

应用层组播路由的一种改进神经网络解法^①刘世栋^② 张顺颐 周井泉

(南京邮电大学信息技术研究所 南京 210003)

摘 要 研究了使用双层递归神经网络(DRNN)模型求解应用层组播路由的问题。对原有模型的神经元矩阵及能量函数进行改变,并引入了新的线性编程神经元,解决了原模型不能求解组播路由的缺陷。与启发式组播路由算法相比,该解决方案的计算复杂度低,速度较快,而且与其它的神经网络相比,由于引入了基尔霍夫限制条件,保障了解的质量,且所使用的神经元数量少,而在解的精确度上则与其它算法相当。

关键词 神经网络,神经元,组播,覆盖网络,限制条件,独立变量

0 引 言

基于覆盖网络实现的应用层组播^[1-6]已成为在因特网上部署大规模流媒体应用的可行途径之一。最优应用层组播路由的发现可以归纳成为一个受限的 steiner 树求解问题。目前已有的解决方案主要有启发式路由算法^[7-9]和神经网络^[10-13]求解算法两种。本文重点研究基于神经网络模型求解受限应用层组播路由问题。Ali-Kamoun 最早将最优路由求解问题的计算用神经网络来实现,从而解决了实时性^[10],该方法需要的神经元数等于网络中节点数的平方。Flip^[12]等人提出了一种双层递归神经网络(dual-layer recurrent neural network, DRNN)模型并用其求解最短路由,该模型所用的每个神经元对应网络的一条边(或称支路、链路),因而神经元数等于边数,通常少于节点的平方,从而简化了神经网络的设计。Xia 近期在文献[11]中提出使用自组织的分布 Kohonen 神经网络来求解覆盖网络的组播路由,并且保证节点接口带宽的负荷均衡。本文在文献[12]的基础上对原有神经网络模型进行改进,从而得到一种受限的 QoS 组播路由的神经网络求解方案。DRNN 在保证节点连接度限制条件得到满足的前提下,使得求解的组播路由的费用最优。其特点是:(1)利用了 DRNN 模型引入的基尔霍夫(Kirchoff)限制,保证了解的有效性;(2)建立了对应于每个组播组成员的神经元矩阵,并通过矩阵之间的互联保证了组播路由的最优;(3)在每个神经元矩阵中引入了

线性编程(linear programmable, LP)型神经元,从而保证了节点连接度限制条件的满足。

1 受限最优路由的数学描述及独立决策变量

1.1 受限最优路径的约束关系

对于一个给定的有向图 $G = (V, LK)$, 设其节点数为 n , 边数为 m , V 和 LK 分别为这些节点和边的集合。任选一个节点作为参考节点, 则余下的节点与支路间的关联可以用一个 $(n-1) \times m$ 阶的关联矩阵 A 表示:

$$A = [a_{ij}] \quad (i = 1, 2, \dots, n-1; j = 1, 2, \dots, m) \quad (1)$$

其中

$$a_{ij} = \begin{cases} 1 & \text{如果支路 } j \text{ 与节点 } i \text{ 关联, 且} \\ & \text{它的方向背离该节点} \\ -1 & \text{如果支路 } j \text{ 与节点 } i \text{ 关联, 且} \\ & \text{它的方向指向该节点} \\ 0 & \text{如果支路 } j \text{ 与节点 } i \text{ 不关联} \end{cases} \quad (2)$$

对于源点为 s , 终点为 d 的一条路径 p_{sd} , 可用与有向图 G 中 m 条边相对应的一组决策变量的取值组成的向量 $\mathbf{v} = [v_1, v_2, \dots, v_m]^T$ 来表示, 其中各决策变量 $v_i (i = 1, 2, \dots, m)$ 的取值为 0 或 1, 当 $v_i = 1$ 时表示第 i 条支路被选中在路径 p_{sd} 中, 否则 $v_i = 0$ 。

若将路径 p_{sd} 上的各支路决策变量的取值 1 看作是该支路上有单位电流通过, 取值 0 看作是零电流, 则根据 Kirchhoff 电流定律可知, 组成一条有效路

