

多状态不变化路径可靠性分析的符号算法

李凤英, 何志伟, 董荣胜

(桂林电子科技大学广西可信软件重点实验室 广西 桂林 541004)

【摘要】传统算法计算两条不变化路径传输的随机流网络可靠性, 是通过获取系统最小容量向量的方法, 需要存储整个网络的边以及移除冗余向量, 运算非常复杂。因此提出基于MDD的多状态两条不变化路径可靠性分析算法MDD_2SMPs, 利用MDD能够双向反映组件状态与系统状态关系的特点, 通过定义MDD操作算子, 在无需对路径进行流量分配的情况下获取路径容量, 并在组合过程中引入约束剪枝策略对无效容量过滤, 提高算法效率。针对路径失效问题, 提出基于MDD的备用路径选择算法MDD_BMPs, 通过将各路径转换为决策图多值变量形式, 降低了计算备用路径可靠性的复杂性。实例结果表明, 算法MDD_2SMPs比传统算法减少了计算可靠性的运算量, 并能精确选择网络备用路径。

关键词 备用路径; MDD; 网络可靠性; 不变化路径

中图分类号 TN927 **文献标志码** A **doi:**10.3969/j.issn.1001-0548.2018.06.004

Reliability Analysis Algorithm for Multi-State Two Separate Minimal Paths Based on Multi-Valued Decision Diagram

LI Feng-ying, HE Zhi-wei, and DONG Rong-sheng

(Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology Guilin Guangxi 541004)

Abstract Traditional methods apply system minimum capacity vectors to calculate the reliability of stochastic-flow network with two separate minimal paths (2SMPs), which needs to store the arc of entire network and the process of removing redundant vectors is complex. In view of this, based on multi-valued decision diagram (MDD) for 2SMPs, we propose a reliability analysis algorithm, called MDD_2SMPs. The main idea of this algorithm is using MDD to reflect the relationship between components status and system status, by defining MDD operators, the path capacity can be obtained without the need for flow distribution. Besides, the constraint pruning strategy is introduced in the process of combination to filter many unnecessary combinations. Furthermore, aiming at the problem of path failure, all paths are converted into MDD variables by using the proposed algorithm MDD_BMPs, resulting a great reduction of the computational complexity. Example analysis shows that the proposed algorithm based on MDD has less calculation burden than traditional methods and can select the network backup paths accurately.

Key words backup paths; MDD; network reliability; separate minimal paths

随着网络技术的不断发展, 网络已融入生活的各个方面, 如运输网络、移动自组网络、计算机和通信网络等^[1-3]。对于上述网络系统, 常借助于随机流网络(stochastic-flow network, SFN)进行刻画描述, 有关随机流网络的理论和应用研究越来越受到人们关注。其中对于随机流网络的可靠性, 既可通过蒙特卡洛方法^[4]计算网络可靠性的近似值, 也可通过空间分解^[5]、多值决策图^[6]、最小割集^[7]或最小路集^[8]等方法计算网络可靠性的精确值。在最小路集基础上, 文献[8]对单条路径传输的网络可靠性进行了分

析。为了缩短流量传输时间, 文献[9]将随机流网络流量从单路径传输扩展到两条及多条不变化路径(separate minimalpaths, SMPs)传输的情况, 并提出选择一组可靠性最高的路径作为网络备用路径, 来保障工作路径失效后网络的顺利传输。在2SMPs传输流量前提下, 文献[10-12]加入预算约束条件, 基于最小路集的方法求解了随机流网络可靠性。文献[13-14]定义了错误率和时间约束的多状态网络可靠性, 并考虑到工作路径失效后启用备用路径来保障网络传输, 提出基于路集的方法对状态空间过滤获

收稿日期: 2017-06-29; 修回日期: 2018-02-26

基金项目: 国家自然科学基金(61762024); 广西省自然科学基金(2017GXNSFDA198050, 2016GXNSFAA380054)

作者简介: 李凤英(1974-), 女, 博士, 副教授, 主要从事网络可靠性、符号计算、形式化方法等方面的研究。

取系统最小容量向量,运用容斥原理求解随机流网络可靠性。文献[15]则认为路径还有可能一条一条地失效,通过枚举单路径失效情况,证实了单条备用路径可以提高网络传输可靠性。

对于一个包含2SMPs的复杂网络系统,一方面获取系统最小容量向量计算网络可靠性需要存储整个网络的边,并且包含了许多冗余路径容量组合,寻找一种更加有效的方式对于该类问题求解有着实际的意义;另一方面,通过枚举网络各路径最小容量向量的方式计算备用路径可靠性,运算量非常庞大,故需要一种简洁的解决方法选择网络可靠的备用路径。为此,本文首先针对算法 $(d, T, B, (P_i, P_j))$ -LBPs^[15]计算2SMPs可靠性需要存储整个网络的边以及存在冗余路径容量的问题,提出基于多值决策图(multi-valued decision diagram, MDD)的可靠性分析算法MDD_2SMPs。该算法利用MDD能够双向反映组件状态与系统状态关系的特点,通过自定义MDD操作算子,实现状态空间、变量组合的隐式表示和搜索,减少路径不相关边的存储;并在组合过程中引入约束剪枝策略过滤许多不必要的路径,从而提高算法效率;同时给出了基于MDD的概率计算模型,使得可靠性计算更加直观。其次,针对工作路径失效后选择网络可靠备用路径的问题,提出算法MDD_BMPs,该算法通过将网络各路径转换为决策图多值变量形式,降低了计算的复杂性。最后,通过基准网络实例,验证了本文所提方法的正确性和有效性。

1 模型描述与假设

本文中,网络边上存在3种属性:容量、时延和成本。容量表示单位时间内边上通过的最大数据量;时延表示数据在边上传输花费的固定时间;成本表示数据在边上传输需要的费用。综上,SFN模型可以形式化为 $G=(N, E, M, L, B)$,其中:

1) N 表示网络节点的集合,用节点 s 表示源点, t 表示目标节点;

2) $E=\{e_i|1\leq i\leq n\}$ 表示网络边的集合;

3) $M=(M_1, M_2, \dots, M_n)$,表示边的状态集, $M_i=\{0, 1, \dots, m_i\}$ 表示边 e_i 所有互斥的状态集,0表示完全失效, m_i 表示完全工作;

4) $L=(l_1, l_2, \dots, l_n)$, l_i 表示边 e_i 的传输时延;

