有限資源の論理
A Logic for Finite Resources

上出　哲広 †
Norihiro KAMIDE

†（独）産業技術総合研究所, システム検証研究センター
AIST, Research Center for Verification and Semantics (CVS)
東京工業高等専門学校情報工学科（非常勤講師）
Tokyo National College of Technology (part-time lecturer)
http://www.geocities.jp/logicincomputerscience2006

A finite resource logic (FRL), which has a bounded soft-exponential operator and a temporal next-time operator, is introduced as a modification of Lafont's soft linear logic. Decidability, cut-elimination, Kripke-completeness and Petri net-interpretation are shown for FRL. A specific finitely usable resource such as a computer virus or vaccine program can appropriately be represented using FRL.

1 Introduction

Soft linear logic (SLL) was introduced by Lafont [10] to cast light on polynomial time computation. SLL has a novel soft-exponential operator $!_s$ which is characterized by the multiplexing rule:

$$\frac{\overbrace{\alpha, ..., \alpha}^{n}, \Gamma \Rightarrow \gamma}{!_s\alpha, \Gamma \Rightarrow \gamma}$$

where n can be *any natural number*. In the present paper, a *finite resource logic* (FRL) is introduced as a modification of SLL, and it has a bounded soft-exponential operator $!_r$ which is characterized by the bounded multiplexing rule:

$$\frac{\overbrace{\alpha, ..., \alpha}^{0 \le n \le r}, \Gamma \Rightarrow \gamma}{!_r\alpha, \Gamma \Rightarrow \gamma}$$

where n can be any natural number between 0 and a *finite fixed natural number* r. Although the decidability of SLL is unknown whether holds or not, the decidability of FRL can be shown by the virtue of the limited multiplicity n.

It is known that the linear-exponential operator ! in Girard's linear logic [1] can express a specific infinitely reusable resource, i.e. it is reusable not only for any number, but also many times. In contrast, by reading the multiplexing rule of SLL from the bottom up, the intuitive meaning of the soft-exponential formula $!_s\alpha$ is "the resource α is usable in any number, but only once (i.e. it is consumed after use)." Since there is no infinite resource in the real world, the number n in the multiplexing rule for $!_s$ is sufficient to be less than a fixed finite positive integer. Such a realistic modification of $!_s$ is a bounded soft-exponential operator $!_r$. By reading the bounded multiplexing rule of FRL from the bottom up, the intuitive interpretation of the formula $!_r\alpha$ is "the resource α is usable in any *finite number less than* $r+1$, but only once."

The next-time operator $[N]$ used in FRL is the modal logic K-type operator, and the relationship between $[N]$ and $!_r$ is expressed axiomatically by $!_r\alpha \to [N]\alpha$, which intuitively means "if the resource α is usable in any finite number less than $r+1$, but only once at the nearest time in the future, then α is usable exactly once at the next moment". In this intuition, **"time" is regarded as a "resource"**. This intuitive meaning may be justified in that the concept of "time" in computer systems, such as CPU-time in process scheduling, is considered to be a "resource". Similarly, in the real world, "time is money, i.e. resource."

The following useful interpretations for some formulas in FRL can be provided as a result.

- $!_r\alpha$: "The resource α is usable in any finite number less than $r+1$, but only once at the nearest time in the future (i.e. it is consumed after use).”

- $[N]\alpha$: “The resource α is usable once at the next moment (i.e. it is usable only once at the nearest time in the future).”
- α: “The resource α is usable once at the present moment (i.e. it is usable only once, and must be only at the present time).”

The operator $!_r$ can be used to express some specific programs, such as computer virus and vaccine programs, and the operator $[N]$ can be used to represent time-dependent software or time-limitted software. Such a program like virus is, roughly speaking, executable or usable simultaneously in any finite number, but only once (i.e. not reusable many times, or only one executable). Indeed, any existing old virus and vaccine programs are regarded as unavailable many times. Some time-dependent software like security-soft and movie-soft are available only within a time limit. These programs and software may appropriately be expressed using $!_r$ and $[N]$. In addition, it is remarked that $!_r$ can be interpreted as the “bounded replication operator” for programs or processes in concurrency theory.

