

文章编号:1001-9081(2005)03-0695-04

CSCW 协同建组协商策略研究

王 英, 张小真

(西南师范大学 计算机与信息科学学院, 重庆 400715)

(waying95@swnu.edu.cn)

摘 要:提出了一个支持协商建立协同工作组的自动协商策略集合。该策略集基于 PFCSP 限制表示框架将时间和资源因素综合考虑,克服了相关协商策略分离研究的限制。其中反提案生成结合使用了“折中”和“放松”两种策略,建组协商中特有的“组策略”被提出用于支持在建组内部统一意见的 meta 协商。通过分析各策略属性、通信开销和计算量,对该策略在协商建组环境下的正确性和有效性进行了验证,并以实例说明了策略的应用过程。

关键词:自动协商;agent;CSCW;协商策略

中图分类号:TP30 **文献标识码:**A

Research on negotiation strategies of establishing cooperative teams in CSCW

WANG Ying, ZHANG Xiao-zhen

(College of Computer & Information Science, Southwest China Normal University, Chongqing 400715, China)

Abstract: A strategy set was put forward to support constructing cooperative work teams by automated negotiation. The set grounded on restriction representation framework PFCSP to take time and resource factors into consideration. It overcame those restrictions coming from separating factors. "Trade-off" and "relax" strategies were used for generating counter-offer. A special "group-strategy" was proposed to support meta-negotiation of establishing team. From the perspectives of communication and computation cost, the correctness and validity of this strategy set were verified under the negotiation environment of establishing cooperative work teams. Finally one example was demonstrated on application of these presented strategies.

Key words: automated negotiation; agent; CSCW; negotiation strategy

CSCW(Computer Supported Cooperative Work)的广泛应用不断改变着人们的生活工作方式。建立适合的工作组是 CSCW 有效实施的基本前提。Ikeda, M 等人提出了利用自动协商建组以克服了单纯系统硬性分组不足的方法。但是所提的 FITS/CL 框架采用 1:1 协商模型,没有考虑多组协商和群决策情况,也没有对其中的协商策略进行相关的研究。

因此我们扩展了 Ikeda, M 等人的工作,提出了支持主动式建组的多对多(简称 m:m)协商模型,支持群决策和多组协商。协商策略作为协商模型的重要组成部分,决定了如何评估方案(offer)与产生反提案(counter-offer)。目前的研究一方面将时间、资源等因素分离,忽略了彼此的相互影响^[1-6];另一方面没有针对建组协商中群决策使用的相关策略进行分析。本文以 PFCSP(Prioritized Fuzzy Constraint Satisfaction Problems)^[3]为基础提出一组针对建组的协商策略。其中的方案评估和反提案产生策略结合了时间和资源因素。提出的“组策略”支持决策群体不断变动地在建组内部 meta 协商。本文就策略的有效性和正确性进行了分析,并以实例来说明策略的应用情况。

1 协商建组

协商建组在减轻人类协商负担的同时也提高了建组满意度。建组方案作为磋商对象,所建组的各种参数,如组大小,工作方式等成为协商主题。参与协同工作的用户是协商者,整个协商过程受时间、资源等条件限制。

1.1 协商建组过程

建组协商属于 m:m 协商模型,协商环境中同时存在多个建组协商任务。买方是想建组的 agent,卖方是愿意参加组的 agent。但与电子商务领域不同的是:供出售的商品不是无意识的物品而是具有主动性的人。建组协商过程参见图 1。其中“组员”在参与者被在建组接受,组还未完全建立期间存在。

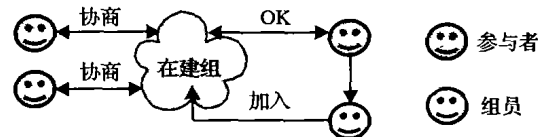


图 1 建组协商协商示意图

1.2 协商建组中的策略集

建组协商采用基于策略的协商方法^[7]。agent 通过策略来评估、决策并确定下一步的行动。传统的协商策略^[4]分两类:方案评估和反提案产生策略。但是在建组内部需要统一意见消除分歧,因此引入了组策略。所以本文存在三种策略:

1) 发起者和组员 agent 使用的策略包括方案评估策略、反提案产生策略和组策略。

2) 参与者 agent 使用的策略包括方案评估策略和反提案产生策略。

3) 参与者 agent 使用方案评估策略对收到的提案评估效益,不能接受则用反提案策略产生自己的方案。在建组员(发起者和组员)采用“组策略”对参与者提出的方案进行评价和组内

