

基于内容的快速事件匹配算法

陈继明^{1,2}, 鞠时光¹, 潘金贵², 邹志文¹, 龚震宇²

(1. 江苏大学 计算机科学技术与通信工程学院, 江苏 镇江 212013;

2. 南京大学 计算机软件新技术国家重点实验室, 江苏 南京 210093)

摘要: 在对分布式虚拟环境应用特点和已有事件匹配算法进行研究和分析的基础上, 设计并实现了一种基于内容的快速事件匹配算法。该算法兼顾了处理订购和事件匹配的效率, 能够有效地解决对称事件匹配问题, 从而使基于内容的发布/订购模型能较好地应用于分布式虚拟环境。实验结果及性能分析证明了该算法的可行性和有效性。

关键词: 分布式虚拟环境; 基于内容的发布/订购模型; 事件匹配效率; 基于内容的快速事件匹配算法

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2011)06-0078-08

Content-based effective event matching algorithm

CHEN Ji-ming^{1,2}, JU Shi-guang¹, PAN Jin-gui², ZOU Zhi-wen¹, GONG Zhen-yu²

(1. School of Computer Science and Telecommunications Engineering, Jiangsu University, Zhenjiang 212013, China;

2. State Key Lab for Novel Software Technology, Nanjing University, Nanjing 210093, China)

Abstract: Based on the study of characteristics of the distributed virtual environment and current event matching algorithms, a content-based effective event matching (CEEM) algorithm was proposed. By considering the efficiency of both subscription processing and event matching, this algorithm can solve the symmetric event matching efficiently, so as to make the content-based publish/subscribe model fit the virtual environment better. Finally, experiments and performance analysis show that CEEM is both practical and effective.

Key words: distributed virtual environment; content-based publish/subscribe model; efficiency of event matching; CEEM

1 引言

分布式虚拟环境 (DVE, distributed virtual environment) 将虚拟现实与网络通信技术相结合, 提供一个共享虚拟空间以支持对象的仿真以及多个分布用户的交互和通信。在 DVE 系统中, 由于整个虚拟环境的信息量非常大, 目前采用的 IP 多播或者基于 C/S 混和结构的通信模型^[1-3]已经不能完全满足系统的通信需求。因此, DVE 系统对其通信模型提出了更高的要求, 以满足虚拟环境可动态扩

展, 低通信开销以及实时性等需求。

基于内容的发布/订购 (content-based publish/subscribe) 通信模型中订购者进行订购, 而发布者发布事件, 只要事件的内容和订购者的订购相匹配, 则该事件就被分发给订购者。由于其具有异步通信、动态性、多对多和丰富的语义性等优点, 目前被广泛的用于股票市场的数据传送, P2P 共享网络搜索等内容相关且通信关系频繁变化的分布式应用中^[4,5]。Oliveira 首先将该模型应用于 DVE, 提出了一种兴趣管理的思想, 即通过信息订购的方

收稿日期: 2010-07-12; 修回日期: 2011-05-10

基金项目: 国家自然科学基金资助项目 (60773049, 60533080); 江苏省自然科学基金资助项目 (BK2010373)

Foundation Items: The National Natural Science Foundation of China (60773049, 60533080); The Natural Science Foundation of Jiangsu Province (BK2010373)

式使得对象只与其相关的对象进行通信，而与其不相关的对象和数据则被过滤^[6]。在此基础上，文献[7,8]进一步将主动路由技术应用到兴趣管理中，实现数据转发和兴趣管理相结合，在主动路由器上根据用户的订购兴趣来主动过滤和转发 DVE 中对象发布的数据，有效地降低系统的整体通信量，提高系统的可扩展性，进一步满足系统中大量对象之间多对多的动态通信要求。

在基于发布/订购模型的 DVE 应用中，虚拟对象通过具有较强表达能力谓词来描述自身的发布和订购信息，系统中的每个路由节点上则将对象发布信息与订购信息进行匹配，并根据匹配结果实现数据转发。当系统中存在大量的订购和发布事件时，每个路由节点上将面临着事件匹配运算带来的巨大压力。因此，路由节点上的事件匹配效率，不仅决定了系统的实时性是否能得到满足，而且会制约着整个系统的可扩展性，可能造成系统拥塞和网络瓶颈，严重影响系统的实用性。

鉴于事件匹配算法的重要性，本文在对虚拟环境的特点和已有事件匹配算法进行讨论的基础上，提出一种适合分布式虚拟环境的快速事件匹配算法 (CEEM, content-based effective event matching algorithm)，并通过实验和性能分析证实了该算法的可行性和有效性。

2 相关工作

现有的事件匹配算法研究都是从减少额外的谓词匹配测试方面进行优化处理，主要的算法一般分为两类。

基于谓词索引 (predicate indexing based) 算法，如计数算法 (counting algorithm)^[9]；汉森算法 (Hanson algorithm) 及其扩展算法^[10-12]等，其主要思想都是对不同的订购谓词分别进行索引及计数，并根据其结果是否满足要求进行匹配过滤。这类算法通过采用较为高效的测试先淘汰相当部分的订购，然后在剩余的订购中进行比较运算，以提高算法的匹配效率。

