

一种基于动态规划的自动信任协商策略

姚 慧,高承实,戴 青,张 徐

(信息工程大学 电子技术学院,郑州 450004)

(yy_yaohui@yahoo.com.cn)

摘 要:动态规划是解决多阶段决策过程最优化的一种数学方法,可将其运用到自动信任协商中。针对目前有关协商策略的研究中没有区分信任凭证的敏感度和格式的问题,引入披露开销的概念,设计了一种新的协商策略。该策略采用动态规划的思想,基于与/或图建模,分解协商过程,自底向上求解最小开销的凭证披露序列。经证明,该策略是可采纳且高效的,能保障协商的安全性和提高协商的效率。

关键词:自动信任协商;协商策略;动态规划;访问控制策略

中图分类号: TP393.08 **文献标志码:** A

Dynamic programming-based strategy for automated trust negotiation

YAO Hui, GAO Cheng-shi, DAI Qing, ZHANG Xu

(Institute of Electronic Technology, Information Engineering University, Zhengzhou Henan 450004, China)

Abstract: Dynamic programming, an approach to solve the optimal question, was introduced into automated trust negotiation. The disclose cost was introduced to solve the problem of treating all credentials with the same sensitivity and the same format. A negotiation strategy modeled on AND/OR graph was proposed under the idea of dynamic programming to achieve the global optimization. The strategy achieved the minimal cost disclose sequence by traversing the AND/OR graph upward. The strategy is proved to be adoptable and efficient to ensure the safety and efficiency of the negotiation.

Key words: Automated Trust Negotiation (ATN); negotiation strategy; dynamic programming; access control policy

0 引言

开放网络环境中安全问题的主要根源之一在于网上实体的匿名性,即无法确切得知网络实体的身份,也就无法获取该实体的授权信息。自动信任协商^[1] (Automated Trust Negotiation, ATN) 给出了一种实体间建立信任关系的新方法,它通过协作主体间信任凭证、访问控制策略的交互披露,逐渐为各方确认身份,完成授权,建立信任关系^[2]。

协商策略^[1]是 ATN 研究的重要方面,决定着整个协商过程的安全和效率。目前提出的协商策略中,绝大多数将每个信任凭证都视为相同的敏感度^[3],对不同格式的凭证给网络流量带来的不同影响也未区分。然而出于对安全和效率的考虑,不同的凭证应有不同的敏感度,不同格式的凭证对网络通信的影响也应考虑在内。比如说信用卡号显然比电话号码敏感,实体对更为敏感的信用卡号凭证制定的访问控制策略肯定比对电话号码凭证制定的访问控制策略要复杂,披露信用卡凭证对网络流量的影响也更大,为了安全和效率,在可选的情况下实体必然愿意向对方披露电话号码而非信用卡号。

Weifeng Chen^[4]提出的基于策略图的策略,将凭证仅按敏感度的不同赋予不同的开销,运用 Dijkstra 算法,旨在找到协商总开销最小的凭证披露序列,却被证明为 NP 完全的。本文设计了一种高效的协商策略,同时考虑凭证的敏感度和格式,对其赋予合理的披露开销,基于动态规划^[5]建模,多阶段决策最优化,只要存在,总能找到协商总开销最小的凭证披

露序列,直至协商成功或失败。

1 基本概念

1.1 ATN 框架描述

信任凭证(简称凭证)是指由证书发布机构签发的证明凭证拥有者有效身份和相关属性的数字断言,具有可验证性和不可伪造性,记为 C_i 或 S_i 。可直接披露的凭证称无保护凭证。用包含布尔运算符 \wedge 、 \vee 、及括号的凭证逻辑表达式 $\Phi_i(C_1, C_2, \dots, C_n)$ 表示访问控制策略,本文中涉及的访问控制策略的逻辑表达式均采用析取范式的表示形式。对于资源 R (包括各种可用资源、服务及凭证、策略本身),其访问控制策略 P 表示为 $R \leftarrow \Phi(C_1, C_2, \dots, C_n)$ 。协商方提供资源当且仅当披露的凭证组合 C_1, C_2, \dots, C_n 使得 Φ 值为真,称 C_1, C_2, \dots, C_n 满足访问控制策略。所披露的凭证满足资源的访问控制策略时称该资源被解锁。