2 Sequent Calculi

Before the detailed discussion, the language used in this paper is introduced. *Formulas* are constructed from propositional variables, $\mathbf{1}$ (multiplicative constant), $\rightarrow$ (implication), $\wedge$ (conjunction), $*$ (fusion), and $!_r$ (bounded soft-exponential). Lower-case letters $p, q, ...$ are used to denote propositional variables, Greek lower-case letters $\alpha, \beta, ...$ are used to denote formulas, and Greek capital letters $\Gamma, \Delta, ...$ are used to represent finite (possibly empty) multisets of formulas. Expressions $!_r\Gamma$ and $[N]\Gamma$ denote the multisets $\{!_r\gamma \mid \gamma \in \Gamma\}$ and $\{[N]\gamma \mid \gamma \in \Gamma\}$, respectively. A *sequent* is an expression of the form $\Gamma \Rightarrow \gamma$ (the conclusion of the sequent is non-empty). If a sequent S is provable in a sequent calculus L, then such a fact is denoted as $L \vdash S$ or $\vdash S$. Since all logics discussed in this paper are formulated as sequent calculi, we will occasionally identify a sequent calculus with the logic determined by it.

Definition 2.1 (FRL and SLL) *The initial sequents of* FRL *are of the form:*

$$\alpha \Rightarrow \alpha \qquad \Rightarrow \mathbf{1}.$$

The cut rule of FRL *is of the form:*

$$\frac{\Gamma \Rightarrow \alpha \quad \alpha, \Sigma \Rightarrow \gamma}{\Gamma, \Sigma \Rightarrow \gamma} \text{ (cut).}$$

The logical (and structural) inference rules of FRL *are of the form:*

$$\frac{\Gamma \Rightarrow \gamma}{\mathbf{1}, \Gamma \Rightarrow \gamma} \text{ (}\mathbf{1}\text{we)}$$

$$\frac{\Gamma \Rightarrow \alpha \quad \beta, \Sigma \Rightarrow \gamma}{\alpha \rightarrow \beta, \Gamma, \Sigma \Rightarrow \gamma} \text{ (}\rightarrow\text{left)} \qquad \frac{\alpha, \Gamma \Rightarrow \beta}{\Gamma \Rightarrow \alpha \rightarrow \beta} \text{ (}\rightarrow\text{right)}$$

$$\frac{\alpha, \beta, \Gamma \Rightarrow \gamma}{\alpha * \beta, \Gamma \Rightarrow \gamma} \text{ (}*\text{left)} \qquad \frac{\Gamma \Rightarrow \alpha \quad \Delta \Rightarrow \beta}{\Gamma, \Delta \Rightarrow \alpha * \beta} \text{ (}*\text{right)}$$

$$\frac{\alpha, \Gamma \Rightarrow \gamma}{\alpha \wedge \beta, \Gamma \Rightarrow \gamma} \text{ (}\wedge\text{left1)} \qquad \frac{\beta, \Gamma \Rightarrow \gamma}{\alpha \wedge \beta, \Gamma \Rightarrow \gamma} \text{ (}\wedge\text{left2)}$$

$$\frac{\Gamma \Rightarrow \alpha \quad \Gamma \Rightarrow \beta}{\Gamma \Rightarrow \alpha \wedge \beta} \text{ (}\wedge\text{right)}$$

$$\frac{\Gamma \Rightarrow \gamma}{!_r\Gamma \Rightarrow !_r\gamma} \text{ (}!_r\text{regu)} \qquad \frac{\overbrace{\alpha, ..., \alpha}^{0 \,\leq\, n \,\leq\, r}, \Gamma \Rightarrow \gamma}{!_r\alpha, \Gamma \Rightarrow \gamma} \text{ (}!_r\text{multi)}$$

$$\frac{\Gamma, \Delta \Rightarrow \alpha}{!_r\Gamma, [N]\Delta \Rightarrow [N]\alpha} \text{ (}[N]\text{regu)}$$

where n in ($!_r$multi) *can be any natural number between* 0 *and an arbitrary finite fixed natural number r.*

The logic FRL $-$ ($[N]$regu) *is called a* bounded soft linear logic*, denoted as* BSL*. The logic obtained from* BSL *by replacing* ($!_r$multi) *by the unbounded (i.e. n can be any natural number) version of* ($!_r$multi) *is called a* soft linear logic*, denoted as* SLL*.*

It is remarked that the exchange rule is omitted in FRL, since we adopt the multiset as the antecedent of the sequent.

Theorem 2.2 (Cut-elimination) (1) *The rule* (cut) *is admissible in cut-free* FRL*.* (2) *The subformula property holds for cut-free* FRL*.*

Using this theorem, the proof of the decidability of FRL will be sketched in the next section.

Finally in this section, in order to consider the difference between $!_r$ in FRL and that in SLL, we introduce an alternative cut-free sequent calculus FRL^g for FRL, which has a generalized version of ($!_r$multi).