收稿日期:2004-08-04;修订日期:2004-10-28

作者简介:王英(1980-),女,四川简阳人,硕士研究生,主要研究方向:ICAI、DAI 系统; 张小真(1945-),女,重庆北碚人,博士生导师,主要研究方向:ICAI、网络软件体系结构。

决策,拒绝接受时运用组策略提出反提案;可接受时再对参与者本人进行评估,最后在建组对参与者的加入作出统一决定。

2 协商策略

协商策略决定 agent 求取最大收益的协商解的最优行为序列^[4]。下面我们先提出单个 agent 使用的策略,之后再讨论组策略。

2.1 提案评估策略

方案评估策略用于发起者、参与者、组员 agent 各自评估方案收益值。时间和资源是影响同一方案收益的重要因素,过去的研究多是分离它们的影响^[4]。实际上越耐心的 agent 出价越晚收益越高,它也同时遵循“物以稀为贵”的准则,即资源越紧缺同一方案收益值越大。因此本文的提案评估策略综合考虑了时间和资源因素(如下所示):

$$U_a = V_a(\delta_a) \left(\frac{1}{rs_a} \right)^t, V_a \in [0, 1] \text{ 是方案满足限制程度。}$$

$$V_a = \alpha(\text{offer}_i^t), \alpha \text{ 是 offer 满意度函数}^{[3]}。$$

$$\delta_a \text{ 为 agent 的时间折扣因子}^{[2]}, \delta_a \begin{cases} \delta_a > 1, \text{ 耐心 agent} \\ \delta_a = 1, \text{ 冷漠 agent} \\ \delta_a < 1, \text{ 急躁 agent} \end{cases}$$

rs_a 为 agent 的资源丰裕因子,这里的资源指可协商 agent。

$$rs_a = \frac{|A_a(t)|}{F_a(t) - |M_a(t)|}$$

$|A_a(t)|$ 为 t 时刻可协商 agent 数目; $F_a(t)$ 为 t 时刻 offer 要求人数, $|M_a(t)|$ 为 t 时刻已加入组员数。

2.2 反提案生成策略

方案收益低于阈值时 agent 拒绝提案并提出反提案,本文采用了两种策略生成反提案:

平衡折中策略 从等收益方案集中选出与对方上次提案最接近的方案作为反提案。

“放松”策略 无法产生等收益反提案时,以时间和资源因素为依据放松限制生成反提案。

2.2.1 平衡折中策略(trade-off)

Agent 要保证收益最大需尽量避免让步。在协商解空间中存在一些方案,它们的一些主题值比当前方案高,一些比当前的低。但两者总收益相等且前者可能令对手更满意(即与对手提案最相似)^[1]。

反提案:

$$\text{offer}_{a \rightarrow b}^t = \text{trade-off}(p, q) = \underset{r \in \text{iso}_a(\theta)}{\text{argmax}} \{ \text{Sim}(r, q) \}$$

$$p = \text{offer}_{a \rightarrow b}^t, q = \text{offer}_{b \rightarrow a}^{t+1},$$

$$\text{iso}_a(\theta) = \{ \text{offer}_{a \rightarrow b}^t \mid U_a(\text{offer}_{a \rightarrow b}^t) = \theta \}$$

$$\text{方案相似度 } \text{Sim}(r, q) = \sum_{j \in Q} p_j^a \cdot \text{sim}_j(p_j, q_j),$$

p_j^a 为 a 中主题 j 的重要度。

主题相似度 $\text{sim}_j(p_j, q_j) = |\mu_j^p - \mu_j^q|$, μ_j^p 表示方案 P 中主题值对应限制 μ 值, Q 为协商主题集。

平衡折中策略的算法^[1]如下:

$$\text{Trade-off}(f^j, E, \mu_i, \rho_i, U_a, Q) \\ // f^j \text{ 为收到方案, } E = \frac{U_a(f^j) - U_a(X)}{\text{step}}$$

$s = \text{step}$; Vector offers = new Vector();
while ($s > 0$)

