(Im)possibility of Safe Exchange Mechanism Design

Tuomas Sandholm
Computer Science Department
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
sandholm@cs.cmu.edu

XiaoFeng Wang
Department of Electrical and Computer Engineering
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213
xiaofeng@andrew.cmu.edu

Abstract

Safe exchange is a key issue in multiagent systems, especially in electronic transactions where nondelivery is a major problem. In this paper we present a unified framework for modeling safe exchange mechanisms. It captures the disparate earlier approaches as well as new safe exchange mechanisms (e.g., reputation locking). Being an overarching framework, it also allows us to study what is *inherently possible and impossible in safe exchange*. We study this under different game-theoretic solution concepts, with and without a trusted third party, and with an offline third party that only gets involved if the exchange fails. The results vary based on the generality of the exchange setting, the existence (or creative construction) of special types of items to be exchanged, and the magnitude of transfer costs, defection costs, and escrow fees. Finally, we present an incentive-compatible negotiation protocol for selecting the best safe exchange mechanism when the agents do not know each others' costs for the different alternatives.

1. Introduction

Safe exchange is a key issue in multiagent systems, especially in electronic transactions. The rapid growth of Internet commerce has intensified this due to anonymous exchange parties, cheap pseudonyms, globality (different laws in different countries), etc. A recent study showed that 6% of people with online buying experience have reported nondelivery (NCL 1999). Software agents further exacerbate the problem due to the ability to vanish by killing their own processes. Without effective solutions, the safe exchange problem is one of the greatest obstacles to the further development of electronic commerce.

AI research has studied the problem using game-theoretic mechanism design. A safe exchange mechanism proposed in (Sandholm and Lesser 1995, Sandholm 1996) splits the exchange into small portions so that each agent benefits more by continuing the exchange than by vanishing. This idea has been operationalized in a safe exchange planner (Sandholm and Ferrandon 2000). The problem of cheap pseudonyms has been tackled by forcing new traders to pay an entry fee (Friedman and Resnick 1998), and schemes have been proposed for minimizing the needed entry fee (Matsubara and Yokoo 2000). Safe exchange was recently modeled as a dynamic game (Buttyan and Hubaux 2001).

Safe exchange has also been studied in computer security. These techniques include the coin ripping protocol (Jakobsson 1995) and zero knowledge proofs for exchanging signatures (Bao et al. 1998), which we will discuss later in this paper. Some of the techniques rely on a trusted third party (Deng et al. 1996).

This paper presents a unified exchange model that captures the previous disparate approaches (Section 2). We also present new safe exchange protocols. Most importantly, our model allows us to study *what is inherently possible and impossible to achieve in safe exchange*. We study this in the context of no trusted third party (Section 3), with a trusted third party (Section 4), and with an offline trusted third party that only gets involved if the exchange fails (Section 5). We ask whether safe exchange is possible under different game-theoretic solution concepts. We study this in the general case, with special types of items to be exchanged, and with various transfer costs, defection costs, and escrow fees. Finally, we present a negotiation mechanism for selecting among multiple safe exchange mechanisms (Section 6). To our knowledge, this is the first work to systematically investigate general exchange problems from the perspective of mechanism design.

2. Our exchange model

Our exchange setting has three parties: two strategic *agents*, $N=\{1,2\}$, that exchange *items* (for example, goods and payment), and a *trusted third party* (TTP) which facilitates the exchange. The TTP is not a strategic agent, and faithfully follows the exchange protocol. At any stage of the exchange, each party is in some *state*. The space of states of party i is denoted by S_i. $S=S_1\times S_2\times S_{TTP}$ is the space of *exchange states*.

The state $s_i^t = \langle P_i^t , A_i^t, \alpha_i^t, c_i^t \rangle \in S_i$ of agent i includes the following components:

- A *possession set* P_i^t which is the set of the items that the agent possesses. These items count toward the agent's utility. Intuitively, the agent can allocate these items to others.[1]

  However, we make the following key generalization which allows us to capture safe exchange mechanisms

[1] The model also captures duplicable items such as software. The duplication is considered to occur before the exchange, so at the start all copies are in the possession sets.

from the literature and new ones that would not fit a model based solely on possession sets. Specifically, sometimes a party may have the right to allocate an item even if it does not possess the item. For example, one can rip a $100 bill into two halves and give one of the halves to another party. Hence, neither party owns the money but can give it to the other. In order to describe an agent's control of such items that do not contribute to the agent's utility, we introduce the notion of an allocation set.

- An *allocation set* A_i^t which is the set of items that an agent does not possess, but can allocate to others (not to itself).
- An *activity* flag $\alpha_i \in \{\text{ACTIVE, INACTIVE}\}$ which defines whether the party is still *active*. This technicality is needed in order to have a well-defined setting (which requires that an *outcome state* is defined). An outcome state is an exchange state where no party is active anymore (it does not necessarily mean that the exchange is complete). An active party can take actions while an inactive one cannot. For example, an agent that vanishes in the middle of an exchange becomes inactive. Once an agent becomes inactive, it cannot become active again.
- A cost $c_i^t \geq 0$ which is the cumulative cost that the agent has incurred in the exchange so far (cost of sending items, defection penalties, etc).

