

全局频繁闭项目集挖掘算法研究^{*})

陈健美 朱玉全 宋顺林 桂长青 宋余庆

(江苏大学计算机科学与通信工程学院 镇江 212013)

摘要 频繁闭项目集挖掘是数据挖掘研究中的一个重要研究课题。目前已有的频繁闭项目集挖掘算法主要针对单机环境,有关分布式环境下的全局频繁闭项目集挖掘算法的研究尚不多见。为此,本文提出了一种快速挖掘全局频繁闭项目集算法,并对其更新问题进行了研究;提出了一种相应的频繁闭项目集增量式更新算法,该算法将充分利用先前的挖掘结果来节省发现新的全局频繁闭项目集的时间开销。实验结果表明算法是有效的。

关键词 数据挖掘,分布式数据库,频繁闭项目集,全局频繁闭项目集

Research on Algorithm for Mining Global Closed Itemsets

CHEN Jian-Mei ZHU Yu-Qian SONG Shun-Lin GUI Chang-Qing SONG Yu-Qing

(School of Computer Science & Communication Engineering, Jiangsu University, Zhenjiang 212013)

Abstract Discovering frequent closed itemsets is a key problem in data mining application. Many sequential algorithms have been proposed for mining frequent closed itemsets. However, very little work has been done in discovering frequent closed itemsets in distributed environment. In this paper, an efficient algorithm GFCIA and its updating algorithm UGFCIA for mining global frequent closed itemsets is presented, which uses far less communication overhead. Experimental results show the feasibility and effectiveness of the algorithm.

Keywords Data mining, Distributed database, Frequent closed itemsets, Global frequent closed itemsets

发现频繁项目集是关联规则和序列模式等数据挖掘应用中的关键技术和步骤。目前,可用的频繁项目集挖掘算法主要有两类:频繁项目集挖掘算法^[1~5]和最大频繁项目集挖掘算法^[6~8]。前者具有候选项目集巨大、扫描事务数据库次数多等缺点。后者有一个明显的缺点,即最大频繁项目集并没有保存其子集的支持度。为了寻找非最大频繁项目集的支持度,需要执行一次额外的数据库扫描。另外,该类算法扫描数据库的次数并不一定比前者少。为此,许多学者提出了基于频繁模式树的频繁项目集和最大频繁项目集挖掘算法^[9,10]。我们在图像数据挖掘技术研究及应用中发现,在许多情况下(如数据量非常大、项目之间非常平衡等情况),该类算法并不一定能产生好的效果,而且其更新问题的复杂性已严重影响了它的实用价值。

频繁闭项目集提供了事务数据库的一个最小描述,其数量介于最大频繁项目集和频繁项目集之间,同时又记录了所有频繁项目集的支持度,因而发现频繁闭项目集对数据挖掘具有十分重要的意义,引起了国内外许多学者的极大关注,并已提出一些频繁闭项目集发现算法,如: A-close^[11]、CLOSE^[12]、CHARM^[13]、CLOSE+^[14]等,这些算法所处理的对象均为单机大型事务数据库系统。然而,在许多现实系统中,数据本身就是分布的,它们之间除了通过网络传递信息外,其他资源全部独立,因此迫切需要设计高效的算法来挖掘分布式事务数据库中的频繁闭项目集,而目前国内外尚无这方面的研究报道。为此,本文提出了一种全局频繁闭项目集挖掘及其更新算法,该算法可以有效地降低网络通信量,提高挖掘全局频繁闭项目集的效率。实验结果表明,本文所提出的算法是可行、有效的。

1 全局频繁闭项目集

设 $I = \{i_1, i_2, \dots, i_m\}$ 表示 m 个不同项目的一个集合,用

户给定的最小支持度阈值为 s 。在本文的研究中,假设分布式数据库是水平划分的,且各部分的数据库模式逻辑同构。设有 n 台基于无共享体系结构的计算机 $P^1, P^2, \dots, P^n$,即它们之间除了通过网络传递信息外,其他资源(如磁盘、内存等)全部独立。 $DB^i (i=1, 2, \dots, n)$ 是存储在计算机 P^i 硬盘上的分事务数据库,总的事务数据库 DB 为 $\bigcup_{i=1}^n DB^i$ 。设 D^i 和 D 分别表示 DB 和 DB^i 中的事务数,对某一项目集 X , $\text{Count}(X)$ 和 $\text{Count}^i(X)$ 分别表示 DB 和 DB^i 中包含 X 的事务数, $\text{Sup}(X)$ 和 $\text{Sup}^i(X)$ 分别表示 X 在 DB 和 DB^i 中的支持度,其中 $\text{Sup}(X) = \text{Count}(X)/D$, $\text{Sup}^i(X) = \text{Count}^i(X)/D^i, i=1, 2, \dots, n$ 。

