



(12)发明专利

(10)授权公告号 CN 104298593 B

(45)授权公告日 2017.04.26

(21)申请号 201410490283.2

(22)申请日 2014.09.23

(65)同一申请的已公布的文献号
申请公布号 CN 104298593 A

(43)申请公布日 2015.01.21

(73)专利权人 北京航空航天大学
地址 100191 北京市海淀区学院路37号

(72)发明人 李大庆 王飞龙 王世海 刘斌

(74)专利代理机构 北京慧泉知识产权代理有限公司 11232

代理人 王顺荣 唐爱华

(51) Int. Cl.
G06F 11/36(2006.01)

审查员 郝强

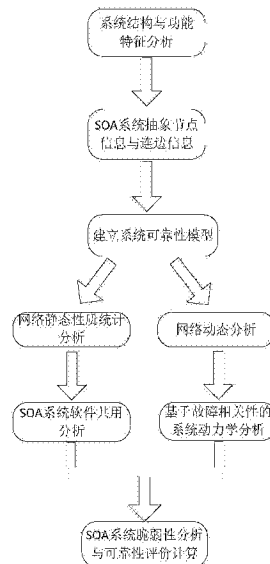
权利要求书3页 说明书9页 附图3页

(54)发明名称

一种基于复杂网络理论的SOA系统可靠性评价方法

(57)摘要

一种基于复杂网络理论的SOA系统可靠性分析方法,它有四大步骤,属于复杂系统可靠性技术领域;其目的是针对SOA系统结构功能复杂,以及传统可靠性分析方法的实施存在困难甚至无法实施,提出一套可以有效分析SOA这一复杂系统可靠性的方法论。方法的特征是:创新地结合复杂网络理论,将SOA系统中的各对象以及对象之间的相互作用抽象为复杂网络中的节点和连边,建立复杂网络失效模型;并提出了针对SOA系统的级联失效模型,在此基础上利用蒙特卡洛仿真实验完成可靠性分析工作,发现影响系统可靠性的关键软件和脆弱结构。该方法不受系统规模限制,系统规模的增大不会显著提高方法计算复杂度;物理意义清晰,易于工程人员操作实施。



1. 一种基于复杂网络理论的SOA系统可靠性评价方法,其特征在于:该方法具体步骤如下:

步骤一:进行SOA系统的结构与功能特征分析,获得系统建模需要的相关拓扑结构信息,建立拓扑结构模型;此外利用网络可视化技术,给出网络的拓扑结构图;

步骤二:由于系统的结构往往会对发生在系统中的动力学行为即故障的传播产生重要影响,所以在进行系统的故障行为分析之前需要分析系统的静态结构特征;基于复杂网络理论,对建立的可靠性模型的基本静态拓扑属性进行统计分析,包括节点入度分布、节点出度分布、软件共用分析;

步骤三:在建立的系统拓扑结构的基础上,基于复杂网络理论,建立针对SOA系统的级联失效模型,得到系统的可靠性模型,设计出适用于SOA系统可靠性评价的仿真实验方案,并完成SOA系统可靠性仿真实验;

步骤四:利用步骤三中可靠性仿真实验中的结果,对系统开展脆弱性分析和可靠性计算,给出对系统可靠性影响较大的关键软件列表并分析原因,同时给出系统的可靠性量化评估结果,为高可靠性系统的设计、规划管理和优化控制提供依据;

其中,步骤一中所述的“获得系统建模需要的相关拓扑结构信息”,具体方法如下:

系统中的节点信息以及软件与服务之间连边信息的获得:整个SOA系统根据使用方的配备分为若干子系统;用户根据需求给出服务列表,由SOA系统设计单位设计出基于用户需求的服务-软件编配表,编配表中包括SOA系统要实现的所有服务,以及实现各个服务需要调用的复数个软件;从服务-软件编配列表中提取所有的软件、服务信息,同时软件被服务调用这一信息被抽象成软件与服务之间的连边;由于一个服务的实现往往需要多个软件的共同协作,一个服务常与多个软件存在连边;

同一子系统中不同服务之间的连边信息:同一子系统中不同服务之间的连边由服务之间数据传递关系建立,并且这种连边是有向的,即如果服务A需要接收来自服务B的数据,那么服务A与服务B之间就存在一条有向边 $B \rightarrow A$;同样的一个服务需要接收来自多个其它服务的数据,所以指向该服务的边有多条;不同子系统间服务间的连边信息:不同子系统间,由于相同服务之间存在信息数据交互,所以相同服务间建立连边。

2. 根据权利要求1所述的一种基于复杂网络理论的SOA系统可靠性评价方法,其特征在于:步骤二中所述的“对建立的可靠性模型的基本静态拓扑属性进行统计分析”,具体内容与方法如下:系统软件共用次数分析:考虑到系统中相同的软件资源被配置到各个子系统各个部门中去,并且由于各部门职能的差异,系统各部门所需的服务存在巨大差别,这最终导致了系统中各个软件的使用频率或者说配置份数不同;对各个软件被不同部门的共用次数的统计,构成了系统软件共用次数分析;分析输出三个结果:各个软件的共用次数、各软件根据共用次数频率的高低排序结果、具有n次共用次数的软件的个数统计结果,其中利用第二个结果得到基于软件共用次数的关键软件列表;

服务节点软件度分布统计:根据用户的需求,每项服务会使用数目不等的软件;一个服务的软件度 k_{soft} 指为该软件配置的可供调用的不同软件的数目;

服务节点的入度分布统计:一个服务的入度 k_{in} 指的是,服务作为 k_{in} 个其它服务的数据接收方,即系统中有多少个其它服务即上游服务向它发送数据; k_{in} 越大代表这个服务对外界的依赖性越大;

服务节点的出度分布统计:一个服务的出度 k_{out} 指的是,服务需要向其它 k_{out} 个服务发送数据;一个服务的出度 k_{out} 越大,说明这个服务当前的工作状态越容易对系统中其它服务的工作状态造成影响,换句话说,该服务的故障是容易传播开的。

3. 根据权利要求1所述的一种基于复杂网络理论的SOA系统可靠性评价方法,其特征在于:步骤三中所述的“建立针对SOA系统的级联失效模型,得到系统的可靠性模型,设计出适用于SOA系统可靠性评价的仿真实验方案”的具体内容与方法如下:

