基于深度强化学习的机械臂控制方法

李鹤宇[1]，赵志龙[1,2,3]，顾蕾[1]，郭丽琴[1,2,3]，曾贲[1]，林廷宇[1,2,3]

(1. 北京市复杂产品先进制造系统工程技术研究中心 北京仿真中心，北京 100854；
2. 复杂产品智能制造系统技术国家重点实验室 北京电子工程总体研究所，北京 100854；
3. 航天系统仿真重点实验室 北京仿真中心，北京 100854)

摘要：深度强化学习在环境中不断探索尝试，通过奖励函数对神经网络参数进行调节。实际的生产线无法作为算法的试错环境，不能提供足够的数据，*构建一个机械臂仿真环境，包括机械臂与物体两部分，根据目标设置状态变量与奖励机制，在模型中对深度确定性策略梯度算法(Deep Deterministic Policy Gradient, DDPG)进行训练，实现通过深度强化学习算法控制机械臂，将抓手移动至物体下方，改进控制算法的适应性，缩短调试时间。*实验结果表明，深度学习算法能够在更短的时间内达到收敛，实现对机械臂的控制。

关键词：系统仿真；Unity；强化学习；神经网络

中图分类号：TP391 文献标识码：A 文章编号：1004-731X (2019) 11-2452-06

DOI: 10.16182/j.issn1004731x.joss.19-FZ0378

Robot Arm Control Method Based on Deep Reinforcement Learning

Li Heyu[1], *Zhao Zhilong*[1,2,3], *Gu Lei*[1], *Guo Liqin*[1,2,3], *Zeng Bi*[1], *Lin Tingyu*[1,2,3]

(1. Beijing Complex Product Advanced Manufacturing Engineering Research Center, Beijing Simulation Center, Beijing 100854, China;
2. State Key Laboratory of Intelligent Manufacturing System Technology, Beijing Institute of Electronic System Engineering, Beijing 100854, China;
3. Science and Technology on Space System Simulation Laboratory, Beijing Simulation Center, Beijing 100854, China)

Abstract: Deep reinforcement learning continues to explore in the environment and adjusts the neural network parameters by the reward function. The actual production line can not be used as the trial and error environment for the algorithm, so there is not enough data. *For that, this paper constructs a virtual robot arm simulation environment, including the robot arm and the object. The Deep Deterministic Policy Gradient (DDPG),in which the state variables and reward function are set,is trained by deep reinforcement learning algorithm in the simulation environment to realize the target of controlling the robot arm to move the gripper below the object. The new method using neural network can improve the adaptability of the control algorithm and shorten the debugging time.* The simulation results show that in the environment constructed in this paper, the deep learning algorithm can converge in a shorter time and control the robot arm to achieve specific goals.

Keywords: system simulation; unity; reinforcement learning; neural network

引言

随着工业技术的发展，机械臂的应用受到学术界的广泛重视。在投入实际应用前，需要对控制系统进行调试，以适应特定的生产环境；当使用机械臂完成精度要求高、流程复杂的操作时，为应对复杂的任务需求和非线性环境，需要对其控制算法进行改进，不断提高控制精度与适应性。

经典控制方法，如比例-积分-微分控制、鲁棒控制、自适应控制绝对控制精度有限，难以满足工

收稿日期：2019-05-21 修回日期：2019-07-25；
基金项目：国家重点研发计划(2018YFB1004005)；
作者简介：李鹤宇(1993-)，男，河北石家庄，硕士，研究方向为深度强化学习，建模仿真技术；赵志龙(1987-)，男，河北廊坊，硕士，助工，研究方向为虚拟样机，智能制造等。