5) $B=(b_1, b_2, \dots, b_n)$, b_i 表示边 e_i 的传输成本。

本文分析过程基于如下假设^[11, 15]:

1) 节点完全可靠;2) 每条边上的容量状态相互独立,并以一定概率随机分布;3) 数据通过2SMPs传输;4) 网络 G 遵循流量守恒定律。

2 方法与问题定义

2.1 多值决策图(MDD)

MDD是一种基于香农分解用于表示和操作多值逻辑函数的有向无环图^[16]。文献[17]的结论表明,在多态网络的可靠性分析中,MDD比OBDD具有较少的变量和较高的效率。在MDD中,用圆圈表示非终节点,代表结构函数的一个变量;方框表示终节点,代表多状态系统的状态,系统有多少个可能的状态,对应的MDD就存在多少个终节点;单向箭线表示非终节点的外向分支,每个非终节点有多少种状态就存在多少条外向分支。假设网络中某条边存在 m 个可能的状态,则MDD终节点从左到右依次表示为最劣状态到最优状态。MDD能够表示组件与系统状态之间的关系,便于计算系统部分或整体处于某种状态的概率。

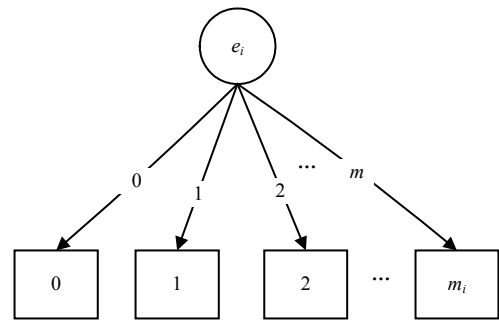


图1 边 e_i 的MDD结构

图1给出了网络中某条边的MDD结构,可以观察到边 e_i 存在 $m+1$ 条外向分支,代表了该边存在的 $m+1$ 个独立状态,分别用 $0, 1, \dots, m_i$ 表示。其中,终节点0表示该边完全失效, m_i 表示该边处于最优状态。

2.2 逻辑操作

根据MDD变量排序,按照顺序将路径中每条边转换为决策图的多值变量 X_i ,该路径中各边的MDD结构同图1。

定义 1 假设 A 和 B 分别代表了两个子MDD的逻辑表达式,并且 A 和 B 的逻辑表达式均为case格式: $A=\text{case}(x, A_0, \dots, A_{m-1})$ 和 $B=\text{case}(x, B_0, \dots, B_{m-1})$ 。通过定义相应MDD操作算子可以得到连接 A 和 B 两个子MDD的结构,按照相同规则递归所有的边或路径则可以得到相应路径或系统的MDD结构,为:

$$A \diamond B = \text{case}(x, A_0, \dots, A_{m-1}) \diamond \text{case}(x, B_0, \dots, B_{m-1})$$

$$= \begin{cases} \text{case}(x, A_0 \diamond B_0, \dots, A_{m-1} \diamond B_{m-1}) & \text{index}(x) = \text{index}(y) \\ \text{case}(x, A_0 \diamond B, \dots, A_{m-1} \diamond B) & \text{index}(x) < \text{index}(y) \\ \text{case}(x, A \diamond B_0, \dots, A \diamond B_{m-1}) & \text{index}(x) > \text{index}(y) \end{cases} \quad (1)$$

2.3 可靠性概率计算

MDD中根节点到达某个终节点的所有路径概率之和表示系统处于该种状态的概率。为了降低MDD概率计算复杂度,使用布尔操作进一步简化MDD结构(根节点到达某个终节点满足需求条件时,将终节点的值由实数类型置为布尔类型)。根据MDD性质,计算系统处于非失效状态下的概率时,从MDD根节点出发,寻找终节点非0的路径,递归调用概率计算函数即可。

定义 2 多状态系统 G 处于非失效状态下的概率 $P(G)$ 通过递归调用概率计算函数,自顶向下遍历MDD结构得出,有:

$$P(G) = P_{y_i=0} P(G_{y_i=0}) + P_{y_i=1} P(G_{y_i=1}) + \dots + P_{y_i=m_i} P(G_{y_i=m_i}) \quad (2)$$

式中, $P_{y_i=j}$ 表示标识变量为 y_i 的边处于状态 j 时的概率。终节点取值为0时, $P(G)=0$; 终节点取值大于0时, $P(G)=1$ 。 $P(G)$ 表示系统处于非失效状态下的概率,对应可靠性的值。

2.4 问题定义

本文中,网络通过2SMPs传输的可靠性定义为:系统 G 在时间 T 和预算 B 约束条件下通过2SMPs从源点 s 到目标节点 t 传输 d 单位数据的概率。不失一般性,假设路径 $P_1=\{e_1, e_2, \dots, e_v\}$ 和 $P_2=\{e_{v+1}, e_{v+2}, \dots, e_u\}$ 为系统给定的2SMPs, d_1 和 d_2 分别表示路径 P_1 和 P_2 传输的单位数据量并且满足 $d=d_1+d_2$ 。相应的,路径 P_1 和 P_2 的传输时间和传输成本分别表示为 $LP_1=le_1+le_2+\dots+le_v$, $LP_2=le_{v+1}+le_{v+2}+\dots+le_u$ 以及 $CP_1=Ce_1+Ce_2+\dots+Ce_v$, $CP_2=Ce_{v+1}+Ce_{v+2}+\dots+Ce_u$ 。

定义 3 任意一条路径 P_e ,其单位时间内传输的数据为 $\min_{e_i \in P_e} w_i$,则路径 P_e 的传输能力 \bar{d} 表示为:

$$\bar{d} = \min_{e_i \in P_e} w_i (T - LP_e) \quad (3)$$

推论 1 P_e 在时间和预算约束下传输的最大数据量表达式,且 A 和 B 的逻辑表达式均为case格式: KP_e ,表示 P_e 在时间约束 T 下数据传输能力 \bar{d} 、预算约束 B 下数据传输能力 d_B 和指定传输单位数据量 d 中的最小值。

$$KP_e = \min(d, d_B, \bar{d}) \quad (4)$$

2.5 备用路径

假设 P_i 和 P_j 分别为网络传输给定单位数据时预先给定的两条工作路径,备用路径定义为:工作路

径 P_i 或 P_j 出现失效情况时,用来替换失效工作路径 P_i 或 P_j 后完成数据传输可靠性最高的路径。

根据文献[15]定义的传输规则,备用路径会在第一条工作路径失效后立即触发,其可靠性 $P_r(P_i P_j, P_k)(i, j, k=1, 2, \dots, n, i \neq j \neq k)$ 计算为:

$$P_r(P_i P_j, P_k) = P_r(\bar{P}_i) P_r(S | P_j P_k) + P_r(\bar{P}_j) P_r(S | P_i P_k) \quad (5)$$

