

基于有向通信拓扑的高阶分布式一致性算法

彭焕新^{1,2*}, 戚国庆², 盛安冬²

(1. 南京工业职业技术学院 机械工程学院, 南京 210046; 2. 南京理工大学 自动化学院, 南京 210094)

(* 通信作者电子邮箱 penghx@niit.edu.cn)

摘要:为了提高有向通信拓扑下分布式一致性算法的收敛速度,提出了一种基于有向通信拓扑的高阶分布式一致性算法。该算法通过有向单跳通信,利用有向二跳邻接节点的前多步信息提高分布式一致性算法的收敛速度。对有向通信拓扑下该算法的收敛性能和收敛速度进行了分析和仿真比较。结果显示,该算法在满足一定条件下能收敛到初始状态的平均值,与其他同样利用二跳邻接节点信息的一致性算法相比,具有通信量小、收敛速度更快的特点,但是能容忍的最大通信延时变小。

关键词:分布式一致性; 多智能体系统; 有向拓扑; 延时; 高阶

中图分类号: TP273.5 **文献标志码:** A

High-order distributed consensus algorithm under directed communication topology

PENG Huanxin^{1,2*}, QI Guoqing², SHENG Andong²

(1. School of Mechanical Engineering, Nanjing Institute of Industry Technology, Nanjing Jiangsu 210046, China;

2. School of Automation, Nanjing University of Science and Technology, Nanjing Jiangsu 210094, China)

Abstract: In order to improve the convergence rate of distributed consensus algorithms under directed communication topologies, a high-order distributed consensus algorithm was proposed. Under directed topologies, the previous state values of two-hop adjacency nodes were utilized to improve the convergence rate based on single-hop communication. The performance and convergence rate of the high-order distributed consensus algorithm were analyzed under directed networks. The simulation results were provided to verify these analytical results. The results show that an average consensus can be reached under certain conditions, the convergence rate of the high-order algorithm is superior to the other algorithms utilizing the information of two-hop adjacency nodes, but the high-order algorithm can tolerate smaller communication time-delays than the other algorithms utilizing the information of two-hop adjacency nodes.

Key words: distributed consensus; multi-Agent system; directed topology; time-delay; high-order

0 引言

近年来,多智能体系统的分布式一致性问题在许多领域应用广泛,已经引起许多研究人员的极大关注。

收敛速度是分布式一致性算法的重要研究内容,特别是对于大规模复杂系统。提高分布式一致性算法的收敛速度,关键是提高通信网络拓扑图的代数连通性^[1]。现有的分布式一致性算法^[1-8]主要通过单跳通信,利用邻接节点的状态进行自身节点状态的更新,收敛速度比较慢。为了提高分布式一致性算法的收敛速度,文献[9]提出通过多跳中继通信,利用非邻接节点的状态进行自身节点状态的更新。由于每个节点获得的节点信息增加,因此这种方法很大程度上提高了分布式一致性算法的收敛速度,但是,多跳中继通信方式复杂,在每次采样间隔中,要求节点间进行多次中继通信,增加了通信要求和通信成本,对中继节点的通信负担较重。文献[10]提出了伪二跳中继分布式一致性算法(下面简称伪二跳算法),该算法通过单跳通信,利用二跳邻接节点的信息来进行节点状态更新。与二跳中继分布式一致性算法^[9]相比,伪二跳算法由于采用单跳通信,通信方式简单,通信量大大减

少,收敛速度与二跳中继分布式一致性算法的收敛速度相当。为了进一步提高分布式一致性算法的收敛速度,文献[10]在伪二跳算法基础上,进一步提出伪多跳中继分布式一致性算法(下面简称伪多跳算法),该算法通过单跳通信,利用多跳邻接节点的信息来进行节点状态更新。伪多跳算法相比伪二跳算法,收敛速度大大提高,但通信量也大大增加。

上述利用非邻接节点信息的分布式一致性算法只考虑了无向通信拓扑下的收敛情况,但是在实际应用中,如电力网络、万维网、神经网络等都是复杂的有向拓扑网络。因此研究如何提高有向通信拓扑下的分布式一致性算法的收敛速度非常必要。文献[11]将伪二跳算法应用在有向通信拓扑上,分析了伪二跳算法在有向拓扑下的收敛性。文献[12]将伪多跳算法应用在有向通信拓扑上,分析了伪多跳算法在有向拓扑下的收敛性,但是伪多跳算法在收敛速度提高的同时,通信量也大大增加。

为了进一步提高有向通信拓扑下的分布式一致性算法的收敛速度,同时又不增加节点间的通信量,在伪二跳算法的基础上,本文提出了一种高阶分布式一致性算法。高阶分布式一致性算法通过有向单跳通信,利用有向二跳邻接节点的前

收稿日期:2013-03-25;修回日期:2013-06-01。

基金项目:国家自然科学基金资助项目(61104186);江苏省智能传感器网络工程技术研究开发中心开发基金资助项目(ZK12-01-12)。

作者简介:彭焕新(1973-),男,湖南涟源人,讲师,博士研究生,主要研究方向:数据融合、非线性滤波;戚国庆(1977-),男,安徽合肥人,副研究员,博士,主要研究方向:非线性滤波、数据融合;盛安冬(1964-),男,浙江杭州人,研究员,博士生导师,博士,主要研究方向:导航、制导与控制。

