

协同基站群的利益树分簇算法

郑丽清 黄开枝 吉 江

(国家数字交换系统工程技术研究中心 郑州 450002)

摘 要 以系统和速率最大化为目标,将协同基站群的分簇问题建模为带权连通图的最大利益树生成问题,提出了一种基于协同度的最大利益树分簇算法。定义了利益树的协同度,并选择了协同度最大的两棵利益树进行合并的方式来并行生成多个规模动态的协同分簇,从而解决了传统顺序分簇导致的系统性能受限的问题,提高了系统的分簇性能。仿真结果表明,本算法的系统频谱效率优于现有的动态分簇算法,并且算法为线性复杂度。

关键词 协同,分簇,小区间干扰,多基站联合处理

中图分类号 TN929 **文献标识码** A

Benefit-tree Clustering Algorithm for Base Station Cooperation

ZHENG Li-qing HUANG Kai-zhi JI Jiang

(National Digital Switching System Engineering & Technological Research Center, Zhengzhou 450002, China)

Abstract Aiming at the objective of maximizing system sum-rate, a benefit-tree clustering algorithm based on maximizing Cooperative Degree (CD) was proposed, where the clustering problem in base station (BS) cooperation system is modeled as constructing benefit-trees of a connected graph with edge costs. The concept of CD between every two benefit-trees was defined, and based on this, the algorithm simultaneously generates some clusters of dynamic size by combining two trees with the maximum CD, which could solve the capacity-limited problem caused by conventional clustering scheme and enhance the system capacity. Results show this algorithm performs better than other existing clustering scheme in the system spectrum efficiency, and the computational complexity is linear.

Keywords Cooperation, Cluster, Inter-cell interference (ICI), Multi-BS joint processing

1 引言

多基站协同技术^[1,2]能够有效抑制小区间干扰(Inter-Cell Interference, ICI),提高系统的频谱效率,因而已经成为无线传输技术研究的热点^[3,4]。但是,该技术需要基站间共享全部或部分的信道状态信息(Channel State Information, CSI)以及大量的用户数据信息^[5],所以当系统规模较大时,将给基站间的传输网络带来难以负荷的信息交互量^[6,7]。为解决此问题,一般对可协同的基站群进行分簇^[8,9],即将整个系统划分为多个小的分簇,且令簇内的基站进行协同处理。按照通信链路上下行过程中协同方式的不同,文献[10]主要针对上行系统,将整个网络静态划分为多个分簇并利用协同技术消除簇内干扰(Intra-Cluster Interference),从而提高了系统的频谱效率;文献[11]则重点研究了下行情况下的静态分簇方案。但这两种分簇方法均属于静态算法,无法适应实际系统中信道的时变性,会损失一定的系统吞吐量。因此近两年出现了动态分簇算法,以弥补静态分簇算法的不足。文献[12-14]利用实时的CSI来设计动态分簇算法,以获得比静态方案更好的系统性能,但仍存在以下问题。

1)算法在分簇时需要预先确定簇的大小。静态的分簇规模无法动态地适应实际蜂窝系统的干扰分布情况,可能导致较差的分簇结构,最终会降低系统的整体性能。

2)算法在分簇时是顺序的。随着顺序生成的基站簇,可候选的协作基站数不断减少,因此可能导致后续生成的基站簇协同性能较差。

3)分簇算法的迭代过程中,搜索协同基站带来了巨大的计算复杂度。

为解决上述问题,本文提出了一种利益树动态分簇算法。算法从协同基站群的拓扑结构着手,将其分簇问题建模为有向带权连通图的最大利益树生成问题,视每个基站为图中的一个节点。然后,利用信道状态信息计算出每两个节点相互合作给双方带来的信噪比增益,并将双方的信噪比增益定义为两个节点间边的权值。考虑到实际系统的处理能力,假设簇的大小不得超过 K ,另外以基站协同性能最大为目标,将其等价于树的利益最大化问题,提出了一种基于协同度的最大利益树分簇算法。算法利用节点间的权值来定义树与树之间的协同度,并不断选取协同度最大的两棵树进行合并,直到所有树间的协同度均等于零,最终生成的多棵树即为系统的分

到稿日期:2011-03-01 返修日期:2011-06-25 本文受国家科技重大项目(2008ZX03003-004),中国国家基础研究发展计划(2007CB310608)资助。

