A Hybrid Linear Logic for Constrained Transition Systems with Applications to Molecular Biology

Kaustuv Chaudhuri
INRIA
kaustuv.chaudhuri@inria.fr

Joëlle Despeyroux
INRIA and CNRS, I3S
joelle.despeyroux@inria.fr

Abstract

Linear implication can represent state transitions, but real transition systems operate under temporal, stochastic or probabilistic constraints that are not directly representable in ordinary linear logic. We propose a general modal extension of intuitionistic linear logic where logical truth is indexed by constraints and hybrid connectives combine constraint reasoning with logical reasoning. The logic has a focused cut-free sequent calculus that can be used to internalize the rules of particular constrained transition systems; we illustrate this with an adequate encoding of the synchronous stochastic pi-calculus. We also present some preliminary experiments of direct encoding of biological systems in the logic.

1 Introduction

To reason about state transition systems, we need a logic of state. Linear logic [20] is such a logic and has been successfully used to model such diverse systems as process calculi, references and concurrency in programming languages, security protocols, multi-set rewriting, and graph algorithms. Linear logic achieves this versatility by representing propositions as *resources* that are composed into elements of state using $\otimes$, which can then be transformed using the linear implication ($\multimap$). However, linear implication is timeless: there is no way to correlate two concurrent transitions. If resources have lifetimes and state changes have temporal, probabilistic or stochastic *constraints*, then the logic will allow inferences that may not be realizable in the system being modeled. The need for formal reaosning in such constrained systems has led to the creation of specialized formalisms such as Computation Tree Logic (*CTL*)[16], Continuous Stochastic Logic (*CSL*) [2] or Probabilistic CTL (*PCTL*) [21]. These approaches pay a considerable encoding overhead for the states and transitions in exchange for the constraint reasoning not provided by linear logic. A prominent alternative to the logical approach is to use a suitably enriched process algebra; a short list of examples includes reversible CCS [11], bioambients [35], brane calculi [6], stochastic and probabilistic π-calculi, the PEPA algebra [22], and the κ-calculus [12]. Each process algebra comes equipped with an underlying algebraic semantics which is used to justify mechanistic abstractions of observed reality as processes. These abstractions are then animated by means of simulation and then compared with the observations. Process calculi do not however completely fill the need for *formal logical reasoning for constrained transition systems*. For example, there is no uniform process calculus to encode different stochastic process algebras[1].

Note that logics like CSL or CTL are not such uniform languages either. These formalisms are not *logical frameworks*[2]: Encoding the stochastic π calculus in CSL, for example, would be inordinately complex because CSL does not provide any direct means of encoding π-calculus dynamics such as the linear production and consumption of messages in a synchronous interaction. Actually CSL and CTL mainly aim at specifying properties of behaviors of constrained transition systems, not the systems themselves.

---

[1]Stochastic process algebras are typical examples of the constrained transition systems we aim at formalizing.

[2]Logical frameworks are uniform languages that allow to formally not only specify and analyse, but also compare, or translate from one to the other, different systems, through their (adequate) encodings.

We propose a simple yet general method to add constraint reasoning to linear logic. It is an old idea—*labelled deduction* [39] with *hybrid* connectives [5]—applied to a new domain. Precisely, we parameterize ordinary logical truth on a *constraint domain*: $A@w$ stands for the truth of A under constraint w. Only a basic monoidal structure is assumed about the constraints from a proof-theoretic standpoint. We then use the hybrid connectives of *satisfaction* (at) and *localization* ($\downarrow$) to perform generic symbolic reasoning on the constraints at the propositional level. We call the result *hybrid linear logic* (HyLL); it has a generic cut-free (but cut admitting) sequent calculus that can be strengthened with a focusing restriction [1] to obtain a normal form for proofs. Any instance of HyLL that gives a semantic interpretation to the constraints enjoys these proof-theoretic properties.

Focusing allows us to treat HyLL as a *logical framework* for constrained transition systems. Logical frameworks with hybrid connectives have been considered before; hybrid *LF* (*HLF*), for example, is a generic mechanism to add many different kinds of resource-awareness, including linearity, to ordinary *LF* [34]. *HLF* follows the usual *LF* methodology of keeping the logic of the framework minimal: its proof objects are β-normal η-long natural deduction terms, but the equational theory of such terms is sensitive to permutative equivalences [40]. With a focused sequent calculus, we have more direct access to a canonical representation of proofs, so we can enrich the framework with any connectives that obey the focusing discipline. The representational adequacy [3] of an encoding in terms of (partial) focused sequent derivations tends to be more straightforward than in a natural deduction formulation. We illustrate this by encoding the synchronous stochastic π-calculus ($S\pi$) in HyLL using rate functions as constraints.

In addition to the novel stochastic component, our encoding of $S\pi$ is a conceptual improvement over other encodings of π-calculi in linear logic. In particular, we perform a full propositional reflection of processes as in [26], but our encoding is first-order and adequate as in [7]. HyLL does not itself prescribe an operational semantics for the encoding of processes; thus, bisimilarity in continuous time Markov chains (*CTMC*) is not the same as logical equivalence in stochastic HyLL, unlike in *CSL* [13]. This is not a deficiency; rather, the *combination* of focused HyLL proofs and a proof search strategy tailored to a particular encoding is necessary to produce faithful symbolic executions. This exactly mirrors $S\pi$ where it is the simulation rather than the transitions in the process calculus that is shown to be faithful to the *CTMC* semantics [30].

This work has the following main contributions. First is the logic HyLL itself and its associated proof-theory, which has a very standard and well understood design in the Martin-Löf tradition. Second, we show how to obtain many different instances of HyLL for particular constraint domains because we only assume a basic monoidal structure for constraints. Third, we illustrate the use of focused sequent derivations to obtain adequate encodings by giving a novel adequate encoding of $S\pi$. Our encoding is, in fact, *fully adequate*, *i.e.*, partial focused proofs are in bijection with traces. The ability to encode $S\pi$ gives an indication of the versatility of HyLL. Finally, we show how to encode (a very simple example of) biological systems in HyLL. This is a preliminary step towards a logical framework for systems biology, our initial motivation for this work.

The sections of this paper are organized as follows: in sec. 2, we present the inference system (natural deduction and sequent calculus) for HyLL and describe the two main semantic instances: temporal and probabilistic constraints. In sec. 3 we sketch the general focusing restriction on HyLL sequent proofs. In sec. 4 we give the encoding of $S\pi$ in probabilistic HyLL, and show that the encoding is representationally adequate for focused proofs (theorems 25 and 27). In sec. 5 we present some preliminary experiments of direct encoding of biological systems in HyLL. We end with an overview of related (sec. 6) and future work (sec. 7).

2   Hybrid linear logic

In this section we define HyLL, a conservative extension of intuitionistic first-order linear logic (ILL) [20] where the truth judgements are labelled by worlds representing constraints. Like in ILL, propositions are interpreted as *resources* which may be composed into a *state* using the usual linear connectives, and the

---

[3]Encodings -of a system or of a property of a system- in a logical framework are always required to be *adequate* in a strong sense sometimes called *representational adequacy* and illustrated here in sec. 4.1.

linear implication ($\multimap$) denotes a transition between states. The world label of a judgement represents a constraint on states and state transitions; particular choices for the worlds produce particular instances of HyLL. The common component in all the instances of HyLL is the proof theory, which we fix once and for all. We impose the following minimal requirement on the kinds of constraints that HyLL can deal with.

Definition 1. *A* constraint domain $\mathcal{W}$ *is a monoid structure* $\langle W, \cdot, \iota \rangle$. *The elements of W are called* worlds, *and the partial order* $\preceq : W \times W$ —*defined as* $u \preceq w$ *if there exists* $v \in W$ *such that* $u \cdot v = w$—*is the* reachability relation *in $\mathcal{W}$*.

The identity world ι is $\preceq$-initial and is intended to represent the lack of any constraints. Thus, the ordinary ILL is embeddable into any instance of HyLL by setting all world labels to the identity. When needed to disambiguate, the instance of HyLL for the constraint domain $\mathcal{W}$ will be written HyLL($\mathcal{W}$).

The reader may wonder why we require worlds to be monoids, instead of lattices, for example. The answer is to give a more general definition, suitable for rates constraints. One may then ask why we don't ask the worlds to be commutative. The answer is to allow lattices.

Atomic propositions are written using minuscule letters $(a, b, \ldots)$ applied to a sequence of *terms* $(s, t, \ldots)$, which are drawn from an untyped term language containing term variables $(x, y, \ldots)$ and function symbols $(f, g, \ldots)$ applied to a list of terms.. Non-atomic propositions are constructed from the connectives of first-order intuitionistic linear logic and the two hybrid connectives *satisfaction* (`at`), which states that a proposition is true at a given world $(w, u, v, \ldots)$, and *localization* ($\downarrow$), which binds a name for the (current) world the proposition is true at. The following grammar summarizes the syntax of HyLL propositions.

$$A, B, \ldots ::= a \, \vec{t} \mid A \otimes B \mid \mathbf{1} \mid A \multimap B \mid A \,\&\, B \mid \top \mid A \oplus B \mid \mathbf{0} \mid \,! A \mid \forall x. \, A \mid \exists x. \, A$$
$$\mid (A \, \mathtt{at} \, w) \mid \downarrow u. \, A \mid \forall u. \, A \mid \exists u. \, A$$

Note that in the propositions $\downarrow u. \, A$, $\forall u. \, A$ and $\exists u. \, A$, the scope of the world variable u is all the worlds occurring in A. World variables cannot be used in terms, and neither can term variables occur in worlds; this restriction is important for the modular design of HyLL because it keeps purely logical truth separate from constraint truth. We let α range over variables of either kind.

The unrestricted connectives $\wedge, \vee, \supset$, *etc.* of intuitionistic (non-linear) logic can also be defined in terms of the linear connectives and the exponential ! using any of the available embeddings of intuitionistic logic into linear logic, such as Girard's embedding [20].

2.1 Natural deduction for HyLL

We start with the judgements from linear logic [20] and enrich them with a modal situated truth. We present the syntax of hybrid linear logic in a natural deduction style, using Martin-Löf's principle of separating judgements and logical connectives. Instead of the ordinary mathematical judgement "A is true", for a proposition A, judgements of HyLL are of the form "A is true at world w", abbreviated as $A@w$. We use dyadic hypothetical derivations of the form $\Gamma \,;\, \Delta \vdash C@w$ where Γ and Δ are sets of judgements of the form $A@w$, with Δ being moreover a *multiset*. Γ is called the *unrestricted context*: its hypotheses can be consumed any number of times. Δ is a *linear context*: every hypothesis in it must be consumed singly in the proof. Note that in a judgement $A@w$ (as in a proposition $A \, \mathtt{at} \, w$), w can be any expression in $\mathcal{W}$, not only a variable.

The rules for the linear connectives are borrowed from [9] where they are discussed at length, so we omit a more thorough discussion here. The rules for the first-order quantifiers are completely standard. The unrestricted context Γ enjoys weakening and contraction; as usual, this is a theorem that is attested by the inference rules of the logic, and we omit its straightforward inductive proof. The notation $A[\tau/\alpha]$ stands for the replacement of all free occurrences of the variable α in A with the expression τ, avoiding capture. Note that the expressions in the rules are to be readen up to alpha-conversion.

Theorem 2 (structural properties)**.**
1. *If* $\Gamma \,;\, \Delta \vdash C@w$, *then* $\Gamma, \Gamma' \,;\, \Delta \vdash C@w$. *(weakening)*
2. *If* $\Gamma, A@u, A@u \,;\, \Delta \vdash C@w$, *then* $\Gamma, A@u \,;\, \Delta \vdash C@w$. *(contraction)*

3

Judgemental rules

$$\overline{\Gamma \; ; \; A@w \vdash A@w} \; \text{hyp} \qquad \overline{\Gamma, A@w \; ; \; \cdot \vdash A@w} \; \text{hyp!}$$

Multiplicatives

$$\frac{\Gamma \; ; \; \Delta \vdash A@w \quad \Gamma \; ; \; \Delta' \vdash B@w}{\Gamma \; ; \; \Delta, \Delta' \vdash A \otimes B@w} \; \otimes I \qquad \frac{\Gamma \; ; \; \Delta \vdash A \otimes B@w \quad \Gamma \; ; \; \Delta', A@w, B@w \vdash C@w'}{\Gamma \; ; \; \Delta, \Delta' \vdash C@w'} \; \otimes E$$

$$\frac{}{\Gamma \; ; \; \cdot \vdash \mathbf{1}@w} \; \mathbf{1}I \qquad \frac{\Gamma \; ; \; \Delta \vdash \mathbf{1}@w \quad \Gamma \; ; \; \Delta' \vdash C@w'}{\Gamma \; ; \; \Delta, \Delta' \vdash C@w'} \; \mathbf{1}E$$

$$\frac{\Gamma \; ; \; \Delta, A@w \vdash B@w}{\Gamma \; ; \; \Delta \vdash A \multimap B@w} \; \multimap I \qquad \frac{\Gamma \; ; \; \Delta \vdash A \multimap B@w \quad \Gamma \; ; \; \Delta' \vdash A@w}{\Gamma \; ; \; \Delta, \Delta' \vdash B@w} \; \multimap E$$

Additives

$$\frac{\Gamma \; ; \; \Delta \vdash A@w \quad \Gamma \; ; \; \Delta \vdash B@w}{\Gamma \; ; \; \Delta \vdash A \,\&\, B@w} \; \&I \qquad \frac{\Gamma \; ; \; \Delta \vdash A_1 \,\&\, A_2@w}{\Gamma \; ; \; \Delta \vdash A_i@w} \; \&E_i$$

$$\frac{\Gamma \; ; \; \Delta \vdash A_i@w}{\Gamma \; ; \; \Delta \vdash A_1 \oplus A_2@w} \; \oplus I_i \qquad \frac{\Gamma \; ; \; \Delta \vdash A \oplus B@w \quad \Gamma \; ; \; \Delta', A@w \vdash C@w' \quad \Gamma \; ; \; \Delta', B@w \vdash C@w'}{\Gamma \; ; \; \Delta, \Delta' \vdash C@w'} \; \oplus E$$

$$\frac{}{\Gamma \; ; \; \Delta \vdash \top@w} \; \top I \qquad \frac{\Gamma \; ; \; \Delta \vdash \mathbf{0}@w}{\Gamma \; ; \; \Delta, \Delta' \vdash C@w'} \; \mathbf{0}E$$

Quantifiers

$$\frac{\Gamma \; ; \; \Delta \vdash A@w}{\Gamma \; ; \; \Delta \vdash \forall \alpha. \, A@w} \; \forall I^\alpha \qquad \frac{\Gamma \; ; \; \Delta \vdash \forall \alpha. \, A@w}{\Gamma \; ; \; \Delta \vdash [\tau/x]A@w} \; \forall E$$

$$\frac{\Gamma \; ; \; \Delta \vdash [\tau/x]A@w}{\Gamma \; ; \; \Delta \vdash \exists \alpha. \, A@w} \; \exists I \qquad \frac{\Gamma \; ; \; \Delta \vdash \exists \alpha. \, A@w \quad \Gamma \; ; \; \Delta', A@w \vdash C@w'}{\Gamma \; ; \; \Delta, \Delta' \vdash C@w'} \; \exists E^\alpha$$

For $\forall I^\alpha$ and $\exists E^\alpha$, α is assumed to be fresh with respect to the conclusion.

For $\exists I$ and $\forall E$, τ stands for a term or world, as appropriate.

Exponentials

$$\frac{\Gamma \; ; \; \cdot \vdash A@w}{\Gamma \; ; \; \cdot \vdash !A@w} \; !I \qquad \frac{\Gamma \; ; \; \Delta \vdash !A@w \quad \Gamma, A@w \; ; \; \Delta' \vdash C@w'}{\Gamma \; ; \; \Delta, \Delta' \vdash C@w'} \; !E$$

Hybrid connectives

$$\frac{\Gamma \; ; \; \Delta \vdash A@w}{\Gamma \; ; \; \Delta \vdash (A \,\text{at}\, w)@w'} \; \text{at}I \qquad \frac{\Gamma \; ; \; \Delta \vdash (A \,\text{at}\, w)@w'}{\Gamma \; ; \; \Delta \vdash A@w} \; \text{at}E$$

$$\frac{\Gamma \; ; \; \Delta \vdash [w/u]A@w}{\Gamma \; ; \; \Delta \vdash \downarrow u. \, A@w} \; \downarrow I \qquad \frac{\Gamma \; ; \; \Delta \vdash \downarrow u. \, A@w}{\Gamma \; ; \; \Delta \vdash [w/u]A@w} \; \downarrow E$$

Figure 1: Natural deduction for HyLL

The full collection of inference rules are in fig. 1. A brief discussion of the hybrid rules follows. To introduce the *satisfaction* proposition $(A \mathrel{\mathtt{at}} w)$ (at any world w'), the proposition A must be true in the world w. The proposition $(A \mathrel{\mathtt{at}} w)$ itself is then true at any world, not just in the world w. In other words, $(A \mathrel{\mathtt{at}} w)$ carries with it the world at which it is true. Therefore, suppose we know that $(A \mathrel{\mathtt{at}} w)$ is true (at any world w'); then, we also know that $A@w$. These two introduction and elimination rules match up precisely to (de)construct the information in the $A@w$ judgement. The other hybrid connective of *localisation*, $\downarrow$, is intended to be able to name the current world. That is, if $\downarrow u.\,A$ is true at world w, then the variable u stands for w in the body A. This interpretation is reflected in its introduction rule $\downarrow I$. For elimination, suppose we have a proof of $\downarrow u.\,A@w$ for some world w. Then, we also know $[w/u]A@w$.

For the linear and unrestricted hypotheses, substitution is no different from that of the usual linear logic.

Theorem 3 (substitution)**.**

1. *If* $\Gamma \; ; \; \Delta \vdash A@u$ *and* $\Gamma \; ; \; \Delta', A@u \vdash C@w$, *then* $\Gamma \; ; \; \Delta, \Delta' \vdash C@w$.
2. *If* $\Gamma \; ; \; \cdot \vdash A@u$ *and* $\Gamma, A@u \; ; \; \Delta \vdash C@w$, *then* $\Gamma \; ; \; \Delta \vdash C@w$.