同理,第二条工作路径失效后的可靠性 $P_r(P_i P_j, P_k P_{kk})(i, j, k, kk=1, 2, \dots, m, i \neq j \neq k \neq kk)$ 计算为:

$$P_r(P_i P_j, (P_k P_{kk})) = 2P_r(\bar{P}_i) P_r(\bar{P}_j) P_r(P_k P_{kk}) + P_r(\bar{P}_k) P_r(P_i P_j, P_{kk}) \quad (6)$$

式中, $P_r(\bar{P}_w)(w=1, 2, \dots, m)$ 表示路径处于完全失效状态。根据定义可知,路径处于完全失效状态表示路径中至少存在一条边出现失效情况,有:

$$P_r(\bar{p}_w) = \left(1 - \prod_{r: e \in P_w} P_r(e_r > 0) \right) \quad (7)$$

2.6 预算判断

定义 4 网络中任意一条路径 P_e ,其传输 d_e 单位数据的预算 $F(d_e, P_e)$ 等于路径传输代价与传输单位数据量的乘积。

$$F(d_e, P_e) = CP_e d_e \quad (8)$$

推论 2 已知 P_1 和 P_2 分别表示网络中传输 d 单位数据的两条工作路径。如果 $CP_1=CP_2$,则路径 P_1 和 P_2 传输 d 单位数据的预算为固定值;如果 $CP_1 < CP_2$,则路径 P_1 和 P_2 传输 d 单位数据的预算随着路径 P_1 传输数据的增加而逐渐减小。

证明:当 $CP_1=CP_2$ 时,预算 $F(d)=CP_1 d_1+CP_2(d-d_1)=CP_1 d$;当 $CP_1 < CP_2$ 时,预算 $F(d)=CP_1 d_1+CP_2(d-d_1)$ 。假设存在一个任意小的正整数 θ 使得路径 P_1 传输的单位数据为 $d_1+\theta$,则路径 P_2 传输的单位数据为 $d-d_1-\theta$,使得预算满足 $F_\theta(d) \geq F(d)$,而 $F_d(d)=CP_1(d_1+\theta)+CP_2(d-d_1-\theta)=F(d)-(CP_2-CP_1)\theta$,根据 $CP_1 < CP_2$ 可知, $(CP_2-CP_1)\theta > 0$,推出 $F_\theta(d) < F(d)$,故假设不成立。

推论2说明网络通过2SMPs传输 d 单位数据的预算情况。

3 基于MDD的可靠性分析算法

利用MDD能够双向反映组件状态与系统状态关系的特点,给出算法MDD_2SMPs计算2SMPs传输的可靠性以及算法MDD_BMPs选择网络备用路径。

3.1 算法MDD_2SMPs

算法MDD_2SMPs以网络的2SMPs作为算法输入,输出结果表示网络满足传输需求的概率,对应可靠性的值。该算法主要由4个步骤构成:1)判断

2SMPs是否满足传输需求并计算传输成本 CP_1 和 CP_2 ; 2) 通过定义的And_Min操作, 按照式(1)中的合成规则得到2SMPs的MDD结构(MDD $_P_1$ 和MDD $_P_2$); 3) 根据MDD结构中终节点值有序表示的路径容量进行相应约束剪枝, 通过执行定义的Or_MDD操作合并MDD $_P_1$ 和MDD $_P_2$ 得到结构MDD_result; 4) 遍历MDD_result中终节点非0的路径, 递归调用概率计算函数得到随机流网络(d, T, B, P_1, P_2)的可靠性REL_ (d, T, B, P_1, P_2) 。算法伪代码如下:

Input: A stochastic-flow network (d, T, B, P_1, P_2) with two separate minimal paths, demand level d , time limitation T , and budget constraint B .

Output: The reliability of (d, T, B, P_1, P_2), that is REL_ (d, T, B, P_1, P_2)

Step0 Initialization MDD $_P_1=0$, MDD $_P_2=0$, MDD_result=0, reliability=0

Step1 for $j=1, 2$, Compute $KP_j(M)$, CP_j . Sort the two given SMPs in an ascending order

Step2 if $KP_1*CP_1+(d-KP_1)*CP_2 \leq B$

Construct MDD $_P_1$, MDD $_P_2$

MDD $_P_1 \leftarrow$ MDD_MPSet[1][1]

MDD $_P_2 \leftarrow$ MDD_MPSet[2][1]

LP $_1 \leftarrow 0$, LP $_2 \leftarrow 0$

for $i \leftarrow 2$ to length[MPSet[1]], length[MPSet[2]]

do

LP $_1 \leftarrow$ LP $_1+l_i$, LP $_2 \leftarrow$ LP $_2+l_i$

MDD $_P_1 \leftarrow$ And_Min(MDD $_P_1$, MDD $_e_i$)

MDD $_P_2 \leftarrow$ And_Min(MDD $_P_2$, MDD $_e_i$)

end for

Step3 Get MDD_result

MDD_result \leftarrow Or_MDD(MDD $_P_1$, MDD $_P_2$)

Step4 Compute Reliability (MDD_result)

if (MDD_result = 0) then

return 0

else if (MDD_result is terminal node) then

if (MDD_result = 1) then

return 1

else return 0

else (MDD_result is not terminal node)

for $j \leftarrow 0$ to m_i do

reliability \leftarrow reliability+ p_j *Compute

Reliability(MDD_result)

end for

REL_ $(d, T, B, P_1, P_2) \leftarrow$ reliability
return REL_ (d, T, B, P_1, P_2)

end if

else return “over budget”

3.2 算法MDD_BMPs

算法MDD_BMPs以网络的MPs作为算法输入, 输出结果为选择的备用路径。算法相关背景工作同文献[11,15], 假设网络中MPs= $\{P_1, P_2, \dots, P_n\}$ 已知。在得到网络中各路径以不同容量传输的单位数据信息后, 该算法主要由3个步骤构成: 1) 执行定义的And_Min操作, 按照式(1)中的合成规则得到网络各路径的决策图多值变量形式MDD $_P_i$; 2) 遍历MPs中所有与工作路径都不相交的路径, 计算概率值; 3) 选择概率值最高的路径作为网络备用路径。算法伪代码如下:

Input: A stochastic flow network with all the minimal paths, demand level d , time limitation T , and budget constraint B .

