基于超图的Cyber空间故障传播研究方法

王友俊 赵耘田 张红旗 张传富 杨 超
(解放军信息工程大学 郑州 450001) (河南省信息安全重点实验室 郑州 450001)

摘 要 针对现有的基于相连链路联系的故障研究方法不能准确描述Cyber空间因资源发生故障而造成的影响这个问题,提出了一种从共享任务联系的角度分析资源故障传播与扩散问题的研究方法。其基本思想是利用Petri网建模方法描述Cyber空间中任务与资源的基本关系,基于超图理论构建任务-资源模型。定义了传播能力、传播系数、扩散系数等指标来刻画故障的传播与扩散过程,采用邻接矩阵分析并推导出了这些指标的量化方法,在此基础上给出了资源故障在网络中产生的影响力的函数,最后利用案例验证了资源的故障传播和扩散过程。

关键词 Cyber空间,超图,故障传播,故障扩散,故障影响力
中图法分类号 TP393.08 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.3.028

Method for Cyberspace Failure Spreading and Diffusion Based on Hypergraph

WANG You-jun ZHAO Yun-tian ZHANG Hong-qi ZHANG Chuan-fu YANG Chao
(The PLA Information Engineering University, Zhengzhou 450001, China)
(Henan Province Key Laboratory of Information Security, Zhengzhou 450001, China)

Abstract Due to inaccurate evaluation of the impact on Cyberspace resource failure caused by the current link-based methods for failure spreading, a mission based method was proposed to analyze failure spreading and diffusion in Cyberspace. This article analyzed basic relations between missions and resources by utilizing the Petri net methodology, and built a task/resource model based on the hypergraph theory. This model defines concepts of spreading ability, spreading coefficient, diffusion coefficient and fault influence, which can be conducted and quantified by adjacent matrix. On this basis, the influence function of resource failure was derived. Finally, we tested process of failure spreading and diffusion in a case study.

Keywords Cyberspace, Hypergraph, Failure spreading, Failure diffusion, Failure influence

1 引言

Cyber空间中的多种资源在支持用户行为和保障应用任务正常运行方面起到了重要作用,但由恶意攻击或系统失灵导致的资源故障也为Cyber空间安全管理带来了许多问题。局部发生的小规模资源故障往往会在网络中向其它资源进一步传播或扩散,导致故障的影响力逐渐扩大,级联式的故障传播[1]最终可能引发Cyber网络"雪崩式"瘫痪。因此Cyber空间资源故障传播分析成为了Cyber空间安全评估与防护研究中的重要内容。

资源之间的故障传播依赖于它们之间存在的关联关系。传统网络中的故障分析方法认为资源之间主要是通过直接相连的链路发生关联(相连链路联系),假设存在相联链路的资源就可以进行故障传播。基于相连链路联系的研究方法采用多种模型对传统网络故障传播进行了研究,攻击树、攻击图等模型刻画了故障的传播路径,多种传染病模型描述了故障在网络中的扩散效果。

随着传统网络向Cyber空间的转变,资源之间的关联关系已经发生了重大变化,资源之间不再仅限于通过链路产生联系。用于Cyber空间资源关系描述的分层建模方法[2,3]将空间中的多种实体进行了映射,提出了一种包含设备层、服务层和任务层的框架来描述Cyber空间中的资源关系,指出Cyber空间中的资源对任务具有支撑关系,资源运行状态和受到的威胁与其参与的任务紧密相关[4,5]。资源通过参与相同的任务建立的联系(共享任务联系)是研究Cyber空间资源故障传播问题不可忽略的方面。

基于相连链路联系的研究思路假设只要资源之间存在通路,就可以进行故障传播。然而,如果资源处于"不活跃"状态(设备可能没有开机或者对应的服务没有在运行),即未被调用到任务执行中时,则资源与资源之间即使存在链路,也无法进行故障传播。Cyber空间中的资源一般只有在参与任务时才处于"活跃"状态,此时可以进行故障传播。只考虑相连链路联系的研究思路在应用于Cyber空间资源故障传播研究时没有考虑任务状态对资源故障传播的影响,会出现故障误定

到稿日期:2015-10-01 返修日期:2015-12-02
王友俊(1990-),男,硕士生,主要研究方向为Cyber空间建模、网络信息安全,E-mail:daxialwang@163.com;赵耘田(1989-),女,硕士生,主要研究方向为Cyber空间建模、网络通信;张红旗(1962-),男,博士,教授,主要研究方向为网络信息安全、计算机应用;张传富(1973-),男,博士后,主要研究方向为计算机建模与仿真技术;杨 超(1988-),男,博士生,主要研究方向为Cyber空间建模、网络信息安全。

位、传播难以控制等问题。

综上所述，基于相连链路联系的故障传播方法应用于Cyber空间中的资源故障传播研究时受到了局限：(1)只考虑了资源的可达性，忽略了资源与任务的实时关联性；(2)没有考虑到资源处于“不活跃”状态时无法进行故障传播的情况。