郑丽清(1987-),女,硕士生,主要研究领域为移动无线通信,E-mail:zhengliqing81ndsc@126.com;黄开枝(1973-),女,副教授,硕士生导师,主要研究领域为移动无线通信。

簇结果。由于在分簇时是完全基于节点间的权值来选择协作基地的,不需要事先确定簇的大小且克服了顺序分簇的缺点,因此大大改善了分簇性能,提高了系统频谱效率。仿真结果表明,相比现有的动态分簇算法,本文算法在复杂度及系统的频谱效率上都有提高。

2 模型建立及问题描述

考虑一个上行蜂窝系统模型, N 个小区, 每个小区中心均分布一个基站。设 N 个基站间采用有线链路相连, 则将此 N 个基站称为一个协同基站群。此时若 N 个基站一起采用协同处理技术, 则系统可以获得最佳的信道容量^[1]。但由于实际系统中网络传输能力及计算能力受限, 因此允许一起合作的基站个数最多为 $K, K \leq N$, 即簇的大小限制为 K 。本文研究的是如何将一个协同基站群划分为多个小簇, 使得最终的分簇性能最佳。

2.1 信道模型

假设基站及终端都配置单根天线, 且同一时频资源块上只服务一个用户, 即小区内不存在用户间干扰。在一个时频资源块上, 有信道矩阵 $\mathbf{H} \in \mathbf{C}^{N \times N}, h_{ij} \in \mathbf{H}$, 其中 h_{ij} 为第 i 用户到第 j 个基站的信道参数。令整个信道模型为

$$\mathbf{y} = \mathbf{H} \cdot \mathbf{B} \cdot \sqrt{P} \cdot \mathbf{u} + \mathbf{n}_0 \quad (1)$$

式中, $\mathbf{y} \in \mathbf{C}^{N \times 1}$ 为接收信号矢量, $\mathbf{u} \in \mathbf{C}^{N \times 1}$ 为发射符号矢量, $\mathbf{E}[\mathbf{u}\mathbf{u}^H] = \mathbf{I}_N, \mathbf{I}_N$ 为 N 维单位对角矩阵, P 为用户的发射功率常量, $\mathbf{B} \in \mathbf{C}^{N \times N}$ 为基站端的协同处理矩阵, $B_{ij} \in \mathbf{B}$ 为第 i 个基站对第 j 个用户发送信号的处理常量, \mathbf{n}_0 为高斯白噪声向量, 服从均值为 0、方差为 σ^2 的高斯分布。定义信干比矩阵 \mathbf{SIM} 来描述各基站间的干扰情况。

$$\mathbf{SIM} = \begin{bmatrix} S_{11} & I_{12} & \cdots & I_{1N} \\ I_{21} & S_{22} & \cdots & I_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ I_{N1} & I_{N2} & \cdots & S_{NN} \end{bmatrix} \quad (2)$$

式中, $S_{ii} = P \times \|h_{ii}\|^2; I_{ij} = P \times \|h_{ij}\|^2; i=1, \dots, N; j=1, \dots, N$ 。

2.2 利益树分簇模型

用一个有向带权连通图 $G=[V, W]$ 来描述这个蜂窝系统。其中 V 为图的顶点集合, 每个基站为一个顶点, 有 $|V|=N$, 符号 $|\cdot|$ 表示集合中元素的个数。 $W \in \mathbf{C}^{N \times N}, w(i, j) \in W, i \neq j$, 表示顶点 i 到顶点 j 的边的权值。用一棵树 $t = \{b_1, b_2, \dots, b_m, \dots, b_l\}$ 来表示一个协同簇, 其中 b_i 为树中节点(基站)的编号。由于系统会生成多个协同簇, 即会生成多棵树, 故用一个森林 $F=[N_T, T]$ 来描述协同基站群的分簇结果, 其中 N_T 为森林中树的棵数, T 为森林中树的集合, 有 $T = \{t_1, t_2, \dots, t_i, \dots, t_{N_T}\}, t_i$ 为森林中的第 i 颗树。基于此, 协同基站群的分簇问题可建模为带权连通图 G 的最大利益树生成问题。即已知一个有向带权连通图 G , 如何生成一个森林, 使得该森林中每棵树中的节点个数不能超过 K , 且该森林中所有树的利益之和最大。为此, 引入利益树的概念。对于一棵树, 如果它是有利益的, 则这棵树称为利益树。对于树 t_i , 它的利益 $\Gamma(t_i)$ 由 t_i 内节点间的权值之和决定, 有

$$\Gamma(t_i) = \sum_{b_m \in t_i} \sum_{b_n \in t_i, b_m \neq b_n} w(b_m, b_n) \quad (3)$$