Output: The optimal minimal path

Step0 Initialization MDD $_P=0$, reliability=0, MDD_result1=0, MDD_result2=0, $S=0$

Step1 Construct MDD $_P_i$, for $i=1, 2, 3, \dots, n$

MDD $_P_i \leftarrow$ MDD_MPSet[i]

end of for

Step2 Compute reliability

for $j \leftarrow 3$ to n do

MDD_result1 \leftarrow Or_MDD(MDD $_P_1$, MDD $_P_j$)

MDD_result2 \leftarrow Or_MDD(MDD $_P_2$, MDD $_P_j$)

end for

use (5) to get reliability

Step3 Return optimal minimal path

reliability=Rel_ $(d, T, B, (P_3))$

if Rel_ $(d, T, B, (P_j)) >$ probability then let Rel_ $(d, T, B, (P_j)) =$ probability, and $S = P_j$

return S

3.3 复杂度分析

对于任意一个存在2SMPs的随机流网络, 算法MDD_2SMPs的时间复杂度分析过程如下: 1) 确定 KP_1 和 KP_2 花费的时间最多为 $O(n)$, 因为每条路径所含边的数量不会超过 n 条。2) 假设 m 表示路径边上存在的最大状态数, 则生成MDD $_e_i$ 的耗时最多为

$O(m)$; 已知MDD执行逻辑与和逻辑或操作的耗时为 $O(n_1n_2)$ (n_1 和 n_2 分别表示参与操作的两个MDD的节点数), 得到构建MDD $_i$ 和MDD $_j$ 过程的耗时为 $O(nm)$ 。3) 对路径容量进行约束剪枝耗时为 $O(nR_1R_2)$ (R_1 和 R_2 分别表示参与操作的两个MDD的终节点数量), 因为MDD $_i$ 和MDD $_j$ 终节点表示的路径容量是唯一有序的, 故只需访问对应的终节点值; 在工作路径合并过程中, 定义的Or_MDD操作同样是一种逻辑或操作, 该过程耗时为 $O(n_1n_2)$ 。4) 文献[6]定理1关于MDD概率计算的复杂度分析过程可知, 遍历MDD_result计算可靠性的耗时为 $O(m^n/n)$ 。综合以上分析, 算法MDD_2SMPs复杂度可以表示为 $O(n^2+nC+m^n)$ 。

4 实例分析

本节以文献[11,15]使用的基准网络为例, 说明可靠性分析算法MDD_2SMPs和备用路径选择算法MDD_BMPs具体执行过程。网络拓扑结构如图2所示, 图中 s 表示源点, t 表示目标节点, $a_1 \sim a_{22}$ 表示有22条边, 各边的详细信息在表1给出。实例中, 网络额定传输的单位数据 d 为200 Gb, 时间 T 不超过13 s, 预算 B 不高于2000元。网络MPs已知条件下, 假设工作路径分别由 $P_1=\{a_1, a_2, a_3\}$ 和 $P_2=\{a_4, a_5, a_6\}$ 组成, 网络中与工作路径 P_1 和 P_2 都不相交的路径为: $P_3=\{a_8, a_9, a_{10}\}$ 、 $P_4=\{a_{11}, a_{12}, a_{13}\}$ 、 $P_5=\{a_{15}, a_{16}, a_{17}\}$ 、 $P_6=\{a_{18}, a_{19}, a_{20}\}$ 、 $P_7=\{a_{10}, a_{11}, a_{14}\}$ 、 $P_8=\{a_{15}, a_{22}\}$ 、 $P_9=\{a_{18}, a_{21}, a_{22}\}$ 及 $P_{10}=\{a_{16}, a_{17}, a_{18}, a_{21}\}$ 。

4.1 可靠性计算

通过前文条件及表1信息, 计算 $KP_1=200$ 、 $KP_2=120$ 、 $CP_1=10$ 、 $CP_2=7$ 。

- 1) 按照传输规则, 路径 (P_1, P_2) 满足传输条件; 已知 $CP_2 < CP_1$, 确定参照路径 P_2 。
- 2) 根据给出的变量顺序, 将路径 (P_1, P_2) 中每条边转换为图1所示的决策图多值变量形式, 并按照公式(1)中的合成规则, 执行定义的And_Min操作, 分别得到图3所示路径 P_1 的MDD结构MDD $_1$ 和图4所示路径 P_2 的MDD结构MDD $_2$ 。
- 3) 利用MDD $_1$ 和MDD $_2$ 结构中终节点值有序表示的路径容量进行约束剪枝, 执行Or_MDD操作合并路径 (P_1, P_2) 。

①已知步骤1)选取的参照路径为 P_2 , 观察其终节点值从左至右依次为0、10、20、30、40, 对应路径 P_2 的5种容量情况。终节点值为0表示路径 P_2 完全失效, 按照传输规则, 计算 P_1 满足传输条件的路径容量为40; 并根据推论2可知预算取得最大值, 终止预算判断过程。

