Under consideration for publication in Formal Aspects of Computing

A Compositional Modelling and Verification Framework for Stochastic Hybrid Systems

Shuling Wang and Naijun Zhan and Lijun Zhang

State Key Laboratory of Computer Science, Institute of Software, Chinese Academy of Sciences, China

Abstract. In this paper, we propose a general compositional approach for modelling and verification of stochastic hybrid systems (SHSs). We extend Hybrid CSP (HCSP), a very expressive process algebra-like formal modeling language for hybrid systems, by introducing probability and stochasticity to model SHSs, which we call stochastic HCSP (SHCSP). Especially, non-deterministic choice is replaced by probabilistic choice, ordinary differential equations are replaced by stochastic differential equations (SDEs), and communication interrupts are generalized by communication interrupts with weights. We extend Hybrid Hoare Logic to specify and reason about SHCSP processes: On the one hand, we introduce the probabilistic formulas for describing probabilistic states, and on the other hand, we propose the notions of local stochastic differential invariants for characterizing SDEs and global loop invariants for repetition. Throughout the paper, we demonstrate our approach by an aircraft running example.

Keywords: Stochastic hybrid systems, Stochastic hybrid CSP, Deductive verification, Invariants

1. Introduction

Probabilistic and stochastic behavior are omnipresent in computer controlled systems, such as safety-critical hybrid systems, because of uncertain environments, or simplifications to overcome complexity. For example, the movement of aircrafts could be influenced by wind; in networked control systems, message loss and other random effects (e.g., node placement, node failure, battery drain, measurement imprecision) may happen.

Stochastic hybrid systems are the systems in which discrete, continuous and stochastic dynamics tightly intertwine. For safety-critical stochastic hybrid systems, validation and verification can enhance the quality of them and, in particular, to fulfill the quality criteria mandated by the relevant standards. However, modeling, analysis and verification of stochastic hybrid systems is a challenging task. One research line is to extend hybrid automata [17], which is the most popular model for traditional hybrid systems, by adding probability and stochasticity. Along this line, several different notions of *stochastic hybrid automata* have been proposed [1, 2, 4, 37, 14, 42, 9], with the difference on where the randomness is introduced. One option is to replace deterministic jumps by probability distribution over deterministic jumps. Another option is to generalize differential equations by stochastic differential equations (SDEs), which have been investigated in [20, 6, 1, 13]. More general models can be obtained by mixing the above two approaches, and by combining them with memoryless timed probabilistic jumps [5], or with a random reset function for each discrete

Correspondence and offprint requests to: Shuling Wang, wangsl@ios.ac.cn, Naijun Zhan, znj@ios.ac.cn, and Lijun Zhan, zhanglj@ios.ac.cn, Building 5, Software Park, No. 4, South 4th Street, Zhongguancun, Haidian District, Beijing, China.

jump [9]. As in classical setting, the verification of automata-based stochastic hybrid systems are normally achieved through reachability analysis, either by probabilistic model-checking [1, 2, 4, 37, 14, 42, 9], or by simulation, i.e., statistical model-checking [24, 45]. An overview of this line can be found in [5]. However, probabilistic model-checking of stochastic hybrid systems does not scale, in particular, taking SDEs into account. For example, it is not clear how to approximate the reachable sets of a simple linear SDEs with more than two variables. Therefore, existing verification techniques based on reachability analysis for stochastic hybrid systems have limitations. On the other hand, the statistical model-checking approach based on simulation may lead to possible unsoundness of analysis results due to numerical error and incomplete coverage.