基于匹配网络 (testing network based) 算法，如基于匹配树 (testing tree based) 的算法^[13, 14]和基于二叉决策图 (BDD, binary decisions diagrams) 的算法^[15]等，在预处理阶段将订购组织成图 (匹配网络) 的结构，事件从图的入口进入，经过各个中间节点的过滤，直到出口判定匹配结果。这类算法

能够充分利用各个订购谓词间的相关性，从而能够实现较高的匹配效率。在此研究的基础上，Silvia 等人提出了基于分布式 R-trees 的匹配路由算法^[16]，根据 MBR^[17]原理构建一棵分布式 R-trees 来实现事件的匹配路由，然而该算法的不足之处是空间复杂度较大，路由信息维护困难。另外，根据基于内容的发布/订购系统中订购覆盖路由的思想，文献[18]提出了一种基于谓词关系二叉树的事件匹配路由技术，有效地降低了路由器上订购信息的存储规模。

由于各种应用的不同特点，对基于内容的发布/订购模型采用的事件匹配算法的要求也不尽相同，但上述两类算法在 DVE 应用中有着较为明显的不足。基于决策网络的事件匹配算法需要的构建匹配树或者二叉决策图，或者有很高的空间要求，难以支持系统的扩展，或者难以进行动态更新，只适用于静态的事件模型，无法适应 DVE 环境中发布和订购信息频繁变动的要求。基于谓词索引的算法，无法利用不同属性之间的相关性，也不能实现较高的事件匹配效率，无法满足 DVE 应用中大量事件实时传输的要求。因此，本文将在现有的两类算法基础上，研究一种基于内容的快速事件匹配算法，不但能够提高事件匹配的效率，而且解决了事件匹配的对称性问题。

3 基于内容的事件匹配算法

3.1 相关定义

在 DVE 应用中，事件匹配过程是将描述虚拟对象的特征及其状态的事件发布信息和描述虚拟对象兴趣范围的订购信息进行逻辑判定并转发的过程。由于虚拟环境中对象订购和发布的信息较为复杂 (不仅包括对象的位置信息以及自身特点，还包括对象的预感区域及兴趣区域等)，通常由多个谓词采用逻辑与的关系组成，订购与发布信息间的匹配其本质上是谓词间的一种匹配运算。因此，我们有必要针对 DVE 应用中的事件匹配做出以下定义。

定义 1 令 $p=A\Theta y$ 为谓词，而 y 是属性 A 值域空间中的一个合理取值。其中， Θ 是比较操作， $p[v]$ 的值可以使用 Θ 比较 v 和 y 所确定。

定义 2 针对谓词 p ，分别定义取属性函数 $Attribute(p)$ ，取比较操作函数 $Operator(p)$ 和取阈值函数 $Threshold(p)$ ：若 $p=A\Theta y$ ，则 $Attribute(p)=A$ ，

$Operator(p)=\Theta, Threshold(p)=y$ 。

定义 3 如果谓词 p 有 $Operator(p)、“=”$ ，则该谓词 p 称为等值谓词；相似地，如果 $Operator(p)、“<”$ ，该谓词 p 称为上阈值谓词；如果 $Operator(p)、“>”$ ，该谓词 p 称为下阈值谓词。

定义 4 若 S 为 m 个谓词的合取式，记做 $S=\bigwedge_{i=1}^m p_i$ ， E 为 n 个等值谓词的合取式，记做 $E=\bigwedge_{j=1}^n p'_j$ ，那么如果有 $\forall i \forall j (Operator(p'_j) = “=” \wedge Attribute(p_i) = Attribute(p'_j) \rightarrow p_i[Threshold(p'_j)])$ ，则称 S 和 E 匹配，记做 $Match(S, E)$ 。

定义 5 若 S 为 m 个谓词的合取式，记做 $S=\bigwedge_{i=1}^m p_i$ ， T 为 n 个谓词的合取式，记做 $T=\bigwedge_{j=1}^n p'_j$ ，若 PN 为所有非空等值谓词合取式的集合，那么 $\exists E, E \in PN (Match(S, E) \wedge Match(T, E))$ ，则称 S 和 T 匹配，记做 $Match(S, T)$ 。