②路径 P_2 容量为10时, 其传输的单位数据最多不超过30 Gb, 按照传输规则, 计算 P_1 满足传输条件的路径容量至少为30。

同理, 可以分别得到路径 P_2 容量为20、30和40时, P_1 上满足传输条件的路径容量。

③根据MDD原理及运算规则, 规约化简得到图5所示的结构。

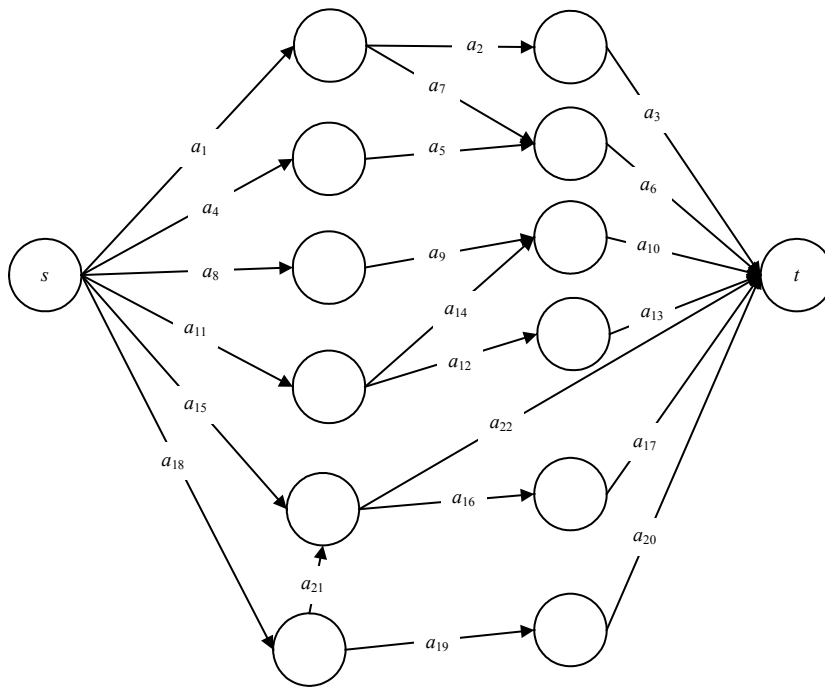


图2 基准网络实例

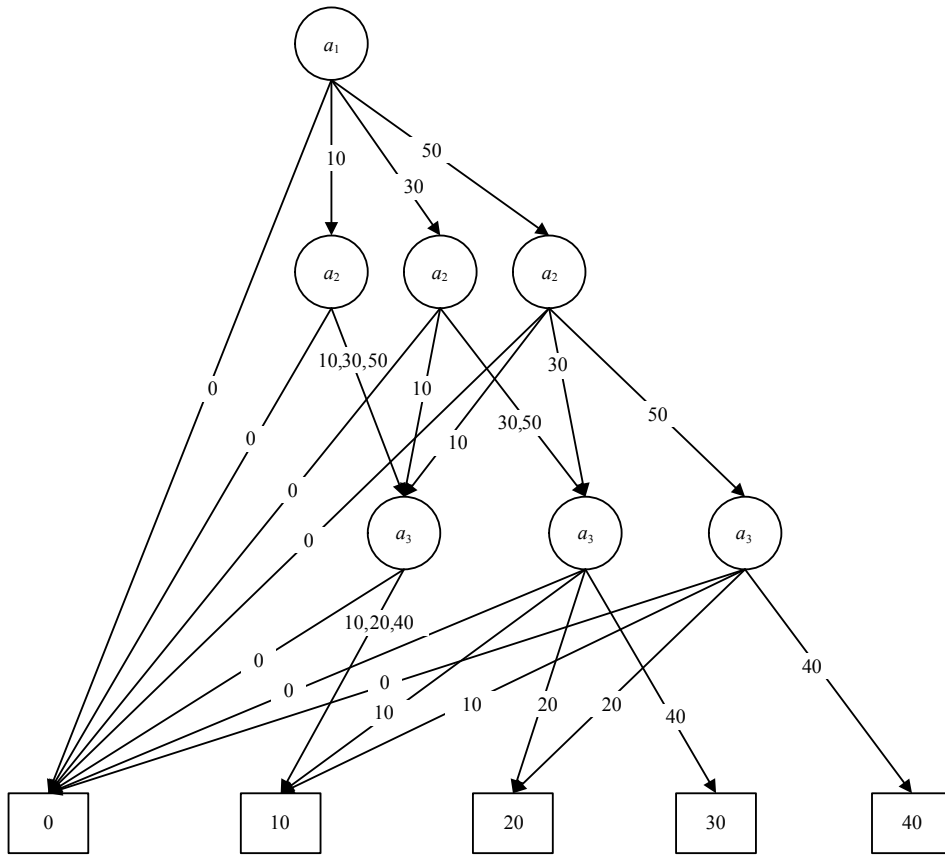


图3 路径 P_1 的MDD结构

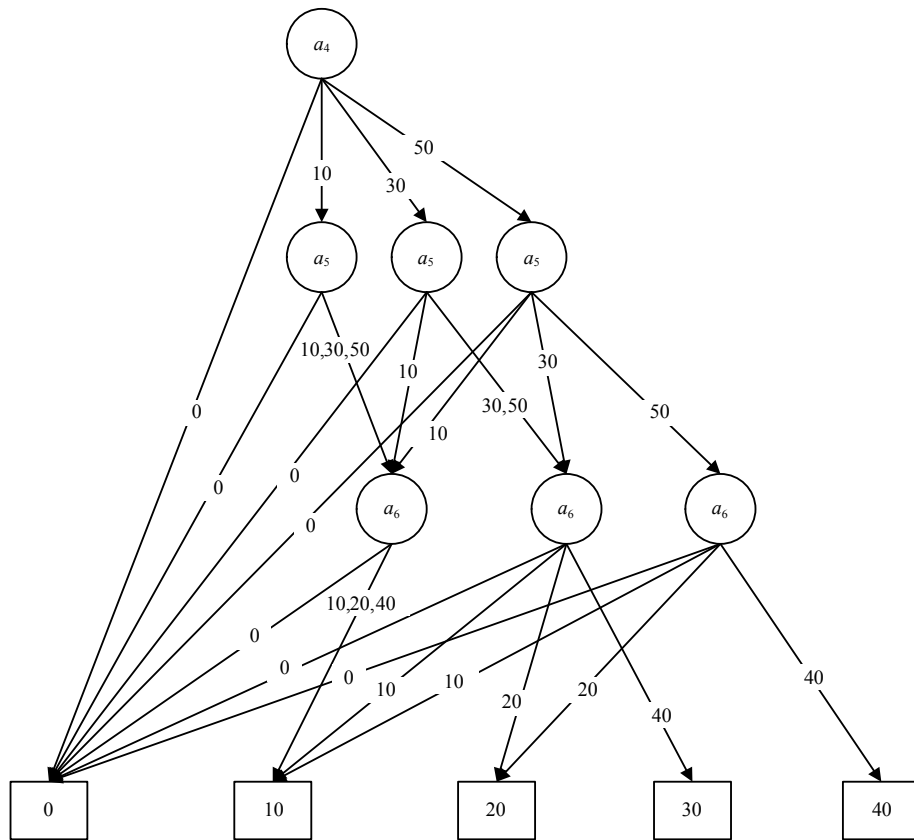


图4 路径 P_2 的MDD结构

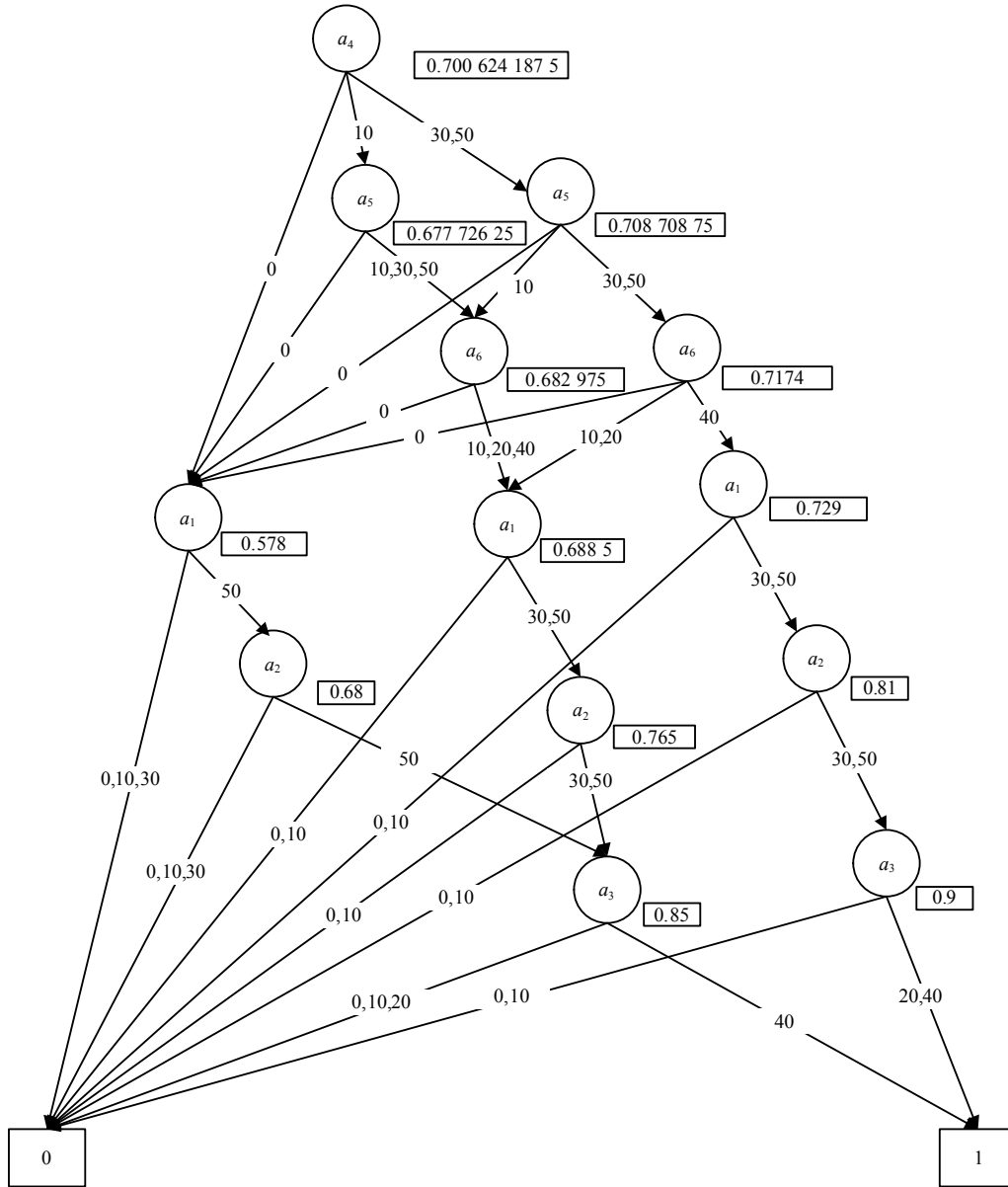


图5 MDD_result

表1 图2中各边数据信息

边	容量	概率	时延	成本	边	容量	概率	时延	成本
a_1	50 ^a	0.85	2	3	a_{12}	60	0.80	2	2
	30	0.05				40	0.05		
	10	0.05				20	0.05		
	0	0.05				10	0.05		
a_2	50	0.80	2	4	a_{13}	0	0.05		
	30	0.10				60	0.75	1	3
	10	0.05				40	0.10		
	0	0.05				20	0.05		
a_3	40	0.85	3	3	a_{14}	10	0.05		
	20	0.05				0	0.05		
	10	0.05				20	0.95	2	2
	0	0.05				0	0.05		
a_4	50	0.85	3	2	a_{15}	70	0.80	3	2
	30	0.05				50	0.05		
	10	0.05				30	0.05		
	0	0.05				10	0.05		

(续表)

边	容量	概率	时延	成本	边	容量	概率	时延	成本
a_5	50	0.80	4	3	a_{16}	0	0.05	2	2
	30	0.10				60	0.75		
	10	0.05				40	0.10		
	0	0.05				30	0.05		
a_6	40	0.85	3	2	a_{17}	10	0.05	3	3
	20	0.05				0	0.05		
	10	0.05				50	0.85		