In contrast, deductive methods increasingly attract more attention in the verification of stochastic hybrid systems as it can scale up to complex systems. The differential invariant generation for SDEs is at the core of deductive verification of stochastic hybrid systems. For pure hybrid systems, the invariant generation problem has been investigated extensively [7, 36, 35, 12, 40, 34, 10, 23]. In [32], Prajna et al. extend differential invariant generation approach based on barrier certificates for traditional hybrid systems to stochastic hybrid systems. Based on the differential invariants, the deductive verification method can be extended to hybrid systems and stochastic hybrid systems. A differential-algebraic dynamic logic for hybrid programs [29] was proposed by extending dynamic logic with continuous statements. Then in [30], the author presents a compositional stochastic differential dynamic logic for stochastic hybrid systems, and for the first time, proposes a special form of probabilistic differential invariants for SDEs. In [3], Hybrid Event-B is proposed by extending Event-B with continuous behaviors, and furthermore, a suite of proof obligations is defined for semantics and verification of Hybrid Event-B. However, stochasticity is not considered in the work.

Hybrid CSP (HCSP) [16, 43] is an extension of CSP [19] by introducing differential equations to model continuous evolution and communication interruptions in hybrid systems. In [22, 41, 39], the Hoare logic is extended to hybrid systems modeled by Hybrid CSP [16, 43] for deductive verification. In this paper, we extend HCSP by introducing probability and stochasticity and define stochastic HCSP (SHCSP), to model stochastic hybrid systems. In SHCSP, ordinary differential equations (ODEs) are generalized to stochastic differential equations (SDEs), and non-deterministic choice is replaced by probabilistic choice, and communication interrupts are generalized to communication interrupts with weights. Compared to other approaches, SHCSP is compositional, and provides more expressive constructs for describing hybrid systems, including communication, parallelism, interruption, and so on. For specifying and reasoning about SHCSP processes, we extend Hybrid Hoare Logic [22], which is an extension of Hoare logic [18] to hybrid systems, to stochastic hybrid systems.

This paper substantially extends the conference paper [28] in the following aspects:

- A more expressive form for SDEs is given, by using probabilistic formulas to define the domain of the SDEs instead of the deterministic formulas. Furthermore, in order to characterize the behavior of SDEs, we introduce the notion of (local) stochastic differential invariants that hold for all reachable states of SDEs. Based on the stochastic differential invariants, we define a new inference rule for reasoning about SDEs, which generalizes the one in [28].
- We propose the notion of (global) loop invariants for characterizing the repetition of SHCSP.
- To handle the probabilistic constructs such as probabilistic choice and communication interrupt with weights, we propose new operations on probabilistic states: conditional test and addition. In correspondence, we define new probabilistic formulas for describing such states. Based on the new formulas, the inference rule for conditional statement can be defined in a more generalized and elegant way.
- Because of the weights attached for communications in communication interrupts, a communication may occur with a probability. In [28], we restrict in the inference system that when any communication is ready to occur, then it occurs with probability 1. In this paper, we loose this restriction and allow any probability for a communication. This can be seen from the improved inference rules of the input and output events.

Related Work

Part of the related work on modelling and verification of stochastic hybrid system has been given above. Here we present some related work on deductive verification of programs with probability, which is the first step to face in the deductive verification of stochastic hybrid systems. The extension of CSP to probabilistic setting has been investigated by Morgan et al. [26]. In [25], a probabilistic predicate transformer is proposed for programs containing probabilistic choice for the first time. In [15], the author defines a Hoare-like logic for programs with probabilistic choice, based on a set of new probabilistic predicates such as conditional test, scaling, and so on. In [21], a compositional verification technique is presented for systems that exhibits both probabilistic and nondeterministic behaviour based on the assume-guarantee approach, and by reducing to the problem of multi-objective probabilistic model checking, the compositional

verification is fully automated. In [11], a weakest pre-expectation semantics is proposed for a simple probabilistic guarded command language. There are two forms of assertions for specifying the probabilistic states: one is based on probabilistic formulas, that intuitively expresses the probability with which a formula holds in syntactic level [15]; and the other is based on traditional formulas, that is evaluated to the expectations of the formulas in the semantics [25, 11]. In our work, we adopt the first approach, and different from other work, focus on the continuous-time world.