Proof sketch. By structural induction on the second given derivation in each case. $\qquad\square$

Note that the $\downarrow$ connective commutes with every propositional connective, including itself. That is, $\downarrow u.\,(A * B)$ is equivalent to $(\downarrow u.\,A) * (\downarrow u.\,B)$ for all binary connectives $*$, and $\downarrow u.\, * A$ is equivalent to $*(\downarrow u.\,A)$ for every unary connective $*$, assuming the commutation will not cause an unsound capture of u. It is purely a matter of taste where to place the $\downarrow$, and repetitions are harmless.

Theorem 4 (conservativity)**.** *Call a proposition or multiset of propositions* pure *if it contains no instance of the hybrid connectives and no instance of quantification over a world variable, and let Γ, Δ and A be pure. Then, $\Gamma \; ; \; \Delta \vdash A@w$ in HyLL iff $\Gamma \; ; \; \Delta \vdash A$ in intuitionistic linear logic.*

Proof. By structural induction on the given HyLL derivation. $\qquad\square$

2.2   Sequent calculus for HyLL

In this section, we give a sequent calculus presentation of HyLL and prove a cut-admissibility theorem. The sequent formulation in turn will lead to an analysis of the polarities of the connectives in order to get a focused sequent calculus that can be used to compile a logical theory into a system of derived inference rules with nice properties (sec. 3). For instance, if a given theory defines a transition system, then the derived rules of the focused calculus will exactly exhibit the same transitions. This is key to obtain the necessary representational adequacy theorems, as we shall see for the $S\pi$-calculus example chosen in this paper (sec. 4.1).

In the sequent calculus, we depart from the linear hypothetical judgement $\vdash$ which has only an "active" right-hand side to a sequent arrow $\Longrightarrow$ that has active zones on both sides. A rule that infers a proposition on the right of the sequent arrow is called a "right" rule, and corresponds exactly to the introduction rules of natural deduction. Dually, introductions on the left of the sequent arrow correspond to elimination rules of natural deduction; however, as all rules in the sequent calculus are introduction rules, the information flow in a sequent derivation is always in the same direction: from the conclusion to the premises, incidentally making the sequent calculus ideally suited for proof-search.

The full collection of rules of the HyLL sequent calculus is in fig. 2. There are only two structural rules: the init rule infers an atomic initial sequent, and the copy rule introduces a contracted copy of an unrestricted assumption into the linear context (reading from conclusion to premise). Weakening and contraction are admissible rules:

Theorem 5 (structural properties)**.**

1. *If* $\Gamma \; ; \; \Delta \Longrightarrow C@w$, *then* $\Gamma, \Gamma' \; ; \; \Delta \Longrightarrow C@w$. (weakening)
2. *If* $\Gamma, A@u, A@u \; ; \; \Delta \Longrightarrow C@w$, *then* $\Gamma, A@u \; ; \; \Delta \Longrightarrow C@w$. (contraction)

Proof. By straightforward structural induction on the given derivations. $\qquad\square$

Judgemental rules

$$\frac{}{\Gamma \,;\, a\ \vec{t}\ @u \Longrightarrow a\ \vec{t}\ @u}\ \text{init} \qquad \frac{\Gamma, A@u \,;\, \Delta, A@u \Longrightarrow C@w}{\Gamma, A@u \,;\, \Delta \Longrightarrow C@w}\ \text{copy}$$

Multiplicatives

$$\frac{\Gamma \,;\, \Delta \Longrightarrow A@w \quad \Gamma \,;\, \Delta' \Longrightarrow B@w}{\Gamma \,;\, \Delta, \Delta' \Longrightarrow A \otimes B@w}\ \otimes R \qquad \frac{\Gamma \,;\, \Delta, A@u, B@u \Longrightarrow C@w}{\Gamma \,;\, \Delta, A \otimes B@u \Longrightarrow C@w}\ \otimes L$$

$$\frac{}{\Gamma \,;\, \cdot \Longrightarrow \mathbf{1}@w}\ \mathbf{1}R \qquad \frac{\Gamma \,;\, \Delta \Longrightarrow C@w}{\Gamma \,;\, \Delta, \mathbf{1}@u \Longrightarrow C@w}\ \mathbf{1}L \qquad \frac{\Gamma \,;\, \Delta, A@w \Longrightarrow B@w}{\Gamma \,;\, \Delta \Longrightarrow A \multimap B@w}\ \multimap R$$

$$\frac{\Gamma \,;\, \Delta \Longrightarrow A@u \quad \Gamma \,;\, \Delta', B@u \Longrightarrow C@w}{\Gamma \,;\, \Delta, \Delta', A \multimap B@u \Longrightarrow C@w}\ \multimap L$$

Additives

$$\frac{}{\Gamma \,;\, \Delta \Longrightarrow \top@w}\ \top R \qquad \frac{\Gamma \,;\, \Delta \Longrightarrow A@w \quad \Gamma \,;\, \Delta \Longrightarrow B@w}{\Gamma \,;\, \Delta \Longrightarrow A \& B@w}\ \&R$$

$$\frac{\Gamma \,;\, \Delta, A_i@u \Longrightarrow C@w}{\Gamma \,;\, \Delta, \Delta', A_1 \& A_2@u \Longrightarrow C@w}\ \&L_i$$

$$\frac{\Gamma \,;\, \Delta \Longrightarrow A_i@w}{\Gamma \,;\, \Delta \Longrightarrow A_1 \oplus A_2@w}\ \oplus R_i \qquad \frac{}{\Gamma \,;\, \Delta, \mathbf{0}@u \Longrightarrow C@w}\ \mathbf{0}L$$

$$\frac{\Gamma \,;\, \Delta, A@u \Longrightarrow C@w \quad \Gamma \,;\, \Delta, B@u \Longrightarrow C@w}{\Gamma \,;\, \Delta, A \oplus B@u \Longrightarrow C@w}\ \oplus L$$

Quantifiers

$$\frac{\Gamma \,;\, \Delta \Longrightarrow A@w}{\Gamma \,;\, \Delta \Longrightarrow \forall \alpha.\ A@w}\ \forall R^\alpha \qquad \frac{\Gamma \,;\, \Delta, [\tau/\alpha]A@u \Longrightarrow C@w}{\Gamma \,;\, \Delta, \forall \alpha.\ A@u \Longrightarrow C@w}\ \forall L$$

$$\frac{\Gamma \,;\, \Delta \Longrightarrow [\tau/\alpha]A@w}{\Gamma \,;\, \Delta \Longrightarrow \exists \alpha.\ A@w}\ \exists R \qquad \frac{\Gamma \,;\, \Delta, A@u \Longrightarrow C@w}{\Gamma \,;\, \Delta, \exists \alpha.\ A@u \Longrightarrow C@w}\ \exists L^\alpha$$

For $\forall R^\alpha$ and $\exists L^\alpha$, α is assumed to be fresh with respect to the conclusion. For $\exists R$ and $\forall L$, τ stands for a term or world, as appropriate.

Exponentials

$$\frac{\Gamma \,;\, \cdot \Longrightarrow A@w}{\Gamma \,;\, \cdot \Longrightarrow\, !A@w}\ !R \qquad \frac{\Gamma, A@u \,;\, \Delta \Longrightarrow C@w}{\Gamma \,;\, \Delta, !A@u \Longrightarrow C@w}\ !L$$

Hybrid connectives

$$\frac{\Gamma \,;\, \Delta \Longrightarrow A@u}{\Gamma \,;\, \Delta \Longrightarrow (A\ \mathtt{at}\ u)@v}\ \mathtt{at}R \qquad \frac{\Gamma \,;\, \Delta, A@u \Longrightarrow C@w}{\Gamma \,;\, \Delta, (A\ \mathtt{at}\ u)@v \Longrightarrow C@w}\ \mathtt{at}L$$

$$\frac{\Gamma \,;\, \Delta \Longrightarrow [w/u]A@w}{\Gamma \,;\, \Delta \Longrightarrow \downarrow u.\ A@w}\ \downarrow R \qquad \frac{\Gamma \,;\, \Delta, [v/u]A@v \Longrightarrow C@w}{\Gamma \,;\, \Delta, \downarrow u.\ A@v \Longrightarrow C@w}\ \downarrow L$$

Figure 2: The sequent calculus for HyLL

.

The most important structural properties are the admissibility of the identity and the cut principles. The identity theorem is the general case of the init rule and serves as a global syntactic completeness theorem for the logic. Dually, the cut theorem below establishes the syntactic soundness of the calculus; moreover there is no cut-free derivation of $\cdot\ ;\ \cdot \Longrightarrow \mathbf{0}@w$, so the logic is also globally consistent.

Theorem 6 (identity)**.** $\Gamma\ ;\ A@w \Longrightarrow A@w$.

Proof. By induction on the structure of A (see sec. A.1). $\qquad\square$

Theorem 7 (cut)**.**
1. If $\Gamma\ ;\ \Delta \Longrightarrow A@u$ and $\Gamma\ ;\ \Delta', A@u \Longrightarrow C@w$, then $\Gamma\ ;\ \Delta, \Delta' \Longrightarrow C@w$.
2. If $\Gamma\ ;\ \cdot \Longrightarrow A@u$ and $\Gamma, A@u\ ;\ \Delta \Longrightarrow C@w$, then $\Gamma\ ;\ \Delta \Longrightarrow C@w$.

Proof. By lexicographic structural induction on the given derivations, with cuts of kind 2 additionally allowed to justify cuts of kind 1. The style of proof sometimes goes by the name of *structural cut-elimination* [9]. See sec. A.2 for the details. $\qquad\square$

We can use the admissible cut rules to show that the following rules are invertible: $\otimes L$, $\mathbf{1}L$, $\oplus L$, $\mathbf{0}L$, $\exists L$, $\multimap R$, $\&R$, $\top R$, and $\forall R$. In addition, the four hybrid rules, $\mathtt{at}R$, $\mathtt{at}L$, $\downarrow R$ and $\downarrow L$ are invertible. In fact, $\downarrow$ and $\mathtt{at}$ commute freely with all non-hybrid connectives:

Theorem 8 (Invertibility)**.** *The following rules are invertible:*
1. *On the right: $\&R$, $\top R$, $\multimap R$, $\forall R$, $\downarrow R$ and $\mathtt{at}R$;*
2. *On the left: $\otimes L$, $\mathbf{1}L$, $\oplus L$, $\mathbf{0}L$, $\exists L$, $!L$, $\downarrow L$ and $\mathtt{at}L$.*

Proof. See §A.3. $\qquad\square$

Theorem 9 (Correctness of the sequent calculus)**.**
1. *If $\Gamma\ ;\ \Delta \Longrightarrow C@w$, then $\Gamma\ ;\ \Delta \vdash C@w$. (soundness)*
2. *If $\Gamma\ ;\ \Delta \vdash C@w$, then $\Gamma\ ;\ \Delta \Longrightarrow C@w$. (completeness)*

Proof. See §A.4. $\qquad\square$

Corollary 10 (consistency)**.** *There is no proof of $\cdot\ ;\ \cdot \vdash \mathbf{0}@w$.*

Proof. See §A.4. $\qquad\square$

HyLL is conservative with respect to ordinary intuitionistic logic: as long as no hybrid connectives are used, the proofs in HyLL are identical to those in ILL [9]. The proof (omitted) is by simple structural induction.

Theorem 11 (conservativity)**.** *If $\Gamma\ ;\ \Delta \Longrightarrow_{HyLL} C@w$ is derivable, contains no occurrence of the hybrid connectives $\downarrow$, $\mathtt{at}$, $\forall u$ or $\exists u$, and each element of Γ and Δ is of the form $A@w$, then $\Gamma\ ;\ \Delta \Longrightarrow_{ILL} C$.*

An example of derived statements, true in every semantics for worlds, is the following:

Proposition 12 (relocalisation)**.**

$$\frac{\Gamma\ ;\ A_1@w_1 \cdots A_k@w_k \vdash B@v}{\Gamma\ ;\ A_1@u \cdot w_1 \cdots A_k@u \cdot w_k \vdash B@u \cdot v}$$

This property is particularly well suited to applications in biology.

In the rest of this paper we use the following derived connectives:

Definition 13 (modal connectives)**.**

$$\Box A \triangleq \downarrow u. \forall w. (A \text{ at } u \cdot w) \qquad \Diamond A \triangleq \downarrow u. \exists w. (A \text{ at } u \cdot w)$$
$$\rho_v A \triangleq \downarrow u. (A \text{ at } u \cdot v) \qquad \dagger A \triangleq \forall u. (A \text{ at } u)$$

The connective ρ represents a form of delay. Note its derived right rule:

$$\frac{\Gamma \ ; \ \Delta \vdash A @ w \cdot v}{\Gamma \ ; \ \Delta \vdash \rho_v A @ w} \ \rho R$$

The proposition $\rho_v A$ thus stands for an *intermediate state* in a transition to A. Informally it can be thought to be "v before A"; thus, $\forall v. \ \rho_v A$ represents *all* intermediate states in the path to A, and $\exists v. \ \rho_v A$ represents *some* such state. The modally unrestricted proposition $\dagger A$ represents a resource that is consumable in any world; it is mainly used to make transition rules applicable at all worlds.

It is worth remarking that HyLL proof theory can be seen as at least as powerful as (the linear restriction of) intuitionistic S5 [39]:

Theorem 14 (HyLL is S5)**.** *The following sequent is derivable:* $\cdot \ ; \ \Diamond A @ w \Longrightarrow \Box \Diamond A @ w.$

Proof. See §A.5. □

Obviously HyLL is more expressive as it allows direct manipulation of the worlds using the hybrid connectives: for example, the ρ connective is not definable in S5.

Let us elaborate a little bit on this point, together with the natural concerns of allowing predicates on worlds in the propositions.

First of all, let us note that allowing predicates on worlds anywhere in the propositions would yield the following false rule:

$$\frac{\Gamma \ ; \ \Delta \vdash (w \neq w) @ w \quad \Gamma \ ; \ \Delta' \vdash A @ w}{\Gamma \ ; \ \Delta, \Delta' \nvdash (\downarrow u. (u \neq w) \otimes A) @ w}$$

It seems that allowing predicates on worlds in the propositions is only possible in a restricted form, by adding constrained conjunction and implication as done in CILL [38] or η [14]. Following these works, we could allow the following expressions in the propositions:

$A, B, \ldots ::= \ldots \mid (!\, wp) \otimes A \mid (!\, wp) \multimap B$

where wp is any predicate on worlds such as $w \neq w'$ or $w \leq w'$, for example.

We might have chosen to define a modal connective at', instead of at, with the following rules:

$$\frac{\Gamma \ ; \ \Delta \Longrightarrow A @ u \quad (w \neq u)}{\Gamma \ ; \ \Delta \Longrightarrow (A \text{ at}' u) @ v} \ \text{at}'R \quad \frac{\Gamma \ ; \ \Delta, A @ u \Longrightarrow C @ w \quad (w \neq u)}{\Gamma \ ; \ \Delta, (A \text{ at}' u) @ v \Longrightarrow C @ w} \ \text{at}'L$$

Remarks:

1. If worlds are just monoid (not group), then S4 can be encoded in HyLL extended with constrained implication as above.

2. If worlds are groups (i.e. ... $\cdot$ admits an inverse, i.e. ... W is a right cumulative magma), then S5 can be encoded in HyLL with the modal connective at' defined above, instead of at.

3. If worlds are Kripke frames (i.e. total and symetric) then the relation $\leq$ on worlds can be internalized by a $\text{at}R$ rule.

As Alex Simpson proved in his PhD thesis the cut elimination theorem for any intuitionistic modal logic, we can be confident that the cut elimination theorem can be proven for HyLL with the modal connective at' instead of at.

2.3 Temporal constraints

As a pedagogical example, consider the constraint domain $\mathcal{T} = \langle \mathbf{R}^+, +, 0 \rangle$ representing instants of time. This domain can be used to define the lifetime of resources, such as keys, sessions, or delegations of authority. Delay (defn. 13) in HyLL($\mathcal{T}$) represents intervals of time; $\rho_d\, A$ means "A will become available after delay d", similar to metric tense logic [33]. This domain is very permissive because addition is commutative, resulting in the equivalence of $\rho_u\, \rho_v\, A$ and $\rho_v\, \rho_u\, A$. The "forward-looking" connectives G and F of ordinary tense logic are precisely $\square$ and $\diamond$ of defn. 13.

In addition to the future connectives, the domain $\mathcal{T}$ also admits past connectives if we add saturating subtraction (*i.e.*, $a - b = 0$ if $b \geq a$) to the language of worlds. We can then define the duals H and P of G and F as:

$$H\, A \triangleq\, \downarrow u.\, \forall w.\, (A\; \texttt{at}\; u - w) \qquad P\, A \triangleq\, \downarrow u.\, \exists w.\, (A\; \texttt{at}\; u - w)$$

While this domain does not have any branching structure like CTL, it is expressive enough for many common idioms because of the branching structure of derivations involving $\oplus$. CTL reachability ("in some path in some future"), for instance, is the same as our $\diamond$; similarly CTL steadiness ("in some path for all futures") is the same as $\square$. CTL stability ("in all paths in all futures"), however, has no direct correspondance in HyLL. Note that model checking cannot cope with temporal expresssions involving the "in all path" notion anyway. Thus approaches using ordinary temporal logics and model checking, like BIOCHAM, for example, cannot deal with those expressions either.

On the other hand, the availability of linear reasoning makes certain kinds of reasoning in HyLL much more natural than in ordinary temporal logics. One important example is of *oscillation* between states in systems with kinetic feedback. In a temporal specification language such as BIOCHAM [8], only finite oscillations are representable using a nested syntax, while in HyLL we use a simple bi-implication; for example, the oscillation between A and B with delay d is represented by the rule $\dagger(A \multimap \rho_d\, B)\; \&\; (B \multimap \rho_d\, A)$ (or $\dagger(A \multimap \diamond B)\; \&\; (B \multimap \diamond A)$ if the oscillation is aperiodic). If HyLL($\mathcal{T}$) were extended with constrained implication and conjunction in the style of CILL [38] or η [14], then we can define localized versions of $\square$ and $\diamond$, such as "A is true everywhere/somewhere in an interval". They would also allow us to define the "until" and "since" operators of linear temporal logic [23].

