Non-monotonic Properties for Proving Correctness in a Framework of Compositional Logic *

Koji Hasebe Mitsuhiro Okada

Department of Philosophy, Keio University
2-15-45 Mita, Minato-ku, Tokyo 108-8345, Japan
{hasebe,mitsu}@abelard.flet.keio.ac.jp

June 12, 2004

Abstract

Following up our previous work [9], we distinguish the monotonic properties and the non-monotonic ones in our inference system based on the framework of compositional logic, and give the way to include some non-monotonic properties. As an example, we present a correctness proof of Challenge Response protocol, and explain how such properties can be used in more powerful derivations. We also give a semantics based on the notion of trace, and present a soundness proof of our inference system including non-monotonic properties.

1 Introduction

Compositional logic (originally introduced by Durgin-Mitchell-Pavlovic [6] and Datta-Derek-Mitchell-Pavlovic [3, 4]) is an inference system based on Floyd-Hoare style logical framework for proving protocol correctness. By means of this framework, a protocol is considered as a program, and a statement "from a principal P's viewpoint, a general property φ holds at the end of his/her protocol action α" can be represented as a formula of the form $[\alpha]\varphi$ (or of the form $\theta[\alpha]\varphi$ in [3, 4]). One of the most advantageous points of this framework is its compositional approach for reasoning about a compound protocol: in order to prove a property about a compound protocol we can reuse already established properties about its components.

In our previous work [9], we presented a way for making more explicit the compositionality property of this framework by introducing a notion of *primitive actions in a role* (i.e., sending, receiving or generating actions). While in [6, 3, 4] an assumption about a principal's honesty is represented as implication of the form $Honest(Q) \supset \varphi$ (which means "if principal Q is honest, then φ holds"), in our framework such assumption (called *honesty assumption*) is represented by the predicates of the form $Honest(\vec{\alpha}^Q)$ (where $\vec{\alpha}^Q$ is a sequence of primitive actions in a role performed by Q) in the left hand side of a sequent style assertion. In a proving process of a property, these honesty assumptions are composed by combining the usual contraction rule of the sequent calculus and the following *weakening rule* (analogous to the weakening rule of the traditional logic).

$$\frac{Honest_Q(\vec{\alpha};\vec{\alpha}'),\Gamma \vdash [\vec{\beta};\vec{\beta}']\varphi}{Honest_Q(\vec{\alpha};\alpha'';\vec{\alpha}'),\Gamma \vdash [\vec{\beta};\beta'';\vec{\beta}']\varphi}\ \textbf{Weakening}$$

*This work was partly supported by Grants-in-Aid for Scientific Research of MEXT, Center of Excellence of MEXT on Humanity Sciences (Keio University), the Japan-US collaborative research program of JSPS-NSF, Oogata-kenkyuu-jyosei grant (Keio University) and Global Security Center grant (Keio University). The first author was also supported by Fellowship for Japanese Young Scientists from Japan Society for the Promotion of Science.

(This means that, "from P's view, if a property φ is derived from Γ with Q's honesty on $\vec{\alpha}; \vec{\alpha}'$ after P's performance of the sequence of actions $\vec{\beta}; \vec{\beta}'$, then φ is also derived from Γ with Q's honesty on $\vec{\alpha}; \alpha''; \vec{\alpha}'$ after P's performance of $\vec{\beta}; \beta''; \vec{\beta}'$, for any addition α'' and β'' in the roles".) In [9], we showed that this type of inferences is used for proving a property about a compound protocol, directly composing proofs of its components.

When we can freely apply the weakening rule to φ, we call φ a *monotonic* property[1]. Freshness, sending-fact, receiving-fact are examples of monotonic properties. In [9], we took as an example a set of monotonic properties, and demonstrated that such an inference system has enough power to prove *(non-injective) agreement property* (in the sense of Woo-Lam [10]) of some protocols, even if we do not use logical negation, nested implications, or any temporal operators as introduced in [3, 4][2].

However, if we want to prove a property stronger than the agreement property, we need some non-monotonic properties. In this paper, we give the way to include some non-monotonic properties in our framework of compositional logic. As an example, we aim at proving *matching conversations* of *Challenge Response Protocol* [5] which was also shown in [3, 4]. This property is stronger than the agreement property, because we need to prove additional properties about the ordering of actions performed by the different principals. To prove this property, we introduce a non-monotonic property "$firstly\_sends$. We show a proof of this property in this extended system only by adding a few restrictions on the weakening rules previously shown and on the inference rules on a principal's honesty (called *honesty inferences*). In particular, we do not use logical negation, nested implication and temporal operators, which are used in the original proof of [3, 4].

In this paper, we use the following notations (cf. Appendix A of [9]). The letters A, B, C,... (P, Q, R,..., resp.) are constants (variables , resp.) of principal's names. The capital letters $K, K', \ldots, K_1, K_2, \ldots$ and $N, N', \ldots, N_1, N_2, \ldots$ are constants of keys and of nonces, respectively, while the small letters $k, k', \ldots, k_1, k_2, \ldots$ and $n, n', \ldots, n_1, n_2, \ldots$ are variables of the same sorts as above. The letters $m, m', \ldots, m_1, m_2, \ldots$ are used to denote messages, and $\{m\}_K$ is the encryption of m with key K, and $\langle m_1, \ldots, m_n \rangle$ is the concatenation of messages $m_1, \ldots, m_n$. We also introduce $m \sqsubseteq m'$ to represent the subterm relation as a meta symbol.

The rest of this paper is organized as follows. In Chapter 2, we shall review the inference system introduced in [9]. In Chapter 3, we shall show how to include non-monotonic properties in the system. Moreover, as an example, we prove the matching conversations of CR protocol which cannot be proved only by the monotonic properties. In Chapter 4, we shall give a semantics based on the notion of trace, and sketch out a soundness proof of the extended system. In Chapter 5, we shall present our conclusions and some further issues.

2 Inference system

2.1 The Language

Predicates of our inference system are as follows: $P\ generates\ n$, $P\ receives\ m$ [3], $P\ sends\ m$, $PK(P,k)$, $P \overset{K}{\leftrightarrow} Q$, $fresh(n)$ and $t = t'$. While the first three predicates are called *action predicates (performed by P)*, the rest of them are called *non-action predicates*. The letters $\alpha, \beta, \gamma, \delta, \ldots, \alpha', \alpha'', \ldots, \alpha_1, \alpha_2, \ldots$ are used to denote action predicates (also $\alpha^P, \beta^P, \gamma^P, \delta^P, \ldots$ to denote action predicates performed by P) and $\theta, \theta', \ldots, \theta_1, \theta_2, \ldots$ are non-action predicates. All those predicates except for equality are chosen from the BAN logic predicates [1]. Equality

[1] This notion of monotonic property is essentially the same as *persistent* property in the sense of [6, 3, 4], except that the notion of monotonicity is related not only to weakening for protocol actions (described in the square bracket "[ ]") but also to weakening for honesty assumptions.

[2] The reason why we do not need logical negation nor nested implications (nor, disjunctions in the right hand side of a sequent) is that we restrict the honesty inferences. By this restriction, in our framework each sequent is expressed by a Horn-clause, however it is trade-off against some kinds of inferences on honesty. (See also Section 5.)