1.2 凭证的披露开销

本文引入凭证的披露开销表示凭证敏感度高和披露时给网络流量带来的影响大小。披露开销定义为一个大于零的整数,开销越高表示凭证的敏感度越强,对网络流量影响越大,拥有方越不愿意披露。例如,若信用卡号凭证的开销为 4,电话号码凭证的开销为 1,协商方将选择披露开销较小的电话号码凭证。受访问控制策略保护的凭证应考虑其敏感度和格式大小两方面,赋予合理的披露开销;无保护的凭证因凭证本身有大小,给网络将带来不同的通信流量,故根据其格式大小赋予不

收稿日期:2007-10-17;修回日期:2007-12-05。

基金项目:国家 973 规划项目(TG1999035801);国家自然科学基金资助项目(6053012)。

作者简介:姚慧(1983-),女,湖南岳阳人,硕士研究生,CCF 会员,主要研究方向:网络信息安全、信任协商;高承实(1973-),男,辽宁新民人,博士,主要研究方向:网络计算、网络信息安全;戴青(1963-),男,辽宁沈阳人,副教授,主要研究方向:网络技术;张徐(1985-),男,湖北荆州人,硕士研究生,主要研究方向:网络计算。

同的披露开销;若协商方不拥有该凭证,其开销记为 ∞ 。

2 基于动态规划的自动信任协商策略

在展开叙述之前,先对下文阐述的协商策略定义三个前提条件:

- 1) 为简化起见,访问控制策略未赋开销,但访问控制策略仍视为敏感资源的一种,并不向对方披露其内容;
- 2) 假设访问控制策略中不存在循环,即不存在如下的访问控制策略: $C_i \leftarrow C_j, C_j \leftarrow C_i$;
- 3) 假设双方初始协商状态至少有一方拥有一个无保护凭证,否则易得出协商必定失败的结论。

2.1 协商开销图

协商开销图是基于与/或图建模,为有向无环图,有且仅有一个入度为0的节点。节点指代凭证,入度为0的节点为协商方申请的资源 R ,终端节点是出度为0的节点,表示协商双方无保护的凭证。有向边由节点 v_i 指向 v_j 表示 v_j 为 v_i 的访问控制策略,称 v_i 是 v_j 的父节点, v_j 是 v_i 的子节点。节点 v 的出度若大于1,则 v 用连接符连接的子节点间为合取关系,不通过连接符连接的子节点间为析取关系。在连接节点与其父节点的有向边上标注表示凭证开销大小的数值,每条边的权重用 $w(v)$ 表示, $w(v)$ 的值为初始状态节点 v 的开销。图1给出了访问控制策略为 $R \leftarrow C_1 \vee (C_2 \wedge C_3) \vee C_4, S_2 \leftarrow C_4, S_3 \leftarrow C_2, C_1 \leftarrow S_3, C_2 \leftarrow S_1 \wedge S_2$ 的协商开销图。

信任协商的目的可表示为在协商开销图中找到一棵 R 为根节点、叶子节点为终端节点的生成树,且该生成树的披露开销在所有存在的此类生成树中最小,称该生成树为最小开销生成树。值得注意的是,协商开销图是分布式的,是协商双方在协商过程中根据收到的消息逐渐建立的,双方均只拥有该图的部分信息。信任协商由协商一方申请资源开始,最初双方仅发送终端节点表示的凭证和相应的开销。协商双方维护并更新协商开销图,在交互中逐渐解锁其他凭证并得出相应的最小开销,最终以最小开销解锁资源 R ,协商成功;否则,协商失败。

