

基于粗神经网络的并行自适应滤波算法^{*}

许 浒¹, 姜长生¹, 辅小荣^{1,2}

(1. 南京航空航天大学 自动化学院, 江苏 南京 210016; 2. 盐城工学院 电信学院, 江苏 盐城 224003)

摘 要: 讨论了机动目标跟踪问题, 针对机动目标并行自适应滤波特点, 结合粗神经网络特有的同一神经元可以双输入的特点, 利用全状态反馈, 提出了基于“当前”统计模型的粗神经网络并行自适应滤波算法, 仿真表明该算法对机动目标具有较强的跟踪能力。

关键词: 粗神经网络; 机动目标; 跟踪算法

中图分类号: TP274

文献标识码: A

文章编号: 1671-5322(2005)01-0027-04

机动目标跟踪问题一直是人们关心的热点问题。不少学者对机动目标模型进行了广泛而深入的研究, 提出了许多机动目标跟踪算法^[1~3]。其中周宏仁提出的机动目标“当前”统计模型以及基于该模型的自适应滤波算法已为越来越多的人认同, 并得到了广泛的应用。

本文受文献[4]启发, 利用全状态反馈, 提出一种基于粗神经网络的机动目标信息融合并行自适应跟踪算法, 该算法充分考虑到网络输入变量的可能变化范围, 利用粗神经网络特有的功能, 对同一输入变量以双值输入进行泛化处理, 通过离线训练, 结合特征向量实际变化在线自动调节网络输出, 以方差自适应形式保持对目标状态的精确跟踪。仿真表明该算法有较好的跟踪机动目标能力。

1 粗神经网络结构

粗神经网络^[4]与传统神经网络的最大区别是神经元的组成及其连接方式不同, 具体说就是用粗糙神经元来代替传统的神经元。传统神经网络一个节点仅一个神经元, 粗神经网络的一个节点有两个神经元, 即上神经元和下神经元。上神经元表示表示粗糙集中的上近似定义, 其符号为 $\bar{r}$ 或 U_i 或 U_o 等; 下神经元用来表示粗糙集中的下

近似定义, 其符号为 $\underline{r}$ 或 L_i 或 L_o 等。上、下神经元分别与下一层的每个节点的上、下神经元互相连接。以三层为例, 其结构如图 1。

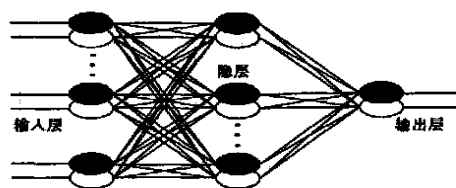


图 1 粗神经网络结构图

Fig. 1 Structure of rough neural network

输入层节点为 p 个, 隐层节点为 q 个, 输出节点为 1 个, 每个节点的神经元都由上下层组成; 输入层神经元只完成数据的输入功能, 隐层神经元和输出层神经元的输入按下列公式计算(以第 j 个神经元为例):

$$In_{Uj} = \sum_{i=1}^p (W_{UUij} \times Out_{Ui} + W_{LUij} \times Out_{Li})$$

$$In_{Lj} = \sum_{i=1}^p (W_{LLj} \times Out_{Li} + W_{ULij} \times Out_{Ui})$$

输出按下列公式计算:

$$Out_{Uj} = \max\{f_U(In_{Uj} - \theta_{Uj}), f_L(In_{Lj} - \theta_{Lj})\}$$

$$Out_{Lj} = \min\{f_U(In_{Uj} - \theta_{Uj}), f_L(In_{Lj} - \theta_{Lj})\}$$

其中 i 为和 j 神经元相连的前一层中的神经元, θ_{Uj} 和 θ_{Lj} 为粗神经网络 j 上下两层的阈值, 函数 f_U 和

* 收稿日期: 2004-11-01

基金项目: 航空一集团基金(01D52025)、航空二集团基金(01C52010)资助项目。

作者简介: 许 浒(1969-), 男, 江苏宿迁人, 南京航空航天大学硕士研究生, 主要研究方向为导航制导与控制。

f_L 分别为上下两的传递函数, 我们取其为 Sigmoid

$$\text{函数即: } f_U(x) = f_L(x) = \frac{2}{1 + e^{-x}} - 1$$

网络的最终输出为输出层粗神经元上、下两层输出的均值。由于每个节点均有两神经元, 因此这种神经网络不仅能处理单值输入问题, 而且可以处理双值输入问题。

2 机动目标“当前”统计模型

根据文献[3]观点, 当目标正以某一加速度机动时, 下一时刻的加速度取值是有限的, 且只能在“当前”加速度的邻域内, 因此提出了机动目标“当前”统计模型。该模型本质上是非零均值时间相关模型, 其机动加速度的“当前”概率密度用修正的瑞利分布描述, 均值为“当前”加速度预测值, 随机机动加速度在时间轴上仍符合一阶时间相关过程, 机动目标“当前”统计模型的状态方程为:

$$\begin{bmatrix} \ddot{x} \\ \dot{x} \\ x \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & -a \end{bmatrix} \begin{bmatrix} \ddot{x}(t) \\ \dot{x}(t) \\ x(t) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ a \end{bmatrix} \bar{a} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega(t) \quad (1)$$