Organization

The rest of the paper is organized as follows: Sec. 2 presents some basic notions and results on probability and stochasticity. Sec. 3 introduces the stochastic Hybrid CSP for modelling stochastic hybrid systems, and in Sec. 4, the operational semantics of this modelling language is defined. Sec. 5 defines the assertions and specifications for specifying stochastic hybrid CSP, and Sec. 6 presents the main inference system. Section 7 addresses some future work and concludes the paper.

2. Background and Notations

Let Ω represent a non-empty sample set. A *σ-algebra* on set Ω is a set $\mathcal{F} \subseteq 2^\Omega$ such that: (i) $\emptyset, \Omega \in \mathcal{F}$; (ii) if $A \in \mathcal{F}$, then its complement $A^c \in \mathcal{F}$; and (iii) if $A_1, A_2, \cdots \in \mathcal{F}$, then $\bigcup_{i=1}^{\infty} A_i \in \mathcal{F}$. A *probability measure* P on the pair $(\Omega, \mathcal{F})$ is a function $P : \mathcal{F} \to [0, 1]$ such that $P(\emptyset) = 0,\ P(\Omega) = 1$, and if $A_1, A_2, \cdots \in \mathcal{F}$ are mutually disjoint, then $P(\bigcup_{i=1}^{\infty} A_i) = \sum_{i=1}^{\infty} P(A_i)$. For any set $A \in \mathcal{F}$, A is called an *event*. In the rest of the paper, we will use $A, A_1, \ldots$ to represent events. If $P(F) = 1$, we say the event F occurs with probability 1, or almost surely. The triple $(\Omega, \mathcal{F}, P)$ is called a *probability space*. It is called *a complete probability space*, if for all $A \in \mathcal{F}$ with $P(A) = 0$, $B \subset A$ implies $B \in \mathcal{F}$. In the following, we assume $(\Omega, \mathcal{F}, P)$ is a complete probability space, if not otherwise stated.

Given any family $\mathcal{U}$ of subsets of Ω, there is a smallest σ-algebra containing $\mathcal{U}$, and we call it the *σ-algebra generated by $\mathcal{U}$*. For instance, the σ-algebra on $\mathbb{R}^n$ that is generated by all open subsets of $\mathbb{R}^n$ is called *Borel σ-algebra*, denoted by $\mathcal{B}$. Given any function $X : \Omega \to \mathbb{R}^n$, the σ-algebra generated by X is the smallest σ-algebra on Ω containing all the sets $X^{-1}(B)$, that is defined as $\{\omega \in \Omega \mid X(\omega) \in B\}$, for any $B \in \mathcal{B}$. A function $Y : \Omega \to \mathbb{R}^n$ is called *$\mathcal{F}$-measurable*, if for any $B \in \mathcal{B}$, $Y^{-1}(B) \in \mathcal{F}$. A $\mathbb{R}^n$-valued *random variable* is a *$\mathcal{F}$-measurable* function $Y : \Omega \to \mathbb{R}^n$. A collection of families $\{\mathcal{H}_i\}_{i\in I}$ of measurable sets of $\mathcal{F}$ is independent, if for any $H_{i_j} \in \mathcal{H}_{i_j}$ with $1 \leq j \leq k$ and $i_j \in I$, $P(H_{i_1} \cap \cdots H_{i_k}) = P(H_{i_1}) \cdots P(H_{i_k})$. A collection of random variables $\{X_i\}_{i\in I}$ is independent if the collection of the σ-algebras generated by them is independent. We will use $U, U_1, \cdots$ to represent a collection of independent random variables which distribute uniformly in $[0, 1]$, which will be used in the definition of the semantics later.