业生产的要求，为此学者对控制方法进行改进，以获得更好的机械臂控制效果。Wopereis 等[1]在状态反馈中加入线性二次型调节器，以实现对连续接触力的控制。李慧洁等[2]使用特定双幂次趋近律具有的全局快速固定收敛特性，对滑膜控制抖动、收敛慢的问题进行改善。Soltanpour 等[3]为客服现有机器人位置跟踪存在不确定性的问题，提出一种最优模糊滑膜控制器实现对机械臂未知的跟踪。Wang 等[4]针对非线性系统，通过反推技术，构造具有严格反馈结构的李雅普诺夫函数，获得稳定的控制器。Yin 等[5]利用知识库和状态机实现一种适用于转弯过程中的控制算法，该算法能够通过启发式规则避免障碍物。Cho 等[6]将控制平面的法向量转化为特定的向量和点，以实现基于矩阵增强的主动控制方法。Li 等[7]将模糊输出反馈镇定控制方法应用于非严格反馈不确定非线性切换系统，使得控制系统的输出收敛于原点附近的极小邻域。Li 等[8]为获得互联非线性系统的渐进稳定性，利用光滑切换函数构建自适应更新律，实现自适应分布式控制方法。Wang 等[9]使用 Lyaounov 方程和模糊系统逼近非线性函数，从而减轻离散切换非线性系统中机械臂参数在线调节压力。

深度强化学习将深度学习与强化学习相结合，随着计算机技术的发展与数据的增加，涌现出 DQN (Deep Q-network)[10] 、DDPG (Deep Deterministic Policy Gradient)[11] 、TRPO (Trust Region Policy Optimization)[12]、A3C (Asynchronous Advantage Actor-Critic)[13] 、DPPO (Distributed Proximal Policy Optimization)[14-15]等。学者将神经网络应用于机械臂，获得了更好的效果。Buchli 等[16]将基于路径积分的策略优化方法用于特定动作的学习，实现优化运动原语模型的参数。Stulp 等[17]将运动原语的参数调整转化为优化问题，提出一种俊华策略方法。Wang 等[18]使用单个变量和阶跃激活函数，实现具有硬限制的神经网络激活函数。Liu 等[19]提出一种单层递归神经网络，用于求解一类具有分段线性目标函数的约束非光滑优化问题。Kormushev 等[20]使用模仿学习的方法，在人机交互环境下使得机械臂掌握学习和再现动作的能力。陈友东等[21]为实现使用机械臂自使用抓取物体，构建单高斯过程模型实现了目标物体的位姿与机械臂关节角度之间的关联。Ngo 等[22]提出基于神经网络的鲁棒控制方案，使用神经网络对干扰及摩擦等参数变化部分进行补偿，达到较高的控制精度。Lee 等[23]针对机械臂接触的环境未知的情况，提出一种基于深度学习的自适应控制方法，采用神经网络估计位置环境的模型。

本文主要构建一个三维仿真环境，包括机械臂和生产线 2 部分，用于模拟生产线中机械臂托取物体的场景，在对深度强化学习算法进行训练。使用 DDPG 算法对虚拟环境中的机械臂进行控制，算法根据仿真环境奖励函数返回的值进行参数调整，通过在虚拟环境中训练实现收敛，完成特定的动作。人工调试需要较长的时间，使用深度学习能够更快实现相应的控制，并且本文采取托取物体的方式，相比于抓取的方式，能够减少对物体造成的形变，但该方法对深度强化学习算法提出了更高的要求，抓盘必须移动至特定区域才能保证物体不跌落。

1 基于深度强化学习的机械臂控制方法

1.1 系统结构

系统包括深度学习算法和仿真两部分，在系统中对神经网络进行训练，实现使用 DDPG 控制机械臂托取对应的物体。仿真环境包括机械臂和生产线 2 部分，构建仿真环境，接收控制变量，将实施相应控制后的环境信息传递给深度学习算法。根据环境信息，深度学习部分获得状态变量和奖励值，通过前者计算机械臂的控制量，通过后者对神经网络参数进行更新，如图 1 所示。

1.2 机械臂模型

深度强化学习算法通过试错进行参数优化，因此首先需要构建机械臂模型为神经网络提供学习

环境。使用 Unity 引擎搭建前端模型，共包括机械臂和物体两部分，模拟利用叉盘式抓手托取物体的场景，如图 2 所示。

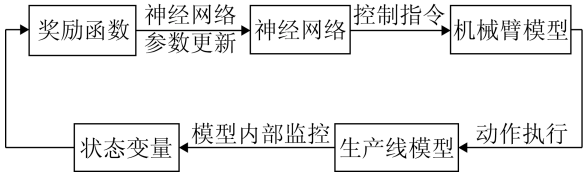
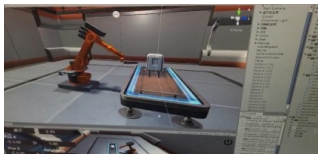
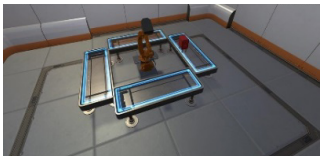


