计算机系统应用 ISSN 1003-3254, CODEN CSAOBN
Computer Systems & Applications,2020,29(10):274−279 [doi: 10.15888/j.cnki.csa.007626]

E-mail: csa@iscas.ac.cn
http://www.c-s-a.org.cn
Tel: +86-10-62661041

基于深度学习的 SDN 虚拟蜜网路由优化①

胡　洋

(广州民航职业技术学院 航空港管理学院, 广州 510403)
通讯作者: 胡　洋, E-mail: huyangfox@163.com

摘　要: 针对传统蜜网所致的成本昂贵、流量控制不便及动态调整困难等问题, 提出使用 SDN、ODL 与 Mininet 技术部署轻量级虚拟蜜罐, 组建虚拟蜜网拓扑, 使用深度学习技术 DDPG 优化路由选择路径. 通过实验表明, 优化后的路由选择机制具备动态调整网络结构, 有较好的收敛性和选择性. 使得网络在遭受攻击时, 能将攻击转向蜜网, 从而减少攻击造成的危害, 增强网络主动防御能力.

关键词: 软件定义网络; 深度学习; 虚拟蜜网; 路由优化

引用格式: 胡洋.基于深度学习的 SDN 虚拟蜜网路由优化.计算机系统应用,2020,29(10):274–279. http://www.c-s-a.org.cn/1003-3254/7626.html

SDN Virtual Honeynet Routing Optimization Based on Deep Learning

HU Yang

(Airport Management College, Guangzhou Civil Aviation College, Guangzhou 510403, China)

Abstract: The traditional honeynet has many drawbacks such as inconvenient deployment, difficult flow control, and complex dynamic adjustment. This study proposes to use SDN, ODL, and Mininet technology to deploy lightweight virtual honeypots, build virtual honeynet topology, and use deep learning technology to optimize route selection. The experimental results show that the proposed SDN routing optimization mechanism has sound convergence and effectiveness, and can turn the attack to honeynet when the network is attacked, so as to reduce the damage caused by the attack and thus reduce the network attack threat.

Key words: software defined networking; deep learning; virtual honey net; routing optimization

随着网络环境的日益复杂, 信息安全已成为互联网时代公众关注并重点研究的对象. 但是如防火墙、入侵检测等传统的网络安全硬件设备, 在云计算技术高速发展的时代, 已经出现了技术的滞后, 攻防不平衡的问题. 现有的防御体系需要依赖先验知识[1], 即具备较为广泛的攻防知识储备和数据支撑, 是一种典型的被动防御. 而蜜网[2,3] 恰恰解决了这个问题, 通过提前设计的蜜罐主机, 诱骗攻击者攻击, 使攻击者误以为攻击对象是真机. 从而迷惑攻击者, 同时获取攻击者的攻击行为和态势信息, 再进行分析评估. 这种机制是一种十分有效的主动防御机制. 但是, 传统的蜜网需要物理机部署, 使得其在部署过程中, 存在实施复杂、成本高昂且流量控制困难等问题. 本文提出结合软件定义网络 (SDN) 与 Mininet 技术相结合, 构建虚拟蜜罐主机和软件定义网络拓扑, 并利用深度学习优化路由网络路径选择, 将攻击流量引入蜜网, 从而提高网络的安全性和收敛性, 并通过连续时间的仿真攻击实验验证了网络防御的有效性.

① 基金项目: 2018 年度广东省普通高校重点科研平台和科研项目 (2018GKTSCX084); 广州民航职业技术学院校级科研项目 (17X0206)
Foundation item: Year 2018, Provincial Level Key Platform and Major Scientific Research Program of Guangdong Higher Educations (2018GKTSCX084); Scientific Research Project of Guangzhou Civil Aviation College (17X0206)
收稿时间: 2020-03-11; 修改时间: 2020-04-10; 采用时间: 2020-04-14; csa 在线出版时间: 2020-09-30

1 基于深度学习的 SDN 虚拟蜜网

1.1 蜜网技术的现状研究分析

蜜网技术是在蜜罐技术的基础提出的, 是由诱骗服务模块集中部署的蜜罐群构成, 是具有高交互、研究型的蜜罐技术[4]. 通过物理机部署一套与真实系统几乎完全一致的模拟仿真系统, 诱骗攻击者对模拟目标进行攻击. 其关键技术主要包括进行对攻击者的网络欺骗、攻击数据的扫描获取、网络流量控制、攻击样本特征提取和预警. 但是这种机制交互性较低, 不能做到对未知攻击的识别, 而且存在设备昂贵、部署不便、动态调整难度加大等问题. 这使得有一些学者开始致力于蜜网的改进研究, 文献 [5] 提出了一种通过拟态的方式来有效防范网络攻击的方法. 文献 [6] 针对网络防范方面攻防不平衡问题, 提出通过网络空间拟态主动防御的方法. 文献 [7] 提出了使用 Openflow 协议构建虚拟蜜网并验证延时较低和动态性的特性. 但是, 在蜜网受到攻击时能够自主动态变化的研究却没有太多的突破.

