Journal of System Simulation

Volume 28 | Issue 4 Article 10

7-2-2020

Generate Network's Disjoint MPs Based on SimEvents Simulation

Tang Jian
1. Mechanic & Electronic Engineering Center, College of Field Engineering, PLA University of Science and Technology, Nanjing 210007, China;;

Fuli Ai
2. Northwest Industries Group CO., LTD, HORINCO Group, Xi'an 710043, China;

Faming Shao
1. Mechanic & Electronic Engineering Center, College of Field Engineering, PLA University of Science and Technology, Nanjing 210007, China;;

Jiaojiao Zhang
1. Mechanic & Electronic Engineering Center, College of Field Engineering, PLA University of Science and Technology, Nanjing 210007, China;;

Generate Network's Disjoint MPs Based on SimEvents Simulation

Abstract

Abstract: The disjoint MPs (Minimal Path sets) algorithm proposed was *improved from the point of signal transmission. The CoA network was transformed into a corresponding signal netw*ork, which was composed of forward and backward transmitting paths. According to the basic principle of disjoint MPs algorithm, *a set of signal transmitting and rewriting rules, including forward and backward processing rules, were designed to make the disjoint MPs could be generated by means of DES (Discrete Event Simulation). The basic realizing idea was studied in SimEvents, in which, the entities acted as the information carriers, and the nodes as the message storage and processing units*. During the operation of the network model, messages carried by entities will be constantly rewritten and transmitted before all the entities arrived to the terminal node. Until then, all the disjoint MPs could be automatically generated. The simulation results of bridged network and a more complex one verified the correctness of the transmitting rules and the feasibility to realize this algorithm by means of DES.

Keywords

network, disjoint Minimal Path sets, discrete event simulation, SimEvents

Recommended Citation

Tang Jian, Ai Fuli, Shao Faming, Zhang Jiaojiao. Generate Network's Disjoint MPs Based on SimEvents Simulation[J]. Journal of System Simulation, 2016, 28(4): 842-850.

第 28 卷第 4 期 系统仿真学报© Vol. 28 No. 4
2016 年 4 月 Journal of System Simulation Apr., 2016

基于 SimEvents 仿真获取网络不交化最小路集

唐建[1]，艾芙莉[2]，邵发明[1]，张蕉蕉[1]
(1. 解放军理工大学 野战工程学院机电工程教研中心，江苏 南京 210007；
2. 中国兵器工业集团 西北工业集团有限公司，陕西 西安 710043)

摘要： *从信息传递角度，将 CoA 网络转换为具有前向和逆向传输路径的信息传输网络；在对网络不交化 MPs(Minimal Path sets)算法原理分析基础上，设计了信息在网络中的传输和改写规则(包括正向和逆向传输规则)，并以离散事件仿真(Discrete Event Simulation，DES)为手段，对网络建模，对算法实现。以 SimEvents 为平台，阐述了基于 DES 进行算法实现的基本思路：即以实体(Entity)为信息载体，以节点为暂存和处理单元。仿真过程中，信息随实体在网络中传输，并不断改写，直至完成不交化 MPs 的生成。对桥型网络和复杂网络的仿真结果验证了信息处理规则的正确性，和基于 DES 进行算法实现的可行性。*

关键词： 网络；不交化最小路集；离散事件仿真；SimEvents
中图分类号：TP391.9 文献标识码：A 文章编号：1004-731X (2016) 04-0842-09

Generate Network's Disjoint MPs Based on SimEvents Simulation

Tang Jian[1], *Ai Fuli*[2], *Shao Faming*[1], *Zhang Jiaojiao*[1]
(1. Mechanic & Electronic Engineering Center, College of Field Engineering, PLA University of Science and Technology, Nanjing 210007, China;
2. Northwest Industries Group CO., LTD, HORINCO Group, Xi'an 710043, China)

Abstract: The disjoint MPs (Minimal Path sets) algorithm proposed was *improved from the point of signal transmission. The CoA network was transformed into a corresponding signal net*work, which was composed of forward and backward transmitting paths. According to the basic principle of disjoint MPs algorithm, *a set of signal transmitting and rewriting rules, including forward and backward processing rules, were designed to make the disjoint MPs could be generated by means of DES (Discrete Event Simulation). The basic realizing idea was studied in SimEvents, in which, the entities acted as the information carriers, and the nodes as the message storage and processing units*. During the operation of the network model, messages carried by entities will be constantly rewritten and transmitted before all the entities arrived to the terminal node. Until then, all the disjoint MPs could be automatically generated. The simulation results of bridged network and a more complex one verified the correctness of the transmitting rules and the feasibility to realize this algorithm by means of DES.