[3] We distinguish two kinds of "$receives$": the simple receiving and the receiving with decryptions. $P\ receives\ m(\{m'\}^*_K)$ means that "P receives a term m and decrypts the indicated subterm $\{m'\}^*_K$ of m. For a more formal description, instead of using $*$, we could introduce a new predicate $decrypts$ and describe it by $(P\ receives\ m) \wedge (P\ decrypts\ \{m'\}_K)$.

is used for explicit treatment of substitutions. As we have mentioned in Section 1, all those predicates except for $sends$ have *monotonic* properties (i.e., properties independent of the weakening rules for principal's actions and for honesty assumptions)[4].

As logical connectives, we introduce only usual conjunction (denoted by ",") and non-commutative conjunction (denoted by ";"). Our intention is to use non-commutative conjunction to represent a sequence of principals' actions, implicitly treated in [9]. While in [3, 4] some temporal operators are used to reason about the ordering of actions, we get rid of any temporal operators: the orderings are directly derived from the axioms, using inference rules presented in Appendix. We introduce the vector notation such as $\vec{\alpha}$ to denote a sequence (i.e., non-commutative conjunct) of action predicates. We also introduce some notions related to a sequence of action. We say $\alpha_i \in \vec{\beta}\ (= \beta_1; \cdots; \beta_n)$ if $\alpha_i = \beta_j$ for some $j = 1, \ldots, n$. For a sequence $\vec{\alpha} = \alpha_1; \ldots; \alpha_n$ and for $\alpha_i, \alpha_j \in \vec{\alpha}$, we denote $\alpha_i \leq_{\vec{\alpha}} \alpha_j$ if $i \leq j$. For $\vec{\alpha}$ and $\vec{\beta}\ (= \beta_1; \ldots; \beta_n)$, if $\alpha_i \in \vec{\beta}$ for all $\alpha_i \in \vec{\alpha}$ and if $\forall \alpha_i, \alpha_j \in \vec{\alpha}.(\alpha_i \leq_{\vec{\alpha}} \alpha_j \Rightarrow \alpha_i \leq_{\vec{\beta}} \alpha_j)$, we say $\vec{\beta}$ is an *extension* of $\vec{\alpha}$ and denote it by $\vec{\alpha} \subseteq \vec{\beta}$.

Our inference system uses a sequent calculus style assertion. The basic form of assertion is as follows (where Q_i may be Q_j in the list of $P, \ldots, Q$).

$$Honest(\vec{\alpha}^P), \ldots, Honest(\vec{\beta}^Q), \Delta \vdash [\vec{\gamma}]_A\ \varphi$$

Here each of $\vec{\alpha}^P, \ldots, \vec{\beta}^Q$ is a sequence of action predicates performed by $P, \ldots, Q$, respectively, which represents a part of his/her role, and $\vec{\gamma}$ is a sequence of concrete actions performed by A.[5] Each of the letters $\varphi, \varphi', \ldots, \varphi_1, \varphi_2, \ldots$ is a sequence (i.e., non-commutative conjunct) of action predicates, or a single non-action predicate. Δ is of the form $\varphi_1, \ldots, \varphi_n$. Each predicate of the form $Honest(\vec{\alpha}^P)$ represents "a principal P honestly follows a part of role $\vec{\alpha}$". We call it P's *honesty assumption*. To formalize such assumptions on honesty, in [6, 3, 4], they introduce the predicate $Honest(P)$ which means "principal P is honest". On the other hand, our intention is to make more explicit the compositionality of honesty assumptions: we separate each honest principal's role into his/her primitive actions, and construct a composed proof by using some basic natural logical rules. (The details of the composing process were presented in [9].)

If $\vec{\alpha}^P$ consists of a sequence of primitive actions $\alpha_1^P; \cdots; \alpha_n^P$, we can consider the predicate $Honest(\vec{\alpha}^P)$ as an abbreviation of $Honest(\alpha_1^P); \cdots; Honest(\alpha_n^P)$, which is a conjunct of non-commutative conjunction.

Therefore, the intuitive meaning of the sequent previously introduced is "if principals $P, \ldots, Q$ honestly follow the parts of their roles $\vec{\alpha}^P, \ldots, \vec{\beta}^Q$, respectively, and if some properties Δ hold, then after A performs a sequence of concrete actions $\vec{\gamma}$, φ holds from A's viewpoint". (Here $\vec{\gamma}$ may be empty. In such case we often use φ, instead of $[\,]\varphi$.)

Finally, we introduce the postfix notation $[\vec{P}, \vec{n}, \vec{k}]$ in order to denote the lists of principal names $\vec{P}$ (list of variables $P_1, \ldots, P_m$), and the lists of variables of nonces and session keys $\vec{n}, \vec{k}$ (as variables). Substitutions are represented in terms of this notation.

2.2 Axioms and inference rules

Our inference system consists of the following four classes **(I)**-**(IV)** of axioms and of inference rules. (The complete list is also presented in Appendix.)

(I) Logical inferences with equality: As logical inferences, we use some structural rules (I-1) (weakening, contraction, exchange rules of the left hand side, and cut rule) and equality inference rules (I-2), and substitution rules

[4] As we shall see in the explanation of the Matching rule of the honesty inferences below, predicate "$sends$" is monotonic w.r.t. the weakening for concrete actions, however it is non-monotonic w.r.t. the weakening for honesty assumptions. In other words, this predicate is non-monotonic in the sense of our terminology, however it is "persistent" in the sense of [6, 3, 4].

[5] Note that for describing a sequence of action, while compositional logic of [6, 3, 4] uses the *cord calculus*, we describe it by the predicates previously shown.

(I-3). These are chosen from the traditional first order logic with equality. We also introduce some inference rules for non-commutative conjunction (I-4).

(II) Action properties axioms: These are composed of the *axioms about actions* and the *axioms for relationship between properties* in the sense of [6]. Our proposed axioms are listed in (II-1) and (II-2), respectively. (Here, axioms including non-monotonic property $firstly\_sends$, which is introduced in Section 3, are marked with the symbol "†".) However, our framework does not depend on any specific set of axioms in this class.

(III) Honesty inferences:

For deriving Q's other actions from P' viewpoint, P may assume Q's honesty and may use his/her own knowledge about Q's role in the protocol. For example, if P knows that Q has sent the message m in a current run, and assumes that Q is honest, then P can derive Q's previous action, because Q should not have sent the message m if he/she has not already performed all the previous actions of his/her role. For formalizing such inferences, compositional logic in [6, 3, 4] uses a special inference (called *honesty rule*) for deriving a conclusion of the form $Honest(Q) \supset \varphi$. On the other hand, in our system, inferences on honesty are formalized by the three kinds of inference rules: *Substitution* (III-1), *Matching* (III-2) and *Deriving another action* (III-3). These are called *honesty inferences*. For example, the following inference rule is the Matching rule.

$$\frac{\Delta \vdash [\vec{\alpha}]_P\ \vec{\beta}; (Q\ sends\ m); \vec{\gamma}}{\Delta, Honest(Q\ sends\ m', m) \vdash [\vec{\alpha}]_P\ \vec{\beta}; (Q\ sends\ m'); \vec{\gamma}}\ \textbf{Hon(Match)}$$

(Here $\vec{\beta}$ and $\vec{\gamma}$ are non-commutative conjuncts of some action predicates, respectively, (where each of them may be empty), and $m \sqsubseteq m'$.)