	0	0.05				30	0.05		
a_7	20	0.90	2	1	a_{18}	10	0.05	3	4
	0	0.10				0	0.05		
a_8	50	0.85	3	3	a_{19}	40	0.85	3	1
	30	0.05				30	0.05		
	10	0.05				10	0.05		
	0	0.05				0	0.05		
a_9	40	0.85	4	1	a_{20}	50	0.85	3	2
	20	0.05				30	0.05		
	10	0.05				10	0.05		
	0	0.05				0	0.05		
a_{10}	40	0.80	2	2	a_{21}	40	0.85	3	2
	20	0.10				20	0.05		
	10	0.05				10	0.05		
	0	0.05				0	0.05		
a_{11}	50	0.85	3	1	a_{22}	20	0.95	2	2
	30	0.05				0	0.05		
	10	0.05				10	0.95		
	0	0.05				0	0.05		

注: 50^a 表示边 a_1 容量处于50时的概率为0.85

表2 两种算法获取 (d, T, B) 下路径所需容量的操作数

不相交路径 (P_i, P_j)	算法 $(d, T, B, (P_i, P_j))$ -LBPs	算法MDD_2SMPs	不相交路径 (P_i, P_j)	算法 $(d, T, B, (P_i, P_j))$ -LBPs	算法MDD_2SMPs
P_1, P_2	3 718	150	P_4, P_{10}	352	126
P_3, P_4	6 358	150	P_5, P_6	6 358	150
P_3, P_5	6 358	150	P_5, P_7	3 718	90
P_3, P_6	3 718	150	P_5, P_9	550	60
P_3, P_8	550	50	P_6, P_7	1 782	90
P_3, P_9	22	60	P_6, P_8	198	50
P_3, P_{10}	198	105	P_7, P_8	0	0
P_4, P_5	9 702	180	P_7, P_9	0	0
P_4, P_6	6 358	180	P_7, P_{10}	0	0
P_4, P_8	1 078	60	P_8, P_{10}	0	0
P_4, P_9	550	72			