针对以上不足，本文提出了一种基于共享任务联系的Cyber空间资源故障传播问题研究方法。第2节通过对任务与资源之间存在的关联关系的分析建立了任务-资源超图模型；第3节界定了资源故障在传播与扩散过程中的多个指标并推导出这些指标的量化方法；第4节结合实例验证了该方法的可行性。

2 基于超图的任务-资源模型

Cyber空间实体包括了物理实体和虚拟实体。物理实体是指物理设备、多种软件、计算机应用服务等资源实体；虚拟实体主要是指被人为赋予意义的实体，如业务、作业、任务等，在本文中以任务作为虚拟实体的称谓。因此可以认为Cyber空间是一个任务空间。任务是在多个资源的支撑下运行的，一个资源可能会参与多个任务的运行[4]。研究Cyber空间故障传播需要首先构建能够准确反映任务与资源关联关系的任务-资源模型。

2.1 Cyber空间中的任务-资源关系分析

Cyber空间衍生自网络，是对目前多种网络构成的任务空间的一个概念性描述。Cyber空间中的网络可分为以下几种：(1)资源-资源网络，其本质是信息在物理实体中的信息流动；(2)任务-任务网络，主要研究基于工作流的任务逻辑关系；(3)任务-资源网络，任务与资源之间以集合方式进行关联。

不同于资源-资源网络或任务-任务网络，任务-资源网络是由两种类型的实体构成的异构网络，其中运行的资源是通过参与共同的任务建立联系的。Cyber任务与包含的资源之间存在集合关系，通常一个任务会包含若干资源，资源对任务具有隶属性，而一个资源也可能会参与到一个或多个任务中，任务与资源之间耦合紧密。

采用Petri网方法对任务和资源之间的关系进行分析，主要分为3个步骤：

(1)对系统中的任务进行划分和描述，包括完成任务包含的作业序列；

(2)确定作业运行过程中需要的资源，对每个资源的传入/传出进行分析；

(3)完成任务与资源之间的关系映射。

任务-资源网络可以用流程图的方式表示：用圆圈 P_i 代表作业库所，用令牌(库所中的小圆点)表示作业库所中拥有的Cyber空间资源，作业之间的逻辑关系为变迁，即 t_i。

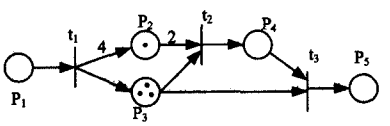

图1 任务-资源Petri网络

在图1所示的任务-资源Petri网络中，一个任务由5个作业通过特定的逻辑变迁完成，资源托肯出现的次数越多表明该资源在任务中越重要。

任务-资源网络是一个由网络的网络组成的包含异构实体的复杂网络，任务实体和资源实体无法用一个简单图涵盖，传统基于图的网络结构建模方法在构建Cyber空间任务-资源模型方面受到了局限[6]，需要寻找适用于由异构实体组成的多层网络的建模方法。

2.2 任务-资源超图构建

由传统图论拓展而来的超图理论[7]适合于对多层面、异构性隶属网络的分析，可以定量地表示具有集合关系的异构实体的关联，适用于任务-资源网络的建模。

任务是由作业序列组成的，一个作业可以看成是多个资源协作完成的。我们假设一个作业运行从开始到结束，其包含的资源是稳定不变的，即一个资源在某个作业中只可能出现一次，而一个任务包含了在不同时间序列上的多个资源，因而会出现一个资源在任务执行过程中多次出现的情况。

将资源在任务中出现的次数作为权重，建立以任务为超边、资源为节点的带权任务-资源超图。

定义1(任务-资源超图) $H=(S,M,C)$

(1)$S=\{s_1,s_2,\cdots,s_g \mid g\geqslant 1\}$为一个有限集，其中的元素 $s_i\in S(g\geqslant i\geqslant 1)$对应于Cyber空间中编号为 i 的资源，称为超图的节点，$g=|S|$称为超图的阶，即资源的数量。

(2)$M=\{m_1,m_2,\cdots,m_h \mid h\geqslant 1\}$为集合 S 上的一个有限子集，其中的元素 $m_j\in M$ 对应于编号为 j 的任务，称为超图的超边，并且满足：

1)$m_j\subset S(m_j\neq\varnothing,1\leqslant j\leqslant h)$，任务超边是由资源集合中的一个或多个资源组成的；

2)$\bigcup_{j=1}^{h} m_j=S$，建立的超图中的所有超边的元素集合即为资源的集合；

3)$s_i\in m_j$ 表示资源节点 s_i 为任务超边 m_j 的一个元素，表示该资源参与了任务 m_j 的运行；

4)若 $s_i,s_k\in m_j$，则称节点 s_i 与节点 s_k 在任务 m_j 中相邻，记为 $s_i\overset{m_j}{\leftrightarrow}s_k$，若不存在任务超边使得节点 s_i 与节点 s_k 相邻，记为 $s_i\overset{\varnothing}{\leftrightarrow}s_k$；