3.2 算法的数据结构

数据结构是 CEEM 算法的重点，也是算法优化的关键（如图 1 所示）。

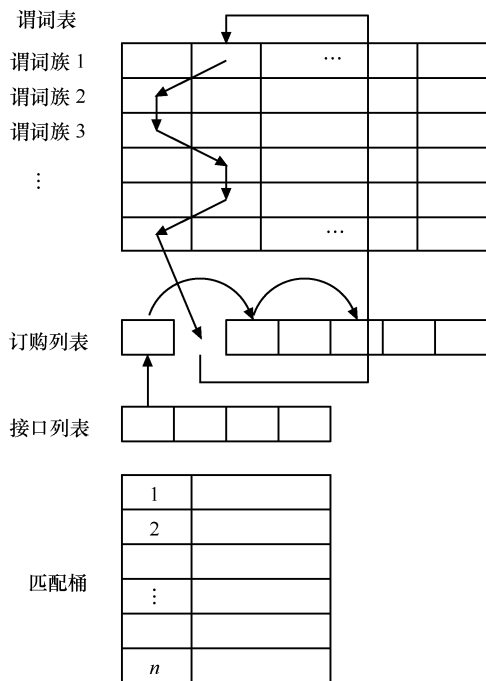


图 1 CEEM 算法主要数据结构

谓词表 (predict table): 谓词表是数据结构的核⼼部分。谓词分为等值谓词、上阈值谓词和下阈值谓词 3 类，具有相同属性和相同比较操作的谓词归入一个谓词族。每个谓词族对应谓词表的一行，并将阈值按序排列，即上阈值谓词族中阈值降序排

列，下阈值谓词族中阈值升序排列，而等值谓词族则无论何种顺序都可以。此外，谓词表中将同一订购中的谓词用指针链接起来，最后都指向订购列表的具体订购。

匹配桶 (match bucket): 用于辅助谓词表进行匹配。由于对谓词表的匹配是逐行进行的，但不是每个订购都涉及所有的谓词族，有的指针可能会跨越某些行，为了使得这样的谓词在逐行扫描中不会丢失，采用匹配桶的方式来暂时存储这些被延缓的谓词的索引。

接口列表 (interface list) 和订购列表 (subscription list): 接口列表和订购列表维护了接口和接口对应的订购信息。订购列表登记了所有加入谓词表的订购，同时记录订购在谓词表中第一个谓词的指针，以便于访问订购中的所有谓词。接口代表了事件的传送方向，是事件匹配的最终目的。每个接口仅对应一个连接（订购者或事件路由节点）的订购信息，通过接口列表和订购列表将所有的订购组织在一起。

3.3 事件的订购处理

在初始化时由于尚没有订购，谓词表和订购列表均为空，接口列表也只是对应当前接口。当接收到新的订购后，需要对谓词表、接口列表和订购列表分别进行更新，主要可分以下 3 步：

step1 将订购加入到订购列表中，即将相应接口的最后一个订购在订购列表中的最后一个条目的后续指向新加入的订购。

step2 在谓词表中登记订购，按照从上往下的顺序，在订购所包含的每个谓词族中通过二分法查找相应位置，并将对应的阈值插入表中；在谓词表插入的过程中，将插入的谓词按从上至下的顺序逐一进行链接，最后指向订购列表中的对应位置。

step3 在订购列表的对应表项中记录订购在谓词表中第一个位置的指针。

3.4 事件的匹配

令 P_i 为第 i 行谓词族的所有谓词的集合， P_{i_match} 表示 P_i 中匹配该事件的谓词集合， S_{match} 表示匹配的订购集合，CEEM 算法事件匹配的主要过程步骤如下。

step1 对匹配桶内容进行初始化。遍历订购列表，找到每个订购中第一个谓词的位置，即如果一个订购的第一个谓词在谓词表中第 i 行的第 j 列的

位置，那么在匹配桶的第 i 行中加入这个位置索引 j 。初始化后，匹配桶中保存了所有订购的第一个谓词的索引。

step2 对谓词表进行逐行扫描。对于谓词表的每一行，主要进行以下处理：首先查找谓词族中匹配该事件的所有谓词，然后根据匹配桶的内容对这些谓词逐个进行考察，将谓词表中的所有谓词分成匹配和不匹配两部分。

在对第 i 行进行测试时，对于非等值谓词族，使用二分法查找已排序的谓词，根据事件中对应的属性值，找到一个位置将 P_i 分成 P_{i_left} 和 P_{i_right} 两部分， P_{i_left} 就对应这一行的 P_{i_match} ，表示该行的谓词中匹配该事件的集合。对于等值谓词族，则查找所有谓词值等于事件对应的属性值的谓词，放入 P_{i_match} 。如果该事件不包含此行对应的属性，则认为这一行所有的谓词都是匹配的， P_{i_match} 包含该行的所有谓词。

在逐行测试中，设在第 i 行进行测试时，匹配桶中第 i 行对应的谓词集合为 B_i ，则该行中所有真正满足匹配测试的谓词集合为 $P_{i_match} \cap B_i$ ；将该集合中是订购最后一个节点的谓词对应的订购加入 S_{match} ，其余的则将其后继谓词索引填入匹配桶中对应位置，并继续上述过程，直至扫描完所有谓词族。

step3 逐行进行测试后， S_{match} 中就包含了所有匹配当前测试事件的订购，并根据这些订购确定下一步的转发动作。

值得一提的是，在 DVE 应用中，如果遵循 HLA 规范发布者拥有发布区域，订购者拥有订购区域的语义，那么必然要求事件中的属性区间同订购中的属性区间的匹配。因此，DVE 应用中的事件匹配具有对称性，而上述匹配过程很难解决事件匹配的对称性问题，因此给出下列定义和定理。