① 863 计划(2005AA121620)资助项目。

② 男, 1971 年生, 博士, 工程师; 研究方向: 通信与信息系统; 联系人, E-mail: liusd-sheldon@126.com
(收稿日期: 2007-12-21)

径的各支路决策变量应满足式

$$\mathbf{A}\mathbf{v} = \mathbf{\Phi} \quad (3)$$

的约束关系。由于该约束条件来自于 Kirchhoff 电流定律,因此我们也把(3)式称为 Kirchhoff 约束条件,它的存在保证了解的有效性。其中 $\mathbf{\Phi}$ 代表一个 $n-1$ 阶的向量,形式如下:

$$\varphi_i = \begin{cases} 1 & i = s \\ -1 & i = d \\ 0 & \text{其它} \end{cases} \quad (4)$$

1.2 独立变量

将计算路径的各决策变量看作是支路电流变量,由网络理论可知,连支电流变量是独立的,其数量为 $m-n+1$ 个。而树支电流变量不独立,可由连支电流变量来表示^[14]。

选取独立变量时,各支路的编号作如下处理:

(1) 在有向图中任取一棵树,并将树支的编号取为 $1 \sim n-1$, 其对应的树支变量 $\mathbf{v}_t = [v_1, v_2, \dots, v_{n-1}]^T$ 是非独立的。

(2) 余下的支路为连支,编号为 $n \sim m$, 相对应的连支变量 $\mathbf{v}_l = [v_n, v_{n+1}, \dots, v_m]^T$ 为独立变量。

(3) 决策变量按先树支后连支的次序编号排列,可以写作: $\mathbf{v} = \begin{bmatrix} \mathbf{v}_t \\ \mathbf{v}_l \end{bmatrix}$ 。对应的关联矩阵记作

$\mathbf{A} = [\mathbf{A}_t | \mathbf{A}_l]$ 。由式(3)可得:

$$\mathbf{v}_l = \mathbf{A}_l^{-1} \mathbf{\Phi} - \mathbf{A}_l^{-1} \mathbf{A}_t \mathbf{v}_t \quad (5)$$

2 基于改进的 DRNN 模型求解受限的组播路由问题

2.1 改进后的神经网络模型

改进后神经网络模型的框架如图 1 所示,它由 $n-1$ 个神经元矩阵组成,每个矩阵中的神经元的构成同文献[12]。第 q 个矩阵用于寻找到目的地 q 的受限单播路由。图中虽然显示了 $n-1$ 个矩阵,实际上只用到 k 个,其中 k 代表组播组中的成员数目。

在每个矩阵 i 中,都引入一个新的 LP 型的独立的神经元,各矩阵中对应其矩阵编号 i 的第 i 个神经元和这个独立神经元都有连接。这个神经元将保证路由的连接度限制得到满足,同时在不同的矩阵中代表同一条链路的神经元之间也有连接。这些连接保证最终获得费用最优的组播路由,另外一方面也协作保证节点的连接度限制。对应于各组播成员的神经元矩阵输出的组合即为最终要求解的组播路由。

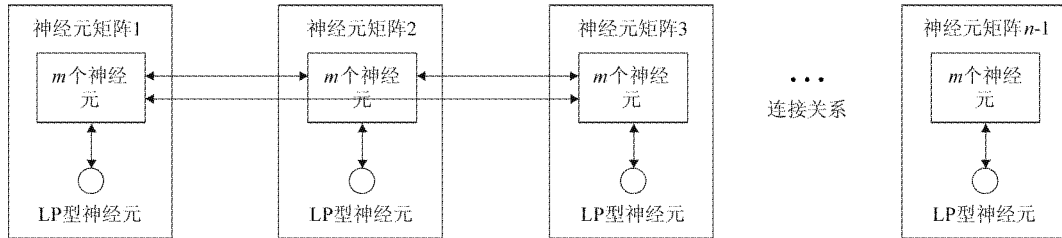


图 1 连接度受限的应用层组播神经网络模型——DRNN

2.2 受限组播路由问题定义

定义 v_i^k 为第 k 个神经元矩阵中对应链路 i 的神经元的输出, C_i 为链路 i 的费用。则连接度受限的应用层组播问题可以定义如下:

$$\text{最小化: } z = \sum_{j=1, j \neq k, j \in D}^n z_j, \quad z_j = \sum_{i=1}^m C_i v_i^k \quad (6)$$

$$\text{限制条件: } \mathbf{A}\mathbf{v} = \mathbf{\Phi} \quad (7)$$

$$v_i^k \in \{0, 1\}, \quad i = 1, 2, \dots, m, k = 1, 2, \dots, n \quad (8)$$

$$\sum_{k=1, k \in D}^n v_i^k \leq \Delta \quad (9)$$

上面定义中式(6)保证了最终组播路由的费用最小化。而(8)、(9)、(10)式分别对应于解的有效

性、合法性及连接度的限制条件。

2.3 非独立变量神经网络

利用改进后的神经网络模型求解最优路径问题,将神经元矩阵中对应每条链路的决策变量对应一个神经元的状态。与独立变量 $\mathbf{v}_l^k$ ($k = 1, 2, \dots, n$) 相对应的神经网络称为独立变量神经网络,与树支变量 $\mathbf{v}_t^k$ ($k = 1, 2, \dots, n$) 相对应的称为非独立变量神经网络。非独立变量神经网络的方程设为:

$$\mathbf{v}_t^k = \mathbf{w}_t \mathbf{v}_l^k + \mathbf{I}_t \quad (10)$$

比较式(5)和(10)得:

$$\mathbf{w}_t = -\mathbf{A}_t^{-1} \mathbf{A}_l = [w_{ij}] \quad (11)$$

$$I_i = A_i^{-1} \Phi \quad (12)$$

式中 w_i 表示从独立变量神经网络的状态反馈到非独立神经网络的连接矩阵, 为 $(n-1) \times (m-n+1)$ 阶矩阵; I_i 表示非独立变量神经网络的偏置, 为 $(n-1) \times 1$ 阶矩阵。

2.4 独立变量神经网络

只有独立变量神经网络的运动方程才取决于最优路径问题, 它的状态能引导整个神经网络趋于最优路径的解。

为每个神经元矩阵定义 lyapunov 能量函数 E^k , 表达式为

$$E^k = \sum_{i=1}^m (\rho_1 C_i f_i^k(\cdot) v_i^k + \rho_2 (v_i^k)^2 (1 - v_i^k)^2) + \rho_3 \int h(z) dz \quad (13)$$

它的最小化过程对应于神经网络的能量最小化。神经网络的稳定状态对应最优路径的解。

(13) 式中第一项保证了所求路径的费用最小。其中

$$f_i^k(\cdot) = \frac{1}{1 + \sum_{j=1, j \neq k, j \in D}^n v_j^k} \quad (14)$$

保证了任意一条被选中链路的费用在最终的组播路由中只被计算一次。第二项保证了 $v_i^k \in \{0, 1\}$ 时 ($i = 1, 2, \dots, m$) 最小。第三项 $\int h(z) dz$ 保证了节点连接度限制条件的满足, 其中 $h(z)$ 是模型中引入的 LP 型的神元元的传递函数, 其形式如下:

$$h(z) = \begin{cases} 0 & \text{如果 } z = \sum_{k=1, k \in D}^n v_i^k - \Delta \leq 0 \\ z & z = \sum_{k=1, k \in D}^n v_i^k - \Delta > 0 \end{cases} \quad (15)$$

$h(z)$ 的取值只在节点满足连接度限制时才为零, 其余情况下都大于零。

对式(13)求导, 并由著名的负梯度公式可得:

$$\begin{aligned} \frac{dv_i^k}{dt} = - \frac{\partial E^k}{\partial v_i^k} = & -\rho_2 (4v_i^3 - 6v_i^2 + 2v_i) \\ & - \rho_2 \sum_{j=1}^{n-1} w_{ij} (4v_i^3 - 6v_i^2 + 2v_i) \\ & - \rho_1 f_i^k(\cdot) C_i - \rho_1 \sum_{j=1}^{n-1} f_j^k(\cdot) w_{ij} C_j \\ & - \rho_3 L_i h(z), \\ & i = n, \dots, m \end{aligned} \quad (16)$$