Definition 2.3 ($\mathbf{FRL}^g$) FRL^g *is obtained from* FRL *by replacing* ($!_r$multi) *by the inference rule of the form:*

$$\frac{\overbrace{\alpha,\ldots,\alpha}^{0\le n\le r^l},\Gamma\Rightarrow\gamma}{\overbrace{!_r\cdots!_r}^{1\le l}\alpha,\Gamma\Rightarrow\gamma}\ (\text{g-}!_r\text{multi})$$

where r is a finite fixed natural number, n is the multiplicity of α, and l is the multiplicity of $!_r$.

A formulation using (g-$!_r$multi) can not be adapted for SLL, because in SLL, such a generalized rule is meaning-less. This is a clear difference between SLL and FRL.

Theorem 2.4 (1) *For any sequent T, if* $\mathrm{FRL}^g \vdash T$*, then* $\mathrm{FRL} \vdash T$*.* (2) *For any sequent T, if* $\mathrm{FRL} - (\mathrm{cut}) \vdash T$*, then* $\mathrm{FRL}^g - (\mathrm{cut}) \vdash T$*.*

Proof The proof of (2) is obvious, because ($!_r$multi) is an instance of (g-$!_r$multi). We show (1) by induction on a proof P of T in FRL^g. We distinguish the cases according to the last inference rules in P. We only show the following case.

Case (g-$!_r$multi): The last inference rule in P is of the form:

$$\frac{\overbrace{\alpha,\ldots,\alpha}^{0\le n\le r^l},\Gamma\Rightarrow\gamma}{\overbrace{!_r\cdots!_r}^{1\le l}\alpha,\Gamma\Rightarrow\gamma}\ (\text{g-}!_r\text{multi}).$$

We show that this is derivable in FRL, by induction on the number l in the rule. The base step $l = 1$ is obvious, because in this case, (g-$!_r$multi) is equal to ($!_r$multi). We show the induction step below. By the hypothesis of induction, we have the fact that the rule

$$\frac{\overbrace{\alpha,\ldots,\alpha}^{0\le n\le r^{(l-1)}},\Gamma\Rightarrow\gamma}{\overbrace{!_r\cdots!_r}^{1\le(l-1)}\alpha,\Gamma\Rightarrow\gamma}\ (\text{ind. hyp.}).$$

is derivable in FRL. On the other hand, by the hypothesis (of induction with respect to P), we have

$$\mathrm{FRL}\vdash \overbrace{\alpha,\ldots,\alpha}^{0\le n\le r^l},\Gamma\Rightarrow\gamma,$$

and hence

$$(\text{hyp.}) : \mathrm{FRL}\vdash \overbrace{\overbrace{\alpha,\ldots,\alpha}^{0\le n'\le r^{(l-1)}},\ldots,\overbrace{\alpha,\ldots,\alpha}^{0\le n'\le r^{(l-1)}}}^{r},\Gamma\Rightarrow\gamma.$$

By (hyp.) and (ind. hyp.), we obtain the required fact

$$\dfrac{\begin{array}{c}\overbrace{\overbrace{\alpha,\ldots,\alpha}^{0\le n'\le r^{(l-1)}},\ldots,\overbrace{\alpha,\ldots,\alpha}^{0\le n'\le r^{(l-1)}}}^{r},\Gamma\Rightarrow\gamma\\ \vdots\ (\text{ind. hyp.})\times r\\ \overbrace{\overbrace{!_r\cdots!_r}^{l-1}\alpha,\ldots,\overbrace{!_r\cdots!_r}^{l-1}\alpha}^{r},\Gamma\Rightarrow\gamma\end{array}}{\overbrace{!_r\cdots!_r}^{l}\alpha,\Gamma\Rightarrow\gamma}\ (!_r\text{multi}).$$

∎

Corollary 2.5 (Cut-elimination) *The rule* (cut) *is admissible in cut-free* FRL^g*.*

Proof Suppose that T is provable in FRL^g. Then, T is also provable in FRL by Theorem 2.4 (1). Moreover, by Theorem 2.2 (1), T is provable in cut-free FRL. Therefore, T is provable in cut-free FRL^g by Theorem 2.4 (2). ∎

3 Decidability

In order to check whether a given sequent S is provable in cut-free FRL or not, we try to find a proof of S in the following way. First, we search for every sequent that can be an upper sequent of some inference rules of cut-free FRL whose lower sequent is S. Then, we write each of them just above the sequent S. We call this process the *decomposition* of S. Second, we decompose each sequent which we have obtained just now, and repeat it again. Of course, we can not decompose a sequent which can not be a lower sequent of any rule of inference. By doing so, we can get a tree such that some sequent is attached to each of its points. Let us call it the *complete proof search tree* of S.