级联失效模型:对于系统中的任意一个服务能接收三种类型的数据:调用其所属的软件、子系统内其它服务以及其它子系统的相同服务传送的数据;这意味着任一服务可能受来自三方面的影响,通常假设各方面对其造成的影响是独立的;系统中的软件由于自身故障可能导致错误信息的传递,这个影响概率与服务使用故障软件的概率成正比;方法假设若有一种软件故障,那么所有子系统配置的同种软件也将发生故障;在系统中的服务因软件的故障而受到影响后,由于子系统内以及子系统之间存在服务之间的数据传递,这种影响也可能继续向其所有下游服务节点传递,发生“级联效应”;同样的,服务节点的故障向其某个下游邻居服务节点传播的概率,也是由该下游服务节点使用故障服务的概率决定;其中下游服务节点包括系统内的服务和系统外的服务;三种传播概率的计算方法为:

软件 i 故障影响使用它的服务 j 的概率:

$$p_{i \rightarrow j} = 1/k_{j1} \quad (1)$$

这里的 k_{j1} 是指服务 j 拥有的软件的数目;

受影响的服务 i 影响其子系统内下游邻居服务 j 的概率:

$$p_{i \rightarrow j} = 1/k_{j2} \quad (2)$$

这里的 k_{j2} 是指子系统内向服务 j 发送数据的服务数目;

受影响的服务 i 影响其子系统外下游邻居服务 j 的概率:

$$p_{i \rightarrow j} = 1/k_{j3} \quad (3)$$

这里的 k_{j3} 是指子系统外向服务 j 发送数据的服务数目;

基于蒙特卡洛方法的可靠性仿真实验:采用不同的策略对系统中的软件进行故障注入,作为系统的初始故障,注入后故障软件根据故障影响概率向调用它的服务产生影响,使系统中部分的服务节点受到影响而发生故障;之后,在实验的每个迭代步,系统中服务节点的故障向其下游服务节点传播,在固定的迭代步结束后,统计系统中受影响的服务的个数或比例,作为此次实验中初始故障对系统影响程度的评估;节点间的故障传播使用上述级联失效模型,在仿真故障传播时,若蒙特卡洛抽样得到的随机数小于由模型计算得到的故障影响概率,则传播成功;故障注入策略包括:依次对系统中所有软件注入故障、随机选取一定比例的软件节点注入故障、由高到低选取共用频率高的软件注入故障;第一种故障注入策略进行的仿真实验输出的结果为系统中各软件的故障对系统的影响程度,第二种故障注入策略进行仿真实验输出的结果为不同比例的软件的故障对系统的影响程度,第三种故障注入策略进行的仿真实验的输出结果为高共用频次软件的故障对系统的影响程度;另外,为考察SOA系统特定的拓扑结构对系统可靠性的影响,两种对比实验方案也能实施:保持网络中的节点总数量与连边总数量不变,将网络拓扑结构随机化,随机化的方式为连边随机重连,实验观察网络可靠性的变化;保持网络中服务节点的出度不变,即各个有向边的数据发送端保持不变,随机的重新选择其接受端,实验观察网络可靠性的变化。

4. 根据权利要求1所述的一种基于复杂网络理论的SOA系统可靠性评价方法,其特征在于:步骤四中所述的“对系统开展脆弱性分析和可靠性计算”的具体内容与方法如下:

软件共用次数统计结果分析:将系统中的每个软件按照共用次数高低进行排序,得到一份包含具有较高共用次数的关键软件列表;这个关键软件列表被用来作为蓄意攻击实验中选取故障软件源节点的依据;

单一软件故障对系统影响结果分析:在评价系统中任一软件的重要度时,每次试验对一个软件进行故障注入,每个软件的故障对系统影响的程度不同;将系统中的各软件按照影响程度由高到低排序后,得到由对系统可靠运行较重要的软件组成的关键软件列表;另外,将影响程度分为四类严酷度等级,然后根据各软件故障影响严酷度等级将各软件分类,在系统可靠性优化阶段根据需要,针对某类严酷度等级以上的软件进行改进优化,以最小的成本实现系统可靠度的最大提升;

软件随机失效对系统影响结果分析:在这个试验中,每次试验从系统中随机的选取一定比例的软件作为初始故障源,利用级联失效模型进行蒙特卡洛仿真实验,观察对系统的影响程度;这个实验得到了不同比例软件故障对系统的影响结果;以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐标,做出系统对随机故障的可靠性曲线图;通过对该结果的分析评估系统的整体可靠性水平;

蓄意攻击对系统影响结果分析:在这个实验中,蓄意攻击是指,以软件共用次数统计结果为依据,针对共用次数高的软件节点进行故障注入,观察系统的最终影响;每次试验选取共用次数高的软件;同样以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐标,做出系统对蓄意攻击的可靠性曲线图;通过对该结果的分析发现系统的脆弱部分,所谓脆弱结构是指系统中任何用来作为蓄意攻击以达到对系统最大程度破坏目的的结构;

随机化的系统结构的可靠性评价:构建与原系统具有相同度均值 $\langle k \rangle$ 的随机化系统网络模型,得到随机故障对随机化系统的影响结果,以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐标,做出具有相同度均值随机化系统对随机故障的可靠性曲线图;构建与原系统具有相同度分布的随机化系统网络模型,得到随机故障对随机化系统的影响结果,同样以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐标,做出具有相同度分布的随机化系统对随机故障的可靠性曲线图;