2.4 Probabilistic Constraints

The material in this section requires some background in probability and measure theory, and can be skipped at a first reading, without significant loss of continuity.

Transitions in practice rarely have precise delays. Phenomenological and experimental evidence is used to construct a probabilistic model of the transition system where the delays are specified as probability distributions of continuous variables.

The meaning of the random variables depends on the intended application. In the applications in the area of systems biology, the variables X can represent the concentration of a product, while in economics, X could be the duration of an activity, for example.

2.5 General Case

Let us recall some basic definitions in probability theory. The probability of X being in A is defined by: $\mu_X(A) = \int_{x \in A} \mu_X dx$. For example: $\mathrm{Prob}(X \leq x) = \mu_X(x) = \int_{-\infty}^{x} \mu_X dt$.

Fact 15 (see [18, 37])**.** *If X and Y are independent random variables in $\mathbb{R}$, with distribution μ_X and μ_Y, respectively, then the distribution μ_{X+Y} of the random variable $X + Y$ is given by*

$$\mu_{X+Y}(A) = \mu_X * \mu_Y(A) = \int_{\{x+y \in A\}} \mu_X(dx) \otimes \mu_Y(dy)$$

for all Borel [4] *subset A of $\mathbb{R}$.*

---

[4]The set of Borel events is the set of the Lebesgue measurable functions.

The space of probability distributions of random variables, together with the convolution operator $*$ and the Dirac mass at 0 (δ_0)[5], as neutral element, forms a monoid. More precisely:

Definition 16. *The* probabilities domain $\mathcal{P}$ *is the monoid* $\langle \mathbf{M_1}(\mathbb{R}), *, \delta_0 \rangle$ *where* $\mathbf{M_1}(\mathbb{R})$ *is the set of the Borel probability measures over* $\mathbb{R}$ *and* δ_0 *is Dirac mass at 0. The instance HyLL[$\mathcal{P}$] will sometimes be called "probabilistic hybrid linear logic".*

An element $w = \mu_X(x)$ of a world $\mathcal{P}$ thus represents the probability of X to have its value in the interval $[-\infty, x]$. $\bar{w} = 1 - \mu_X(x)$ represents the probability of X to have its value greater than x. $A@\bar{w}$ therefore means 'A is true with probability greater than x".

2.6 Markov Processes

The standard model of stochastic transition systems is continuous time Markov chains (CTMCs) where the delays of transitions between states are distributed according to the Markov assumption of memorylessness (Markov processes) with the further condition that their state-space are countable sets [37].

Fact 17 (see [17, 37]). *Given a continuous-time Markov process $(X_t, t \geq 0)$ taking values in a measurable space $(E, \mathcal{E})$, the family $(P(t), t \geq 0)$ of linear operators on the set of bounded Borel functions $\mathcal{B}(E)$ defined by: for all $f \in \mathcal{B}(E)$ and for all $x \in E$,*

$$(P(t)f)(x) = \mathbf{E}[f(X_t) \mid X_0 = x],$$

where $\mathbf{E}$ is the expectation function, is a semigroup for the convolution: for all $s, t \geq 0$,

$$P(t + s) = P(t) * P(s)$$

with neutral element $P(0)$, the identity operator. When the process X is a Feller process (see [37], chapter 3, section 2), $(P(t), t \geq 0)$ is a Feller semigroup, i.e. strongly continuous and conservative.

For a given continuous-time Markov process $(X_t, t \geq 0)$, the associated monoid is the set $(P(t), t \geq 0)$ defined above. More precisely, we can define the Markov domain as follows:

Definition 18. *For a given continuous-time Markov process $(X_t, t \geq 0)$, taking values in a measurable space $(E, \mathcal{E})$, the* Markov domain $\mathcal{M}$ *is the monoid* $\langle (P(t), t \geq 0), *, P(0) \rangle$ *where $(P(t), t \geq 0)$ is the sub-markov semigroup of linear operators on the set of bounded Borel functions $\mathcal{B}(E)$ defined by for all $f \in \mathcal{B}(E)$ and for all $x \in E$, $(P(t)f)(x) = \mathbf{E}[f(X_t) \mid X_0 = x]$. The instance HyLL($\mathcal{M}$) will sometimes be called "Markov hybrid linear logic".*

In the above definition, f can be any function in $\mathcal{B}(E)$, and we have

$$(P(t)f)(x) = \mathbf{E}[f(X_t) \mid X_0 = x].$$

An element $w = P(t)$ of $\mathcal{M}$ represents a function which associates to any function f (where $f \in \mathcal{B}(E)$) and to any initial value x for the variable X_t, the expectation of $f(X_t)$, knowing that $X_0 = x$. We can choose $f = \mathbf{1}_A$: the indicator (i.e. characteristic) function of a set A. In this case (recalling that $\mathbf{E}(\mathbf{1}_A) = P(A)$),

$$(P(t)f)(x) = (P(t)\mathbf{1}_A)(x) = \mathbf{E}[\mathbf{1}_A(X_t) \mid X_0 = x] = P\{X_t \in A \mid X_0 = x\}$$

For example, for $A = [-\infty, y]$: $(P(t)\mathbf{1}_A)(x) = P\{X_t \leq y \mid X_0 = x\} = \mathbf{F}_{X_t|X_0=x}(y)$ where $\mathbf{F}$ is the cumulative distribution function of X_t. Other interesting examples for f are the square function $\mathrm{sq}(y) = y^2$ and the identity function $\mathrm{id}(y) = y$. Using these functions, we can define the variance of X_t:

$$(P(t)\,\mathrm{sq})(x) - (P(t)\,\mathrm{id})^2(x) = \mathbf{E}(X_t^2 \mid X_0 = x) - (\mathbf{E}(X_t \mid X_0 = x))^2 = \mathbf{Var}(X_t \mid X_0 = x)$$

In principle, using suitable functions f, we should be able to define *any descriptors of the probability distribution of our variable X_t.*

The meaning of $A@w$ varies depending on the choice for the function f. For example, we have seen that in the case of $f = \mathbf{1}_{[-\infty, y]}$, $P(t)f(x) = \mathbf{F}_{X_t|X_0=x}(y)$. In this case, $A@w$ means "A is true with probability less than y at time t". For $\bar{w} = 1 - P(t)$, $A@\bar{w}$ means "A is true with probability greater than y at time t".

---

[5]The Dirac mass at 0 δ_0 is not a real function but a generalized function. It is just a measure.

Rates The cumulative distributions of the continuous time Markov chains (CTMCs) used in $S\pi$ are exponential [32]. More precisely: $\mathrm{Prob}(X_t \leq x + rt \mid X_0 = x) = \mathbf{F}_{X_t|X_0=x}(x + rt) = 1 - e^{rt}$, where r (rates) are functions depending on the time t. In this case, we can use $f = \mathbf{1}_{[-\infty, x+rt]}$. However, it is simpler to work in $\mathrm{HyLL}(\mathcal{P})$ (defn. 16) and define the worlds by particularizing the general case of probabilities domains to the case where the cumulative distributions of all variables X are exponential with the above meaning. The worlds w will therefore be defined by $w = \mu_X(x + rt) = \mathrm{Prob}(X_t \leq x + rt \mid X_0 = x) = \mathbf{F}_{X_t|X_0=x}(x + rt) = 1 - e^{rt}$.

3   Focusing

As HyLL is intended to represent transition systems adequately, it is crucial that HyLL derivations in the image of an encoding have corresponding transitions. However, transition systems are generally specified as rewrite algebras over an underlying congruence relation. These congruences have to be encoded propositionally in HyLL, so a HyLL derivation will generally require several inference rules to implement a single transition; moreover, several trivially different reorderings of these "micro" inferences would correspond to the same transition. It is therefore futile to attempt to define an operational semantics directly on HyLL inferences.

We restrict the syntax to focused derivations [1], which ignores many irrelevant rule permutations in a sequent proof and divides the proof into clear *phases* that define the grain of atomicity. The logical connectives are divided into two classes, *negative* and *positive*, and rule permutations for connectives of like polarity are confined to *phases*. A *focused derivation* is one in which the positive and negative rules are applied in alternate maximal phases in the following way: in the *active* phase, all negative rules are applied (in irrelevant order) until no further negative rule can apply; the phase then switches and one positive proposition is selected for *focus*; this focused proposition is decomposed under focus (*i.e.*, the focus persists unto its sub-propositions) until it becomes negative, and the phase switches again.

As noted before, the logical rules of the hybrid connectives $\mathtt{at}$ and $\downarrow$ are invertible, so they can be considered to have both polarities. It would be valid to decide a polarity for each occurrence of each hybrid connective independently; however, as they are mainly intended for book-keeping during logical reasoning, we define the polarity of these connectives in the following *parasitic* form: if its immediate subformula is positive (resp. negative) connective, then it is itself positive (resp. negative). These connectives therefore become invisible to focusing. This choice of polarity can be seen as a particular instance of a general scheme that divides the $\downarrow$ and $\mathtt{at}$ connectives into two polarized forms each. To complete the picture, we also assign a polarity for the atomic propositions; this restricts the shape of focusing phases further [10]. The full syntax of positive $(P, Q, \ldots)$ and negative $(M, N, \ldots)$ propositions is as follows:

$$P, Q, \ldots ::= p\,\vec{t} \mid P \otimes Q \mid \mathbf{1} \mid P \oplus Q \mid \mathbf{0} \mid\, !N \mid \exists\alpha.\ P \mid \downarrow u.\ P \mid (P\ \mathtt{at}\ w) \mid \Downarrow N$$
$$N, M, \ldots ::= n\,\vec{t} \mid N \mathbin{\&} N \mid \top \mid P \multimap N \mid \forall\alpha.\ N \mid \downarrow u.\ N \mid (N\ \mathtt{at}\ w) \mid \Uparrow P$$

The two syntactic classes refer to each other via the new connectives $\Uparrow$ and $\Downarrow$. Sequents in the focusing calculus are of the following forms.

$$\left. \begin{array}{l} \Gamma\ ;\ \Delta\ ;\ \Omega \Longrightarrow \cdot\ ;\ P@w \\ \Gamma\ ;\ \Delta\ ;\ \Omega \Longrightarrow N@w\ ;\ \cdot \end{array} \right\} \text{active} \qquad \left. \begin{array}{l} \Gamma\ ;\ \Delta\ ;\ \left[N@u\right] \Longrightarrow P@w \\ \Gamma\ ;\ \Delta \Longrightarrow \left[P@w\right] \end{array} \right\} \text{focused}$$

In each case, Γ and Δ contain only negative propositions (*i.e.*, of the form $N@u$) and Ω only positive propositions (*i.e.*, of the form $P@u$). The full collection of inference rules are in fig. 3. The sequent form $\Gamma\ ;\ \Delta\ ;\ \cdot \Longrightarrow \cdot\ ;\ P@w$ is called a *neutral sequent*; from such a sequent, a left or right focused sequent is produced with the rules lf, cplf or rf. Focused logical rules are applied (non-deterministically) and focus persists unto the subformulas of the focused proposition as long as they are of the same polarity; when the polarity switches, the result is an active sequent, where the propositions in the innermost zones are decomposed in an irrelevant order until once again a neutral sequent results.

Soundness of the focusing calculus with respect to the ordinary sequent calculus is immediate by simple structural induction. In each case, if we forget the additional structure in the focused derivations, then we

Focused logical rules

$$\frac{}{\Gamma\;;\;[n\,\vec{t}@w]\Longrightarrow\,\Downarrow n\,\vec{t}@w}\ \mathrm{li}\qquad
\frac{\Gamma\;;\;\Delta\;;\;P@u\Longrightarrow\cdot\;;\;Q@w}{\Gamma\;;\;\Delta\;;\;[\Uparrow P@u]\Longrightarrow Q@w}\ \Uparrow L\qquad
\frac{\Gamma\;;\;\Delta\;;\;[N_i@u]\Longrightarrow Q@w}{\Gamma\;;\;\Delta\;;\;[N_1\,\&\,N_2@u]\Longrightarrow Q@w}\ \&L_i$$

$$\frac{\Gamma\;;\;\Delta\Longrightarrow[P@u]\quad\Gamma\;;\;\Xi\;;\;[N@u]\Longrightarrow Q@w}{\Gamma\;;\;\Delta,\Xi\;;\;[P\multimap N@u]\Longrightarrow Q@w}\ \multimap L\qquad
\frac{\Gamma\;;\;\Delta\;;\;[[\tau/\alpha]N@u]\Longrightarrow Q@w}{\Gamma\;;\;\Delta\;;\;[\forall\alpha.\ N@u]\Longrightarrow Q@w}\ \forall L$$

$$\frac{\Gamma\;;\;\Delta\;;\;[[v/u]N@v]\Longrightarrow Q@w}{\Gamma\;;\;\Delta\;;\;[\downarrow u.\ N@v]\Longrightarrow Q@w}\ \downarrow LF\qquad
\frac{\Gamma\;;\;\Delta\;;\;[N@u]\Longrightarrow Q@w}{\Gamma\;;\;\Delta\;;\;[(N\ \mathtt{at}\ u)@v]\Longrightarrow Q@w}\ \mathtt{at}LF$$

$$\frac{}{\Gamma\;;\;\Uparrow p\,\vec{t}@w\Longrightarrow[p\,\vec{t}@w]}\ \mathrm{ri}\qquad
\frac{\Gamma\;;\;\Delta\;;\;\cdot\Longrightarrow N@w\;;\;\cdot}{\Gamma\;;\;\Delta\Longrightarrow[\Downarrow N@w]}\ \Downarrow R\qquad
\frac{\Gamma\;;\;\Delta\Longrightarrow[P@w]\quad\Gamma\;;\;\Xi\Longrightarrow[Q@w]}{\Gamma\;;\;\Delta,\Xi\Longrightarrow[P\otimes Q@w]}\ \otimes R$$

$$\frac{}{\Gamma\;;\;\cdot\Longrightarrow[\mathbf{1}@w]}\ \mathbf{1}R\qquad
\frac{\Gamma\;;\;\Delta\Longrightarrow[P_i@w]}{\Gamma\;;\;\Delta\Longrightarrow[P_1\oplus P_2@w]}\ \oplus R_i\qquad
\frac{\Gamma\;;\;\cdot\;;\;\cdot\Longrightarrow N@w\;;\;\cdot}{\Gamma\;;\;\cdot\Longrightarrow[!N]@w}\ !R$$

$$\frac{\Gamma\;;\;\Delta\Longrightarrow[[\tau/\alpha]P@w]}{\Gamma\;;\;\Delta\Longrightarrow[\exists\alpha.\ P@w]}\ \exists R\qquad
\frac{\Gamma\;;\;\Delta\Longrightarrow[[w/u]P@w]}{\Gamma\;;\;\Delta\Longrightarrow[\downarrow u.\ P@w]}\ \downarrow RF\qquad
\frac{\Gamma\;;\;\Delta\Longrightarrow[P@u]}{\Gamma\;;\;\Delta\Longrightarrow[(P\ \mathtt{at}\ u)@w]}\ \mathtt{at}RF$$

Active logical rules (R of the form $\cdot\;;\;Q@w$ or $N@w\;;\;\cdot$, and L of the form $\Gamma\;;\;\Delta\;;\;\Omega$)

$$\frac{\mathsf{L},P@u,Q@u\Longrightarrow\mathsf{R}}{\mathsf{L},P\otimes Q@u\Longrightarrow\mathsf{R}}\ \otimes L\qquad
\frac{\mathsf{L}\Longrightarrow\mathsf{R}}{\mathsf{L},\mathbf{1}@u\Longrightarrow\mathsf{R}}\ \mathbf{1}L\qquad
\frac{\mathsf{L},P@u\Longrightarrow\mathsf{R}\quad\mathsf{L},Q@u\Longrightarrow\mathsf{R}}{\mathsf{L},P\oplus Q@u\Longrightarrow\mathsf{R}}\ \oplus L\qquad
\frac{}{\mathsf{L},\mathbf{0}@u\Longrightarrow\mathsf{R}}\ \mathbf{0}L$$

$$\frac{\mathsf{L},[v/u]P@v\Longrightarrow\mathsf{R}}{\mathsf{L},\downarrow u.\ P@v\Longrightarrow\mathsf{R}}\ \downarrow LA\qquad
\frac{\mathsf{L},P@u\Longrightarrow\mathsf{R}}{\mathsf{L},(P\ \mathtt{at}\ u)@v\Longrightarrow\mathsf{R}}\ \mathtt{at}LA\qquad
\frac{\mathsf{L},P@u\Longrightarrow\mathsf{R}}{\mathsf{L},\exists\alpha.\ P@u\Longrightarrow\mathsf{R}}\ \exists L^\alpha$$

$$\frac{\Gamma,N@u\;;\;\Delta\;;\;\Omega\Longrightarrow\mathsf{R}}{\Gamma\;;\;\Delta\;;\;\Omega,!N@u\Longrightarrow\mathsf{R}}\ !L\qquad
\frac{\Gamma\;;\;\Delta,N@w\;;\;\Omega\Longrightarrow\mathsf{R}}{\Gamma\;;\;\Delta\;;\;\Omega,\Downarrow N@w\Longrightarrow\mathsf{R}}\ \Downarrow L\qquad
\frac{\Gamma\;;\;\Delta,\Uparrow p\,\vec{t}\;;\;\Omega\Longrightarrow\mathsf{R}}{\Gamma\;;\;\Delta\;;\;\Omega,p\,\vec{t}@w\Longrightarrow\mathsf{R}}\ \mathrm{lp}$$

$$\frac{\mathsf{L}\Longrightarrow M@w\;;\;\cdot\quad\mathsf{L}\Longrightarrow N@w\;;\;\cdot}{\mathsf{L}\Longrightarrow M\,\&\,N@w\;;\;\cdot}\ \&R\qquad
\frac{}{\mathsf{L}\Longrightarrow\top@w\;;\;\cdot}\ \top R\qquad
\frac{\mathsf{L},P@w\Longrightarrow N@w\;;\;\cdot}{\mathsf{L}\Longrightarrow P\multimap N@w\;;\;\cdot}\ \multimap R$$

$$\frac{\mathsf{L}\Longrightarrow[w/u]N@w\;;\;\cdot}{\mathsf{L}\Longrightarrow\downarrow u.\ N@w\;;\;\cdot}\ \downarrow RA\qquad
\frac{\mathsf{L}\Longrightarrow N@u}{\mathsf{L}\Longrightarrow(N\ \mathtt{at}\ u)@w}\ \mathtt{at}RA\qquad
\frac{\mathsf{L}\Longrightarrow N@u\;;\;\cdot}{\mathsf{L}\Longrightarrow\forall\alpha.\ N@u\;;\;\cdot}\ \forall R^\alpha$$

$$\frac{\mathsf{L}\Longrightarrow\cdot\;;\;P@w}{\mathsf{L}\Longrightarrow\Uparrow P@w\;;\;\cdot}\ \Uparrow R\qquad
\frac{\mathsf{L}\Longrightarrow\cdot\;;\;\Downarrow n\,\vec{t}@w}{\mathsf{L}\Longrightarrow n\,\vec{t}@w\;;\;\cdot}\ \mathrm{rp}$$

Focusing decisions (L of the form $\Gamma\;;\;\Delta$)

$$\frac{\Gamma\;;\;\Delta\;;\;[N@u]\Longrightarrow Q@w\quad N\ \text{not}\ \Uparrow p\,\vec{t}}{\Gamma\;;\;\Delta,N@u\;;\;\cdot\Longrightarrow\cdot\;;\;Q@w}\ \mathrm{lf}\qquad
\frac{\Gamma,N@u\;;\;\Delta\;;\;[N@u]\Longrightarrow Q@w}{\Gamma,N@u\;;\;\Delta\;;\;\cdot\Longrightarrow\cdot\;;\;Q@w}\ \mathrm{cplf}$$

$$\frac{\Gamma\;;\;\Delta\Longrightarrow[P@w]\quad P\ \text{not}\ \Downarrow n\,\vec{t}}{\Gamma\;;\;\Delta\;;\;\cdot\Longrightarrow\cdot\;;\;P@w}\ \mathrm{rf}$$

Figure 3: Focusing rules for HyLL.

obtain simply an unfocused proof. We omit the obvious theorem. Completeness, on the other hand, is a hard result. We omit the proof because focusing is by now well known for linear logic, with a number of distinct proofs via focused cut-elimination (see *e.g.* the detailed proof in [10]). The hybrid connectives pose no problems because they allow all cut-permutations.