Because the TTP can control some items during the exchange, it has an allocation set, but because it is not a strategic player and thus does not have a utility function, it has no possession set. The TTP itself does not have a cost, but again, in order to have well-defined outcomes, the TTP does have an activity flag. Put together, the state of the TTP is $s_i^t = \langle A_i^t, \alpha_i^t \rangle \in S_{TTP}$.

The set of outcome states is O. We say that in any outcome state, all allocation sets become empty because the parties are inactive and cannot allocate items anymore. An exchange starts from an *initial state* s^0, where the possession sets P_i^0 contain the items that are to be exchanged, and the allocation sets A_i^0 are empty. In any *complete state* $s^{complete} \in O$, the exchange is successfully completed: each agent possesses the items it was supposed to receive and lost only the items it was supposed to lose.

Each party can take *actions*. $M=M_1 \times M_2 \times M_{TTP}$ is the action space of the exchange. All three exchange parties have the following types of actions:

1. **Wait**: A party can wait for others to take actions.
2. **Transfer**: A party can transfer items from a *source* set (possession set or allocation set) to a set of *destination* sets. Each party i has a Boolean function $T_i(src, item, DES)$ to determine whether the agent can transfer *item* from set *src* to *all* sets in *DES* (this does not imply the possibility of transferring it to a subset of sets in *DES*). For example, if $item \in P_1$, then $T_1(P_1, item, \{P_1\})=1$. Some items (e.g., $100 bill) can be moved into allocation sets.
3. **Exit**: Agent i can deactivate itself by exiting at any state. Both agents and the TTP automatically exit if any complete state $s^{complete}$ is reached.[2]

Each action taken by an agent may incur a cost for that agent. The TTP also has an extra action type: it can *punish* an agent by adding to the agent's cumulative cost.

We assume that each agent $i \in N$ has quasi-linear preferences over states, so its utility function can be written as $u_i(s_i^t) = v_i(\lambda_1, \lambda_2, ..., \lambda_k) - c_i$, where v_i is agent i's valuation, λ_j is the quantity of item j the i possesses, and c_i is the cumulative cost defined above.

The tuple $E = \langle N, TTP, S, M, u_1, u_2 \rangle$ is called an *exchange environment*. An instance of the environment is an exchange. The mechanism designer operates in E to design an *exchange mechanism* $EM=\langle S', \rho, M', F, o \rangle$, where $S' \subseteq S$ and $M' \subseteq M$. The *player function* $\rho: S' \backslash O' \rightarrow N \cup \{\text{TTP}\}$ determines which party takes actions in a state. The space of *strategy profiles* is $F=F_1 \times F_2 \times F_{TTP}$. Agent i's strategy $f_i: S' \backslash O' \rightarrow M_i'$ specifies the action agent i will take in a state. The *outcome function* $o(s, f_1, f_2, f_{TTP}) \in O$ denotes the resulting outcome if parties follow strategies f_1, f_2, f_{TTP} starting from state s. Since the TTP is not a strategic player, we omit its static strategy from this function.

We denote an agent by i, and the other agent by $-i$. An exchange mechanism EM has a *dominant strategy equilibrium (DSE)* if and only if each agent $i \in N$ has a strategy f_i^* that is its best strategy no matter what strategy the other agent chooses. Formally, $u_i(o(s, f_i^*, f_{-i})) \geq u_i(o(s, f_i, f_{-i}))$ for all $f_i \in F_i$, $f_{-i} \in F_{-i}$ (the other agent's strategy) and $s \in S'$. The mechanism has a *subgame perfect Nash equilibrium* (SPNE) if and only if $u_i(o(s, f_i^*, f_{-i}^*)) \geq u_{-i}(o(s, f_i, f_{-i}^*))$ for all $f_i \in F_i$ and $s \in S'$. In either equilibrium concept, if the inequality is strict, the equilibrium is *strict*. Otherwise, it is *weak*.

An exchange can be represented as an *exchange graph* (Sandholm and Ferrandon 2000), where states are represented as vertices and actions as directed edges between states. Each edge has a *weight* indicating the cost of the move. An exchange mechanism is presented as a subgraph (see all figures[3]). A *path* from s^0 to any $s^{complete}$ is called a *completion path*. An exchange mechanism is a *safe exchange mechanism* (SEM) for an exchange in environment E if the mechanism has at least one equilibrium (f_1^*, f_2^*) in which the path of play is a completion path, that is, $o(s^0, f_1^*, f_2^*) = s^{complete}$. Such a path is called a *safe exchange path*. If the equilibrium is a (strict/weak) DSE, we say the *safe exchange is implemented in (strict/weak) DSE*. If the equilibrium is a (strict/weak) SPNE, we say the *safe exchange is implemented in (strict/weak) SPNE*.