Let *TS* $\subseteq \mathbb{R}$ represent a time set. A *stochastic process* X is a function X : *TS* $\times\, \Omega \to \mathbb{R}^n$ such that for each $t \in$ *TS*, $X(t, \cdot) : \Omega \to \mathbb{R}^n$ is a random variable, and for each $\omega \in \Omega$, $X(\cdot, \omega)$: *TS* $\to \mathbb{R}^n$ corresponds to a *sample path*. We will use X_t and X_ω to represent them respectively. In this paper, *TS* is a time interval, e.g. $[0, \infty)$, or $[a, b]$ for some $a, b \in \mathbb{R}^n$. A *filtration* on $(\Omega, \mathcal{F})$ is a family of σ-algebras $\{\mathcal{M}_t\}_{t\geq 0}$ that are increasing with $\mathcal{M}_{t_1} \subset \mathcal{M}_{t_2}$ for all $0 \leq t_1 < t_2$. A *Markov time* (or stopping time) with respect to a filtration $\{\mathcal{M}_t\}_{t\geq 0}$ is a random variable $\tau : \Omega \to [0, \infty)$ such that for any $t \geq 0$, $\{\omega : \tau(\omega) \leq t\} \in \mathcal{M}_t$, i.e. the event $\{\omega : \tau(\omega) \leq t\}$ is determined by (at most) the information up to time t. A stochastic process X is *adapted* to a filtration $\{\mathcal{M}_t\}_{t\geq 0}$ if X_t is $\mathcal{M}_t$-measurable. A function defined on $\mathbb{R}$ is *càdlàg* iff it is *right continuous* and has *left limit*. A stochastic process X is *càdlàg* iff all of its paths $t \to X_t(\omega)$ (for each $\omega \in \Omega$) are *càdlàg*.

A d-dimensional *Brownian motion* W is a stochastic process with $W_0 = 0$ that is continuous almost surely everywhere and has independent increments with time, i.e. $W_t - W_s \sim N(0, t-s)$ (for $0 \leq s < t$), where $N(0, t-s)$ denotes the normal distribution with mean 0 and variance $t - s$. Brownian motion W can be understood as the limit of a random walk. It is is almost surely continuous everywhere but differentiable nowhere.

We use *stochastic differential equation* (SDE) to model stochastic continuous evolution, which is of the form $dX_t = b(X_t)dt + \sigma(X_t)dW_t$, where W_t is a Brownian motion[1]. Intuitively, the drift coefficient $b(X_t) : \mathbb{R}^n \to \mathbb{R}^n$ determines how the deterministic part of X_t changes with respect to time and the diffusion coefficient $\sigma(X_t) : \mathbb{R}^n \to$

[1] Here we only consider time homogenous SDEs, as any general SDE can be reduced to this case by introducing additional variables [27].

$\mathbb{R}^{n\times m}$ determines the stochastic influence to X_t with respect to the Brownian motion W_t. A solution to an SDE is a stochastic process. The existence and uniqueness of solution to SDEs is guaranteed by the following theorem [27].

Theorem 2.1. Let $T > 0$, and $b(x)$ and $\sigma(x)$ be measurable functions. If the following two conditions are satisfied:

(a) there exists some constant C such that for any x, $|b(x)| + |\sigma(x)| \leq C(1 + |x|)$;
(b) there exists some constant D such that for any x, y, $|b(x) - b(y)| + |\sigma(x) - \sigma(y)| \leq D|x - y|$.

Let Z be a random variable satisfying $E[|Z|^2] < \infty$, then the SDE

$$dX_t = b(X_t)dt + \sigma(X_t)dW_t, \qquad 0 \leq t \leq T, X_0 = Z$$

has a unique continuous solution X_t with the property that, X_t is adapted to the filtration $\mathcal{F}_t^Z$ generated by Z and $(W_s)_{s\leq t}$, and $E[\int_0^T |X_t|^2 dt] \leq \infty$.

In the following parts, we always assume that the SDEs satisfy the conditions (a) and (b).