4) 自顶向下遍历生成的结构MDD_result, 递归调用Compute Reliability函数, 计算网络可靠性值为 $REL_{(200, 13, 2000, P_1, P_2)}=0.700\ 624$ 。

针对该网络, 表2给出了算法MDD_2SMPs与算法 $(d, T, B, (P_i, P_j))$ -LBPs计算2SMPs可靠性时, 获取 (d, T, B) 下路径所需容量的操作数情况。以工作路径 $P_1=\{a_1, a_2, a_3\}$ 和 $P_2=\{a_4, a_5, a_6\}$ 为例, 本文算法通过

定义And_Min操作算子, 直接得到唯一且有序的路径容量, 只需根据MDD中终节点的值进行约束剪枝, 并且只需存储工作路径所含边的信息(6条)。因此, 算法MDD_2SMPs获取 (d, T, B) 下工作路径所需容量的操作数为 $6 \times 5 \times 5 = 150$ 次。而文献[15]中算法 $(d, T, B, (P_i, P_j))$ -LBPs根据路径 P_1 和 P_2 存在的流量分配情况计算, 通常会得到相同的路径容量, 如路径 (P_1, P_2) 通过的流量为(80,120)、(90,110)或(100,100)

时, 得到的路径容量均为(20,40)。为便于计算可靠性的值, 还需构建系统状态空间向量(即文献[15]中 X 向量)过滤重复的以及非最小的解, 这些系统状态空间向量存储了整个网络的边(22条), 合计需要

$22 \times 13 + 22 \times (13 \times 12) = 3\ 718$ 次移除操作。对于网络可靠性概率值, 本文算法基于节点概率迭代计算; 算法 $(d, T, B, (P_i, P_j))$ -LBP_s获取系统最小容量向量后, 还需借助容斥原理等方法进行计算。

表3 各路径可能传输的单位数据

路径	单位容量	最大容量	概率	时延	成本	路径	单位容量	最大容量	概率	时延	成本
P_1	40 ^a	200	0.578	7	10	P_5	50	200	0.541 875	8	7
	30	180	0.110 5				40	200	0.072 25		
	20	120	0.040 5				30	150	0.114 875		
	10	60	0.128 375				10	50	0.128 375		
	0	0	0.142 625				0	0	0.142 625		
P_2	40	120	0.578	10	7	P_6	40	160	0.614 125	9	7
	30	90	0.110 5				30	120	0.047 25		
	20	60	0.040 5				20	80	0.067 625		
	10	30	0.128 375				10	40	0.128 375		
	0	0	0.142 625				0	0	0.142 625		
P_3	40	160	0.578	9	6	P_7	20	120	0.769 5	7	5
	30	120	0.034				10	60	0.087 875		
	20	80	0.117				0	0	0.142 625		
	10	40	0.128 375				10	60	0.902 5		
	0	0	0.142 625				0	0	0.097 5		
P_4	50	200	0.51	6	6	P_9	10	40	0.857 375	9	7
	40	200	0.104 125				0	0	0.142 625		
	30	200	0.036 125				20	60	0.692 55		
	20	140	0.078 75				10	30	0.121 956 25		
	10	70	0.128 375				0	0	0.185 493 75		
	0	0	0.142 625								

注: 40^a表示路径 P_1 容量为40时, 在 (d, T, B) 约束下通过的单位数据为200 Gb

4.2 备用路径选择

利用MDD能够双向反映组件状态与系统状态的特点, 算法MDD_BMPs通过将表3中各路径传输的单位数据转换为相应的多状态变量, 大大简化了计算的复杂性, 具体过程如下。

1) 根据表3信息构建各路径的决策图多值变量形式MDD_ P_i 。

2) 遍历MPs中与路径 (P_1, P_2) 都不相交的路径 $(P_3, P_4, P_5, P_6, P_7, P_8, P_9$ 和 $P_{10})$, 计算可靠性值。

① 以路径 P_3 为例, 由步骤1)得到图6所示的决策图多值变量形式MDD_ P_3 。

② 计算 $CP_1=10$ 、 $CP_2=7$ 、 $CP_3=9$, 确定路径 (P_1, P_3) 和 (P_2, P_3) 变量序分别为 $P_3 < P_1$ 和 $P_2 < P_3$ 。

③ 由得到的变量序选取参照路径, 运用推论2获取对应路径的终节点状态。

④ 最后, 根据MDD概率计算函数及式(5)定义的传输规则, 计算备用路径为 P_3 时的可靠性值 $P_r(P_1 P_2, P_3)=0.183\ 823\ 96$ 。

3) 根据各路径取值情况返回可靠性最高的路径。

图7为第一条工作路径失效后, 各路径的可靠性取值情况, 由此选择备用路径为 P_4 , 可靠性值 $P_r(P_1 P_2, P_4)=0.227\ 665$ 。

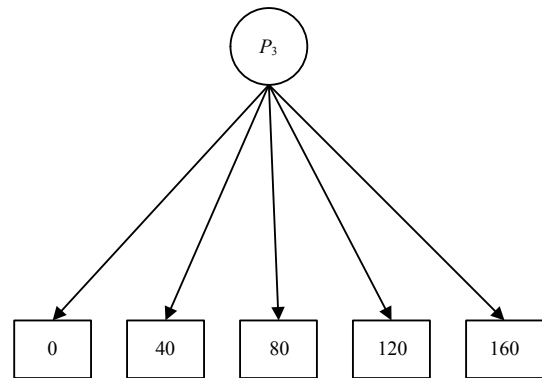


图6 路径 P_3 的MDD结构

同理, 第二条工作路径失效后, 根据图8选择备用路径为 P_5 , 可靠性值 $P_r(P_1 P_2, (P_4 P_5))=0.068\ 328$ 。

对于图2所示网络, 算法MDD_BMPs的计算结果与文献[15]算法 $(d, T, B, (P_k))$ -LBP_s的计算结果一致。不同的是, 算法MDD_BMPs通过将网络各路径转换为决策图多值变量形式, 大大降低了计算的复杂性。以路径 P_3 执行过程为例, 算法MDD_BMPs判断路径 (P_1, P_3) 和 (P_2, P_3) 终节点状态需要的操作数为 $50+50=100$ 次; 算法 $(d, T, B, (P_k))$ -LBP_s则将路径 (P_1, P_2) 分别与路径 P_3 组合获取系统最小容量向量, 该过程所需操作数为 $6\ 358+3\ 718=10\ 076$ 次。