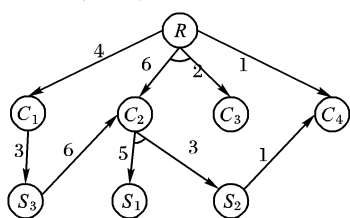


图1 协商开销图

2.2 策略基本思想

动态规划通常用于求解具有某种最优性质的问题,解得整体最优解。本文要解决的问题是生成使得协商开销最小的披露序列,运用动态规划的思想,多阶段决策最优化,找出使协商总开销最小的凭证披露序列。

协商双方均需维护己方和对方的两个已解锁凭证集合 $UnlockSet$ 。收到对方的已解锁凭证集,则搜索协商开销图,试图解锁己方未解锁的凭证或更新己方已解锁凭证的披露开销,直至所申请的资源 R 被解锁并生成 R 的最小开销生成树,协商成功。否则,协商失败。下面将给出策略的详细描述。

2.3 策略算法

策略算法中需要用到以下七种类型的消息和三个定义:

$Req(R)$:请求对方披露资源 R 。

$Success$:协商成功消息。

$Failure$:协商失败消息。

$NewlyUnlockSet$:最新已解锁凭证描述集合。包括最近一轮新解锁的凭证和最近一轮开销有更新的凭证,初始状态为协商双方无保护的凭证,表示为 $\{C_i(c(C_i)), C_j(c(C_j)), \dots, C_k(c(C_k))\}$,其中 C_i 为凭证描述, $c(C_i)$ 为凭证披露开销。 $NewlyUnlockSet_{local}$ 指己方最新已解锁凭证集, $NewlyUnlockSet_{remote}$ 指对方最新已解锁凭证集。 $NewlyUnlockSet$ 发送给对方后立即清空。

\emptyset :通知对方 $NewlyUnlockSet$ 为空的消息。

$SolvedDisclose$:最小开销生成树中对方的凭证描述,表示为 $\{C_i, C_j, \dots, C_k\}_{solved}$,其中 C_i 为凭证描述。

$FinalDisclose$:正式披露的凭证,表示为 $\{C_i, C_j, \dots, C_k\}$,其中 C_i 为实际的凭证。

定义1 $c(v)$ 为节点 v 的披露开销,设有向边 e 连接节点 v 与其父节点, e 的权重记为 $w(v)$ 。 $c(v)$ 的计算公式如下:

- 1) v 为终端节点,则 $c(v) = w(v)$;
- 2) v 为非终端节点,设 $\alpha_{i,j \in [1, l]}$ 为 v 的连接符, $v_{i+1}, v_{i+2}, \dots, v_{i+N}$ 为节点 v 不通过连接符相连的子节点,则 $c(v) = w(v) + \min\{c(\alpha_i), c(v_{i+1}), c(v_{i+2}), \dots, c(v_{i+N})\}$ 。其中, $c(\alpha_i)$ 由定义2计算。

定义2 $c(\alpha)$ 为连接符 α 的开销,令节点 v 通过连接符 α 连接子节点 v_1, v_2, \dots, v_n ,定义 $c(\alpha) = c(v_1) + c(v_2) + \dots + c(v_n)$ 。

定义3 节点 v 为非终端节点,设 $\alpha_{i,j \in [1, l]}$ 为 v 的连接符, $v_{i1}, v_{i2}, \dots, v_{iN}$ 为节点 v 通过连接符 α_i 相连的子节点, $v_{i+1}, v_{i+2}, \dots, v_{i+N}$ 为节点 v 不通过连接符相连的子节点。若 v 为最小开销生成树中的节点,则满足条件 $c(v) = w(v) + \min\{c(\alpha_i), c(v_{i+1}), c(v_{i+2}), \dots, c(v_{i+N})\}$ 的节点 v 相应的子节点也为最小开销生成树中的节点。

策略的具体算法描述如下:

Handle_Incoming_Mes()

msg = Receive_mes();

case(msg)

($Req(R)$): //当节点收到请求资源 R 的消息时

If $NewlyUnlockSet$ is null, send \emptyset ;

Else send $NewlyUnlockSet$;