We make the following assumptions:

---

[2] One concern is that an agent could postpone (indefinitely) without declaring exit. However, since we study exchanges for which we design well-defined exchange *protocols*, the parties can treat others' out-of-protocol actions as exit actions.

[3] When we illustrate mechanisms in this paper, for simplicity of drawing, we draw one vertex to represent all the states that have the same item allocation (but which may have different cumulated costs).

1. **Sequentiality.** *For any given state of the exchange, the SEM specifies exactly one agent that is supposed to make transfers. We make this sequentiality assumption for convenience only. Allowing for parallel actions does not affect safety because if an agent is safe in a parallel action, it would have to be safe even if the other party did not complete its portion of the parallel action.*
2. **Possessions close.** *If an agent possesses an item, the other agent has no possession of that item. There is no exogenous subsidy to the exchange. Formally, item* $\in P_i^t$ *at state* s^t *only if* $(item \in P_i^0 \cup P_{-i}^0) \wedge (item \notin P_{-i}^t)$.
3. **Exit rules.** *Both agents can exit at any state. Exiting is costless in any complete state* $s^{complete}$. *Exiting in any other state may subject the exiting agent to a cost (reputation loss, some chance of getting caught and financially penalized, etc.).*
4. **Nondecreasing utility.** *An agent's utility will not decrease from possessing additional items. Formally, if* $P_i^{t'} \subseteq P_i^t$, $u_i(s_i^{t'}) \le u_i(s_i^t)$ *given the same costs.*

An immediate result, which we will use in several places, is that during any exchange, no agent can take action to improve its own immediate utility:

Lemma 2.1. *It is impossible to have two states* s^t, s^{t+1} *of the exchange such that* $(i \in N) \wedge ((s^t s^{t+1}) \in M_i) \wedge (u_i(s^t) < u_i(s^{t+1}))$.

3. SEM design without a TTP

In this section, we study the possibility of SEM design in an exchange environment with no TTP.

3.1. Results for unrestricted items and costs

Here we derive *general* results for safe exchange without a TTP.[4] The results are *general* in that the exchange may contain any types of items and exit costs can be arbitrary.

Proposition 3.1. *Without a TTP, there exist exchanges that cannot be implemented safely in* (*even weak*) *DSE.*

Proof. Consider an exchange of an item k. Without loss of generality, let agent i make the first move from the initial state s^0. Denote the resulting state by s^1. Let (1) $v_i(\lambda_k) < v_i(\lambda_k')$ if $\lambda_k < \lambda_k'$, and (2) say item k cannot be transferred to any allocation set. The only possible move is to transfer some amount of item k to the other agent's possession set. Thus, from (1) above and Assumption 2, we get $u_i(s^0) > u_i(s^1)$. Let f_i^* be any strategy containing the above first move. Let there be free exit at the initial state s^0, and let f_i and f_{-i} be strategies that prescribe exit at s^0. Without a TTP, agent i becomes the only player once the other agent exits. According to Lemma 2.1, $u_i(o(s^0, f_i^*, f_{-i})) \le u_i(s^1) < u_i(s^0) = u_i(o(s^0, f_i, f_{-i}))$. Thus, f_i^* is not a dominant strategy. Since any safe exchange path should contain the first move, we have that it is impossible to implement the safe exchange in DSE. □

In fact, we can prove a stronger claim which would imply Proposition 3.1 (we nevertheless presented the proof of 3.1 because it is based on a different principle):

[4] The impossibility of a similar notion, *fair exchange*, has been studied in a non-game-theoretic framework (Pagnia and Gaertner 1999).

Proposition 3.2. *Without a TTP, there exist exchanges that cannot be implemented safely in* (*even* weak) *SPNE.*

Proof. Consider an exchange where properties (1) and (2) from the previous proof hold (at least for the last item to be delivered), and exit costs are zero. Consider any particular state s^t that precedes a complete state $s^{complete}$. Without loss of generality, let $i \in N$ make the last action to get to $s^{complete}$. By properties (1) and (2), $u_i(s^t) > u_i(s^{complete})$. Let $f_i^* \in F_i, f_{-i}^* \in F_{-i}$ be any strategy profile which forms a path from s^0 to s^t and includes the last move. Let f_i be a strategy identical to f_i^* except that "exit" is played at s^t. Since there is no defection cost, $u_i(o(s^t, f_i^*, f_{-i}^*)) = u_i(s^{complete}) < u_i(s^t) \le u_i(o(s^t, f_i, f_{-i}^*))$. Thus any strategy profile containing the last move is not a SPNE. However, a completion path has to include the last move. Therefore, it is impossible to implement the safe exchange in SPNE. □

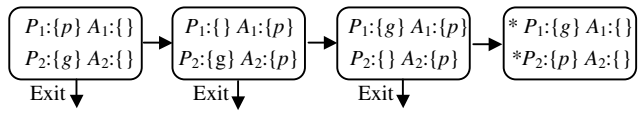
3.2. Results for exchanges including one-way items

The results above show that there are exchange settings where safe exchange is impossible without a TTP. In this section we study in more detail the conditions under which the impossibility holds. It turns out that the existence of *one-way items* to be exchanged plays a key role. A one-way item is an item the can be moved into allocation set(s). Recall that an agent that has an item in its allocation set cannot transfer the item to its own possession set. The following results highlight the importance of our introduction of allocation sets into the exchange model.