多步信息进行节点状态更新。本文对有向拓扑下的高阶分布式一致性算法的通信量、收敛性能、收敛速度以及带通信延时下的收敛性能进行了分析、计算和仿真比较。

1 有向拓扑下高阶分布式一致性算法

首先,介绍本文需要用到的一些相关概念。有向拓扑图 $G = (V, E)$ 代表带 n 个节点的多智能体系统的有向通信拓扑,节点集 V 内的元素代表智能体,边集 E 的元素代表智能体之间的通信连接。节点 $i, j \in V$, 用有序对 $(i, j) \in E$ 表示边,边的前后节点分别称之为边的头和尾,信号由节点 j 传送给节点 i 。对于有向拓扑 $G = (V, E)$, $(i, j) \neq (j, i)$ 。同节点 i 相连,且头为 i 的边数的和,称为节点 i 的出度,即 $deg_{out}(i) = \sum_{j=1}^N a_{ij}$; 和节点 i 相连,且尾为 i 的边数的和,称为节点 i 的入度,即 $deg_{in}(i) = \sum_{j=1}^N a_{ji}$ 。如果每个节点的出度等于其入度,则称该图是平衡图。有向路径是指有序节点集 $(1, 2, \dots, m)$, 其任意相邻的两个有序节点所构成的边 $(i, j) \in E$ 。如果图 G 的任意两个有序节点间都存在一条有向路径相连,两节点分别为路径的头与尾,则称有向图 G 是强连通图。对于节点 i , 所有满足 $(i, j) \in E$ 的节点的集合称为 i 的邻接节点集,用 N_i 表示节点 i 的邻接节点集,邻接节点集的元素为节点 i 的邻接节点 $\mathbf{A} = \{a_{ij}\}$, 其中 $a_{ij} = \begin{cases} 1, & (i, j) \in E \\ 0, & (i, j) \notin E \end{cases}$, \mathbf{L} 分别为有向拓扑 $G = (V, E)$ 对应的邻接矩阵和拉普拉斯矩阵,设 $\hat{G} = (V, \hat{E})$ 为二跳通信拓扑^[9], $\tilde{\mathbf{A}} = \{\beta_{il}\}_{n \times n}$, $\tilde{\mathbf{L}}$ 分别为二跳通信拓扑的邻接矩阵和拉普拉斯矩阵:

$$\mathbf{L} = \mathbf{A} - \mathbf{A}; \mathbf{\Delta} = \text{diag}\{\sum_j a_{ij}\}$$

$$\tilde{\mathbf{A}} = \{\beta_{il}\} = \begin{cases} \sum_j a_{ij} a_{jl}, & i \neq l \\ 0, & i = l \end{cases}$$

$$\tilde{\mathbf{L}} = \tilde{\mathbf{A}} - \tilde{\mathbf{A}}; \tilde{\mathbf{\Delta}} = \text{diag}\{\sum_l \beta_{il}\}$$

本文针对有向通信拓扑,提出一种高阶分布式一致性算法,节点间通过单跳通信进行信息交换,对于有向通信拓扑 $G = (V, E)$, 每个节点将其状态值和状态派生值传送给有向相邻的节点。有向通信拓扑下的高阶分布式一致性算法可以写成:

$$\begin{aligned} x_i(k+1) &= x_i(k) - \varepsilon \sum_{j \in N_i} a_{ij} \{ (x_i(k) - x_j(k)) + \\ &\gamma \sum_{l \in N_j} a_{jl} (x_i(k-1) - x_j(k)) + \\ &\gamma \sum_{l \in N_j} a_{jl} (x_i(k-2) - x_j(k-1)) + \dots + \\ &\gamma \sum_{l \in N_j} a_{jl} (x_i(k-m+1) - x_j(k-m+2)) \} \end{aligned} \quad (1)$$

$$y_i(k+1) = \sum_{j \in N_i} a_{ij} x_j(k) \quad (2)$$

其中: $x_i(k)$ 是节点 i 在 k 时刻的状态值, $y_i(k)$ 是节点 i 在 k 时刻的状态派生值, ε 是采样间隔, γ 为实参数。在高阶分布式一致性算法中,节点将 k 时刻的自身状态值和状态派生值传送给有向相邻的节点,而有向相邻节点 j 在 $k-1, k-2$ 等时刻的状态值和状态派生值已经分别在 $k-1, k-2$ 等时刻传送给有向相邻的节点 i , 并保存在有向相邻的节点 i 中,因此,在 k 时刻,节点 j 需要传送的信息只有 $x_j(k), y_j(k)$ 两个值。高

阶分布式一致性算法与文献[11]中的有向拓扑下的伪二跳算法的通信量相同,计算量相同;远小于有向拓扑下的伪多跳算法^[12]的通信量,计算量也比伪多跳算法的计算量小。

2 动态分析

2.1 收敛性分析

将式(2)代入式(1),可以得到:

$$\begin{aligned} x_i(k+1) &= x_i(k) - \varepsilon \sum_{j \in N_i} a_{ij}(k) \{ (x_i(k) - x_j(k)) + \\ &\gamma \sum_{l \in N_j} a_{jl} (x_i(k-1) - x_j(k-1)) + \dots + \\ &\gamma \sum_{l \in N_j} a_{jl} (x_i(k-m+1) - x_j(k-m+1)) \} \end{aligned} \quad (3)$$