{
 $t = N$; // 每次产生的 offer 个数
while ($t > 0$)

```
{ fj = fj+1;
  offers.add( trade-off( fj ) );
  t --; }
fj+1 = getMaxSimilar( offers );
s --;
return fj+1;
}

trade-off( fj )
{ while( i ∈ Q ) Ei = { Δμi | Δμi = | μi( rif ) - μi( fij ) | }
  Emax = ∑ ρi · max( Ei );
  if ( E + δ < Emax )
  { k = 0; E' = 0; // E' 为当前总收益
    while( E' < E )
    { k ++;
      while ( i ∈ Q )
      { if ( E' < E )
          eik = random{ Δμi | Δμi ∈ Ei, Δμi ≤  $\frac{E - E'}{\rho_i}$  } ∪ {0} }
          else eik = 0; }
        E' = E' + ρi · eik;
        Ei = { Δμi | Δμi = | μi( rif ) - μi( fij ) + ∑j=1k eij | }
      }
    } while ( i ∈ Q )
    { Ei = ∑j=1k eij; fij+1 = Ui-1( Ui( fij ) + Ei ); }
    return fj+1; }
```

2.2.2 放松策略

找不到等效益反提案时,让步是最优选择,因为达成一致显然比协商失败好。让步策略每次放松一个重要度最低的限制以保证收益损失最少,放松度由当前的时间和资源情况决定。两者的相互影响如表 1 所示:从中可以发现只在资源和时间均不足情况下才与单由时间确定的反提案不同^[2],其他情况下与单独考虑时间因素时无差别。

表 1 时间与资源相互影响统计表

时间条件	相互影响	
	资源充足	资源不足
时间充足	$\delta_a > 1 \quad \eta = 1, \beta_1 > 1, \beta_2 = 1$ $\delta_a = 1 \quad \eta = 1, \beta_1 = 1, \beta_2 > 1$ $\delta_a < 1 \quad \eta = 1, \beta_1 = 1, \beta_2 > 1$	$\eta = 1, \beta_1 > 1, \beta_2 = 1$ $\eta = 1, \beta_1 = 1, \beta_2 > 1$ $\eta = 1, \beta_1 = 1, \beta_2 > 1$
时间不足	$\delta_a > 1 \quad \eta = 1, \beta_1 > 1, \beta_2 = 1 \quad \eta = \left\lfloor \frac{1}{\delta_a} \cdot \frac{1}{rs(t)} \right\rfloor \beta_1 > 1, \beta_2 = 1$ $\delta_a = 1 \quad \eta = 1, \beta_1 = 1, \beta_2 > 1 \quad \eta = \left\lfloor \frac{1}{\delta_a} \cdot \frac{1}{rs(t)} \right\rfloor \beta_1 = 1, \beta_2 > 1$ $\delta_a < 1 \quad \eta = 1, \beta_1 = 1, \beta_2 > 1 \quad \eta = \left\lfloor \frac{1}{\delta_a} \cdot \frac{1}{rs(t)} \right\rfloor \beta_1 = 1, \beta_2 > 1$	

结合时间和资源因素后的放松策略如下:

$$\text{offer}_{a \rightarrow b}^t = \{ r_i \mid r_i \in R, R = \{ r_{i/k} \in \text{offer}_{a \rightarrow b}^t \} \cup r_k \}$$

r_k : relax(μ_k); i/k : 除 k 外其他元素;

relax(μ_k): 放松限制限制 r_k 为 r_k^c ;

c : μ 值在序列 $\mu_k^1 \leq \mu_k^2 \leq \dots \leq \mu_k^{n_i}$ 中的序号;

$$c = n_i - \left(\left\lfloor \left(\frac{t' - t_{\text{start}}}{\Delta t} \right) / \beta_1 \right\rfloor \right) \cdot (\beta_2)^\eta$$

t_{start} : 协商线程开始时刻; β_2 : μ 变化步长;

β_1 (agent 放松容忍度因子): 坚持多久不改变 μ ,

$$\begin{cases} \delta_a > 1: \beta_1 > 1, \beta_2 = 1; \\ \delta_a = 1: \beta_1 = 1, \beta_2 = 1; \\ \delta_a < 1: \beta_1 = 1, \beta_2 > 1; \end{cases}$$

η (出价加速因子): 取决于资源情况;