Theorem 19 (focusing completeness)**.** *Let Γ^- and C^- @w be negative polarizations of Γ and C @w (that is, adding $\Uparrow$ and $\Downarrow$ to make C and each proposition in Γ negative) and Δ^+ be a positive polarization of Δ. If $\Gamma \;;\; \Delta \Longrightarrow C$ @w, then $\cdot \;;\; \cdot \;;\; !\,\Gamma^-, \Delta^+ \Longrightarrow C^-$ @$w \;;\; \cdot$.*

4 Encoding the synchronous stochastic π-calculus

In this section, we shall illustrate the use of HyLL($\mathcal{P}$) as a logical framework for constrained transition systems by encoding the syntax and the operational semantics of the synchronous stochastic π-calculus ($S\pi$), which extends the ordinary π-calculus by assigning to every channel and internal action an *inherent* rate of synchronization. In $S\pi$, each rate characterises an exponential distribution[32], such that the probability of a reaction with rate r is given by $\mathrm{Prob}(X_t \leq x + rt \mid X_0 = x) = \mathbf{F}_{X_t \mid X_0 = x}(x + rt) = 1 - e^{rt}$, where the *rates* r are functions depending on the time t. We have seen in sec. 2.4 that, in this case, we can use a particular instance of HyLL($\mathcal{P}$), where the worlds w are $\mu_X(x + rt)$, representing the probability of X to have its value less or equal than $x + rt$. Note that the distributions have the same shape for any variables X_t; They only depend on *rates* $r(t)$ and time t. We shall use this fact to encode $S\pi$ in HyLL($\mathcal{P}$): a $S\pi$ reaction with rate r will be encoded by a transition of probability $w = \mu_X(x + rt) = \mathrm{Prob}(X_t \leq x + rt \mid X_0 = x) = \mathbf{F}_{X_t \mid X_0 = x}(x + rt) = 1 - e^{rt}$. In the rest of this section, worlds w of this shape, defined by a rate r, will simply be written r (see defn. 20 and defn. 22).

HyLL($\mathcal{P}$) can therefore be seen as a formal language for expressing $S\pi$ executions (traces). For the rest of this section we shall use $r, s, t, \ldots$ instead of $u, v, w, \ldots$ to highlight the fact that the worlds represent (probabilities defined by) rates, with the understanding that $\cdot$ is convolution (fact 15) and ι is $\ominus$. We don't directly use rates because the syntax and transitions of $S\pi$ are given generically for a π-calculus with labelled actions, and it is only the interpretation of the labels that involves probabilities.

We first summarize the syntax of $S\pi$, which is a minor variant of a number of similar presentations such as [32]. For hygienic reasons we divide entities into the syntactic categories of *processes* ($P, Q, \ldots$) and *sums* ($M, N, \ldots$), defined as follows. We also include environments of recursive definitions (E) for constants.

(Processes)	$P, Q, \ldots$	$::= \nu_r\, P \mid P \mid Q \mid 0 \mid X_n\, x_1 \cdots x_n \mid M$
(Sums)	$M, N, \ldots$	$::= !x(y).\, P \mid ?x.\, P \mid \tau_r.\, P \mid M + N$
(Environments)	E	$::= E, X_n \triangleq P \mid \cdot$

$P \mid Q$ is the parallel composition of P and Q, with unit 0. The restriction $\nu_r\, P$ abstracts over a free channel x in the process $P\, x$. We write the process using higher-order abstract syntax [29], *i.e.*, P in $\nu_r\, P$ is (syntactically) a function from channels to processes. This style lets us avoid cumbersome binding rules in the interactions because we reuse the well-understood binding structure of the λ-calculus. A similar approach was taken in the earliest encoding of (ordinary) π-calculus in (unfocused) linear logic [26], and is also present in the encoding in CLF [7].

A sum is a non-empty choice ($+$) over terms with *action prefixes*: the output action $!x(y)$ sends y along channel x, the input action $?x$ reads a value from x (which is applied to its continuation process), and the internal action τ_r has no observable I/O behaviour. Replication of processes happens via guarded recursive definitions [27]; in [36] it is argued that they are more practical for programming than the replication operator !. In a definition $X_n \triangleq P$, X_n denotes a (higher-order) defined constant of arity n; given channels $x_1, \ldots, x_n$, the process $X_n\, x_1 \cdots x_n$ is synonymous with $P\, x_1 \cdots x_n$. The constant X_n may occur on the right hand side of any definition in E, including in its body P, as long as it is prefixed by an action; this prevents infinite recursion without progress.

Interactions are of the form $E \vdash P \xrightarrow{r} Q$ denoting a transition from the process P to the process Q, in a global environment E, by performing an action at rate r. Each channel x is associated with an inherent

Interactions

$$\frac{}{!x(y).\, P + M \mid ?x.\, Q + M' \xrightarrow{\text{rate}(x)} P \mid Q\, y}\ \text{SYN} \qquad \frac{}{\tau_r.\, P \xrightarrow{r} P}\ \text{INT}$$

$$\frac{P \xrightarrow{r} P'}{P \mid Q \xrightarrow{r} P' \mid Q}\ \text{PAR} \qquad \frac{\forall x_s.\left(P\, x \xrightarrow{r} Q\, x \right)}{\nu_s\, P \xrightarrow{r} \nu_s\, Q}\ \text{RES} \qquad \frac{P \xrightarrow{r} Q \quad P \equiv P' \quad Q \equiv Q'}{P' \xrightarrow{r} Q'}\ \text{CONG}$$

- -

Congruence

$$\frac{}{P \mid 0 \equiv P} \qquad \frac{}{P \mid Q \equiv Q \mid P} \qquad \frac{}{P \mid (Q \mid R) \equiv (P \mid Q) \mid R} \qquad \frac{}{\nu_r\, 0 \equiv 0} \qquad \frac{X_n \triangleq P \in E}{E \vdash X_n\, x_1 \cdots x_n \equiv P\, x_1 \cdots x_n}$$

$$\frac{}{\nu_r(\lambda x.\nu_s(\lambda y.\, P)) \equiv \nu_s(\lambda y.\nu_r(\lambda x.\, P))} \qquad \frac{\forall x_r.\, (P\, x \equiv Q\, x)}{\nu_r\, P \equiv \nu_r\, Q} \qquad \frac{}{\nu_r(\lambda x.\, P \mid Q(x)) \equiv P \mid \nu_r\, Q}$$

$$\frac{P \equiv P'}{P \mid Q \equiv P' \mid Q} \qquad \frac{P \equiv P'}{!x(m).\, P \equiv !x(m).\, P'} \qquad \frac{\forall n.\, (P\, n \equiv Q\, n)}{?x.\, P \equiv ?x.\, Q} \qquad \frac{P \equiv P'}{\tau_r.\, P \equiv \tau_r.\, P'}$$

$$\frac{}{M + N \equiv N + M} \qquad \frac{}{M + (N + K) \equiv (M + N) + K} \qquad \frac{M \equiv M'}{M + N \equiv M' + N} \qquad \frac{M \equiv N}{M + N \equiv M}$$

Figure 4: Interactions and congruence in $S\pi$. The environment E is elided in most rules.

rate specific to the channel, and internal actions τ_r have rate r. The restriction $\nu_r\, P$ defines the rate of the abstracted channel as r.

The full set of interactions and congruences are in fig. 4. We generally omit the global environment E in the rules as it never changes. It is possible to use the congruences to compute a normal form for processes that are a parallel composition of sums and each reaction selects two suitable sums to synchronise on a channel until there are no further reactions possible; this refinement of the operational semantics is used in $S\pi$ simulators such as SPiM [31].

Definition 20 (syntax encoding).

1. *The encoding of the process P as a positive proposition, written $\llbracket P \rrbracket_{\mathrm{p}}$, is as follows (*sel* is a positive atom and* rt *a negative atom).*

$$\llbracket P \mid Q \rrbracket_{\mathrm{p}} = \llbracket P \rrbracket_{\mathrm{p}} \otimes \llbracket Q \rrbracket_{\mathrm{p}} \qquad\qquad \llbracket \nu_r\, P \rrbracket_{\mathrm{p}} = \exists x.\, !(\mathtt{rt}\, x \,\mathtt{at}\, r) \otimes \llbracket P\, x \rrbracket_{\mathrm{p}}$$
$$\llbracket 0 \rrbracket_{\mathrm{p}} = \mathbf{1} \qquad\qquad \llbracket X_n\, x_1 \cdots x_n \rrbracket_{\mathrm{p}} = X_n\, x_1 \cdots x_n$$
$$\llbracket M \rrbracket_{\mathrm{p}} = {\Downarrow}(\mathtt{sel} \multimap \llbracket M \rrbracket_{\mathrm{s}})$$

2. *The encoding of the sum M as a negative proposition, written $\llbracket M \rrbracket_{\mathrm{s}}$, is as follows (*out*,* in *and* tau *are positive atoms).*

$$\llbracket M + N \rrbracket_{\mathrm{s}} = \llbracket M \rrbracket_{\mathrm{s}} \,\&\, \llbracket N \rrbracket_{\mathrm{s}} \qquad\qquad \llbracket !x(m).\, P \rrbracket_{\mathrm{s}} = {\Uparrow}(\mathtt{out}\, x\, m \otimes \llbracket P \rrbracket_{\mathrm{p}})$$
$$\llbracket ?x.\, P \rrbracket_{\mathrm{s}} = \forall n.\, {\Uparrow}(\mathtt{in}\, x\, n \otimes \llbracket P\, n \rrbracket_{\mathrm{p}}) \qquad\qquad \llbracket \tau_r.\, P \rrbracket_{\mathrm{s}} = {\Uparrow}(\mathtt{tau}\, r \otimes \llbracket P \rrbracket_{\mathrm{p}})$$

3. *The encoding of the definitions E as a context, written $\llbracket E \rrbracket_{\mathrm{e}}$, is as follows.*

$$\llbracket E, X_n \triangleq P \rrbracket_{\mathrm{e}} = \llbracket E \rrbracket_{\mathrm{e}}, {\dagger}\forall x_1, \ldots, x_n.\, X_n\, x_1 \cdots x_n \multimapdotinv\multimapdot \llbracket P\, x_1 \cdots x_n \rrbracket_{\mathrm{p}}$$
$$\llbracket \cdot \rrbracket_{\mathrm{e}} = \cdot$$

where $P \multimapdotinv\multimapdot Q$ is defined as $(P \multimap {\Uparrow}Q) \,\&\, (Q \multimap {\Uparrow}P)$.

The encoding of processes is positive, so they will be decomposed in the active phase when they occur on the left of the sequent arrow, leaving a collection of sums. The encoding of restrictions will introduce a fresh unrestricted assumption about the rate of the restricted channel. Each sum encoded as a processes undergoes

a polarity switch because $\multimap$ is negative; the antecedent of this implication is a *guard* `sel`. This pattern of guarded switching of polarities prevents unsound congruences such as $!x(m).\,!y(n).\,P \equiv\, !y(n).\,!x(m).\,P$ that do not hold for the synchronous π calculus.[6] This guard also *locks* the sums in the context: the $S\pi$ interaction rules INT and SYN discard the non-interacting terms of the sum, so the environment will contain the requisite number of `sel`s only when an interaction is in progress. The action prefixes themselves are also synchronous, which causes another polarity switch. Each action releases a token of its respective kind—`out`, `in` or `tau`—into the context. These tokens must be consumed by the interaction before the next interaction occurs. For each action, the (encoding of the) continuation process is also released into the context.

The proof of the following congruence lemma is omitted. Because the encoding is (essentially) a $\otimes/\&$ structure, there are no distributive laws in linear logic that would break the process/sum structure.

Theorem 21 (congruence).
$E \vdash P \equiv Q$ *iff both* $[\![E]\!]_{\mathrm{e}} @\iota\ ;\ \cdot\ ;\ [\![P]\!]_{\mathrm{p}} @\iota \Longrightarrow \cdot\ ;\ [\![Q]\!]_{\mathrm{p}} @\iota$ *and* $[\![E]\!]_{\mathrm{e}} @\iota\ ;\ \cdot\ ;\ [\![Q]\!]_{\mathrm{p}} @\iota \Longrightarrow \cdot\ ;\ [\![P]\!]_{\mathrm{p}} @\iota.$

Now we encode the interactions. Because processes were lifted into propositions, we can be parsimonious with our encoding of interactions by limiting ourselves to the atomic interactions SYN and INT (below); the PAR, RES and CONG interactions will be ambiently implemented by the logic. Because there are no concurrent interactions—only one interaction can trigger at a time in a trace—the interaction rules must obey a locking discipline. We represent this lock as the proposition `act` that is consumed at the start of an interaction and produced again at the end. This lock also carries the net rate of the prefix of the trace so far: that is, an interaction $P \xrightarrow{r} Q$ will update the lock from `act`$@s$ to `act`$@s \cdot r$. The encoding of individual atomic interactions must also remove the `in`, `out` and `tau` tokens introduced in context by the interacting processes.

Definition 22 (interaction).
Let `inter` $\triangleq \dagger(\texttt{act} \multimap \Uparrow\texttt{int}\, \&\, \Uparrow\texttt{syn})$ *where* `act` *is a positive atom and* `int` *and* `syn` *are as follows:*

$$\texttt{int} \triangleq (\texttt{sel at }\iota) \otimes \Downarrow\forall r.\Big((\texttt{tau}\,r\,\texttt{at }\iota) \multimap \rho_r \Uparrow\texttt{act}\Big)$$

$$\texttt{syn} \triangleq (\texttt{sel} \otimes \texttt{sel at }\iota) \otimes \Downarrow\forall x, r, m.\Big((\texttt{out}\,x\,m \otimes \texttt{in}\,x\,m\,\texttt{at }\iota) \multimap \Downarrow(\texttt{rt}\,x\,\texttt{at }r) \multimap \rho_r \Uparrow\texttt{act}\Big).$$

The number of interactions that are allowed depend on the number of instances of `inter` in the linear context: each focus on `inter` implements a single interaction. If we are interested in all finite traces, we will add `inter` to the unrestricted context so it may be reused as many times as needed.

4.1 Representational adequacy.

Adequacy consists of two components: completeness and soundness. Completeness is the property that every $S\pi$ execution is obtainable as a HyLL derivation using this encoding, and is the comparatively simpler direction (see thm. 25). Soundness is the reverse property, and is false for unfocused HyLL as such. However, it *does* hold for focused proofs (see thm. 27). In both cases, we reason about the following canonical sequents of HyLL.

Definition 23. *The* canonical context of P, *written* $\wr P\wr$, *is given by:*

$$\wr X_n\,x_1\cdots x_n\wr = \Uparrow X_n\,x_1\cdots x_n \qquad \wr P \mid Q\wr = \wr P\wr, \wr Q\wr \qquad \wr 0\wr = \cdot \qquad \wr\nu_r\ P\wr = \wr P\,a\wr$$

$$\wr M\wr = \texttt{sel} \multimap [\![M]\!]_{\mathrm{s}}$$

For $\wr\nu_r\ P\wr$, *the right hand side uses a* fresh *channel* a *that is not free in the rest of the sequent it occurs in.*

As an illustration, take $P \triangleq\, !x(a).\,Q \mid ?x.\,R$. We have:

$$\wr P\wr = \quad \texttt{sel} \multimap \Uparrow(\texttt{out}\,x\,a \otimes [\![Q]\!]_{\mathrm{p}}),\ \texttt{sel} \multimap \forall y.\Uparrow(\texttt{in}\,x\,y \otimes [\![R\,y]\!]_{\mathrm{p}})$$

Obviously, the canonical context is what would be emitted to the linear zone at the end of the active phase if $[\![P]\!]_{\mathrm{p}}$ were to be present in the left active zone.

---

[6]Note: $(x \multimap a \otimes (x \multimap b \otimes c)) \multimap (x \multimap b \otimes (x \multimap a \otimes c))$ is not provable in linear logic.