将式(3)写成向量形式:

$$\mathbf{x}(k+1) = (\mathbf{I}_n - \varepsilon \mathbf{L}) \mathbf{x}(k) - \varepsilon \gamma \tilde{\mathbf{L}} \mathbf{x}(k-1) - \dots - \varepsilon \gamma \tilde{\mathbf{L}} \mathbf{x}(k-m+1) \quad (4)$$

其中 $\mathbf{x}(k) = [x_1(k), x_2(k), \dots, x_n(k)]^T$ 。

定义

$$\begin{aligned} \mathbf{X}(k+1) &= [\mathbf{x}(k+1)^T, \mathbf{x}(k)^T, \dots, \mathbf{x}(k-m+2)^T]^T \\ \mathbf{H} &= \begin{bmatrix} \mathbf{I} - \varepsilon \mathbf{L} & \dots & -\varepsilon \gamma \tilde{\mathbf{L}} & -\varepsilon \gamma \tilde{\mathbf{L}} \\ \mathbf{I} & \dots & 0 & 0 \\ \vdots & & \vdots & \vdots \\ 0 & \dots & \mathbf{I} & 0 \end{bmatrix}, \mathbf{J} = \begin{bmatrix} K & 0 & \dots & 0 \\ K & 0 & \dots & 0 \\ \vdots & \vdots & & \vdots \\ K & 0 & \dots & 0 \end{bmatrix} \end{aligned}$$

其中 $K = \frac{1}{n} \mathbf{1} \mathbf{1}^T$, $\mathbf{1} = [1, 1, \dots, 1]^T$ 是 n 维向量。式(4)可以写成:

$$\mathbf{X}(k+1) = \mathbf{H} \mathbf{X}(k) \quad (5)$$

设初始条件为: $\mathbf{X}(0) = [\mathbf{x}(0)^T, \mathbf{x}(-1)^T, \mathbf{x}(-2)^T, \dots, \mathbf{x}(-m+1)^T]^T$, $\mathbf{x}(0)^T = \mathbf{x}(-1)^T = \mathbf{x}(-2)^T = \dots = \mathbf{x}(-m+1)^T$ 。

有下列引理:

引理 1^[13] 除了矩阵 \mathbf{H} 的特征值 1, 即 $\lambda(\mathbf{H}) = 1$ 被矩阵 $\mathbf{H} - \mathbf{J}$ 的 0 特征值, 即 $\lambda(\mathbf{H} - \mathbf{J}) = 0$ 取代之外, 矩阵 $\mathbf{H} - \mathbf{J}$ 与矩阵 \mathbf{H} 有相同的特征值。

证明 引理的证明参考文献[13], 在这里不再详细证明。

下面分析有向通信拓扑下, 高阶分布式一致性算法的收敛性, 有下列结论。

定理 1 当有向通信拓扑 G 为强连通拓扑, 且矩阵 $\mathbf{H} - \mathbf{J}$ 的谱半径 $\rho(\mathbf{H} - \mathbf{J}) < 1$, 则高阶分布式一致性算法能收敛到一致性, 即: $\lim_{k \rightarrow \infty} \mathbf{x}(k) = \frac{1}{n} \mathbf{1} c$, 其中 c 为一常数。

证明 如果矩阵 $\mathbf{H} - \mathbf{J}$ 的谱半径 $\rho(\mathbf{H} - \mathbf{J}) < 1$, 则根据引理 1 可知, 矩阵 \mathbf{H} 有唯一值为 1 的特征值, 其他特征值的模小于 1。定义向量 $\mathbf{w}_l, \mathbf{w}_r$ 分别为矩阵 \mathbf{H} 的特征值 1 所对应的左、右特征向量, 由于 G 为有向强连通拓扑, 所以矩阵 $\mathbf{L}, \tilde{\mathbf{L}}$ 为行和等于 0 的矩阵, 列和不一定等于 0, 从而向量 \mathbf{w}_r 满足 $\mathbf{H} \mathbf{w}_r = \mathbf{w}_r$, 其中 $\mathbf{w}_r = \frac{1}{n} [1^T, 1^T, \dots, 1^T]^T, \mathbf{1} = [1, 1, \dots, 1]^T$ 是 n 维向量。从式(5)可以得到:

$$\lim_{k \rightarrow \infty} \mathbf{X}(k+1) = \lim_{k \rightarrow \infty} \mathbf{H}^{k+1} \mathbf{X}(0) = \mathbf{w}_r \mathbf{w}_l^T \mathbf{X}(0)$$

即: $\lim_{k \rightarrow \infty} \mathbf{x}(k) = \frac{1}{n} \mathbf{1} c$, 其中 $c = \mathbf{w}_l^T \mathbf{X}(0)$ 。证毕。

定理 2 当有向通信拓扑 G 为强连通平衡拓扑, 且二跳拓扑 $\hat{G} = (V, \hat{E})$ 为平衡拓扑, 如果矩阵 $\mathbf{H} - \mathbf{J}$ 的谱半径 $\rho(\mathbf{H} -$

$J) < 1$, 则高阶分布式一致性算法能收敛到初始状态的平均值, 即: $\lim_{k \rightarrow \infty} \mathbf{x}(k) = \frac{1}{n} \mathbf{1} \mathbf{1}^T \mathbf{x}(0)$ 。