1.2 SDN 虚拟蜜网架构

正是因为 SDN 的提出, 具备了分类转发控制架构的能力, 使得全局网络的动态管控和优化成为在云计算时代得以快速发展的重要原因. 也正是利用这个优点, 可以很好的解决上述传统蜜网的诸多问题. 本文在前人的研究基础上, 利用 SDN 的 OpenDaylight 控制器, 实现对 Open VSwitch 虚拟交换机流表的控制, 简化网络设计, 快速部署虚拟网络, 再利用 Mininet 构建轻量级虚拟蜜罐主机及网络, 能够实现高速、轻量部署蜜网的目标, 虚拟蜜网构建如图 1 所示. 构建的 3 层网络包括: 1) 虚拟基础设施层由网络设备搭建业务网络, 通过配置 OpenVSwitch 部署虚拟交换机, 利用 OpenFlow 协议流表管理交换机, 利用 Mininet 技术连接 SDN 底层, 搭建蜜罐服务器和真实业务服务器, 构建快速、动态调整的虚拟蜜网底层; 2) 决策控制层利用 OpenDaylight 控制器对虚拟交换机的流表信息收集和流量转发控制, 优化网络资源, 实现快速的网络业务部署; 3) 业务应用层则是提供各项网络应用服务, 为外部提供业务访问及主动防御虚拟蜜网, 其中虚拟蜜网的作用可诱导攻击者, 保护真实业务服务器, 同时根据需要动态调整网络结构, 获取攻击者的有效态势数据, 实施主动防御, 如图 1 所示.

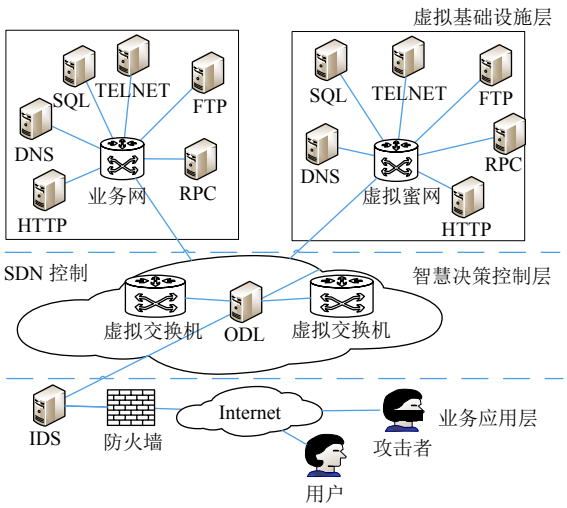

图 1 加入深度学习的 SDN 虚拟蜜网构建

2 构建深度学习的 SDN 蜜网架构

业界学者的研究重心开始关注智能决策和多维数据处理能力的机器学习领域. 如何将深度学习融入 SDN 的转发功能, 让 SDN 具备智能化也是一个值得学者深入研究的问题[8,9].

本节将在 DDPG 技术的基础上, 介绍将深度学习融入 SDN 蜜网的可行性方案, 由此实现 SDN 蜜网的智慧化网络转换.

2.1 SDN 融入深度学习机制

SDN 蜜网架构上叠加机器学习的能力, 使得虚拟蜜网在受到网络攻击时, 主动做出响应, 是智慧体能控制网络, 具备网络转发时延低、动态性强, 主动将攻击转向蜜网的特性.

正如图 1 所示, 在 SDN 虚拟蜜网架构的基础上, 增加智能决策层, 通过 SDN 的全网视图和对流表的控制转发, 可以实现全局的、实时的蜜网智慧管控, 从而达到提高网络安全的作用. 智慧蜜网的关键是对蜜网的状态进行实时感知. 同时, 在受到网络攻击时能够通过智慧决策来调控网络的运行, 其主要的功能是能够智慧化选择路由通路、资源分配, 优化网络资源, 实现网络空间智慧运行.