3. Stochastic HCSP

A system in Stochastic HCSP (SHCSP) consists of a finite set of sequential processes in parallel which communicate via channels synchronously. Each sequential process is represented as a collection of stochastic processes, each of which arises from the interaction of discrete computation and stochastic continuous dynamics modeled by SDEs. Let *Proc* represent the set of SHCSP processes, Σ the set of channel names. The syntax of SHCSP is given as follows:

$$\begin{array}{lcl} P, Q & ::= & \textbf{skip} \mid x := e \mid ch?x \mid ch!e \mid P; Q \mid B \to P \mid P^* \mid \\ & & P \sqcup_p Q \mid \langle ds = bdt + \sigma dW \& \textit{PB} \rangle \mid \langle ds = bdt + \sigma dW \& \textit{PB} \rangle \trianglerighteq []_{i\in I}(\omega_i \cdot ch_i* \to Q_i) \\ S & ::= & P \mid P_1 \| P_2 \| \cdots \| P_n \end{array}$$

Here $ch, ch_i \in \Sigma$, ch_i* stands for a communication event, e.g. $ch?x$ or $ch!e$, x for a discrete variable, B and *PB* for boolean and probabilistic boolean expressions, e for an arithmetic expression, $P, Q, Q_i \in \textit{Proc}$ for sequential processes, $p \in [0, 1]$ stands for the probability of the choice between P and Q, s for a vector of continuous variables, b and σ for functions of s, W for the Brownian motion process, $\omega_i \in \mathbb{Q}^+$ for the weight, and I for a non-empty finite set of indices, n for a natural number with $n > 1$. At the end, S stands for a SHCSP process, which is either a sequential process, or a parallel composition of multiple sequential processes. The sharing of variables between parallel processes is not allowed in (S)HCSP, and instead, the exchange of messages between them is achieved solely via synchronized communications along the common channels of them. Furthermore, to simplify the semantics of SHCSP, we assume that each input or output channel can only be possessed by one sequential process.

As defined in the syntax of P, the processes in the first line are original from HCSP, while the second line is new for SHCSP. The individual constructs can be understood intuitively as follows:

- **skip**, the assignment $x := e$, the sequential composition $P; Q$, and the conditional statement $B \to P$ are defined as usual.
- $ch?x$ receives a value along channel ch and assigns it to x, while $ch!e$ sends the value of e along channel ch. A communication takes place when both the sending and the receiving parties are ready, and may cause one side to wait. Note that the waiting time could be different at different samples in Ω due to the stochastic behaviour of the partner.
- The repetition P^* executes P for some finite number of times.
- $P \sqcup_p Q$ denotes probabilistic choice. It behaves as P with probability p and as Q with probability $1 - p$.
- $\langle ds = bdt + \sigma dW \& \textit{PB} \rangle$ specifies that the system evolves according to the stochastic process defined by the SDE $ds = bdt + \sigma dW$. *PB* defines the domain of s. It is a probabilistic boolean expression, corresponding to the quantifier-free probabilistic state formula defined in Sec. 5. As soon as *PB* turns false, the stochastic process terminates. Thus we call $\langle ds = bdt + \sigma dW \& \textit{PB} \rangle$ the *boundary interrupt* to the stochastic continuous evolution. For example, suppose *PB* is $P(s < s_m) > 99\%$, it means that within the domain, $s < s_m$ holds with probability greater than 99%. Whenever the probability that $s < s_m$ holds is equal to or less than 99%, the process terminates. For future use, we will denote by $\textit{dim}(s)$ the dimension of s.
- $\langle ds = bdt + \sigma dW \& \textit{PB} \rangle \trianglerighteq []_{i\in I}(\omega_i \cdot ch_i* \to Q_i)$ denotes the probabilistic communication interrupt, where $\omega_i \in \mathbb{Q}^+$ represents the *weight* of ch_i*. It behaves like $\langle ds = bdt + \sigma dW \& \textit{PB} \rangle$, except that the stochastic process

is preempted as soon as one of the communications ch_i* takes place, after that the respective Q_i is executed. However, if there are multiple communications along $\{ch_{i_j}\}_{1\leq j\leq n}$ with $i_j \in I$ and $n > 1$ that become ready simultaneously, then for each ch_{i_j}, the communication action $ch_{i_j}*$ occurs with probability $\frac{w_{i_j}}{\sum_{k=1}^{n} w_{i_k}}$.