证明 如果矩阵 $\mathbf{H} - \mathbf{J}$ 的谱半径 $\rho(\mathbf{H} - \mathbf{J}) < 1$, 则根据引理 1 可知, 矩阵 \mathbf{H} 有唯一值为 1 的特征值, 其他特征值的模小于 1。定义向量 $\mathbf{w}_1, \mathbf{w}_r$ 分别为矩阵 \mathbf{H} 的特征值 1 所对应的左、右特征向量。由于有向通信拓扑 G 为强连通平衡拓扑, 且二跳拓扑为平衡拓扑, 所以矩阵 $\mathbf{L}, \bar{\mathbf{L}}$ 为行和与列和都等于 0 的矩阵, 从而满足 $\mathbf{w}_1^T \mathbf{H} = \mathbf{w}_1^T, \mathbf{H} \mathbf{w}_r = \mathbf{w}_r, \mathbf{w}_r^T \mathbf{w}_1 = 1$, 其中 $\mathbf{w}_1 = [1^T, \mathbf{0}^T, \mathbf{0}^T, \dots]^T, \mathbf{w}_r = \frac{1}{n} [1^T, 1^T, \dots, 1^T]^T, \mathbf{1} = [1, 1, \dots, 1]^T$ 与 $\mathbf{0} = [0, \dots, 0]^T$ 是 n 维向量。从式(5)可以得到:

$$\lim_{k \rightarrow \infty} \mathbf{X}(k+1) = \lim_{k \rightarrow \infty} \mathbf{H}^{k+1} \mathbf{X}(0) = \mathbf{w}_r \mathbf{w}_1^T \mathbf{X}(0)$$

即: $\lim_{k \rightarrow \infty} \mathbf{x}(k) = \frac{1}{n} \mathbf{1} \mathbf{1}^T \mathbf{x}(0)$ 。 证毕。

从上述定理可以看出, 在有向通信拓扑下, 如果通信拓扑为强连通拓扑, 不管是不是平衡拓扑, 只要满足 $\rho(\mathbf{H} - \mathbf{J}) < 1$, 高阶分布式一致性算法都能收敛, 但是不一定能收敛到初始状态的平均值。要使在有向通信拓扑下, 高阶分布式一致性算法收敛到初始状态的平均值, 有向通信拓扑必须是强连通平衡拓扑, 且其二跳拓扑为平衡拓扑。

2.2 收敛速度分析

下面对有向通信拓扑下的高阶分布式一致性算法的收敛速度进行分析。根据文献[1]可知, 分布式一致性算法的收敛速度取决于通信拓扑图的代数连通性, 因此当高阶分布式一致性算法收敛, 其收敛速度由矩阵 \mathbf{H} 的第二最大特征值, 或矩阵 $\mathbf{H} - \mathbf{J}$ 的谱半径 $\rho(\mathbf{H} - \mathbf{J}) < 1$ 决定。矩阵 \mathbf{H} 的第二最大特征值的计算比较困难, 在本文, 对有向拓扑下的高阶分布式一致性算法的收敛速度, 采用一种比较简单的方法粗略地与有向拓扑下的伪二跳算法^[11]的收敛速度进行比较。高阶分布式一致性算法可以写成:

$$\begin{aligned} \mathbf{x}(k+1) = & (\mathbf{I}_n - \varepsilon \gamma \mathbf{L} - \varepsilon \gamma \bar{\mathbf{L}} - \dots - \varepsilon \gamma \bar{\mathbf{L}}) \mathbf{x}(k) + \\ & \varepsilon \gamma \bar{\mathbf{L}} [\mathbf{x}(k) - \mathbf{x}(k-1)] + \dots + \\ & \varepsilon \gamma \bar{\mathbf{L}} [\mathbf{x}(k) - \mathbf{x}(k-m+1)] \end{aligned} \quad (6)$$

当高阶一致性算法收敛, 在有限步后, $\|\mathbf{x}(k) - \mathbf{x}(k-1)\|_2 \rightarrow 0, \|\mathbf{x}(k) - \mathbf{x}(k-2)\|_2 \rightarrow 0, \dots, \|\mathbf{x}(k) - \mathbf{x}(k-m+1)\|_2 \rightarrow 0$, 其中 $\|\cdot\|_2$ 指向量的 2 范数。从而式(6)可以近似地得到:

$$\begin{aligned} \mathbf{x}(k+1) \approx & (\mathbf{I} - \varepsilon \mathbf{L} - \varepsilon \gamma \bar{\mathbf{L}} - \dots - \varepsilon \gamma \bar{\mathbf{L}}) \mathbf{x}(k) = \\ & (\mathbf{I} - \varepsilon \mathbf{L} - (m-1) \varepsilon \gamma \bar{\mathbf{L}}) \mathbf{x}(k) \end{aligned} \quad (7)$$

将式(7)与文献[11]的有向拓扑下的伪二跳算法比较可以发现, 高阶分布式一致性算法的收敛速度可以近似地由矩阵 $\mathbf{I} - \varepsilon \mathbf{L} - (m-1) \varepsilon \gamma \bar{\mathbf{L}}$ 的第二最大特征值决定。在高阶分布式一致性算法收敛的前提下, 当 $(m-1) > 1$, 有向拓扑下的高阶分布式一致性算法的收敛速度要高于伪二跳算法的收敛速度。