The intended meaning of this inference rule is that "if P assumes that Q is honest and follows the sending action "$Q\ sends\ m'$", and if P knows that Q sends a message m containing m'", then we can conclude that "P knows that Q has sent m'". This inference holds whenever the additional condition is satisfied such that "Q's honesty assumption does not include any other sending action of a message which includes m as a subterm". This means that the formula $Q\ sends\ m'$ appearing in the lower sequent is non-monotonic. Thus, to keep our system monotonic, we restrict all applications of honesty inferences and of weakening rule for honesty assumptions (explained in the next item (IV)) so as to preserve this condition. More formally, we extend the language by introducing a new predicate $Honest(\alpha, m)$ (here $Honest(\alpha)$ previously defined can be regarded as a special case such that m is empty), and all applications of the honesty inferences and the weakening rule for honesty assumptions are restricted by the following condition (denoted by ($\sharp$)).

> ($\sharp$) Both predicates $Honest(Q\ sends\ m', m)$ and $Honest(Q\ sends\ m'')$ (with $m \sqsubseteq m''$) do not appear in the left hand side of the lower sequent.

(IV) Weakening rules for actions and for honesty assumptions: We now introduce the *weakening rules for honesty assumptions* and the *weakening rule for concrete actions*. All the applications of these weakening rules are restricted so as to satisfy the ($\sharp$) condition of Matching rule of honesty inferences.

If we introduce a non-monotonic predicate, as we shall explain in the next section, some additional condition should be necessary. In other words, our choice of predicates is one of the simplest formalism with respect to the weakening rules.

Finally we point out a limitation of our system. For a protocol including duplications of the same actions, we cannot distinguish one from another in our logic, because our logic does not explicitly deal with position during the run of a protocol. In this paper we consider only protocols which does not include any duplication.

3 Introducing a non-monotonic property in correctness proofs

In this section, we give the way to include some non-monotonic properties in our inference system presented in the previous section. In Section 3.1, as an example, we introduce a non-monotonic predicate "$firstly\_sends$" which is used to reason about some ordering of actions, and explain some additional restrictions on the honesty inferences and the weakening rules. In Section 3.2, we show a proof of *matching conversations* of *Challenge Response Protocol* [5]. This property was already proved in [3, 4], however logical negation, nested implications and temporal operators are not used in our proof.

3.1 Inferences for non-monotonic properties

In order to prove our aimed property, whereas the predicate $Fresh$ is used in [3, 4], we introduce a new predicate $firstly\_sends$ (also denoted by $fsends$ for readability). $P\ fsends\ (m, n)$ means "P sends a message m containing n as a subterm, and P does not send any other message m' containing n before the sending of m". Clearly, this predicate is non-monotonic, because if $\vdash [\vec{\alpha}; \alpha''; \vec{\alpha}']\ P\ fsends\ (m, n)$ holds, where α'' is P's sending of m, and if we insert another P's sending of m' with $n \sqsubseteq m'$ before α'', then this predicate becomes false under this weakened assumption (in square bracket "[ ]"). As this observation tells us, if we introduce a non-monotonic predicate in our framework, we must restrict all applications of the weakening rules (both for honesty assumptions and for actions) and all the honesty inferences ((III-1)-(III-3)) by the following additional conditions (denoted by ($\sharp\sharp$)).[6]

> ($\sharp\sharp$) For each sequence $\vec{\alpha}^P, \vec{\beta}^Q, \vec{\gamma}$ and for each $\vec{\delta}$ (which appears in Δ) in the lower sequent $Honest(\vec{\alpha}^P), \ldots, Honest(\vec{\beta}^Q), \Delta \vdash [\vec{\gamma}]\varphi$, it is no extension of a sequence of the form $(P\ sends\ m'); (P\ fsends\ (m, n))$ (with $n \sqsubseteq m, m'$) for any P.

Some non-monotonic properties are useful for reasoning about ordering of actions. Actually in the case of $fsends$, the order between different principals' actions can be derived by the additional inference rule and axiom as follows.

Firstly Sends:

$$\frac{Honest(\vec{\alpha}^P), \ldots, Honest(\vec{\beta}^Q), \Delta \vdash [\vec{\gamma}]\vec{\delta}; (R\ sends\ m); \vec{\delta}'}{Honest(\vec{\alpha}'^P), \ldots, Honest(\vec{\beta}'^Q), \Delta \vdash [\vec{\gamma}']\vec{\delta}; (R\ fsends\ (m, n)); \vec{\delta}'}$$

(Here R may be in the list of $P, \ldots, Q$, and $n \sqsubseteq m$, and each of $\vec{\alpha}'^P, \vec{\beta}'^Q$ and $\vec{\gamma}'$ is obtained from $\vec{\alpha}^P, \vec{\beta}^Q$ and $\vec{\gamma}$ by replacing all occurrences of $(R\ sends\ m)$ with $(R\ fsends\ (m, n))$.

Ordering of Actions:

$(P\ generates\ n), (P\ fsends\ (m, n)), \alpha \vdash (P\ fsends\ (m, n)); \alpha$
(where α is an action predicate of message m' with $n \sqsubseteq m'$.)

Firstly Sends is used to derive a $fsends$ predicate, and **Ordering of Actions** is used to derive an order of actions performed by different principals. (This is essentially the same as **AF3** presented in Table 5 of p.23 of [4].) For the same reason as the case of weakening rules and honesty inferences, we should restrict the application of **Firstly Sends** by the same condition ($\sharp\sharp$).

[6] Here we point out another way for keeping the weakening rule meaningful: introducing the logical negation and separating the weakening rules into two rules as follows.

$$\frac{\Gamma \vdash [\vec{\alpha}; \vec{\alpha}']P\ fsends\ (m, n)}{\Gamma \vdash [\vec{\alpha}; \alpha''; \vec{\alpha}']\varphi}\ \textbf{W(Act)}$$

Here if α'' is P's sending of message m' such that $n \sqsubseteq m'$, and $\vec{\alpha}'$ includes another P's sending of message m, then φ is $\neg P\ fsends\ (m, n)$, and if not, then φ is $P\ fsends\ (m, n)$. This way is essentially the same as [3, 4]. (The *Freshness Loss Axiom* in Table 4 of p.22 in [4] is the corresponding axiom.)