Keywords: network; disjoint Minimal Path sets; discrete event simulation; SimEvents

引言

不交化最小路集(Minimal Path sets，MPs)算法是网络联通可靠性分析的重要方法之一。它运用不交积和定理，将网络可靠度表示为全部最小路集的并，然后将这些并化为不交项的和，继而计算网络的可靠度[1]。常用的不交化最小路集获取方法包括容斥原理法(Inclusion-exclusion principle)[2-3]，SDP(Sum of Disjoint Products)法[4-6]、二元决策图(Binary Decision Diagram，BDD)[7-8]、Ahmad 法[9]、

收稿日期：2014-11-11 修回日期：2015-01-18；
基金项目：国家自然科学基金 (51175511)；解放军理工大学青年基金 (42413461A)；
作者简介：唐建(1977-)，女，四川成都，副教授，博士后，硕导，研究方向为复杂系统可靠性及仿真；艾芙莉(1975-)，女，辽宁昌图，硕士，研究方向为机电系统可靠性建模与分析。

真值表法、卡诺图法、全概率分解法、节点搜索法等等。这其中，真值表法、卡诺图法等只适用于小规模网络，而容斥原理法则需要在网络全部最小路(MPs)已知的基础上，再通过不交化计算，才能获取其不交化 MPs，计算过程繁琐。而对于大规模网络，网络的状态将面临组合爆炸问题，不交化过程的计算量将随着网络规模的增大而急剧增大[10]。因此，为使大规模网络可靠度的计算高效且可靠，首先应当进行算法研究，避免先获取 MPs，再进行不交化计算的繁琐，这类方法称为一步法；在此基础上，需进行基于计算机的算法实现研究，以提高网络，尤其是大规模网络不交化 MPs 获取的效率和可靠性。而在基于计算机实现的算法创新方面，Yeh[11]、Levitin[12]等都进行过深入研究。

对于网络不交化 MPs 的获取，武小悦[13]从网络的结构函数入手，提出了一种直接获取不交化 MPs 的算法；Xing[14]基于不交化运算法原理，提出了改进的不交化 MPs 算法；Rouchdy Y[15]也基于容斥原理，提出一步不交化 MPs 算法。上述算法的共同特点：一是避免了先获取最小路集，再进行不交化的繁琐；二是，其实现还是基于人工推导，未对基于计算机的算法实现做进一步的研究。本文在分析上述算法特点的基础上，将不交化 MPs 的获取过程视为信息在网络中的传输过程，在制定一套完整的信息传输和改写规则基础上，探索以离散事件仿真(Discrete Event Simulation，DES)为手段实现网络不交化 MPs 的自动获取。

本文的主要工作包括：(1)针对网络不交化 MPs 的获取，制定了一套信息处理规则，该规则可实现网络不交化 MPs 的获取，同时又符合 DES 建模和运行的特点；(2)以 SimEvents 为平台，对算法进行实现，由此详细分析在 DES 平台中，网络建模和算法实现的基本思路；(3)以简单桥型网络和复杂网络为例，验证信息传递和处理规则的正确性和基于 DES 进行算法实现的可行性。

1 算法原理及实现思路

1.1 定义和假设

设 $G(V,A)$ 为一个两终端网络，其中 $V=\{i,i=1,2,\cdots,n\}$ 为其节点集；$A=\{a_k,k=1,\cdots,m\}$ 为弧线集，n 和 m 分别为节点和弧线的数量。节点集中，节点 1 为源点，n 为终点，其它为中间节点。节点与弧线之间存在映射 $\psi:A\to V\times V$，$\psi(a_k)=(i,j)$。i 和 j 分别为弧线 a_k 的起始和终止节点。

该网络为一个 CoA(Component on Arc)网络，且满足以下假设条件：

(1) 网络中的节点始终可靠，弧线可能失效；

(2) 弧线状态相互独立，且不考虑修复问题。

图 1 所示即为一个 CoA 结构的桥型网络。

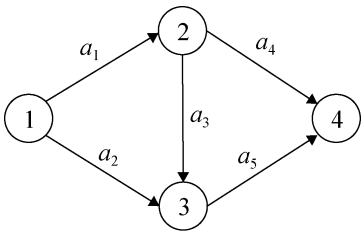


图 1 桥型网络

1.2 基于信息传递的不交化 MPs 的生成规则

本文将不交化 MPs 的获取过程视为信息在网络中的传输过程。在这个过程中，首先使一个空信息从网络的源点出发，通过网络的中间节点向网络的终点传输；在传输中，信息将根据所制定的一套传输和改写规则，根据其自身所携带的信息以及其所经过的路径信息，不断被传输、改写、复制和删除，直至到达终点，其所记录的信息即为网络的不交化 MPs。

这其中，信息处理规则的制定是关键，其内容详细描述如下：

令 IS_i 为节点 i ($i\in V$)的信息集，m_r 为信息的元素集($m_r\in IS_i$)，从信息传递角度分析不交化 MPs 的获取规则如下：

第 1 步：初始化 IS_i，使 $IS_1=\{m_0\}$，$m_0=\varnothing$，$IS_i=\varnothing$ $(i\in V, i\neq 1)$。

第 28 卷第 4 期 系统仿真学报 Vol. 28 No. 4
2016 年 4 月 Journal of System Simulation Apr., 2016

第 2 步：$\forall i \in V$ 且 $i \neq n$ (n 为终点)，将 IS_i 中的每个 m_r 进行前向和逆向传送：

(1) 前向传送时，依次序扫描节点 i 的后续弧线集 OA_i，并按以下规则传递和改写 m_r：

1) 确定 m_r 的发送弧线集 OA_i^*，规则为：$\forall a_k \in OA_i$，当 $a_k \notin m_r$ 或 $\bar{a}_k \notin m_r$ 时，使 $a_k \in OA_i^*$ ($OA_i^* \subset OA_i$)。

2) 前向发送信息 m_r 至 OA_i^*：

首先，将 m_r 经 OA_i^* 中的第 1 条弧线 a_{k_1} 发送至节点 j_1 ($\psi(a_{k_1}) = (i, j_1)$)，若 $a_k \notin m_r$ 且 $\bar{a}_k \notin m_r$，则 m_r 改写为 $\{m_r, a_{k_1}\}$，并存入信息集 IS_{j_1} 中；否则，不传输；