结果综合分析:将基于参照物系统的分析结果与原系统上得到的结果进行比较分析,揭示系统失效的传播机理以及脆弱结构之所以脆弱的原因。

一种基于复杂网络理论的SOA系统可靠性评价方法

技术领域

[0001] 本发明提出一种基于复杂网络理论的SOA系统可靠性评价方法,属于软件可靠性技术领域。

背景技术

[0002] 面向服务的架构(Service-Oriented Architecture--SOA)是一种粗粒度、松耦合的软件体系架构,它将应用程序的不同功能单元即服务(service),通过服务间定义良好的接口和契约(contract)联系起来,以完成用户请求的业务。这个概念由Gartner在1996年首次提出,并认为未来SOA会成为占绝对优势的软件工程方法。但之后仅有少数如BEA, Microsoft, IBM等公司试验性的使用过这一架构方式。纵观软件业的发展历程,从最初的面向过程、面向对象,发展到时下流行面向组件、面向集成,在这个螺旋式的上升曲线中,软件开发始终都没有完全做到系统模型与系统实现相分离,开发人员在多样化的需求中承受痛苦,公司蒙受经济损失。传统软件架构以无法很好地满足不断变更的需求,需要一种新的软件架构方式来满足这种需要。正是在这样的情况下,近年来SOA再次成为Microsoft, Sun, IBM等等IT界巨头关注的焦点。而随着XML Web services技术的成熟,也给当下SOA的发展奠定了坚实的基础。

[0003] SOA系统本身往往是由数量巨大的用户、业务、服务集成的庞大系统,由于不同用户的需求多样,使得SOA系统的结构与运行机理呈现出复杂性,具体表现为:1)为完成一项用户请求的业务,系统中服务与服务之间需要并存在通信与互操作;2)根据用户的需求,业务与业务之间也存在相互协作;3)作为SOA的一大优势(节省成本),服务可以被不同业务重复使用;4)随着时间的推移,用户数量的增长以及业务范围的扩大,越来越多的服务会被加入,使得系统越来越庞大;5)若干SOA子系统可以组合为一个更为庞大的系统。目前对复杂系统进行可靠性建模并进行可靠性评价存在很大的困难,传统的系统可靠性建模与分析方法,如可靠性框图方法(RBD)、故障模式与影响分析方法(FMECA)、故障树分析方法(FTA)等受限于系统规模,往往针对系统的静态结构进行建模分析,同时很难顾及发生在系统上的复杂的动力学行为对系统造成的影响。由于许多系统及其构成对象之间的相互作用都可以用网络来描述,而复杂网络理论可以用来抽象描述复杂系统,是研究复杂系统动力学行为的有效工具。

[0004] 自从1998和1999年在Nature和Science上刊登了小世界效应与无标度特征的科学发现以来,复杂网络科学在近十年里迅速兴起,掀起了对现实世界普遍特征和规律认识的研究热潮,而且随着学科之间的相互交叉和融合趋势不断加强,也促进了对复杂网络共有特征和性质的揭示,提高了人们对现实世界的科学认识,因此它是当前最有希望用来认识“复杂系统之所以复杂”的有力工具。作为对复杂系统的抽象性描述方式和研究框架,复杂网络突出强调系统层次的复杂性特征,是复杂系统的其他属性的分析基础,因此对它的研究被认为有助于理解复杂系统的复杂性来源及其复杂程度等重要问题。

[0005] 复杂网络的可靠性已经成为人们日益关心的一个重要问题。人们做出了很多的努

力,但是复杂系统的大规模故障仍然时有发生。大范围电网是典型的复杂系统,近年来,国内外发生了多次大规模电网级联故障导致的停电事故。1998年和2003年,美国都发生了由于级联故障导致的大规模停电,其中2003年大停电影响范围多达5000多万人。此外,英国伦敦、北欧部分国家、意大利都发生了类似的重大停电事故。研究发现,复杂系统的故障通常出现较突然,大部分由看似微小的事件引起,并迅速地“蝴蝶效应”般对整个系统造成大规模影响;由于系统对象间广泛而紧密的联系,使系统各对象往往具有较强的故障相关性,造成系统内的故障极易传播;并且故障传播方式多样,没有确定性规律。近来的发现表明,级联故障是复杂网络系统的主要故障模式。复杂网络的级联故障,就是指一个或少数几个节点或边发生的故障会通过节点之间的耦合关系引起其它节点发生故障,这样就会产生连锁效应,最终导致网络中相当一部分节点的失效甚至整个网络的崩溃,也形象的称为“雪崩”。在SOA系统中,由于系统组成对象之间紧密的功能联系,系统中某一对象的故障往往会影响到系统其它对象的正常工作,形成级联效应,因此SOA系统的故障机理使得运用复杂网络方法来分析其可靠性成为必要。

[0006] 在上面的SOA系统与复杂网络理论技术应用的背景下,提出如下一种基于复杂网络理论的SOA系统可靠性评价方法。

发明内容

[0007] 对一种复杂系统—面向服务的软件架构(SOA)系统进行结构与功能特征分析,针对SOA系统结构与失效行为复杂、传统可靠性分析方法无法分析等特点,本发明提供了一种基于复杂网络理论的SOA系统可靠性评价方法。

[0008] 本发明是一种基于复杂网络理论的SOA系统可靠性评价方法,其目的是:分析提取SOA系统的相关信息,利用复杂网络技术对系统进行抽象描述,建立系统的可靠性评价模型;在建立的可靠性模型的基础上,利用复杂网络分析技术对系统进行脆弱性分析和可靠性评价,分析系统的故障机理,得到影响系统可靠性的关键软件列表,从而为提升SOA系统的可靠性提供依据。

[0009] 本发明的基本思想是:如果将SOA系统中的各个对象(包括服务与软件两类对象)抽象为网络中的节点,将对象间的相互作用(如调用、数据传送等)抽象为节点中的连边,可以建立系统的可靠性模型;在建立的可靠性模型上,基于复杂网络理论,通过不同的故障注入模式,分析对比各种节点故障方式对整个系统的影响程度(用最终系统中受影响的服务的个数来衡量),实现对系统当前可靠性水平的评估,并可得到系统关键对象(软件)列表,为系统可靠性评价提供依据。这里所谓系统关键对象是指,该类对象的失效会对系统的运行造成相对较大影响,同时通过提升该类对象的可靠性,系统的可靠性可以有效地得到改进。

[0010] 本方法首先进行SOA系统结构分析与信息提取,得到系统内服务与软件两类节点信息以及各节点之间的连边信息,并以此建立系统的拓扑结构模型;然后对建立的模型进行拓扑结构统计分析,包括节点度分布统计、软件共用次数统计等;之后基于复杂网络技术,建立针对SOA系统的级联失效模型,对系统进行脆弱性分析,得到影响系统可靠性的关键软件列表;系统的拓扑结构模型与级联失效模型共同组成了系统的可靠性模型;最后通过不同的故障注入模式,基于级联失效模型完成蒙特卡洛仿真实验,实现对系统的可靠性

