第 27 卷第 1 期
2010 年 1 月

控 制 理 论 与 应 用
Control Theory & Applications

Vol. 27 No. 1
Jan. 2010

文章编号: 1000−8152(2010)01−0111−05

机器人动态神经网络导航算法的研究和实现

乔俊飞, 樊瑞元, 韩红桂, 阮晓钢

(北京工业大学 电子信息与控制工程学院, 北京 100124)

摘要: 针对Pioneer3-DX 移动机器人, 提出了基于强化学习的自主导航策略, 完成了基于动态神经网络的移动机器人导航算法设计. 动态神经网络可以根据机器人环境状态的复杂程度自动地调整其结构, 实时地实现机器人的状态与其导航动作之间的映射关系, 有效地解决了强化学习中状态变量表的维数爆炸问题. 通过对Pioneer3-DX移动机器人导航进行仿真和实物实验, 证明该方法的有效性, 且导航效果明显优于人工势场法.

关键词: 移动机器人; 导航; 动态神经网络

中图分类号: TP273　　**文献标识码**: A

Research and realization of dynamic neural network navigation algorithm for mobile robot

QIAO Jun-fei, FAN Rui-yuan, HAN Hong-gui, RUAN Xiao-gang

(College of Electronic Information and Control Engineering, Beijing University of Technology, Beijing 100124, China)

Abstract: For the navigation of Pioneer3-DX mobile robot in unknown environment, we propose a self-navigation strategy with learning reinforcement, and develop the navigation algorithm based on the dynamical neural network. The dynamically self-organizing neural network can automatically adjust its structure according to the complexity of the working environments of the mobile robot to realize the mapping between environmental states and robot actions, effectively avoiding the dimension explosion in learning reinforcement. Simulations and real robot navigation experiments are carried out; results show that the proposed method is effective in applications. It gives a better navigation performance than that of the artificial potential-field method.

Key words: mobile robot; navigation; dynamically structured neural network

1 引言(Introduction)

移动机器人是一个集环境感知, 动态决策, 行为控制和执行等多种功能的复杂系统. 随着移动机器人在航空航天、医疗服务、工业生产等方面应用的不断扩展, 移动机器人成为机器人学研究的一个热点[1]. 在机器人的各项研究和应用中, 导航是最基本也是最重要的问题[2]. 由于机器人工作环境的不可预见性和易变性, 需要机器人尽可能的适应环境, 以不断提高学习能力和决策能力. Q学习具有不需要环境模型的特点, 并且可以在线学习, 在机器人导航中得到了广泛的应用[3]. 经典的Q学习方法需要将状态空间和动作空间离散划分, 这样很容易产生维数爆炸问题[3]. 而且对离散环境状态的学习会导致泛化能力较差[4]. 针对以上经典 Q 学习方法的缺点, 不少文献提出了神经网络和强化学习结合使用的方法[5~7]. 但目前使用的神经网络结构固定, 信息处理的能力受到了很大限制. 为此, 提出了一种动态结构自组织神经网络DSSONN(dynamic structure self-organizing neural network)的设计和训练方法. 该网络具有可变的隐层结构, 可以动态地插入和删除网络节点. 根据信息处理的需要, 动态地实现网络结构的调整和优化. 通过对Pioneer3-DX移动机器人进行实物测试证明了该方法的有效性. 使用动态结构自组织神经网络和强化学习结合, 完成了机器人在楼道环境下导航.

2 移动机器人导航系统设计(Design of mobile robot navigation system)

2.1 Pioneer3-DX 移动机器人(Pioneer3-DX mobile robot)

本文以斯坦福大学研制的Pioneer3-DX轮式移动

收稿日期: 2008−12−23; 收修改稿日期: 2009−04−28.

基金项目: 国家自然科学基金资助项目(60674066,60873043); 国家“863”计划资助项目(2007AA04Z160); 北京市自然科学基金资助项目(4092010); 教育部博士点基金资助项目(200800050004).

机器人为对象, 研究移动机器人导航技术. 该机器人配有两个声纳环, 共16个声纳传感器, 可以探测到以机器人为中心的0∼360°范围内的障碍物, 声纳最大探测距离是5000 mm.

2.2 移动机器人自主导航系统设计(Design of mobile robot auto navigation system)

移动机器人自主导航系统主要由检测模块、导航模块、控制模块和执行模块组成, 如图1所示.