随后，将 m_r 向 OA_i^* 中的第 2 条弧线 a_{k_2} ($\psi(a_{k_2}) = (i, j_2)$)传递，若 $a_k \notin m_r$ 且 $\bar{a}_k \notin m_r$，则传至 j_2 的 m_r 改写为 $\{m_r, \bar{a}_{k_1}, a_{k_2}\}$，存入 IS_{j_2} 中；否则，不发送；

以同样的规则将 m_r 向 OA_i^* 中的第 3，4，…，l_i 条(l_i 为 OA_i^* 中弧线的数量)发送。

(2) 逆向传送时：首先，设节点 i 的前导弧线集为 IA_i，获取 m_r 对于节点 i 的发送弧线集 IA_i^*。求取规则为：$\forall a_k \in IA_i$，当且仅当 $a_k \in m_r$ 时，使 $a_k \in IA_i^*$。随后，向 IA_i^* 中的弧线逆向发送 m_r，规则为：$\forall a_k \in OA_i$ (注意：OA_i 为节点的后续弧线集)，当且仅当 $a_k \notin OA_i$ 且 $\bar{a}_k \notin OA_i$ 时，向 a_k 的始端节点 j 发送 m_r，并改写。改写规则为：$\forall\ a_{k_r} \in OA_i$，当 $a_{k_r} \notin m_r$ 或 $\bar{a}_{k_r} \notin m_r$ 时，将 m_r 改写为 $\{m_r, \bar{a}_{k_r}, \cdots\}$，存入 IS_j。

(3) 从 IS_i 中删除 m_r。

反复进行第 2 步，直到 $\forall i \in V$ 且 $i \neq n$，$IS_i = \varnothing$，终点 n 的 IS_n 即为该网络的不交化 MPs。

该算法可避免先求取 MPs，再进行不交化的两阶段过程，且具有分布计算的特点。可以证明：该算法必将在有限步内实现 $\forall i \in V$ 且 $i \neq n$，$IS_i = \varnothing$，使信息传输终止。

1.3 基于 DES 的算法实现思路

改进后的算法具有并行计算的特点，所有信息都进行局部处理和存储，信息之间不发生相互作用。因此，可将网络节点视作信息的存储和处理单元，原网络可转换为信息传输网络，其正向传输和逆向传输都有独立的传输路径。在此基础上，可将信息传输网络视为一个离散事件系统，并可用离散事件仿真(Discrete Event Simulation)对算法进行实现。

在离散事件仿真过程中，信息可以属性(Attribute)的形式赋给实体(Entity)，实体则携带信息在网络中按照既定规则不断被生成、传递和丢弃，其所携带的信息则被不断改写和暂存。可以预见，虽然 m_r 会随着传递次数的增多而变长，但基于信息前向和后向传输规则，其 IA_i^* 和 OA_i^* 的规模会逐渐变小，直到除终点外的所有节点中都不再存有实体，仿真将自动结束，并记录下网络的全部不交化 MPs。

2 算法在 SimEvents 中的实现

作为 MATLAB 中的离散事件仿真平台，SimEvents 具有以下特点[16-17]：(1) 提供基于实体和令牌的建模；(2) 实体具有高负荷性，可携带大规模向量和矩阵，且数据的层次和同步性通过属性聚合于实体；(3) 提供队列、服务器、实体生成、信号生成、事件生成、路径选择等多种模块；(4) 支持事件触发。同时，鉴于所建模型的直观性，研究中选择 SimEvents 为算法实现平台。

下面，以图 1 所示的桥型网络为例，对网络不交化 MPs 算法的实现思路进行介绍。

2.1 信息传输网络模型

图 2 所示为转换后的桥型网络(图 1)的信息传输网络模型。其中，“□”表示网络的节点，可视为信息的暂存单元。节点之间有 2 种连接路径，一种为正向传输路径，信息从弧线的起点传至弧线的终点，在图中用“⟶”表示；一种为逆向传输路径，信息从弧线的终点传回弧线的起点，用“⇢”表示。

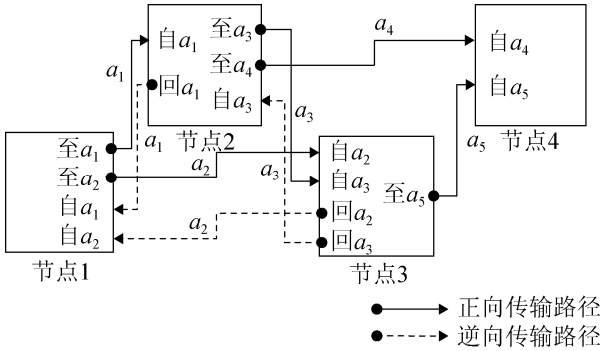


图 2 桥型网络的信息传输网络模型

在图 1 中的桥型网络中，弧线 a_1 的起点为节点 1，终点为节点 2，它们之间有 2 条传输路径，一条自节点 1 传至弧线 a_1，再传至节点 2，为正向传输路径；一条自节点 2 传回弧线 a_1，再回传至节点 1，为逆向传输路径。因此，节点中的“至”表示信息从该节点正向传至弧线 a_i，“自”表示信息自弧线 a_i 传至该节点；“回”表示信息从该节点传回弧线 a_i。