2.3 带通信延时的收敛性分析

当智能体之间存在通信延时, 分布式一致性算法的收敛性通常会变差。下面分析有向拓扑下, 高阶分布式一致性算法在带延时的通信拓扑下的收敛性能。设每条通信连接的延时相同, 为 τ , 则带通信延时的高阶分布式一致性算法可以写成:

$$\mathbf{x}_i(k+1) = \mathbf{x}_i(k) - \varepsilon \sum_{j \in N_i} a_{ij} \{ (\mathbf{x}_i(k-\tau) - \mathbf{x}_j(k-\tau)) +$$

$$\begin{aligned} & \gamma \sum_{i \in N_j} a_{ji} (\mathbf{x}_i(k-1-2\tau) - \mathbf{x}_j(k-\tau)) + \\ & \gamma \sum_{i \in N_j} a_{ji} (\mathbf{x}_i(k-2-2\tau) - \mathbf{x}_j(k-1-\tau)) + \dots + \\ & \gamma \sum_{i \in N_j} a_{ji} (\mathbf{x}_i(k-m+1) - \mathbf{x}_j(k-m+2)) \} \end{aligned} \quad (8)$$

$$y_i(k+1) = \sum_{j \in N_i} a_{ij} x_j(k-\tau) \quad (9)$$

将式(9)代入式(8), 写成向量形式:

$$\begin{aligned} \mathbf{x}(k+1) = & \mathbf{x}(k) - \varepsilon \mathbf{L} \mathbf{x}(k-\tau) - \varepsilon \gamma \bar{\mathbf{L}} \mathbf{x}(k-1-2\tau) - \\ & \dots - \varepsilon \gamma \bar{\mathbf{L}} \mathbf{x}(k-m+1-2\tau) \end{aligned} \quad (10)$$

对于带通信延时的分布式一致性算法的收敛性分析, 可以采用时域中的 Lyapunov 方法进行分析, 对于离散的带延时系统, 要求其特征值都位于单位圆内, 系统才能够一致性收敛。从式(10)可以得到:

$$\begin{aligned} \mathbf{z} \mathbf{x}(k) = & \mathbf{x}(k) - \varepsilon \mathbf{L} \mathbf{z}^{-\tau} \mathbf{x}(k) - \varepsilon \gamma \bar{\mathbf{L}} \mathbf{z}^{-(2\tau+1)} \mathbf{x}(k) - \\ & \dots - \varepsilon \gamma \bar{\mathbf{L}} \mathbf{z}^{-(2\tau+m-1)} \mathbf{x}(k) \end{aligned} \quad (11)$$

从式(11)可以推出:

$$(z - 1 + \varepsilon \mathbf{L} \mathbf{z}^{-\tau} + \varepsilon \gamma \bar{\mathbf{L}} \mathbf{z}^{-(2\tau+1)} + \dots + \varepsilon \gamma \bar{\mathbf{L}} \mathbf{z}^{-(2\tau+m-1)}) \mathbf{x}(k) = 0 \quad (12)$$

其特征方程等于:

$$\det \left(\mathbf{I} + \frac{\varepsilon \mathbf{L} \mathbf{z}^{-\tau} + \varepsilon \gamma \bar{\mathbf{L}} \mathbf{z}^{-(2\tau+1)} + \dots + \varepsilon \gamma \bar{\mathbf{L}} \mathbf{z}^{-(2\tau+m-1)}}{z - 1} \right) = 0 \quad (13)$$

显然, 如果式(13)所示特征方程的根除了一个根等于 1, 其余根都位于单位圆内, 则当有向拓扑 G 为强连通平衡拓扑, 且二跳拓扑为平衡拓扑时, 带通信延时的高阶分布式一致性算法能收敛到初始状态的平均值。

从式(13)可以看出, 要计算出高阶分布式一致性算法收敛所能容忍的最大通信延时比较困难。为了简单起见, 本文不具体计算有向拓扑下的高阶分布式一致性算法保持收敛所能容忍的最大通信延时, 而是与有向拓扑下的伪二跳算法所能容忍的最大通信延时进行比较。

当式(10)收敛, 在有限步后, $\|\mathbf{x}(k-\tau) - \mathbf{x}(k-1-\tau)\|_2 \rightarrow 0, \|\mathbf{x}(k-2\tau) - \mathbf{x}(k-2-2\tau)\|_2 \rightarrow 0, \dots, \|\mathbf{x}(k-2\tau) - \mathbf{x}(k-m+1-2\tau)\|_2 \rightarrow 0$, 其中 $\|\cdot\|_2$ 指向量的 2 范数。从式(10)可以近似地得到:

$$\begin{aligned} \mathbf{x}(k+1) = & \mathbf{x}(k) - \varepsilon \mathbf{L} \mathbf{x}(k-\tau) - \varepsilon \gamma \bar{\mathbf{L}} \mathbf{x}(k-2\tau) - \\ & \dots - \varepsilon \gamma \bar{\mathbf{L}} \mathbf{x}(k-2\tau) = \mathbf{x}(k) - \varepsilon \mathbf{L} \mathbf{x}(k-\tau) - \\ & \varepsilon (m-1) \gamma \bar{\mathbf{L}} \mathbf{x}(k-2\tau) \end{aligned} \quad (14)$$