评价。

[0011] 本发明提出的基于复杂网络理论的SOA系统可靠性评价方法的开展需要获得SOA系统的如下信息：

[0012] 1) 节点相关：软件数量，服务数量，软件的可靠性；

[0013] 2) 服务与软件连接：实现服务所需调用的若干软件，由软件编配列表提取得到；

[0014] 3) 服务之间连接：由于数据传递与接收产生的服务之间的依赖关系，由服务之间数据交互列表得到；

[0015] 4) 软件之间连接：SOA系统中不同软件之间的松耦合联系（软件之间可能是独立的，由系统需求决定）。

[0016] 基于上述SOA系统结构信息，本发明提出一种基于复杂网络理论的SOA系统可靠性评价方法，如图1所示，该方法具体步骤如下：

[0017] 步骤一：进行SOA系统的结构与功能特征分析，获得系统建模需要的相关拓扑结构信息，建立拓扑结构模型；此外利用网络可视化技术，给出网络的拓扑结构图（鉴于SOA系统结构往往可分解为多个子系统，为不失一般性，下面针对具有多个子系统的SOA系统展开方法陈述）；

[0018] 步骤二：由于系统的结构往往会对发生在系统中的动力学行为（故障的传播）产生重要影响，所以在进行系统的故障行为分析之前需要分析系统的静态结构特征；基于复杂网络理论，对建立的可靠性模型的基本静态拓扑属性进行统计分析，包括节点入度分布、节点出度分布、软件共用分析等，以实现系统更好的认识和理解；

[0019] 步骤三：在建立的系统拓扑结构的基础上，基于复杂网络理论，建立针对SOA系统的级联失效模型，得到系统的可靠性模型，设计出适用于SOA系统可靠性评价的仿真实验方案，并完成SOA系统可靠性仿真实验；

[0020] 步骤四：利用步骤三中可靠性仿真实验中的结果，对系统开展脆弱性分析和可靠性计算，给出对系统可靠性影响较大的关键软件列表并分析原因，同时给出系统的可靠性量化评估结果，为高可靠性系统的设计、规划管理和优化控制提供依据。

[0021] 其中，步骤一中所指的“获得系统建模需要的相关拓扑结构信息”具体方法如下：

[0022] 系统中的节点信息以及软件与服务之间连边信息的获得：整个SOA系统可根据使用方的配备分为若干子系统；用户根据需求给出服务列表，由SOA系统设计单位设计出基于用户需求的服务-软件编配表，编配表中包括SOA系统要实现的所有服务，以及实现各个服务需要调用的若干软件；从服务-软件编配列表中可以提取所有的软件、服务信息（包括个数，名称），同时软件被服务调用这一信息被抽象成软件与服务之间的连边；由于一个服务的实现往往需要多个软件的共同协作，一个服务常与多个软件存在连边。

[0023] 同一子系统中不同服务之间的连边信息：同一子系统中不同服务之间的连边由服务之间数据传递关系建立，并且这种连边可以有向的，即如果服务A需要接收来自服务B的数据，那么服务A与服务B之间就存在一条有向边 $B \rightarrow A$ ；同样的一个服务可能需要接收来自多个其它服务的数据，所以指向该服务的边可能有多条；不同子系统间服务间的连边信息：不同子系统间，由于相同服务之间存在信息数据交互，所以相同服务间建立连边。

[0024] 其中，步骤二中所指的“对建立的可靠性模型的基本静态拓扑属性进行统计分析”具体内容与方法如下：

[0025] 系统软件共用次数分析:考虑到系统中相同的软件资源可能被配置到各个子系统各个部门中去,并且由于各部门职能的差异,系统各部门所需的服务存在巨大差别,这最终导致了系统中各个软件的使用频率或者说配置份数不同;对各个软件被不同部门的共用次数的统计,构成了系统软件共用次数分析;分析输出三个结果:各个软件的共用次数、各软件根据共用次数频率的高低排序结果、具有n次共用次数的软件的个数统计结果,其中利用第二个结果得到基于软件共用次数的关键软件列表。

[0026] 服务节点软件度分布统计:根据用户的需求,每项服务会使用数目不等的软件;一个服务的软件度 k_{soft} 指为该软件配置的可供调用的不同软件的数目。

[0027] 服务节点的入度分布统计:一个服务的入度 k_{in} 指的是,服务作为 k_{in} 个其它服务的数据接收方,即系统中有多少个其它服务(上游服务)向它发送数据; k_{in} 越大代表这个服务对外界的依赖性越大。

[0028] 服务节点的出度分布统计:一个服务的出度 k_{out} 指的是,服务需要向其它 k_{out} 个服务发送数据;一个服务的出度 k_{out} 越大,说明这个服务当前的工作状态越容易对系统中其它服务的工作状态造成影响,换句话说,该服务的故障是容易传播开的。

[0029] 其中,步骤三中所述的“建立针对SOA系统的级联失效模型,得到系统的可靠性模型,设计出适用于SOA系统可靠性评价的仿真实验方案”的具体内容与方法如下:

[0030] 级联失效模型:对于系统中的任意一个服务可能接收三种类型的数据:调用其所属的软件、子系统内其它服务以及其它子系统的相同服务传送的数据;这意味着任一服务可能受来自三方面的影响,通常假设各方面对其造成的影响是独立的;系统中的软件由于自身故障(BUG)可能导致错误信息的传递,这个影响概率与服务使用故障软件的概率成正比;方法假设若有一种软件故障,那么所有子系统配置的同种软件也将发生故障;在系统中的服务因软件的故障而受到影响后,由于子系统内以及子系统之间存在服务之间的数据传递,这种影响也可能继续向其所有下游服务节点传递,发生“级联效应”;同样的,服务节点的故障向其某个下游邻居服务节点传播的概率,也是由该下游服务节点使用故障服务的概率决定;其中下游服务节点包括系统内的服务和系统外的服务;三种传播概率的计算方法为:

[0031] 软件i故障影响使用它的服务j的概率:

$$[0032] \quad p_{i \rightarrow j} = 1/k_{j1} \quad (1)$$

[0033] 这里的 k_{j1} 是指服务j拥有的软件的数目;

[0034] 受影响的服务i影响其子系统内下游邻居服务j的概率:

$$[0035] \quad p_{i \rightarrow j} = 1/k_{j2} \quad (2)$$

[0036] 这里的 k_{j2} 是指子系统内向服务j发送数据的服务数目;

[0037] 受影响的服务i影响其子系统外下游邻居服务j的概率:

$$[0038] \quad p_{i \rightarrow j} = 1/k_{j3} \quad (3)$$

[0039] 这里的 k_{j3} 是指子系统外向服务j发送数据的服务数目;

[0040] 基于蒙特卡洛方法的可靠性仿真实验:采用不同的策略对系统中的软件进行故障注入,作为系统的初始故障,注入后故障软件根据故障影响概率向调用它的服务产生影响,使系统中部分的服务节点受到影响而发生故障;之后,在实验的每个迭代步,系统中服务节点的故障向其下游服务节点传播,在固定的迭代步结束后,统计系统中受影响的服务的个

数或比例,作为此次实验中初始故障对系统影响程度的评估;节点间的故障传播使用上述级联失效模型,在仿真故障传播时,若蒙特卡洛抽样得到的随机数小于由模型计算得到的故障影响概率,则传播成功;故障注入策略包括:依次对系统中所有软件注入故障、随机选取一定比例的软件节点注入故障(随机失效)、选取共用频率高(由高到低选取)的软件注入故障(蓄意攻击);第一种故障注入策略进行的仿真实验输出的结果为系统中各软件的故障对系统的影响程度(用受影响的服务的个数来衡量),第二种故障注入策略进行仿真实验输出的结果为不同比例的软件故障对系统的影响程度,第三种故障注入策略进行的仿真实验的输出结果为高共用频次软件的故障对系统的影响程度;另外,为考察SOA系统特定的拓扑结构对系统可靠性的影响,两种对比实验方案也可以实施:保持网络中的节点总数量与连边总数量不变,将网络拓扑结构随机化,随机化的方式为连边随机重连,实验观察网络可靠性的变化;保持网络中服务节点的出度不变,即各个有向边的数据发送端保持不变,随机的重新选择其接受端,实验观察网络可靠性的变化。

[0041] 其中,步骤四中所述的“对系统开展脆弱性分析和可靠性计算”的具体内容与方法如下:

[0042] 软件共用次数统计结果分析:将系统中的每个软件按照共用次数高低进行排序,可以得到一份包含具有较高共用次数的关键软件列表;这个关键软件列表被用来作为蓄意攻击实验中选取故障软件源节点的依据。

[0043] 单一软件故障对系统影响结果分析:在评价系统中任一软件的重要度时,每次试验对一个软件进行故障注入,每个软件的故障对系统影响的程度不同;将系统中的各软件按照影响程度由高到低排序后,得到由对系统可靠运行较重要的软件组成的关键软件列表;另外,将影响程度分为四类严酷度等级,然后根据各软件故障影响严酷度等级将各软件分类,我们可以在系统可靠性优化阶段根据需要,针对某类严酷度等级以上的软件进行改进优化,以最少的成本实现系统可靠度的最大提升。

[0044] 软件随机失效对系统影响结果分析:在这个试验中,每次试验从系统中随机的选取一定比例的软件作为初始故障源,利用级联失效模型进行蒙特卡洛仿真实验,观察对系统的影响程度;这个实验得到了不同比例软件故障对系统的影响结果;以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐标,可以做出系统对随机故障的可靠性曲线图;通过对该结果的分析可以评估系统的整体可靠性水平。

[0045] 蓄意攻击对系统影响结果分析:在这个实验中,蓄意攻击是指,以软件共用次数统计结果为依据,针对共用次数高的软件节点进行故障注入,观察系统的最终影响;每次试验选取共用次数高的软件;同样以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐标,可以做出系统对蓄意攻击的可靠性曲线图;通过对该结果的分析可以发现系统的脆弱部分,所谓脆弱结构是指系统中任何可以用来作为蓄意攻击以达到对系统最大程度破坏目的的结构。

[0046] 随机化的系统结构的可靠性评价:构建与原系统具有相同度均值 $\langle k \rangle$ 的随机化系统网络模型,得到随机故障对随机化系统的影响结果,以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐标,可以做出具有相同度均值随机化系统对随机故障的可靠性曲线图;构建与原系统具有相同度分布的随机化系统网络模型,得到随机故障对随机化系统的影响结果,同样以选取的软件的的比例作为横坐标,以受影响的服务的比例作为纵坐

标,可以做出具有相同度分布的随机化系统对随机故障的可靠性曲线图。

[0047] 结果综合分析:将基于参照物系统的分析结果与原系统上得到的结果进行比较分析,可以揭示系统失效的传播机理以及脆弱结构之所以脆弱的原因。

[0048] 本发明是一种基于复杂网络理论的SOA系统可靠性评价方法,其优点是:

[0049] 目前对SOA复杂系统进行可靠性建模并进行可靠性评价存在很大的困难,传统的系统可靠性建模与分析方法受限于系统规模,并往往针对系统的静态结构进行建模分析,很难顾及发生在系统上复杂的动力学行为对系统造成的影响。

[0050] 针对SOA系统本身是由数量巨大的用户、业务、服务集成的复杂系统,以及其运行机理复杂等特点,充分利用复杂网络科学的优势,将系统中的对象抽象为复杂网络中的节点,将系统中节点之间的相互作用抽象为网络节点之间的连边,这样就建立了针对SOA复杂系统的复杂性分析的拓扑基础—可靠性模型。利用复杂网络理论建立的SOA系统可靠性模型,突出强调系统结构的拓扑特征,对它的研究被有助于理解SOA系统的复杂性来源及其复杂程度等重要问题。

[0051] 针对SOA系统的运行机理,发明建立了系统的可靠性模型,并通过蒙特卡洛仿真实验来对系统中故障的传播机理进行仿真,物理意义清晰,易于工程分析人员操作和实现;方法可以有效发现系统中的脆弱结构,揭示影响系统可靠性的原因与关键因素,为有针对性的改进系统的可靠性薄弱部分提供科学依据,以最少的成本实现系统可靠性的最大提升。