需要注意的是，节点 4 为整个网络的终点，信息传入后将不再回传，因此，与其连接的只有正向传输路径，没有逆向传输路径。

2.2 中间节点模型

中间节点为信息的暂存和处理单元，承担着 3 个功能：(1)对自前导节点传入的信息按正向传输处理规则处理；(2)对自后续节点返回的信息按逆向传输处理规则处理；(3)将处理过的信息暂时存储。由于信息正向传输与逆向传输的处理规则完全不同，故其含有一个正向处理器和一个逆向处理器。

以节点 2 为例，信息正向处理器 Forward Process_Arc1(节点 2 的前导弧线为 a_1)、暂存单元 Storage_Node2 和逆向处理器 BackwardProcess_Arc3/4(节点 2 的后续弧线为 a_3 和 a_4)都为封装子系统。

由于节点 2 仅有前导弧线 a_1，故为信息正向输出配置输入端口 ForA1，为逆向输出配置输出端口 BackA1。同时，其后续弧线为 a_3 和 a_4，则为 a_3 配置逆向输入端口 BackA3 和正向输出端口 ForA3，为 a_4 配置正向输出端口 ForA4。没为 a_4 配置逆向输入端口是因为其终端节点 4 为网络终点，信息不会在 a_4 中逆向传输。

下面，将分别介绍信息暂存单元、正向处理器和逆向处理器的建模。

2.2.1 信息暂存单元

信息暂存单元模型如图 3。其中，Path Combiner 为所有输入的实体提供输入端口；FIFO Queue 为实体暂存器；Get Attribute 用以获取实体所携带的信息(属性名为 Message)，并通知 Events-Based Entity Generator 生成一个新实体，由 Set Attribute1 则将 Message 值赋给新实体。旧实体经 Entity Sink 离开网络，不再传输。新实体在进入后续处理之前，由 Set Attribute2 告之其后续弧线(属性名为 PostArc)和前导弧线(属性名为 PreArc)。以节点 2 为例，其所到达的实体，PreArc=[1]，PostArc=[3 4]。

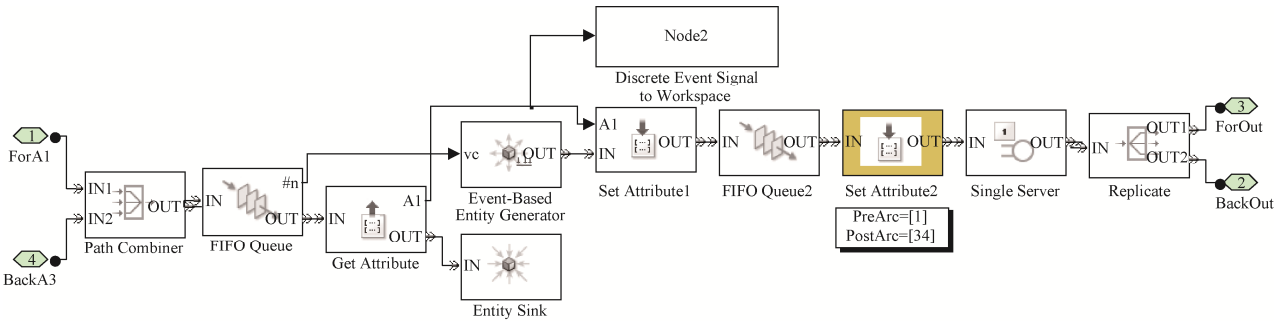


图 3 信息暂存单元

此外，它有 2 个实体输入端口，ForA1 为前导弧线 a_1 到节点 2 的输入端口，BackInA3 为后续弧

线 a_3 的逆向输入端口；有 2 个实体输出端口，ForOut 和 BackOut 分别用于向信息正(逆)向处理器输出实体。

2.2.2 正向处理器

(1) 信息正向处理流程

设 M_i 为节点 i 的信息集；$m_r \in M_i$，为其中的第 r 条信息；OA_i 为节点 i 的后续弧线集；$a_k \in OA_i$，k 为节点 i 第 k 条后续弧线的编号；OV_i 为节点 i 的后续节点集，且有映射 $\psi(a_k)=(i,j_k)$；n_{M_i} 为 M_i 的容量，n_{OA_i} 为 OA_i 的容量。

基于前述的信息处理规则，在 SimEvents 中，信息在正向处理器中的处理流程如图 4 所示。其中，MC 用以记录 m_r 离开节点 i 后是否被改写，若未被改写，$MC=0$，否则 $MC=1$；CA 为一个向量，用以记录 m_r 在哪些弧线中曾被改写。