从式(14)可以看出, 当参数 $\gamma = 0$, 这时高阶分布式一致性算法就等于文献[1]所示的一阶分布式一致性算法, 当 $\gamma = 1/(m-1)$, 这时高阶分布式一致性算法就等于文献[9]中二跳算法, 因此当 $0 \leq \gamma \leq 1/(m-1)$, 高阶分布式一致性算法收敛所能容忍的最大通信延时介于一阶算法和伪二跳算法所能容忍的最大通信延时之间。当 $\gamma > 1/(m-1)$, 则高阶分布式一致性算法能容忍的最大通信延时小于伪二跳算法所能容忍的最大通信延时^[11]。

从上述分析可以看出, 在有向通信拓扑下, 参数 γ 反映了利用有向二跳邻接节点的程度, 当 $\gamma = 0$, 完全没有利用有向二跳邻接节点的信息, 这时高阶一致性算法就是一阶一致性算法。 γ 值越大, 利用有向二跳邻接节点信息越多, 因此高阶一致性算法收敛速度越快, 但是当通信拓扑存在延时, 所能容忍的最大延时也越小。因此当智能体存在通信延时, 应该在收敛速度与能容忍的最大延时之间进行平衡, 选择适当的参

数 γ 。

3 仿真分析

为了验证高阶分布式一致性算法在有向通信拓扑下的收敛一致性以及收敛速度,采用图 1、图 2 所示的有向拓扑图 G_1 和有向拓扑图 G_2 进行仿真, G_1 为强连通平衡拓扑,且其二跳拓扑也是平衡拓扑, G_2 为强连通非平衡拓扑。如未特别说明,选择采样间隔 $\varepsilon = 0.02 \text{ s}$, $\gamma = 1$ 。节点初始值为 0 到 50 之间的随机值,通过 500 次蒙特卡洛仿真得到下列仿真结果。