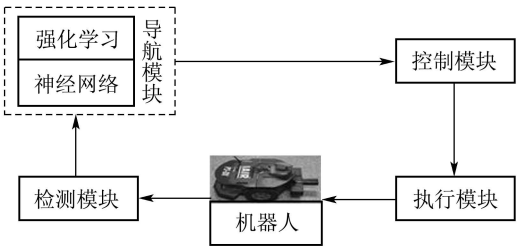

图 1 系统结构图

Fig. 1 Block graph of the system

导航模块是该机器人系统中最核心的部分, 本文采用强化学习和动态结构自组织神经网络完成机器人导航任务. 结构自组织神经网络通过训练完成对强化学习中状态和动作之间的关系的映射. 下面将对导航模块作详细的介绍.

3 移动机器人导航算法设计(Design of mobile robot navigation algorithm)

3.1 基于强化学习的导航算法(Navigation algorithm based on reinforcement learning)

定义 1 机器人的状态空间S:

$$S = \{d_{\rm l}, d_{\rm f}, d_{\rm r}, d_{\rm g}, \theta\}. \tag{1}$$

其中: $d_{\rm l}$是机器人左侧距障碍物的距离, $d_{\rm f}$是机器人前方距障碍物的距离, $d_{\rm r}$是机器人右侧距障碍物的距离, $d_{\rm g}$为机器人与目标点之间的距离, θ为机器人当前方向和目标点的夹角. 这5个量作为状态空间的5个维度. 机器人和障碍物的距离定义为

$$d = \min(d_{\rm l}, d_{\rm f}, d_{\rm r}). \tag{2}$$

定义 2 机器人的动作空间A:

$$A = \{a_1, a_2, a_3, a_4, a_5\}. \tag{3}$$

其中:

a_1: 机器人转动 $+15°$同时前进100 mm;

a_2: 机器人转动 $-15°$同时前进100 mm;

a_3: 机器人转动 $+10°$同时前进100 mm;

a_4: 机器人转动 $-10°$同时前进100 mm;

a_5: 机器人前进100 mm.

t时刻机器人的状态S_t为1个五维向量:

$$S_t = \{d_{\rm l}(t), d_{\rm f}(t), d_{\rm r}(t), d_{\rm g}(t), \theta(t)\}. \tag{4}$$

其中: $d_{\rm l}(t)$, $d_{\rm f}(t)$, $d_{\rm r}(t)$分别表示在t时刻机器人左侧、前方、右侧障碍物的距离. $d_{\rm g}(t)$表示t时刻机器人距目标点的距离, $\theta(t)$表示t时刻机器人前进方向和目标点的夹角, 如图2所示.

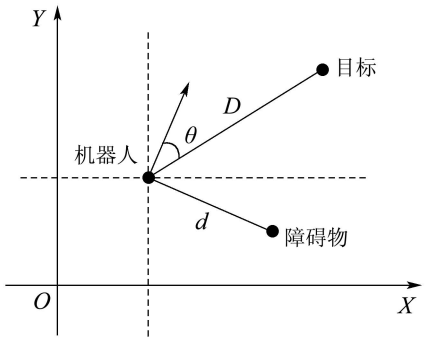

图 2 机器人、障碍物和目标点的位置示意图

Fig. 2 Position relationship among robot, obstacle and goal

1) 机器人和目标点的距离缩小, 表现为

$$\begin{cases} r_{\rm g}(t) = d_{\rm g}(t+1) - d_{\rm g}(t), \\ r_{\rm g}(t) < 0. \end{cases} \tag{5}$$

2) 机器人和障碍物之间的距离扩大, 表现为

$$\begin{cases} r_{\rm d}(t) = d(t+1) - d(t), \\ r_{\rm d}(t) > 0. \end{cases} \tag{6}$$

3) 机器人朝着目标点运动, 表现为

$$\begin{cases} r_\theta(t) = |\theta(t+1)| - |\theta(t)|, \\ r_\theta(t) < 0. \end{cases} \tag{7}$$

4) 机器人尽可能的沿着通道中线行走, 表现为

$$\begin{cases} r_{\rm m}(t) = |d_{\rm l}(t+1) - d_{\rm r}(t+1)| - |d_{\rm l}(t) - d_{\rm r}(t)|, \\ r_{\rm m}(t) < 0. \end{cases} \tag{8}$$

机器人的强化信号定义为

$$r(t) = -\alpha r_{\rm g}(t) + \beta r_{\rm d}(t) - \gamma r_\theta(t) - \eta r_{\rm m}(t). \tag{9}$$

其中: 一般取$0 < \alpha, \beta, \gamma, \eta < 1$, 可根据具体情况设定.