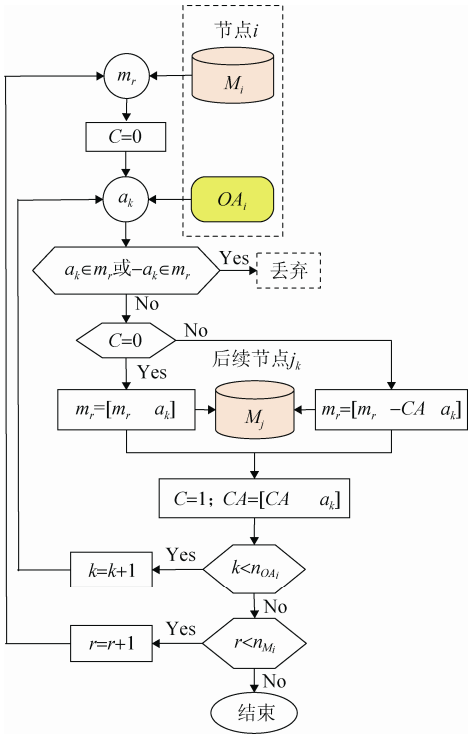
(2) 正向处理器建模

以节点 2 为例，正向处理器建模见图 5(a)，其中：

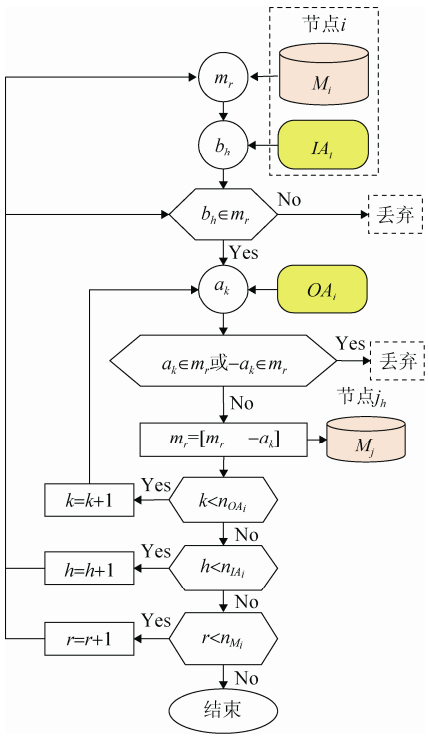
1) Set Attribute 将初始化实体的 ChangedInArc，TransArcNumber 和 ValidTransArc 3 个属性，各属性的功能见表 1。

2) Forward_Arc3 为弧线 a_3 的正向处理单元，为一个封装子系统，其构成见图 5(b)。经其处理的实体，一方面从输出端口 toA3 传至 a_3，另一方面经 toNextArc 传入 OA_2 中的下一条弧线 a_4。由于 a_4 是 OA_2 中的最后一条弧线，其 toNextArc 与一个 Entity Sink 相连。根据按改进后的信息处理规则，所有弧线的正向处理单元结构一致，不需修改任何参数而直接复制，降低了建模的难度和工作量。

3) 图 6 中，Attribute Function 用于按既定规则对信息进行处理，并对属性进行初始化和改写。Replicate 将提供两条输出路径，一是经 OUT1 由 Get Attribute 获取实体所携带的信息，并由 Output Switch 判断其流向。如果信息在该弧线被改写(Changed=1)，则经 OUT1 流入后续节点；否则，经 OUT2 被 Entity Sink 丢弃；二是传入 OA_i 中的下一条弧线。



(a) 正向传输时

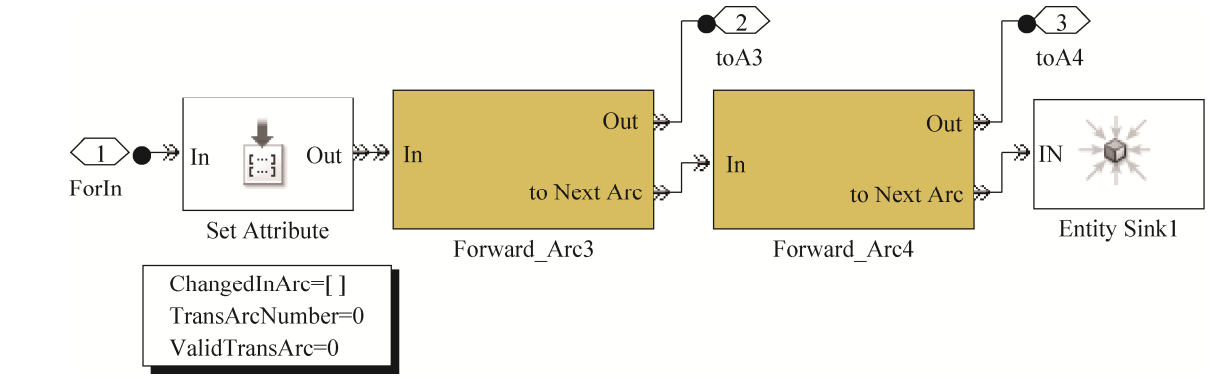


(b) 逆向传输时信息的处理流程

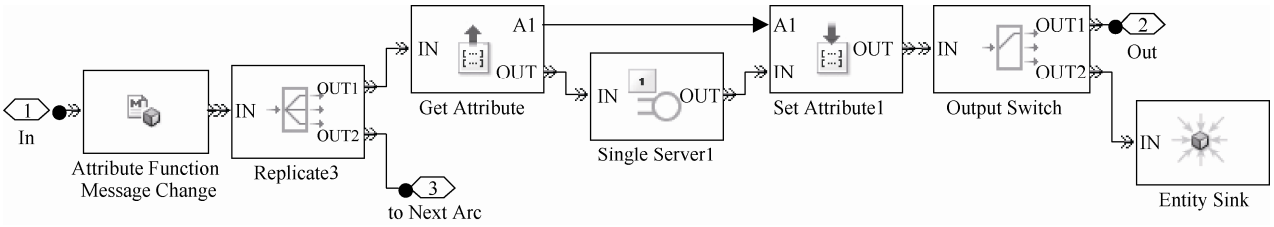
图 4 信息的处理流程

表 1 正向处理时属性的初始化和改写规则

属性	变量	功能	属性值
	a	当前弧线编号 a_k	a=PostArc(t+1)
TransArcNumber	T	实体所经过的弧线的计数	初始值为 0，每流经一条弧线加 1
ChangedInArc	CA	记录信号流经哪些弧线时被改写	初始值为[ ]，若信号被改写，CA=[CA a]
Changed	C	记录信号流经本条弧线时是否被改写	是，C=1；否，C=2
ValidTransArc		记录信息从节点 i 流出后，被改写的次数	初始值为 0，每改写一次加 1