定义 6 令 Op 为比较操作的集合，即 $Op = \{“=”, “>”, “<”\}$ 。可以针对比较操作定义求反函数 $Opposite: Op \rightarrow Op$ ，使得 $Opposite(“=”) = “=”$ ； $Opposite(“>”) = “<”$ ； $Opposite(“<”) = “>”$ 。

定理 1 若 S 为 m 个谓词的合取式，记做 $S = \bigwedge_{i=1}^m p_i$ ， T 为 n 个谓词的合取式，记做 $T = \bigwedge_{j=1}^n p'_j$ ，若 $\forall i, j (Attribute(p_i) = Attribute(p'_j) \wedge Operator(p_i) = Opposite(Operator(p'_j)) \rightarrow p_i[Threshold(p'_j)])$ 成立，那么 S 和 T 匹配， $Match(S, T)$ 成立。

证明 首先构造如下结构的等值谓词合取式

E ，记做 $E = \bigwedge_{j=1}^n p''_j$ ，有 $Attribute(p''_j) = Attribute(p'_j)$ ：

$$p''_j = \begin{cases} \min(Threshold(p_i), Threshold(p'_j)) , \\ \exists p_i, Attribute(p_i) = Attribute(p'_j) \wedge \\ Operator(p_i) = Operator(p'_j) = “<” \\ \max(Threshold(p_i), Threshold(p'_j)) , \\ \exists p_i, Attribute(p_i) = Attribute(p'_j) \wedge \\ Operator(p_i) = Operator(p'_j) = “>” \\ p'_j , 其他 \end{cases}$$

利用已知条件：

$$\forall i, j (Attribute(p_i) = Attribute(p'_j) \wedge Operator(p_i) = Opposite(Operator(p'_j)) \rightarrow p_i[Threshold(p'_j)])$$

根据定义4，可以证明得到 $Match(S, E)$ 与 $Match(T, E)$ ，即 E 与 S 匹配且 E 与 T 匹配。再根据定义 5，即 $\exists E, E \in PN(Match(S, E) \wedge Match(T, E))$ ，可得 S 与 T 匹配即 $Match(S, T)$ 。命题得证。

在 CEEM 算法对谓词表中逐行扫描过程中，对于等值谓词，只要属性值相等就匹配，不相等就不匹配；而对于上阈值谓词和下阈值谓词，由于一般都成对出现以确定一个属性的范围，则可采用交叉的谓词匹配方法，即将订购的上阈值谓词与发布的下阈值谓词的阈值比较，订购的下阈值谓词则与发布的上阈值谓词的阈值比较，这样的交叉匹配方法的正确性由定理 1 所保证。CEEM 算法的伪码如图 2 所示，其中输入参数为谓词表 $table$ 和待匹配事件 $event$ 。

```

List SubscriptionList; //订购列表
procedure Match(table[], event[]) {
    S_match=Φ;
    Bucket=Φ;
    InitBucket(Bucket, SubscriptionList); //初始化匹配桶
    for i=0 to Bucket.length {
        for each predicate index pi in Bucket[i] {
            if operator(table[i]) is ">" or "<" {
                divisionPosition=Segmentation(table[i], event[i]);
                //根据事件 event 对应 i 谓词族的属性值，找到一个位置将谓词表第 i 行分成两部分
                if pi<=divisionPosition {
                    if pi is the last predicate of a subscription s in SubscriptionList add s to S_match;
                    else add the next predicate of pi to Bucket;
                }
            } else if pi match event[i] { //等值谓词匹配
                if pi is the last predicate of a subscription s add s to S_match;
                else add the next predicate of pi to Bucket;
            }
        }
    }
}
    
```

图 2 CEEM 匹配算法伪码

CEEM 匹配算法在性能上的优势主要体现在以下 2 个方面: 1) 动态性, 即该算法不需要像现有的匹配算法, 如基于匹配网络的算法需构造静态的匹配网络, 可以实现动态的更新订购信息; 2) 高效性, 即该算法采用以谓词表为核心的数据结构, 利用谓词间的关联性和属性间的相关性来加快匹配速度; 同时对谓词族进行了划分, 并对每个谓词族用排序或者散列表的方式进行了索引, 使得在对每个谓词进行匹配时能获得很高的效率。