T_{\max} : 建组任务总时限; t_d : 协商线程时限;

t_{κ} : 资源出现不足时刻; t_c : 时间出现不足时刻;

rs_u : 资源丰裕度因子 $rs_u(t) \in [0, \infty)$

$$\begin{cases} t' \leq t_c, t_{\kappa} < t_c \text{ 或 } t_{\kappa} \geq t_c: \eta = 1; \\ t' > t_c, t_{\kappa} < t_c: \eta = 1; \\ t' > t_c, t_{\kappa} \geq t_c: \eta = \left\lfloor \frac{1}{\delta_u} \cdot \frac{1}{rs_u} \right\rfloor \end{cases}$$

2.3 组策略

组策略用于在建组的 meta 协商。组策略以群决策论中的投票机制为核心,以组相对稳定为前提——避免新人加入使已有组员大量离开。在建组随组员加入退出不断变动。方案评估、反提案产生和参与者加入都要求在建组综合各方意见做出统一决策。因此在建组的 meta 协商是必需的,为支持 meta 协商我们提出了以下组策略。

2.3.1 组提案评估策略

组采用“投票”策略评估提案,在建组根据统计者 agent 收集的组员结果使用以下策略进行决策:

$$dec_{a \rightarrow b}^g(t') = \begin{cases} \text{接收,} & U_g = \sum_{i \in \text{member}} U_i^g \cdot \varphi_i \geq \delta_g^a \\ \text{提出反提案,} & \text{otherwise} \end{cases}$$

δ_g^a : 组方案接受阈值; U_i^g : 组员 i 的结果 $U_i^g = 1/0$;

$$\varphi_i = \frac{\varphi_i}{\sum_{j \in M} \varphi_j}; \text{ 组员 } i \text{ 在组中的相对权重。}$$

2.3.2 组参与者加入策略

同样采用“投票加权统计”策略:

$$dec_{a \rightarrow b}^p(t') = \begin{cases} \text{加入,} & D_p = \sum_{i \in \text{member}} D_i \cdot \varphi_i \geq \delta_g^p \\ \text{拒绝,} & \text{否则} \end{cases}$$

$$D_i = \begin{cases} 1, & u_i \oplus_p cd_i^{[3]} \geq \delta_p^p \\ 0, & \text{否则} \end{cases}$$

u_i 为参与者 i 方案收益; cd_i 为参与者信誉度。

2.3.3 组反提案产生策略

组产生反提案先用 I 策略,再采用 II 策略。

I 当前初始提案

当前组方案(G)是组员达成共识的共享互斥变量。组与一个参与者协商时对互斥量 G 的修改会影响它与其他参与者协商的反提案产生。该策略为:当在建组与某一参与者达成一致时,改变 G 中主题值后变为 G' 。在建组与其他参与者协商时首先将 G' 作为反提案。若没有修改 G 中主题值则此策略跳过。

II 反提案投票策略

该策略由投票和统计两阶段组成。组员先提交预案(使用单人折中或放松策略),组再对预案进行投票。预案的产生有两种方法:

串行提案方式 按组员权重或随机方式传递提案权,每次 1 人提交 1 个预案,在建组就是否接受该方案为组反提案投票;

并行提案方式 愿意的组员均可提交预案,同时有多个预案时,组员进行多选一投票确定组的反提案。

串行投票方法简单计算量小开销少,但提案不一定最优;并行方式通信量较大成本较高,但方案相对较优。

统计投票有两种方式可选:

固定统计人 投票统计者由固定组员充当,在建组中发起者是较为稳定的成员,故一般由他充当固定统计者,因而发起者负担较重但没有额外管理开销。

轮流统计人 统计者由在建组组员轮流充当,每人一次;统计负担被分担但增加管理开销。

3 策略和实例分析

3.1 策略分析

实验以 Java 语言为工具,以多线程技术和并发访问控制方法为基础进行设计。协商环境中每个发起者的建组要求都产生一个协商过程(process),所有愿意参加该协商过程的参与者产生一个与在建组的协商线程,如图 2 所示。

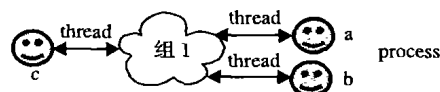


图2 实验设计示意图

一个线程协商成功,参与者加入在建组,它与组之间线程被杀死,生成组员 agent 后,参与者 agent 死亡,组员 agent 加入在建组的 meta 协商(图 3)。