(a) 节点 2 的正向处理器



(b) 弧线 a_3 的正向处理单元

图 5 正向处理器

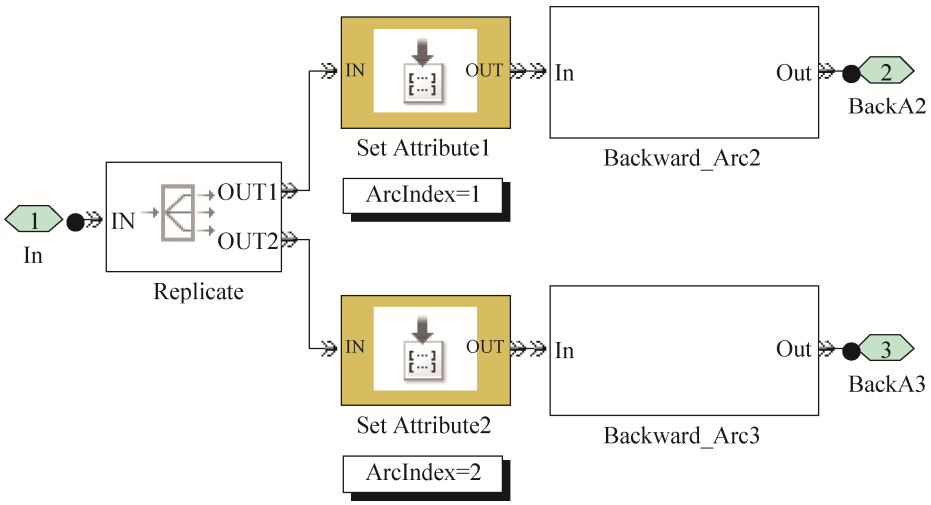


图 6 逆向处理器

第 28 卷第 4 期 系统仿真学报 Vol. 28 No. 4
2016 年 4 月 Journal of System Simulation Apr., 2016

2.2.3 逆向处理器

(1) 信息逆向处理流程

在前述假设的基础上，设 IA_i 为节点 i 的前导弧线集；$b_h \in IA_i$，为节点 i 第 h 条前导弧线；IV_i 为节点 i 的前导节点集，且存在映射：$\psi(b_h)=(j_h,i)$，$j_k \in IV_i$；n_{IA_i} 为 IA_i 的容量。M_i 中的每一条信息的逆向处理流程如图 4 所示。

(2) 逆向处理器建模

为节点 i 的 IA_i 建立一个被封装的逆向处理器。以节点 3 的逆向处理器为例(图 6)，其中：

① Replicate 为每一条前导弧线分别提供一个端口。

② Set Attribute 为每一个前导弧线提供一个编号，并以向属性 ArcIndex 赋值的形式告知到达的实体。

③ Backward_Arc 为信息的逆向处理单元，为一个封装子系统。其中包括：一个 Attribute Function，用于按逆向处理规则对各属性进行初始化和改写；一个 Output Switch，用以根据 Message Changed 判断实体流向。

按照图 4 所示信息逆向处理流程，所有弧线的逆向处理单元的结构完全一样，不需修改任何参数，可直接复制。逆向处理器中属性的初始化和改写规则见表 2。

表 2 逆向处理器中属性的初始化和改写规则

序号	属性名	变量	功能	属性值
1		b	当前前导弧线 b_h	b=PreArc(AI)
2	MessageChanged	MC	记录信号是否被改写	初始值为 2，如果信号被改写，MC=1。
3	ArcIndex	AI	实体流入的是 IA_i 中的第几条弧线	将 IA_i 中的弧线依次编号为 1，2，…。

2.3 网络源点及终点

2.3.1 源点模型

网络源点(节点 1)除了含有一个信息暂存单元(参见图 3)和一个信息正向处理单元(参见图 6)之外，需要一个实体生成模块 Time-Based Generator 和赋值模块 Set Attribute。Time-Based Generator 用于在仿真初始生成一个不携带任何信息的实体；Set Attribute 则给该实体赋上一个空信息。由于源点是信息逆向传输的终点，信息从该节点出发，只会正向传输，不再逆向传输，因此源点中不含有逆向处理单元。

2.3.2 终点模型

终点只接受信息，不再输出信息，因此不需要生成新的实体。但需要一个 Path Combiner，用于接收自前导节点输入的实体；一个 Get Attribute 获取实体携带的信息，并由 Discrete Event Signal 输出到 MATLAB 工作空间。仿真结束后到达空间的全部信息即网络的不交化 MPs。

2.4 仿真结果分析

由于不存在连续状态，因此仿真所用 Solver 选用 Variable 类中的 Discrete Solver，最大步长设为 Auto。同时，由于一个仿真周期即可获得所有的不交化 MPs，且模型中所有 Single Server 时长都为 0，因此仿真时长只要大于 0 即可。