定义 1 项目集 $X \subseteq I$, 事务集 $T \subseteq DB$, 定义映射关系:

$t: 2^I \rightarrow 2^T, t(X) = \{t \in T \mid \text{事务 } t \text{ 支持项目集 } X\}$ 。

$i: 2^T \rightarrow 2^I, i(T) = \{c \in I \mid T \text{ 中的任何事务均支持项目 } c\}$ 。

定义 2 设项目集 $X \subseteq I$, 如果满足 $i(t(X)) = X$, 且 X 为频繁项目集, 则称 X 为最大频繁闭项目集, 简称频繁闭项目集。

定义 3 设频繁项目集 $X \subseteq I$ 。对于 X 的任意真子集 Y , 均有 $\text{Sup}(X) < \text{Sup}(Y)$ 成立, 则称 X 为最小频繁闭项目集。

定义 4 若 $\text{Sup}(X) \geq s$, 则称 X 为全局频繁项目集。若 $\text{Sup}^i(X) \geq s$, 则称 X 为计算机 P^i 上的局部频繁项目集。

定义 5 若 $\text{Sup}(X) \geq s$, 且对于 X 的任意真子集 $Y \subset X$, 均有 $\text{Sup}(X) < \text{Sup}(Y)$, 则称 X 为全局最小频繁闭项目集。若 $\text{Sup}^i(X) \geq s$, 且对于 X 的任意真子集 $Y \subset X$, 均有 $\text{Sup}^i(X) < \text{Sup}^i(Y)$, 则称 X 为计算机 P^i 上的局部最小频繁闭项目集。

定义 6 若 $\text{Sup}(X) \geq s$, 且对于 X 的任意超集 $Y \supset X$, 均有 $\text{Sup}(X) > \text{Sup}(Y)$, 则称 X 为全局最大频繁闭项目集, 简称全局频繁闭项目集。若 $\text{Sup}^i(X) \geq s$, 且对于 X 的任意超集 Y

^{*}) 国家自然科学基金(No. 60572112)。陈健美 博士研究生, 主要研究领域为数据挖掘、医学图像处理。

$\sup^i(X) > \sup^i(Y)$, 则称 X 为计算机 P^i 上的局部最大频繁闭项目集, 简称局部频繁闭项目集。

全局频繁闭项目集挖掘的任务是: 在给定的事务数据库 $DB(DB = \bigcup_{i=1}^n DB^i)$, 其中 DB^i 是分布的) 中, 发现 DB 中所有的全局频繁闭项目集, 并考虑事务数据库不变, 最小支持度发生变化后全局频繁闭项目集的更新问题。

2 挖掘全局频繁闭项目集算法

定理 1 设项目集 $X, X_1, X_2, X_3 \subseteq I$, 事务集 $T, T_1, T_2, T_3 \subseteq DB$, 则

- (1) 如果 $X_1 \subseteq X_2$, 则 $t(X_1) \supseteq t(X_2)$ 。
- (2) 如果 $T_1 \subseteq T_2$, 则 $i(T_1) \supseteq i(T_2)$ 。
- (3) $i(t(X)) \supseteq X, t(i(T)) \supseteq T$ 。
- (4) $i(t(i(t(X)))) = i(t(X))$ 。

证明: (1) 设事务 $t \in t(X_2)$, 则 t 支持 X_2 。由于 $X_2 \supseteq X_1$, 因此 t 支持 X_1 , 即 $t \in t(X_1)$, $t(X_1) \supseteq t(X_2)$ 成立。

(2) 设项目 $c \in i(T_2)$, 则 T_2 中的各事务均支持 c , 由于 $T_2 \supseteq T_1$, 因此 T_1 中的各事务也支持 c , 即 $c \in i(T_1)$, $i(T_1) \supseteq i(T_2)$ 成立。