Definition 3.1. *An item k is a* ***one-way item*** *if there exists an agent i such that* $(i,j \in N) \wedge (A_j \in Y) \wedge (T_i(P_i, \lambda_k, Y) = 1)$.[5] *An item is a* ***one-way item*** *also if it is worthless to some agent (this is because the possession of such an item does not bring its owner any value while allocating it to others may increase their utility).*

The next two protocols enable safe exchange without a TTP by constructing one-way items in different ways.

Protocol 3.1. Coin ripping (Jakobsson 1995). This protocol uses a cryptographic digital coin which can be ripped into two halves. A single half has no value and once a half coin has been spent, it cannot be spent again.[6] The exchange proceeds as follows: 1) agent 1 rips a coin p and gives the first half coin to agent 2; 2) agent 2 deliveries the good g to agent 1; 3) agent 1 gives the other half of the coin to agent 2. In this protocol, the coin serves as a one-way item. In the figure, "*" denotes an inactive agent.



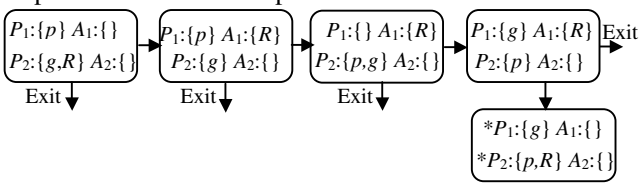
[5] An item may be in several agents' allocations sets simultaneously, and in some other agent's possession set.

[6] (Jakobsson 1995) proposed a scheme which allows a buyer to give a seller the hash value of a digital coin verifiable to a bank. The original coin cannot be spent (again) after the seller has given the hash value to the bank.

A weakness here is that agent 1 is indifferent between delivering the second half of the coin and not. Hence, the safe exchange is only a *weak* SPNE.

The coin ripping protocol requires a special cryptographic digital coin. Here, we introduce a new protocol which is applicable more broadly because it enables safe exchange even if money is not one of the items to be exchanged.

Protocol 3.2. Reputation locking. This protocol uses *reputation* as a one-way item! Suppose there is a public online reputation database. In our protocol, an agent's reputation record can be encrypted by other agents with the agent's permission, and only the agents that encrypted it can read/decrypt it. We call this *reputation locking* because the agent does not have an observable reputation while it is encrypted. The protocol proceeds as follows: 1) agent 2 permits agent 1 to lock its reputation R; 2) agent 1 gives agent 2 payment $p \le v_2(R)$; 3) agent 2 sends good g to agent 1; 4) agent 1 unlocks the reputation. The reputation R is a one-way item which can be moved into agent 1's allocation set. This protocol implements the safe exchange in SPNE. However, as in coin ripping, it is only a weak implementation because agent 1 is indifferent between cooperation and exit at step 4.



The two protocols above show that one can enable safe exchange by creatively constructing one-way items. (This is not always the case, for example, if the one-way items are too minor compared to the other items). Interestingly, creation of one-way items is the *only* approach that works in anonymous commerce where there is no trusted third party and no costs to premature exit from the exchange!

Proposition 3.3. *With zero exit costs and no TTP, an exchange can be implemented in weak SPNE only if there exists a one-way item.*

Proof. Suppose there exists an SEM for an exchange including no one-way items. By the definition of a one-way item, if an item is not a one-way item, it satisfies properties (1) and (2) from the proof of Proposition 3.1. Therefore, the proof of Proposition 3.2 applies. □

It turns out that the weakness of the coin ripping and reputation locking protocols is inevitable:

Proposition 3.4. *With zero exit costs and no TTP, no exchange can be implemented in strict SPNE (even with one-way items).*

Proof. If agent i exits at state s^t, (any particular state before a complete state) without a TTP, the other agent (say j) becomes the only player. According to Lemma 2.1, j cannot improve its own utility, so exiting becomes one of j's best response actions. In that case, agent i obtains its final utility already at s^t. Therefore, exiting at s^t becomes one of i's best response actions. Therefore, there exists a continuation equilibrium at s^t where both exit, and the exchange does not complete. □

3.3. Defection cost (cost of premature exit)

Proposition 3.4 showed that with free exit and no TTP, weak SPNE is the best one can achieve. Proposition 3.3 showed that even weak implementation requires the existence of one-way items. However, if there are costs to premature exit (defection cost) – such as loss of reputation, chance of getting caught and punished, loss of future business, etc. – then safe exchange can be achieved more broadly (Sandholm and Lesser 1995) (Sandholm 1996) (Sandholm and Ferrandon 2000). We model this by an exit cost $d_i(s^j)$ that may depend on the agent i and the exchange state s^j. We allow for the possibility that the exit cost is zero in some states (for example in the initial state in cases where participation in the exchange is voluntary).