由于本文中的树表示一个簇, 故这里的树由节点组成, 不存在边的含义。后面本文中所涉及的树都是指这种利益

树, 相应的森林称为利益森林。其中森林的利益定义为森林中每棵树的利益之和, 即

$$\Gamma(F) = \sum_{t_i \in T} \{\Gamma(t_i)\} \quad (4)$$

因此, 协同基站群的分簇问题等价于寻找一个利益最大的森林 $F^*=[N^*, T^*]$, 这个森林需满足条件

$$\begin{cases} \Gamma(F^*) = \max_F \{\Gamma(F)\} \\ \max_{t_i \in T^*} (|t_i|) \leq K \end{cases} \quad (5)$$

2.3 权值设计

由于森林 F^* 是系统最终的分簇结果, 因此森林的利益 $\Gamma(F^*)$ 与系统的分簇性能息息相关。而从式(3)中知, 最终森林的利益与节点间的权值 $w(i, j)$ 有关。因此连通图 G 中的权值设计是影响系统分簇性能的关键。下面分析分簇性能与 $w(i, j)$ 的关系。

令系统的和速率为 R_{sum} , 则

$$R_{\text{sum}} = \sum_{i=1}^N \log_2(1 + \text{SINR}_i) \quad (6)$$

若不采用接收端的协同处理技术, 对于基站 i , 其接收信干噪比为

$$\text{SINR}_i = \frac{|h_{ii} \cdot B_{ii}|^2}{\sigma^2/P + \sum_{k \neq i, k=1}^N |h_{ki} \cdot B_{ki}|^2} \quad (7)$$

若用户 i 与用户 j 所在基站采用简单的线性波束成形处理技术^[12], 即令

$$\mathbf{B} = [\mathbf{H}_{ij}^H \cdot \mathbf{H}_{ij}]^{-1} \cdot \mathbf{H}_{ij}^H \quad (8)$$

式中, $\mathbf{H}_{ij} \in \mathbf{C}^{2 \times 2}$ 为用户 i, j 到相应两个基站的信道矩阵。由于用户 j 对用户 i 的小区间干扰被消除, 故基站 i 的接收信干噪比增大为

$$\text{SINR}_i^{\text{oop}} = \frac{|h_{ii} \cdot B_{ii}|^2}{\sigma^2/P + \sum_{k \neq i, k \neq j, k=1}^N |h_{ki} B_{ki}|^2} \quad (9)$$

对比式(7)与式(9), 采用协同处理技术可以提高基站端的接收信干噪比。故将分簇目标近似简化为分簇后各基站协同所带来的信干噪比增益之和最大, 则设计节点 i 到节点 j 的权值为节点 i 与 j 合作给节点 i 带来的信噪比增益, 即 $w(i, j) = \text{SINR}_i^{\text{oop}} / \text{SINR}_i$, 从而

$$w(i, j) = \frac{\sigma^2/P + \sum_{k \neq i, k \neq j, k=1}^N |h_{ki} B_{ki}|^2}{\sigma^2/P + \sum_{k \neq i, k=1}^N |h_{ki} B_{ki}|^2 + |h_{ji} B_{ji}|^2} \quad (10)$$

式(10)中 $w(i, j)$ 的设计相对复杂, 而从式(10)中可看出基站 i 与 j 合作和 i 不与 j 合作对 i 的信噪比增益与 h_{ji} 的大小非常相关, 即 h_{ji} 越大, 带来的增益越大。因此可以直接利用信干比矩阵 \mathbf{SIM} 中的干扰分量来定义节点间的权值。令 $w(i, j) = \|h_{ji}\|^2 / \|h_{ii}\|^2$, 即

$$w(i, j) = \frac{I_{ji}}{S_{ii}} \quad (11)$$

若将式(11)带入式(3)中, 则从式(5)中知算法的目标就是使所有树内节点的干扰最大。这样, 利用协同处理技术可以消除的干扰分量也最大, 系统最终的分簇性能也最佳。