Suppose $\mathsf{L} = \mathtt{rt}\,x@r, \mathtt{inter}@\iota$ and $\mathsf{R} = (\llbracket S \rrbracket_{\mathrm{p}} \ \mathtt{at}\ \iota) \otimes \mathtt{act}@t$. (All judgements $@\iota$ omitted.)

$$
\cfrac{
\cfrac{
\cfrac{
\cfrac{
\cfrac{
\cfrac{
\mathsf{L}\ ;\ \wr Q\wr, \wr R\,a\wr, \Uparrow\mathtt{act}@s \cdot r\ ;\ \cdot \Longrightarrow \cdot\ ;\ \mathsf{R}
}{
\mathsf{L}\ ;\ \begin{array}{l} \wr Q\wr, \Uparrow\mathtt{out}\,x\ a, \Uparrow\mathtt{in}\,x\ a, \wr R\,a\wr, \\ \forall x, r, m.\,((\mathtt{out}\,x\ m \otimes \mathtt{in}\,x\ m\ \mathtt{at}\ \iota) \multimap \Downarrow(\mathtt{rt}x\ \mathtt{at}\ r) \multimap \rho_r\,\mathtt{act})@s \end{array}\ ;\ \cdot \Longrightarrow \cdot\ ;\ \mathsf{R}
}\ {\scriptstyle 5}
}{
\mathsf{L}\ ;\ \begin{array}{l} \Uparrow\mathtt{out}\,x\ a, \wr Q\wr, \mathtt{sel} \multimap \forall y.\,\Uparrow(\mathtt{in}\,x\ y \otimes \llbracket R\,y \rrbracket_{\mathrm{p}}), \\ \Uparrow\mathtt{sel}, \forall x, r, m.\,((\mathtt{out}\,x\ m \otimes \mathtt{in}\,x\ m\ \mathtt{at}\ \iota) \multimap \Downarrow(\mathtt{rt}x\ \mathtt{at}\ r) \multimap \rho_r\,\mathtt{act})@s \end{array}\ ;\ \cdot \Longrightarrow \cdot\ ;\ \mathsf{R}
}\ {\scriptstyle 4}
}{
\mathsf{L}\ ;\ \begin{array}{l} \mathtt{sel} \multimap \Uparrow(\mathtt{out}\,x\ a \otimes \llbracket Q \rrbracket_{\mathrm{p}}), \mathtt{sel} \multimap \forall y.\,(\mathtt{in}\,x\ y \otimes \llbracket R\,y \rrbracket_{\mathrm{p}}), \\ \Uparrow\mathtt{sel}, \Uparrow\mathtt{sel}, \forall x, r, m.\,((\mathtt{out}\,x\ m \otimes \mathtt{in}\,x\ m\ \mathtt{at}\ \iota) \multimap \Downarrow(\mathtt{rt}x\ \mathtt{at}\ r) \multimap \rho_r\,\mathtt{act})@s \end{array}\ ;\ \cdot \Longrightarrow \cdot\ ;\ \mathsf{R}
}\ {\scriptstyle 3}
}{
\mathsf{L}\ ;\ \Uparrow\mathtt{act}@s, \mathtt{sel} \multimap \Uparrow(\mathtt{out}\,x\ a \otimes \llbracket Q \rrbracket_{\mathrm{p}}), \mathtt{sel} \multimap \forall y.\,\Uparrow(\mathtt{in}\,x\ y \otimes \llbracket R\,y \rrbracket_{\mathrm{p}})\ ;\ \big[\mathtt{inter}\big] \Longrightarrow \mathsf{R}
}\ {\scriptstyle 2}
}{
\mathsf{L}\ ;\ \Uparrow\mathtt{act}@s, \mathtt{sel} \multimap \Uparrow(\mathtt{out}\,x\ a \otimes \llbracket Q \rrbracket_{\mathrm{p}}), \mathtt{sel} \multimap \forall y.\,\Uparrow(\mathtt{in}\,x\ y \otimes \llbracket R\,y \rrbracket_{\mathrm{p}})\ ;\ \cdot \Longrightarrow \cdot\ ;\ \mathsf{R}
}\ {\scriptstyle 1}
}{
\mathsf{L}\ ;\ \Uparrow\mathtt{act}@s, \wr !x(a).\,Q \mid ?x.\,R\wr\ ;\ \cdot \Longrightarrow \cdot\ ;\ \mathsf{R}
}
$$

Steps

1: focus on $\mathtt{inter} \in \mathsf{L}$	3: $\mathtt{sel}$ for output + full phases	5: cleanup
2: select $\mathtt{syn}$ from $\mathtt{inter}$, active rules	4: $\mathtt{sel}$ for input + full phases	

Figure 5: Example interaction in the $S\pi$-encoding.

Definition 24. *A neutral sequent is* canonical *iff it has the shape*

$$\llbracket E \rrbracket_{\mathrm{e}}, \mathtt{rates}, \mathtt{inter}@\iota\ ;\ \Uparrow\mathtt{act}@s, \wr P_1 \mid \cdots \mid P_k\wr@\iota\ ;\ \cdot \Longrightarrow \cdot\ ;\ (\llbracket Q \rrbracket_{\mathrm{p}} \ \mathtt{at}\ \iota) \otimes \mathtt{act}@t$$

where $\mathtt{rates}$ *contains elements of the form* $\mathtt{rt}\,x@r$ *defining the rate of the channel x as r, and all free channels in* $\llbracket E \rrbracket_{\mathrm{e}}, \wr P_1 \mid \cdots \mid P_k \mid Q\wr$ *have a single such entry in* $\mathtt{rates}$.

Figure 5 contains an example of a derivation for a canonical sequent involving P. Focusing on any (encoding of a) sum in $\wr P\wr@\iota$ will fail because there is no $\mathtt{sel}$ in the context, so only $\mathtt{inter}$ can be given focus; this will consume the $\mathtt{act}$ and release two copies of $(\mathtt{sel}\ \mathtt{at}\ \iota)$ and the continuation into the context. Focusing on the latter will fail now (because $\mathtt{out}\,x\ m$ and $\mathtt{in}\,x\ m$ (for some m) are not yet available), so the only applicable foci are the two sums that can now be "unlocked" using the $\mathtt{sel}$s. The output and input can be unlocked in an irrelevant order, producing two tokens $\mathtt{in}\,x\ a$ and $\mathtt{out}\,x\ a$. Note in particular that the witness a was chosen for the universal quantifier in the encoding of $?x.\,Q$ because the subsequent consumption of these two tokens requires the messages to be identical. (Any other choice will not lead to a successful proof.) After both tokens are consumed, we get the final form $\mathtt{act}@s \cdot r$, where r is the inherent rate of x (found from the $\mathtt{rates}$ component of the unrestricted zone). This sequent is canonical and contains $\wr Q \mid R\,a\wr$.

Our encoding therefore represents every $S\pi$ action in terms of "micro" actions in the following rigid order: one micro action to determine what kind of action (internal or synchronization), one micro action per sum to select the term(s) that will interact, and finally one micro action to establish the contract of the action. Thus we see that focusing is crucial to maintain the semantic interpretation of (neutral) sequents. In an unfocused calculus, several of these steps could have partial overlaps, making such a semantic interpretation inordinately complicated. We do not know of any encoding of the π calculus that can provide such interpretations in unfocused sequents without changing the underlying logic. In CLF [7] the logic is extended with explicit monadic staging, and this enables a form of adequacy [7]; however, the encoding is considerably more complex because processes and sums cannot be fully lifted and must instead be specified in terms of a lifting computation. Adequacy is then obtained via a permutative equivalence over the lifting operation. Other encodings of π calculi in linear logic, such as [19] and [3], concentrate on the easier asynchronous fragment and lack adequacy proofs anyhow.

Theorem 25 (completeness). *If $E \vdash P \xrightarrow{\ r\ } Q$, then the following canonical sequent is derivable.*

$$\llbracket E \rrbracket_{\mathrm{e}}, \mathtt{rates}, \mathtt{inter}@\iota\ ;\ \Uparrow\mathtt{act}@s, \wr P\wr@\iota\ ;\ \cdot \Longrightarrow \cdot\ ;\ (\llbracket Q \rrbracket_{\mathrm{p}} \ \mathtt{at}\ \iota) \otimes \mathtt{act}@s \cdot r.$$

16

Proof. By structural induction of the derivation of $E \vdash P \stackrel{r}{\longrightarrow} Q$. Every interaction rule of $S\pi$ is implementable as an admissible inference rule for canonical sequents. For CONG, we appeal to thm. 21. □

Completeness is a testament to the expressivity of the logic – all executions of $S\pi$ are also expressible in HyLL. However, we also require the opposite (soundness) direction: that every canonical sequent encodes a possible $S\pi$ trace. The proof hinges on the following canonicity lemma.

Lemma 26 (canonical derivations). *In a derivation for a canonical sequent, the derived inference rules for* `inter` *are of one of the two following forms (conclusions and premises canonical).*

$$\overline{[\![E]\!]_{\mathrm{e}}, \mathtt{rates}, \mathtt{inter} @\iota \; ; \; \Uparrow \mathtt{act} @s, \wr P \wr @\iota \; ; \; \cdot \Longrightarrow \cdot \; ; \; ([\![P]\!]_{\mathrm{p}} \text{ at } \iota) \otimes \mathtt{act} @s}$$

$$\frac{[\![E]\!]_{\mathrm{e}}, \mathtt{rates}, \mathtt{inter} @\iota \; ; \; \Uparrow \mathtt{act} @s \cdot r, \wr Q \wr @\iota \; ; \; \cdot \Longrightarrow \cdot \; ; \; ([\![R]\!]_{\mathrm{p}} \text{ at } \iota) \otimes \mathtt{act} @t}{[\![E]\!]_{\mathrm{e}}, \mathtt{rates}, \mathtt{inter} @\iota \; ; \; \Uparrow \mathtt{act} @s, \wr P \wr @\iota \; ; \; \cdot \Longrightarrow \cdot \; ; \; ([\![R]\!]_{\mathrm{p}} \text{ at } \iota) \otimes \mathtt{act} @t}$$

where: either $E \vdash P \stackrel{r}{\longrightarrow} Q$, or $E \vdash P \equiv Q$ with $r = \iota$.

Proof. This is a formal statement of the phenomenon observed earlier in the example (fig. 5): $[\![R]\!]_{\mathrm{p}} \otimes \mathtt{act}$ cannot be focused on the right unless $P \equiv R$, in which case the derivation ends with no more foci on `inter`. If not, the only elements available for focus are `inter` and one of the congruence rules $[\![E]\!]_{\mathrm{e}}$ in the unrestricted context. In the former case, the derived rule consumes the $\Uparrow \mathtt{act} @s$, and by the time `act` is produced again, its world has advanced to $s \cdot r$. In the latter case, the definition of a top level X_n in $\wr P \wr$ is (un)folded (without advancing the world). The proof proceeds by induction on the structure of P. □

Lemma 26 is a strong statement about HyLL derivations using this encoding: every partial derivation using the derived inference rules represents a prefix of an $S\pi$ trace. This is sometimes referred to as *full adequacy*, to distinguish it from adequacy proofs that require complete derivations [28]. The structure of focused derivations is crucial because it allows us to close branches early (using init). It is impossible to perform a similar analysis on unfocused proofs for this encoding; both the encoding and the framework will need further features to implement a form of staging [7, Chapter 3].

Corollary 27 (soundness).
If $[\![E]\!]_{\mathrm{e}}, \mathtt{rates}, \mathtt{inter} @\iota \; ; \; \Uparrow \mathtt{act} @\iota, \wr P \wr @\iota \; ; \; \cdot \Longrightarrow \cdot \; ; \; ([\![Q]\!]_{\mathrm{p}} \text{ at } \iota) \otimes \mathtt{act} @r$ is derivable, then $E \vdash P \stackrel{r}{\longrightarrow}^{*} Q$.

Proof. Directly from lem. 26. □

4.2 Stochastic correctness with respect to simulation

So far the HyLL($\mathcal{P}$) encoding of $S\pi$ represents any $S\pi$ trace symbolically. However, not every symbolic trace of an $S\pi$ process can be produced according to the operational semantics of $S\pi$, which is traditionally given by a simulator. This is the main difference between HyLL (and $S\pi$) and the approach of CSL [2], where the truth of a proposition is evaluated against a CTMC, which is why equivalence in CSL is identical to CTMC bisimulation [13]. In this section we sketch how the execution could be used directly on the canonical sequents to produce only correct traces (proofs). The proposal in this section should be seen by analogy to the execution model of $S\pi$ simulators such as SPiM [30], although we do not use the Gillespie algorithm.

The main problem of simulation is determining which of several competing enabled actions in a canonical sequent to select as the "next" action from the *race condition* of the actions enabled in the sequent. Because of the focusing restriction, these enabled actions are easy to compute. Each element of $\wr P \wr$ is of the form `sel` $\multimap [\![M]\!]_{\mathrm{s}}$, so the enabled actions in that element are given precisely by the topmost occurrences of $\Uparrow$ in $[\![M]\!]_{\mathrm{s}}$. Because none of the sums can have any restricted channels (they have all been removed in the active decomposition of the process earlier), the rates of all the channels will be found in the `rates` component of the canonical sequent.

The effective rate of a channel x is related to its inherent rate by scaling by a factor proportional to the *activity* on the channel, as defined in [30]. Note that this definition is on the *rate constants* of exponential

distributions, not the rates themselves. The distribution of the minimum of a list of random variables with exponential distribution is itself an exponential distribution whose rate constant is the sum of those of the individual variables. Each individual transition on a channel is then weighted by the contribution of its rate to this sum. The choice of the transition to select is just the ordinary logical non-determinism. Note that the rounds of the algorithm do not have an associated *delay* element as in [30]; instead, we compute (symbolically) a distribution over the delays of a sequence of actions.

Because stochastic correctness is not necessary for the main adequacy result in the previous subsection, we leave the details of simulation to future work.

5 Direct encoding of molecular biology

Models of molecular biology have a wealth of examples of transition systems with temporal and stochastic constraints. In a biochemical reaction, molecules can interact to form other molecules or undergo internal changes such as phosphorylation, and these changes usually occur as parts of networks of interacting processes with continuous kinetic feedback. $S\pi$ has been used in a number of such models; since we have an adequate encoding of $S\pi$, we can use these models via the encoding.

However, biological systems can also be encoded directly in HyLL. As an example, consider a simplified *repressilator* gene network consisting of two genes, each causing the production of a protein that represses the other gene by negative feedback. This is a simplification of the three-gene network constructed in [15]. We note that each gene can be in an "on" (activated) or an "off" (deactivated) state, represented by the unary predicates on and off. Molecules of the transcribed proteins are represented with the unary predicate prot. Transitions in the network are encoded as axioms.

Example: the repressilator, using temporal constraints   The system consists of the following components:

- *Repression*: Each protein molecule deactivates the next gene in the cycle after (average) deactivation delay d

$$\texttt{repress}\, a\, b \;\overset{\text{def}}{=}\; \texttt{prot}\, a \otimes \texttt{on}\, b \multimap \rho_d(\texttt{off}\, b \otimes \texttt{prot}\, a).$$

- *Reactivation*: When a gene is in the "off" state, it eventually becomes "on" after an average delay of r:

$$\texttt{react} \;\overset{\text{def}}{=}\; \forall a.\, \texttt{off}\, a \multimap \rho_r\, \texttt{on}\, a.$$

It is precisely this reactivation that causes the system to oscillate instead of being bistable.

- *Synthesis*: When a gene is "on", it transcribes RNA for its protein taking average delay t, after which it continues to be "on" and a molecule of the protein is formed.

$$\texttt{synt} \;\overset{\text{def}}{=}\; \forall a.\, \texttt{on}\, a \multimap \rho_t(\texttt{on}\, a \otimes \texttt{prot}\, a).$$

- *Dissipation*: If a protein does not react with a gene, then it dissipates after average delay s:

$$\texttt{diss} \;\overset{\text{def}}{=}\; \forall a.\, \texttt{prot}\, a \multimap \rho_s\, \mathbf{1}.$$

- *Well defined*: We need to say that a gene cannot be on and off at the same time, that a gene has to be on or off, and that all delays are different:

$$\texttt{well\_def} \;\overset{\text{def}}{=}\; (\forall x.\; \texttt{on}\, x \otimes \texttt{off}\, x \multimap 0) \wedge (\forall x.\; \texttt{on}\, x \vee \texttt{off}\, x) \wedge (d \neq r \neq s \neq t).$$

The system consists of a repression cycle for genes a and b, and the other processes:

$$\texttt{system} \;\overset{\text{def}}{=}\; \texttt{repress}\, a\, b, \texttt{repress}\, b\, a, \texttt{react}, \texttt{synt}, \texttt{diss}, \texttt{well\_def}.$$

Examples of valid sequents are (0 is the initial instant of time):

$$\dagger\, \texttt{system}@0 \;;\; \underbrace{\rho_{r+t}\, \texttt{on}\, \texttt{a}@0, \texttt{off}\, \texttt{b}@0}_{\text{initial state}} \Longrightarrow \underbrace{\rho_{r+t+d}\, \texttt{off}\, \texttt{a} \otimes \top @0}_{\text{final state}}$$

18

From `off b` we get `on b ⊗ prot b` after interval $r + t$; then `prot b` together with `on a` forms `prot b ⊗ off a` after a further delay d.

Note. In general, delays are functions of the elements involved. Handling this is feasable; However it would require to extend HyLL syntax.