In the following, we show that the complete proof search tree of each sequent is finite. To show this, we use the following proposition.

Proposition 3.1 (König's Lemma) *A tree is finite if and only if both* (1) *there are only finitely many points connected directly by lines to a given point (finite fork property) and* (2) *each path is finite (finite path property).*

(**Finite fork property**): First, we show that the finite fork property for the complete proof search tree of each sequent in cut-free FRL holds. The subformula property holds for cut-free FRL by Theorem 2.2 (2). Thus, for each sequent S in a complete proof search tree, we have a finite number of the upper sequents which can be an upper sequent of some inference rules in cut-free FRL whose lower sequent is S. This is because of the following reasons: (a) S is constructed from a finite multiset of formulas, (b) the number of the sorts of subformulas of some formulas occurring in S is finite, (c) the number of the inference rules in FRL is finite, (d) for each inference rule in cut-free FRL, the number of the application forms are determined by less than a finite number, because, in particular, the multiplicity n in the rule ($!_r$multi) in FRL is bounded by a fixed finite number r, and (e) the number of (sub)formulas occurring in the upper sequent(s) in the inference rules of cut-free FRL is finite. We remark that (d) and (e) do not hold for cut-free SLL: the number of the sequents which can be an upper sequent of the multiplexing rule

$$\frac{\overbrace{\alpha, ..., \alpha}^{n}, \Gamma \Rightarrow \gamma}{!_s\alpha, \Gamma \Rightarrow \gamma} \ (!_s\text{multi})$$

can be infinite, and the number of subformula α of $!_s\alpha$ in the upper sequent in this rule can also be infinite, because the multiplicity n in this rule can be infinite. For example, a bottom-up proof search with respect to ($!_s$multi) derives an infinite branch (in a proof search tree) such as

$$\begin{array}{ccccccc} \vdots & & \vdots & & \vdots & & \vdots \\ \Gamma \Rightarrow \gamma & & \alpha, \Gamma \Rightarrow \gamma & \cdots & \alpha, ..., \alpha, \Gamma \Rightarrow \gamma & \cdots & \infty \\ & & & \vdots & & & \\ & & & !_s\alpha, \Gamma \Rightarrow \gamma. & & & \end{array}$$

(**Finite path property**): Second, we show that the finite path property for the complete proof search tree of each sequent in cut-free FRL holds. Obviously, we have the finite path property for the complete proof search tree of each sequent in the {(cut), ($!_r$multi), ($!_r$regu), ([N]regu) }-free FRL. Moreover, the rule ($!_r$regu) or ([N]regu) is not a cause that produces an infinite path in a complete proof search tree. The possible cause is only from the rule:

$$\frac{\overbrace{\alpha, ..., \alpha}^{0 \le n \le r}, \Gamma \Rightarrow \gamma}{!_r\alpha, \Gamma \Rightarrow \gamma} \ (!_r\text{multi})$$

where $!_r\alpha$ is called the *principal formula* in ($!_r$multi), and α in the upper sequent is called the *side formula* of $!_r\alpha$ in ($!_r$multi). However, all the side formulas $\alpha, ..., \alpha$ of $!_r\alpha$ in ($!_r$multi) are proper subformulas, and hence the number of the connective $!_r$ in each α is less than that in $!_r\alpha$. Moreover, we have the properties (a), (b) and (e) discussed in the case for the "finite fork property". Thus, the number of the repeated applications of ($!_r$multi) to the (side) subformulas of the principal formulas of the form $!_r\alpha$ is finite, because $!_r$ occurring in $!_r\alpha$ is considered to be finite. Therefore, the number of the applications of ($!_r$multi) in any path in the complete proof search tree of each sequent in cut-free FRL is finite.

Theorem 3.2 (Decidability) FRL *is decidable.*

Proof Suppose that an arbitrary sequent S is given. We construct the complete proof search tree of S. The complete proof search tree becomes finite, by using Proposition 3.1 with (1) finite fork property and (2) finite path property. Then, we check whether the complete proof search tree contains a subtree which forms a proof of S or not. If it does, then clearly S is provable, and otherwise, it is unprovable. ∎

4 Kripke Semantics

The semantics presented here is regarded as an extended version of a Kripke semantics presented by Ono and Komori [11].