(3) 设项目 $c \in X$, 由于 $t(X)$ 中的各事务均支持 X , 因而 $t(X)$ 中的各事务均支持项目 c , 即 $c \in i(t(X))$, $i(t(X)) \supseteq X$ 成立。同样可以证明 $t(i(Y)) \supseteq Y$ 成立。

(4) 令 (3) 中的 X 为 $i(t(X))$, 则有 $i(t(i(t(X)))) \supseteq i(t(X))$ 成立。令 (3) 中的 T 为 $t(X)$, 则有 $t(i(t(X))) \supseteq t(X)$, 根据 (2), 有 $i(t(i(t(X)))) \subseteq i(t(X))$ 成立。因此, $i(t(i(t(X)))) = i(t(X))$ 。

综上, 定理 1 得证。

定理 2 设项目集 $X = \{x_1, x_2, \dots, x_k\}$ 为全局最小频繁闭 k -项目集, 则其任意 $(k-1)$ -项目子集均为全局最小频繁闭 $(k-1)$ -项目集。

证明: 反证法。假设存在项目集 X 的 $(k-1)$ -项目子集 $X_1 = \{x_1, x_2, \dots, x_{i-1}, x_{i+1}, \dots, x_k\}$ 为非全局最小频繁闭 $(k-1)$ -项目集, 则存在项目集 X_1 的 $(k-2)$ -项目子集 X_2 , 有 $\sup(X_2) = \sup(X_1)$ 成立, 即 $t(X_2) = t(X_1)$ 。令 $X_3 = \{x_i\} \cup X_2$, 则 $X_3 \subseteq X$, 由定理 1 可知 $t(X_3) \supseteq t(X)$ 。对于任意事务 $t \in t(X_3)$, t 支持项目集 X_3 或 $(\{x_i\} \cup X_2)$, t 必支持 $\{x_i\}$ 及 X_2 , 由于 $t(X_2) = t(X_1)$, 因此 t 支持项目集 X_1 , 又由于 $X = \{x_i\} \cup X_1$, 因此 t 支持项目集 X , $t \in t(X)$, 故 $t(X_3) \subseteq t(X)$, 从而有 $t(X_3) = t(X)$ 成立, 即 $\sup(X) = \sup(X_3)$ 。与项目集 X 为全局最小频繁闭 k -项目集矛盾。因此项目集 X 的所有 $(k-1)$ -项目子集均为全局最小频繁闭 $(k-1)$ -项目集。定理 2 得证。

推论 1 设项目集 X 为非全局最小频繁闭项目集, 则其任意超集均为非全局最小频繁闭 $(k-1)$ -项目集。

推论 2 设项目集 $X = \{x_1, x_2, \dots, x_k\}$ 为计算机 P^i 上的局部最小频繁闭 k -项目集, 则其任意 $(k-1)$ -项目子集均为计算机 P^i 上的局部最小频繁闭 $(k-1)$ -项目集。

推论 3 设项目集 X 为计算机 P^i 上的非局部最小频繁闭项目集, 则其任意超集均为计算机 P^i 上的非局部最小频繁闭项目集。

定理 3 设 $\text{Maxcount}_i(X) = \min\{\text{Count}^i(Y) \mid Y \subseteq X \text{ and } |Y| = |X| - 1\}$, 其中 $|X|$ 表示项目集 X 中的项目数, 则 $\text{Count}^i(X) \leq \text{Maxcount}_i(X)$ 。

证明: 因为 $Y \subseteq X$, 所以 $\text{Count}^i(X) \leq \text{Count}^i(Y)$, 由于 Y 的任意性, 因此 $\text{Count}^i(X) \leq \{\text{Count}^i(Y) \mid Y \subseteq X \text{ and } |Y| =$

$|X| - 1\}$, 即 $\text{Count}^i(X) \leq \text{Maxcount}_i(X)$ 。定理 3 得证。

定理 4 设 $\text{Maxcount}_{j \neq i}(X) = \min\{\text{Count}(Y) - \text{Count}^i(Y) \mid Y \subseteq X \text{ and } |Y| = |X| - 1\}$, 则 $\text{Maxcount}_{j \neq i}(X) \geq \sum_{j=1, j \neq i}^n \text{Maxcount}_j(X)$ 。