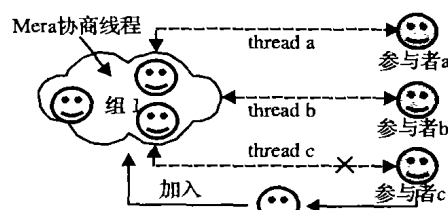


图3 协商线程转换图

实验系统提供 4 种工作方式(表 2)。

表 2 投票机的工作方式

提案方式	统计方式	
	固定统计	轮流统计
串行提案	串行/固定	串行/轮流
并行提案	并行/固定	并行/轮流

本文的协商策略在这样的环境下进行时,若将在建组整体看成一个协商者,建组协商过程就类似于 Rubinstein 的讨价还价模型,存在无穷个 Nash 均衡。使用“组策略”的在建组的 meta 协商就好似投票博弈。就单 agent 策略来看,文献[1]已经就 trade-off 策略的可行性和正确性给出了详细论述,而文献[2]就时限下的协商策略属性进行了深入探讨。本文的放松策略只在时间和资源均不足情况下加速出价,即将文献[2]中达成一致的最优时间 J 提前为 J' ,对于出价顺序和图示的出价曲线走向没有影响,所以该策略仍然具有文献[2]中策略的“序贯均衡性”,因而策略的正确性得到保证。

表 3 协商 process 通信成本表

协商成员	通信成本
参与者	$C_{pt}^k = h_i^k(T, p_k) \cdot h_i^k(T, p_k) $ 参与者 p_k 与在建组 T 间协商线程 i 的序列长度
发起者	$C_{pr} = h_{\text{meta}}^j(pr, m_j) + N_{ad}$ m_j : meta 协商统计者, N_{ad} : 发出建组邀请数
组员	$C_m = h_r^j(m_r, m_j) $ 即组员 r 与统计者 j 间 meta 协商序列长度
Meta 协商统计者	$C_j = \sum_{m_i \in N_m / \{m_j\}} h_r^j(m_i, m_j) $
总计	$C_{total} = N_{pr} \cdot C_{pt} + C_{pr} + N_m \cdot C_m + C_j$

在上述实验环境下协商过程的通信成本耗费如表 3 所示。通信开销与各类 agent 数量成正比例线性关系。当发起者或组员兼任统计者时各自的通信负荷增加,但仍然保持线性关系。协商策略所引起的计算量(参见表 4)与协商主题成正比例线性关系。因此运用本文所提策略进行建组协商其通信和计算开销都保持线性关系,保证了策略的有效性。

表 4 1 次协商 process 计算成本

步骤	计算成本			
	$O(pt)$	$O(pr)$	$O(mb)$	$O(st)$
方案评估	$O(n_i)$	$O(n_i)$	$O(n_i)$	$O(n_i)$
反提案	$O(n_i)$	$O(n_i)$	$O(n_i)$	无
产生	无	无	无	$O(1)$
组 -I	$O(n_i)$	$O(n_i)$	$O(n_i)$	$O(n_i)$
组 -II	$O(n_i)$	$O(n_i)$	$O(n_i)$	$O(n_i)$
总计	$4O(n_i)$	$4O(n_i)$	$4O(n_i)$	$O(n_i) + O(N_m) + O(1)$

(n_i : 主题数, pt : 参与者, pr : 发起者, mb : 组员, st : 统计者)

3.2 实例分析

考虑有发起者 p , 参与者 a, b, c , 在建组为 $\{p\}$ 。各 agent 需求限制见图 4。发起者发出建组邀请(含初始建组方案)后, agent a, b 和 c 都愿意参加建组协商过程。但是根据 offer 收益:

$$u_a(\text{offer}) = 0.6 \times 0.6 + 1 \times 0.4 = 0.76 < 0.8 (\text{offer-level})$$

$$u_b(\text{offer}) = 0.1 \times 0.5 + 0.5 \times 1 = 0.55 < 0.6$$

$$u_c(\text{offer}) = 1 \times 0.8 + 0.7 \times 0.2 = 0.94 > 0.9$$

因此 a 和 b 不能接受方案, 而 c 可接受。 c 被在建组评估:

$$u_p(c) = 1 \oplus_p 0.9 =$$

$$\frac{(1-0.7) \times 1 \times 0.9}{(1-0.7) \times 1 \times 0.9 + (1-1) \times (1-0.9) \times 0.7} = 1 > 0.7$$