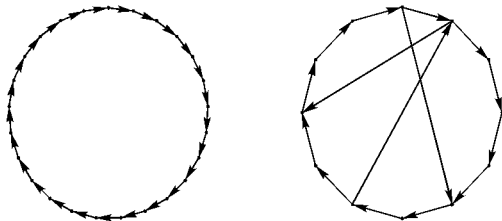


图 1 有向强连通平衡拓扑 图 2 有向强连通非平衡拓扑

图 3 是三阶分布式一致性算法在有向通信拓扑 G_1 下,不

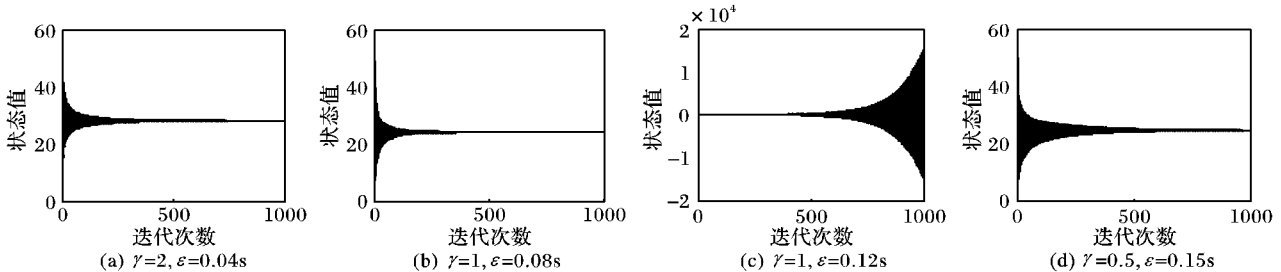


图 3 拓扑 G_1 下的三阶算法的收敛性

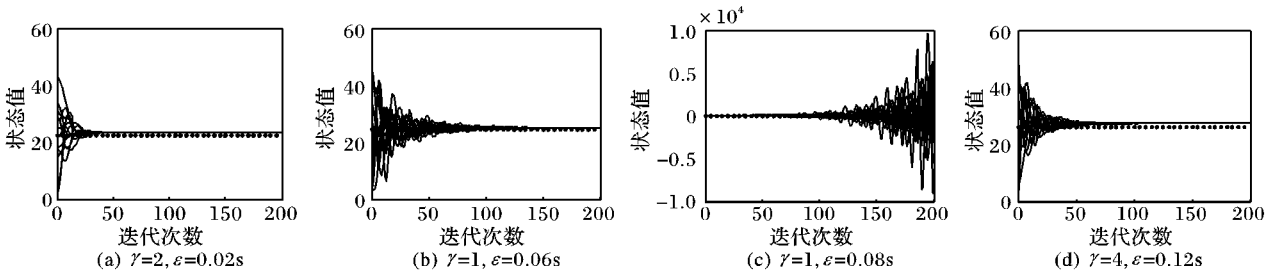


图 4 拓扑 G_2 下四阶算法的收敛性

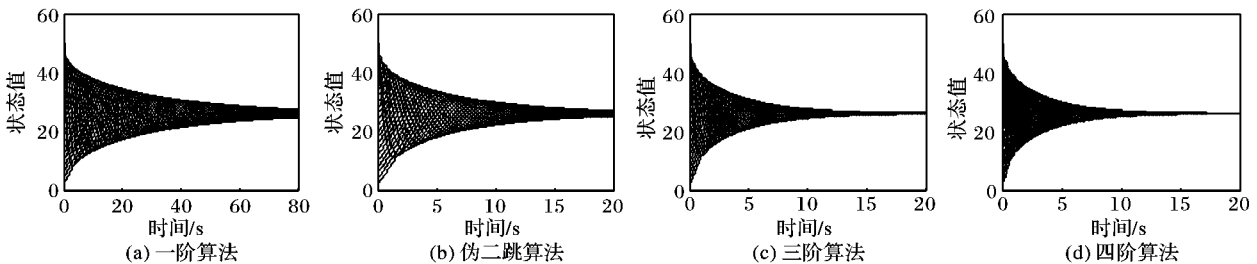


图 5 拓扑 G_1 下四种算法的收敛速度比较

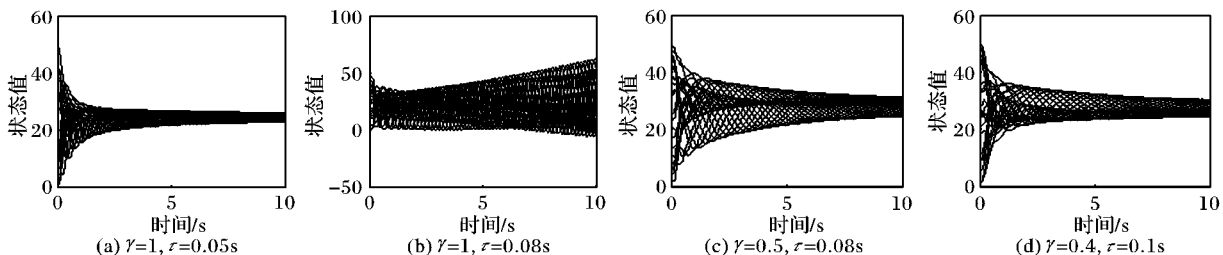


图 6 拓扑 G_1 下带通信延时的四阶算法的收敛性

从图 6 可以看出,当通信延时在一定范围内时,高阶分布式一致性算法能够收敛;当 γ 值越小,高阶分布式一致性算法

同的 ε, γ 下的收敛性,图 4 是四阶分布式一致性算法在有向通信拓扑 G_2 下的收敛性,其中虚线为初始状态的平均值。从图 3、图 4 可以看出,当参数 ε, γ 满足使矩阵 $\mathbf{H} - \mathbf{J}$ 的谱半径小于 1,则高阶分布式一致性算法收敛,反之则发散。例如图 3 (c)和图 4 (c),由于参数 ε, γ 不满足使 $\rho(\mathbf{H} - \mathbf{J}) < 1$,因此不收敛;如果有向拓扑强连通,则高阶分布式一致性算法能收敛,但是不一定能收敛到初始状态的平均值,而当有向拓扑是强连通平衡拓扑,且二跳拓扑也是平衡拓扑,则高阶分布式一致性算法能收敛到初始状态的平均值;通常 γ 较大时,允许的采样周期较小,而 γ 值较小时,允许采样周期较大。

图 5 是三阶、四阶分布式一致性算法(其中 $\gamma = 1$)与一阶算法^[1]、伪二跳算法^[11]在通信拓扑 G_1 下、相同初始条件下的收敛速度比较,其中节点初始状态为 2 到 50 的等差数列。从图 5 可以看出三阶、四阶分布式一致性算法的收敛速度远快于一阶算法、伪二跳算法的收敛速度,其中四阶分布式一致性算法的收敛速度最快。图 6 是在带通信延时时,四阶分布式一致性算法在通信拓扑 G_1 下的收敛性。

能容忍的通信延时也越大;通常通信延时越大时,高阶分布式一致性算法的收敛性变差。

4 结语

本文提出了在有向通信拓扑下的高阶分布式一致性算法,该算法通过利用二跳邻接节点的前多步信息来提高一致性算法收敛速度。当有向通信拓扑是强连通拓扑,高阶分布式一致性算法能收敛到一致性,当有向通信拓扑是强连通平衡拓扑,且二跳拓扑为平衡拓扑,则高阶分布式一致性算法能收敛到初始状态的平均值。由于利用了二跳邻接节点的前多步信息,因此高阶分布式一致性算法在不增加通信量的前提下,其收敛速度大大高于相同条件下的伪二跳算法的收敛速度。当存在通信延时,高阶分布式一致性算法存在延时累计的现象,可以通过选择较小的 γ 参数来提高其在带通信延时情况下的稳定性。

参考文献:

- [1] OLFATI-SABER R, MURRAY R M. Consensus problems in networks of agents with switching topology and time-delays[J]. IEEE Transactions on Automatic Control, 2004, 49(9): 1520-1533.
- [2] MATHUR R, SHARMA M K, MISRA A, *et al.* Energy-efficient deployment of distributed mobile sensor networks using fuzzy logic systems[C]// Proceedings of the 2009 International Conference on Advances in Computing, Control, and Telecommunication Technologies. Washington, DC: IEEE Computer Society, 2009: 121-125.
- [3] AYSAL T C, ORESHKIN B N, COATES M J. Accelerated distributed average consensus via localized node state prediction[J]. IEEE Transactions on Signal Processing, 2009, 57(4): 1563-1576.
- [4] AYSAL T C, YILDIZ M E, SARWATE A D, *et al.* Broadcast gossip algorithms for consensus[J]. IEEE Transactions on Signal Pro-

cessing, 2009, 57(7): 2748-2761.