网络模型运行耗时为 0.13s。在仿真过程中，各个节点暂存的信息如表 3 所示，而在仿真结束后经节点 4 输出到 MATLAB 工作空间中的信息有 5 条，分别对应 $\{a_1,\overline{a}_3,a_4\}$，$\{\overline{a}_1,a_2,a_5\}$，$\{a_1,a_3,a_5\}$，$\{a_1,a_3,a_4,\overline{a}_5\}$ 和 $\{a_1,a_2,\overline{a}_3,\overline{a}_4,a_5\}$。SimEvents 中，属性不支持“String”类型，故将 m_r 中的元素 a_1，a_2，a_3，…用 1，2，3，…表示，$\overline{a}_1$，$\overline{a}_2$，$\overline{a}_3$，…用 -1，-2，-3，…；信息 m_r 以向量表示。经卡诺图验证，其确为网络的不交化 MPs。初步验证了基于 DES 对网络不交化 MPs 算法进行实现的可行性和信息处理规则的正确性。

表 3 仿真过程中各节点暂存的信息

仿真时钟	节点中的暂存信息			
	1	2	3	4
0	[ ]			
1		[1]	[-1 2]	
2	[1 -3 -4] [-1 2 -5]		[1 3]	①[1 -3 4] ②[-1 2 5]
3		[1 3 -5]	[1 2 -3 -4]	③[1 3 5]
4	[1 3 -4 -5]			④[1 3 4 -5] ⑤[1 2 -3 -4 5]
5	[1 2 -3 -4 -5]		[1 2 3 -4 -5]	

3 复杂网络举例

下面将按照上述思路，在 SimEvents 环境中，以 DES 为手段获取图 7 所示的具有 8 个节点、13 条弧线的网络的不交化 MPs。该网络共有 11 条最小路，其不交化过程相对于桥型网络而言，要复杂得多。

网络模型的运行总耗时为 0.79 s，通过 DES 仿真自动获取的不交化 MPs 共计 44 条(参见表 4)。经逐一核对，其确为该网络的不交化 MPs。

需要说明的是：节点后续弧线集 OA_i 中弧线的排序会改变信息传输网络的结构，并进而影响所生成的不交化 MPs 的表现形式，故在实验中，将 OA_i 中的弧线按升序排列，表 4 所示也是基于该排序原则获取的不交化 MPs。

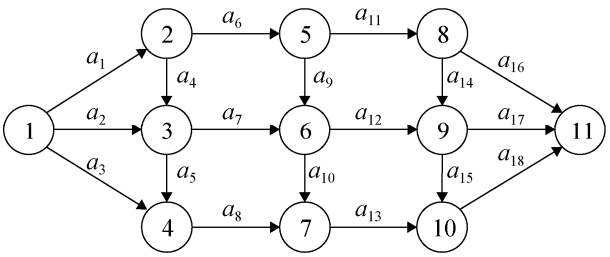


图 7 示例网络

表 4 基于 DES 获取的复杂示例网络的不交化 MPs

序号	不交化最小路	序号	不交化最小路	序号	不交化最小路	序号	不交化最小路
1	[1 -4 6 -9 11]	12	[-1 2 -5 7 10 -13 12]	23	[1 4 5 8 13]	34	[1 4 5 -8 -7 6 9 -10 -12 11]
2	[-1 2 -5 7 -10 12]	13	[-1 2 -5 -7 3 8 13]	24	[1 4 5 -8 7 -10 12]	35	[1 4 -5 -7 -6 -2 3 8 13]
3	[-1 -2 3 8 13]	14	[1 4 -5 -7 6 9 -10 12]	25	[1 4 5 -8 -7 6 -9 11]	36	[-1 2 5 -8 7 10 -13 12]
4	[1 -4 6 9 -10 12]	15	[1 4 -5 7 10 13]	26	[1 4 -5 -7 6 9 10 13]	37	[1 4 -5 -7 6 9 10 -13 -12 11]
5	[-1 2 -5 7 10 13]	16	[-1 2 5 -8 7 -10 12]	27	[1 -4 -6 2 -5 7 10 -13 12]	38	[1 4 5 -8 7 10 13]
6	[1 4 -5 7 -10 12]	17	[1 4 -5 -7 6 9 -10 -12 11]	28	[1 -4 -6 2 5 8 13]	39	[1 -4 -6 2 5 -8 7 10 13]
7	[1 -4 6 9 -10 -12 11]	18	[1 -4 -6 -2 3 8 13]	29	[1 -4 -6 2 5 -8 7 -10 12]	40	[1 4 5 -8 7 10 -13 12]
8	[-1 2 5 8 13]	19	[1 -4 6 9 10 -13 12]	30	[-1 2 5 -8 7 10 13]	41	[1 4 5 -8 -7 6 9 10 13]
9	[1 -4 -6 2 -5 7 -10 12]	20	[1 -4 -6 2 -5 7 10 13]	31	[1 4 5 -8 -7 6 9 -10 12]	42	[1 -4 -6 2 5 -8 7 10 -13 12]
10	[1 4 -5 -7 6 -9 11]	21	[1 4 -5 7 10 -13 12]	32	[1 -4 -6 2 -5 -7 3 8 13]	43	[1 4 5 -8 -7 6 9 10 -13 12]
11	[1 -4 6 9 10 13]	22	[1 -4 6 9 10 -13 -12 11]	33	[1 4 -5 -7 6 9 10 -13 12]	44	[1 4 5 -8 -7 6 9 10 -13 -12 11]