证明: 设 $Y_1 \subseteq X$, 且 $|Y_1| = |X| - 1$, 满足 $\text{Maxcount}_{j \neq i}(X) = \min\{\text{Count}(Y) - \text{Count}^i(Y) \mid Y \subseteq X \text{ and } |Y| = |X| - 1\}$, 即 $\text{Maxcount}_{j \neq i}(X) = \text{Count}(Y_1) - \text{Count}^i(Y_1) = \sum_{j=1, j \neq i}^n \text{Maxcount}_j(Y_1)$ 。由于 $\text{Maxcount}_j(X) = \min\{\text{Count}^j(Y) \mid Y \subseteq X \text{ and } |Y| = |X| - 1\}$, 因此 $\text{Count}(Y_1) \geq \text{Maxcount}_j(X)$, 故 $\text{Maxcount}_{j \neq i}(X) = \sum_{j=1, j \neq i}^n \text{Count}^j(Y_1) \geq \sum_{j=1, j \neq i}^n \text{Maxcount}_j(X)$ 。定理 4 得证。

定理 5 对于任意项目集 X , 必有 $\text{Count}(X) \leq \text{Count}^i(X) + \text{Maxcount}_{j \neq i}(X)$ 成立。

证明: $\text{Count}(X) = \sum_{j=1}^n \text{Count}^j(X) = \text{Count}^i(X) + \sum_{j=1, j \neq i}^n \text{Count}^j(X) \leq \text{Count}^i(X) + \sum_{j=1, j \neq i}^n \text{Maxcount}_j(X) \leq \text{Count}^i(X) + \text{Maxcount}_{j \neq i}(X)$ 。定理 5 得证。

定理 6 设 GMinFCI 为全局最小频繁闭项目集的集合。项目集 $Y \in \text{GMinFCI}$, $t(Y) = \{T_{Y_1}, T_{Y_2}, \dots, T_{Y_m}\}$, 则 $i(t(Y)) = \bigcap_{j=1}^m T_{Y_j}$ 。

证明: 根据映射关系 i 和 t 的定义, 有 $T_{Y_j} \supseteq i(t(Y))$ ($j = 1, 2, \dots, m$) 成立, 因此 $i(t(Y)) \subseteq \bigcap_{j=1}^m T_{Y_j}$ 。另外, 如果项目 $c \in \bigcap_{j=1}^m T_{Y_j}$, 则 T_{Y_j} ($j = 1, 2, \dots, m$) 支持项目 c , 因此 $c \in i(t(Y))$, 因而 $\bigcap_{j=1}^m T_{Y_j} \subseteq i(t(Y))$ 。故 $i(t(Y)) = \bigcap_{j=1}^m T_{Y_j}$ 。定理 6 得证。

综上所述, 我们可以得到全局频繁闭项目集挖掘算法, 具体描述见算法 1。

算法 1 全局频繁闭项目集挖掘算法 GFCIA (Global Frequent Closed Itemsets Algorithm)。

输入: 存储在计算机 P^i 中的事务数据库 DB^i ($i = 1, 2, \dots, n$); 最小支持度阈值 s 。

输出: 全局最小频繁闭项目集, 全局频繁闭项目集。

方法: 在计算机 P^i 上执行如下操作。

- (1) $\text{GL}_{c1} = \{\text{frequent closed 1-itemsets}\}$; // GL_{c1} 为全局最小频繁闭 1-项目集的集合
- (2) for ($k = 2$; $\text{GL}_{ck} \neq \Phi$; $k++$) do begin
- (3) $\text{CG}_{ck} = \text{A priori-gen}(\text{GL}_{c(k-1)})$; // CG_{ck} 为候选全局最小频繁闭 k -项目集的集合
- (4) for each transaction $t \in DB^i$ do begin
- (5) $C_t = \text{subset}(\text{CG}_{ck}, t)$;
- (6) for each candidate $c \in C_t$ do
- (7) $\text{Count}^i(c)++$;
- (8) end;
- (9) for each $c \in \text{CG}_{ck}$ do begin
- (10) calculate $\text{Maxcount}_{j \neq i}(c)$;
- (11) if ($\text{Count}^i(c) + \text{Maxcount}_{j \neq i}(c) < sD$) then
- (12) delete c from CG_{ck} ; // 定理 5
- (13) end;
- (14) broadcast CG_{ck} 中各项目集的支持数;
- (15) receive from 计算机 P^i 发来的 CG_{ck} 中各项目集的支持数 ($j \neq i$);
- (16) $\text{GL}_{ck} = \{c \in \text{CG}_{ck} \mid \sup(c) \geq s\}$;
- (17) for each $c \in \text{GL}_{ck}$ do begin
- (18) for each $(k-1)$ -subset s of c do
- (19) if ($\sup(s) = \sup(c)$) then
- (20) $\text{GL}_{ck} = \text{GL}_{ck} - \{c\}$; // 删除 GL_{ck} 中的非全局最小频繁闭项目集
- (21) end;
- (22) end;
- (23) $\text{GMinFCI} = Y \cup \text{GL}_{ck}$;
- /* 根据全局最小频繁闭项目集 GMinFCI 确定全局频繁闭项目集 GMaxFCI */
- (24) for each $Y \in \text{GMinFCI}$ do