式中 $x(t)$ 、 $\dot{x}(t)$ 和 $\ddot{x}(t)$ 分别为目标的位置、速度和加速度。

$$\text{观测方程为: } Y(k) = H(k)X(k) + V(k) \quad (2)$$

其中当令含有噪声的目标位置数据可观测时, 有

$$H(k) = [1 \quad 0 \quad 0]$$

$V(k)$ 是均值为零, 方差为 $R(k)$ 的高斯观测噪声。

3 自适应卡尔曼滤波算法

根据状态方程式(1)和观测方程(2), 如果把加速度的观测值看作为 kT 瞬时的“当前”加速度即随机机动加速度的均值, 根据标准卡尔曼滤波方程, 就可得到加速度的均值自适应算法。

其状态更新方程

$$\hat{x}(k|k) = \hat{x}(k|k-1) + K(k)[Z(k) - h(\hat{x}(k|k-1))]$$

$$P(k|k) = [I - K(k)H(k)]P(k|k-1)$$

状态预测方程为

$$\hat{x}(k|k-1) = \Phi(k, k-1)\hat{x}(k-1|k-1) + U(k)\bar{a}(k)$$

$$P(k|k-1) = \Phi(k, k-1)P(k-1|k-1)$$

$$\Phi'(k, k-1) + Q(k-1)$$

滤波增益方程

$$K(k) = P(k|k-1)H'(k) [H(k)P(k|k-1)H'(k) + R]^{-1}$$

由前文假定知: $\bar{a}(k) = \hat{x}(k/k-1)$

根据“当前”统计模型理论有

$$\sigma_a^2 = \begin{cases} \frac{4-\pi}{\pi}(a_{\max} - \bar{a})^2 & (\bar{a} \geq 0) \\ \frac{4-\pi}{\pi}(a_{-\max} - \bar{a})^2 & (\bar{a} < 0) \end{cases}$$

$a_{\max}$ 为目标的可能最大加速度, α 是机动频率。

4 基于粗神经网络的并行自适应跟踪算法

由于上述算法采用机动加速度方差自适应, 因此无需机动检测便能较好地响应目标的机动情况, 大量实验表明该算法在跟踪机动目标时具有良好的跟踪结果^[3]。同时, 实验中也发现一旦最大加速度确定以后, 跟踪的精度与实际目标的加速度具有直接关系, 目标机动加速度与最大加速度越接近, 跟踪的精度越高, 反之则下降。为此, 文献[4]又提出了基于 BP 神经网络的并行自适应跟踪算法。具体结构如下:

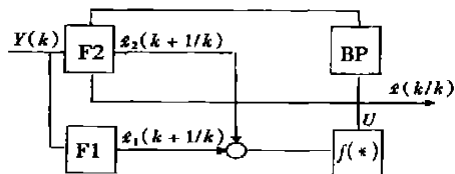


图 2 基于 BP 网络的并行自适应跟踪算法结构

Fig. 2 BP-based Parallel adaptive tracking structure

图 2 中滤波器 F1 采用“当前”统计模型, 选取较大的加速度方差 σ_{a1} 以适应目标的各种运动变化, 并保持对机动的快速响应, 滤波器 F2 也采用“当前”统计模型, 但其加速度方差 σ_{a2} 与 F1 中的相应项不同, σ_{a2} 的自动调整是综合 F1 和 F2 的全部信息即全状态反馈信息 (u_1, u_2, u_3), 并利用 BP 网络输出 $O_{BP} \in (0, 1)$, 使得 $\sigma_{a2} = O_{BP} \sigma_{a1}$ 适应目标的强弱机动和非机动变化。即 F2 以足够准确的系统方差适应目标的运动变化并保持对目标状态的高精度跟踪。F2 的输出即为目标的状态估计。其中 u 是来自两个滤波器的全状态的反馈信息, 具体是

$$u_1 = f_1(*) = \frac{[\hat{x}_1(k+1|k) - \hat{x}_2(k+1|k)]^2}{s_1(k+1) + s_2(k+1)}$$

$$u_2 = f_2(*) = \frac{[\dot{x}_1(k+1|k) - \dot{x}_2(k+1|k)]^2}{p_{1,22}(k+1) + p_{2,22}(k+1)}$$

$$u_3 = f_3(*) = \frac{[\ddot{x}_1(k+1|k) - \ddot{x}_2(k+1|k)]^2}{p_{1,33}(k+1) + p_{2,33}(k+1)}$$