Example: stochastic repressilator We now revisit our example but this time using rates. Note that the encodings can be very similar in the temporal and probabilistic fragments of our logic; the only differences being the interpretation of the constraints: Here, d, t, r and s are interpreted as (probabilities defined by) rates.

$$\text{repress } a\ b \overset{\text{def}}{=} \text{prot } a \otimes \text{on } b \multimap \rho_d(\text{off } b \otimes \text{prot } a)$$

$$\text{synt} \overset{\text{def}}{=} \forall a.\, \text{on } a \multimap \rho_t(\text{on } a \otimes \text{prot } a).$$

$$\text{react} \overset{\text{def}}{=} \forall a.\, \text{off } a \multimap \rho_r \text{ on } a.$$

$$\text{diss} \overset{\text{def}}{=} \forall a.\, \text{prot } a \multimap \rho_s\, \mathbf{1}.$$

Suppose we want to show that in the two-gene repressilator, the state $\text{on}(a) \otimes \text{off}(b)$ can oscillate to $\text{off}(a) \otimes \text{on}(b)$. The proof looks as below, with one sub-proofs named P, and most of the worlds and a second sub-proof elided:

$$
\cfrac{
\text{off b} \Longrightarrow \text{off b}
\quad
\cfrac{
\text{on b} \Longrightarrow \text{on b}
\quad
\cfrac{
\cfrac{\text{on a}, \rho_r\, \rho_t\, \text{prot b} \Longrightarrow \rho_r\, \rho_t\, \rho_d\, \text{off a} \quad P \quad \cdots}{\text{on a}, \rho_r\, \rho_t(\text{on b} \otimes \text{prot b}) \Longrightarrow \exists k.\ \rho_k(\text{off a} \otimes \text{on b})}
}{\text{on a}, \rho_r\, \text{on b} \Longrightarrow \exists k.\ \rho_k(\text{off a} \otimes \text{on b})}\ \text{synt}
}{\text{on a}, \text{off b} \Longrightarrow \exists k.\ \rho_k(\text{off a} \otimes \text{on b})}\ \text{react}
$$

$$
P\ =\ \cfrac{
\cfrac{
\text{on a} \Longrightarrow \rho_r\, \rho_t\, \text{on a} \quad \rho_r\, \rho_t\, \text{prot b} \Longrightarrow \rho_r\, \rho_t\, \text{prot b}
}{\text{on a}, \rho_r\, \rho_t\, \text{prot b} \Longrightarrow \rho_r\, \rho_t(\text{on a} \otimes \text{prot b})}\ \otimes I
\quad
\rho_r\, \rho_t\, \rho_d\, \text{off a} \Longrightarrow \rho_r\, \rho_t\, \rho_d\, \text{off a}
}{\text{on a}, \rho_r\, \rho_t\, \text{prot b} \Longrightarrow \rho_r\, \rho_t\, \rho_d\, \text{off a}}\ \text{repress b a}
$$

In this proof we are using the transition rules at many different worlds. This is allowed because the rules are prefixed with † and therefore available at all worlds. Importantly, in the first premise of P we need to show that $\text{on a} \Longrightarrow \rho_r\, \rho_t\, \text{on a}$. This is only possible if the rate of a self-transition on `on a` is $r \cdot t$. Of course, this is not derivable from the rest of the theory (and may not actually be true), so it must be added as a new rule; it is the contract that must be satisfied by the repressilator in order for it to oscillate in the desired fashion.

All existing methods for modelling biology have algebraic foundations and none treats logic as the primary inferential device. In this section, we have sketched a mode of use of HyLL that lets one represent the biological elements directly in the logic. Note, however, that unlike formalisms such as the brane or κ-calculi, we do not propose HyLL as a new idealisation of biology. Instead, as far as systems biology is concerned, our proposal should be seen as a uniform language to encode biological systems; providing genuine means to reason about them is left for future work.

6 Related work

Logically, the HyLL sequent calculus is a variant of labelled deduction, a very broad topic not elaborated on here. The combination of linear logic with labelled deduction isn't new to this work. In the η-logic [14] the constraint domain is intervals of time, and the rules of the logic generate constraint inequalities as a side-effect; however its sole aim is the representation of proof-carrying authentication, and it does not deal with genericity or focusing. The main feature of η not in HyLL is a separate constraint context that gives

new constrained propositions. HyLL is also related to the Hybrid Logical Framework (HLF) [34] which captures linear logic itself as a labelled form of intuitionistic logic. Encoding constrained π calculi directly in HLF would be an interesting exercise: we would combine the encoding of linear logic with the constraints of the process calculus. Because HLF is a very weak logic with a proof theory based on natural deduction, it is not clear whether (and in what forms) an adequacy result in HyLL can be transferred to HLF.

Temporal logics such as CSL and PCTL [21] are popular for logical reasoning on temporal properties of transition systems with probabilities. In such logics, truth is defined in terms of correctness with respect to a constrained forcing relation on the constraint algebra. In CSL and PCTL states are formal entities (names) labeled with atomic propositions. Formulae are interpreted on algebraic structures that are discrete (in PCTL) or continuous (in CSL) time Markov chains. Transitions between states are viewed as couples of states labeled with a probability (the probability of the transition), which is defined as a function from $S \times S$ into $[0, 1]$, where S is the set of states. While such logics have been very successful in practice with efficient tools, the proof theory of these logics is very complex. Indeed, such modal logics generally cannot be formulated in the sequent calculus, and therefore lack cut-elimination and focusing. In contrast, HyLL has a very traditional proof theoretic pedigree, but lacks such a close correspondence between logical and algebraic equivalence. Probably the most well known and relevant stochastic formalism not already discussed is that of stochastic Petri-nets [25], which have a number of sophisticated model checking tools, including the PRISM framework [24]. Recent advances in proof theory suggest that the benefits of model checking can be obtained without sacrificing proofs and proof search [4].

7   Conclusion and future work

We have presented HyLL, a hybrid extension of intuitionistic linear logic with a simple notion of situated truth, a traditional sequent calculus with cut-elimination and focusing, and a modular and instantiable constraint system (set of worlds) that can be directly manipulated using hybrid connectives. We have proposed three instances of HyLL (i.e three particular instances of the set of worlds): one modelling temporal constraints and the others modelling probabilistic or stochastic (continuous time Markov processes) constraints. We have shown how to obtain representationally adequate encodings of constrained transition systems, such as the synchronous stochastic π-calculus in a suitable instance of HyLL. We have also presented some preliminary experiments of direct encoding of biological systems, viewed as transition systems, in HyLL, using either temporal or probabilistic constraints.

Several instantiations of HyLL besides the ones in this paper seem interesting. For example, we can already use disjunction ($\oplus$) to explain disjunctive states, but it is also possible to obtain a more extensional branching by treating the worlds as points in an arbitrary partially-ordered set instead of a monoid. Another possibility is to consider lists of worlds instead of individual worlds – this would allow defining periodic availability of a resource, such as one being produced by an oscillating process. The most interesting domain is that of discrete probabilities: here the underlying semantics is given by discrete time Markov chains instead of CTMCs, which are often better suited for symbolic evaluation [41].

The logic we have provided so far is a logical framework well suited *to represent* constrained transition systems. The design of a logical framework *for* (i.e. to reason about) constrained transition systems is left for future work -and might be envisioned by using a two-levels logical framework such as the Abella system.

An important open question is whether a general logic such as HyLL can serve as a framework for specialized logics such as CSL and PCTL. A related question is what benefit linearity truly provides for such logics – linearity is obviously crucial for encoding process calculi that are inherently stateful, but CSL requires no such notion of single consumption of resources.

In the κ-calculus, reactions in a biological system are modeled as reductions on graphs with certain state annotations. It appears (though this has not been formalized) that the κ-calculus can be embedded in HyLL even more naturally than $S\pi$, because a solution—a multiset of chemical products—is simply a tensor of all the internal states of the binding sites together with the formed bonds. One important innovation of κ is the ability to extract semantically meaningful "stories" from simulations. We believe that HyLL provides a natural formal language for such stories.

We became interested in the problem of encoding stochastic reasoning in a resource aware logic because we were looking for the logical essence of biochemical reactions. What we envision for the domain of "biological computation" is a resource-aware stochastic or probabilistic λ-calculus that has HyLL propositions as (behavioral) types. First step in this direction consists in exploiting and polishing the logic we have provided; This is the focus of our efforts at the CNRS, I3S.

References

[1] Jean-Marc Andreoli. Logic programming with focusing proofs in linear logic. *J. of Logic and Computation*, 2(3):297–347, 1992.

[2] A. Aziz, K. Sanwal, V. Singhal, and R. Brayton. Model checking continuous time Markov chains. *ACM Transactions on Computational Logic*, 1(1):162–170, 2000.

[3] David Baelde. Logique linéaire et algèbre de processus. Technical report, INRIA Futurs, LIX and ENS, 2005.

[4] David Baelde, Andrew Gacek, Dale Miller, Gopalan Nadathur, and Alwen Tiu. The Bedwyr system for model checking over syntactic expressions. In F. Pfenning, editor, *21th Conf. on Automated Deduction (CADE)*, number 4603 in LNAI, pages 391–397, New York, 2007. Springer.

[5] Torben Braüner and Valeria de Paiva. Intuitionistic hybrid logic. *Journal of Applied Logic*, 4:231–255, 2006.

[6] Luca Cardelli. Brane calculi. In *Proceedings of BIO-CONCUR'03*, volume 180. Elsevier ENTCS, 2003.

[7] Iliano Cervesato, Frank Pfenning, David Walker, and Kevin Watkins. A concurrent logical framework II: Examples and applications. Technical Report CMU-CS-02-102, Carnegie Mellon University, 2003. Revised, May 2003.

[8] Nathalie Chabrier-Rivier, François Fages, and Sylvain Soliman. The biochemical abstract machine BIOCHAM. In *International Workshop on Computational Methods in Systems Biology (CMSB-2)*, LNCS. Springer, 2004.

[9] Bor-Yuh Evan Chang, Kaustuv Chaudhuri, and Frank Pfenning. A judgmental analysis of linear logic. Technical Report CMU-CS-03-131R, Carnegie Mellon University, December 2003.

[10] Kaustuv Chaudhuri, Frank Pfenning, and Greg Price. A logical characterization of forward and backward chaining in the inverse method. *J. of Automated Reasoning*, 40(2-3):133–177, March 2008.

[11] Vincent Danos and Jean Krivine. Formal molecular biology done in CCS. In *Proceedings of BIO-CONCUR'03*, volume 180, pages 31–49. Elsevier ENTCS, 2003.

[12] Vincent Danos and Cosimo Laneve. Formal molecular biology. *Theor. Comput. Sci.*, 325(1):69–110, 2004.

[13] Josée Desharmais and Prakash Panangaden. Continuous stochastic logic characterizes bisimulation of continuous-time Markov processes. *Journal of Logic and Algebraic Programming*, 56:99–115, 2003.

[14] Henry DeYoung, Deepak Garg, and Frank Pfenning. An authorization logic with explicit time. In *Computer Security Foundations Symposium (CSF-21)*, pages 133–145. IEEE Computer Society, 2008.

[15] Michael B. Elowitz and Stanislas Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338, 20 January 2000.

[16] E. Allen Emerson. Temporal and modal logic. In *TCS*, pages 995–1072. Elsevier, 1995.

[17] Stewart N. Ethier and Thomas G. Kurtz. *Markov Processes; Characterization and Convergence*. Wiley series in Probability and Mathematical Statistics. Wiley-interscience, 1986.

[18] Dominique Foata and Aimé Fuchs. *Calcul des probabilités, cours exercices et problèmes corrigés*. Dunod, 2e edition, 2003.

[19] Deepak Garg and Frank Pfenning. Type-directed concurrency. In Martín Abadi and Luca de Alfaro, editors, *16th International Conference on Concurrency Theory (CONCUR)*, volume 3653 of *LNCS*, pages 6–20. Springer, 2005.

[20] Jean-Yves Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.

[21] H. Hansson and B. Jonsson. A logic for reasoning about time and probability. *Formal Aspects of Computing*, (6), 1994.

[22] Jane Hillston. *A compositional approach to performance modelling*. Cambridge University Press, 1996.

[23] Johan Anthony Willem Kamp. *Tense Logic and the Theory of Linear Order*. PhD thesis, University of California, Los Angeles, 1968.

[24] M. Kwiatkowska, G. Norman, and D. Parker. Probabilistic symbolic model checking using PRISM: a hybrid approach. *International Journal of Software Tools for Technology Transfer*, 6(2), 2004.

[25] M. Ajmone Marsan, G. Balbo, G. Conte, S. Donatelli, and G. Francenschinis. *Modelling with Generalised Stochastic Petri Nets*. Wiley Series in Parallel Computing. Wiley and Sons, 1995.

[26] Dale Miller. The π-calculus as a theory in linear logic: Preliminary results. In E. Lamma and P. Mello, editors, *3rd Workshop on Extensions to Logic Programming*, number 660 in LNCS, pages 242–265, Bologna, Italy, 1993. Springer.

[27] Robin Milner. *Communicating and Mobile Systems : The π-Calculus*. Cambridge University Press, New York, NY, USA, 1999.

[28] Vivek Nigam and Dale Miller. Focusing in linear meta-logic. In *Proceedings of IJCAR: International Joint Conference on Automated Reasoning*, volume 5195 of *LNAI*, pages 507–522. Springer, 2008.

[29] Frank Pfenning and Conal Elliott. Higher-order abstract syntax. In *Proceedings of the ACM-SIGPLAN Conference on Programming Language Design and Implementation*, pages 199–208. ACM Press, June 1988.

[30] Andrew Phillips and Luca Cardelli. A correct abstract machine for the stochastic pi-calculus. *Concurrent Models in Molecular Biology*, August 2004.

[31] Andrew Phillips and Luca Cardelli. A correct abstract machine for the stochastic pi-calculus. In *Proceedings of BioConcur'04*, ENTCS, 2004.

[32] Andrew Phillips, Luca Cardelli, and Giuseppe Castagna. A graphical representation for biological processes in the stochastic pi-calculus. *Transactions on Computational Systems Biology VII*, pages 123–152, 2006.

[33] Arthur N. Prior. *Time and Modality*. Oxford: Clarendon Press, 1957.

[34] Jason Reed. Hybridizing a logical framework. In *International Workshop on Hybrid Logic (HyLo)*, Seattle, USA, August 2006.

[35] A. Regev, E. M. Panina, W. Silverman, L. Cardelli, and E. Shapiro. Bioambients: an abstraction for biological compartments. *Theoretical Computer Science*, 325(1):141–167, 2004.

[36] A. Regev, W. Silverman, and E. Shapiro. Representation and simulation of biochemical processes using the π-calculus and process algebra. In L. Hunter R. B. Altman, A. K. Dunker and T. E. Klein, editors, *Pacific Symposium on Biocomputing*, volume 6, pages 459–470, Singapore, 2001. World Scientific Press.

[37] L. C. G. Rogers and D. Williams. *Diffusions, Markov Processes and Martingales*, volume 1: Foundations. Cambridge Mathematical Library, 2nd edition, 2000.

[38] Uluç Saranli and Frank Pfenning. Using constrained intuitionistic linear logic for hybrid robotic planning problems. In *IEEE International Conference on Robotics and Automation (ICRA)*, pages 3705–3710. IEEE, 2007.

[39] Alex Simpson. *The Proof Theory and Semantics of Intuitionistic Modal Logic*. PhD thesis, University of Edinburgh, 1994.

[40] Kevin Watkins, Iliano Cervesato, Frank Pfenning, and David Walker. A concurrent logical framework I: Judgments and properties. Technical Report CMU-CS-02-101, Carnegie Mellon University, 2003. Revised, May 2003.

[41] Peng Wu, Catuscia Palamidessi, and Huimin Lin. Symbolic bisimulations for probabilistic systems. In *QEST'07*, pages 179–188. IEEE Computer Society, 2007.

A   Proofs

A.1   Identity principle

Theorem 28 (Identity principle). *The following rule is derivable.*

$$\overline{\Gamma \; ; \; A@w \Longrightarrow A@w} \;\; \text{init*}$$

Proof. By induction on the structure of A. We have the following cases.

case A is an atom $p \, \vec{t}$. Then, $\Gamma \; ; \; p \, \vec{t}@w \Longrightarrow p \, \vec{t}@w$ by init.

case A is $B \,\&\, C$.

$$\cfrac{\cfrac{\overline{\Gamma \; ; \; B@w \Longrightarrow B@w} \;\; \text{i.h.}}{\Gamma \; ; \; B \,\&\, C@w \Longrightarrow B@w} \;\&L_1 \qquad \cfrac{\overline{\Gamma \; ; \; C@w \Longrightarrow C@w} \;\; \text{i.h.}}{\Gamma \; ; \; B \,\&\, C@w \Longrightarrow C@w} \;\&L_2}{\Gamma \; ; \; B \,\&\, C@w \Longrightarrow B \,\&\, C@w} \;\&R$$

case A is $\top$.

$$\overline{\Gamma \; ; \; \top@w \Longrightarrow \top@w} \;\; \top R$$

case A is $B \oplus C$.

$$\cfrac{\cfrac{\overline{\Gamma \; ; \; B@w \Longrightarrow B@w} \;\; \text{i.h.}}{\Gamma \; ; \; B@w \Longrightarrow B \oplus C@w} \;\oplus R_1 \qquad \cfrac{\overline{\Gamma \; ; \; C@w \Longrightarrow C@w} \;\; \text{i.h.}}{\Gamma \; ; \; C@w \Longrightarrow B \oplus C@w} \;\oplus R_2}{\Gamma \; ; \; B \oplus C@w \Longrightarrow B \oplus C@w} \;\oplus L$$

case A is $\mathbf{0}$.

$$\overline{\Gamma \; ; \; \mathbf{0}@w \Longrightarrow \mathbf{0}@w} \;\; \mathbf{0}L$$

case A is $B \multimap C$.

$$\cfrac{\cfrac{\overline{\Gamma \; ; \; B@w \Longrightarrow B@w} \;\; \text{i.h.} \quad \overline{\Gamma \; ; \; C@w \Longrightarrow C@w} \;\; \text{i.h.}}{\Gamma \; ; \; B \multimap C@w, B@w \Longrightarrow C@w} \;\multimap L}{\Gamma \; ; \; B \multimap C@w \Longrightarrow B \multimap C@w} \;\multimap R$$

case A is $B \otimes C$.

$$\cfrac{\cfrac{\overline{\Gamma \; ; \; B@w \Longrightarrow B@w} \;\; \text{i.h.} \quad \overline{\Gamma \; ; \; C@w \Longrightarrow C@w} \;\; \text{i.h.}}{\Gamma \; ; \; B@w, C@w \Longrightarrow B \otimes C@w} \;\otimes R}{\Gamma \; ; \; B \otimes C@w \Longrightarrow B \otimes C@w} \;\otimes L$$

case A is $\mathbf{1}$.