Proposition 3.5. *Without a TTP, an exchange can be implemented safely in SPNE if and only if there exists a path $s^0 \ldots s^t s^T$ (where s^T is a complete state) such that $u_i(s^j) - u_i(s^T) \le d_i(s^j)$ for all $j \in [0,t]$. The exchange can be implemented safely in DSE if and only if $u_i(s^j) - \min_{q=j,j+1,\ldots,T} u_i(s^q) \le d_i(s^j)$ for all $j \in [0,t]$ on such a path. In either case, if each inequality is strict, the equilibrium is strict. Otherwise the equilibrium is weak.*

Proof. *If part for SPNE:* Construct an exchange mechanism as follows. The players should follow the completion path. If either agent deviates from the path, both agents exit, and the deviator has to pay the exit cost $d_i(s^j)$. For any agent i, let f_i^* be a strategy that follows the path and f_i be a strategy that defects at s^j. So, $u_i(o(s^j, f_i^*, f_{-i}^*)) = u_i(s^{complete}) \ge u_i(s^j) - d_i(s^j) = u_i(o(s^j, f_i, f_{-i}^*))$. Thus (f_i^*, f_{-i}^*) is a SPNE.

Only if part for SPNE: Let $s^0 s^1 \ldots s^t s^{complete}$ be any particular safe exchange path for an SPNE. Thus, $u_i(s^{complete}) = u_i(o(s^j, f_i^*, f_{-i}^*)) \ge u_i(o(s^j, f_i, f_{-i}^*)) = u_i(s^j) - d_i(s^j)$ for all states on the path and for both agents.

If part for DSE: If $u_i(s^j) - \min_{q=j,j+1,\ldots,T} u_i(s^q) \le d_i(s^j)$ *for all $j \in [0,t]$*, then agent i is better off by following the exchange no matter what the other agent does.

Only if part for DSE: If there exists a state s^j where the inequality does not hold for agent i, then if the other agent's strategy is to defect at state s^m satisfying $u_i(s^j) - d_i(s^j) > u_i(s^m)$, agent i is better off by exiting at s^j. Thus following the completion path is not a dominant strategy for i. □

4. SEM design with an online TTP

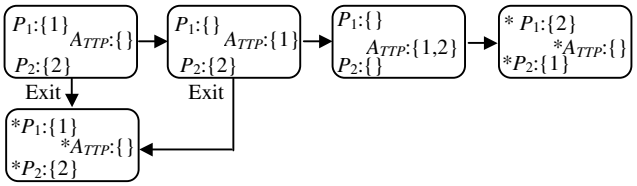
A simple way of achieving safe exchange is to use a TTP. A TTP facilitates exchange by helping agents allocate items and by punishing a defector. We assume that any item from either agent's possession set can be moved to the TTP's allocation set and vice versa. We also assume that

the TTP can observe the state of the exchange.

TTP-based safe exchange mechanisms have been explored in computer security (Buttyan and Hubaux 2001). Two types of TTPs have been proposed: *online TTPs* (Deng et al. 1996) and *offline TTPs* (Ba et al. 2000) (Bao et al. 1998) (Asokan et al. 1997). An online TTP is always involved in the exchange while an offline TTP only gets involved if a defection has occurred. We discuss online TTPs first.[7]

The existence of online TTP makes the safe exchange implementable in DSE:

Protocol 4.1. Online TTP-based SE. Each agent gives its items to be exchanged to the TTP. If both agents do this, the TTP swaps the items. Else the TTP returns the items.



If the online TTP requires an escrow fee (as most of the current ones do), we say that the escrow fee is paid before the exchange begins. With this understanding we have:

Proposition 4.1. *With an online TTP, if (1) each agent's utility of the complete state is greater than that of the initial state, and (2) for each state on the exchange path and for each agent, the agent's sum of action costs (for transfer actions and wait actions) from that state to the complete state is less than the agent's exit cost at that state, then Protocol 4.1 implements the exchange safely in strict DSE.* The proof is not hard, and we omit it due to limited space.

5. SEM design with an offline TTP

With no TTP, the safety of the exchange can usually be assured only in weak SPNE. With an online TTP, dominant strategy implementation is achievable, but the TTP is closely involved, incurs operating expenses, and thus usually charges an escrow fee even if the exchange completes without problems. A tradeoff between these two extremes is to use an *offline TTP* which does not participate in the exchange as long as it executes correctly, but gets involved if either agent exits prematurely. Offline TTPs have been practically implemented (such as ebay's feedback system) and theoretically investigated (Matsubara and Yokoo 2000) (Asokan et al. 1997).

5.1. General results

Here we investigate what can be achieved with an offline TTP when there are no limits on item types and exit costs. If the TTP does not have (and cannot obtain) allocation rights on the defector's items after defection, the TTP can do no more than punish the defector. This is equivalent to imposing an exit cost, so Proposition 3.5 suffices to characterize what is (im)possible in this case.