Definition 4.1 *Let r be a fixed finite natural number. A* Kripke frame *is a structure $\langle M, \cdot, \dagger, N, \varepsilon, \geq \rangle$ satisfying the following conditions:*

1. *$\langle M, \cdot, \varepsilon \rangle$ is a commutative monoid with the identity ε,*
2. *$\langle M, \geq \rangle$ is a pre-ordered set,*
3. *$\dagger$ and N are unary operations on M such that*

   C0*: $\varepsilon \geq \dagger\varepsilon$,*
   C1*: $\dagger x \cdot \dagger y \geq \dagger(x \cdot y)$ for all $x, y \in M$,*
   C2*: $\dagger y \cdot x \geq x$ for all $x, y \in M$,*
   C3*: $\dagger x \geq \overbrace{x \cdots x}^{n}$ for all $1 \leq n \leq r$ and all $x \in M$ if $1 \leq r$,*
   C4*: $\dagger x \geq Nx$ for all $x \in M$,*
   C5*: $\varepsilon \geq N\varepsilon$,*
   C6*: $Nx \cdot Ny \geq N(x \cdot y)$ for all $x, y \in M$,*

4. *$\cdot$ is monotonic with respect to $\geq$, that is,*

   C7*: $y \geq z$ implies $x \cdot y \geq x \cdot z$ for all $x, y, z \in M$.*

Definition 4.2 *A* valuation *$\models$ on a Kripke frame $\langle M, \cdot, \dagger, N, \varepsilon, \geq \rangle$ is a mapping from the set of all propositional variables to the power set of M and satisfyning the following hereditary condition: $x \in \models(p)$ and $y \geq x$ imply $y \in \models(p)$ for any propositional variable p and any $x, y \in M$. We will write $x \models p$ for $x \in \models(p)$. Each valuation $\models$ can be extended to a mapping from the set of all formulas to the power set of M by*

1. *$x \models \mathbf{1}$ iff $x \geq \varepsilon$,*
2. *$x \models \alpha \rightarrow \beta$ iff $y \models \alpha$ implies $x \cdot y \models \beta$ for all $y \in M$,*
3. *$x \models \alpha \wedge \beta$ iff $x \models \alpha$ and $x \models \beta$,*
4. *$x \models \alpha * \beta$ iff $y \models \alpha$ and $z \models \beta$ for some $y, z \in M$ with $x \geq y \cdot z$,*
5. *$x \models !_r\alpha$ iff $y \models \alpha$ for some $y \in M$ with $x \geq \dagger y$,*
6. *$x \models [N]\alpha$ iff $y \models \alpha$ for some $y \in M$ with $x \geq Ny$.*

Definition 4.3 *A* Kripke model *is a structure $\langle M, \cdot, \dagger, N, \varepsilon, \geq, \models \rangle$ such that* (1) *$\langle M, \cdot, \dagger, N, \varepsilon, \geq \rangle$ is a Kripke frame, and* (2) *$\models$ is a valuation on $\langle M, \cdot, \dagger, N, \varepsilon, \geq \rangle$. A formula α is* true *in a Kripke model $\langle M, \cdot, \dagger, N, \varepsilon, \geq, \models \rangle$ if $\varepsilon \models \alpha$, and* valid *in a Kripke frame $\langle M, \cdot, \dagger, N, \varepsilon, \geq \rangle$ if it is true for any valuation $\models$ on the Kripke frame. A sequent $\alpha_1, \cdots, \alpha_n \Rightarrow \beta$ (or $\Rightarrow \beta$) is* true *in a Kripke model $\langle M, \cdot, \dagger, \varepsilon, \geq, \models \rangle$ if the formula $\alpha_1 * \cdots * \alpha_n \rightarrow \beta$ (or β respectively) is true in it, and* valid *in a Kripke frame if so is $\alpha_1 * \cdots * \alpha_n \rightarrow \beta$ (or β respectively).*

The following theorem can be proved in the same way as that for a soft linear logic discussed in [4].

Theorem 4.4 (Completeness) *Let C be the class of all Kripke frames, $L := \{S \mid \mathrm{FRL} \vdash S\}$ and $L(C) := \{S \mid S$ is valid in all frames of $C\}$. Then, $L = L(C)$.*

5 Petri Net Interpretation

Definition 5.1 *A* Petri net *is a structure $\langle P, T, (\cdot)^{\bullet}, (\cdot)_{\bullet} \rangle$ such that*

1. *P is a set of* places*,*
2. *T is a set of* transitions*,*
3. *$(\cdot)^{\bullet}$ and $(\cdot)_{\bullet}$ are mappings from T to the set $\mathcal{M}$ of all multisets over P.*

Each element of $\mathcal{M}$ is called a marking.