5)若 $m_i\cap m_j\neq\varnothing$，则称任务超边 m_i 与 m_j 相邻，记为 $m_i\leftrightarrow m_j$，相邻的超边存在共同的资源节点；

6)对于 $\forall s_i,\exists m_j$ 使得 $s_i\in m_j$，即不存在一个没有参与任何任务的孤立资源。

(3)$C=\{c_{ij} \mid s_i\in m_j,1\leqslant i\leqslant g,1\leqslant j\leqslant h\}$表示资源在任务中被调用的次数的集合，$c_{ij}$ 为节点 s_i 隶属于超边 m_j 的权值，若 $s_i\in m_j$ 则 c_{ij} 的值不为0，否则记为0。

(4)资源节点 $s_i\in S$ 由资源编号、故障发生概率属性描述，表示为 $s_i=(Id,Fail)$：

1)$s_i\_Id=i$ 表示资源的序号；

2)s_i_Fail 是资源自身故障的概率，$0\leqslant s_i\_Fail\leqslant 1$，$s_i\_Fail$ 越大，资源 s_i 发生故障的概率越大，s_i_Fail 是资源的固有属性，表示没有外界因素的情况下的故障概率。

(5)任务超边 $m_j\in M$ 由任务编号、安全属性和容错属性共同描述，即 $m_j=(Id,Sec,Tor)$：

1)$m_j\_Id=j$ 唯一标识任务序号；

2)m_j_Sec 表示任务的安全状态，在不同安全级别的任务中运行的资源发生故障的概率不同，$0\leqslant m_j\_Sec\leqslant 1$，$m_j\_Sec$ 值越大，安全级别越高，资源故障概率越小，反之越大，当 $m_j\_$

$Sec=1$ 时，资源故障不会发生传播；

3）m_j_Tor 表示任务的容错能力，容错能力会限制故障从其它任务中的资源节点向该任务中的资源节点进行传播，$0\leqslant m_j\_Tor\leqslant 1$，$m_j\_Tor$ 值越大，容错能力越强，当 $m_j\_Tor=1$ 时表示其它任务资源的故障无法向该任务中的资源进行传播。

(6)任务-资源超图 H 可以用集合或者矩阵进行表示，矩阵表示可以将元素关系表现得更加明确，本文研究资源故障传播情况，需要清晰地刻画资源与任务的关系以及资源与资源的关系，因此用矩阵对任务-资源超图进行描述。超图 H 对应矩阵 E_H：

1）E_H 中的 h 行分别对应于任务超边 $m_1,m_2,\cdots,m_h$，g 列分别对应资源节点 $s_1,s_2,\cdots,s_g$；

2）用 $E_H[i,j]$ 表示矩阵 E_H 中第 i 行、第 j 列的元素，$E_H[i,j]=c_{ij}$。

(7)由于不可达的资源之间不会存在故障传播，任务-资源超图模型只对具有可达性的资源进行研究，即任务-资源超图对应的矩阵 E_H 是连通的。

3 Cyber 资源故障量化方法

3.1 故障传播与扩散中的基本定义

资源发生故障后，首先向参与相同任务的资源进行传播，进一步在任务-资源超图中通过任务超边的联系扩散给其它资源。某个资源发生故障会对网络造成什么样的影响一直是网络规划和管理人员十分关心的问题，对故障影响力的计算可以指导风险评估、网络安全防护工作的进行。在上文建立的任务-资源超图模型的基础上，以图 2 中资源与任务的关系为例，定义以下基本概念来描述资源进行故障传播与扩散的过程。

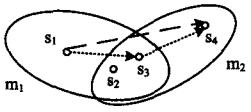

图 2 资源的故障传播与扩散

定义 2(故障传播) 若资源 s_i 出现故障，并且有 $s_i \overset{m_j}{\leftrightarrow} s_k$，则资源 s_i 会将故障向 s_k 传递，称这个过程为资源的故障传播。如图 2 所示，任务 m_1 中的资源 s_1 发生故障，导致 s_3 出现故障，任务 m_2 中 s_3 发生故障导致 s_4 也出现故障，这些都属于故障传播。

定义 3(传播能力) 隶属于某个任务的资源在发生故障后向该任务中其他资源进行故障传播的概率，传播能力与资源自身属性和包含它们的任务的属性相关。

定义 4(传播系数) 传播系数表示资源向特定资源进行传播故障的概率，传播系数与参与故障传播的两个资源的传播能力相关。

定义 5(故障扩散) 若 $s_i \overset{m_x}{\leftrightarrow} s_j, s_i \overset{m_y}{\leftrightarrow} s_j, s_i \overset{\varnothing}{\leftrightarrow} s_k$，则称资源 s_i 的故障最终导致资源 s_k 也出现故障的现象称为故障扩散。扩散是资源故障进一步传播的结果，没有参与共同任务的资源之间可以通过公共资源进行故障的传递。