[7] (Ketchpel and Garcia-Molina 1996) studied, in a non-game-theoretic way, how different parts of an exchange should be sequenced when there are several online TTPs, but each TTP is only trusted by some subset of the parties.

So, what can be achieved with an offline TTP depends on how much penalty the TTP can impose on a defector. Punishing under different forms of information asymmetry is difficult (Friedman and Resnick 1998) (Matsubara and Yokoo 2000), for example due to cheap pseudonyms on the Internet, different laws in different countries, etc. Therefore, it is important to study what can be achieved when the TTP has too little power to punish defectors. That is what we address in the rest of this section.

5.2. Revocable and relinquishable items

When there is no reliable penalty for premature exit (defection cost is difficult to estimate or the TTP has inadequate power to punish), a TTP that has the ability to reallocate the defectors' possessions could facilitate safe exchange. Unfortunately, an offline TTP *only gets involved after the defection* at which time it has no control on any items (its allocation set is *empty*). In this case, the *active* agent (the defector is inactive) is the only one that can *give the TTP such reallocation rights on (some of) the defector's items*. This further requires that the active agent have control of the items. In the language of our exchange model, such items are in one agent's allocation set and the other agent's possession/allocation set at the same time. We now analyze such special items that an offline TTP can use to facilitate safe exchange.

We call an item *revocable* if its possessor can transfer it to the other agent's allocation or possession set while transferring it into its own allocation set (thus keeping the right to transfer the item from the other agent to the TTP). We call an item *relinquishable* if its possessor can keep it in the possession set while transferring it into the other agent's allocation set (thus giving the other agent the right to transfer the item from the former agent to the TTP).[8] Similar concepts have been discussed in the context of a particular exchange protocol for exchanging two items (Asokan et al. 1997).

Definition 5.1. *Denote by x_i the possession set or allocation set of agent i, and denote the other agent by –i. An item k is* ***revocable*** *to agent i if* $T_i(P_i, \lambda_k, \{A_i, x_{-i}\}) = 1$.[9]

(To handle the trivial case where an item is not of strictly positive value to its original possessor, we also call such items revocable.) *An item k is* ***relinquishable*** *if there exists an agent i such that* $T_i(P_i, \lambda_k, \{x_i, A_{-i}\}) = 1$.

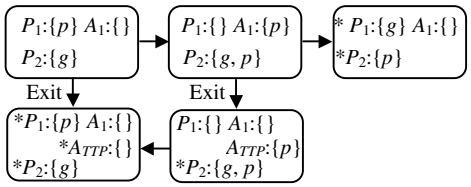
The following protocols use these types of special items.

Protocol 5.1. Credit card payment. A credit card payment can be viewed as a revocable item. Agent 1 pays agent 2 a payment p for good g with a credit card. At that point, $p \in A_1 \cap P_2$. If agent 2 does not deliver g, agent 1 sends a request to the offline TTP (credit card company). This corresponds to transferring p from A_1 to A_{TTP}. The company then revokes the payment (transfers p from A_{TTP}

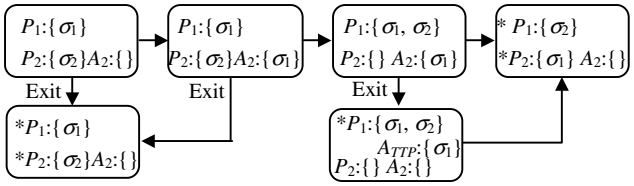
[8] Recall that by the definition of allocation set, the other agent can transfer the item to the TTP's allocation set or the former agent's allocation set, but not into its own possession set.

[9] Recall that agent *i* cannot give items that are in its allocation set into its possession set, and that agent *i* can give items in its allocation set to other parties, particularly the TTP's allocation set. Then the TTP can transfer the item to agent i's possession set.

and from P_2 to P_1). With zero action costs, the safe exchange path is followed in DSE.



Protocol 5.2. Escrowed signature (Bao et al. 1998). The protocol is for exchanging signatures on a contract. A digital signature can be converted into a relinquishable item. The protocol proceeds as follows: 1) agent 1 encrypts its digital signature (σ_1) with the public key of an offline TTP and then sends it along with a zero knowledge proof (Bao et al. 1998) to agent 2; 2) agent 2 checks that the data is an encrypted version of agent 1's signature, and gives its signature (σ_2) to agent 1; 3) agent 1 sends agent 2 σ_1. If agent 1 instead exits at step 3, agent 2 sends the data from step 2 to the TTP for decryption, and the TTP will give the decrypted signature of agent 1 to agent 2. With zero action costs, the safe exchange path is followed in DSE. [10]



It turns out that revocable or relinquishable items are in a sense *necessary* for safe exchange!

Proposition 5.1. *Let there be only an offline TTP and zero action costs (for transfer, wait, and exit actions). Let the items to be exchanged include no revocable or relinquishable items. Now, the exchange cannot be implemented in strict SPNE or even in weak DSE.*[11]

Proof. *Strict SPNE:* After a defection, the defector is inactive, and the offline TTP's allocation set is empty. So, the only way the TTP can affect a defector's possession set is if the active agent can put items that are in the defector's possession set into the TTP's allocation set (this requires the items to be in the active agent's allocation set).[12] By the definitions of revocable/relinquishable items, such a state can be reached only if revocable or relinquishable items exist. If, on the other hand, the TTP cannot affect the defector's possession set, then Proposition 3.4 applies.