Definition 5.2 *A* firing relation *$[t\rangle$ for $t \in T$ on $\mathcal{M}$ is defined as follows: for any $m_1, m_2 \in \mathcal{M}$,*

$$m_1\ [t\rangle\ m_2 \text{ iff } m_1 = m_3 + t^{\bullet} \text{ and } t_{\bullet} + m_3 = m_2 \text{ for some } m_3 \in \mathcal{M}.$$

A reachability relation *$\gg$ on $\mathcal{M}$ is defined as follows: for any $m, m' \in \mathcal{M}$,*

$$m \gg m' \text{ iff } m\ [t_1\rangle\ m_1\ [t_2\rangle \cdots [t_n\rangle\ m_n = m' \text{ for some } t_1, ..., t_n \in T,\ m_1, ..., m_n \in \mathcal{M} \text{ and } n \geq 0.$$

Definition 5.3 *A* Petri net structure *is a structure* $\langle M, +, \emptyset, \gg \rangle$ *such that*

1. *M is the set of all markings,*
2. *$+$ is a multiset union operation on N,*
3. *$\emptyset$ is an empty multiset,*
4. *$\gg$ is a reachability relation on N.*

Proposition 5.4 *A Petri net structure $\langle M, +, \emptyset, \gg \rangle$ is just a $\{\dagger, N\}$-free reduct of a Kripke frame.*

Definition 5.5 *A syntactical Kripke model $\langle M, \cdot, \dagger, N, \varepsilon, \geq, \models \rangle$ is defined by:*

1. *$M := \{\Gamma \mid \Gamma$ is a finite multiset of formulas$\}$,*
2. *for any $\Gamma, \Delta \in M$, $\Gamma \cdot \Delta := \Gamma \cup \Delta$ (the multiset union of Γ and Δ),*
3. *for any $\Gamma \in M$, $\dagger\Gamma := !_r\Gamma$ and $N\Gamma := [N]\Gamma$,*
4. *ε is an empty multiset,*
5. *for any $\Gamma, \Delta \in M$, $\Gamma \geq \Delta$ is defined by $\vdash \Gamma \Rightarrow \Delta^\star$ where $\Delta^\star \equiv \gamma_1 * \cdots * \gamma_n$ if $\Delta \equiv \{\gamma_1, \cdots, \gamma_n\}$ $(0 < n)$, and $\Delta^\star \equiv \mathbf{1}$ if Δ is empty,*
6. *a valuation $\models$ on $\langle M, \cdot, \dagger, N, \varepsilon, \geq \rangle$ is a mapping from the set* PROP *of all propositional variables to the power set of M defined by*

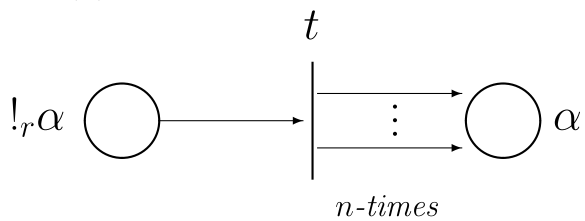
   $\Gamma \in \models (p)$ iff $\vdash \Gamma \Rightarrow p$ for any $p \in$ PROP and any $\Gamma \in M$.

   This valuation $\models$ can be extended to a mapping from the set FORM *of all formulas to the power set of M in a natural way.*

By Prposition 5.4 and the syntactical Kripke model defined in Definition 5.5, we can obtain a Petri net interpretation for FRL: a place name (or token) in a Petri net corresponds to formula in FRL, the reachability relation in a Petri net corresponds to an antecedent (or consequenct) of a sequent in FRL, i.e. $\Gamma \gg \Delta$ corresponds to FRL $\vdash \Gamma \Rightarrow \Delta^\star$ for any multisets Γ and Δ.

On the other hand, we have a question: "What is the Petri net interpretation of the modal operators?" We answer this question in the following. A marking (or token) with $[N]$ can be interpreted as a timed marking (or token). The bounded soft-exponential $!_r$ can represent the (variable) weight of arcs (i.e. the (variable) number of arcs).