PA	$fresh(N_1) \vdash [\vec{\alpha}]A\ sends\ \langle a, b, N_1\rangle; A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(1)
(1),**NV1**	$fresh(N_1) \vdash [\vec{\alpha}]A\ sends\ \langle a, b, N_1\rangle; B\ sends\ m; A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(2)
(2),**Hon(M)**	$H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}]A\ sends\ \langle a, b, N_1\rangle; B\ sends\ \langle q, p, n_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle;$ $A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(3)
(3),**Hon(S)**	$H(\vec{\gamma}), H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}]p = A, q = B, n_1 = N_1, n_2 = N_2$	(4)
(3),(4),**Eq**	$H(\vec{\gamma}), H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}]A\ sends\ \langle a, b, N_1\rangle;$ $B\ sends\ \langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle; A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(5)
(4),**Hon(R)**,**Eq**	$H(\vec{\beta}); H(\vec{\gamma}), H(\vec{\gamma}), H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}]B\ receives\ \langle a, b, N_1\rangle;$ $B\ sends\ \langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle$	(6)
(1),**FS**	$fresh(N_1) \vdash [\vec{\alpha}']A\ fsends\ \langle a, b, N_1\rangle; A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(7)
(6),**FS**	$H(\vec{\beta}); H(\vec{\gamma}), H(\vec{\gamma}), H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}']B\ receives\ \langle a, b, N_1\rangle;$ $B\ fsends\ (\langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle, N_2)$	(8)
(7),(8),**OA**	$H(\vec{\beta}); H(\vec{\gamma}), H(\vec{\gamma}), H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}']A\ fsends\ \langle a, b, N_1\rangle; B\ receives\ \langle a, b, N_1\rangle$	(9)
(5),(8),**OA**	$H(\vec{\beta}); H(\vec{\gamma}), H(\vec{\gamma}), H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}']B\ fsends\ (\langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle, N_2);$ $A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(10)
(8),(9),(10), **;**	$H(\vec{\beta}); H(\vec{\gamma}), H(\vec{\gamma}), H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}']A\ fsends\ \langle a, b, N_1\rangle; B\ receives\ \langle a, b, N_1\rangle;$ $B\ fsends\ \langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle; A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(11)
Hon(W),**Cont** (11), **;**,	$H(\vec{\beta}); H(\vec{\gamma}'), fresh(N_1) \vdash [\vec{\alpha}']A\ fsends\ \langle a, b, N_1\rangle; B\ receives\ \langle a, b, N_1\rangle;$ $B\ fsends\ \langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle; A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$	(12)

Table 1: A's view at end of run following the initiator's role of Challenge-Response Protocol

3.2 An example of correctness proof

Table 1 is the full proof of matching conversations of CR protocol from initiator A's viewpoint. A proof of the same conclusion is presented in Table 10 of p.49 in [4]. CR protocol $\Pi[P, Q, n_1, n_2]$ described in an informal description is as follows.

1. $P \to Q$. $\langle p, q, n_1\rangle$
2. $Q \to P$. $\langle q, p, n_2, \{n_2, n_1, p\}_{K_Q^{-1}}\rangle$
3. $P \to Q$. $\langle p, q, \{n_1, n_2, q\}_{K_P^{-1}}\rangle$

In this table, for readability we use some abbreviations as follows. $\vec{\alpha}$ is a sequence $A\ sends\ \langle a, b, N_1\rangle$; $A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$; $A\ sends\ \langle a, b, \{N_1, N_2, b\}_{K_A^{-1}}\rangle$. $\vec{\alpha}'$ is the sequence obtained from $\vec{\alpha}$ by replacing the first action $A\ sends\ \langle a, b, N_1\rangle$ with $A\ fsends\ (\langle a, b, N_1\rangle, N_1)$. Each symbols $H(\vec{\beta})$, $H(\vec{\gamma})$ and $H(\vec{\gamma}')$ are abbreviations of $Honest(Q\ receives\ \langle p, q, n_1\rangle)$, $Honest(Q\ sends\ \langle q, p, n_2, \{n_2, n_1, p\}_{K_Q^{-1}}\rangle)$ and $Honest(Q\ fsends\ (\langle q, p, n_2, \{n_2, n_1, p\}_{K_Q^{-1}}\rangle, \{n_2, n_1, p\}_{K_Q^{-1}}))$, respectively. We also omit the predicates concerning information about keys: in this protocol, K_A and K_B are the public keys for A and B, respectively, and K_A^{-1} and K_B^{-1} are the secret part of these keys, respectively. Moreover, some predicates not related to the derived predicates at each line are omitted. (i.e., we implicitly use the contraction rules of non-commutative conjunction.) Finally, m on Line (2) is a message such that $\{N_2, N_1, a\}_{K_B^{-1}} \sqsubseteq m$.

First we would like to focus attention on the use of non-monotonic predicate *fsends*. On Line (6), the conclusion is the agreement property from A's viewpoint. Note that we do not use *fsends* to prove the agreement property. Therefore, if we want to prove the agreement property of this protocol, as we have also shown in our previous paper [9], we do not need to introduce any non-monotonic predicate.

The predicate $fsends$ is used to derive the orderings between A's action and B's one. Particularly, the order $B\ fsends\ \langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle;\ A\ receives\ \langle b, a, N_2, \{N_2, N_1, a\}^*_{K_B^{-1}}\rangle$ on Line (10) is derived by **Firstly Sends** and **Ordering of Actions**, and it cannot be derived without such non-monotonic notion. (The same property is derived on Line (10) in the example of [4] by means of the **AF3** axiom.) In this proof, $fsends$ is introduced on Line (7) by using the cut rule and the weakening rule for actions as follows: first, by applying the cut rule to (1) and **FSends** we obtain the sequent $fresh(N_1) \vdash [\,]A\ sends\ \langle a, b, N_1\rangle; A\ receives\ \langle b, a, \ldots\rangle$, and then by applying some weakening rules for actions we obtain (7). Here we point out that at the second step each application of the weakening rule is restricted by ($\sharp\sharp$) condition that "no predicate of the form $A\ sends\ m$ (with $N_1 \sqsubseteq m$) does not appear before $A\ fsends\ (\langle a, b, N_1\rangle, N_1)$ in the action operator (i.e., in the square bracket [ ])". However, in this case, we can obtain the sequent (7) by any order of applications of weakening rules for the components of this sequence. (Line (8) is similar to (7).)

By introducing the non-monotonic predicate, all weakening inferences for concrete actions and for honesty assumptions below the application of **Firstly Sends** on Line (7) and (8) are restricted by ($\sharp\sharp$) conditions that "A's sending of $\langle a, b, N_1\rangle$ (introduced at Line (7)) and B's sending of $\langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle$ (introduced at Line (8)) cannot be inserted before the corresponding actions". The information about orderings are necessary for proving our aimed property. Therefore, if we want to prove a weaker property such as the agreement property, as we have shown in our previous paper [9], we don't have to introduce such non-monotonic predicates. Note that in our system we do not use any temporal operators for deriving properties about the orders of principals' actions: our formalization only use the non-commutative conjunction.

Remark. In our proof, Line (3) is derived by honesty inference **Hon(Match)**. The intended meaning of this inference is "if B sends a message including $\{N_2, N_1, a\}_{K_B^{-1}}$, then B should send $\langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle$ under the assumption that B honestly follows $\vec{\gamma}$". we restrict applications of the weakening rules to satisfy the ($\sharp$) condition that "Q's sending action of a message including $\{N_2, N_1, a\}_{K_B^{-1}}$ does not appear in the honesty assumptions". Here we point out that we formalize this kind of inference by some restrictions instead of using logical negation. On the other hand, in the proof in [4], the same conclusion is derived by the honesty assumption that "if B does not freshly generate N_2 as a fresh value, then B should send $\langle b, a, N_2, \{N_2, N_1, a\}_{K_B^{-1}}\rangle$ under the assumption B is honest". This implication is obtained by using $\neg Fresh$.

4 Trace Semantics and Soundness of the System

In this section we give a semantics for our inference system. First we give a definition of the semantics (in Section 4.1), and next we give a sketch of soundness proof of our system (in Section 4.2).