3.5 算法性能分析

假设事件模型所支持的属性数为 A , 处理的总订购数是 S 。

空间复杂度: 算法的空间复杂度主要是谓词表的空间需求。每条订购最多由 A 个谓词组成, 在谓词表的谓词数是 SA , 即总的空间复杂度是 $O(AS)$ 。

事件订购时间复杂度: 设第 i 行的谓词数 P_i , 插入一个谓词的时间复杂度是 $O(P_i + \text{lb}P_i)$ ($1 \leq i \leq A, 1 \leq P_i \leq S$); 插入完整的一条订购的时间复杂度是 $O(A^2(P_i + \text{lb}P_i))$ ($1 \leq i \leq A, 1 \leq P_i \leq S$), 最坏的情况是每一行都有 S 个谓词, 时间复杂度则是 $O(A^2(S + \text{lb}S))$ 。

事件匹配时间复杂度: 事件匹配算法运行时间为 $\sum_i [\text{lb}(P_i + 1)C_s + P_i C_i] \approx O(S)$, 其中, C_s 是二分查找的时间花费, C_i 是谓词填入匹配桶的花费。

4 实验结果及分析

在基于发布/订购模型的 DVE 系统中, 事件传递的目标是实时性和准确性, 即要求用最短的时间将事件传给正确的接收者。由于在每个中间节点都需要对事件进行匹配后传递, 经过多个节点的累积, 将带来很大的延迟, 提高事件匹配效率是保证事件传递实时性的重要条件。因此, 下面将从事件传递的正确性和匹配效率 2 个方面对 CEEM 算法进行分析和验证。

4.1 正确性

将通过定理 2 及其证明来简要分析 CEEM 算法的正确性。

定理 2 系统在保证订购语义正确情况下, CEEM 算法能保证事件传递的正确性。

证明 假设系统中发布方 HP 发送事件 T , 订购方 HS 的订购为 S 。下面将从 2 个方面来进行论证。

1) 证明如果发送事件 T 能够满足能够订购 S ,

则 T 能到达 HS 。

因为发布方 HP 和订购方 HS 属于同一个系统, 那么 HP 与 HS 之间至少存在一条路径, 记为 $Path(HP, HS) = AR_1, \dots, AR_i, \dots, AR_n$, 其中, AR_i 为系统中事件路由节点。

对于该路径上的任意一个 $AR_i (1 \leq i \leq n)$, 由于系统保证订购语义的正确性, 则有 $S \subseteq Sub(AR_i)$, 其中, $Sub(AR_i)$ 表示 AR_i 包含的订购信息。设 $S = \bigwedge_{k=1}^m p_k$ 和 $Sub(AR_i) = \bigwedge_{j=1}^n p_j^i$, 根据 $S \subseteq Sub(AR_i)$ 可以得到:

$$\begin{aligned} & \forall k \forall j (Attribute(p_k) \\ & = Attribute(p_j^i) \rightarrow p_j^i [Threshold(p_k)]) \end{aligned} \quad (1)$$

又因为发送事件 T 能够满足能够订购 S , 即 $Match(T, S)$ 成立, 由定义 5 可知必定 $\exists E, E \in PN(Match(S, E) \wedge Match(T, E))$ 。设 $E = \bigwedge_{q=1}^l p'_q$, 根据定义 4 可得:

$$\begin{aligned} & \forall k \forall q (Operator(p'_q) = "=" \wedge Attribute(p_k) \\ & = Attribute(p'_q) \rightarrow p'_q [Threshold(p'_q)]) \end{aligned} \quad (2)$$

由式(1)、式(2)可得到:

$$\begin{aligned} & \forall j \forall q (Operator(p'_q) = "=" \wedge Attribute(p_j^i) \\ & = Attribute(p'_q) \rightarrow p_j^i [Threshold(p'_q)]) \end{aligned} \quad (3)$$

根据式(3), 由定义 4 可得到 $Match(E, Sub(AR_i))$ 成立, 再根据定义 5 可得 $Match(T, Sub(AR_i))$, 即路由节点 AR_i 的订购与事件 T 相匹配。

因此, 在事件 T 向订购方 HS 发送的过程中, 在路径 $Path(HP, HS)$ 上的每个路由节点都能保证其订购都与事件 T 匹配并转发, 直到事件 T 达到 HS 。

2) 证明如果发送事件 T 不满足订购 S , 则 T 不能到达 HS 。

(反证法) 假设 T 能够到达 HS 。由定理 1 可以得到, 对于任何与 HS 直接相连的路由节点的接口 VIF_{hs} (其订购信息记为 $Sub(VIF_{hs})$), 均有 $Match(T, Sub(VIF_{hs}))$ 成立。由于 CEEM 算法采用的接口列表保证接口和订购方 HS 的订购信息为一一对应关系, 则有 $Sub(VIF_{hs}) = S$ 。从而可以得到 $Match(T, S)$ 成立, 即事件 T 能够满足订购 S , 与条件相矛盾。故假设不成立, 即 T 不能达到 HS 。