Example 5.6 (Bounded soft-exponential) *Let n be less than $r+1$ with a finite fixed natural number r. We give a Petri net $N := \langle P, T, (\cdot)^\bullet, (\cdot)_\bullet \rangle$ with $P := \{!_r\alpha, \alpha\}$, $T := \{t\}$, $t^\bullet := \{!_r\alpha\}$ and $t_\bullet := \{\alpha\}$. Graphically this becomes the following:*



This net corresponds to the fact $\vdash !_r\alpha \Rightarrow \overbrace{\alpha * \cdots * \alpha}^{0 \leq n \leq r}$. If the place $!_r\alpha$ has a token, then the transition t is enabled. Moreover, if t fires then the place α get a number of tokens, but t can only one fiable with respect to one token. This interpretation expresses the essence of the computational meaning of the bounded soft-exponential.

6 Illustrative Examples

The notion of "resource", encompassing concepts such as processor time, memory, cost of components and energy requirements, is fundamental to computational systems [12]. In the area of AI, this notion is also very important in handling real scheduling problems to construct complex plans of actions, since many actions consume resources, such as money, gas and raw materials [13]. [1] It is known that linear logics can elegantly represent the concept of "resource consumption" [1]. The central motivation of this section is to express a more fine-grained form of resource-sensitive reasoning based on FRL.

In the following, for a formula α and a positive

[1] See Section 12 in [13].

integer n, an expression α^n means $\overbrace{\alpha * \cdots * \alpha}^{n}$, and an expression $[N]^n$ means $\overbrace{[N] \cdots [N]}^{n}$.

Example 6.1 (Vending machine) *A vending machine example is known as a good example of the resource-sensitive reasoning based on linear logics. For example, the following expression is considered to be appropriate:*

$$\vdash coin^2 \Rightarrow juice^2.$$

This expression means "if we put two coins in a vending machine, then we can get exactly two cans of juice from the machine". This expression also means that "⇒" represents "resource consumption". We now consider more fine-grained examples based on FRL.

We assume that the vending machine discussed can deal with less than 101 cans of juice. Using the restricted soft exponential operator $!_r$*, such a situation is briefly expressed:*

$$\vdash coin^n \Rightarrow\ !_{100}\ juice \quad (0 \leq n \leq 100)$$

where "$coin^0$*" means* **1**. *This expression means "we can get 100 cans of juice (i.e. less than 101)". Using the temporal next time operator* $[N]$*, we can express a time-dependent situation as follows:*

$$\vdash coin \Rightarrow [N]^2\ juice.$$

This means that the juice can be obtained with a lag of two time units.

Example 6.2 (Digestion) *Digestion is regarded as a model of resource consumption, i.e. any person consume any food by the bowels as energy. Such a situation for digestion can be viewed as time-dependent, i.e. there are many sorts of foods that digest slowly or fastly. The following expressions are thus useful:*

$$\vdash\ !_3\ hamburger \Rightarrow [N]^4\ digest,$$

$$\vdash\ !_3\ hamburger\ *\ digestive \Rightarrow [N]^3\ digest.$$

The second expression means "we can digest three hamburgers with a digestive within 3-time units".

Example 6.3 (Medicine consumption) *The model of digestion discussed is applicable to a similar situation for medical treatment by medicines. Suppose that a patient has diabetes, and in order to recover this disease, the patient must take the medicines of insulin before the meal. Then, such a situation is expressed:*

$$\vdash\ !_3\ insulin\ *\ [N]^{30}\ food \Rightarrow [N]^{60}\ recover.$$

If we consider a situation "if the patient takes over-administration of insulin, then the patient is in the insulin-shock", then, generally speaking, we can express such a medically worse fact as follows:

$$\vdash medicine \Rightarrow \mathbf{0}$$

where **0** *means the multiplicative falsum constant, which is not disccussed in the setting of* FRL. *This expression means "if a patient uses a medicine to recover from a disease, the patient makes no recovery from the disease with the medicine". In other words, to use the medicine does not derive a medically beneficial effect. In this expression,* **0** *represents a broad range of facts that the patient takes a turn for worse, such as death, the onset of a critical condition or side-effects of less consciousness, i.e.* **0** *represents a width of contradiction. This feature of* **0** *corresponds to the fact that the algebraic counter-part of* **0** *in some algebraic semantics such as phase semantics has a width. This fact is followed from the fact that* FRL *has no usual contraction and weakening rules.*

Example 6.4 (Resource recycling system) *We consider some resource recycling systems below. Suppose that garbage is divided into two sorts, i.e. reusable (recyclable) and non-reusable. In the case of recyclable garbage, the following expressions may be useful:*

$$\vdash plastic\_vessel^{100} \Rightarrow\ !_{50}\ recycle\_plastic,$$

$$\vdash newspaper^{10} \Rightarrow recycle\_paper^5\ *\ toilet\_tissue^5$$

where "⇒" means a recycling relation, i.e. the antecedent represents garbage and the consequent represents recycle goods. In the case of non-reusable garbage, the following expressions may be usuful:

$\vdash flammable\_garbage^{100}$
$\Rightarrow [N]^3 \, !_{80} \, thermal\_energy * !_{20} \, [N]^5 \, cinder,$

$\vdash nonflammable\_garbage \Rightarrow \mathbf{0}$

where the second expression represents a limitation of such a thermal powerplant. This case also means that energy is ragarded as resource.