[0052] 方法针对复杂系统的可靠性评价设计,不受系统规模的限制,系统规模的增大不会显著提高方法计算复杂度。

附图说明

[0053] 图1为本发明所述方法的流程图;

[0054] 图2为本发明针对SOA系统的级联失效模型图;

[0055] 图3为本发明应用于某SOA系统所得的系统拓扑结构模型图;

[0056] 图4(a)为本发明应用于某SOA系统所得的软件共用次数统计结果;

[0057] 图4(b)为本发明应用于某SOA系统所得的系统软件度分布;

[0058] 图4(c)为本发明应用于某SOA系统所得的服务节点入度分布;

[0059] 图4(d)为本发明应用于某SOA系统所得的服务节点出度分布;

[0060] 图5为本发明应用于某SOA系统所得的不同故障注入策略下对系统的影响结果图。

具体实施方式:

[0061] 下面将结合附图1方法流程图和实施例对本发明做进一步的详细说明。

[0062] 首先利用步骤一中的方法,在某SOA系统的可靠性评价中,进行系统结构与功能特征分析之后,利用获得的系统结构信息,建立系统的拓扑结构模型;并利用网络可视化软件(本例利用Pajek软件)将模型可视化,如图3。图3是一个包括4个子系统的SOA系统可靠性模型图,其中方形节点代表服务,圆形节点代表软件,节点之间的连边代表软件、服务间的调用数据传递等相互作用或依赖关系,不同的颜色灰度表示了不同的子系统。

[0063] 之后利用步骤二中的方法,对该系统进行静态拓扑结构分析,包括:

[0064] 1) 软件共用次数统计分析,结果见图4(a):分析发现系统中的软件具有较高的共

用率,这在一定程度上降低了系统对蓄意攻击的鲁棒性。

[0065] 2) 度分布统计:度分布是指度为 k 的节点占整个网络的比例 $p(k)$ 。从概率统计的角度看, $p(k)$ 也可以看作是网络中一个随机选择的节点的度为 k 的概率。针对我们建立的可靠性模型,一个服务类节点的度值为三类:实现该服务需要 k 个软件, k 值越大说明这个服务越脆弱;服务需要向其它 k_{out} 个服务发送数据, k_{out} 越大表示这个服务的运行状态对其它服务的影响越大;服务作为 k_{in} 个其它服务的数据接收方, k_{in} 越大代表这个服务对外界的依赖性越大。由图4(b)可以发现系统中的服务拥有的软件的数目的差异比较大,分布范围在2-10之间,表现出服务功能的异质性;图4(c)是对系统中服务节点的入度进行统计得到的入度分布图,可以观察到,系统中入度大的服务数量相对较多,这说明该系统中存在较多的服务需要接收来自不少数目其它服务的数据,容易受到影响;图4(d)是对系统中的服务节点的出度进行统计得到的出度分布图,系统中服务的出度值普遍较大,这说明系统中的服务向较多的其它服务发送数据,这将有可能增大故障传播的范围。

[0066] 然后利用步骤三中的方法,基于建立的可靠性模型,对系统进行考虑故障相关性的脆弱性分析。分析方法:依次对各软件(共96个)注入故障,采用蒙特卡洛方法对系统的级联失效过程进行计算机仿真,观察当前软件的故障可能对系统造成的影响。方法采用的影响评价指标为系统中受影响的服务的数目(或比例)。其中对任一软件的蒙特卡洛仿真实验方法如流程图2,实验采用C++编程实现;下表1为每次影响的传播迭代步数为60步,100次实验平均得到的部分软件影响结果(排序后,另外需要说明的是,由于只需比较各软件故障对系统影响的相对大小,迭代步数的选取的原则是使故障在系统中充分传播)。

[0067] 表1软件故障对系统的影响

软件编号	受影响服务个数	软件编号	受影响服务个数
60	22	57	10
66	15	63	10
58	12	56	10
107	12	71	10
64	12	69	9
34	12	72	9
65	12	73	9
59	12	68	9
33	12	106	8
74	12	111	8
35	11	108	8
61	11	70	8
62	11	67	7
36	11	110	6
17	11	44	6

[0069] 结合复杂网络理论知识以及实验得到的结果,可以分析部分软件之所以对系统影响较大的一般原因:

[0070] 1) 结合软件共用统计分析得到的结果进行分析,可以发现对系统影响较大的软件往往也是共用次数高的;

[0071] 2) 另外,其它一些软件虽然也被较多的部门共用,但是它们的故障并不会对系统产生较大的影响,这是由于调用它们的服务不会向大量的其它服务发送数据,或者数据接

收方对该服务的依赖性不强。

[0072] 利用步骤三中的方法,接下来通过不同的故障注入方式,以及在参照物系统(与原系统具有某方面类似特征的随机化系统)上的实验,分析系统的不同故障方式对系统的可靠性的影响,并探索影响系统可靠性的因素:

[0073] 1) 观察系统中软件的随机故障对系统服务的影响;故障源即初始被注入故障的软件通过随机选取一定比例的软件得到;图5中三角形标识的曲线为得到的结果;

[0074] 2) 观察对系统的脆弱部分进行的蓄意攻击对系统造成的影响:基于软件的共用次数分析结果,将软件按照共用次数由高到低进行排序,按照排序后的次序由高到低对系统中的软件注入初始故障;图5中五角星标识的曲线为蓄意攻击时对系统造成的影响。

[0075] 3) 与原系统具有相同度均值的网络模型的构建:保持系统中各类对象的总数目、服务与软件之间的连边数以及服务之间的边数不变,这样就可以保证度均值 $\langle K \rangle$ 不变;之后,对每一条边(包括服务-软件连边及服务-服务连边)随机地选择边的起始点。可以知道此时构建的随机化网络中服务的入度和出度应服从均匀分布,是均匀化的网络模型;图5中圆形标识的曲线表示的是具有相同度均值的随机网络中软件的故障对系统的影响。