$$\cfrac{\cfrac{}{\Gamma \; ; \; \cdot \Longrightarrow \mathbf{1}@w} \;\; \mathbf{1}R}{\Gamma \; ; \; \mathbf{1}@w \Longrightarrow \mathbf{1}@w} \;\; \mathbf{1}L$$

case A is $\forall x. B$.

$$\cfrac{\cfrac{\overline{\Gamma \; ; \; B@w \Longrightarrow B@w} \;\; \text{i.h.}}{\Gamma \; ; \; \forall \alpha. \, B@w \Longrightarrow B@w} \;\forall L}{\Gamma \; ; \; \forall \alpha. \, B@w \Longrightarrow \forall \alpha. \, B@w} \;\forall R^\alpha$$

case A is $\exists x.\, B$.

$$\dfrac{\dfrac{\overline{\Gamma \; ; \; B@w \Longrightarrow B@w} \;\; \text{i.h.}}{\Gamma \; ; \; B@w \Longrightarrow \exists \alpha.\; B@w} \;\; \exists R}{\Gamma \; ; \; \exists \alpha.\; B@w \Longrightarrow \exists \alpha.\; B@w} \;\; \exists L^\alpha$$

case A is $!\,B$.

$$\dfrac{\dfrac{\dfrac{\overline{\Gamma, B@w \; ; \; B@w \Longrightarrow B@w} \;\; \text{i.h.}}{\Gamma, B@w \; ; \; \cdot \Longrightarrow B@w} \;\; \text{copy}}{\Gamma, B@w \; ; \; \cdot \Longrightarrow !\,B@w} \;\; !R}{\Gamma \; ; \; !\,B@w \Longrightarrow !\,B@w} \;\; !L$$

case A is $\downarrow u.\, B$.

$$\dfrac{\dfrac{\overline{\Gamma \; ; \; [w/u]B@w \Longrightarrow [w/u]B@w} \;\; \text{i.h.}}{\Gamma \; ; \; \downarrow u.\, B@w \Longrightarrow [w/u]B@w} \;\; \downarrow L}{\Gamma \; ; \; \downarrow u.\, B@w \Longrightarrow \downarrow u.\, B@w} \;\; \downarrow R$$

case A is $(B \text{ at } v)$.

$$\dfrac{\dfrac{\overline{\Gamma \; ; \; B@v \Longrightarrow B@v} \;\; \text{i.h.}}{\Gamma \; ; \; (B \text{ at } v)@w \Longrightarrow B@v} \;\; \text{at}L}{\Gamma \; ; \; (B \text{ at } v)@w \Longrightarrow (B \text{ at } v)@w} \;\; \text{at}R$$

$\square$

A.2 Cut admissibility

Theorem 29 (Cut admissibility). *The following two rules are admissible.*

$$\dfrac{\Gamma \; ; \; \Delta \Longrightarrow A@w \quad \Gamma \; ; \; \Delta', A@w \Longrightarrow C@w'}{\Gamma \; ; \; \Delta, \Delta' \Longrightarrow C@w'} \;\; \text{cut}$$

$$\dfrac{\Gamma \; ; \; \cdot \Longrightarrow A@w \quad \Gamma, A@w \; ; \; \Delta \Longrightarrow C@w'}{\Gamma \; ; \; \Delta \Longrightarrow C@w'} \;\; \text{cut!}$$

Proof. Name the two premise derivations $\mathcal{D}$ and $\mathcal{E}$ respectively. The proof proceeds by induction on the structure of the derivations $\mathcal{D}$ and $\mathcal{E}$, and more precisely on a lexicographic order that allows the induction hypothesis to be used whenever:

1. The cut formula becomes strictly smaller (in the subformula relation), or
2. The cut formula remains the same, but an instance of cut is used to justify an instance of cut!.
3. The cut formula remains the same, but the derivation $\mathcal{D}$ is strictly smaller, or
4. The cut formula remains the same, but the derivation $\mathcal{E}$ is strictly smaller, or

In each case, we consider derivations to be identical that differ in such a way that one can be derived from the other simply by weakening and contracting the unrestricted contexts of their respective sequents. The lexicographic order is well-founded because the given derivations $\mathcal{D}$ and $\mathcal{E}$ are finite, and cut! is used at most once per subformula of A (see "copy cuts" below). All the cuts break down into the following four major categories.

Atomic cuts   where the formula A is an atom $p\,(\vec{t})$. We have the following two cases;

Case. $\mathcal{D}$ is:

$$\frac{}{\Gamma \; ; \; p\,(\vec{t})@w \Longrightarrow p\,(\vec{t})@w} \; \text{init}$$

Then the result of the cut has the same conclusion as that of $\mathcal{E}$.

Case. $\mathcal{E}$ is

$$\frac{}{\Gamma \; ; \; p\,(\vec{t})@w \Longrightarrow p\,(\vec{t})@w} \; \text{init}$$

Then the result of the cut has the same conclusion as that of $\mathcal{D}$.

Principal cuts   where a non-atomic cut formula A is introduced by a final right rule in $\mathcal{D}$ and a final left-rule in $\mathcal{E}$. We have the following cases.

Case. A is $A_1 \mathbin{\&} A_2$, and:

$$\mathcal{D} = \frac{\mathcal{D}_1 :: \Gamma \; ; \; \Delta \Longrightarrow A_1@w \quad \mathcal{D}_2 :: \Gamma \; ; \; \Delta \Longrightarrow A_2@w}{\Gamma \; ; \; \Delta \Longrightarrow A_1 \mathbin{\&} A_2@w} \; \&R \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma \; ; \; \Delta', A_i@w \Longrightarrow C@w'}{\Gamma \; ; \; \Delta', A_1 \mathbin{\&} A_2@w \Longrightarrow C@w'} \; \&L_i$$

Then:

$$\Gamma \; ; \; \Delta, \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}_i \text{ and } \mathcal{E}'.$$

Case. A is $A_1 \oplus A_2$, and:

$$\mathcal{D} = \frac{\mathcal{D}' :: \Gamma \; ; \; \Delta \Longrightarrow A_i@w}{\Gamma \; ; \; \Delta \Longrightarrow A_1 \oplus A_2@w} \; \oplus R_i \qquad \mathcal{E} = \frac{\begin{array}{c} \mathcal{E}_1 :: \Gamma \; ; \; \Delta', A_1@w \Longrightarrow C@w' \\ \mathcal{E}_2 :: \Gamma \; ; \; \Delta', A_2@w \Longrightarrow C@w' \end{array}}{\Gamma \; ; \; \Delta', A_1 \oplus A_2@w \Longrightarrow C@w'} \; \oplus L$$

Then:

$$\Gamma \; ; \; \Delta, \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}' \text{ and } \mathcal{E}_i.$$

Case. A is $A_1 \multimap A_2$, and:

$$\mathcal{D} = \frac{\mathcal{D}' :: \Gamma \; ; \; \Delta, A_1@w \Longrightarrow A_2@w}{\Gamma \; ; \; \Delta \Longrightarrow A_1 \multimap A_2@w} \; \multimap R \qquad \mathcal{E} = \frac{\mathcal{E}_1 :: \Gamma \; ; \; \Delta_1' \Longrightarrow A_1@w \quad \mathcal{E}_2 :: \Gamma \; ; \; \Delta_2', A_2@w \Longrightarrow C@w'}{\Gamma \; ; \; \Delta_1', \Delta_2', A_1 \multimap A_2 \Longrightarrow C@w'} \; \multimap L$$

Then:

$$\Gamma \; ; \; \Delta, A_1@w, \Delta_2' \Longrightarrow C@w' \qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}' \text{ and } \mathcal{E}_2.$$
$$\Gamma \; ; \; \Delta, \Delta_1', \Delta_2' \Longrightarrow C@w' \qquad\qquad\qquad\qquad \text{cut on } \mathcal{E}_1 \text{ and above.}$$

Case. A is $A_1 \otimes A_2$, and:

$$\mathcal{D} = \frac{\mathcal{D}_1 :: \Gamma \; ; \; \Delta_1 \Longrightarrow A_1@w \quad \mathcal{D}_2 :: \Gamma \; ; \; \Delta_2 \Longrightarrow A_2@w}{\Gamma \; ; \; \Delta_1, \Delta_2 \Longrightarrow A_1 \otimes A_2@w} \; \otimes R \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma \; ; \; \Delta', A_1@w, A_2@w \Longrightarrow C@w'}{\Gamma \; ; \; \Delta', A_1 \otimes A_2@w \Longrightarrow C@w'} \; \otimes L$$

Then:

$$\Gamma \; ; \; \Delta', \Delta_2, A_1@w \Longrightarrow C@w' \qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}_2 \text{ and } \mathcal{E}'.$$
$$\Gamma \; ; \; \Delta', \Delta_1, \Delta_2 \Longrightarrow C@w' \qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}_1 \text{ and above.}$$

Case. A is $\mathbf{1}$, and:

$$\mathcal{D} = \frac{}{\Gamma \; ; \; \cdot \Longrightarrow \mathbf{1}@w} \; \mathbf{1}R \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma \; ; \; \Delta' \Longrightarrow C@w'}{\Gamma \; ; \; \Delta', \mathbf{1}@w \Longrightarrow C@w'} \; \mathbf{1}L$$

The result of the cut is the conclusion of $\mathcal{E}'$.

Case. A is $\forall x.\, B$, and:

$$\mathcal{D} = \frac{\mathcal{D}'(\alpha) :: \Gamma \;;\; \Delta \Longrightarrow B@w}{\Gamma \;;\; \Delta \Longrightarrow \forall \alpha.\, B@w} \; \forall R^{\alpha} \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma \;;\; \Delta', [\tau/\alpha]B@w \Longrightarrow C@w'}{\Gamma \;;\; \Delta', \forall \alpha.\, B@w \Longrightarrow C@w'} \; \forall L$$

Let a be any parameter. Then:

$$\Gamma \;;\; \Delta, \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}'(\tau) \text{ and } \mathcal{E}'.$$

Case. A is $\exists x.\, B$, and:

$$\mathcal{D} = \frac{\mathcal{D}' :: \Gamma \;;\; \Delta \Longrightarrow [\tau/\alpha]B@w}{\Gamma \;;\; \Delta \Longrightarrow \exists \alpha.\, B@w} \; \exists R \qquad \mathcal{E} = \frac{\mathcal{E}'(\alpha) :: \Gamma \;;\; \Delta', B@w \Longrightarrow C@w'}{\Gamma \;;\; \Delta', \exists \alpha.\, B@w \Longrightarrow C@w'} \; \exists L^{\alpha}$$

Let a be any parameter. Then:

$$\Gamma \;;\; \Delta, \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}' \text{ and } \mathcal{E}'(\alpha).$$

Case. A is $!\,B$, and:

$$\mathcal{D} = \frac{\mathcal{D}' :: \Gamma \;;\; \cdot \Longrightarrow B@w}{\Gamma \;;\; \cdot \Longrightarrow !\,B@w} \; !R \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma, B@w \;;\; \Delta' \Longrightarrow C@w'}{\Gamma \;;\; \Delta', !\,B@w \Longrightarrow C@w'} \; !L$$

Then:

$$\Gamma \;;\; \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad\qquad \text{cut! on } \mathcal{D}' \text{ and } \mathcal{E}'.$$

Case. A is $\downarrow u.\, B$, and:

$$\mathcal{D} = \frac{\mathcal{D}' :: \Gamma \;;\; \Delta \Longrightarrow [w/u]B@w}{\Gamma \;;\; \Delta \Longrightarrow \downarrow u.\, B@w} \; \downarrow R \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma \;;\; \Delta', [w/u]B@w \Longrightarrow C@w'}{\Gamma \;;\; \Delta', \downarrow u.\, B@w \Longrightarrow C@w'} \; \downarrow L$$

Then:

$$\Gamma \;;\; \Delta, \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}' \text{ and } \mathcal{E}'.$$

Case. A is $(B \,\texttt{at}\, v)$, and:

$$\mathcal{D} = \frac{\mathcal{D}' :: \Gamma \;;\; \Delta \Longrightarrow B@v}{\Gamma \;;\; \Delta \Longrightarrow (B \,\texttt{at}\, v)@w} \; \texttt{at}R \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma \;;\; \Delta', B@v \Longrightarrow C@w'}{\Gamma \;;\; \Delta', (B \,\texttt{at}\, v)@w \Longrightarrow C@w'} \; \texttt{at}L$$

Then:

$$\Gamma \;;\; \Delta, \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad\qquad \text{cut on } \mathcal{D}' \text{ and } \mathcal{E}'.$$

Copy cuts where the cut formula in $\mathcal{E}$ was transferred using copy, i.e.:

$$\mathcal{D} :: \Gamma \;;\; \cdot \Longrightarrow A@w \qquad \mathcal{E} = \frac{\mathcal{E}' :: \Gamma, A@w \;;\; \Delta', A@w \Longrightarrow C@w'}{\Gamma, A@w \;;\; \Delta' \Longrightarrow C@w'} \; \text{copy}$$

Here,

$$\Gamma, A@w \;;\; \cdot \Longrightarrow A@w \qquad\qquad\qquad\qquad\qquad\qquad \text{weakening on } \mathcal{D}.$$
$$\Gamma, A@w \;;\; \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad\qquad \text{cut on } \mathcal{D} \text{ and } \mathcal{E}'.$$
$$\Gamma \;;\; \Delta' \Longrightarrow C@w' \qquad\qquad\qquad\qquad\qquad\qquad\qquad \text{cut! on } \mathcal{D} \text{ and above.}$$

The first cut is applied on a variant of $\mathcal{D}$ that differs from $\mathcal{D}$ only in terms of a weaker unrestricted context. In the last step, a cut was used to justify a cut!, which is allowed by the lexicographic order.

Left-commutative cuts   where the cut formula A is a side formula in the derivation $\mathcal{D}$. The following is a representative case.

$$\mathcal{D} = \frac{\mathcal{D}' :: \Gamma \; ; \; \Delta, D@w'', E@w'' \Longrightarrow A@w}{\Gamma \; ; \; \Delta, D \otimes E@w'' \Longrightarrow A@w} \; \otimes L \qquad \mathcal{E} :: \Gamma \; ; \; \Delta', A@w \Longrightarrow C@w'.$$

Here,

$$\Gamma \; ; \; \Delta, D@w'', E@w'', \Delta' \Longrightarrow C@w' \qquad\qquad\qquad \text{cut on } \mathcal{D}' \text{ and } \mathcal{E}.$$
$$\Gamma \; ; \; \Delta, \Delta', D \otimes E@w'' \Longrightarrow C@w' \qquad\qquad\qquad\qquad \otimes L.$$

Right-commutative cuts   where the cut formula A is a side formula in the derivation $\mathcal{E}$. The following is a representative case.

$$\mathcal{D} :: \Gamma \; ; \; \Delta \Longrightarrow A@w \qquad \mathcal{E} = \frac{\mathcal{E}_1 :: \Gamma \; ; \; \Delta', A@w \Longrightarrow D@w' \quad \mathcal{E}_2 :: \Gamma \; ; \; \Delta', A@w \Longrightarrow E@w'}{\Gamma \; ; \; \Delta', A@w \Longrightarrow D \,\&\, E@w'} \; \&R$$

Here,

$$\Gamma \; ; \; \Delta, \Delta' \Longrightarrow D@w' \qquad\qquad\qquad\qquad \text{cut on } \mathcal{D} \text{ and } \mathcal{E}_1.$$
$$\Gamma \; ; \; \Delta, \Delta' \Longrightarrow E@w' \qquad\qquad\qquad\qquad \text{cut on } \mathcal{D} \text{ and } \mathcal{E}_2.$$
$$\Gamma \; ; \; \Delta, \Delta' \Longrightarrow D \,\&\, E@w' \qquad\qquad\qquad\qquad\qquad \&R.$$

This completes the inventory of all possible cuts. $\square$

A.3   Invertibility

Theorem 30 (Invertibility). *The following rules are invertible:*
1. *On the right: $\&R$, $\top R$, $\multimap R$, $\forall R$, $\downarrow R$ and $@R$;*
2. *On the left: $\otimes L$, $\mathbf{1}L$, $\oplus L$, $\mathbf{0}L$, $\exists L$, $!L$, $\downarrow L$ and $\mathtt{at}L$.*

Proof. Each inversion is shown to be admissible using a suitable cut.
Case of $\&R$:

$$\frac{\Gamma \; ; \; \Delta \Longrightarrow A_1 \,\&\, A_2@w \quad \dfrac{\overline{\Gamma \; ; \; A_i@w \Longrightarrow A_i@w} \;\; \text{init}^*}{\Gamma \; ; \; A_1 \,\&\, A_2@w \Longrightarrow A_i@w} \; \&L_i}{\Gamma \; ; \; \Delta \Longrightarrow A_i@w} \; \text{cut}$$

Case of $\top R$: trivial.
Case of $\multimap R$:

$$\frac{\Gamma \; ; \; \Delta \Longrightarrow A \multimap B@w \quad \dfrac{\overline{\Gamma \; ; \; A@w \Longrightarrow A@w} \;\; \text{init}^* \quad \overline{\Gamma \; ; \; B@w \Longrightarrow B@w} \;\; \text{init}^*}{\Gamma \; ; \; A \multimap B@w, A@w \Longrightarrow B@w} \; \multimap L}{\Gamma \; ; \; \Delta, A@w \Longrightarrow B@w} \; \text{cut}$$

Case of $\forall R$:

$$\frac{\Gamma \; ; \; \Delta \Longrightarrow \forall \alpha.\; A@w \quad \dfrac{\overline{\Gamma \; ; \; A@w \Longrightarrow A@w} \;\; \text{init}^*}{\Gamma \; ; \; \forall \alpha.\; A@w \Longrightarrow A@w} \; \forall L}{\Gamma \; ; \; \Delta \Longrightarrow A@w} \; \text{cut}$$