定义 6(扩散系数) 扩散系数表示某个资源向其他资源进行故障扩散并成功传递故障的概率。

定义 7(扩散轨迹) 故障从资源 s_1 到资源 s_{t+1} 的扩散轨迹由一组资源节点序列 $(s_1,s_2,\cdots,s_t,s_{t+1})$ 组成，序列中没有重复节点，相邻节点属于同一个任务超边，通过共享任务产生关联，不相邻的节点属于不同的超边，从节点 s_i 到 s_{i+1} 的轨迹长度计为一步，从 s_i 到 s_{i+k} 的轨迹长度计为 k 步，并称 s_{i+k} 为 s_i 的 k 步可达资源。

定义 8(故障影响力) 资源发生故障后在任务-资源网络中造成的传播和扩散的影响的总和。

3.2 资源的故障传播与扩散

3.2.1 资源传播故障能力

资源在任务中的传播能力是反映 Cyber 空间资源与任务关系的重要指标，资源的故障传播能力与资源自身发生故障的概率、所属任务的安全属性及其在任务中出现的次数相关。同一个资源参与不同的任务时，其传播能力可能不一样，同一个任务中的不同资源传播能力各异。根据建立的任务-资源超图可知，当资源在任务执行过程中多次出现时，其传播故障的机会越大，相应地传播能力越强。用函数 $\alpha(s_i,m_j)$ 来表示资源 s_i 在任务 m_j 中的故障传播能力：

$$\alpha(s_i,m_j)=s_i\_Fail*(1-m_j\_Sec)*\frac{E_H[i,j]}{\sqrt{\sum_{k=0}^{g-1}(E_H[k,j])^2}} \tag{1}$$

其中，$0\leqslant\alpha(s_i,m_j)\leqslant 1$。

利用关联矩阵的相关性质[8]可以计算出资源在各个任务中的传播能力矩阵 D，其中 $D[i,j]=\alpha(s_i,m_j)$。资源的传播能力越强，其向其他资源进行故障传播的概率越大，在矩阵 D 中对应的元素值越大。

3.2.2 资源故障传播系数

资源的故障传播以参与了相同的任务为前提，但仍然是在资源之间进行传播，需要根据资源的传播能力矩阵进一步计算出资源之间的传播系数矩阵。超图的邻接矩阵可以表示节点之间通过共享超边建立的可达情况，对应于任务-资源超图中可以表示资源通过参与相同的任务建立的故障可传播情况，采用传播能力矩阵的邻接矩阵表示，如式(2)所示：

$$R_H=DD^{\mathrm{T}}-I \tag{2}$$

其中，D^{T} 是矩阵 D 的转秩矩阵，I 是元素为矩阵 DD^{T} 的对角矩阵，由于故障传播发生在不同资源之间，因此不需要考虑传播系数矩阵中的对角线元素，将其置为 0。矩阵 R_H 中的元素 $R_H[i,j]$ 可以表示为：

$$R_H=\begin{cases}0, & i=j\\ \sum_{k=0}^{g}(D[i,k]\cdot D^{\mathrm{T}}[k,j]), & i\neq j\end{cases} \tag{3}$$

传播系数矩阵表达了资源之间通过任务进行直接故障传播的情况，此时故障未发生进一步扩散。非对角线元素 $R_H[i,j]$ 表示资源 s_i 与 s_j 之间的故障传播系数，即当其中一个资源发生故障后，传播给另一个资源的概率。非对角线元素值为 0 表示两个资源之间没有共同的任务超边，也就是资源之间没有直接连接关系，不会进行故障传播。

3.2.3 资源故障扩散系数

资源传播系数矩阵表示了资源之间通过共享任务超边的方式进行故障传播的情况，实际中，故障不仅发生一次传播，还会进一步扩散，在网络中形成多步故障。

如图 2 中的资源节点 s_1、s_3 和 s_4 以及任务超边 m_1 和 m_2

存在关系：$s_1, s_3 \in m_1, s_3, s_4 \in m_2$，$s_1$ 可以向 s_2 传播故障，s_2 可以向 s_3 传播故障，资源 s_1 无法直接将故障传播给资源 s_3。但由于 s_1 与 s_3 同时参与任务 m_1 和 m_2，s_1 故障仍然可以通过 s_2 的“跳板”作用将故障传递给 s_3，我们将这种通过媒介进行故障传播的情况定义为故障扩散，用传播轨迹来描述故障的扩散路径。

根据故障的扩散轨迹步长来计算故障扩散系数矩阵，称 $R_H^2 = R_H * R_H - I_2$（I_2 为 $R_H * R_H$ 的对角矩阵）为二步故障扩散系数矩阵，其中的元素如果满足 $R_H^2[i,j] \neq 0 (i \neq j)$，则表示从资源 s_i 到资源 s_j 不存在二步距离可达，$R_H^2[i,j]$ 的值为故障从 s_i 扩散到 s_j 的概率。相应地，称 R_H^k 为 k 步故障扩散系数矩阵，其中的元素 $R_H^k[i,j]$ 表示故障从 s_i 经过 k 步扩散到 s_j 的概率。