定义 3 时刻t环境状态为S_t时, 动作a_k对应的Q值为$Q_t(S_t, a_k)$, 其中$t = 1, 2, 3, \cdots, k = 1, 2, 3, 4, 5$. Q函数定义为在状态S_t时执行动作a_k, 且此后按最优动作序列执行的强化信号折扣和.

$$Q_t(S_t, a_t) = r(t) + \rho \max_{a_k \in A} Q_{t-1}(S_{t+1}, a_{\rm k}). \tag{10}$$

式(10)在最优策略的前提下才成立, 在学习阶段, 上

式两端不相等, 误差为

$$\Delta Q_t(S_t,a_k)=r(t)+\rho\max Q_{t-1}(S_{t+1},a_k)-Q_{t-1}(S_t,a_k), \tag{11}$$

$$Q_t(S_t,a_k)=Q_{t-1}(S_t,a_k)+\mu\Delta Q_t(S_t,a_{\rm k}). \tag{12}$$

其中: ρ为折扣因子, μ为学习率, 可根据情况选取.

3.2 动态结构神经网络设计(Design of dynamically structured neural network)

经典Q学习存在的一个问题是动作–状态空间组合的维数爆炸和泛化能力较弱. 根据Pioneer3-DX移动机器人系统的特点, 采用动态结构自组织神经网络DSSONN(dynamically structured self-organizing neural network)映射Q学习中状态和动作之间的关系. 神经网络的输入层由5个神经元组成, 分别是5个状态维度$d_{\rm l},d_{\rm f},d_{\rm r},d_{\rm g},\theta$; 输出层由5个神经元组成, 对应动作空间的5个动作a_1,a_2,a_3,a_4,a_5的Q值. 网络初始化为单隐层结构, 且该隐层只有一个神经元, 权值初始化为较小的随机数. 用$\mathrm{Net}(l,n)$表示神经网络的结构. 其中参数l代表网络有l个隐层, 参数n表示网络紧邻输出层的隐层含有n个神经元节点. 当前时刻t网络$\mathrm{Net}(l,n)$的训练误差表示为:

$$E(l,n)=\frac{1}{2}\sum_{m=t-\lambda}^{t}\sum_{k=1}^{5}(\Delta Q_{\rm m}(S_{\rm m},a_k))^2. \tag{13}$$

其中$\Delta Q_{\rm m}(S_{\rm m},a_k)$的计算参照式(11), 是$\lambda$一个正整数, 表示要累计$\lambda$次误差. 这样, 网络会滚动计算误差, 进行训练调整. 输入层和输出层神经元的个数在训练的过程中不发生变化. 隐含层的层数以及每个隐含层所含的神经元的个数在训练中动态调整. 若当前网络结构$\mathrm{Net}(l,n)$, 满足式(14), 则向网络中紧邻输出层的隐层插入新的神经元节点, 网络的结构变为$\mathrm{Net}(l,n+1)$, 如图 3 中虚线所示.

$$\begin{cases}\mid E(l,n)-E(l,n-p)\mid/E(l,n)>\xi,\\ E(l,n)>E_0.\end{cases} \tag{14}$$

其中: E_0为用户设定的误差水平, ξ为1个较小的正数, 表示误差变化的显著程度, p为1个较小的正整数. 插入新节点的同时, 建立该节点与邻接层神经元的权值连接, 新产生的权值初始化为$[-0.5,0.5]$之间的随机数, 即

$$w_{ij}=\mathrm{random}(-0.5,0.5). \tag{15}$$

若在同一隐层内连续插入p个神经元仍不能使误差显著减小, 即:

$$\begin{cases}\mid E(l,n)-E(l,n-p)\mid/E(l,n)\leqslant\xi,\\ E(l,n)>E_0.\end{cases} \tag{16}$$

则向网络中紧邻输出层的位置插入新的隐层, 并向该新隐层中插入新的神经元节点, 如图4所示, 网络的结构变为$\mathrm{Net}(l+1,1)$.