- [5] ZHANG Y, TIAN Y P. Consensus of data-sampled multi-Agent systems with random communication delay and packet loss[J]. IEEE Transactions on Automatic Control, 2010, 55(4): 939-943.
- [6] FRANCESCHELLI M, GIUA A, SEATZU C. Distributed averaging in sensor networks based on broadcast gossip algorithms[J]. IEEE Sensors Journal, 2011, 11(3): 808-817.
- [7] RAJAGOPAL R, WAINWRIGHT M J. Network-based consensus averaging with general noisy channels[J]. IEEE Transactions on Signal Processing, 2011, 59(1): 373-385.
- [8] THANOU D, KOKIOPOULOU E, PU Y, *et al.* Distributed average consensus with quantization refinement[J]. IEEE Transactions on Signal Processing, 2013, 61(1): 194-205.
- [9] JIN Z P, MURRAY R M. Multi-hop relay protocols for fast consensus seeking[C]// Proceedings of the 45th Conference on Decision and Control. Piscataway: IEEE, 2006: 1001-1006.
- [10] 彭焕新, 戚国庆, 盛安冬. 伪多跳中继分布式一致性算法[J]. 控制理论与应用, 2012, 29(5): 623-628.
- [11] PENG H X, WANG W K. Pseudo two-hop distributed consensus algorithm under directed topologies[EB/OL]. [2013-01-20]. www.scientific.net/AMR.591-593.1599.pdf.
- [12] PENG H X, SHENG A D, WANG W K. Pseudo multi-hop distributed consensus algorithm under directed topologies[C]// Proceedings of the 12th International Conference on Control, Automation, Robotics and Vision. Piscataway: IEEE, 2012: 1616-1619.
- [13] MEYER C D. Matrix analysis and applied linear algebra[M]. Philadelphia, PA: Society for Industrial and Applied Mathematics, 2000.

(上接第 2756 页)

- [2] FU F, CHEN X, LIU L, *et al.* Social dilemmas in an online social network: the structure and evolution of cooperation[J]. Physics Letters A, 2007, 371(1/2): 58-64.
- [3] KUMAR R, NOVAK J, TOMKINS A. Structure and evolution of online social networks[C]// Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2006: 611-617.
- [4] YU L, KING V. The evolution of friendships in Chinese online social networks[C]// Proceedings of the 2010 IEEE International Conference on Privacy, Security, Risk and Trust. Piscataway: IEEE, 2010: 81-87.
- [5] AHN Y Y, HAN S. Analysis of topological characteristics of huge online social networking services[C]// Proceedings of the 16th International Conference on World Wide Web. New York: ACM, 2007: 835-844.
- [6] 陶少华, 赵会洋, 平源, 等. 基于吸引因子的无尺度网络演化模型研究[J]. 复杂网络与复杂性科学, 2008, 5(2): 88-92.
- [7] KWAK H, LEE C H, PARK P, *et al.* What is Twitter, a social network or a news media? [C]// Proceedings of the 19th International Conference on World Wide Web. New York: ACM, 2010: 1772690-1772751.
- [8] WU S M, HOFMAN J M, MASON W A, *et al.* Who says what to whom on Twitter [C]// Proceedings of the 20th International Conference on World Wide Web. New York: ACM, 2011: 705-714.
- [9] SONG H H, CHO T W, DAVE V, *et al.* Scalable proximity estimation and link prediction in online social networks[C]// Proceedings of the 9th ACM SIGCOMM Conference on Internet

Measurement Conference. New York: ACM, 2009: 322-335.

- [10] YU L, ASUR D, HUBERMAN B A. What trends in Chinese social media[C]// Proceedings of the 5th ACM Workshop on Social Network Mining and Analysis. New York: ACM, 2011: 2-4.
- [11] 尹书华. 基于复杂网络的微博用户关系网络特征研究[J]. 西南师范大学学报, 2011, 36(6): 58-61.
- [12] 樊鹏翼, 王晖, 姜志宏, 等. 微博网络测量研究[J]. 计算机研究与发展, 2012, 49(4): 691-699.
- [13] YIN D W, HONG L J, XIONG X. Link formation analysis in microblogs[C]// Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval. New York: ACM, 2011: 1235-1236.
- [14] MITSUHIRO H D, KAMURA N. Cognitive-costed Agent model of the microblogging network [C]// Proceedings of the 6th International Workshop on Agent based Approaches in Economic and Social Complex Systems. Berlin: Springer, 2009: 93-98.
- [15] ROMERO D, KLEINBERG J. The directed closure process in hybrid social-information networks, with an analysis of link formation on Twitter[EB/OL]. [2013-01-20]. http://www.cs.cornell.edu/home/kleinber/icwsm10-closure.pdf.
- [16] YIN D, HONG L, DAVISON B D. Structural link analysis and prediction in microblogs [C]// Proceedings of the 20th ACM International Conference on Information and Knowledge Management. New York: ACM, 2011: 1163-1168.
- [17] 中国互联网数据中心. 2010 微博与社区调查数据[R]. 中国互联网数据中心, 2011.
- [18] 汪小帆, 李翔, 陈关荣. 复杂网络理论及其应用[M]. 北京: 清华大学出版社, 2006.