假设资源节点 s_i 通过任务 m_j 向节点 s_k 进行故障扩散，当故障从 $k-1$ 步可达资源扩散至 k 步可达资源时，k 步可达资源发生故障的概率是以 $k-1$ 步可达资源发生故障为前提的条件概率：

$$P = \sum_{r=0}^{g, r \neq i, j} (R_H^{k-1}[i,r] \cdot R_H[r,j]) \tag{4}$$

故障在隶属于不同任务的资源之间传递、扩散时，任务的容错能力会降低资源扩散概率，将条件概率与任务容错能力相结合，可以得到扩散系数矩阵中的元素值：

$$R_H^k[i,j] = \begin{cases} R_H^k[i,j] = 0, & i = j \\ \sum_{r=0}^{g, r \neq i, j} (R_H^{k-1}[i,r] \cdot R_H[r,j]), & i \neq j, R_H^{k-1}[i,j] \neq 0 \\ \sum_{r=0}^{g, r \neq i, j} \{R_H^{k-1}[i,r] \cdot R_H[r,j] \cdot (1 - m_j\_Tor)\}, & i \neq j, R_H^{k-1}[i,j] = 0 \end{cases} \tag{5}$$

由于模型只考虑资源之间通过任务相互连通的情况，不考虑孤立任务或资源节点，因此故障扩散系数矩阵是连通的。证明过程如下：

假设 $\exists s_i \in m_x, s_j \in m_y$，$s_i$ 与 s_j 的故障传递关系可以分为以下几种情况：

1）$m_x = m_y$，则 s_i 与 s_j 直接传递故障，具有可达性；

2）$m_x \neq m_y, m_x \cap m_y \neq \emptyset$，则 s_i 与 s_j 经过二步距离可达；

3）$m_x \neq m_y, m_x \cap m_y = \emptyset$，由于任务-资源超图连通，因此总是存在一条扩散轨迹 $s_i \overset{m_x}{\leftrightarrow} s_{k1}, \cdots, s_{k2} \overset{m_y}{\leftrightarrow} s_j$ 使得 s_i 与 s_j 可达。

因此故障扩散系数矩阵中任意两个元素之间可达，即矩阵是连通的。

3.3 资源故障影响力函数

假设资源故障 i 发生故障后进行了 k 步传播和扩散，其对网络中其它资源造成的影响用故障影响力函数来表示：

$$F(i,k) = f(i,1) + f(i,2) + \cdots + f(i,k) \tag{6}$$

其中，$1 \leqslant i \leqslant g, 1 \leqslant k \leqslant h$。

若 $k=1$，只发生了一步传播，只需要计算 $F(i,1)$，即故障直接传播的影响，可以根据故障直接传播系数矩阵得到。

$$F(i,1) = f(i,1) = \sum_{j=0}^{g} R_H[i,j] (i \neq j) \tag{7}$$

若 $k>1$，资源的故障发生进一步扩散。需要注意的是，计算 $F(i,k)$ 时不需要重复计算在 k 步已经发生故障的资源受到的影响。

故障影响力函数的基本计算步骤：

（1）输入故障直接传播系数矩阵 R_H 和多步故障扩散系数矩阵 R_H^k、资源 s_i 与步数 k；

（2）初始化计步器 $count=0$、扩散轨迹数组 $path[g]=0$；

（3）每扩散一步，$count+1$，并且在 $path[g]$ 数组中，将已经被计算过的故障资源在数组中对应的值置为 1，表示下一步扩散时不再计算该资源受到的影响；

（4）当 $count \leqslant k$ 时，计算第 $k-1$ 步未发生故障扩散而第 k 步发生故障的资源，得到 $f(i,k)$；

（5）根据式（6）输出 $F(i,k)$。

4 资源故障传播与扩散实例分析

基于 Cyber 空间中的一个局部军事网络对资源故障传播与扩散过程进行实例分析。如图 3 所示，网络中包含了多种复杂的网络设施，其上运行着指挥控制任务、安全监控任务、数据处理任务以及业务保障任务等，某段时间内这些任务在运行过程中只涉及到 6 个资源节点。假设通过对这些任务以及资源节点运行的历史数据进行分析可以得到任务的安全状态、任务容错能力以及资源节点发生故障的概率的评估值，如表 1 和表 2 所列。