($NewlyUnlockSet$): //当节点收到 $NewlyUnlockSet$ 消息时

For each $v_i (c(v_i)), i \in [1, N]$ in $NewlyUnlockSet$

//对于 $NewlyUnlockSet$ 集合中的每一凭证自底向上

//搜索与或图

search the local AND/OR graph upward:

If $v_j \notin NewlyUnlockSet_{local}$, 且 v_j 未被解锁

calculate $c(v_i)$; add $v_j (c(v_j))$ to

$NewlyUnlockSet_{local}$;

send $NewlyUnlockSet_{local}$;

If $v_j \in NewlyUnlockSet_{local}$, 且新计算出来的 $c(v_j)$ 与

$NewlyUnlockSet_{local}$ 中的不同

更新 $NewlyUnlockSet_{local}$ 中的 $c(v_j)$ 值;

send $NewlyUnlockSet_{local}$;

If $NewlyUnlockSet_{local}$ is null, send \emptyset ;

Else send $NewlyUnlockSet_{local}$;

(\emptyset):

If 节点未发送过 $NewlyUnlockSet$ 消息

send $NewlyUnlockSet_{local}$;

If $R \notin UnlockSet$, send $Failure$;

//资源 R 不在 $UnlockSet$ 中, 协商失败

Else send $SolvedDisclose$ of $\{v_i, v_j, \dots, v_k\}_{solved}$, 其中 v_i, v_j, \dots, v_k 据定义3自顶向下计算生成;

```

({vi, vj, ..., vk}solved): //当节点收到最小开销生成树中
//对方的凭证描述 SolvedDisclose 时
If vi, vj, ..., vk ∈ terminal nodes, send {vi, vj, ..., vk};
Else 依次对{vi, vj, ..., vk}solved中的每个节点运用定义 3
    自顶向下生成其相应的子节点 vl, vm, ..., vn
    send{vl, vm, ..., vn}solved;
({vi, vj, ..., vk}):
    //当节点收到正式披露的凭证消息 FinalDisclose 时
    If R ∈ vi, vj, ..., vk, send Success;
    Else 依次对{vi, vj, ..., vk}中的每个节点运用定义 3 自
        底向上生成其相应的子节点 vl, vm, ..., vn
        send{vl, vm, ..., vn};
(Success): //当节点收到协商成功消息时
    End the negotiation;
(Failure): //当节点收到协商失败消息时
    End the negotiation.

```

定理 1 基于动态规划的自动信任协商策略算法是可采纳的。

证明 首先证明如果理论上协商能够成功,则该算法一定会找到一棵生成树而结束。

分析算法,可以看到算法要么以 *Success* 消息结束,要么以 *Failure* 消息结束。只有在服务器方收到 \emptyset 消息时,查看本地 *NewlyUnlockSet* 为空,且请求的资源 R 不在己方的已解锁集合中,此时算法以 *Failure* 消息结束,即当且仅当 R 仍未解锁且对方拒绝披露可满足其解锁条件的任何凭证时协商才失败结束。其他可穷举的情况下,参照算法,可知协商均将合理运行直至成功。故若理论上协商能够成功,则算法一定会找到一棵生成树而结束。

其次证明该算法一定将找到最小开销生成树而结束。

设算法未能找到一最小开销生成树即找到一非最小开销生成树而结束,即 $v(R) \leq v'(R)$,其中 $v(R)$ 表示最小开销生成树中根节点 R 的最终披露开销, $v'(R)$ 表示非最小开销生成树中根节点 R 的最终披露开销。据算法详细描述可得出若算法在请求的资源 R 的披露开销不是最小时,应据定义 1 重新计算 R 的开销并取其小值进行更新。这与假设相矛盾。