7 Remarks and Related Works

It is remarked that the cut-elimination and decidability results for FRL can be extended to the extended language with $\vee$ (disjunction), $\top$ (additive truth constant), $\perp$ (additive falsum constant) and $\sim$ (strong negation connective), and the Kripke-completeness and Petri net-interpretation results for FRL can also be extended to this language using the method introduced in [3, 7]. It is also remarked that the Kripke-completeness result for FRL can be extended to the extended language with $\forall$ (first-order universal quantifier) using the method presented in [4].

Finally, recent developments of various linear logics are reviewed briefly below. *Temporal linear logics* were introduced by Tanabe [14] and by Kanovich and Ito [8], an *intuitionistic temporal linear logic* (ITLL) was introduced by Hirai [2, 15], a *heigher-order multi-modal linear logic* (MLL) was studied by Kobayashi, Shimizu and Yonezawa [9], a *spatio-temporal soft linear logic* (SSLL) [4] was introduced as a modified extension of both ITLL and propositional MLL, a *temporal spatial epistemic intuitionistic linear logic* (TSEILL) [5] was presented, and a *classical temporal soft linear logic* (TSLL) [6] was introduced as an extension of both Girard's classical linear logic and Lafont's classical soft linear logic. It is remarked that FRL is regarded as a decidable subsystem of ITLL, SSLL, TSEILL and TSLL. It is also remarked that FRL is an intuitionistic refinement of a subsystem of TSLL in [6], and the decidability method for FRL is also an intuitionistic refinement of that for a (classical) *bounded soft linear logic* (BSLL) introduced in [6].

参考文献

[1] Girard, J-Y., Linear logic, *Theoretical Computer Science* 50, 1-102, 1987.

[2] Hirai, T., Propositional temporal linear logic and its application to concurrent systems, *IEICE Transactions. Fundamentals* E83-A 11, 2219–2227, 2000.

[3] Kamide, N., Normal modal substructural logics with strong negation, *Journal of Philosophical Logic* 32, 589–612, 2003.

[4] Kamide, N., Combining soft linear logic and spatio-temporal operators, *Jornal of Logic and Computation* 14 (5), 625–650, 2004.

[5] Kamide, N., Linear and affine logics with temporal, spatial and epistemic operators, *Theoretical Computer Science* 353 (1–3), 165–207, 2006.

[6] Kamide, N., Towards a theory of resource: An approach based on soft exponentials, *Journal of Applied Non-Classical Logics* (to appear).

[7] Kamide, N., Phase semantics and Petri net interpretations for resource-sensitive strong negation, *Journal of Logic, Language and Information* (to appear).

[8] Kanovich, M. I. and Ito, T., Temporal linear logic specifications for concurrent processess (extended abstract). *Proceedings of 12th Annual IEEE Symposium on Logic in Computer Science*, 48–57, 1998.

[9] Kobayashi, N. Shimizu, T. and Yonezawa, A., Distributed concurrent linear logic programming, *Theoretical Computer Science* 227, 185–220, 1999.

[10] Lafont, Y., Soft linear logic and polynomial time, *Theoretical Computer Science* 318, 163–180, 2004.

[11] Ono, H. and Komori, Y., Logics without the contraction rule, *Journal of Symbolic Logic* 50, 169-201, 1985.

[12] Pym, D. J., O'Hearn, P. W. and Yang, H., Possible worlds and resources: The semantics of BI, *Theoretical Computer Science* 315 (1), 257–305, 2004.

[13] Russell, S. and Norvig, P., Artificial intelligence: A modern approach (Second edition), Pearson Education, Inc., 2003.

[14] Tanabe, M., Timed Petri nets and temporal linear logic, *Lecture Notes in Computer Science* 1248, 156–174, 1997.

[15] Tamura, N., Hirai, T., Yoshikawa, H., Kang, Kyoung-San. and Banbara, M., Logic programming in an intuitionistic temporal linear logic (in japanease), *Information Processing Society of Japan, Transactions on Programming* 41, SIG4 (PRO 7), 11-23, 2000.