(25) $M_Y = I$; // M_Y 用来存放全局最小频繁闭项目集 Y 所对应的全局最大频繁闭项目集, 初始为全集 I
 (26) for all transaction $t \in DB$ do
 (27) for all $Y \in GMinFCI$ do
 (28) if $t \supseteq Y$ then
 (29) $M_Y = M_Y \cap t$; // 根据定理 6
 (30) broadcast 所有 M_Y ;
 (31) receive from 计算机 P_i 发来的所有 $M_Y (j \neq i)$;
 (30) $GMaxFCI = \Phi$; // 赋初值
 (31) for all $Y \in GMinFCI$ do
 (32) $GMaxFCI_{UD} = GMaxFCI_{UD} \cup \bigcap_{i=1}^n M_Y$;

3 算法实现与比较

为了测试算法的性能, 我们在 100M 的局域网中, 用两台 PC 机构成分布式系统, 两台 PC 机的配置均相同, 实验数据采用文[2]的合成数据方法, 并将事务数据库均分到两台计算机上, 其有关参数与文[2]相同。在本实验中, $N=1000$, $|L|=1000$, $|D_1|=10K$, $|T|=25$, $|I|=15$, 实验结果如图 1 所示。图 1 表明算法 GFCIA 是有效可行的, 其效率是全局频繁项目集挖掘算法 FPM 的 2~4 倍。

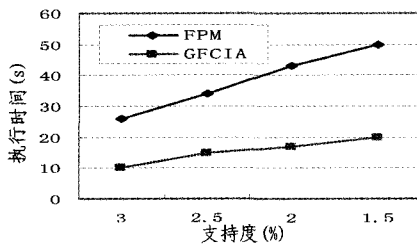


图 1 算法执行时间

结束语 本文提出了一种全局频繁闭项目集挖掘算法, 并就其更新问题进行了研究, 提出了一种全局频繁闭项目集更新算法, 该算法可以有效地降低网络通信量, 提高全局频繁闭项目集挖掘算法的效率。实验表明, 本文所提算法是有效和可行的。下一步研究的主要目标是, 在实际应用中对算法进行进一步的测试, 并就其它更新问题进行深入研究, 以推广其应用。