综合 1) 和 2) 所述, 命题得证。

4.2 算法匹配效率

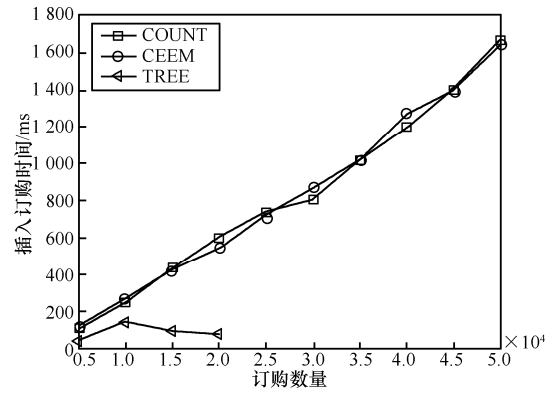
汉森算法及其扩展算法的效率很大程度上取决于优先进行的测试是否高效，而 DVE 应用中各属性之间互相独立，很难选择优先测试的标准。二叉决策图只能给出事件是否匹配该二叉决策图表示的订购集合，无法区分匹配和不匹配的订购，如果在 DVE 应用中使用基于二叉决策图的匹配算法，那么需为每个订购构造一个二叉决策图，或者扩充二叉决策图的语义，增加二叉决策图的出口节点，但这对于需要良好空间可扩展性和动态更新的 DVE 应用来说显然是不合适的。因此，在实验中，分别选择最具有代表性的计数算法^[9]和匹配树算法^[13]，作为基于谓词索引和基于匹配网络这 2 类算法的代表。

实验环境将以笔者自主开发的基于发布/订购通信模式的分布式虚拟环境系统 AIMNET 为仿真平台，主要包括一台运行 CEEM 算法的路由服务器、一台数据服务器以及 6 台主机。6 台主机连接到路由服务器上并通过多播方式进行通信，每台主机上都运行若干仿真对象，数据服务器则直接与路由服务器相连。其中路由服务器配置为：CPU P4 3.2GHz，RAM 1GB，System：Windows XP Professional。实验参数值如表 1 所示。

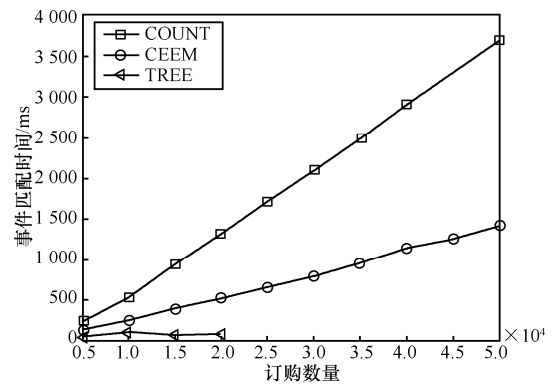
表 1 实验选取的参数值		
参数	实验 1	实验 2
属性数	5	7
一个订购中的谓词族数	8	11
等值谓词族	2	3
非等值谓词族	6	8
非等值谓词的值域	{0,65 535}	{0,65 535}
等值谓词的值域	{0,1,2,3,4,5}	{0,1,2,3,4,5}

实验中，根据上述 2 组参数，分别插入 50 000 条订购，每隔 5 000 条订购进行取样，记录当时插入 5 000 条订购所需的时间和匹配 200 个事件所需的时间。图 3 和图 4 分别记录了 2 组实验的结果。

根据两组实验结果可以看到，匹配树算法无论是插入订购还是事件匹配的效率均优于计数算法和 CEEM 算法，但由于匹配树自身的结构决定了它有很高的空间需求，该算法在拥有 15 000 条订购时还能继续插入 5 000 条订购，并进行匹配，但当订购数目达到 20 000 时，如果再试图插入 5 000 条订购将因内存不足而导致系统崩溃。