由上可知基于动态规划的自动信任协商策略算法是可采纳的。

2.4 协商实例

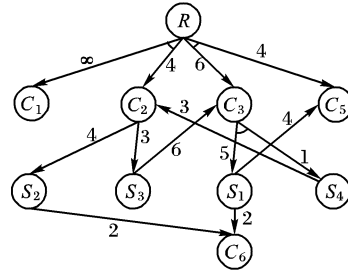
下面用实例说明采用基于动态规划的协商策略的协商过程。高性能计算中心 M 以免费和收费两种方式为用户提供网格计算资源 R 。若用户为高校 A 的学生 (C_1) 且为 CCF 会员 (C_2),则免费提供 R ;若用户为企业 S 的研究员 (C_5) 且拥有个人网银账户 (C_3),则以收费方式提供 R 。访问控制策略表示为 $R \leftarrow (C_1 \wedge C_2) \vee (C_3 \wedge C_5)$ 。张三为企业 S 的一名研究员,拥有无保护的凭证员工证 C_5 和银行账户 C_3 。张三要求对方必须为银联合作单位 (S_1) 和支付宝合作单位 (S_4) 才披露银行账户 C_3 ; $C_3 \leftarrow S_1 \wedge S_4$ 。 M 为 S_1 和 S_4 也制定了相应的保护策略: $S_1 \leftarrow C_5 \vee C_6$, $S_4 \leftarrow C_2$ 。张三对于 CCF 会员证制定访问控制策略 $C_2 \leftarrow S_2 \vee S_3$,要求对方为 CCF 合作伙伴 (S_2) 或高校 A 合作伙伴 (S_3)。对于 S_2 和 S_3 , M 也有相应的保护策略: $S_2 \leftarrow C_6$, $S_3 \leftarrow C_3$ 。双方对各自拥有的凭证均根据其敏感度和格式制定了初始披露开销。协商由张三向 M 请求使用网格计算资源 R 开始,协商的推进由协商代理调用 *Handle_Incoming_Message()* 进行。

图 2(a)列出了协商双方的访问控制策略和各自拥有凭证的初始披露开销,其中张三为 Client 方,高性能计算中心 M

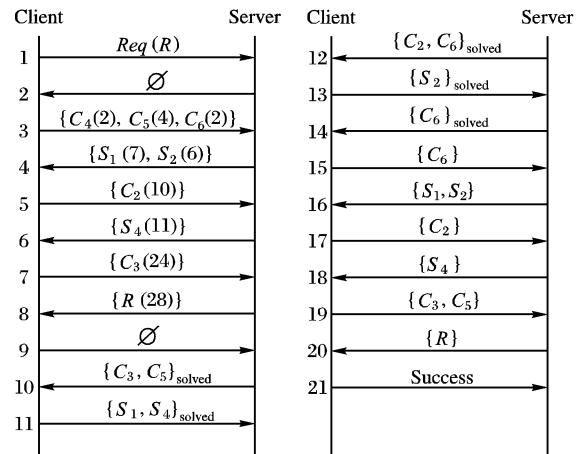
为 Server 方,张三非高校 A 的学生,不拥有学员证 C_1 ,故 C_1 的披露开销为 ∞ 。

Client		Server	
$C_2 \leftarrow S_2 \vee S_3$		$R \leftarrow (C_1 \wedge C_2) \vee (C_3 \wedge C_5)$	
$C_3 \leftarrow S_1 \vee S_4$		$S_1 \leftarrow C_5 \vee C_6$	
$C_4 \leftarrow \text{True}$		$S_2 \leftarrow C_6$	
$C_5 \leftarrow \text{True}$		$S_3 \leftarrow C_3$	
$C_6 \leftarrow \text{True}$		$S_4 \leftarrow C_2$	
Cred	S_1 S_2 S_3 S_4 C_2 C_3 C_4 C_5 C_6		
Weight	5 4 3 1 4 6 2 4 2		