4.1 Trace Semantics

Our semantics is based on the notion of *trace*, which is a sequence of *states*. A state is a multiset of *primitive states* of the form "principal P has information m", and denoted by $P(m), Q(m), \ldots$. We also introduce a special kind of primitive state "message m sent by P is currently transmitted through the network", and denoted by $Net(m, P)$. The notion of state is defined by the same way in *Multiset Rewriting System* [2], however we use it as semantic notion.

For preparing the definition of the semantics, here we introduce some notions and notations. $s_0, s_1, \ldots$ are used to denote states and $\mathtt{s}, \mathtt{s}', \ldots$ to denote sequences of states, namely traces. We also introduce some notions related to traces. The notions of membership relation (denoted by $s_i \in \mathtt{s}$), order relation on $\mathtt{s}$ (denoted by $\leq_{\mathtt{s}}$), and extension (denoted by $\mathtt{s} \subseteq \mathtt{s}'$) are defined by the same ways as those of sequences of actions. (See Section 2.1.) When s_i is the i-th element of $\mathtt{s}$, the number i is called the *position* of s_i in $\mathtt{s}$. We denote the number of occurrence of facts $P(m)$ in a state s_i by $\parallel s_i \parallel_{P(m)}$ (e.g. if $s_i = \{P(m), P(m), Q(m)\}$, then $\parallel s_i \parallel_{P(m)} = 2$).

$Key(P, s_i)$ is used to denote the set of key possessed by principal P at position s_i. For messages m, m' and a set of keys $\{k_1, \ldots, k_l\}$, "m is *accessible* in m' with keys $\{k_1, \ldots, k_l\}$" (denoted by $m \in_{\{k_1,\ldots,k_l\}} m'$) is the reflexive-transitive closure satisfying the following conditions: (i) $m_i \in_{\{k_1,\ldots,k_l\}} \langle m_1, \ldots, m_n \rangle$ for some $i = 1, \ldots, n$, (ii) $m \in_{\{k_1,\ldots,k_l\}} \{m\}_{k_j}$ for some $j = 1, \ldots, l$.

In this section, we assume that all states except for states of network are monotonously increasing for any trace. That is, we consider only traces where, once an information is possessed by a principal, it does not disappear in his/her memory.

By means of the notion of traces, truth conditions for predicates of our syntax is defined as follows. We denote the basic semantic relation "φ is true at state s_i in $\mathtt{s}$" by "$\models_{\langle \mathtt{s}, i \rangle} \varphi$".

Truth condition for predicates:

- $\models_{\langle \mathtt{s}, i \rangle} PK(P, k)$ iff $P(k'), KeyPair(k, k') \in s_i$ and $\forall X \neq P.(X(k') \notin s_i)$.
- $\models_{\langle \mathtt{s}, i \rangle} P \overset{k}{\leftrightarrow} Q$ iff $P(k), Q(k) \in s_i$ and $\forall X \neq P, Q.(X(k) \notin s_i)$.
- $\models_{\langle \mathtt{s}, i \rangle} t = t'$ (for any terms t and t') iff $s_i[t/x] = s_i[t'/x]$.
- $\models_{\langle \mathtt{s}, i \rangle} P \; sends \; m$ iff $P(m) \in s_{i-1}, Net(m, P) \notin s_{i-1}$ and $Net(m, P) \in s_i$.
- $\models_{\langle \mathtt{s}, i \rangle} P \; receives \; m(\{m_1\}^*_{k_1}, \ldots, \{m_n\}^*_{k_n})$ iff $\exists X.\ (Net(m, X) \in s_{i-1}$ and $Net(m, X) \notin s_i)$ and
  $\| \; s_{i-1} \; \|_{P(m)} + 1 = \| \; s_i \; \|_{P(m)}$, and
  $\{m_j\}_{k_j} \in_{Key(P, s_i)} m$ and
  $\| \; s_{i-1} \; \|_{P(m_j)} + 1 = \| \; s_i \; \|_{P(m_j)}$ for each $j = 1, \ldots, n$.
- $\models_{\langle \mathtt{s}, i \rangle} P \; generates \; m$ iff $P(m) \notin s_{i-1}$ and $P(m) \in s_i$.
- $\models_{\langle \mathtt{s}, i \rangle} fresh(m)$ iff $\exists X.(X(n) \notin s_{i-1}$ and $X(n) \in s_i)$ and $n \sqsubseteq m$.
- $\models_{\langle \mathtt{s}, i \rangle} P \; fsends \; (m, n)$ iff $\models_{\langle \mathtt{s}, i \rangle} P \; sends \; m$ and $n \sqsubseteq m$ and
  $\forall j < i. \forall m' \sqsupseteq m.(\not\models_{\langle \mathtt{s}, j \rangle} A \; sends \; m')$.
- $\models_{\langle \mathtt{s}, i \rangle} \alpha_1; \cdots; \alpha_n$ iff $\models_{\langle \mathtt{s}, i_1 \rangle} \alpha_1$ and $\cdots$ and $\models_{\langle \mathtt{s}, i_n \rangle} \alpha_n$,
  and $i_1 \leq \cdots \leq i_n \leq s_i$.

Next, the definition "φ is true for trace $\mathtt{s}$" (denoted by $\models_{\mathtt{s}} \varphi$) is as follows.

- $\models_{\mathtt{s}} \beta$ iff $\forall s_i \in \mathtt{s}.(\models_{\langle \mathtt{s}, i \rangle} \beta)$ (where $\beta = PK(P, k)$ or $P \overset{k}{\leftrightarrow} Q$, or $t = t'$.)
- $\models_{\mathtt{s}} fresh(m)$ iff $\exists s_i \in \mathtt{s}.(\models_{\langle \mathtt{s}, i \rangle} fresh(m))$.
- $\models_{\mathtt{s}} \alpha_1; \cdots; \alpha_n$ iff $\exists s_i \in \mathtt{s}.(\models_{\langle \mathtt{s}, i \rangle} \alpha_1; \cdots; \alpha_n)$ (where each α_i is an action predicate.)

We define that $\models_{\mathtt{s}} \Gamma$ iff "$\models_{\mathtt{s}} \vec{\alpha}$ and ... and $\models_{\mathtt{s}} \vec{\beta}$, and $\models_{\mathtt{s}} \theta_i$ for each $i = 1, \ldots, n$" (where $\Gamma = \vec{\alpha}, \ldots, \vec{\beta}, \theta_1, \ldots, \theta_n$). By the above definition, it is clear that for any φ except for $fsends$ (i.e., for the case that φ is monotonic), if $\models_{\mathtt{s}} \varphi$ then $\models_{\mathtt{s}'} \varphi$ for any $\mathtt{s} \subseteq \mathtt{s}'$.

In terms of the above definitions, we define that the basic form of assertion is true under $\mathtt{s}$, namely,

$$Honest(\alpha^P_1); \cdots; Honest(\alpha^P_n), \ldots Honest(\alpha^Q_1); \cdots; Honest(\alpha^Q_k), \ldots, \Gamma \models_{\mathtt{s}} [\vec{\alpha}]\varphi$$

holds if and only if the following is satisfied (where φ is a state predicate or a sequence of action predicates).