2.2 深度学习优化路由选择分析

深度学习是指智慧体在环境中具备感知和决策能力, 并能够获取最大奖励值的一种学习方法. 但是基于值的学习机制往往是对离散时间的算法控制, 而对于连续动作的建模则显得力不从心. 这种算法如果应

用在需要实时性和动态性调整的并遭受着频繁攻击的蜜网系统场景, 则会变得毫无意义. 针对这些问题, DeepMind 团队提出了 DDPG (Deep Deterministic Policy Gradient), 得到了更高效稳定的离散动作控制模型, 其主要的方法是将利用 DQN (Deep Q-learning)[10] 和 DPG (Deterministic Policy Gradient) 两种方法的结合, 再通过 μ 神经网络来拟和策略函数和 Q 函数, 构建高效稳定的离散动作模型[11], 最终通过实验验证, 得到求解.

DeepMind 团队在其提出的 DDPG 架构中, 设计了 Actor-Critic 整合架构[12], 其中 Critic 模块使用 DQN 方法, Actor 模块使用 DPG 方法. 这两个模块分别由两组神经网络构成. Critic 模块用于 DQN 神经网络更新, 主要负责阻断训练的 Target 网络; 而 Actor 模块用于拟合 DPG 策略函数更新 Online 网络, 主要负责训练学习的 Online 网络. 其中 Target 网络结构与 Online 相同, 但是参数是由前上一轮时间 Online 网络传递过来的网络参数. 每一轮的学习样本都是从之前互动转换信息经过存储的经验池中提取构成, 再经过学习训练, 不断的更换神经网络参数, 两个模块相互融合、渗透, 训练框架如图 2 所示.

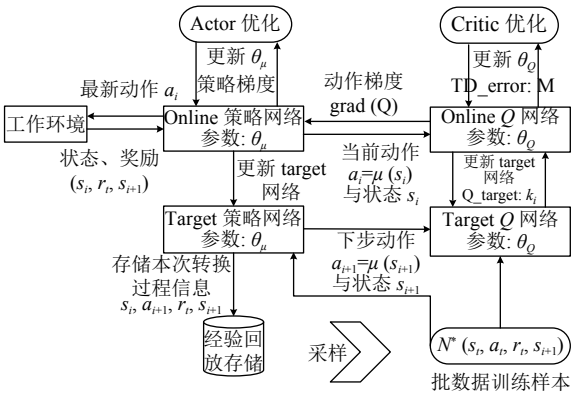

图 2 DDPG 训练框架

在 Actor 模块中, 对于 DPG 神经网络的更新反向传递可以理解为是策略梯度, 计算式如下所示:

$$\nabla_{\theta\mu} K = \text{grad}[Q] * \text{grad}[\mu] \approx \frac{1}{n}\sum_i \nabla_a Q(s_I, a \mid \theta_Q)\Big|_{s=s_i, a=\mu(s_i)} \nabla_{\theta\mu}\mu\left(s \mid \theta_\mu\right)\Big|_{s_i} \tag{1}$$

其中, K 为 μ 策略的性能函数. 表达式右边由两个乘积项组成, 前面一个乘数表示为 Actor 神经网络获得的更大回报时动作移动的方向向量, 即 Critic 神经网络的动作梯度, 后面一个乘数为 Actor 神经网络自身调整的参数, 即 Actor 网络的参数梯度. 前后两者相乘, 得到以获得最高回报来调整自身参数.

在 Critic 模块中, DQN 神经网络的更新反向传递可以表示为式 (2):

$$M = \frac{1}{N}\sum_i (k_i - Q(s_i, a_i \mid \theta_Q))^2 \tag{2}$$

其中, k 是由前上一轮时间 Online 网络传递过来的网络参数作为 Target 网络下一步状态 (s) 与动作 (a) 的值均方差的结果, 所以 y 又可以进一步分解为式 (3):

$$k_i = r_i + \gamma Q'\left(s_{i+1}, \mu'\left(s_{i+1} \mid \theta_{\mu'}\right) \mid \theta_{Q'}\right) \tag{3}$$

前面介绍过, 这里的 μ 代表神经网络来拟和策略函数, Q 代表 Q 值. 即将 Actor 模块中的 Online 网络与 Critic 模块中 Target 网络进行评估, 然后生成 Q 值.