图 1　系统结构
Fig. 1　System structure



(a) 控制界面



(b) 随机初始化位置

(c) 目标状态

图 2　机械臂模型
Fig. 2　Robot arm model

模型接收控制量后，根据逆运动学计算机械臂末端位置，6 轴机械臂运动学方程如下：

$$ {}^{0}_{5}T=\begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}= {}^{0}_{1}T(\theta_1)\,{}^{1}_{2}T(\theta_2)\,{}^{2}_{3}T(\theta_3)\,{}^{3}_{4}T(\theta_4)\,{}^{4}_{5}T(\theta_5) \tag{1} $$

式中：夹角 θ_i 为在垂直于关节轴线的平面内，相邻连杆 i 与 i–1 的夹角。通过对公式(1)的两端同时左乘一个其次变换的逆：

$$ \begin{bmatrix} C_1 & S_1 & 0 & 0 \\ -S_1 & C_1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}= {}^{0}_{5}T \tag{2} $$

1.3 基于 DDPG 算法的机械臂控制方法

深度强化学习根据环境的状态变量(state)，输出特定的动作(action)，并根据环境根据该动作获得的奖励(reward)，更新神经网络的参数。

DDPG 算法使用 eval 和 target 两套神经网络表示策略函数 actor 和值函数 critic。actor 接收环境信息，输出对应的动作变量，critic 网络根据相应的动作变量计算奖励值，如图 3 所示。

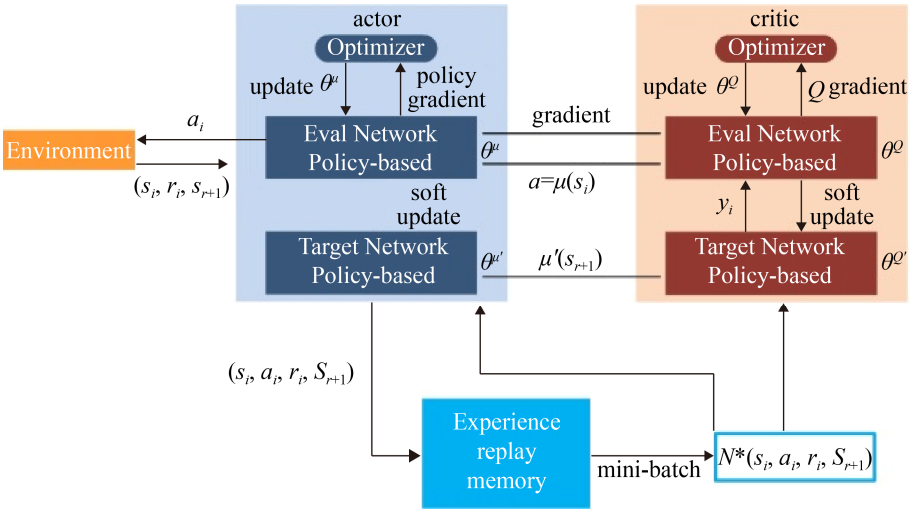


图 3　DDPG 算法流程
Fig. 1　Algorithm flow of DDPG

target 网络的参数采用软更新的方式进行：

$$\text{soft update}\begin{cases}\theta^{Q'} \leftarrow \gamma\theta^{Q}+(1-\gamma)\theta^{Q'}\\ \theta^{\mu'} \leftarrow \gamma\theta^{\mu}+(1-\gamma)\theta^{\mu}\end{cases} \tag{3}$$