独立变量神经元采用 Hopfield 模型, 由一个加法器、一个电容及输出函数组成, 其运动方程表示

为:

$$C_i \frac{dv_i^k}{dt} = \sum_{j=1}^m w_{ij} x_j(t) + I_{li} \quad (17)$$

比较(16)和(17)式并令 $j = 1/\rho_2, k = j\rho_1 = \rho_1/\rho_2, l = \rho_3/\rho_2$, 可得

$$x_j = g(v_j) = 4v_j^3 - 6v_j^2 + 2v_j \quad (18)$$

$$w_{lij} = \begin{cases} -1, & j = i \\ 0, & j \neq i, j = n, \dots, m \\ -w_{lij}, & j = 1, 2, \dots, n-1 \end{cases} \quad (19)$$

$$I_{li} = -j f_i^k(\cdot) C_i - k \sum_{j=1}^{n-1} f_j^k(\cdot) w_{lij} C_j - l L_i h(z) \quad (20)$$

(19)和(20)式是独立变量神经元的权值矩阵和偏置矩阵, 非独立变量神经元依靠(5)式的线性关系由独立变量神经元求得, 其对应的权值矩阵及偏置矩阵见式(11)和(12)。

3 性能模拟

3.1 模拟过程中需要考虑的一些因素

DRNN 的运动方程式采用四阶龙格-库塔法求解, 计算时间步长 $\Delta t = 10^{-5}$, 输出收敛精度 $\Delta v = 10^{-5}$, 各边的费用值选择为在 1 与 3 之间的随机数。

运动方程中系数的选择会影响最终解的质量。DRNN 的可选系数有 3 个。第一个系数是电容值 j , 它对应每个神经元的时间常量。 j 越小, 收敛速度越快, 但过分小的 j 不能保证足够时间搜索最优值, 使计算误差增大, 甚至无解。第二个系数 k 影响路径计算的精度。 k 越小, 路径费用项在能量函数中的份额越小, 可能增大神经网络计算得到的路径与最优路径的误差; k 越大, 神经网络计算得到的路径越接近最短路径, 但过大时会影响神经网络收敛于 0 或 1。第三个系数 l 同 k 类似, 它同样影响路径计算的精度。 l 越小, 连接度限制条件在能量函数中占的比例越小, 计算出来的路径可能不能满足节点的连接度限制条件; l 越大, 计算所得的路径能满足限制条件, 但同时也会影响神经网络的收敛。

提高所得路径的准确率与最终结果变量收敛于 0 或 1 是矛盾的, 但这两个因素在运动方程中的作用是相互分离的, 费用参数和度限制条件仅表现在神经网络的偏置项即 I_l 中。因此, DRNN 采取如下措施: 先考虑偏置项 I_l , 使神经网络的状态在最短路径和时延限制要求收敛到一定的精度, 然后再将偏置项 I_l 置 0, 从而让状态变量收敛于 0 或 1。即

DRNN 的收敛经过两个阶段。在第一个收敛阶段中,由于需要满足连接度限制的条件,采用一种在循环过程中自适应地修改参数的方法。每个神经元不同的 ρ_1 值对应不同的 k 值,在收敛过程中,一旦 (21) 式的条件得不到满足,就使用 (23) 式改变组成非法路径的链路的权重系数,通过神经元方程的运动,它们最终将构建一条满足条件的路由。

$$\sum_{k=1, k \in D}^n \text{sgn}(v_i^k) \leq \Delta \quad (21)$$

式中

$$\text{sgn}(z) = \begin{cases} 1 & \text{如果 } z \geq 0.5 \\ 0 & \text{其它} \end{cases} \quad (22)$$

$$\rho_{1,i}^{k,j+1} = \rho_{1,i}^{k,j} \left(1 + \frac{1}{1000}\right) \quad (23)$$