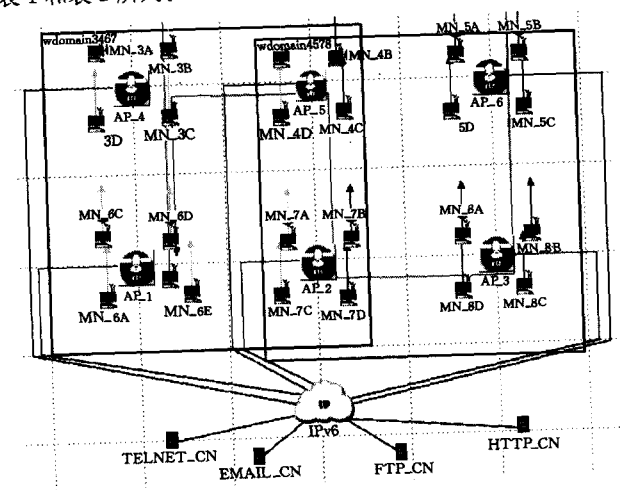

图 3 Cyber 空间中的局部军事网络

表 1 资源的故障发生概率

资源节点	故障发生概率
节点 1	0.35
节点 2	0.21
节点 3	0.28
节点 4	0.44
节点 5	0.52
节点 6	0.34

表 2 资源的故障发生概率

军事任务	网络节点	安全状态	容错能力
指挥控制任务	节点 1，节点 2，节点 3	0.5	0.3
安全监控任务	节点 2，节点 3，节点 4	0.8	0.7
数据处理任务	节点 4，节点 5	0.3	0.8
业务保障任务	节点 5，节点 6	0.4	0.5

用 $s_1, s_2, \cdots, s_6$ 表示资源节点，$m_1, m_2, \cdots, m_4$ 表示任务，资源与任务的超图对应关系如图 4 所示，可以用关联矩阵式（8）表示。

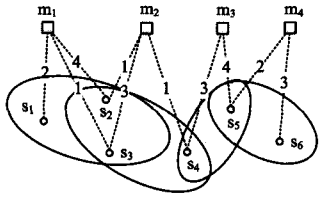

图 4　Cyber 空间中的任务-资源关联关系

$$E_H=\begin{bmatrix}2&0&0&0\\4&1&0&0\\1&3&0&0\\0&1&3&0\\0&0&4&2\\0&0&0&3\end{bmatrix} \tag{8}$$

传播能力矩阵综合考虑了资源在所属任务中的出现次数和任务的安全属性，表示了资源节点向其它节点传播故障的能力。分析可知，同一个资源节点在不同任务中表现出不同的故障传播能力，与任务的安全状态及其在不同任务中出现的次数相关。

$$D=\begin{bmatrix}0.076&0&0&0\\0.092&0.013&0&0\\0.031&0.051&0&0\\0&0.133&0.185&0\\0&0&0.291&0.173\\0&0&0&0.17\end{bmatrix} \tag{9}$$

根据式(3)可以计算出军事网络任务-资源超图 H 的资源传播系数矩阵，如式(10)所示，此时资源故障只能在参与相同任务的资源之间进行传播，例如资源 s_1 会将故障向 s_2、s_3 传播，但由于 s_1 与 s_4 没有共同参与的任务，故障不能传播到 s_4。

$$R_H=\begin{bmatrix}0&0.7&0.23&0&0&0\\0.7&0&0.34&0.17&0&0\\0.23&0.34&0&0.67&0&0\\0&0.17&0.67&0&5.38&0\\0&0&0&0.54&0&2.94\\0&0&0&0&2.94&0\end{bmatrix}\times10^{-2} \tag{10}$$

由式(5)可以计算得到多步故障扩散系数矩阵，如式(11)—式(13)所示，通过观察扩散系数矩阵可以得知资源故障扩散轨迹。

$$R_H^2=\begin{bmatrix}0&8.03&2.409&0.099&0&0\\0.803&0&2.762&0.006&7.233&0\\2.409&2.762&0&0&28.93&0\\0.823&2.313&0.578&0&0&158.1\\0&6.329&25.31&0&0&0\\0&0&0&126.5&0&0\end{bmatrix}\times10^{-5} \tag{11}$$

$$R_H^3=\begin{bmatrix}0&1.152&1.567&0.175&0.517&0\\0.6445&0&1.741&40.78&12.45&10.62\\1.934&1.783&0&156.1&3.111&42.49\\1.754&0.775&0.988&0&371.5&0\\10.337&8.712&2.178&14.46&0&0\\0&14.87&59.49&0&340.3&0\end{bmatrix}\times10^{-7} \tag{12}$$

$$R_H^4=\begin{bmatrix}0&0.083&0.157&2.906&0.944&0.76\\0.0041&0&2.755&6.814&25.06&3.656\\0.125&2.759&0&1.704&96.51&0.914\\0.077&0.157&0.068&0&0&54.57\\0.661&1.041&1.513&0.293&0&0\\2.429&2.047&0.512&187.4&0&0\end{bmatrix}\times10^{-8} \tag{13}$$

分析可知，资源故障在经过二步扩散后，资源 s_1 可能会将故障扩散到 s_4，经过四步扩散，最远可以将故障扩散到资源 s_6，扩散轨迹为：$s_1 \xrightarrow{m_1} s_3 \xrightarrow{m_2} s_4 \xrightarrow{m_3} s_5 \xrightarrow{m_4} s_6$ 或者 $s_1 \xrightarrow{m_1} s_2 \xrightarrow{m_2} s_4 \xrightarrow{m_3} s_5 \xrightarrow{m_4} s_6$。