式中：θ^{Q} 为 eval 网络中 actor $Q(s,a|\theta^{Q})$ 的参数；θ^{μ} 为 eval 网络中 critic $\mu(s|\theta^{\mu})$ 的参数；$\theta^{Q'}$ 为 target 网络中 actor $Q'(s,a|\theta^{Q'})$ 的参数；$\theta^{\mu'}$ 为 target 网络中 critic $\mu'(s|\theta^{\mu'})$ 的参数。eval 网络中的 actor 部分采用 policy gradient 的方法进行优化：

$$\nabla_{\theta^{\mu}}J\approx\frac{1}{N}\sum_{i}\nabla_{a}Q(s,a|\theta^{Q})\nabla_{\theta^{\mu}}\mu(s|\theta^{\mu}) \tag{4}$$

式中：s_i 为当前时刻的状态变量。eval 网络中的 critic 采用类似于监督学习的方法，使用均方根误差定义 loss：

$$\begin{cases}\text{loss}=\dfrac{1}{N}\sum_{i}(y_i-Q(s_i,a_i|\theta^{Q}))^2\\ y_i=r_i+\gamma Q'(s_{i+1},\mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})\end{cases} \tag{5}$$

采用梯度下降的方法优化网络参数。

在物体位置随机初始化时，致力于使用 DDPG 算法对机械臂进行控制，将抓手移动至物体下方指定位置。为此算法对状态获取机制与奖励机制进行设计。由环境信息生成的状态变量共 79 维，如式(6)所示：

$$\begin{cases}dis\_jt_i=(joint_i-tgt_i)/2\\ dis\_jj_i=(joint_i-joint_0)/2\\ dis\_th_i=(tpoint_i-hpoint_i)/4, i=1,2,3,4\\ dis\_hj_i=(hpoint_i-joint_i)/4\\ dis\_col\end{cases} \tag{6}$$

式中：*joint* 是机械臂每个关节的三维坐标，*tgt* 是物体中心点的三维坐标，*tpoint* 是物体下方点的三维坐标，*hpoint* 是抓手上方点的三维坐标，*dis_col* 是碰撞发生情况。公式(5)的左侧是状态变量。

使用奖励机制引导机器人手臂做出正确的动作，共分为 2 个阶段。第一阶段将夹具引导到物体下方的位置：

$$\begin{cases}jre_1=\left(\sum_{i=0}^{3}\|tpoint_i-hpoint_i\|/4\right)\\ jre_2=\left(\sum_{i=0}^{3}|hpoint\_x_i|/4\right)\\ jre_3=\left(\sum_{i=0}^{3}|hpoint\_y_i|/4\right)\\ jre_4=\cos(hvect,tvect)\end{cases} \tag{7}$$

式中：*jre* 为奖励值；*hpoint_x* 为 *hpoint* 在 *x* 轴上的值；*hpoint_y* 为 *hpoint* 在 *y* 轴上的值；*hvect* 为抓手平面的法向量；*tvect* 为物体下表面的法向量。

第二阶段引导抓手垂直向上移动：

$$\begin{cases}part_1=\left(\sum_{i=0}^{3}\|tpoint_i-hpoint_i\|/4\right)\\ part_2=\left(\sum_{i=0}^{3}\cos((tpoint_i-hpoint_i),y)/4\right)\end{cases} \tag{8}$$

2 仿真验证

为了本文算法的先进性，从运行时间、奖励值、机械臂与物体距离等角度进行比对，以验证算法在计算效率与精确度上的效果。

传统机械臂调试根据熟练度不同，需要时间在 2~5 日的范围，采取中值。算法统计单机械臂神经网络训练时间，其结果如表 1 所示。

表 1 不同方法所需时间

Tab. 1 Time for different methods

方法	传统调试方法	本文方法
时间/h	84.0	33.2

由表中数据可知，传统调试方法耗时最长。使用深度强化学习算法对机械臂进行控制，通过调试参数实现机械臂能够托取物体仅需要 33.2 h，效率提升 60.5%。

其次统计每一次循环中奖励值的大小，验证是否随着学习的进行，在每一个周期内的神经网络选择的动作能够获取更大的奖励值，即达到更好的控制效果。当神经网络每层神经元的维度为 300，该网络在每个周期的奖励值逐渐上升，如图 4 所示，经过数据填充，进入学习阶段后，奖励值开始逐渐升高，说明神经网络通过训练能够改变自身参数，