式中 $\rho_{1,i}^{k,j}$ 代表神经元矩阵 k 中的第 i 个神经元在第 j 次循环过程中 ρ_1 的取值。系数 l 和 j 在循环开始时的取值较小以发现一条最小费用路径,系数 k 的取值与前两者相比要大一些,但也不能设置得过大,以防止收敛过程进行得过快,不利于所有矩阵之间的协作。基于此原因,系数的初始值设置如下:

$$j = 0.01; \quad k = 10; \quad l = 20$$

经过多次模拟实验,在各边费用取值在 1~3 之间, j 、 k 、 l 的取值范围分别在 (0.001, 0.02)、(4.5, 13.5)、(7.5, 22.5) 之间为宜。

3.2 模拟结果

为检验改进的神经网络模型——DRNN 的有效性,将它与已有的启发式的组播路由算法(包括独立启发式(Ind)算法,由 Widyono^[8]提出的 CAO,以及文献[9]中提出的 BRON 算法)进行比较,同时也把该神经网络算法与文献[7]提出的改进的 Hopfield NN 组播算法 MHMNN 进行了比较。用于比较的网络拓扑图包含有 10 个节点和 5 个目的地,图中连接节点的边及边的费用都是随机产生的。表 1 是比较结果。

从表 1 中可以看出,与最简单的启发式组播路由算法 Ind 比较,DRNN 发现最优解的概率大大提高(从 20% 提高到 60%),而与其它改进后的启发式算法以及 MHMNN 神经网络算法比较时,发现最优解的概率大体相同(都在 60%~70% 之间)。

表 1 网络规模相同条件下几种算法的性能比较

网络情况	连接度 限制(Δ)	Ind	CAO	BRON	MHMNN	DRNN	最优解
拓朴(1)	3	10	12	12	10	10	10
	4	10	11	10	10	10	10
	5	10	9	8	8	8	8
拓朴(2)	3	15	12	12	9	9	9
	4	15	10	10	9	9	9
	5	15	8	8	8	9	8
拓朴(3)	3	9	9	9	9	10	9
	4	9	8	8	9	9	8
	5	9	8	8	9	9	8
拓朴(4)	3	12	10	10	10	10	10
	4	12	9	9	9	10	9
	5	12	9	9	9	9	9
拓朴(5)	3	9	9	9	9	9	7
	4	9	6	6	7	6	6
	5	9	6	6	7	6	6

在接下来的实验中,我们改变网络的规模,分别为 N(5,8)、N(10,16)、N(15,27)和 N(20,37),它们分别对应于包含有 5、10、15 和 20 个节点,8、16、27 和 37 条边的 4 种图规模,边的费用仍然随机产生。对应于这 4 种图其组播成员数分别为 3、5、10、15,连接度限制条件 Δ 取 4。对每种网络我们重复实验 50 次。在这些前提下对各种方法发现最优组播路由的概率及平均计算时间作比较,结果见表 2,表中的值是所有 50 次实验的平均值。计算时间的统计是基于以下的计算机配置:CPU 为 AMD 的 1800+, RAM 为 512MB。

表 2 不同网络规模条件下几种算法的性能比较

	N(5,8)		N(10,16)		N(15,27)		N(20,37)	
	发现最优解 的概率(%)	平均计算 时间(ms)	发现最优解 的概率(%)	平均计算 时间(ms)	发现最优解 的概率(%)	平均计算 时间(ms)	发现最优解 的概率(%)	平均计算 时间(ms)
Ind	27	80.298	20	87.3132	20	103.8741	33	133.5231
CAO	60	63.348	60	72.4231	67	78.6334	53	90.8971
BRON	67	42.1837	73	48.7371	80	62.4398	80	75.3219
MHNN	80	20.0512	67	21.332	73	25.266	60	43.106
DRNN	80	10.1076	60	13.6844	80	16.492	67	16.2428

从表 2 中看出,与启发式的组播路由算法比较,基于神经网络的组播路由算法在计算时间上大大缩短,而发现最优解的概率则与启发式算法相当。而与基于 Hopfield 神经网络的算法比较起来,其性能相当,但所用的神经元数目有比较大的减少,当组播组有 k 个成员时, MHMNN 需要 $k \times n^2$ 个神经元,而 DRNN 只需要 $k \times m$ 个,其中 n, m 分别为有向图中的顶点数和边数。