If C1 $\forall i \leq n. \forall i' < i.(\models_{\mathtt{s}} \alpha^P_i \Rightarrow \models_{\mathtt{s}} \alpha^P_{i'})$, and $\forall j \leq k. \forall j' < j.(\models_{\mathtt{s}} \alpha^Q_j \Rightarrow \models_{\mathtt{s}} \alpha^Q_{j'})$,
C2 $\exists \mathtt{s}'.(\mathtt{s} \subseteq \mathtt{s}' \wedge \forall i \leq n.(\models_{\mathtt{s}'} \alpha_i) \wedge \forall j \leq k.(\models_{\mathtt{s}'} \alpha_j))$,
C3 $\models_{\mathtt{s}} \Gamma$,
C4 $\models_{\mathtt{s}} \vec{\alpha}$,
then $\models_{\mathtt{s}} \varphi$.

Here for each honest predicate $Honest(\alpha^X_i)$, if it is of the form $Honest(\alpha^X_i, m^X_i)$ for $X = P, Q$ and for $i = 1, \ldots, n$ or $1, \ldots, k$ (i.e., m^X_i is not empty), then the following condition is also satisfied:

C5 $\forall m'.((m' \sqsupseteq m_i^X) \wedge (m' \neq m'') \wedge (\alpha_i^X = X\ sends\ m'') \Rightarrow \forall \mathtt{s}' \sqsupseteq \mathtt{s}.(\not\models_{\mathtt{s}'} X\ sends\ m'))$.

The reason why we need this additional condition is as follows: first recall that $Honest(\alpha_i^X, m_i^X)$ (where m_i^X is not empty term) means that "X honestly follows the sending action α_i^X (say, $X\ sends\ m''$) and he/she does not follow any other sending actions of the message m' including m_i^X". Therefore, to satisfy this restriction, we assume $X\ sends\ m''$ is false for any extension $\mathtt{s}'$ of $\mathtt{s}$.

If the above form of assertion is true for any trace $\mathtt{s}$, then this assertion is called *valid* and we omit the subscription $\mathtt{s}$.

Remind that in this paper we consider only protocols which do not include any duplication of primitive actions. We assume that all traces considered here are also restricted by the same condition. (Formally, for any state $\mathtt{s}$ and for any action predicate α, if $\models_{\langle \mathtt{s},i \rangle} \alpha$ and $\models_{\langle \mathtt{s},j \rangle} \alpha$ then $i = j$.)

4.2 Soundness of the System

In this subsection we show a sketch of a soundness proof of our system. In our previous paper [9], we presented a soundness proof for the system including only monotonic predicates. Then, here we only consider some of cases related to the non-monotonic predicate $fsends$.

Nonce verification 2: (where $\{m_1\}_K \sqsubseteq m_2, m_3$, and $\{m_1\}_K \not\sqsubseteq m_5$, and $n \sqsubseteq m_1, m_4, m_5$.)

$(PK(K,Q)), (P\ fsends\ (m_2,n)), (P\ generates\ n), (P\ receives\ m_5)$
$\vdash (P\ fsends\ (m_2,n)); (Q\ receives\ m_3(\{m_1\}_K^*)); (Q\ sends\ m_4); (P\ receives\ m_5)$

Assume that all the predicates appearing in the left hand side are valid. That is, for any $\mathtt{s}$ $(= s_1, \ldots, s_n)$, (i) $\forall j < n. \forall X \neq Q.\ ((Q(K^{-1}) \in s_j) \wedge (X(K^{-1}) \notin s_j))$, (ii) $\exists s_{i_1}.((\models_{\langle \mathtt{s},i_1 \rangle} P\ sends\ m_2) \wedge (\forall j' < i_1.(\not\models_{\langle \mathtt{s},j' \rangle} P\ sends\ m'$ with $n \sqsubseteq m')))$, (iii) $\exists i_2 < i_1.(\models_{\langle \mathtt{s},i_2 \rangle} P\ generates\ n)$, (iv) $\exists i_3 > i_1.(\models_{\langle \mathtt{s},i_3 \rangle} P\ receives\ m_5$ with $n \sqsubseteq m_5$, $\{m_1\}_K \not\sqsubseteq m_5)$. From (ii), (iii) and (iv), $\exists X \neq P. \exists i_4 < i_3.(\models_{\langle \mathtt{s},i_4 \rangle} X\ sends\ m_5)$ holds, and then $\exists Y \neq P. \exists i_5 < i_4.((Y(n) \in s_5) \wedge (\forall Z \neq P, Y.(Z(n) \notin i_5)))$. Then by (i) and (ii), $\exists l.((i_1 < l < i_5) \wedge (\models_{\langle \mathtt{s},l \rangle} Q\ receives\ m_3(\{m_1\}_K^*)))$ and $\exists l'.((l < l') \wedge (\models_{\langle \mathtt{s},l' \rangle} Q\ sends\ m_5)$ with $n \sqsubseteq m_4)$. This is the truth condition for $(Q\ receives\ m_3(\{m_1\}_K^*)); (Q\ sends\ m_4)$, and therefore, the sequence of actions in the right hand side of the sequent is true.

Firstly Sends:

(Without loss of generality, here we only consider a special case such that only P's honesty assumptions appear in the left hand side of each sequent and omit any context for readability.)

$$\frac{Honest(\alpha_1^P; \ldots; \alpha_n^P) \vdash [\vec{\gamma}]\vec{\delta}; (P\ sends\ m); \vec{\delta}'}{Honest(\alpha_1'^P; \ldots; \alpha_n'^P) \vdash [\vec{\gamma}]\vec{\delta}; (P\ fsends\ (m,n)); \vec{\delta}'}$$

(Here $n \sqsubseteq m$, and for each $\alpha_i'^P$, $\alpha_i'^P = P\ fsends\ (m,n)$ if $\alpha_i^P = P\ sends\ m$, otherwise $\alpha_i'^P = \alpha_i^P$.)

Here it is clear that the soundness holds when $\alpha_i'^P = \alpha_i^P$ for all $i < n$ (i.e., $\vec{\alpha}^P$ does not contain $P\ sends\ m$). Then, from now we shall consider only the case that $\alpha_i^P = P\ sends\ m$ and $\alpha_i'^P = P\ fsends\ (m,n)$ for some $i < n$. (In this case, by the condition ($\sharp\sharp$), $\vec{\delta}$ does not include $\delta_j = P\ sends\ m'$ with $n \sqsubseteq m'$ for all $j < i$.)

First, consider a trace $\mathtt{s}$ satisfying the conditions C1 and C2 (previously shown in the definition of truth condition for the sequent) for $Honest(\alpha_1'^P; \cdots; \alpha_n'^P)$. By the definition of the truth condition for $fsends$, $\mathtt{s}$ also satisfies the conditions C1 and C2 for $Honest(\alpha_1^P; \cdots; \alpha_n^P)$". Here we assume that the upper sequent is valid, then $\models_{\mathtt{s}} \vec{\delta}; P\ sends\ m; \vec{\delta}'$ holds. Moreover, if we assume that $\mathtt{s}$ satisfies C1 and C2 for $Honest(\alpha_1'^P; \cdots; \alpha_n'^P)$, then by C1, the following holds that "if $\models_{\mathtt{s}} \alpha_i'$ $(= P\ fsends\ (m',n))$, then $\forall j < i.(\not\models P\ sends\ m')$ with $n \sqsubseteq m'$ and $m' \neq m$". Therefore, $\forall j < i.\ \not\models_{\langle \mathtt{s},j \rangle} P\ sends\ m''$ for any m'' with $m \sqsubseteq m''$. This is the truth condition for $\models_{\mathtt{s}} \vec{\delta}; P\ sends\ m$. Therefore the right hand side of the lower sequent is valid.