27

Case of ↓R:

$$
\cfrac{\Gamma \; ; \Delta \Longrightarrow \downarrow u.\, A @ w \quad \cfrac{\cfrac{}{\Gamma \; ; [w/u]A @ w \Longrightarrow [w/u]A @ w} \; \text{init}^*}{\Gamma \; ; \downarrow u.\, A @ w \Longrightarrow [w/u]A @ w} \; \downarrow L}{\Gamma \; ; \Delta \Longrightarrow [w/u]A @ w} \; \text{cut}
$$

Case of atR:

$$
\cfrac{\Gamma \; ; \Delta \Longrightarrow (A \,\text{at}\, v) @ w \quad \cfrac{\cfrac{}{\Gamma \; ; A @ v \Longrightarrow A @ v} \; \text{init}^*}{\Gamma \; ; (A \,\text{at}\, v) @ w \Longrightarrow A @ v} \; \text{at}L}{\Gamma \; ; \Delta \Longrightarrow A @ v} \; \text{cut}
$$

Case of ⊗L:

$$
\cfrac{\cfrac{\cfrac{}{\Gamma \; ; A @ w \Longrightarrow A @ w} \; \text{init}^* \quad \cfrac{}{\Gamma \; ; B @ w \Longrightarrow B @ w} \; \text{init}^*}{\Gamma \; ; A @ w, B @ w \Longrightarrow A \otimes B @ w} \; \otimes R \quad \Gamma \; ; \Delta, A \otimes B @ w \Longrightarrow C @ w'}{\Gamma \; ; \Delta, A @ w, B @ w \Longrightarrow C @ w'} \; \text{cut}
$$

Case of 1L:

$$
\cfrac{\cfrac{}{\Gamma \; ; \cdot \Longrightarrow \mathbf{1} @ w} \; \mathbf{1}R \quad \Gamma \; ; \Delta, \mathbf{1} @ w \Longrightarrow C @ w'}{\Gamma \; ; \Delta \Longrightarrow C @ w'} \; \text{cut}
$$

Case of ⊕L:

$$
\cfrac{\cfrac{\cfrac{}{\Gamma \; ; A_i @ w \Longrightarrow A_i @ w} \; \text{init}^*}{\Gamma \; ; A_i @ w \Longrightarrow A_1 \oplus A_2 @ w} \; \oplus R_i \quad \Gamma \; ; \Delta, A_1 \oplus A_2 @ w \Longrightarrow C @ w'}{\Gamma \; ; \Delta, A_i @ w \Longrightarrow C @ w'} \; \text{cut}
$$

Case of 0L: trivial.
Case of ∃L:

$$
\cfrac{\cfrac{\cfrac{}{\Gamma \; ; A @ w \Longrightarrow A @ w} \; \text{init}^*}{\Gamma \; ; A @ w \Longrightarrow \exists \alpha.\, A @ w} \; \exists R \quad \Gamma \; ; \Delta, \exists x.\, A @ w \Longrightarrow C @ w'}{\Gamma \; ; \Delta, A @ w \Longrightarrow C @ w'} \; \text{cut}
$$

Case of !L:

$$
\cfrac{\cfrac{\cfrac{\cfrac{}{\Gamma, A @ w \; ; A @ w \Longrightarrow A @ w} \; \text{init}^*}{\Gamma, A @ w \; ; \cdot \Longrightarrow A @ w} \; \text{copy}}{\Gamma, A @ w \; ; \cdot \Longrightarrow \,! A @ w} \; !R \quad \cfrac{\Gamma \; ; \Delta, ! A @ w \Longrightarrow C @ w'}{\Gamma, A @ w \; ; \Delta, ! A @ w \Longrightarrow C @ w'} \; \text{weaken}}{\Gamma, A @ w \; ; \Delta \Longrightarrow C @ w'} \; \text{cut}
$$

Case of ↓L:

$$
\cfrac{\cfrac{\cfrac{}{\Gamma \; ; [w/u]A @ w \Longrightarrow [w/u]A @ w} \; \text{init}^*}{\Gamma \; ; [w/u]A @ w \Longrightarrow \downarrow u.\, A @ w} \; \downarrow R \quad \Gamma \; ; \Delta, \downarrow u.\, A @ w \Longrightarrow C @ w'}{\Gamma \; ; \Delta, [w/u]A @ w \Longrightarrow C @ w'} \; \text{cut}
$$

Case of `atL`:

$$\frac{\dfrac{\overline{\Gamma \; ; \; A@v \Longrightarrow A@v} \; \text{init}^*}{\Gamma \; ; \; A@v \Longrightarrow (A \text{ at } v)@w} \; \text{atR} \qquad \Gamma \; ; \; \Delta, (A \text{ at } v)@w \Longrightarrow C@w'}{\Gamma \; ; \; \Delta, A@v \Longrightarrow C@w'} \; \text{cut}$$

□

A.4 Correctness and consistency

Theorem 31 (Correctness of the sequent calculus)**.**
   1. If $\Gamma \; ; \; \Delta \Longrightarrow C@w$, then $\Gamma \; ; \; \Delta \vdash C@w$. (soundness)
   2. If $\Gamma \; ; \; \Delta \vdash C@w$, then $\Gamma \; ; \; \Delta \Longrightarrow C@w$. (completeness)

Proof. The right rules of the sequent calculus and the introduction rules of natural deduction coincide. Therefore, for (1), we need only to show that the judgemental and left rules of the sequent calculus are admissible in natural deduction, and for (2), only to show that the judgemental and elimination rules of natural deduction are admissible in the sequent calculus. The following are the main cases.
$\Longrightarrow/\vdash$ case. (init)

$$\overline{\Gamma \; ; \; p \; (\vec{t})@w \vdash p \; (\vec{t})@w} \; \text{hyp}$$

$\Longrightarrow/\vdash$ case. (copy)

$$\frac{\overline{\Gamma, A@w \; ; \; \cdot \vdash A@w} \; \text{hyp!} \qquad \Gamma, A@w \; ; \; \Delta, A@w \vdash C@w'}{\Gamma, A@w \; ; \; \Delta \vdash C@w'} \; \text{subst}$$

$\Longrightarrow/\vdash$ case. ($\&L_i$)

$$\frac{\dfrac{\overline{\Gamma \; ; \; A_1 \& A_2@w \vdash A_1 \& A_2@w} \; \text{hyp}}{\Gamma \; ; \; A_1 \& A_2@w \vdash A_i@w} \; \&E_i \qquad \Gamma \; ; \; \Delta, A_i@w \vdash C@w'}{\Gamma \; ; \; \Delta, A_1 \& A_2@w \vdash C@w'} \; \text{subst}$$

$\Longrightarrow/\vdash$ case. ($\oplus L$)

$$\frac{\overline{\Gamma \; ; \; A_1 \oplus A_2@w \vdash A_1 \oplus A_2@w} \; \text{hyp} \quad \Gamma \; ; \; \Delta, A_1@w \vdash C@w' \quad \Gamma \; ; \; \Delta, A_2@w \vdash C@w'}{\Gamma \; ; \; \Delta, A_1 \oplus A_2@w \vdash C@w'} \; \oplus E$$

$\Longrightarrow/\vdash$ case. ($\mathbf{0}L$)

$$\frac{\overline{\Gamma \; ; \; \mathbf{0}@w \vdash \mathbf{0}@w} \; \text{hyp}}{\Gamma \; ; \; \Delta, \mathbf{0}@w \vdash C@w'} \; \mathbf{0}E$$

$\Longrightarrow/\vdash$ case. ($\otimes L$)

$$\frac{\overline{\Gamma \; ; \; A \otimes B@w \vdash A \otimes B@w} \; \text{hyp} \quad \Gamma \; ; \; \Delta, A@w, B@w \vdash C@w'}{\Gamma \; ; \; \Delta, A \otimes B@w \vdash C@w'} \; \otimes E$$

29

$\Longrightarrow/\vdash$ case. $(\mathbf{1}L)$

$$\cfrac{\cfrac{}{\Gamma \; ; \; \mathbf{1}@w \vdash \mathbf{1}@w} \text{ hyp} \qquad \Gamma \; ; \; \Delta \vdash C@w'}{\Gamma \; ; \; \Delta, \mathbf{1}@w \vdash C@w'} \; \mathbf{1}E$$

$\Longrightarrow/\vdash$ case. $(\multimap L)$

$$\cfrac{\cfrac{\cfrac{}{\Gamma \; ; \; A \multimap B@w \vdash A \multimap B@w} \text{ hyp} \qquad \Gamma \; ; \; \Delta \vdash A@w}{\Gamma \; ; \; A \multimap B@w \vdash B@w} \; \multimap E \qquad \Gamma \; ; \; \Delta', B@w \vdash C@w'}{\Gamma \; ; \; \Delta, \Delta', A \multimap B@w \vdash C@w'} \; \text{subst}$$

$\Longrightarrow/\vdash$ case. $(\forall L)$

$$\cfrac{\cfrac{\cfrac{}{\Gamma \; ; \; \forall\alpha. \; A@w \vdash \forall\alpha. \; A@w} \text{ hyp}}{\Gamma \; ; \; \forall\alpha. \; A@w \vdash [\tau/\alpha]A@w} \; \forall E \qquad \Gamma \; ; \; \Delta, [\tau/\alpha]A@w \vdash C@w'}{\Gamma \; ; \; \Delta, \forall\alpha. \; A@w \vdash C@w'} \; \text{subst}$$

$\Longrightarrow/\vdash$ case. $(\exists L)$

$$\cfrac{\cfrac{}{\Gamma \; ; \; \exists\alpha. \; A@w \vdash \exists\alpha. \; A@w} \text{ hyp} \qquad \Gamma \; ; \; \Delta, A@w \vdash C@w'}{\Gamma \; ; \; \Delta, \exists\alpha. \; A@w \vdash C@w'} \; \exists E^\alpha$$

$\Longrightarrow/\vdash$ case. $(!L)$

$$\cfrac{\cfrac{}{\Gamma \; ; \; !\,A@w \vdash !\,A@w} \text{ hyp} \qquad \Gamma, !\,A@w \; ; \; \Delta \vdash C@w'}{\Gamma \; ; \; \Delta, !\,A@w \vdash C@w'} \; !E$$

$\Longrightarrow/\vdash$ case. $(\downarrow L)$

$$\cfrac{\cfrac{\cfrac{}{\Gamma \; ; \; \downarrow u. \, A@w \vdash \downarrow u. \, A@w} \text{ hyp}}{\Gamma \; ; \; \downarrow u. \, A@w \vdash [w/u]A@w} \; \downarrow E \qquad \Gamma \; ; \; \Delta, [w/u]A@w \vdash C@w'}{\Gamma \; ; \; \Delta, \downarrow u. \, A@w \vdash C@w'} \; \text{subst}$$

$\Longrightarrow/\vdash$ case. $(\texttt{at } L)$

$$\cfrac{\cfrac{\cfrac{}{\Gamma \; ; \; (A \,\texttt{at}\, v)@w \vdash (A \,\texttt{at}\, v)@w} \text{ hyp}}{\Gamma \; ; \; (A \,\texttt{at}\, v)@w \vdash A@v} \; \texttt{at } E \qquad \Gamma \; ; \; \Delta, A@v \vdash C@w'}{\Gamma \; ; \; \Delta, (A \,\texttt{at}\, v)@w \vdash C@w'} \; \text{subst}$$

$\vdash/\Longrightarrow$ case. (hyp)

$$\cfrac{}{\Gamma \; ; \; A@w \Longrightarrow A@w} \; \text{init}^*$$

$\vdash/\Longrightarrow$ case. (hyp!)

$$\cfrac{\cfrac{}{\Gamma, A@w \; ; \; A@w \Longrightarrow A@w} \; \text{init}^*}{\Gamma, A@w \; ; \; \cdot \Longrightarrow A@w} \; \text{copy}$$

$\vdash/\Longrightarrow$ case. $(\&E_i)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow A_1\ \&\ A_2\,@\,w \quad \dfrac{\overline{\Gamma\ ;\ A_i\,@\,w \Longrightarrow A_i\,@\,w}\ \text{init}^*}{\Gamma\ ;\ A_1\ \&\ A_2\,@\,w \Longrightarrow A_i\,@\,w}\ \&L_i}{\Gamma\ ;\ \Delta \Longrightarrow A_i\,@\,w}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(\oplus E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow A\oplus B\,@\,w \quad \dfrac{\Gamma\ ;\ \Delta',A\,@\,w \Longrightarrow C\,@\,w' \quad \Gamma\ ;\ \Delta',B\,@\,w \Longrightarrow C\,@\,w'}{\Gamma\ ;\ \Delta',A\oplus B\,@\,w \Longrightarrow C\,@\,w'}\ \oplus L}{\Gamma\ ;\ \Delta,\Delta' \Longrightarrow C\,@\,w'}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(\mathbf{0}E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow \mathbf{0}\,@\,w \quad \dfrac{}{\Gamma\ ;\ \Delta',\mathbf{0}\,@\,w \Longrightarrow C\,@\,w'}\ \mathbf{0}L}{\Gamma\ ;\ \Delta,\Delta' \Longrightarrow C\,@\,w'}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(\otimes E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow A\otimes B\,@\,w \quad \dfrac{\Gamma\ ;\ \Delta',A\,@\,w,B\,@\,w \Longrightarrow C\,@\,w'}{\Gamma\ ;\ \Delta',A\otimes B\,@\,w \Longrightarrow C\,@\,w'}\ \otimes L}{\Gamma\ ;\ \Delta,\Delta' \Longrightarrow C\,@\,w'}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(\mathbf{1}E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow \mathbf{1}\,@\,w \quad \dfrac{\Gamma\ ;\ \Delta' \Longrightarrow C\,@\,w'}{\Gamma\ ;\ \Delta',\mathbf{1}\,@\,w \Longrightarrow C\,@\,w'}\ \mathbf{1}L}{\Gamma\ ;\ \Delta,\Delta' \Longrightarrow C\,@\,w'}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(\forall E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow \forall\alpha.\ A\,@\,w \quad \dfrac{\overline{\Gamma\ ;\ [\tau/\alpha]A\,@\,w \Longrightarrow [\tau/\alpha]A\,@\,w}\ \text{init}^*}{\Gamma\ ;\ \forall\alpha.\ A\,@\,w \Longrightarrow [\tau/\alpha]A\,@\,w}\ \forall L}{\Gamma\ ;\ \Delta \Longrightarrow [\tau/\alpha]A\,@\,w}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(\exists E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow \exists\alpha.\ A\,@\,w \quad \dfrac{\Gamma\ ;\ \Delta',A\,@\,w \Longrightarrow C\,@\,w'}{\Gamma\ ;\ \Delta',\exists\alpha.\ A\,@\,w \Longrightarrow C\,@\,w'}\ \exists L^\alpha}{\Gamma\ ;\ \Delta,\Delta' \Longrightarrow C\,@\,w'}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(!E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow {!}\,A\,@\,w \quad \dfrac{\Gamma,A\,@\,w\ ;\ \Delta' \Longrightarrow C\,@\,w'}{\Gamma\ ;\ \Delta',{!}\,A\,@\,w \Longrightarrow C\,@\,w'}\ !L}{\Gamma\ ;\ \Delta,\Delta' \Longrightarrow C\,@\,w'}\ \text{cut}$$

$\vdash/\Longrightarrow$ case. $(\downarrow E)$

$$\dfrac{\Gamma\ ;\ \Delta \Longrightarrow \downarrow u.\ A\,@\,w \quad \dfrac{\overline{\Gamma\ ;\ \Delta',[w/u]A\,@\,w \Longrightarrow [w/u]A\,@\,w}\ \text{hyp}}{\Gamma\ ;\ \Delta',\downarrow u.\ A\,@\,w \Longrightarrow [w/u]A\,@\,w}\ \downarrow L}{\Gamma\ ;\ \Delta,\Delta' \Longrightarrow [w/u]A\,@\,w}\ \text{cut}$$

31

$\vdash/\Longrightarrow$ case. (at E)

$$\cfrac{\Gamma \ ; \ \Delta \Longrightarrow (A \text{ at } v)@w \qquad \cfrac{\cfrac{}{\Gamma \ ; \ A@v \Longrightarrow A@v} \text{ init*}}{\Gamma \ ; \ (A \text{ at } v)@w \Longrightarrow A@v} \text{ at } L}{\Gamma \ ; \ \Delta \Longrightarrow A@v} \text{ cut}$$

$\square$

Corollary 32 (Consistency of HyLL). *There is no proof of $\cdot \ ; \ \cdot \vdash \mathbf{0}@w$.*

Proof. Suppose $\cdot \ ; \ \cdot \vdash \mathbf{0}@w$ is derivable. Then, by the completeness and cut-admissibility theorems on the sequent calculus, $\cdot \ ; \ \cdot \Longrightarrow \mathbf{0}@w$ must have a cut-free proof. But, we can see by simple inspection that there can be no cut-free proof of $\cdot \ ; \ \cdot \Longrightarrow \mathbf{0}@w$, as this sequent cannot be the conclusion of any rule of inference in the sequent calculus. Therefore, $\cdot \ ; \ \cdot \vdash \mathbf{0}@w$ is not derivable. $\square$

A.5 Connection to IS5

Theorem 33 (HyLL is intuitionistic S5). *The following sequent is derivable: $\cdot \ ; \ \Diamond A@w \Longrightarrow \Box \Diamond A@w$.*

Proof.

$$\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{\cfrac{}{\cdot \ ; \ A@a \Longrightarrow A@a} \text{ init*}}{\cdot \ ; \ A@a \Longrightarrow (A \text{ at } a) \text{ at } b} \text{ at}R}{\cdot \ ; \ A@a \Longrightarrow \exists v. \, (A \text{ at } v)@b} \exists R}{\cdot \ ; \ (A \text{ at } a)@w \Longrightarrow (\exists v. \, (A \text{ at } v) \text{ at } b)@w} \text{ at}L, \text{ at}R}{\cdot \ ; \ (A \text{ at } a)@w \Longrightarrow \forall u. \, (\exists v. \, (A \text{ at } v) \text{ at } u)@w} \forall R^b}{\cdot \ ; \ \exists u. \, (A \text{ at } u)@w \Longrightarrow \forall u. \, (\exists v. \, (A \text{ at } v) \text{ at } u)@w} \exists L^a}{\cdot \ ; \ \Diamond A@w \Longrightarrow \Box \Diamond A@w} \text{ defn}$$

$\square$