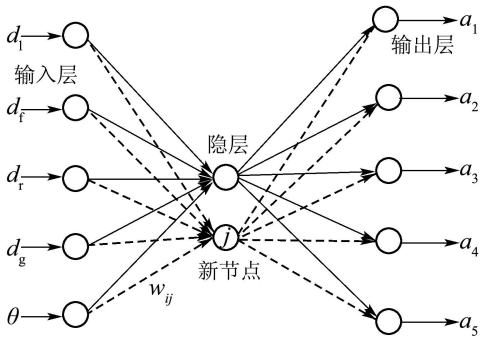

图 3　插入新节点过程示意图

Fig. 3　Process of inserting new node to the net

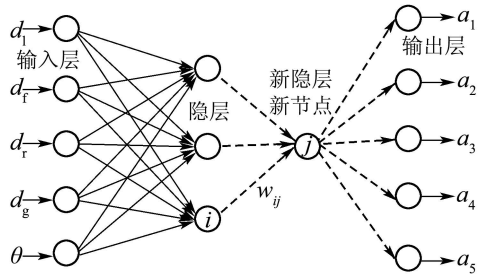

图 4　插入新隐层过程示意图

Fig. 4　Process of inserting new hidden layer

新插入的神经元节点和相邻隐层的神经元全连接, 并且和输出层全连接. 权值同样按照式(15)的方式初始化. 在每次插入神经元之后, 网络都要进行足够次数的训练, 直到网络的误差减小不再明显. 当网络的误差要求达到时, 此时的网络中可能会存在一些冗余节点, 这些节点对网络的贡献很小, 却花费了较长的训练时间, 而且容易使网络的泛化能力变差. 因此, 需要将这些节点剔除. Joog-Sock LEE提出一种叫做Imf(impact factor)的析构算法[8]. 该方法的核心思想是将每个神经元对其后层神经元的影响量化. 不失一般性, 图5所示为网络的一部分. 下一层的神经元j对应于第m个样本的输入x_j^m可以表示为

$$x_j^m=\sum_i w_{ij}y_i^m+b_j, \tag{17}$$

$$x_j^m=\sum_i w_{ij}(y_i^m-\bar{y}_i)+\sum_i w_{ij}\bar{y}_i+b_j. \tag{18}$$

其中: b_j是下一层神经元j的偏置, y_i^m为神经元i对第m个样本的输出, $\bar{y}_i$是神经元i对所有输入样本对应输出的平均值. 神经元i对下一层神经元的总的贡献C为

$$C = \sum_j w_{ij}^2 (y_i^m - \bar{y_i}). \tag{19}$$

神经元i的Imf可以定义为

$$\mathrm{Imf}_i = \sum_j w_{ij}^2 \sigma_i^2. \tag{20}$$

其中σ_i^2是神经元i的输出量的方差.

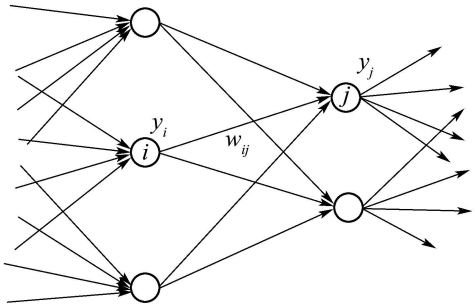

图 5 神经元的Imf示意图

Fig. 5 Imf of neural nodes

3.3 系统导航算法(Navigation algorithm of system)

在强化学习的初期, 学习的主要任务是对环境探索, 因此要求动作选择的随机性大一些, 在强化学习的后阶段, 动作选择的随机性应小一些, 以便函数收敛. 为此, 这里采用Boltzmann机进行退火运算. 设T_0为初始温度值, 随着时间t的增加, T由T_0衰减, 参数τ用来控制退火的速度. 导航学习算法如下:

1) 初始化神经网络权值为较小的随机数. 设置退火初始温度T_0和退火参数τ, 置初始值t.

2) 读取当前状态信息$d_{\mathrm{l}}(t)$, $d_{\mathrm{f}}(t)$, $d_{\mathrm{r}}(t)$, $d_{\mathrm{g}}(t)$, $\theta(t)$并输入到神经网络. 计算网络的输出$Q_t(s_t, a_k)$, $k = 1, 2, 3, 4, 5$.

3) 计算动作a_k被选择的概率$P(a_k)$:

$$P(a_k) = \frac{\mathrm{e}^{Q_t(s_t, a_{\mathrm{k}})/(T_0 t^{-\frac{1}{\tau}})}}{\sum\limits_{a_{\mathrm{k}} \in A} \mathrm{e}^{Q_t(s_t, a_{\mathrm{k}})/(T_0 t^{-\frac{1}{\tau}})}}. \tag{21}$$

采用轮盘赌博方法, 从动作空间选择一个动作执行.