[0076] 4) 与原系统具有相同出度分布的网络模型的构建:保持网络中各类节点数目以及服务节点的出度不变,即各个有向边的数据发送端保持不变,以概率 p 随机的重新选择其接受端;图5中方框标识的曲线表示的是具有相同出度分布的随机网络($p=0.5$)中软件的故障对系统的影响。

[0077] 根据步骤四中的方法,结合步骤三中得到的实验结果,进行系统的可靠性分析评价:

[0078] 1) 系统的随机故障结果显示,当系统初始故障的软件数量增加时,系统影响先是随初始故障比例的增加而急剧攀升,在这之后曲线趋缓,说明当初始故障大于一定比例后,系统影响趋于饱和;

[0079] 2) 分析蓄意攻击对系统的影响结果,可以发现,对少量的特定的节点实施攻击即可使系统迅速崩溃;另外,通过对比系统的随机故障的结果,可以发现系统在蓄意攻击下的脆弱性,这说明系统中确实存在关键软件。

[0080] 3) 分析保持出度分布不变的随机化网络中的实验结果可以得到:与原系统的随机失效结果对比,可以发现系统在随机化之后,其可靠性有了较大的提高;比较两种系统在拓扑结构上的差异,可以分析得到这主要是由于原系统在保持系统的出度分布不变而进行随机化之后,系统的出度分布被均匀化,因为原系统服务的出度值偏大,容易造成系统中故障的传播,而通过随机化消除了这种影响,有效地提高了系统的可靠性;由此可得出一个结论,系统的出度分布对系统可靠性影响较大。

[0081] 4) 分析保持度均值不变的随机化网络中的实验结果可以得到:与原系统的随机失效结果对比,可以发现系统在随机化之后,其鲁棒性有了较大的提高;同时,与保持入度分布不变的随机网络的结果进行对比发现,完全随机化网络要比后者的可靠性要低一些;通过比较系统在拓扑结构上的差异,得到随机化后的网络的入度分布被均匀化,使得网络中服务节点对其它单一服务的依赖性增强,从而要比保持入度分布不变的随机网络的可靠性低;另外,由于随机化网络改变入度分布的同时也改变了出度分布,在之前的分析中我们提到,出度分布的随机化可提高系统的可靠性,这就使系统可靠性高于原系统。

[0082] 综上可以得出结论：一方面，该系统的服务的入度分布的特殊性（绝大多数服务的入度值偏大）保证了系统的具有一定的可靠性；另一方面，系统中较多服务的出度值较大使系统具有一定的脆弱性，通过优化系统结构，改变系统的出度分布将有效地提高系统可靠性。

[0083] 上述各步骤的结果分析表明，本发明方法能够有效地对SOA系统进行可靠性评价，发现系统中影响系统可靠性的关键软件，并可定量的分析各软件的故障对系统的影响程度，并可进一步分析讨论系统中故障的传播机理以及初始故障造成系统大范围破坏的原因，可作为有针对性地提升SOA系统可靠性的依据。