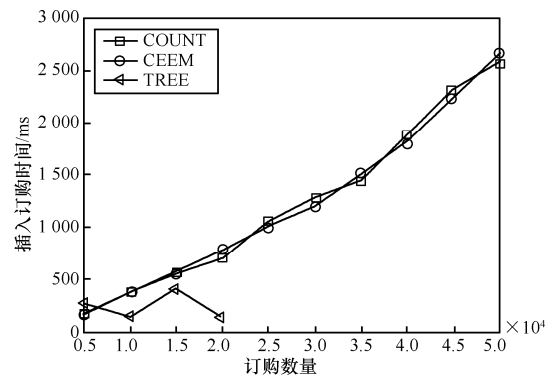


(a) 订购时间

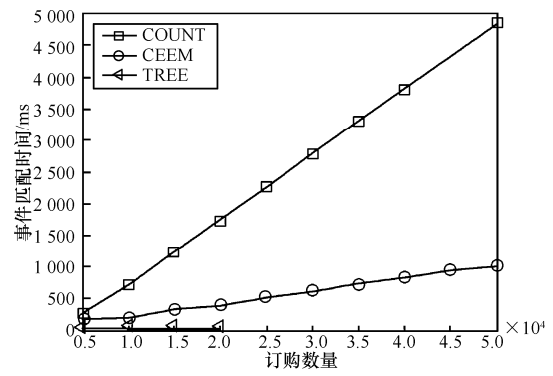


(b) 匹配时间

图 3 实验 1 的结果



(a) 订购时间



(b) 匹配时间

图 4 实验 2 的结果

对比计数算法和 CEEM 算法,无论是插入订购还是事件匹配效率均基本上随着订购数量的增长呈线性增长趋势。由于计数算法也采用了类似 CEEM 算法的谓词表以减少相同谓词族内的冗余比较,在插入订购时计数算法和 CEEM 算法拥有基本相同的效率,且插入订购的时间与订购中的谓词数以及已插入的订购数成正比。在匹配效率方面,计数算法和 CEEM 算法中的匹配时间也随着订购数量的不断增加呈线性增长,但是由于 CEEM 算法利用了谓词间的相关性进一步减少了不必要的比较操作,其斜率远远小于计数算法,即在具有相同订购数量的情况下,CEEM 算法的匹配效率极大的优于计数算法。此外,通过比较可以进一步发现,在实验 2 中,CEEM 算法的匹配效率优于实验 1。分析其原因,随着谓词数目的增加,事件匹配的成功率将会随之降低,CEEM 算法中能够提前舍弃订购的概率也将随着谓词数目的增加而增加。因此,CEEM 算法不仅在谓词数目较少的情况下效率优于计数算法,而且在谓词数目较多时,这种优势将更加明显。

另外,为了进一步验证匹配树算法和 CEEM 算法对存储空间的需求变化情况,将实验环境中的路由服务器配置调整为: CPU P4 3.2GHz, RAM 10GB (其他参数不变),在系统运行时分别记录了 2 组实验中路由服务器的内存使用情况。

从图 5 和图 6 中可以看到,CEEM 算法对路由服务器的内存空间需求明显低于匹配树算法,并且随着订购数量的不断增加,这种优势表现得越来越明显。分析可知,匹配树自身的树形结构决定了它有很高的空间需求,并且在实现时,为了区分匹配和不匹配的订购,需要为每个订购构造一个匹配树;而 CEEM 算法主要的匹配空间需求是谓词表,在实现时只需采用二维数组的形式,且所有的订购共用同一个谓词表,可避免相同订购谓词重复记录,减少了存储空间。因此,当 DVE 中大量的虚拟对象频繁地向系统发送订购和发布信息时,CEEM 算法能够有效解决匹配树算法面临的因路由器存储空间需求快速增加而导致的“内存危机”问题,不会对系统的运行效率产生造成较大影响。

综上所述,相对于其他算法,CEEM 算法更加适应于订购数目众多、并拥有较多谓词数目的 DVE 应用。

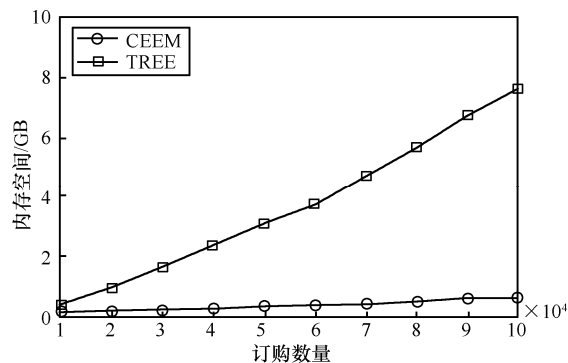


图 5 实验 1 的内存使用情况

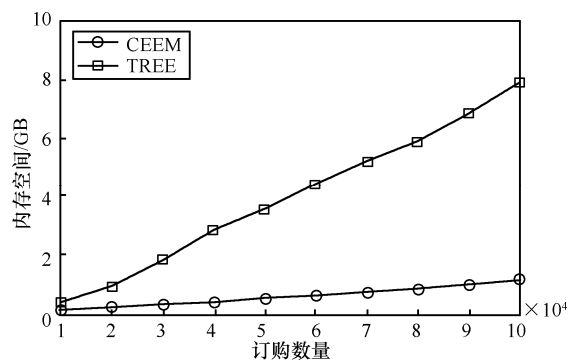


图 6 实验 2 的内存使用情况

5 结束语

本文提出的基于内容的快速事件匹配算法和现有相关算法相比,具有较大的改进。该算法利用了谓词间的关联性和属性间的相关性,加快了匹配的速度,使基于谓词索引和基于匹配网络算法各自的优点都能得到体现,同时又尽量避免了它们的不足。另外,本算法同时还实现了现有其他算法无法解决的事件匹配的对称性问题。最后通过实验分析,证实了该算法的正确性和有效性。

参考文献:

- [1] CAPPS M, MCGREGOR D, BRUTZMAN D, *et al.* Npsnet-v: a new beginning for dynamically extensible virtual environments[J]. IEEE Computer Graphics and Applications, 2003, 20(5):12-15.
- [2] JEHN R J, JIUN S C, SHUN Y H. Enhancing neighbor ship consistency for peer-to-peer distributed virtual environments[A]. 27th International Conference on Distributed Computing Systems Workshops[C]. Toronto, IEEE Computer Society, 2007.61-71.
- [3] NGUYEN B D, SUIPIN Z. Efficient client-to-server assignments for distributed virtual environments[A]. 27th International Conference on Parallel and Distributed Processing Symposium[C]. Rhodes Island, IEEE Computer Society, 2006.10-20.
- [4] 苑洪亮, 史殿习, 王怀民. 内容发布订阅中支持订阅覆盖的路由算

- 法研究[J]. 计算机学报, 2006, 29(10):1804-1812.
- YUAN H L, SHI D X, WANG H M. Research on routing algorithm based on subscription covering in content-based publish/subscribe[J]. Chinese Journal of Computers, 2006, 29(10): 1804-1812.
- [5] TRAN D A, PHAM C. PUB-2-SUB: a content-based publish/subscribe framework for cooperative P2P networks[A]. 8th International IFIP TC 6 Network Conference[C]. Germany, Springer, 2009. 770-781.
- [6] OLIVERIA M, CROWCROFT J, DIOS C. Router level filtering for receiver interest delivery[A]. NGC on Networked Group Communication[C]. New York, ACM Press, 2000.141-150.
- [7] ZABELS S, DORSCH M, GE Z. SANDS: specialized active networking for distributed simulation[A]. 2002 DARPA Active Networks Conference and Exposition[C]. Washington, IEEE Computer Society, 2004. 356-365
- [8] 贝佳, 曾定浩, 翟磊. 层次式主动兴趣管理研究[J]. 软件学报, 2006, 17 (10): 2163-2172.
- BEI J, ZENG D H, ZHAI L. Research on active leveled interest management[J]. Journal of Software, 2006, 17 (10): 2163-2172.
- [9] YAN T W, GARC H. Index structures for selective dissemination of information under the Boolean model[J]. ACM Trans Database System, 1994, 19(2): 332-334.
- [10] HANSON E, CHAABOUNI M, KIM C. A predicate matching algorithm for database rule systems[A]. International Conference of the ACM SIGMOD[C]. New York, ACM Press, 1990. 271-280.
- [11] FABRET F, A JACOBSEN, LIBAT F. Filtering algorithms and implementation for very fast publish/subscribe systems[A]. International Conference of the ACM SIGMOD[C]. New York, ACM Press, 2001. 115-126.
- [12] SHEN Z H, TIRTHAPURA S, ALURU S. Indexing for subscription covering in publish-subscribe systems[A]. IEEE International Conference on Data Engineering[C]. Tokyo, IEEE Computer Society, 2005. 32-43
- [13] AGUILERA M, STROM R, STURMAN D. Matching events in a content-based subscription system[A]. Eighteen ACM Symposium on Principles of Distributed Computing[C]. Atlanta, ACM Press, 1999. 53-61.
- [14] ANCEAUME E, DATTA A K, GRADINARIU M. A semantic overlay for self-peer-to-peer publish/subscribe[A]. 26th IEEE International Conference on Distributed Computing Systems[C]. Lisbon, Portugal, IEEE Computer Society, 2006.22-26.
- [15] CAMPAILLA A, CHAKI S, CLARKE E. Efficient filtering in publish-subscribe systems using binary decision diagrams[A]. 23rd International Conference on software Engineering[C]. Toronto, ACM Press, 2001. 443-452.
- [16] SILVIA B, PASCAL F, MARIA G. Content-based publish/subscribe using distributed R-trees[A]. International Conference on Parallel and Distributed Computing[C]. Rennes, France, 2007.537-548.
- [17] LI G, HOU S, JACOBSEN H. A unified approach to routing covering and merging in publish/subscribe systems based on modified binary decision diagrams[A]. 25th International Conference of ICDCS[C]. Columbus , USA, 2005. 447-457.
- [18] 张凯隆. 基于谓词覆盖路由的发布/订购机制研究[D]. 南京: 南京大学, 2009.
- ZHANG K L. Research on Publish/Subscribe Mechanism based on Predicate-covering Content Routing[D]. Nanjing: Nanjing University, 2009.

作者简介:



陈继明 (1977-), 男, 江苏镇江人, 博士, 江苏大学讲师, 主要研究方向为分布式虚拟环境和计算机网络。



鞠时光 (1955-), 男, 江苏镇江人, 博士, 江苏大学教授、博士生导师, 主要研究方向为数据库和信息安全技术。

潘金贵 (1952-), 男, 江苏南京人, 南京大学教授、博士生导师, 主要研究方向为多媒体技术和计算机图形。

邹志文 (1968-), 男, 江西宜黄人, 江苏大学博士生、副教授, 主要研究方向为空间数据库和虚拟现实。

龚震宇 (1978-), 男, 江苏南京人, 南京大学硕士生, 主要研究方向为分布式虚拟环境。