做出的动作越发符合要求。

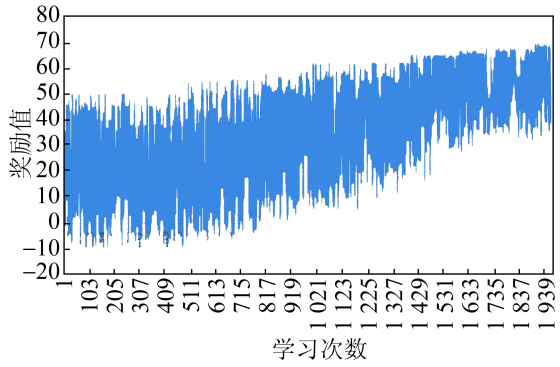


图 4 算法周期奖励值

Fig. 4 Reward value for each cycle

此外，项目通过测量机械臂抓手上方 4 个点与物体下方对应 4 个点距离的平均值，直观地反应机械臂是否能够移动到物体下方的对应位置。每次初始化时，机械臂移动至固定位置，平均距离为 80 cm，当距离小于 1 cm 时，认为机械臂成功移动到物体下方指定位置。图 5 为学习稳定后一个循环中的平均距离，从图中可以看出，机械臂与物体的距离不断降低，最终稳定在 1 cm，因此收敛后的神经网络能够很好地执行对应任务。

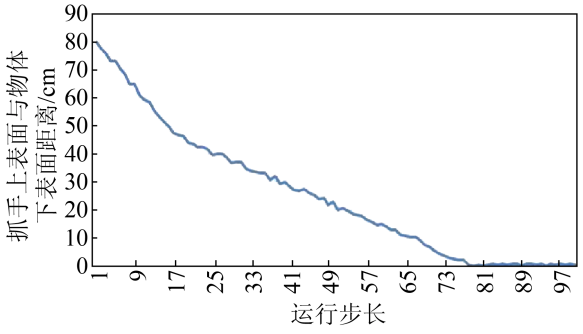


图 5 机械臂抓手上方 4 个点与物体下方 4 个点对应距离的平均值

Fig. 5 The average distance between 4 points above the gripper and corresponding points below the object

3 结论

本文提出的一种基于数字孪生的智能制造计划管理模式在虚拟现实融合的智能计划排程系统 6 大功能模块的支持下，定义了计划与排程、排程与执行、订单承诺与履约 3 大业务闭环过程，构成了完整的计划 PDCA 循环。基于虚拟制造系统的智能计划动态调整技术在计划排程 PDCA 闭环业务中专门处理各类不确定因素的扰动，支持周期性滚动和事件驱动的应急优化排程两种方式，能够应对工序调整、资源调整、订单调整等多种类型的扰动。

本文所述系统能够为数字化向智能化转型升级提供仿真决策支撑和执行监控一体化解决方案。同时，作为智能制造系统的虚拟仿真实验平台，为开展基于深度强化学习的计划与排程 Agent 模型验证、多机械臂智能装配动作路径规划仿真验证、智能制造系统仿真模型验证与改进等研究提供基础支撑。此外，计划管理中核心的优化问题建模方法与求解工具可以为后续军事运筹相关问题的建模与求解提供支撑。

参考文献：

[1] Wopereis H W, Hoekstra J J, Post T H, et al. Application of substantial and sustained force to vertical surfaces using a quadrotor[C]. 2017 IEEE international conference on robotics and automation (ICRA). Macau: IEEE, 2017: 2704-2709.

[2] 李慧洁，蔡远利. 基于双幂次趋近律的滑模控制方法[J]. 控制与决策, 2016, 31(3): 498-502.
Li Huijie, Cai Yuanli. Sliding mode control with double power reaching law[J]. Control and Decision, 2016, 31(3): 498-502.

[3] Soltanpour M R, Khooban M H. A particle swarm optimization approach for fuzzy sliding mode control for tracking the robot manipulator[J]. Nonlinear Dynamics (S0924-090X), 2013, 74(1/2): 467-478.

[4] Wang Z, Liu X, Liu K, et al. Backstepping-based Lyapunov function construction using approximate dynamic programming and sum of square techniques[J]. IEEE Transactions on Cybernetics (S1083-4419), 2016, 47(10): 3393-3403.