4 结 语

提出了一种基于改进的双层递归神经网络模型求解受限组播路由问题的方案。在原有模型中引入了神经元矩阵及 LP 型神经元,并利用了原模型中的 Kirchhoff 条件,从而使得改进后的方案与其它神经网络求解方案相比,具有神经元数目少、运算速度快、寻找最优路径的成功率高等特点。但该方案仍然是一种集中式的算法,每个节点都需要了解网络中其它节点的情况,因此本方案的扩展性不强,只适用于中小型规模的网络。

参考文献:

- [1] Chu Y, Rao S, Seshan S, et al. Enabling conferencing applications on the Internet using an overlay multicast architecture. In: Proceedings of the ACM SIGCOMM, San Diego, USA, 2001. 55-67
- [2] Banerjee S, Bhattacharjee B, Kommareddy C, et al. Scalable application layer multicast. In: Proceedings of the ACM SIGCOMM, Pennsylvania, USA, 2002, 32(4):205-217
- [3] Chalmers R, Almeroth K. On the topology of multicast trees. *IEEE/ACM Trans on Networking*, 2003, 11:153-165
- [4] Hua K, Tran D, Villafane R. Overlay multicast for video on demand on the Internet. In: Proceedings of the ACM Symposium on Applied Computing, Florida, USA, 2003, 935-942
- [5] Jannotti J, Gifford D, Johnson K, et al. Overcast: Reliable multicasting with an overlay network. In: Proceedings of the USENIX Symposium on Operating System Design and Implementation, San Diego, USA, 2000, 192-212
- [6] Shi S, Turner J. Routing in overlay multicast networks. In: Proceedings of INFOCOM 2002, New York, USA, 2002. 1200-1208
- [7] Kompella V. Multicast routing algorithms for multimedia traffic: [Ph.D dissertation]. San Diego: University of California, 1993
- [8] Widyono R. The design and evaluation of routing algorithms for real-time channels: [Technical Report TR-94-024]. California: International Computer Science Institute, 1994
- [9] Zhang X, Zhang G. A multicast routing algorithm for overlay network built on leased lines. In: Proceedings of the Symposium on Applications & the Internet, Tokyo, Japan, 2005. 118-124
- [10] Ali M, Kamount F. Neural networks for shortest path computation and routing in computer networks. *IEEE Trans on Neural Networks*, 1993, 4(6):941-953
- [11] Xia Z, Li P, Yen I. A neural network based approach for overlay multicast in media streaming systems. In: Proceedings of the International Parallel and Distributed Processing Symposium, Santa, USA, 2004, 54-62
- [12] Filip A, Bernardete R, Luis R, et al. A neural network for shortest path computation. *IEEE Trans on Neural Networks*, 2001, 12(5):1067-1073
- [13] Parnavalai C, Chakraborty G, Shiratori N. A neural network approach to multicast routing in real-time communication networks. In: Proceedings of the International Conference on Network Protocols, Tokyo, Japan, 1995, 332-339
- [14] Strum R D, Ward J R. Electric Circuits and Networks (2nd edition). New Jersey: Prentice-Hall Inc, 1985

A modified neural network based approach for overlay multicast

Liu Shidong, Zhang Shunyi, Zhou Jingquan

(Institute of Information Network Technology, Nanjing University of Posts and Telecommunications, Nanjing 210003)

Abstract

This paper presents a modified dual-layer recurrent neural network (DRNN) to solve the optimal path problem for application multicast routing. A linear programmable-type neuron is introduced and neuron matrix connections of the existing two-layer recurrent architecture are changed to enable the constrained multicast routing algorithm to be implemented. Due to the Kirchhoff term, the quality of the solution is improved. Compared with other algorithms, it has the advantages of shorter computation time, smaller numbers of neurons and higher solution precision.

Key words: neural network (NN), neuron, multicast, overlay network, constrained term, independent variables