集合的分布来近似系统的可靠度。

结论 本文在随机流网络的可靠度问题的数学模型的基础上，着重介绍了在这个研究方向上的主要研究成果。可以看到，MPs，MCs是解决这类问题的简单而有效的方法，但是也有其局限性，如上面提到的，这些方法的前提是系统的所有MPs，MCs已知。而对二态网络来说，求解MPs，MCs是NP-难的。另外，本文提到的算法都是针对弧容量有限定的网络，现实世界的系统的节点也是有容量限定的，甚至会失效的。目前对这类问题的研究结果还非常少。文[7]提出的解决办法是基于概念"The failure of a node implies the failure of links incident from it"的，只适用于二态系统。因此可对这类问题进行深入的研究。

参考文献

1 Doulliez P, Jamoulle J. Transportation Networks with Random Arc Capacities. RAIRO, Recherche Operationnelle, (Operations Research), 1972, 3: 45～60

2 Fishman G S. Monte Carlo Estimation of the Maximum Flow Distribution with Discrete Stochastic Arc Capacity Levels. Naval Research Logistrics Quarterly, 1989, 36: 829～849

3 Clancy D P, Gross G, Wu F F. Probabilistic Flows for Reliability Evaluation of Multiarea Power System Interconnections. Electrical Power and Energy Systems, 1983, 5: 100～114

4 Lee S H. Performance Indexes of a Telecommunication Network. IEEE Trans. Reliability, 1980, R-29: 24～26

5 Abraham J A. An Improved Algorithm for Network Reliability. IEEE Trans. Reliability, 1979, 28: 58～61

6 Aggarwal K K, Chopra Y C, Bajwa J S. Capacity Consideration in Reliability Analysis of Communication System. IEEE Trans. Reliability, 1982, 31: 177～180

7 Aggarwal K K, Gapta J S, Misbra K B. A Simple Method for Reliability Evaluation of a Communication System. IEEE Trans. Communications, 1975, COM-23: 563～566

8 Block H W, Savits T H. A Decomposition for Multistate Momotone Systems. J. Appl. Prob., 1982, 19: 391～402

9 Davio M, Deschamps J D, Thaysc A. Discrete and Swithing Functions. Georgi Publishing Company, McGraw-Hill International Book Company, 1978

10 Xue J. On Multistate System Analysis. IEEE Trans. Reliability, 1985, R-34: 329～337

11 Lin J S, Jane C C, Yuan J. On Reliability Evaluation of a Capacitated-Flow Network in Terms of Minimal Pathsets. Network, 1995, 25: 131～138

12 Lin Y K. A Simple Algorithm for Reliability Evaluation of a Stochastic-Flow Network with Node Failure. Computers & Operations Research, 2001, 28: 1277～1285

13 Jane C C, Lin J S, Yuan J. Reliability Evaluation of a Limited-Flow Network in Terms of Minimal Cutsets. IEEE Trans. Reliability, 1993, 42(3): 354～361

14 Lin Y K. On Reliability Evaluation of a Stochastic-Flow Network in Terms of Minimal Cuts. J. Chinese Institute of Industrial Engineers, 2001, 18(3): 49～54

(上接第15页)

7 Li Z. Theories and algorithm for verification of bisimulation equivalence in value-passing CCS and the π-calculus: [PhD Thesis]. Changsha Institute of Technology, Changsha, 1999

8 Lin H. Complete inference systems for weak bisimulation equivalences in the π-calculus. In: P. D. Mosses, M. Nielsen, M. I. Schwarzbach, eds. Proc. of TAPSOFT'95, volume 915 of LNCS, Springer, 1995. 187～201

9 Milner R. Communication and Concurrency. Prentice Hall, 1989

10 Milner R. Functions as process. Journal of Mathem. Structures in Computer Science, 1992, 2(2): 119～141

11 Milner R, Parrow J, Walker D. A Calculus of Mobile Process, part Ⅰ/Ⅱ. Journal of Information and Computation, 1992, 100: 1～77

12 Milner R, Sangiorgi D. Barbed bisimulation. In: W. Kuich, ed. Proc. of ICALP'92, volume 623 of LNCS, Springer, 1992. 685～695

13 Park D. Concurrency and automata on infinite sequence. volume 154 of LNCS, Springer-Verlag, 1987. 561～572

14 Parrow J. An introduction to the π-calculus. Chapter to appear in Handbook of Process Algebra, Elesvier

15 Parrow J, Sangiorgi D. Algebraic theories for name-passing calculi. Journal of Information and Computation, 1995, 120(2): 174～197

16 Reisig W. Petri nets: an introduction. Springer-Verlag, 1985

17 Sangiorgi D. Expressing Mobility in Process Algebra: First-Order and Higher-Order Paradigms: [PhD Thesis]. LFCS, University of Edinburg, 1993

18 Sangiorgi D. A theory of bisimulation for the π-calculus. Acta Informatica, 1996, 33: 69～97

19 Sangiorgi D, Walker D. On barbed equivalence in π-calculus. CONCUR'01, 2001

20 Sangiorgi D, Walker D. The pi -calculus: a theory of mobile processes. Cambridge University Press, 2001

21 Winskel G. Event Structure. Volume 255 of LNCS. Springer-Verlag, 1987. 325～392

22 Cardelli L, Gordon A. Mobile ambients. Theoretical Computer Science, 2000, 240: 177～213

23 Levi F, Sangiorgi D. Controlling interference in ambients. In: Proc. POPL'00, Boston, Massachusetts, 2000. 352～364

24 Guan X, Yang Y, You J. Making ambients more robust. In: Proc. Intl Conf. on Software: Theory and Practice, Beijing, China, 2000. 377～384

25 Fu Y. An open problem of mobile processes. Journal of Computer (in Chinese), 2001, 24(7)

26 Honda K, Toloro M. On asynchronous communication semantics. object-based concurrent computing, volume 612 of LNCS, Springer-Verlag, 1991. 21～51

27 Merro M, Sangiorgi D. on asynchrony in name-passing calculi, ICALP'98, volume 1443 of LNCS, Springer-Verlag, 1998. 856～867