4) 执行选中的动作a_k, 读取当前状态信息$d_{\mathrm{l}}(t+1)$, $d_{\mathrm{f}}(t+1)$, $d_{\mathrm{r}}(t+1)$, $d_{\mathrm{g}}(t+1)$, $\theta(t+1)$, 按式(9)计算$r(t)$, 按照式(11)计算$\Delta Q_t(s_t, a_{\mathrm{k}})$, 计算网络误差:

$$E(l, n) = \frac{1}{2} \sum_{h=0}^{H} \sum_{k=1}^{5} (\Delta Q_h(s_h, a_k))^2. \tag{22}$$

其中H为一个较大的正整数, 可以根据实际情况设定.

5) 调整网络结构和权值.

6) 若$E(l, n) < \delta$, 转到7) , 否则转到2) . 其中δ是由用户设定的一个最终网络的训练误差.

7) 保存网络结构和权值, 退出学习过程.

4 移动机器人导航实验研究(Experiment of mobile robot navigation)

选用Pioneer3-DX为实验用机器人, 在实验室所在的楼道做真实实验. 走廊宽度为1740 mm, 其中最窄的地方宽度为1540 mm, 形状如图6, 7所示. 以楼层西南角为全局坐标的原点, 设机器人起始点的坐标为Start(6000 mm, 870 mm), 目标点的坐标为Goal(32030 mm, 9345 mm).

4.1 仿真实验(Simulation experiments)

根据走廊环境特征和机器人物理特点, 对一些重要参数进行合理选择, 本实验中取机器人强化学习参数$\alpha = 0.1$, $\beta = 0.1$, $\gamma = 0.15$, $\eta = 0.05$, 折扣因子$\rho = 0.12$, 学习率$\mu = 0.1$. 神经网络训练相关的参数$\xi = 0.03$, $E_0 = 0.1$, $p = 3$. 这里比较了结合使用DSSONN和强化学习的机器人导航效果和基于人工势场法的机器人导航效果. 图6, 7所示为仿真实验结果. 显然, 仿真结果表明前者较后者具有更好的效果.

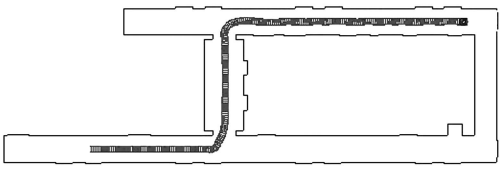

图 6 基于DSSONN强化学习的实验结果

Fig. 6 Experiment result based on DSSONN and reinforcement learning

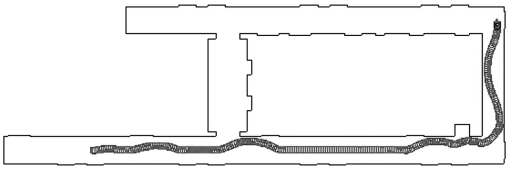

图 7 基于人工势场法的实验结果

Fig. 7 Experiment result based on artificial potential field method

为了说明DSSONN的结构自组织特性, 进行了10次独立的实验, 网络隐层节点的数目如图8所示. 显然, 由于网络根据信息处理的需要, 对结构进行了自组织调整, 因此, 每次实验得到的隐层神经元个数有所不同, 但总体相对稳定. 而要达到同样的效果, 需要通过尝试设计BP网络的结构. 这也正体现了结构自组织的特性.

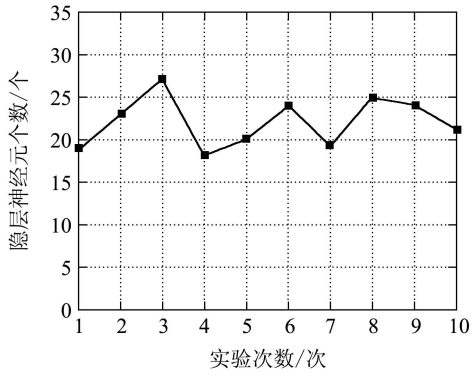

图 8　DSSONN 隐层神经元节点数

Fig. 8　Number of hidden layer nodes in DSSONN

4.2　实物实验(Real experiments)

Pioneer3-DX机器人是一款轻巧型移动机器人, 适合于楼道等狭小环境下的导航实验. 机器人配备了声纳传感器, 里程记, 电子罗盘, 两轮具有独立的驱动电机. 为了形象的显示机器人导航的效果, 在实物实验中采集了机器人运动轨迹的坐标数据. 根据采集的实际数据分别绘出了采用DSSONN强化学习法和人工势场法导航的路径轨迹, 如图9所示.