Weakening rules:

$$\frac{\Gamma, Honest(\vec{\alpha}^P; \vec{\alpha}'^P) \vdash [\vec{\beta}]\varphi}{\Gamma, Honest(\vec{\alpha}^P; \alpha''^P; \vec{\alpha}'^P) \vdash [\vec{\beta}]\varphi}\ \textbf{W(Hon)} \qquad \frac{\Gamma \vdash [\vec{\alpha};^P \vec{\alpha}'^P]\varphi}{\Gamma \vdash [\vec{\alpha}^P; \alpha''^P; \vec{\alpha}'^P]\varphi}\ \textbf{W(Act)}$$

(Here we only consider the case that α''^P is $P\ sends\ m$. It is similar way to prove the case that α''^P is $P\ fsends\ (m, n)$.)

(1) Weakening (Honesty):

Here we assume that the lower sequent satisfies the ($\sharp\sharp$) condition. That is, $\vec{\alpha}'^P$ does not include any action of the form $P\ fsends\ (m', n)$ with $n \sqsubseteq m'$. It is sufficient to show that for any trace $\mathtt{s}$, "if $\mathtt{s}$ satisfies the conditions C1 and C2 for $Honest(\vec{\alpha}^P; \alpha''^P; \vec{\alpha}'^P)$, then $\mathtt{s}$ also satisfies the condition C1 and C2 for $Honest(\vec{\alpha}^P; \vec{\alpha}^P)$", however it immediately follows from the definition of the truth condition of $fsends$ and the ($\sharp\sharp$) condition.

(2) Weakening (Actions):

By ($\sharp\sharp$) condition, we can assume that $\vec{\alpha}'^P$ does not include $P\ fsends\ (m', n)$ with $n \sqsubseteq m, m'$. It is sufficient to show that for any trace $\mathtt{s}$, $\models_{\mathtt{s}} \vec{\alpha}^P; (P\ sends\ m); \vec{\alpha}'^P$ then $\models_{\mathtt{s}} \vec{\alpha}^P; \vec{\alpha}'^P$, however, this immediately follows from the definition of $\models_{\mathtt{s}} \vec{\alpha}^P; (P\ sends\ m); \vec{\alpha}'^P$.

Remark. As for the weakening rule for honesty assumptions, if the lower sequent violate the ($\sharp\sharp$) condition (i.e., if $\vec{\alpha}'^P$ includes action predicate of the form $P\ fsends\ (m', n)$ with $n \sqsubseteq m'$), then there exists a trace such that it does not satisfy the conditions for $Honest(\vec{\alpha}^P; \alpha''^P; \vec{\alpha}'^P)$ whereas it satisfies $Honest(\vec{\alpha}^P; \vec{\alpha}'^P)$. The same is true of the other weakening rule.

5 Conclusions and future work

By the distinction between monotonic and non-monotonic predicates, we gave the way to include non-monotonic properties, and showed that they can be used in more powerful derivation to prove correctness properties of a protocol. As an example, we proved the matching conversations of CR protocol, where the ordering of actions performed by different principals are derived by **Firstly Sends** and **Ordering of Actions**. Other examples are **Nonce Verification** 2 and 3 presented in (II-2) of Appendix. These are formalizations of the notion of *Outgoing test* in the *Authentication tests* based *Strand space method* (cf. Guttman-Fábrega [7]). This notion can be formalized only by using non-monotonic property $fsends$ or similar one, which can represent the same notion of *uniquely originates* (in [7]).

In our extended system, we did not use logical negation, nested implications and any temporal operators to prove our aimed property, which were used in [3, 4]. (In other words, in our system each sequent is the form of Horn-clause.) This simplification is realized by the restriction on the honesty inferences. However, this restriction is a trade-off. For example, the following kind of inferences cannot be expressed in our system: assume that a principal (say P) is honest following a role $\vec{\alpha} = \alpha_1; \alpha_2; \alpha_3$ of a protocol. Then from this assumption, we can conclude that "$Honest(P) \wedge (P\ \text{performs}\ \beta) \supset (\beta = \alpha_1) \vee (\beta = \alpha_2) \vee (\beta = \alpha_3)$" (i.e., "if P is honest and he/she performs an primitive action β then it is α_1 or $alpha_2$ or α_3"), because honest principal does not perform any other actions than the actions defined by his/her role. One of our next aims is to investigate the formalization of such inferences and clarify what kinds of properties (useful for a correctness proof) become provable in such extended system.

We also gave a semantics based on the notion of trace and show a sketch of soundness proof. This direction should make a contribution to our further target, namely, automated generation of correctness proofs or correct protocols for example.

Acknowledgments

We would like to express our sincere thanks to Drs. Andre Scedrov and Iliano Cervesato for their invaluable comments and discussions. We also would like to express our sincere thanks to Mr. Lam Ta-Minh for his helpful

comments. Finally, helpful comments from the anonymous reviewers results in improvements to this paper.

References

[1] M. Burrows, M. Abadi and R. Needham. A Logic of Authentication. *Technical Report 39*, Digital System Research Center, 1989.

[2] I. Cervesato, N.A. Durgin, P.D. Lincoln, J.C. Mitchell and A. Scedrov. Multiset rewriting and the complexity of bounded security protocols. *Journal of Computer Security*, vol.12, no.1, pp.677-622, 2004.

[3] A. Datta, A. Derek, J. C. Mitchell and D. Pavlovic. Secure Protocol Composition. *Proceedings of 19th Annual Conference on Mathematical Foundations of Programming Semantics, ENTCS Vol. 83*, 2004.

[4] A. Datta, A. Derek, J. C. Mitchell and D. Pavlovic. A Derivation System for Security Protocols and its Logical Formalization. *Journal of Computer Security, Special Issue of Selected Papers from CSFW-16*, 2004.

[5] W. Diffie, P. C. van Oorschot and M. J. Wiener. Authentication and authenticated key exchange *Designs, Codes and Cryptography*, vol.2, pp.107-125, 1992.

[6] N. Durgin, J. Mitchell and D. Pavlovic. A Compositional Logic for Protocol Correctness. *Journal of Computer Security (Special Issue of Selected Papers from CSFW-14)*, 11(4), pp.677-721, 2003.

[7] J. D. Guttman and F. J. T. Fábrega. Authentication tests and the structure of bundles. *Theoretical Computer Science*, vol. 283(2), pp.333-380, 2002.

[8] K. Hasebe and M. Okada. A Logical Verification Method for Security Protocols Based on Linear Logic and BAN Logic. M. Okada, B. Pierce, A. Scedrov, H. Tokuda and A. Yonezawa (eds.), *Software Security — Theories and Systems*, Lecture Notes in Computer Science, Hot Topics, vol.2609, Springer-Verlag, pp.417-440, 2003.

[9] K. Hasebe and M. Okada. Inferences on Honesty in Compositional Logic for Security Analysis. to appear in *Proceedings of the International Symposium on Software Security (ISSS2003)*, Springer LNCS, 2004.

[10] T. Y. C. Woo and S. S. Lam. Verifying authentication protocols: Methodology and example. *Proceedings of the International Conference on Network Protocols*, 1993.