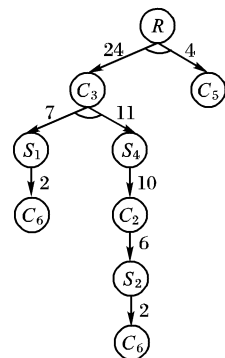
(a) 访问控制策略和初始披露开销



(b) 初始全局协商开销图



(c) 实际消息交换序列



(d) 全局最小开销生成树

图 2 运用基于动态规划的协商策略的一个协商示例

图 2(b) 是初始全局协商开销图,节点表示双方的凭证,有向边上的数值表示各节点的初始披露开销。图 2(c) 是协商过程中实际的消息交换序列。Server 收到 *Req(R)*,查看 *NewlyUnlockSet_{local}* 为空,则发送 \emptyset ,Client 的 *NewlyUnlockSet_{local}* 非空,故发送消息 *NewlyUnlockSet*: $\{C_4(2), C_5(4), C_6(2)\}$ 。Server 对收到的凭证依次自底向上搜索己方的协商开销图,得出 C_6 可解锁 S_1 和 S_2 ,据定义 1 算出更新的披露开销,在 4 处发送 $\{S_1(7), S_2(6)\}$ 。依次进行直至在 8 处解锁 R ,Client 收到 $\{R(28)\}$,无新的解锁凭证也无更新的披

露开销,故发送 \emptyset 。至此,双方可生成各自的最小开销生成树,图2(d)给出了全局的最小开销生成树。10到14处发送的消息是依据定义3自顶向下遍历最小开销生成树生成。15到20处为实际的凭证披露,则是依据定义3自底向上遍历最小开销生成树生成。21处发送 *Success*,协商成功结束,即高性能计算中心 *M* 将向张三以收费的方式提供网格计算资源 *R*。

3 性能分析

对于自动信任协商策略的性能分析一般从通信复杂度和时间复杂度两个方面进行。ATN 发生在网络中需要交互的两个实体之间,因此不可避免地给网络通信带来额外的负担;对于凭证的解锁、开销更新和最小开销生成树的遍历会加重本机处理器的运算量,将影响到本机其他进程的运行。

3.1 策略的通信复杂度

通信复杂度取决于消息的数量和消息的大小。策略包括七类的消息,主要考虑其中的 *NewlyUnlockSet*、*SolvedDisclose* 和 *FinalDisclose*,其余四类容易得出其通信复杂度为线性。对于 *NewlyUnlockSet*,在最坏的情况下,每个凭证均对应一条解锁的消息,其开销在每条访问控制策略下均更新一次,设访问控制策略数量为 *m*,则对应于 *n* 个凭证最多有 $O(nm)$ 条消息。设 *NewlyUnlockSet* 消息的大小为一常数,则 *NewlyUnlockSet* 的通信复杂度为 $O(nm)$ 。对于 *SolvedDisclose*,*n* 个凭证至多发送 *n* 条 *SolvedDisclose* 消息,消息内容为凭证描述,大小为常数,故通信复杂度不超过 $O(n)$ 。对于 *FinalDisclose*,消息的数量同 *SolvedDisclose*,但消息内容为真正的凭证,设每个凭证的大小不超过 $O(n)$,则复杂度为 $O(n^2)$ 。综上所述,在最坏的情况下,通信复杂度为 $O(mn + n^2)$,其中,若 $m \gg n$,则复杂度为 $O(mn)$;反之,若 $n \gg m$,则复杂度为 $O(n^2)$ 。

3.2 策略的计算复杂度

计算复杂度主要涉及到凭证的解锁、凭证披露开销的更新以及最小开销生成树的搜索。影响因素包括凭证的数量 *n* 和访问控制策略的数量 *m*。现考虑最复杂的情况,假设 *n* - 1 个凭证(至少有一个凭证除外,否则协商无法进行)均对应一条访问控制策略。在最坏的情况下,对每个凭证的解锁将遍历协商开销图的 *m* 个节点,凭证的披露开销的更新与解锁同时进行,故凭证的解锁和开销更新的计算复杂度不超过 $O(nm)$ 。最小开销生成树的节点数至多为 *n* 个,则生成 *SolvedDisclose* 和 *FinalDisclose* 消息时对最小开销生成树的搜