在建组变成 $\{p, c\}$, 组方案中 $gs = 3$, 还需要组员。 a 和 b 将提出反提案, 我们以 a 为例。

a 先使用折中策略, 但找不到当前收益为 1 的其他方案, 运用“放松策略”选择重要度最低限制的“角色 r ”放松。放松后的 u 值计算如下: $t_{\text{start}} = 0, t' = 1, \Delta t = 1, \beta_1 = 1, \beta_2 = 2, t_c = 12 \times 0.2 = 2.4, t' < t_c$ 。

假设此时 a 有个组邀请它, $rs_a = 2$, 根据表 2, $\eta = 1, c = 4 - \left\lfloor \frac{1-0}{1} / 1 \right\rfloor \times 2^1 = 2$

反提案为: $\{cm = \text{"短信"}, r4\}$ 。假设系统采用串行/固定的工作方式, 在建组收到该反提案, 各组员计算自己收益后投票:

$$u_p(\text{offer}) = 0.94 > 0.8, \text{投票赞成};$$

$$u_c(\text{offer}) = 0.94 > 0.9, \text{赞成}。$$

发起者为统计员, 统计结果为:

$$0.94 \times \frac{0.4}{0.4+0.2} + 0.94 \times 0.2 = 0.931 > 0.8$$

(groupofferlevel)

组接受 a 的方案, 评估 a :

$$u_p(a) = 0.94 \oplus_p 0.9 = \frac{0.2538}{0.258} = 0.984$$

$$u_c(a) = 0.94 \oplus_p 0.9 = \frac{0.2538}{0.258} = 0.984$$

$$\text{统计结果为: } 0.984 \times \frac{0.4}{0.6} + 0.984 \times \frac{0.2}{0.6} = 0.984 > 0.7$$

a 被接受加入, 此时在建组人数满足方案 $gs = 3$, 建组成功。如果系统采用并行/固定工作方式, 同一段时间内多个参与者提案中收益最高的参与者先加入组。

如果组投票方案收益 < 0.8 , 则组提出反提案。假设并行方式下 p 提出的反提案为: ①: $\{gs = 3, r1, cm = \text{"bbs"}\}$, c 的反提案为 ②: $\{wt = 2, r4\}$ 。两个预案组员各自收益为: $u_p(\text{①}) = 0.92$,

$u_p(\text{②}) = 1$, 发起者投方案 ②; $u_c(\text{①}) = 0.94, u_c(\text{②}) = 0.92, c$

投方案 ①; 投票统计 ② 结果为: $1 \times \frac{0.4}{0.6} + 0 \times \frac{0.2}{0.6} = 0.667$; ① 为:

$$0 \times \frac{0.4}{0.6} + 1 \times \frac{0.2}{0.6} = 0.333。最后在组选取方案 ② 为反提案。$$

P	gs (组大小)		m (限制重要度)		r (角色)		u m	
	gs=3	1	0.6		r1	1	0.3	
	gs=2	0.8			r2	0.8		
	$5 \geq gs \geq 3$	0.8			r3	0.4		
	gs>1	0.2			r4	0.2		
	gs>5	0.2						
P	cm (联系方式)		u m		pm (组参数)			
	cm="email"	1	0.1		roles (角色集合)={r1(0.4), r2(0.3), r3(0.2), r4(0.1)} cutlevel=1; relaxlevel=0.8 offerlevel=0.8; participantlevel=0.7; groupOfferLevel=0.8 groupParticipantLevel=0.7 $\delta_a = 1.2; \beta_1 = 2; \beta_2 = 2;$ $t_d = 10s; T_{\max} = 30s;$ $rs_a^{\text{level}} = 0.4; rt_a^{\text{level}} = 0.3$			
	cm="bbs"	0.6						
	cm="file"	0.4						
	cm="短信"	0.2						