Appendix: axioms and inference rules of the system

(I) Logical inference rules

(1) Structural rules: weakening, contraction and exchange rules in the left hand side, and cut rule (**Cut**) (2) Inference rules for equality (**Eq**) (a typical rule which we often use is presented below), (3) Substitution rule (**Subst**), (4) Inference rules for non-commutative conjunction (**;**): Concatenation, Weakening and Contraction.

$$\frac{\Gamma \vdash [\vec{\alpha}]\varphi \quad \varphi, \Delta \vdash [\vec{\alpha}]\varphi'}{\Gamma, \Delta \vdash [\vec{\alpha}]\psi}\ \textbf{Cut} \qquad \frac{\Gamma \vdash [\vec{\alpha}]x = t \quad \Delta \vdash [\vec{\alpha}]\varphi[x]}{\Gamma, \Delta \vdash [\vec{\alpha}]\varphi[t/x]}\ \textbf{Eq} \qquad \frac{\Gamma[x] \vdash [\vec{\alpha}[x]]\varphi[x]}{\Gamma[t/x] \vdash [\vec{\alpha}[t/x]]\varphi[t/x]}\ \textbf{Subst}$$

$$\frac{\Gamma \vdash [\vec{\alpha}]\vec{\beta}; \beta'' \quad \Delta \vdash [\vec{\alpha}']\beta''; \vec{\beta}'}{\Gamma, \Delta \vdash [\vec{\alpha}; \vec{\alpha}']\vec{\beta}; \beta''; \vec{\beta}'}\ \textbf{Concat-;} \qquad \frac{\vec{\beta}; \vec{\beta}', \Gamma \vdash [\vec{\alpha}]\varphi}{\vec{\beta}; \beta''\vec{\beta}', \Gamma \vdash [\vec{\alpha}]\varphi}\ \textbf{Weak-;} \qquad \frac{\Gamma \vdash [\vec{\alpha}]\vec{\beta}; \beta''; \vec{\beta}'}{\Gamma \vdash [\vec{\alpha}]\vec{\beta}; \vec{\beta}'}\ \textbf{Cont-;}$$

(II-1) Axioms about primitive actions

$\vdash [\alpha_1; \cdots; \alpha_n]\alpha_1; \cdots; \alpha_n$

(II-2) Axioms for relationships between properties

(Here axioms including non-monotonic property are marked by †.)

Freshness 1:
$P\ generates\ n \vdash fresh(n)$

Freshness 2: (where $m \sqsubseteq m'$.)
$fresh(m) \vdash fresh(m')$

Nonce Verification 1: (where $\{m\}_{K^{-1}} \sqsubseteq m', m''$.)
$(PK(K,Q)), (fresh(m)), (P\ receives\ m'(\{m\}^*_{K^{-1}})) \vdash (Q\ sends\ m''); (P\ receives\ m'(\{m\}^*_{K^{-1}}))$

Nonce verification 2†: (where $\{m_1\}_K \sqsubseteq m_2, m_3$ and $\{m_1\}_K \not\sqsubseteq m_5$ and $n \sqsubseteq m_1, m_4, m_5$.)
$(PK(K,Q)), (P\ fsends\ (m_2, n)), (P\ generates\ n), (P\ receives\ m_5)$
$\vdash (P\ fsends\ (m_2, n)); (Q\ receives\ m_3(\{m_1\}^*_K)); (Q\ sends\ m_4); (P\ receives\ m_5)$

Nonce verification 3†: (additionally to the condition for **Nonce Verification 2**, $\{m_1\}'_K \not\sqsubseteq m_5$ is also satisfied.)
$(PK(K,Q)), (P\ fsends\ (m_2, n)), (P\ generates\ n), (P\ receives\ m_5), (Q\ sends\ \{m_4\}'_K), (PK(K', A))$
$\vdash m_5 = \{m_4\}'_K$

We also admit axioms obtained from Nonce verification 1-3 by replacing $PK(K,Q)$ with $P \overset{K}{\leftrightarrow} Q$, respectively.

Shared secret: (where $K' \sqsubseteq m_1, m_2$.)
$(P\ sends\ \{m_1\}_{K_1}), (P\ sends\ \{m_2\}_{K_2}), (P\ generates\ K'), (P \overset{K_1}{\leftrightarrow} Q), (P \overset{K_2}{\leftrightarrow} R) \vdash (Q \overset{K'}{\leftrightarrow} R)$

Firstly Sends†:

This rule satisfies (♯♯) condition (cf. Section 3.1).

$$\frac{Honest(\vec{\alpha}^P), \ldots, Honest(\vec{\beta}^Q), \Delta \vdash [\vec{\gamma}]\vec{\delta}; (R\ sends\ m); \vec{\delta'}}{Honest(\vec{\alpha'}^P), \ldots, Honest(\vec{\beta'}^Q), \Delta \vdash [\vec{\gamma'}]\vec{\delta}; (R\ fsends\ (m,n)); \vec{\delta'}}$$

Ordering of Actions†: (where α is an action predicate of message m' with $n \sqsubseteq m'$.)
$(P\ generates\ n), (P\ fsends\ (m,n)), \alpha \vdash P\ fsends\ (m,n); \alpha$

(III) Honesty inferences

For (1) Substitution and for (3) Deriving another action, we also admit the inference rules obtained by replacing "*receives*" with "*generates*" or "*sends*", respectively. These rules satisfy the (♯) (cf. Section 2.2 (III)) and (♯♯) conditions.

(1) Substitution:

$$\frac{\Gamma \vdash [\vec{\alpha}](Q\ receives\ m[t/x])}{\Gamma, Honest(Q\ receives\ m) \vdash [\vec{\alpha}]x = t}\ \textbf{H(S)}$$

(2) Matching: (where $m \sqsubseteq m'$.)

$$\frac{\Gamma \vdash [\vec{\alpha}]\vec{\beta}; (Q\ sends\ m); \vec{\gamma}}{\Gamma, Honest(Q\ sends\ m', m) \vdash [\vec{\alpha}]\vec{\beta}; (Q\ sends\ m'); \vec{\gamma}}\ \textbf{H(M)}$$

(3) Deriving another action in a role:

$$\frac{\Gamma \vdash [\vec{\alpha}]\vec{\beta}; (Q\ sends\ m); \vec{\gamma}}{\Gamma, Honest_Q(Q\ receives\ m'; Q\ sends\ m) \vdash [\vec{\alpha}]\vec{\beta}; (Q\ receives\ m'); (Q\ sends\ m); \vec{\gamma}}\ \textbf{H(R)}$$

(IV) Weakening rules for actions and honesty assumptions

Weakening rule for honesty assumptions (left below) satisfies (♯) and (♯♯) conditions, and weakening rule for actions (right below) satisfies (♯♯) condition.

$$\frac{\Gamma, Honest(\vec{\alpha}^P; \vec{\alpha'}^P) \vdash [\vec{\beta}]\varphi}{\Gamma, Honest(\vec{\alpha}^P; \alpha''^P; \vec{\alpha'}^P) \vdash [\vec{\beta}]\varphi}\ \textbf{W(Hon)} \qquad \frac{\Gamma \vdash [\vec{\alpha}^P; \vec{\alpha'}^P]\varphi}{\Gamma \vdash [\vec{\alpha}^P; \alpha''^P; \vec{\alpha'}^P]\varphi}\ \textbf{W(Act)}$$