通过利用上述深度学习机制将其部署到 SDN 的虚拟蜜网当中, 对 SDN 路由选择中最小延迟等参数的优化, 实现了自动优化选路. 当网络遇到攻击时, 将攻击流量引入蜜网, 从而迷惑攻击者, 制造假象. 其中, 深度学习机制的智慧体是通过奖励 (r)、动作 (a)、状态 (s)3 个信号与外界进行信息通讯. 我们将当前网络负载量设定为状态 (s); 将设置网络链路权值来改变路由路径选择的行为设定为动作 (a); 最后, 通过改变 Reward 的设置来实现对均衡延时、吞吐量等多个性能参数的优化, 达到奖励 (r) 的目的. 在实际的网络环境中, 可以理解为对网络流量的异常变化分析, 反映出网络受到攻击的情况, 通过捕获当前的网络状态 (s), 具备深度学习的智慧体 SDN 控制器确定一个优化的链路权值, 再根据权值大小, 重新计算链路, 控制器下发实施规则, 最后分析新路径的实施奖励 (r) 的状态, 重复多次的运行, 将攻击流量引入虚拟蜜网, 达到对真实业务网络的保护作用. 如图 3 所示.

3 实验与评估

3.1 实验部署

本文实验平台硬件环境为 CPU 为 Intel E5-2600V4, 内存为 DDR4 的 256 GB, GPU 为英伟达 TESLA V100

32 GB, 利用 Mininet 构建轻量级服务器和蜜网. 服务器安装 Ubuntu 16.04 系统, 在系统上安装 Beryllium 版本的 SDN 控制器 ODL, 以及机器学习框架 TensorFlow 搭建 KerasAPI. 另外服务器通过 ODL 对蜜网进行流量控制, 通过深度学习对 SDN 进行收敛性和有效性进行测试, 验证结果.

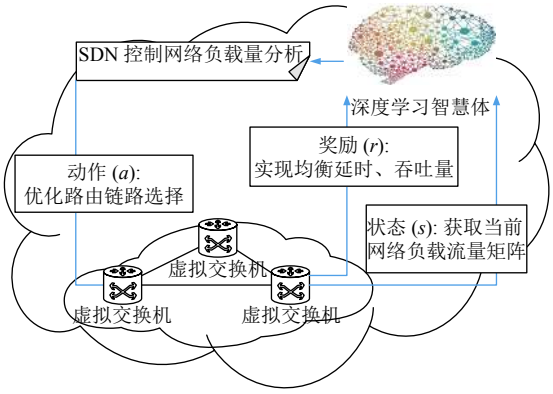

图 3 深度学习优化虚拟蜜网路由选路

3.2 优化路由选择后引入虚拟蜜网测试

实验网络拓扑采用云计算体系下 SDN 虚拟架构, 为了模拟实际业务网络的真实性, 实验网络由 30 个节点及 50 个网络链路构成, 使用 ODL 控制器对蜜网进行动态部署, 营造逼近真实应用环境. 在实验测试中, 使用 Mininet 创建 2 台虚拟交换机及 2 组虚拟服务器, 每台交换机分别连接 1 组服务器, 每组服务器分别配置 HTTP 服务、数据库服务、DNS 服务、TELNET 服务、RPC 服务和 FTP 服务共 6 种服务, 其中一组为真实业务服务器, 另一组则为构建在蜜网中的虚拟蜜罐服务器. 实验时, 分别设置几种不同的流量强度及形成多个差异化的流量矩阵, 以模拟不同环境下网络受到不同程度攻击的效果. 部署如图 4 所示, 其中图中右侧为真实提供服务的业务服务系统, 左侧为诱导攻击者的虚拟蜜网系统. SDN 蜜网测试效果图如图 5.

为了验证本实验受到攻击时具备深度学习的路由选择路径的收敛效果, 实验为对加入优化训练的智慧体路由机制与大量随机生成的路由机制进行比较性训练, 将对其流量强弱设定为不同的步数, 从而对比测试它们在不同环境下的状态, 验证实验结果.

在实验中, 采用占全网 5%、20%、50%、80%、100% 共 5 种带宽强度的流量进行测试, 均生成 100 个流量矩阵, 而后再测试优化后路由路径选择性能, 作出的反应测试如图 6 所示.