索的复杂度不超过 $O(n)$ 。因此基于动态规划的协商策略的计算复杂度为 $O(nm)$ 。

4 结语

本文在定义披露开销的基础上,基于协商开销图建模,运用动态规划的思想,设计了一种能找到最小开销凭证披露序列的协商策略。该策略区分了凭证的解锁和披露,使协商过程实际上分成凭证披露序列确立和实际凭证披露两个阶段,有以下两个优点:1) 凭证的先解锁后披露保证了协商过程的安全性;2) 协商过程的前阶段只发送凭证描述,减小交互的消息大小,降低通信量,后阶段才实际披露凭证,提高了协商效率。经证明,该策略为高效的,其通信复杂度为 $O(mn + n^2)$,计算复杂度为 $O(nm)$,其中 *n* 为凭证的数量,*m* 为访问控制策略的数量。

本文提出的基于动态规划的协商策略还存在一些不足:该策略虽然确保最小开销生成树生成的一定是协商总开销最小的凭证披露序列,但却不是凭证披露数目最少的披露序列,如何均衡最小开销和最少数目将是下一步的研究方向;其次,本文尚未考虑访问控制策略的披露开销,对于实际的协商,访问访问控制策略本身也具有披露开销,我们将进一步开展这方面的研究;另外,该策略虽然未将访问控制策略的内容向对方披露,但对协商过程进行深入分析仍能得出部分访问控制策略,加强策略的安全性也是进一步的研究方向。

参考文献:

- [1] WINSBOROUGH W H, SEAMONS K E, JONES V E. Automated trust negotiation[C]// DARPA Information Survivability Conference and Exposition. Washington D C, USA: IEEE Press, 2000: 88 - 102.
- [2] 李建欣, 怀进鹏, 李先贤. 自动信任协商研究[J]. 软件学报, 2006, 17(1): 124 - 133.
- [3] YU T, MA X, WINSLETT M. PRUNES: An efficient and complete strategy for trust negotiation over the Internet[C]// Proceedings of the 7th ACM Conference on Computer and Communications Security. New York: ACM Press, 2000: 210 - 219.
- [4] CHEN W, CLARKE L, KUROSE J, et al. Optimizing cost - sensitive trust-negotiation protocols[C]// INFOCOM 2005. Washington DC: IEEE Computer and Communications Society, 2005, 2: 1431 - 1442.
- [5] BELLMAN R E. Dynamic Programming[M]. Princeton, NJ: Princeton University Press, 1957.

(上接第 891 页)

反变换进行还原的研究。但是,由于变换的周期和图像的大小有关,当图像比较大的时候,利用周期性进行还原就显得不太现实。本文提出的逆变换算法不仅克服了这个问题,使得在图像被置乱任意多次后,能很快地进行恢复,而且对于更普遍的花托自同构映射同样可以进行操作,拓宽了它的应用范围。

参考文献:

- [1] DING WEI, XU QI-DONG. Digital image transformation and information hiding and disguising technology[J]. Chinese Journal of Computers, 1998, 21(9): 838 - 843.
- [2] XU QI-DONG. Matrix transformation and its application to image hiding[J]. Journal of North China University of Technology, 1999, 11(1): 24 - 28.
- [3] ZOU JIAN-CHENG, LI GUO-FU, XU QI-DONG Generalized gray

code and its application in the scrambling technology of digital images[J]. Applied Mathematics(A), A Journal of Chinese Universities, 2002, 17(3): 363 - 370.

- [4] 黎罗. Arnold 型置乱变换周期分析[J]. 中山大学学报: 自然科学版, 2005, 44(2): 1 - 4.
- [5] 李兵, 徐家伟. Arnold 变换的周期及其应用[J]. 中山大学学报: 自然科学版, 2004, 43(A02): 139 - 142.
- [6] 孔涛, 张晔. Arnold 反变换的一种新算法[J]. 软件学报, 2004, 15(10): 1558 - 1564.
- [7] VOYATZIS G, PITAS I. Applications of toral automorphisms in image watermarking[J]. IEEE International Conference on Image Processing. Washington DC: IEEE Computer Society, 1996, 2: 237 - 240.
- [8] 孙圣和, 陆哲明, 牛夏牧. 数字水印技术及应用[M]. 北京: 科学出版社, 2004.