3 基于协同度的最大利益树分簇算法

3.1 最大利益树分簇算法

已知图 G , 如何生成森林 $F^*=[N^*, T^*]$, 使得森林的利益最大。为了解决这个问题, 设计了基于协同度的最大利益

树分簇算法。定义树 t_i 与树 t_j 间的协同度 $C(t_i, t_j), i \neq j; i=1, \dots, N; j=1, \dots, N$ 为树 t_i 中节点与树 t_j 中节点相互间的权值之和的归一化, 即

$$C(t_i, t_j) = \frac{1}{(|t_i| + |t_j|)} \left\{ \sum_{b_m \in t_i} \sum_{b_n \in t_j} w(b_m, b_n) + w(b_n, b_m) \right\} \quad (12)$$

树的节点数越多, 其权值之和就越大。为了消除树的规模对协同度的影响, 在式(12)中进行了简单的归一化。算法开始时, 将 G 视为一个初始森林, 森林中有 N 棵树, 每棵树只含一个节点(基站)。在迭代过程中, 不断选择协同度最大的两棵树进行合并, 并更新森林中其他树与新生成的树间的协同度, 直到所有树间协同度为零, 算法结束。具体算法流程如下。

1) 算法初始化

将图 G 转化为一个初始森林 $F^0 = [N_{T_0}, T_0]$, 其中 $N_{T_0} = N, T_0$ 中每棵树只含一个节点, 每个节点为独立的一棵树。初始森林中每两棵树间的协同度为

$$C(t_i, t_j) = (w(b_i, b_j) + w(b_j, b_i)) / 2 \quad (13)$$

2) 算法迭代

步骤一 选出相互协同度最大的两棵树 t_m', t_n' , 并将它们合并为一棵新树 t_{new} , 此时森林中的树减少一颗。

$$C(t_m', t_n') = \max_{t_i \in T, t_j \in T} \{C(t_i, t_j)\} \quad (14)$$

$$t_{new} = t_m' \cup t_n' \quad (15)$$

$$N_T \leftarrow N_T - 1 \quad (16)$$

步骤二 更新其它树与新生成的树 t_{new} 的协同度。如果两棵树中的节点个数之和超过 K , 则令两棵树的协同度为零。

$$C(t_i, t_j) = \begin{cases} \frac{\sum_{b_m \in t_i} \sum_{b_n \in t_j} (w(b_m, b_n) + w(b_n, b_m))}{(|t_i| + |t_j|)}, & |t_i| + |t_j| \leq K \\ 0, & \text{否则} \end{cases} \quad (17)$$

步骤三 判断所有树间的协同度是否都为零, 若为零, 则算法结束, 输出分簇结果, 否则返回步骤一。

在算法迭代中, 由于树的规模越来越大, 因此会逐渐出现两棵树的节点个数大于 K 的情况, 此时根据式(17), 对应的协同度将被置零, 最终会导致所有树间的协同度都为零, 算法结束。在算法中, 最后生成的森林中树的规模及树的棵树都是动态的, 完全是基于权值而自适应变化的, 所以不存在文献[12]中预先确定簇大小的问题。而且算法中的多棵树同时生成, 并不是按顺序逐一生成, 不会破坏最佳的分簇结构, 故系统频谱效率优于现有的动态分簇算法。

下面利用定理 1 说明所提算法的有效性。

定理 1 最大利益树分簇算法能够生成树规模限制下利益最大的森林 $F^* = [N^*, T^*]$ 。

证明: 记 t_m', t_n' 为第 i 次迭代时找到的协同度最大的两棵树, $F^i = [N_{T_i}, T_i], i=1, \dots, (N-N^*)$ 为第 i 次迭代时生成的森林, t_{new}^i 为森林 F^i 中新生成的一棵树, 有 $t_{new}^i = t_m' \cup t_n'$ 。

对一个初始森林 F^0 , 有树 $t_i^0, i=1, 2, \dots, N$, 每棵树中只含一个节点。由式(3)知 $\Gamma(t_i) = 0$, 故此时森林的利益为 0, 即 $\Gamma(F^0) = 0$ 。根据算法步骤一, 第一次迭代时两棵吸引力最大的树 t_m^0, t_n^0 合并为一颗新的树 t_{new}^1 , 此时

$$\Gamma(F^1) = \sum_{t \in T_0, t \neq t_{new}^1} \Gamma(t) + \Gamma(t_{new}^1) \quad (18)$$