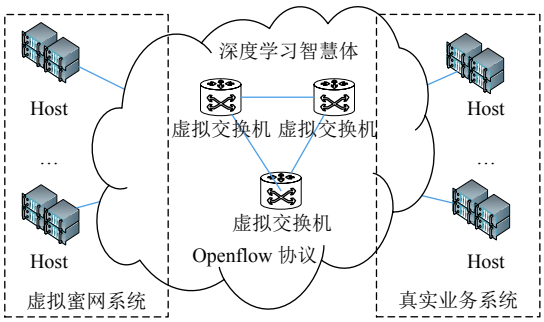

图 4 实验部署图

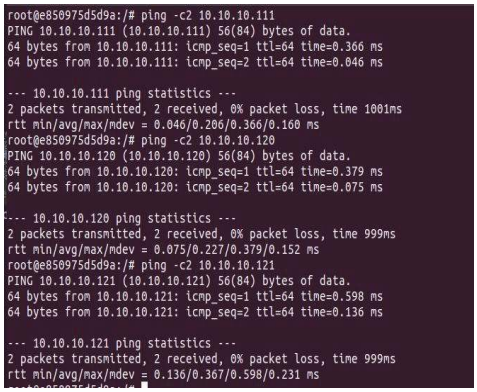

图 5 SDN 蜜网测试效果图

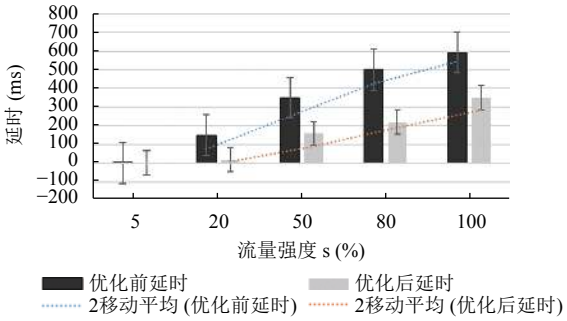

图 6 不同强度实验变化结果

实验结果显示, 在随着网络流量的不断增加, 即网络遭到攻击越来越严重的情况下, 经过训练后的优化路由选择都有明显提升, 其中在流量强度为 80% 的时候, 优化最为明显, 提升效率接近 40.6%. 这说明, 深度学习后的 SDN 虚拟路由选择将攻击流量较好的引入

了不负责实际业务的蜜网, 由蜜网担负了大量的攻击流量, 实验证明了优化路径选择具备明显的收敛性. 从而在虚拟蜜网的构建中, 对抵御网络攻击起到了良好的保护作用, 为实际业务网络的运行提供了保障, 增加了攻击难度. 同时为蜜网获取有效的攻击态势数据, 提供了良好的网络基础.

这里值得注意的是: 该算法的时间复杂度为 $O(n^2)$, 其中的 n 为网络边数. 智慧体在经过训练得到收敛网络路径后, 就可以通过矩阵乘法计算出最优路径. 对比启发式算法, 例如蚁群算法的时间复杂度 $O(n(n–1)mt)$ 而言, 本文的算法在 SDN 虚拟蜜网的应用中具备更低的时延性. 而且与传统的静态蜜网相比, 具备更高的保护性和抗攻击性, 更难识别. 基于深度学习的 SDN 网络是目前云计算时代网络发展的一个必然趋势, 应用前景广阔. 本架构设计采用 Python 代码编写网络拓扑(如图 7), 能够根据需要动态的增减主机和网络拓扑, 智能选择路由, 提高网络安全.

图 7 Python 编写网络拓扑脚本

本实验在测试的同时, 还对真实业务服务器及蜜网中的蜜罐服务器在同一时间间隔下优化前后受到的攻击做出了统计与对比, 具体数据如表 1 所示. 其中用“n/n”两组数据来表示“优化前/优化后”的比较, 空白处为该项服务未开启, 0 表示该时间段该服务器未收到攻击.