4 结论

本文在深入分析已有网络不交化 MPs 算法原理的基础上，将不交化 MPs 的获取过程视为信息在网络中的传递过程，在对算法进行进一步改进的基础上，基于其分布协同预算的特点，提出基于 DES 对算法进行实现的思路。在 SimEvents 中对示例网络进行建模和运算的实验表明：基于 DES 可实现网络不交化 MPs 的自动生成，可提高网络尤其是复杂大规模网络不交化 MPs 计算的效率和可靠性。在建模过程中，对节点和弧线的功能和模块实施封装，可实现子系统重用，使得网络建模更简单；同时，模型还具有结构直观，运行高效等特点。对于示例网络的运算结果的验证说明了改进后的算法规则的正确性，以及基于 DES 自动生成网络不交化 MPs 的可行性。

本文详细介绍了算法在 SimEvents 中实现的基本思路，可为算法在其他离散事件仿真平台，如 Witness，JMT，GRASP 等中的实现提供参考。

参考文献:

[1] Chen Y, Ren Y, Liu L L, et al. A new algorithm of GO methodology based on minimal path set [J]. AASRI Procedia (S2212-6716), 2012, 3(10): 368-374.

[2] Dohmen K. An improvement of the inclusion-exclusion principle [J]. Archiv der Mathematik (S0003-889X), 1999, 72(4): 298-303.

[3] Grzegorzewski P. The inclusion–exclusion principle for IF-events [J]. Information Sciences (S0020-0255), 2011. 181(3): 536-546.

[4] Jane C C, Yuan J. A sum of disjoint products algorithm for reliability evaluation of flow networks [J]. European Journal of Operational Research (S0377-2217), 2001, 131(3): 664-675.

[5] Xing J, Feng C G, Qian X M, et al. A simple algorithm for sum of disjoint products [C]// Reliability and Maintainability Symposium (RAMS) (S0149-144X), 2012 Proceedings-Annual, Reno, NV, USA. 2012: 1-5.

[6] Yeh W C. An improved sum-of-disjoint-products technique for the symbolic network reliability analysis with known minimal paths [J]. Reliability Engineering & System Safety (S0951-8320), 2007, 92(2): 260-268.

[7] Bjorkman K. Solving dynamic flowgraph methodology models using binary decision diagrams [J]. Reliability Engineering & System Safety (S0951-8320), 2013, 111(10): 206-216.

[8] Singhal M. Binary decision diagram based reliability evaluation [J]. International Journal of Engineering and Advanced Technology (S2249-8958), 2012, 2(2): 618-626.

[9] Ahmad S H. Simple enumeration of minimal cutsets of acyclic directed graph [J]. Reliability, IEEE Transactions on (S0018-9529), 1988, 37(5): 484-487.

[10] 史玉芳, 陆宁, 李慧民. 基于改进的不交化最小路集的网络系统可靠性算法 [J]. 计算机工程与科学, 2011, 33(1): 31-35.

[11] Yeh W C. A modified universal generating function algorithm for the acyclic binary-state network reliability [J]. Reliability, IEEE Transactions on (S0018-9529), 2012, 3(8): 554-563.

[12] Levitin G. Reliability evaluation for acyclic consecutively connected networks with multistate elements [J]. Reliability Engineering & System Safety (S0951-8320), 2001, 73(2): 137-143.

[13] 武小悦, 沙基昌. 构造网络不交化最小路集的一种新算法 [J]. 系统工程理论与实践, 2000, 27(1): 62-66.

[14] Rouchdy Y, Cohen L D, Pascual O, et al. Minimal path techniques for automatic extraction of microglia extensions [J]. Int J Comput Vis Biomech, 2011, 4(1): 35-42.

[16] Bala A, Osais Y. Modelling and simulation of DDOS attack using Simevents [J]. International Journal of Scientific Research in Network Security and Communication (S2321-3256), 2013, 1(2): 5-14.

[17] Clune M I, Mosterman P J, Cassandras C G. Discrete Event and Hybrid System Simulation with SimEvents [M]. New York, USA: IEEE, 2006.

(上接第 841 页)

[14] Shangfeng Mo, Hong Chen, Yinglong Li. Single Attribute Join Queries within Latest Sampling Periods in Sensor Networks [C]// The Seventeenth IEEE Symposium on Computers and Communications, ISCC 2012. USA: IEEE, 2012.

[15] D Zeinalipour Yazti, Z Vagena, D Gunopulos, et al. The threshold join algorithm for top-k queries in distributed sensor networks [C]// Proceedings of the 2nd International Workshop on Data Management for Sensor Networks (DMSN'05), Trondheim, Norway. Germany: Springer Berlin Heidelberg, 2005: 61-66.

[16] 回春立, 崔莉. 无线传感器网络中基于预测的时域数据融合技术 [J]. 计算机工程与应用, 2007, 43(21): 121-125.

[17] 吕效国, 王金华, 马阿芹, 等. 自回归模型的自回归分析及其应用 [J]. 扬州大学学报, 2010, 13(3): 31-33.

[18] Samuel Madden, Michael J Franklin, Joseph Hellerstein, et al. TAG: a Tiny Aggregation Service for Ad-Hoc Sensor Networks [C]// Proc. Usenix Fifth Symp. Operating Systems Design and Implementation (OSDI '02), Boston, Massachusetts, USA. USA: USENIX Association, 2002: 131-146.