根据式(3),

$$\Gamma(t_{new}^1) = \Gamma(t_m^0) + \Gamma(t_n^0) + C(t_m^0, t_n^0) \cdot (|t_m^0| + |t_n^0|) \quad (19)$$

因此,

$$\begin{aligned} \Gamma(F^1) &= \sum_{t \in T_0, t \neq t_{new}^1} \Gamma(t) + \Gamma(t_m^0) + \Gamma(t_n^0) + C(t_m^0, t_n^0) \cdot \\ &\quad (|t_m^0| + |t_n^0|) \\ &= \Gamma(F^0) + C(t_m^0, t_n^0) \cdot (|t_m^0| + |t_n^0|) \\ &= C(t_m^0, t_n^0) \cdot (|t_m^0| + |t_n^0|) \end{aligned} \quad (20)$$

由式(14)及(12)可知 F^1 为森林棵数为 $(N-1)$ 时利益最大的森林。

若已知 $G(F^i)$, 对第 $i+1$ 次迭代, 同理可得

$$\Gamma(F^{i+1}) = \Gamma(F^i) + C(t_m^i, t_n^i) \cdot (|t_m^i| + |t_n^i|) \quad (21)$$

故 F^{i+1} 为森林棵数为 $(N-i-1)$ 时利益最大的森林。这样, 可同理推出 F^* 为森林棵数为 N^* 时利益最大的森林。其中, 最大利益森林中的棵数 N^* 受分簇规模限制参数 K 影响, 故最大利益树分簇算法可以得到树规模限制条件下的最大利益森林, 算法得证。

3.2 利益树规模调整

式(12)中协同度的大小取决于节点间的权值大小, 并不考虑树的规模对协同度的影响。因此可能导致基站群的分簇结果中有些簇较大, 有些簇较小, 分布相对不集中的现象。对此, 对于两棵规模较大的树, 可以适当降低它们之间的协同度, 减少它们合并的可能性, 把合并的机会更多地留给规模较小的树。因此可在协同度 $C(t_i, t_j)$ 设计中引入排斥因子 η , 令

$$C(t_i, t_j) = \eta \cdot \left(\sum_{b_m \in t_i} \sum_{b_n \in t_j} I_{b_m, b_n} + I_{b_n, b_m} \right) / (|t_i| + |t_j|) \quad (22)$$

式中, $\eta = f(|t_i| + |t_j|)$, $f(\cdot)$ 是一个递减函数。当两棵树的节点数之和越大时, 也即簇内协同基站个数越多时, 排斥因子越大, 簇内基站个数越少的分簇就越有可能与别的基站合作。通过这种方式可以调整树的规模, 使最终簇的大小集中在较小的范围, 以降低基站协同处理的开销及复杂度。

3.3 算法的开销及复杂度分析

利益树分簇算法要求已知完全的信道状态信息矩阵 \mathbf{H} , 但根据不同的信道环境可以选用不同的信道参数。即可以选用实时完整的信道参数, 同时考虑大、小尺度的信道衰落影响, 也可以只考虑大尺度衰落参数的影响。本文在分簇时仅考虑了大尺度衰落参数, 即只要大尺度参数不变, 系统的分簇结果也不变。从算法复杂度来讲, 由于算法中每次迭代森林中都有两棵树进行合并, 故迭代次数必然小于初始森林中树的个数, 即小于 N 。图 1、图 2 分别给出了算法迭代次数与分簇规模限制参数 K 以及小区个数 N 的关系。利用曲线拟合方式得出算法迭代次数与 K 是 $\log(\cdot)$ 的关系, 与 N 是线性关系。故算法的时间复杂度为 $O(\log(K) \cdot N)$ 。

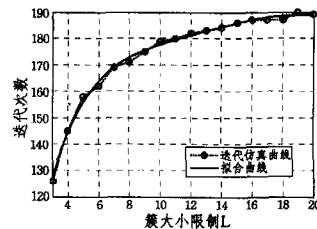


图 1 算法迭代次数与 K 的关系 ($N=200$)