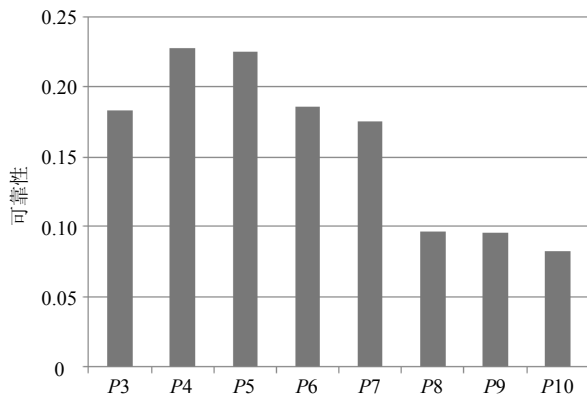


图7 第一条路径失效后各路径的可靠性取值情况

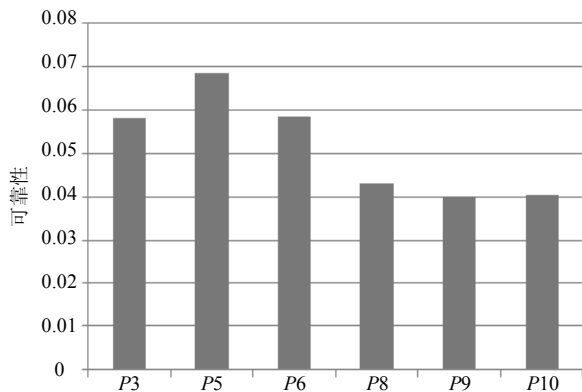


图8 第二条路径失效后各路径的可靠性取值情况

5 结束语

2SMPs传输的随机流网络可靠性分析,通常是求取系统最小容量向量,运用容斥原理计算可靠性值。本文根据MDD能够双向反映组件状态与系统状态关系的特点,提出算法MDD_2SMPs,实例结果表明,相比获取系统最小容量向量方法,本文算法主要有以下特点:1)通过定义MDD操作算子,得到路径容量是唯一有序的,减少了路径容量的约束判断;2)在系统结构不变情况下,可用于不同参数的可靠性分析,避免了存储整个网络的边以及运用容斥原理计算的繁琐。此外,针对网络工作路径失效的问题,算法MDD_BMPs通过将网络各路径转换为决策图多值变量形式,大大降低了选择可靠备用路径的计算复杂性。

参考文献

[1] CHANG S, LI L. Reliability analysis of highway and transportation network with paths failure[C]//Advanced Research and Technology in Industry Applications (WARTIA). [S.l.]: IEEE, 2014: 50-53.
 [2] VAZIFEHDAN J, PRASAD R V, NIEMEGEREERS I. Energy-efficient reliable routing considering residual energy in wireless ad hoc networks[J]. IEEE Transactions on Mobile Computing, 2014, 13(2): 434-447.

[3] 李振, 孙新利, 姬国勋, 等. 计算多状态网络可靠度的不交化改进算法[J]. 通信学报, 2011, 21(9A): 166-172.
 LI Zhen, SUN Xin-li, JI Guo-xun, et al. An improved algorithm for the non intersection of multi-state network reliability[J]. Chinese Journal of Communications, 2011, 21(9A): 166-172.
 [4] RAMIREZ-MARQUEZ J E, COIT D W. A Monte-Carlo simulation approach for approximating multi-state two-terminal reliability[J]. Reliability Engineering & System Safety, 2005, 87(2): 253-264.
 [5] JANE C C, LAIH Y W. Computing Multi-State Two-Terminal Reliability through critical arc states that interrupt demand[J]. IEEE Transactions on Reliability, 2010, 59(2): 338-345.
 [6] SHRESTHA A, XING L, DAI Y. Decision diagram based methods and complexity analysis for multi-state systems[J]. IEEE Transactions on Reliability, 2010, 59(1): 145-161.
 [7] YEH W C. A simple MC-based algorithm for evaluating reliability of stochastic-flow network with unreliable nodes[J]. Reliability Engineering & System Safety, 2004, 83(1): 47-55.
 [8] LIN Y K. Extend the quickest path problem to the system reliability evaluation for a stochastic-flow network[J]. Computers & Operations Research, 2003, 30(4): 567-575.
 [9] LIN Y K. A method to evaluate routing policy through p, minimal paths for stochastic case[J]. Information Sciences, 2010, 180(23): 4595-4605.
 [10] LIN Y K. Spare routing reliability for a stochastic flow network through two minimal paths under budget constraint[J]. IEEE Transactions on Reliability, 2010, 59(1): 2-10.
 [11] LIN Y K. Spare routing reliability for a stochastic flow network through two minimal paths under budget constraint[J]. IEEE Transactions on Reliability, 2010, 59(1): 2-10.
 [12] FORGHANI-ELAHABAD M, MAHDAVI-AMIRI N. An efficient algorithm for the multi-state two separate minimal paths reliability problem with budget constraint[J]. Reliability Engineering & System Safety, 2015, 142: 472-481.
 [13] LIN Y K, HUANG C F. A multi-state computer network within transmission error rate and time constraints[J]. Journal of the Chinese Institute of Industrial Engineers, 2012, 29(7): 477-484.
 [14] LIN Y K, HUANG C F. Assessing reliability within error rate and time constraint for a stochastic node-imperfect computer network[J]. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability, 2013, 227(1): 80-85.
 [15] ZHANG Y, XU Z, WANG X, et al. Single minimal path based backup path for multi-state network[J]. Proceedings of the Institution of Mechanical Engineers Part O Journal of Risk & Reliability, 2013, 228(2): 152-165.
 [16] 古天龙, 徐周波. 有序二叉决策图及应用[M]. 北京: 科学出版社, 2009.
 GU Tian-long, XU Zhou-bo. Ordered binary decision diagram and its application[M]. Beijing: The Science Publishing Company, 2009.
 [17] DONG R, ZHU Y, XU Z, Li F. Decision diagram based symbolic algorithm for evaluating the reliability of multistate flow network[J]. Mathematical Problems in Engineering, 2016, doi:10.1155/2016/6908120.