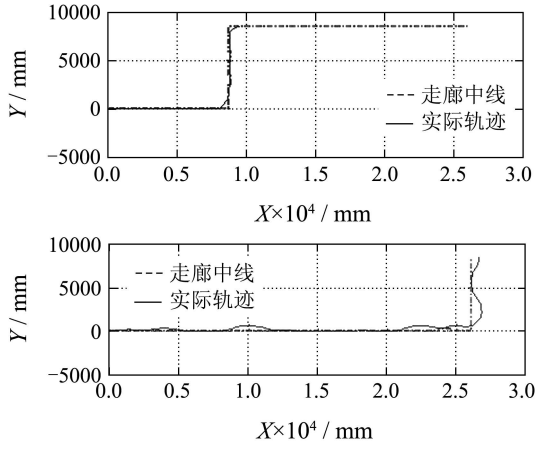

图 9　移动机器人导航轨迹

Fig. 9　Navigation trajectory of mobile robot

从实际数据生成的路径轨迹看出, 基于DSSONN强化学习的导航轨迹和走廊中线差距很小, 机器人中心偏离通道中线的距离最大为182 mm, 而基于人工势场法的导航路径常发生摆动现象, 机器人中心距离通道中线的距离最大达到591 mm. 这是由于机器人在人工势场中存在势场的平衡点和弯曲段, 机器人的运行轨迹完全靠局部的势场力作用的结果. 而强化学习是一个和环境交互作用的过程, 通过学习, 可以在全局范围内找到可行的路径.

5　结论(Conclusion)

实现移动机器人自主导航是目前普遍关心的问题. 通过强化学习策略实现移动机器人导航是一个有效的方法, 采用动态结构自组织神经网络有效地解决了强化学习中的维数爆炸问题. 该网络可以根据信息处理的需要动态调整其规模和结构. 结合使用动态结构自组织神经网络和强化学习完成了Pioneer3-DX 移动机器人在走廊环境下的导航. 通过仿真和实物实验证明了该方法的可行性和有效性.与常用的人工势场法比较, 该方法具有更好的导航效果.

参考文献(References):

[1] BAUER A, WOLLHERR D, BUSS M. Human-robot collaboration: a survey[J]. *International Journal of Humanoid Robotics*, 2008, 5(1): 47 – 66.

[2] JAN G E, CHANG K Y, PARBERRY I. Optimal path planning for mobile robot navigation[J]. *IEEE-ASME Transactions on Mechatrioics*, 2008, 13(4): 451 – 460.

[3] BUSONIU L, BABUSKA R, DE SCHUTTER B. A comprehensive survey of multiagent reinforcement learning[J]. *IEEE Transactions on Systems, Man and Cybernetics*. 2008, 38(2): 156 – 172.

[4] CARRERSA M, YUB J K, BATLLE J, et al. Application of SONQL for real-time learning of robot behaviors[J]. *Robotics and Autonomous System*, 2007, 55(8): 628 – 642.

[5] ARLEO A, SMERALDI F, GERSTNER W. Cognitive navigation based on nonuniform Gabor space sampling unsupervised growing networks and reinforcement learning[J]. *IEEE Transactions on Neural Networks*, 2004, 15(3): 639 – 652.

[6] MA X L, LIKHAREV K K. Global reinforcement learning in neural networks[J]. *IEEE Transactions on Neural Networks*, 2007, 18(2): 573 – 577.

[7] TAN A H, LU N, XIAO D. Integrating temporal difference methods and self-organizing neural networks for reinforcementLearning with delayed evaluative feedback[J]. *IEEE Transactions on Neural Networks*, 2008, 19(2): 230 – 244.

[8] LEE J S, LEE H, KIM J Y, et al. Self-organizing neural networks by construction and pruning[J]. *IEICE Transactions on Information & Systems*, 2004, E87-D(11): 2489 – 2498.

作者简介:

乔俊飞　(1968—), 男, 博士, 教授, 博士生导师, 目前研究方向复杂过程建模与控制、计算智能与智能优化控制等, E-mail: isibox@sina.edu.cn;

樊瑞元　(1982—), 男, 硕士研究生, 目前研究方向神经网络优化设计、移动机器人导航, E-mail: fanruiyuan@gmail.com;

韩红桂　(1983—), 男, 博士研究生, 主要研究方向为复杂过程建模与控制、模式识别与智能系统等, E-mail: Rechardhan@sina.com;

阮晓钢　(1958—), 男, 教授, 研究领域为机器人、自动控制与人工智能等, E-mail: adrxg@bjut.edu.cn.