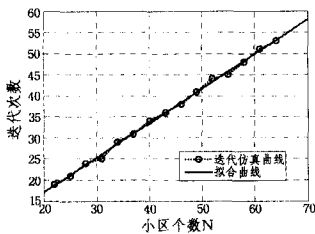


图2 算法迭代次数与N的关系(K=6)

4 仿真结果及分析

假设小区个数 $N=61$, 半径为 1km, 分簇时只考虑大尺度衰落的影响, 即令信道参数 $h_{ij} = a_{ij}$, a_{ij} 表示大尺度衰落, 有 $a_{ij}^2 = PL_{ij} \cdot \phi_{ij}$, 其中 ϕ_{ij} 为阴影衰落, 方差设为 8dB, PL_{ij} 为路径衰落, 参数设置参照 3GPP LTE 信道模型^[12]; $PL_{ij} = 148.1 + 37.6 \log(d_{ij})$ 。其中, d_{ij} 表示用户 i 到基站 j 的距离。

令 $K=6$, 利用利益树分簇算法进行分簇, 分簇结果如图 3 所示。可以看出, 簇的大小是根据实际的信道情况自适应动态变化的, 且不超过 K 。

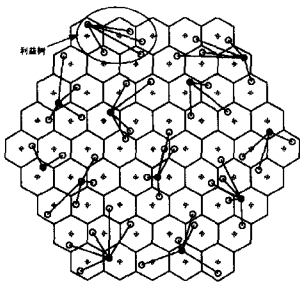


图3 算法的分簇结果(N=61)

为了便于比较, 图 4 中给出了静态、半动态^[12] (在文献^[13]中将文献^[12]中的分簇算法定义为半动态分簇算法) 以及完全动态^[13] 3 类分簇算法的仿真性能。半动态分簇方案中固定簇大小为 6, 完全动态方案采用 0.1 的门限方式。从图 4 可以看出, 动态分簇方案性能优于静态分簇性能, 且本文方案的频谱效率高於其它两种动态方案。

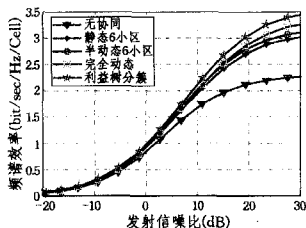


图4 几种不同分簇算法性能比较

在本文权值设计节中, 给出了两种不同的权值设计: 第一种是利用信噪比增益来刻画图中边的权值, 见式(10); 第二种为直接利用干扰分量设计权值的方案, 见式(11)。而在利益树规模调整中, 通过增加协同度中的排斥因子来调整树规模的分布。图 5 比较了这 3 种不同参数设计方案对系统性能的影响。可以看出, 利用信噪比增益的方案系统频谱效率最佳, 而采用干扰分量的方案略差, 但在设计上相对简单。在第 3 种方案中, 权值采取直接干扰分量定义, 并设计排斥因子为 $\eta = (K^2 - |i_j| + |i_j|)$ 。

从图 5 中可知, 引入排斥因子后性能下降。这主要是因為增加了排斥因子, 导致簇的大小相对变小, 降低了系统开

销, 但损失了系统频谱效率。

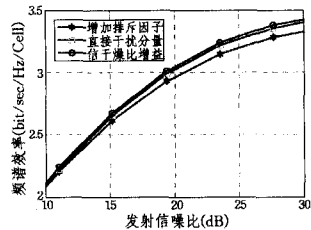


图5 3类不同参数设计对分簇性能的影响

结束语 本文将协同基站群的分簇问题建模为带权连通图的最大利益树生成问题, 提出了一种基于协同度的最大利益树分簇算法。算法利用信道状态信息计算每两棵树中节点相互合作给双方带来的信噪比增益之和, 定义为协同度; 并选择协同度最大的两棵树进行合并, 最终能够并行生成多个协同分簇。仿真结果表明, 本算法优于现有的动态分簇算法, 且算法复杂度为 $O(\log(K) \cdot N)$ 。本文算法虽然基于上行蜂窝系统, 但可以容易扩展到下行情况。