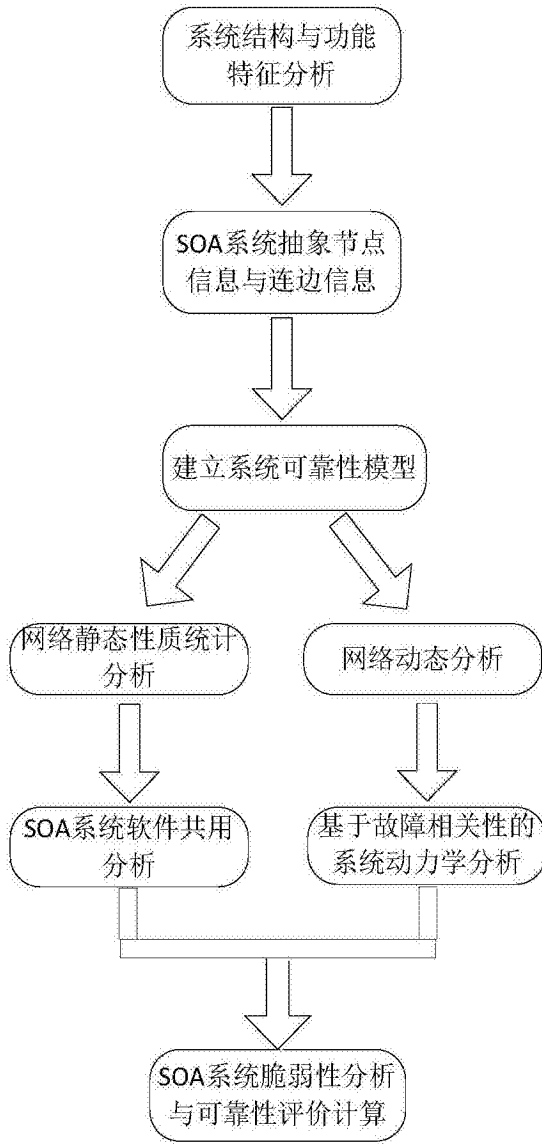


图1

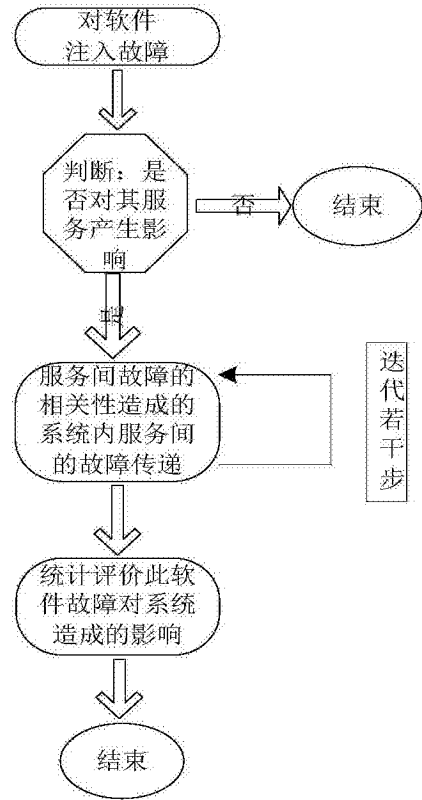


图2

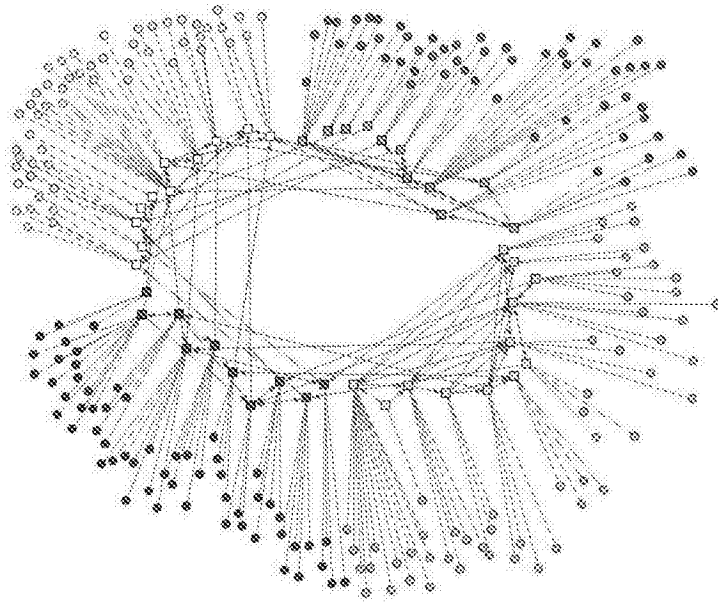


图3

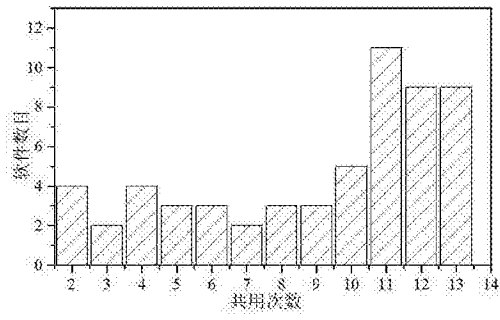


图4 (a)

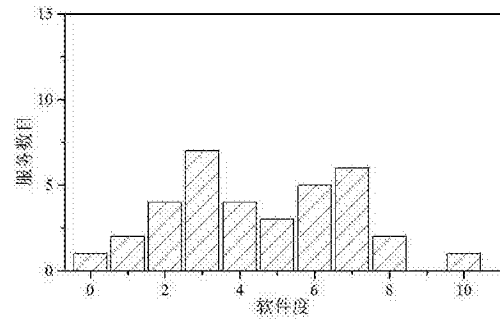


图4 (b)

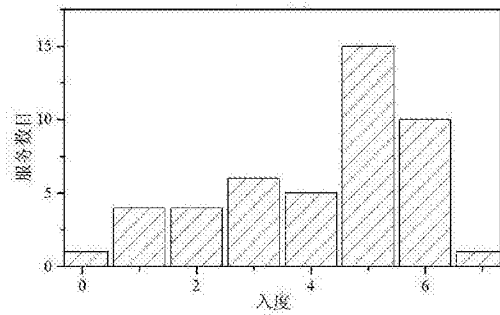


图4 (c)

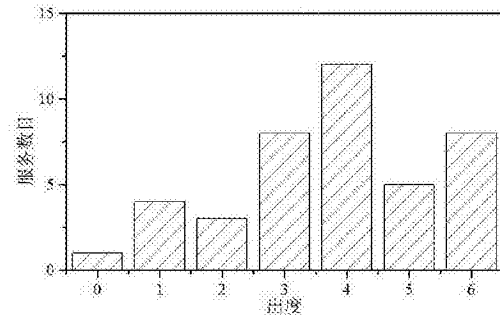


图4 (d)

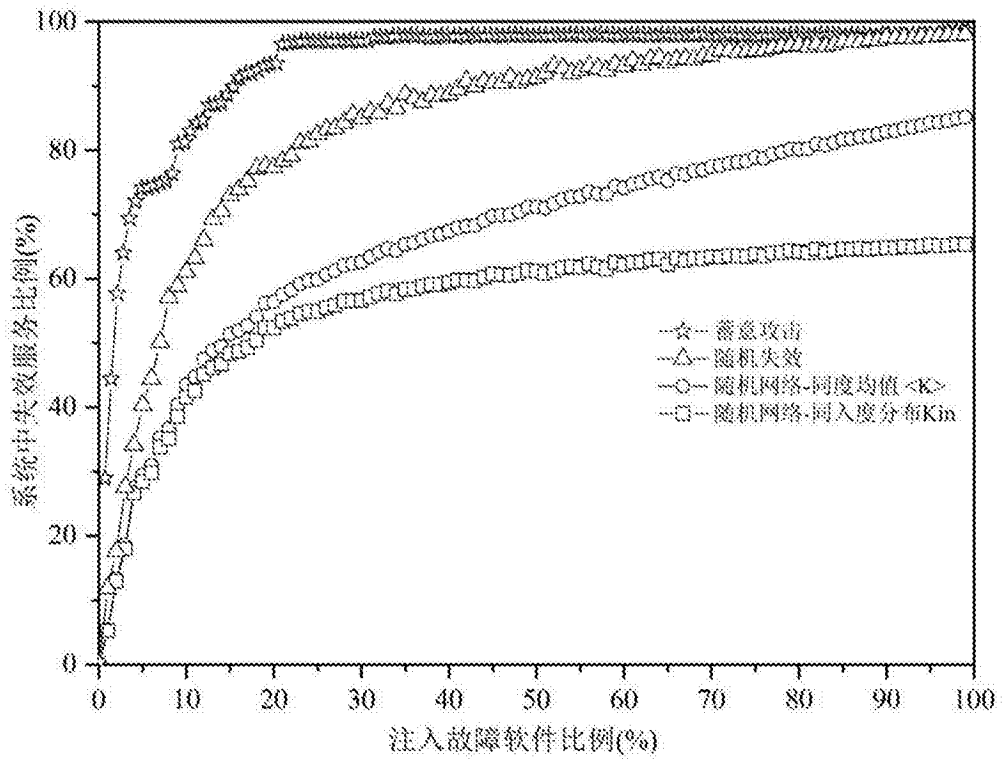


图5