Weak DSE: Suppose there exists a completion path implemented in weak DSE. By the assumption of sequential actions and the fact that eventually all items are exchanged, there has to be some state where one agent (say A) has transferred an item *I* into the other agent's (say B) possession set before receiving any items into its own possession set. At that state, because there are no revocable or relinquishable items, *I* cannot be in anyone's allocation set. If B now defects, A will have received nothing, and will have lost *I* which is of value to A. Therefore, A would have been strictly better off exiting in the initial state. Thus A's strategy of following the safe exchange is not a weak dominant strategy. Contradiction. □

5.3. Transfer costs and offline TTP's escrow fee

In many settings, especially when exchanging physical goods, there is a cost associated with each transfer action. Another type of cost that is associated with a transfer action is the fee that an agent has to pay an offline TTP when the agent asks the offline TTP to carry out a transfer action against a defector. (In the case of online TTPs, the escrow fee had no strategic effects because it had to be paid anyway, but in the offline TTP case it has strategic effects because it has to be paid only if the TTP's help is used).

Proposition 5.2. *With an offline TTP and* ***no relinquishable items****, no exchange can be safely implemented in weak DSE if the completion path contains any positive transfer cost.*

Proof. Suppose there exists a completion path implemented in weak DSE. By the assumption of sequential actions and the fact that eventually all items are exchanged, there has to be some state where one agent (say A) has transferred an item *i* into the other agent's (say B) possession set before receiving any items into its own possession set. At that state (say *s**), because there are no relinquishable items, A cannot control any of B's original items, but may be able to take back some of the items it gave to B. However, because there was a positive transfer cost, A would have been strictly better off exiting in the initial state. Thus A's strategy of following the safe exchange is not a weak dominant strategy. Contradiction. □

Proposition 5.3. *With a positive offline TTP fee and* ***no relinquishable items****, no exchange can be safely implemented in weak DSE.*

Proof. The proof is analogous to that of Proposition 5.2. □

Proposition 5.4. *With* ***no revocable items****, an exchange can be safely implemented in weak DSE only if for each agent i, the offline TTP's escrow fee plus i's sum of transfer costs on the completion path is at most* $u_i(s^{complete}) - u_i(s^0)$. The proof is not hard, and we omit it due to limited space.

6. Selecting a safe exchange mechanism

In the real world, different types of SEMs co-exist. For example, on the Internet, online TTPs such as TradeSafe exist, offline TTPs such as the Better Business Bureau exist, and obviously direct exchange is possible (and safe exchange planners for that exist (Sandholm and Ferrandon 2000)). Now, which SEM should the agents select? For a given exchange, different SEMs have different costs. Online TTPs have an escrow fee. Direct exchange and

[10] Recall we assume that the TTP can observe states so that an agent cannot get the signature decrypted without giving its own signature to the other. The protocol works even if the TTP does not observe states: in this case, each agent needs to give its own signature to the TTP (which will pass it to the other agent) to get the other's signature decrypted.

[11] As shown earlier in the paper, a weak SPNE can exist if there exists a one-way item.

[12] Recall that possessions close, so the items in the defector's possession set cannot be in the active agent's possession set. Also, by the definition of an allocation set, the active agent cannot move items from its allocation set to its own possession set.

offline TTPs may have various costs: some require agents to expose their fixed entities (e.g., credit based exchange) thus incurring privacy cost; some need intensive computation (e.g., escrowed signature); almost all of them expose the agents to risks (irrational play by the other party, accidents, etc.). Furthermore, agents may have different costs for a given SEM, and the agents' costs are generally only privately known by the agent.

We present a mechanism that will select the best SEM and motivates the agents to truthfully report their costs. We present it as choosing between an online TTP based SEM (TSEM) and another SEM (ASEM). We assume that 1) the online TTP's escrow fee c is commonly known and the agents have an agreement to share it in proportions d_1 and d_2 (where $d_1+d_2=1$), and 2) agents prefer exchange through either SEM to no exchange at all.

Protocol 6.1. SEM selection. Each agent reveals to the other which SEM it prefers. If both agents prefer the same SEM, that SEM is chosen. Otherwise, the agents resolve the conflict as follows: 1) each agent i transfers a payment c (the total amount of the escrow fee) and reveals its ASEM cost $\hat{c}_i$ to the online TTP. (If the other agent does not submit its payment and cost information, the TTP returns the former agent's payment.); 2a) If $\hat{c}_1 + \hat{c}_2 < c$, ASEM is chosen, the TTP returns a payment $c - \hat{c}_{-i} + d_{-i}c$ to the agent i who preferred ASEM, and returns the entire amount c to the other agent (who preferred TSEM). So, the TTP ends up keeping a nonnegative amount, which we consider its fee for resolving the SEM selection conflict. 2b) If $\hat{c}_1 + \hat{c}_2 \geq c$, TSEM is chosen, the TTP returns a payment $\hat{c}_{-i}$ to the agent i that preferred TSEM, and returns $d_i c$ to the other agent $-i$. At this point, the TTP has gotten paid the escrow fee plus a nonnegative conflict resolution fee.