参考文献

- Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large databases. In: Proc. ACM SIGMOD Int Conf. Management of Data, Washington D C, 1993. 207~216
- Hughes B, Cahill V. Achieving Real-time Guarantees in Mobile Ad Hoc Wireless Networks. In: Proceedings of the Work-in-Progress session of the 24th IEEE Real-time Systems Symposium, December 2003
- Al-karaki J N, Kamal A E. Routing Techniques in Wireless Sensor Networks: A Survey. Wireless Communications, IEEE [see also IEEE Personal Communications], 2004, 11(6): 6~28
- He Tian, Stankovic J A, Lu Chenyang, et al. SPEED: A Stateless Protocol for Real-time Communication in Sensor Networks. In: Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS 2003), May 2003. 46~55
- Sohrabi K, Gao J, Ailawadhi V, et al. Protocols for Self-organization of a Wireless Sensor Network. In: IEEE Personal Communications, October 2000. 16~27
- Giordano S, Stojmenovic I, Blazevic L. Position Based Routing Algorithms for Ad Hoc Networks: A Taxonomy. Ad Hoc Wireless Networks, 2003
- Finn G G. Routing and Addressing Problem in Large Metropolitan-Scale Inter-networks. ISI res Rep ISU/RF-87-180. March 1987
- Kranakis E, Singh H, Urrutia J. Compass Routing on Geometric Networks. In: Proceedings of the 11th Canadian Conference on Computational Geometry, Vancouver, Canada, August 1999. 51~54
- Berkhovskikh L, Lysanov Y. Fundamentals of Ocean Acoustics. New York: Springer, 1982
- Johnson D B, Maltz D A. Dynamic source routing in ad hoc wireless networks. In: Mobile Computing, edited by Tomasz Imielinski and Hank Korth, Kluwer Academic Publishers, 1996. 153~181
- Agrawal R, Srikant R. Fast algorithm for mining association rules. In: Proc. 20th Int. Conf. on VLDB, Santiago, Chile, 1994. 487~499
- Houtsma M, Swami A. Set-oriented mining for association rules in relational databases. In: Yu P, Chen A, eds. Proc. of the Int. Conf. on Data Engineering, Los Alamitos, CA: IEEE Computer Society Press, 1995. 25~33
- Han J, Fu Y. Discovery of multiple-level association rules from large databases. In: Proc. 21st Int. Conf. on VLDB, Zurich, Switzerland, 1995. 420~431
- Han J, Kamber M. Data Mining: Concepts and Techniques. Beijing: High Education Press, 2001
- Bayardo R. Efficiently mining long patterns from databases. In: Haas L M, Tiwary A, eds. In: Proc. of the ACM SIGMOD Int Conf. on Management of Data. New York: ACM Press, 1998. 85~93
- Lin Dao-I, Kedem Z M. Pincer-Search: a new algorithm for discovering the maximum frequent set. In: Schek H J, Saltor F, Ramos I. et al., eds. Proc. of the 6th European Conf on Extending database Technology. Heidelberg: Springer_Verlag, 1998. 105~119
- 路松峰, 卢正鼎. 快速开采最大频繁项目集. 软件学报, 2001, 12(2): 293~297
- Han J, Jian Pei, Yiwen Y. Mining Frequent Patterns Without Candidate Generation. In: Proceedings 2000 ACM-SIGMOD Int. Conf. Management of Data, Dallas. TX, 2000. 1~12
- 宋余庆, 朱玉全, 孙志挥, 等. 一种基于 FP-tree 的最大频繁项目集挖掘及其更新算法. 软件学报, 2003, 14(9): 1586~1592
- Pasquier N, Bastide Y, Taouil R, et al. Discovering Frequent Closed Itemsets for Association Rules. In: Proc. of 7th Int. Conf. Database Theory, Jerusalem Israel, 1999, 1: 398~416
- Pei J, Han J, Mao R. CLOSET: An efficient algorithm for mining frequent closed itemsets. In: DMKD'00, May 2000
- Zaki Z, Hsiao C. CHARM: An efficient algorithm for closed itemset mining. In: SDM'02, April 2002
- Wang J Y, Han J, Pei J. CLOSET+: Searching for the best strategies for mining frequent closed itemsets. In: SIGKDD'03, August 24-24, Washington, DC, USA
- Cheung D W, Lee S D, Xiao Y Q. Effect of data skewness and workload balance in parallel data mining. IEEE Trans on Knowledge and Data Engineering, 2002, 14(3): 498~514

(上接第 41 页)

- Hughes B, Cahill V. Achieving Real-time Guarantees in Mobile Ad Hoc Wireless Networks. In: Proceedings of the Work-in-Progress session of the 24th IEEE Real-time Systems Symposium, December 2003
- Al-karaki J N, Kamal A E. Routing Techniques in Wireless Sensor Networks: A Survey. Wireless Communications, IEEE [see also IEEE Personal Communications], 2004, 11(6): 6~28
- He Tian, Stankovic J A, Lu Chenyang, et al. SPEED: A Stateless Protocol for Real-time Communication in Sensor Networks. In: Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS 2003), May 2003. 46~55
- Sohrabi K, Gao J, Ailawadhi V, et al. Protocols for Self-organization of a Wireless Sensor Network. In: IEEE Personal Communications, October 2000. 16~27
- Giordano S, Stojmenovic I, Blazevic L. Position Based Routing