组的用户信誉度记录

a	0.9
b	0.75
c	0.6

Initial-offer={gs=2, roles={r2, r3, r4}, cm="email"}

	cm	u	m	r	u	m
a	cm="短信"	1	0.6	r2	1	0.4
	cm="bbs"	0.8		r4	0.8	
	cm="file"	0.7		r3	0.7	
	cm="email"	0.6		r1	0.3	
	pm					
cutlevel=1; relaxlevel=0.6; offerLevel=0.8; participantlevel=0.8			$\delta_a = 0.8; \beta_1 = 1;$ $\beta_2 = 2; t_d = 12s;$ $rs_a^{level} = 0.3; rt_a^{level} = 0.2$			
	gs	u	m	wt (工作时间)	u	m
b	gs =4	1	0.5	1h	1	0.5
	$7 \geq gs \geq 4$	0.8		2h	0.9	
	gs>7	0.2		3h	0.6	
	gs<4	0.1		wt>3	0.4	
	pm					
cutlevel=1; relaxlevel=0.7; offerlevel=0.6; participantlevel=0.7;			$\delta_a = 1.8; \beta_1 = 2; \beta_2 = 2; t_d = 21s;$ $rs_b^{level} = 0.2; rt_b^{level} = 0.2$			
	wt	u	m	r	u	m
c	2	1	0.8	r1	1	0.2
	1	0.9		r2	0.7	
	3	0.6		r4	0.6	
	wt>3	0.4		r2	0.2	
	pm					
cutlevel=1; relaxlevel=0.7; offerlevel=0.9; participantlevel=0.7			$\delta_a = 0.6; \beta_1 = 2; \beta_2 = 1;$ $t_d = 15s; rs_c^{level} = 0.5;$ $rt_c^{level} = 0.5$			

图 4 agent 的需求限制和各种参数情况

4 结语

目前协商策略的研究主要单独考虑各因素作用, 也没有提出针对变动决策体内 meta 协商的相关策略。

(下转第 705 页)

FDL集成信息检索系统中的数据源接口描述文档采用RDF格式表示,包含了数据源的地址、接口格式等信息。包装器通过解析接口描述文档,完成查询请求的转换。将查询结果封装成RDF文档的过程较为简单,不考虑元数据标准问题,只以RDF文档格式将结果返回,由中介器进行结果的整合。

为了减少系统负荷,包装器产生器采取对象池(pool)的方式进行管理。系统初始化时,首先创建一定数量(该数量由系统规模确定,可修改)的包装器对象实例,将这些实例以队列方式存放,即组成一个pool。当包装器产生器接收到来自中介器的查询请求后,从pool中获取当前空闲实例(即一个包装器),根据接口描述文档,将某一数据源ID传递给该实例,开始检索工作。当任务结束后,实例并不释放,而是重新进入空闲状态,保存在队列中,以便下次使用。

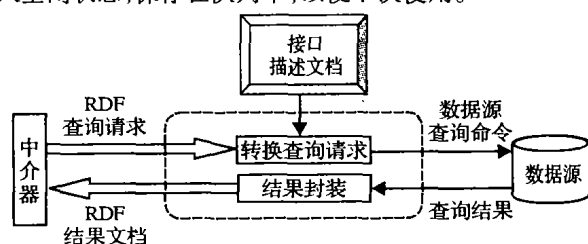


图3 包装器结构图

3 原型系统实现与评价

3.1 原型系统实现

在中国数字图书馆有限责任公司现有的异构数据源的基础上,我们目前实现了基于上述检索机制的FDL集成信息检索原型系统。通过配置,原型系统支持对SDLIP(Simple Digital Library Interoperability Protocol)服务器和Z39.50服务器的集成检索。整个原型系统以Java语言实现。

SDLIP协议是由美国的斯坦福大学、伯克利大学等共同合作开发的服务器交互、数据检索协议。我们采用斯坦福大学的SDLIP Toolkit建立了SDLIP服务器供原型系统测试运行^[7]。Z39.50协议描述了Client/Server下检索的数据结构与交互规则,它是网络中的应用层协议,定义了客户机/服务器之间数据交换的标准。我们以现有的Z39.50服务器作为原型系统的测试数据源。

3.2 系统性能评价

因为本系统是针对FDL领域中的异构数据源的检索而设计的,所以本系统基于元数据进行检索,而并非全文检索。FDL中信息源的对象数据通常采用标准元数据进行描述,所以检索精度主要取决于不同标准元数据之间映射、转换时产

生的偏差。因此在评价系统的检索精确度时,主要衡量元数据映射机制。因为元数据转换贯穿整个系统中信息交换过程,要提高系统效率,必须考虑元数据转换效率。

在本系统中,以DC作为标准统一元数据。当前应用较为普遍的元数据体系为MARC(Machine Readable Catalogue),因此原型系统仅考虑了DC和MARC之间的映射。之所以选择DC作为系统标准,是因为MARC修订程序相当复杂,不易处理动态的多媒体信息,而且不太适应Internet环境,最重要的是,MARC著录项要比DC的复杂得多。因此,我们选择DC作为系统转换标准,而且在具体使用中,只考虑DC的15个基本著录项,忽略修饰词,这样大大缩减了元数据映射表的规模,缩短了中介器检索映射表的时间,从整体上提高了系统效率。同时这也符合一般FDL信息检索规律,因为FDL的信息具有海量、动态性等特点,用户更注重检索的易用性、精度和速度,而不是通常信息检索系统所关注的召回率(recall)。