Proposition 6.1. *Protocol 6.1 is ex post individually rational, weak DSE incentive compatible, and efficient (that is, the cheapest SEM is chosen).*

Proof. *Sketch.* The mechanism is an application of the Clarke tax voting scheme (Clark 1971), which has these properties. □

7. Conclusions and future research

Safe exchange is a key problem in multiagent systems, especially in electronic transactions. A large number of different approaches have been proposed for safe exchange. In this paper we presented a unified framework for modeling safe exchange mechanisms. Our framework captures the disparate earlier approaches, as well as new SEMs (e.g., reputation locking). Being an overarching framework, it also allowed us to study what is *inherently* possible and impossible in safe exchange. We showed what role special types of items play, and derived quantitative conditions on defection costs. The following table summarizes the qualitative results at a high level.

<table>
<tr><th></th><th>General results</th><th>Special items</th><th>With costs</th></tr>
<tr><td>No TTP</td><td>No weak SPNE.</td><td>No strict SPNE.
One-way item ← weak SPNE</td><td>Sufficient exit costs → weak/strict SPNE/DSE.</td></tr>
<tr><td rowspan="2">Offline TTP</td><td rowspan="2">Sufficient punishment → weak/strict SPNE/DSE.</td><td rowspan="2">(Revocable or relinquishable item)
← strict SPNE.</td><td>No relinquishable item: (transfer cost or escrow fee) → no weak DSE</td></tr>
<tr><td>No revocable item: weak DSE → (low escrow fee & low transfer cost)</td></tr>
<tr><td>Online TTP</td><td colspan="3">(Sufficient exit costs & low transfer costs) → strict DSE</td></tr>
</table>

Finally, we presented an incentive-compatible mechanism for selecting the best SEM when the agents do not know each others' costs for the different SEMs.

Future work includes extending the results to exchanges with more than 2 agents, and to settings where the agents and/or the TTP are uncertain about the exchange state.

Acknowledgements

We thank Kartik Hosanagar for illuminating discussions at the early stage of this work. We also thank Ramayya Krishnan and Pradeep Khosla for their encouragement. Sandholm is supported by NSF CAREER Award IRI-9703122, and NSF grants IIS-9800994, ITR IIS-0081246, and ITR IIS-0121678. Wang is supported by NSF grant IIS-0118767, the DARPA OASIS program, and the PASIS project at CMU.

References

Asokan, N; Schunter, M; and Waidner, M.1997. Optimistic protocols for fair exchange. *ACM Computer & Communication Security Conerence*. p. 7-17.

Ba, S; Whinston, A. B.; Zhang, H. 2000. The dynamics of the electronic market: an evolutionary game approach. *Information System Frontiers* 2:1, 31-40.

Clarke, E. 1971. Multi-part pricing of public goods. *Public Choice*, 11:17-33.

Bao, F; Deng, R; and Mao, W; 1998. Efficient and practical fair exchange protocols with off-line TTP. *IEEE symposium S&P*. p. 77-85.

Buttyan, L; Hubaux, J.P. 2000. Toward a formal model of fair exchange-a game theoretic approach. *Internaional workshop on ecommerce*.

Deng, R; Gong, L; Lazar, A; and Wang, W. 1996. Practical protocols for certified electronic mail. *Journal of Network & Systems Management* 4(3), 279-297.

Friedman, E; Resnick, P. 1998. The social cost of cheap pseudonyms. *Journal of Economics and Management Strategy* 10(2): 173-199.

Jakobsson, M. 1995. Ripping coins for a fair exchange. *EUROCRYPT*, p. 220-230.

Ketchpel, S. P; Garcia-Molina, H. 1996. Making Trust Explicit in Distributed Commerce Transactions. *International Conference on Distributed Computing Systems*, p. 270-281.

Matsubara, S; Yokoo, M. 2000. Defection-free exchange mechanism for information goods. *ICMAS*, p.183-190

National Consumers League. 1999. New NCL survey shows consumers are both excited and confused about shopping online, www.natconsumersleague.org/BeEWisepr.html.

Pagnia, H; Gaertner. F. 1999. On the impossibility of fair exchange without a trusted third party. Darmstadt University of Technology, Department of Computer Science technical report TUD-1999-02.

Sandholm, T. 1996. Negotiation among Self-Interested Computationally Limited Agents. PhD Thesis. UMass Amherst, Computer Science Dept.

Sandholm, T; Lesser, V. 1995. Equilibrium analysis of the possibilities of uneforced exchange in multiagent systems. *IJCAI*, p.694-701.

Sandholm,T; Ferrandon. V. 2000. Safe exchange planner. *ICMAS*. p.255-262.