[5] Yin X G, Wang H P, Wu G. Path planning algorithm for bending robots[C]. 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO). Guilin: IEEE, 2009: 392-395.

[6] Cho H C, Song J B. Null space motion control of a redundant robot arm using matrix augmentation and saturation method[C]. 12th International Conference on

Motion and Vibration Control, MOVIC 2014. Sapporo: Japan Society of Mechanical Engineers, 2014.

[7] Li Y M, Tong S C. Adaptive fuzzy output-feedback stabilization control for a class of switched nonstrict-feedback nonlinear systems[J]. IEEE Transactions on Cybernetics (S1083-4419), 2016, 47(4): 1007-1016.

[8] Li X J, Yang G H. Adaptive decentralized control for a class of interconnected nonlinear systems via backstepping approach and graph theory[J]. Automatica (S0005-1098), 2017, 76: 87-95.

[9] Wang H, Wang Z, Liu Y J, et al. Fuzzy tracking adaptive control of discrete-time switched nonlinear systems[J]. Fuzzy Sets and Systems (S0165-0114), 2017, 316: 35-48.

[10] Mnih V, Kavukcuoglu K, Silver D, et al. Playing atari with deep reinforcement learning[J]. arXiv preprint arXiv:1312.5602, 2013.

[11] Lillicrap T P, Hunt J J, Pritzel A, et al. Continuous control with deep reinforcement learning[J]. arXiv preprint arXiv:1509.02971, 2015.

[12] Schulman J, Levine S, Abbeel P, et al. Trust region policy optimization[C]. International Conference on Machine Learning. 2015: 1889-1897.

[13] Mnih V, Badia A P, Mirza M, et al. Asynchronous methods for deep reinforcement learning[C]. International conference on machine learning. New York: dblp, 2016: 1928-1937.

[14] Schulman J, Wolski F, Dhariwal P, et al. Proximal policy optimization algorithms[J]. arXiv preprint arXiv: 1707.06347, 2017.

[15] Heess N, Sriram S, Lemmon J, et al. Emergence of locomotion behaviours in rich environments[J]. arXiv preprint arXiv:1707.02286, 2017.

[16] Buchli J, Stulp F, Theodorou E, et al. Learning variable impedance control[J]. The International Journal of Robotics Research (S0278-3649), 2011, 30(7): 820-833.

[17] Stulp F, Sigaud O. Robot skill learning: From reinforcement learning to evolution strategies[J]. Paladyn, Journal of Behavioral Robotics (S2081-4836), 2013, 4(1): 49-61.

[18] Wang J. Analysis and design of a k-winners-take-all model with a single state variable and the heaviside step activation function[J]. IEEE Transactions on Neural Networks (S2162-237X), 2010, 21(9): 1496-1506.

[19] Liu Q, Wang J. Finite-Time Convergent Recurrent Neural Network With a Hard-Limiting Activation Function for Constrained Optimization With Piecewise-Linear Objective Functions[J]. IEEE Transactions on Neural Networks (S2162-237X), 2011, 22(4): 601-613.

[20] Kormushev P, Calinon S, Caldwell D G. Imitation Learning of Positional and Force Skills Demonstrated via Kinesthetic Teaching and Haptic Input[J]. Advanced Robotics (S0169-1864), 2011, 25(5): 581-603.

[21] 陈友东, 郭佳鑫, 陶永. 基于高斯过程的机器人自适应抓取策略[J]. 北京航空航天大学学报, 2017, 43(9): 1738-1745.
Chen Youdong, Guo Jiaxin, Tao Yong. Adaptive Grasping Strategy of Robot Based on Gaussian Process[J]. Journal of Beijing University of Aeronautics and Astronautics, 2017, 43(9): 1738-1745.

[22] Ngo T Q, Wang Y N, Mai T L, et al. Robust adaptive neural-fuzzy network tracking control for robot manipulator[J]. International Journal of Computers Communications & Control (S1841-9836), 2012, 7(2): 341-352.

[23] Lee C H, Wang W C. Robust adaptive position and force controller design of robot manipulator using fuzzy neural networks[J]. Nonlinear Dynamics (S0924-090X), 2016, 85(1): 343-354.