根据式(6)可以得到资源 s_1 的四步故障扩散影响力为：

$$F(1,4)=R_H[1,2]+R_H[1,3]+R_H^2[1,4]+R_H^3[1,5]+R_H^4[1,6] \tag{14}$$

相应地，可以计算得到 6 个节点分别在不同扩散步造成的影响力，如表 3 所列。

表 3　资源的故障影响力

资源	第 1 步	第 2 步	第 3 步	第 4 步
s_1	0.0093333	0.0093525	0.0093536	0.0093536
s_2	0.0121215	0.0121938	0.0121959	
s_3	0.0124948	0.0127841	0.0127883	
s_4	0.0622138	0.063822		
s_5	0.0831898	0.0835062	0.0835072	
s_6	0.029376	0.0306407	0.0306481	0.03064813

通过对资源故障传播系数矩阵、扩散系数矩阵以及故障影响力的分析，可以得到以下结论：

(1)随着扩散步数的增大，故障的扩散系数不断下降，但是影响力逐渐增大。随着扩散步数的不断增多，资源的故障传播概率受到其他任务容错能力、目标资源的故障概率等的影响，扩散概率会逐渐降低。

(2)资源故障扩散存在多路径可达，不同扩散轨迹的传播系数不同。在资源的故障从传播演变为扩散的过程中，可能会存在多个扩散媒介，即 $|m_x \cap m_y|>1$，从扩散源到扩散目的节点可能存在多条扩散轨迹。

(3)若任务-资源超图是全连通的，则故障最终会全超图可达。尽管故障扩散的系数会随着故障传播与扩散的步数的增大而降低，但是只要资源之间存在扩散轨迹，就有可能进行故障传播或扩散。并且，Cyber 空间资源高度耦合、联系紧密，当多个资源同时发生故障时，会出现故障叠加效应，验证了 Cyber 空间中发生级联故障的情况。

结束语　Cyber 空间中资源的故障传播与 Cyber 任务之间存在着复杂的关联关系，本文从共享任务联系的角度出发，构建了基于超图的任务-资源模型，在此基础上研究了资源故障在任务-资源超图中的传播与扩散问题，并给出了资源故障传播能力、传播系数、扩散系数以及故障影响力等参数的量化方法，最后结合案例对资源故障传播和扩散过程进行了分析。在此基础上，如何对 Cyber 空间中的资源进行安全管理、以最小的安全防护代价实现最大程度控制资源的故障传播与扩散是下一步研究的重点。

参 考 文 献

[1]　Huang Zhen, Wang Cheng, Nayak A, et al. Small Cluster in Cy-

ber Physical Systems: Network Topology, Interdependence and Cascading Failures [J]. IEEE Transactions on Parallel & Distributed Systems, 2015, 26: 2340-2351

[2] Jakobson G. Mission Cyber security situation assessment using impact dependency graphs[C]// 2011 Proceedings of the 14th International Conference on Information Fusion (FUSION). IEEE, 2011: 1-8

[3] Barnett A, Smith S R, Whittington R P. Using Causal Models to Manage the Cyber Threat to C2 Agility: Working with the Benefit of Hindsight[R]. Defence Science and Technology Lab Porton Down(United Kingdom), 2014

[4] Neuman C, Tan K. Mediating Cyber and Physical Threat Propagation in Secure Smart Grid Architectures[C]// 2011 IEEE International Conference on IEEE Smart Grid Communications (Smart Grid Comm). 2011: 238-243

[5] Cheng Yi, Deng Ju-lia, Li J, et al. Cyber Defense and Situational Awareness[M]. Springer International Publishing, 2014

[6] Halappanavar M, Choudhury S, Hogan E, et al. Towards a network-of-networks framework for cyber security [C] // 2013 IEEE International Conference on IEEE Intelligence and Security Informatics (ISI). 2013: 106-108

[7] Berge C. Graphes et hypergraphes [M]. Dunod, Paris-Brussels-Montreal, Que., 1970

[8] Wang Zhong-tuo. Introduction to System Engineering [M]. Beijing: Electronic Industry Press, 2012(in Chinese)
王众托. 系统工程引论[M]. 北京：电子工业出版社，2012

(上接第 117 页)

足率高于 PS²O 算法和 PSO-RBF+PSO 算法，虚拟机迁移次数少于 PS²O 算法和 PSO-RBF+PSO 算法。

总之，PSO-RBF+PS²O 算法在迭代次数较多、节点较多、有效资源较少的情况下工作效率明显比 PSO 资源分配算法效率高。本算法在提高了云计算中心资源利用率的同时，也保证了云应用的服务质量，同时具有较高的应用性能。