此外,系统在检索过程中表现了良好的透明性和可维护性,对维护人员而言,很容易保证系统兼容动态变化的FDL异构数据源。

参考文献:

- [1] GONÇALVES MA, FRANCE RK, FOX EA, et al. Flexible Interoperability in a Federated Digital Library of Theses and Dissertations [EB/OL]. <http://www.nudl.org/papers/ICDE2001.pdf>, 2004-04-05.
- [2] 欧洁. 联合数字图书馆信息服务的关键技术研究[D]. 中国科学院计算技术研究所, 2002. 18-19.
- [3] 李石君. Web信息集成中的包装器归纳学习构造[D]. 武汉大学, 2001. 2-10.
- [4] CHANG YS, HO M-H, SUN W-C, et al. Supporting unified interface to wrapper generator in integrated information retrieval[J]. Computer Standards & Interfaces, 2002, 24(4): 291-309.
- [5] LASSILA O, SWICK R. Resource Description Framework (RDF) Model and Syntax Specification [S/OL]. <http://www.w3.org/TR/REC-rdf-syntax/>, 2003-10-16.
- [6] SHI X-L, NIU Z-D, SONG H-T, et al. Intelligent Agent-based System for Digital Libraries Information Retrieval [J]. Journal of Beijing Institute of Technology, 2003, 12(4): 450-454.
- [7] The Simple Digital Library Interoperability Protocol (SDLIP-core) [EB/OL]. <http://www.diglib.stanford.edu/~testbed/doc2/SDLIP/>, 2004-04-20.
- [8] DECKER S, van HARMELEN F, BROEKSTRA J, et al. The Semantic Web - on the respective Roles of XML and RDF [EB/OL]. <http://www-db.stanford.edu/~stefan/>, 2004-01-20.

(上接第698页)

策略对自动协商起着十分重要的作用^[7]。我们从现实生活出发探索了在建组协商中采用的协商策略,尤其探讨了结合时间和资源因素的放松策略以及支持组meta协商开展的“组策略”。考虑到协同工作的性质本文策略以整体利益为先。建组协商的agent是非完全理性的,受到计算能力和资源的限制^[7]。在分析协商策略的通信和计算开销后我们发现协商线程数对agent的协商效率有直接影响。因此研究协商中参与者数量和各自参加建组线程间的关系将是我们下一步的努力方向和工作重心。

参考文献:

- [1] FARATIN P, SIERRA C, JENNINGS NR. Using Similarity Criteria to make issue trade-off in automated negotiations[J]. Artificial Intelligence, 2002, 142(2): 205-237.
- [2] FATIMA SS, WOOLDRIDGE M, JENNINGS NR. An agenda-based framework for multi-issue negotiation[J]. Artificial Intelligence,

2004, 152(1): 1-45.

- [3] LUO XD, JENNINGS NR, SHADBOLT N, et al. A fuzzy constraint based model for bilateral, multi-issue negotiations in semi-competitive environments [J]. Artificial Intelligence, 2003, 148(12): 53-102.
- [4] FARATIN P, SIERRA C, JENNINGS NR. Negotiation Decision Functions for Autonomous Agents [J]. Internal Journal of Robotics and Autonomous Systems, 1998, 24(3/4): 159-182.
- [5] FARATIN P, SIERRA C, JENNINGS NR, et al. Designing Responsive and Deliberative Automated Negotiations [A]. Proceedings of the AAAI Workshop on Negotiation: Settling Conflicts and Identifying Opportunities [C], 1999. 12-18.
- [6] FATIMA SS, WOOLDRIDGE M, JENNINGS NR. Optimal Negotiation Strategies for Agents with Incomplete Information [A]. ATAL-2001 [C], Seattle, USA, 2001. 53-68.
- [7] JENNINGS NR, FARATIN P, LOMUSCIO AR, et al. Automated-negotiation: prospects, methods and challenges [J]. International Journal of Group Decision and Negotiation, 2001, 10(2): 199-215.