式中 $\hat{x}_i(k+1|k)$, $\dot{x}_i(k+1|k)$, $\ddot{x}_i(k+1|k)$ ($i = 1, 2$) 分别为两滤波器的位置、速度、加速度预测值; $s_1(k+1)$ 和 $s_2(k+1)$ 分别为两滤波器的位置残差方差; $p_{i,22}(k+1)$ 和 $p_{i,33}(k+1)$, ($i = 1, 2$) 分别为两滤波器的速度和加速度预测方差。可见 u 为两滤波器状态预测误差的范数, 对噪声有一定的抑制作用。但就某一时刻具体的 u 值而言并

$$\alpha = \frac{|\hat{x}_1(k+1|k) - \hat{x}_2(k+1|k) + 3\sqrt{\epsilon_1} + 3\sqrt{\epsilon_2}|}{s_1(k+1) + s_2(k+1)}$$

$$\beta_1 = \frac{|\dot{x}_1(k+1|k) - \dot{x}_2(k+1|k) - 3\sqrt{\epsilon_1} - 3\sqrt{\epsilon_2}|}{s_1(k+1) + s_2(k+1)}$$

$$\alpha_2 = \frac{|\dot{x}_1(k+1|k) - \dot{x}_2(k+1|k) + 3\sqrt{p_{1,22}(k+1)} + 3\sqrt{p_{2,22}(k+1)}|}{p_{1,22}(k+1) + p_{2,22}(k+1)}$$

$$\beta_2 = \frac{|\dot{x}_1(k+1|k) - \dot{x}_2(k+1|k) - 3\sqrt{p_{1,22}(k+1)} - 3\sqrt{p_{2,22}(k+1)}|}{p_{1,22}(k+1) + p_{2,22}(k+1)}$$

$$\alpha_3 = \frac{|\ddot{x}_1(k+1|k) - \ddot{x}_2(k+1|k) + 3\sqrt{p_{1,33}(k+1)} + 3\sqrt{p_{2,33}(k+1)}|}{p_{1,33}(k+1) + p_{2,33}(k+1)}$$

$$\beta_3 = \frac{|\ddot{x}_1(k+1|k) - \ddot{x}_2(k+1|k) - 3\sqrt{p_{1,33}(k+1)} - 3\sqrt{p_{2,33}(k+1)}|}{p_{1,33}(k+1) + p_{2,33}(k+1)}$$

式中 ϵ_i 为 $\hat{x}_i(k+1|k)$ 的方差 ($i = 1, 2$)。

由标准卡尔曼滤波方程知滤波残差(新息)向量被定义为

$$d(k) = Y(k) - H(k)\hat{x}(k/k-1)$$

故滤波器距离残差的方差按下式计算

$$S(k) = H(k)P(k/k-1)H^T(k) + R(k)$$

通过对目标各种机动水平和运动形式的大量仿真发现, 特征向量的变化是有规律的, 无论目标进行何种机动 α_i, β_i ($i = 1, 2, 3$) 都始终在一定范围内变动。显然, 如果每一时刻都以 α_i, β_i ($i = 1, 2, 3$) 作为同一神经元的输入, 则可削除状态预测噪声对其预估值带来的影响, 从而缩短对机动目标的跟踪延时, 提高跟踪精度。即双值输入比 u_i 以单值输入更为稳定、更有效。

粗神经网络是通过离线训练, 样本选取原则同 BP 网络相似, 当所有特征变量处于低值时, 认为目标处于非机动状态, 粗神经网络的输出 $netout$ 为一接近于零的值; 当所有特征变量处于高值时, 认为目标处于强机动状态, 网络输出 $netout$ 为一

未充分考虑预测值 $\hat{x}_i(k+1|k)$, $\dot{x}_i(k+1|k)$, $\ddot{x}_i(k+1|k)$, ($i = 1, 2$) 本身噪声的影响。如我们把两个预测值都放大到其瞬时可能的取值范围, 则能更好地抑制本身噪声的影响。为此我们对上述算法作如下修改, 用粗神经网络来综合全状态信息和噪声信息, 即将 BP 网络换成粗神经网络, 输入层节点为 3 个, 隐层节点为 7 个, 输出节点为 1 个, 每个节点的神经元都由上下层组成; 输入层神经元只完成数据的输入功能, 3 对数据分别是 (α_1, β_1) , (α_2, β_2) , (α_3, β_3) , 隐层神经元和输出层神经元的输入分别按下列公式计算。其中:

接近于 1 的值; 当目标在强机动和非机动之间变化时, 认为目标处于弱机动状态, 网络输出一个 $(0, 1)$ 中的值, 因为特征向量各元素具有不同的敏感性, 不一定同时达到高值, 为使滤波器 F2 能较快地响应机动, 当特征变量中 1 个或 2 个达到高值时, 也应设置一些较大的学习样本值, 即网络的输出 $netout$ 设置为一些较大的值。

5 仿真实验

为检验本算法的有效性, 我们在一维情况下, 仿真了基于粗神经网络的自适应系统对弱加速、强加速直线运动以及变加速度目标的响应特性, 同时与基于 BP 神经网络的自适应系统进行比较 (—为基于 BP 网络, …为基于粗神经网络)。

仿真中我们分别取目标的初速度为 300 m/s, 初始位移为 0 km, 弱机动目标加速度为 5 m/s², 强机动目标加速度为 40 m/s², 变加速是加速度从 5 m/s² 阶跃到 25 m/s²。

从图 3 ~ 图 5 可明显看出, 无论是对弱机动

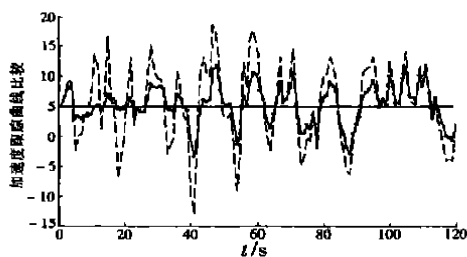


图 3 弱机动时加速度跟踪曲线

Fig. 3 Acceleration tracking curve of weak maneuvering

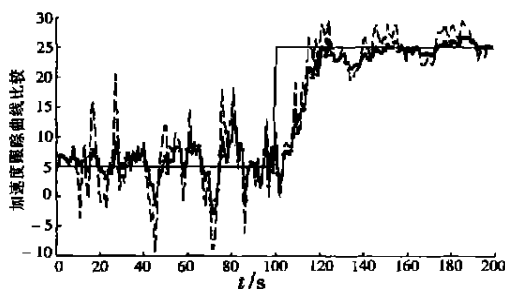


图 5 加速度阶跃变化时跟踪曲线

Fig. 5 Acceleration tracking curve of step maneuvering
络的并行自适应滤波系统对目标的跟踪,都要优于基于 BP 神经网络的并行自适应滤波系统。

6 结论

理论分析和仿真结果都表明:基于粗神经网络的并行自适应滤波系统对机动目标的加速度的估计,比基于 BP 神经网络的并行自适应滤波系统更精确,即其滤波算法跟踪机动目标的能力更强,效果更佳。前者与后者相比较,基于粗神经网络的并行自适应滤波算法不仅缩短了跟踪机动目标的动态时延,而且还提高了跟踪精度,目标机动加速度越大这种优势就越为明显。

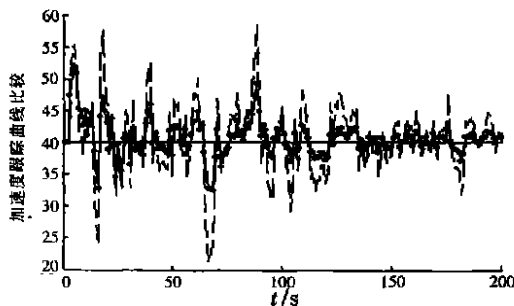


图 4 强机动时加速度跟踪曲线

Fig. 4 Acceleration tracking curve of strong maneuvering
目标、强机动目标还是变加速目标,基于粗神经网络

参考文献:

- [1] 刘作良,冯新喜. C³I 信息获取系统[M]. 西安:陕西科学技术出版社,1998.
- [2] Chong C-Y, Moris S, Barker W H. Architectures and algorithms track association and fusion[J]. IEEE—AES system magazine, 2000, (2): 5—13.
- [3] 周宏仁,敬忠良,王培德. 机动目标跟踪[M]. 北京:国防工业出版社,1991.
- [4] 张兆礼. 神经网络数据融合算法研究[D]. 哈尔滨:哈尔滨工业大学,2001.

Rough Neural Network Based Parallel Adaptive Filtering Algorithm

XU Hu¹, JIANG Chang—sheng¹, FU Xiao—rong^{1,2}

1. College of Automation, Nanjing University of Aeronautics and Astronautics, Jiangsu Nanjing 210016 China
2. College of Electronics and Communication, Yancheng Institute of technology, Jiangsu Yancheng 224003, China

Abstract: This paper proposes “Current” statistics mode based rough neural network parallel adaptive filtering algorithm with reference to its features with two inputs in the same neural element. The results of simulation shows that this algorithm has strong tracking ability to moving objectives.

Keywords: rough neural network; moving objectives; tracking algorithm