结束语 本文首先设计了云计算资源调度的两级动态调度管理框架，并给出了框架系统的具体结构和形式；其次，采用 PSO-RBF 算法对应用的资源需求量进行预测，并提出了一种多目标优化算法 PSO-RBF+PS²O，该算法综合考虑了云应用的服务质量、应用性能和资源利用率；最后，在 CloudSim 平台进行了仿真，结果表明此调度模型和多目标算法能有效地提高云计算系统中应用的服务质量和资源的利用率。下一步将考虑在开源云计算平台如 OpenStack 上应用文中的调度策略，并验证相关策略的有效性。

参 考 文 献

[1] Armbrust M, Fox A, Griffith R. A view of Cloud computing[J]. Communications of the ACM, 2010, 53(4): 50-58

[2] Rivoire S, Shan M, Ranganathan P. A balanced energy-efficiency benchmark[C]// ACM SIGMOD International Conference on Management of Data. Beijing, China, 2007: 376-380

[3] Goirii I, Guitart J, Torres J. Characterizing Cloud Federation for Enhancing Providers Prot[C]// 3rd International Conference on Cloud Computing. Miami, USA, 2010: 123-130

[4] Goiri I, Juli A F, Torres J. Energy-aware Scheduling in Virtualized Data Centers[C]// 12th IEEE International Conference on Cluster Computing. Heraklion, Greece, 2010: 58-65

[5] Shi X L, Xu K. Utility Maximization Model of Virtual Machine Scheduling in Cloud Environment [J]. Chinese Journal of Computers, 2013, 36(2): 252-262(in Chinese)
师雪霖，徐恪. 云虚拟机资源分配的效用最大化模型[J]. 计算机学报，2013，36(2)：252-262

[6] Li Q, Hao Q F, Xiao L M. Adaptive Management and Multi-Objective Optimization for Virtual Machine Placement in Cloud Computing[J]. Chinese Journal of Computers, 2011, 34(12): 2256-2257(in Chinese)
李强，郝沁汾，肖利民. 云计算中虚拟机放置的自适应管理与多目标优化[J]. 计算机学报，2011，34(12)：2256-2257

[7] Tian G H, Ment D, Zhan J F. Reliable Resource Provision Policy for Cloud Computing[J]. Chinese Journal of Computers, 2010, 33(10): 1859-1872(in Chinese)
田冠华，孟丹，詹剑锋. 云计算环境下基于失效规则的资源动态提供策略[J]. 计算机学报，2010，33(10)：1859-1872

[8] Lei D M, Yan X P. Intelligent Multi-objective optimization algorithm and Application[M]. Beijing: Science Press, 2009(in Chinese)
雷德明，严新平. 多目标智能优化算法及其应用[M]. 北京：科学出版社，2009

[9] Coello C, Carlos A, Lechuga M S. MOPSO: a proposal for multiple objective particle swarm optimization[C]// Proceedings of IEEE Congress on Evolutionary Computation. Honolulu, USA: IEEE, 2002: 1051-1056

[10] Xu X B, Zheng K F, Li D. New chao-particle swarm optimization algorithm[J]. Journal on Communications, 2012, 33(1): 24-30(in Chinese)
胥小波，郑康锋，李丹. 新的混沌粒子群优化算法[J]. 通信学报，2012，33(1)：24-30

[11] Wang J Z, Chen Y J. Research on I/O Resource Scheduling Algorithms for Utility Optimization Towards Cloud Storage[J] Journal of Computer Research and Development, 2013, 50(8): 1657-1666(in Chinese)
王健宗，谌炎俊. 面向云存储的 I/O 资源效用优化调度算法研究[J]. 计算机研究与发展，2013，50(8)：1657-1666

[12] Luo J Z, Jin J H, Song A B. Cloud computing: architecture and key technologies[J]. Journal on Communications, 2011, 32(7): 2-19(in Chinese)
罗军舟，金嘉晖，宋爱波. 云计算：体系架构与关键技术[J]. 通信学报，2011，32(7)：2-19

[13] Grit L, Irwin D, Yumerefendi A. Virtual Machine Hosting for Networked Clusters[C]// Proceedings of the 2nd International Workshop on Virtualization Technology in Distributed Computing. Durham, USA, 2006: 1-5

[14] Andr H, Lagar-Cavilla S, Whitney J A. SnowFlock: virtual machine cloning as a first-class cloud primitive[J]. ACM Trans Comput Syst, 2011, 29(1): 1-45

[15] Han H G, Qiao J F, Bo Y C. On Structure Design for RBF Neural Network Based on Information Strength [J]. Acta Automatica Sinica, 2012, 38(7): 1083-1090(in Chinese)
韩红桂，乔俊飞，薄迎春. 基于信息强度的 RBF 神经网络结构设计研究 [J]. 自动化学报，2012，38(7)：1083-1090

[16] Abido M A. Optimal Power Flow using Particle Swarm Optimization[J]. International Journal of Electrical Power & Energy Systems, 2002, 24(7): 563-571