表 1 优化前后业务服务器与蜜罐受到攻击对比情况

时间	HTTP		FTP		MySQL		DNS		TELNET		REC	
	真机	蜜罐	真机	蜜罐	真机	蜜罐	真机	蜜罐	真机	蜜罐	真机	蜜罐
T1			2/0	1/1	1/1	1/1					3/1	1/1
T2	2/1	2/3	2/1	1/2	2/0	1/4	2/1	1/3	2/1	2/3	2/0	1/2
T3			1/0	2/3							1/0	3/3
T4	2/0	1/4	3/0	1/4	1/1	2/3			1/2	2/4	4/1	2/4

实验结果可以看出, 在经过深度学习的训练后, SDN 路径选择将攻击转向蜜网, 蜜网中的蜜罐服务器受到的攻击次数比真机受到的攻击次数明显增多. 其访问过程可以解释为: 当外来访问者正常请求某种服务时, 首先是正常访问真实业务服务器, 由业务服务器提供服务; 当外来访问者想利用业务服务器漏洞发起攻击时, 访问强度、频率或流量会出现明显增大的情况, 这时会触发智慧体认知为是攻击行为, 并同时采取路径选择优化机制, 将数据包的目的 IP 和 Mac 地址转向蜜网中的蜜罐服务器, 由蜜罐服务器提供相关服务, 同时记录攻击者态势数据, 而攻击者并不会察觉出任何异常. 这时系统将主动阻断攻击链, 从而起到保护业务真机服务器, 实现防御攻击的目的. 同时, 在攻击者对网络进行较为强烈的攻击时, 智慧体会再次自主调整蜜网系统, 转化网络路由选择, 从而使得网络更加隐蔽性. 通过实验验证了深度学习的 SDN 虚拟蜜网具备主动防御的可行性, 减少了网络攻击威胁, 获取了更多攻击态势数据. 另外, 由于正常访问过大而引起的流量增大情况, 如何解决? 其实前面介绍过, 蜜罐服务器有着与真实业务服务器一样的服务能力, 因此, 在正常访问过大而引起的流量增大时, 蜜罐服务器也是可以提供正常服务的, 经过优化分类从而分担真实业务服务器的压力, 这里就不再讨论了.

4 结语

本文在前人的研究基础上, 结合深度学习技术、SDN 网络技术、ODL 和 Mininet 技术组建轻量级蜜罐服务器, 构建动态虚拟蜜网. 通过验证, 证实了本文设计的具备深度学习的智能蜜网机制能够根据网络访问强度和访问流量的变化自主路由选择, 将攻击引向蜜网机制. 解决了传统蜜网部署复杂、造价昂贵、流量控制不便等问题, 同时有效的提高了网络的隐蔽性, 实现了主动防御, 保障了业务网络有效运行.

参考文献

1 Zhang Y, Tan H, Zhao SL. Research on the application of

firewall in network security. Geomatics & Spatial Information Technology, 2011, 34(6): 249–250, 254.
2 Gautam R, Kumar S, Bhattacharya J. Optimized virtual honeynet with implementation of host machine as honeywall. Proceedings of 2015 Annual IEEE India Conference. New Delhi, India. 2015. 1–6.
3 HUSS. Analysis of attack based on honey net. Beijing: Beijing University of Posts and Telecommunications, 2015.
4 Zhuge JW, Tang Y, Han XH, *et al*. Honeypot technology research and application. Journal of Software, 2013, 24(4): 825–842.
5 仝青, 张铮, 张为华, 等. 拟态防御 Web 服务器设计与实现. 软件学报, 2017, 28(4): 883–897. [doi: 10.13328/j.cnki.jos.005192]
6 邬江兴. 网络空间拟态安全防御. 保密科学技术, 2014, (10): 4–9.
7 胡毅勋, 郑康锋, 武斌, 等. Openflow 下的动态虚拟蜜网系统. 北京邮电大学学报, 2015, 38(6): 104–108.
8 Boutaba R, Salahuddin MA, Limam N, *et al*. A comprehensive survey on machine learning for networking: Evolution, applications and research opportunities. Journal of Internet Services and Applications, 2018, 9(1): 16. [doi: 10.1186/s13174-018-0087-2]
9 Mestres A, Rodriguez-Natal A, Carner J, *et al*. Knowledge-defined networking. ACM SIGCOMM Computer Communication Review, 2017, 47(3): 2–10. [doi: 10.1145/3138808.3138810]
10 Mnih V, Kavukcuoglu K, Silver D, *et al*. Human-level control through deep reinforcement learning. Nature, 2015, 518(7540): 529–533. [doi: 10.1038/nature14236]
11 Silver D, Lever G, Heess N, *et al*. Deterministic policy gradient algorithms. Proceedings of the 31st International Conference on International Conference on Machine Learning. Beijing, China. 2014. 1-387–1-395.
12 Paul LT, James HJ, Alexander P, *et al*. Continuous control with deep reinforcement learning: USA, WO2017019555A1. [2016-07-22].