参考文献

- [1] Shamaï S, Zaidel B M. Enhancing the cellular downlink capacity via co-processing at the transmitting end [C]// IEEE Vehicular Technology Conference (VTC). Rhodes, Greece, 2001: 1745-1749
- [2] 谢显中, 雷维嘉. 抑制 MIMO 蜂窝系统中小区间干扰的多基站联合处理研究进展[J]. 数字通信, 2009, 36(3): 1-8
- [3] Gesbert D, Hanly S, Huang H, et al. Multi-cell MIMO Cooperative Networks: A New Look at Interference [J]. IEEE Journal on Selected Areas in Communications, 2010, 28(9): 1-29
- [4] Nam Y-H, Liu Ling-jia, Yan Wang, et al. Cooperative communication technologies for LTE-advanced [C]// Acoustics Speech and Signal Processing (ICASSP). Dallas, TX, 2010: 5610-5613
- [5] Yoon S, Somekh O, Simeone O, et al. A Comparison of Opportunistic Transmission Schemes with Reduced Channel Information Feedback in OFDMA Downlink [C]// Personal, Indoor and Mobile Radio Communications (PIMRC). Athens, 2007: 1-5
- [6] Niu Bo, Simeone O, Somekh O, et al. Ergodic and Outage Performance of Fading Broadcast Channels with 1-Bit Feedback [J]. IEEE Transactions on Vehicular Technology, 2010, 59(3): 1129-1282
- [7] Papadogiannis A, Bang H J, Gesbert D, et al. Downlink overhead reduction for multi-cell cooperative processing enabled wireless networks [C]// Personal, Indoor and Mobile Radio Communications (IEEE PIMRC). Cannes, France, 2008: 1-5
- [8] Hoydis J, Kobayashi M, Debbah M. On the Optimal Number of Cooperative Base Stations in Network MIMO Systems [J]. IEEE Transaction on Vehicular Technology, 2010, 59(3): 1282-1293
- [9] Ng C, Huang H. Linear Precoding in Cooperative MIMO Cellular Networks with Limited Coordination Clusters [J]. Selected Areas in Communications, 2010, 28(9): 1446-1454
- [10] Venkatesan S. Coordinating Base Stations for Greater Uplink Spectral Efficiency in a Cellular Network [C]// Personal, Indoor and Mobile Radio Communications (PIMRC). Athens, Greece, 2007: 1-5

各保留特征相应乘上各自的权重,产生加权保留特征,最后基于加权保留特征构建入侵检测模型。

4 仿真实验

4.1 数据来源

本研究实验数据通过模拟麻省理工学院林肯实验室提供的 KDD CUP99 数据集产生,总共包含 7000000 条连接信息,被标记为正常或被攻击。每条连接信息包括流量特征集、主机流量特征集、基本特征集和内容特征集等 41 维网络数据特征。另外,包含基于主机审计数据选择重要端口扫描次数、探测操作系统版本次数、IP 扫描次数、用户在磁盘上所建目录数量、敏感数据访问次数、用户的源、非法连接数、特殊文件访问数量、用户的地址、超级用户权限访问次数、用户程序所占线程数、在当前时期口令错误的次数、用户连接建立成功率等 13 个特征。随机抽取 2000000 条信息作为独立测试样本,剩余为训练样本。

4.2 模型及识别结果

基于上述新方法对训练样本构建入侵检测模型(screen-weight),并对独立测试进行判别;同时另外构建了 6 个参比模型,用于结果对比。这 6 个参比模型是基于遗传算法的入侵检测模型(GA);基于神经网络的模型(NN)^[7];基于 SVM 的模型,既不进行特征筛选也不加权;仅基于主机审计数据的模型(HIDS),除数据特征外,其它过程与新模型 screen-weight 一致;仅基于网络数据包的模型(NIDS);screen 模型,除不进行加权外,其它与新模型一致。另外,为检测不同网络流量下各模型的入侵检测情况,7 个检测模型分别在 900Mb/S 与 1300Mb/s 两种情况下进行了仿真实验。各模型检测精度见表 1。

表 1 不同模型检测精度

模型	网络流量为 900Mb/s 时检测精度	网络流量为 1300Mb/s 时检测精度
GA	83%	66%
NN	85%	71%
SVM	90%	75%
HIDS	89%	89%
NIDS	91%	87%
screen	96%	90%
screen-weight	98%	95%

4.3 结果分析

从以上结果可知,本文模型(screen-weight)在所有参比模型中检测精度最高,稳定性最好。

首先,从机器学习算法方面分析,通过遗传算法模型(GA)、神经网络模型(NN)、支持向量机模型(SVM)的比较可知,在网络流量为 900 Mb/s 时,SVM 精度最好;当网络速度增加时,其检测精度降低得最慢。显然,相比另外两种学习

方法,基于结构风险最小的 SVM 在入侵检测系统中具有更好的适用性。

其次,从数据特征选择方面来看,在网络流量为 900Mb/s 时,基于网络数据模型(NIDS)的检测精度略高于基于主机审计数据的 HIDS 模型,但两个模型检测精度都明显低于本文模型。可见,融合两类数据能有效增加特征信息量,提高检测模型的检测精度。

从特征筛选方面来看,当网络流量为 900 Mb/s 时,经过特征筛选的 screen 模型其检测精度高于未经过特征筛选的 SVM 模型 6 个百分点。可见,未经过筛选的特征中包含了大量的冗余或无用特征,导致模型检测精度下降;当网络流量增加到 1300 Mb/s 时,screen 模型精度仅下降了 6 个百分点,而 SVM 模型精度下降了 15 个百分点。可以认为,过多的特征使得模型计算复杂度增加,而丢失了大量的信息数据,最终导致了模型精度急剧下降。

最后,在特征加权方面,screen-weight 模型检测精度在两种网络流速下都高于 screen 模型。显然,每个保留特征对模型贡献并非一致,应该根据其信息含量赋予相应的权值,以相互区分。

结束语 本文提出了一种新的入侵检测模型,通过融合主机审计数据与网络数据,提高数据的覆盖面,增加了特征信息量;并基于 SVM 交叉测试,删除其中冗余与无用特征,降低模型运算复杂度;最后基于强制筛选为各保留特征赋予相应的权值,以进一步提高模型的检测精度。新模型能在保证运算复杂度的情况下,有效利用信息量,具有较高入侵检测精度,在入侵检测领域有较好的应用前景。

参考文献

- [1] 顾钧. 基于 KPCA 和 SVM 的网络入侵检测研究[J]. 计算机仿真, 2010, 27(7): 105-107
- [2] 张艳芳, 郭郁杰. 入侵检测关键技术研究[J]. 信息与电脑, 2010, (8): 16-17
- [3] 王树, 杜启军, 余桂贤, 等. 网络入侵检测系统的最优特征选择方法[J]. 计算机工程, 2010, 36(15): 140-142, 144
- [4] Lee W K. Feature Selection of Intrusion Data Using a Hybrid-Genetic Algorithm Approach [J]. Wireless Networks, 2007, 13(6): 459-460
- [5] 汪世义, 陶亮, 王华彬. 几种机器学习方法在 IDS 中的性能比较[J]. 计算机仿真, 2010, 27(8): 92-94, 121
- [6] Cortes C, Vapnik V. Support vector networks [J]. Machine Learning, 1995, 20: 273-297
- [7] 常卫东, 王正华, 鄢喜爱. 基于集成神经网络入侵检测系统的研究与实现[J]. 计算机仿真, 2007, 24(3): 134-137
- [8] 肖敏, 柴蓉, 杨富平, 等. 基于可拓集入侵检测模型[J]. 重庆邮电大学学报: 自然科学版, 2010, 22(3): 345-349
- [9] (上接第 88 页)
- [11] Boccardi F, Huang H. Limited Downlink Network Coordination in Cellular Networks [C]// Personal, Indoor and Mobile Radio Communications (PIMRC). Athens, Greece, 2007: 1-5
- [12] Papadogiannis A, Gesbert D, Hardouin E. A Dynamic Clustering Approach in Wireless Networks with Multi-cell Cooperative Processing [C]// IEEE International Conference on Communications (ICC). Beijing, China, 2008: 4033-4037
- [13] Liu Jingxin, Wang Dongming. An improved dynamic clustering algorithm for multi-user distributed antenna system [C]// Wireless Communications & Signal Processing (WCSP). China, 2009: 1-5
- [14] Zhou Sheng, Gong Jie, Niu Zhisheng, et al. A Decentralized Framework for Dynamic Downlink Base Station Cooperation [C]// Global Telecommunications Conference (GLOBECOM). China, 2009: 1-6