The stochastic behavior of SHCSP originates from the stochastic continuous evolution, the probabilistic choice $\sqcup_p$, and the choice in probabilistic communication interrupt. Later we will see that, the semantics of the stochastic continuous evolution is defined by the solution of the SDE, and the semantics of the last two are defined by a family of independent random variables. As a result, the event that a communication action occurs, is independent from the event that its partner action occurs. If a communication action occurs with probability p, and its partner action occurs with probability q, then the corresponding communication will occur with probability $p * q$. The semantics of parallel composition indicates this result.

- $P_1 \| P_2$ behaves as if P_1 and P_2 run independently except that all communications along the common channels connecting P_1 and P_2 are to be synchronized.

Let S be a SHCSP process. We denote $\mathit{Var}(S)$ as the set of state variables, including both discrete and continuous variables, that occur in S. For any channel $ch \in \Sigma$ occurring in S, if both the input and output ends of ch, i.e. $ch?$ and $ch!$, occur in S, then ch is called an *internal* channel of S, otherwise an *external* channel of S. We denote $\Sigma(S)$ as the set of external channels of S. For a parallel process $S_1 \| S_2$ of SHCSP, we have the following facts:

$$\mathit{Var}(S_1) \cap \mathit{Var}(S_2) = \emptyset \qquad \mathit{Var}(S_1 \| S_2) = \mathit{Var}(S_1) \cup \mathit{Var}(S_2)$$
$$\Sigma(S_1 \| S_2) = (\Sigma(S_1) \cup \Sigma(S_2)) \setminus (\Sigma(S_1) \cap \Sigma(S_2))$$

An example Consider the classic plant-controller example in the stochastic setting: A plant is sensed by a computer periodically (say every d time units), and receives a control (u) from the digit controller immediately after the sensing. Thus, it can be modelled by the following SHCSP process:

$$((\langle ds = b(s,u)dt + \sigma(s,u)dW \& \textbf{True}\rangle \trianglerighteq (c_{p2c}!s \to \textbf{skip})); c_{c2p}?u)^* \parallel (\textbf{wait}\ d; c_{p2c}?x; c_{c2p}!e(x))^*$$

where $ds = b(s,u)dt + \sigma(s,u)dW$ describes the behaviour of the plant with stochastic influence, and $e(x)$ computes the control value based on the state x of the plant.

Remark 3.1. SHCSP provides two kinds of interrupts to stochastic continuous evolution modelled by SDEs: boundary interrupt and communication interrupt. When the interrupt occurs, the SDE stops and another process executes. But as we know, a physical plant that is undergoing evolution according to a (S)DE is not going to suddenly stop obeying its physical law and freeze its state for a positive duration of time. To avoid such design error, when we model a (stochastic) hybrid system using (S)HCSP, a continuous plant should be only interrupted by a piece of discrete activities, which compared to continuous evolution, the execution time is negligible thus is considered as zero in our semantics. The discrete computation acts as controllers sensing the state of the plant and calculating a new control value based on the state, and then the interrupted continuous evolution continues again by following the new control value in the next period. This is very consistent with designing a hybrid system in reality. Obviously the above plant-controller example indicates this point.

Not surprisingly, as a modeling language, SHCSP itself cannot guarantee the correctness of a model built with it. This is quite similar to most of programming languages, which cannot guarantee the correctness of programs coded with them. For example, the normal movement of a train follows a SDE $ds = vdt, dv = adt + \sigma dW$ subject to the constraint that the proportion of the velocity larger than the emergence brake intervention (V_{ebi}) is less than 0.1%, i.e., $P(v \leq V_{ebi}) > 99.9\%$, where s stands for the distance, v for the velocity, and σdW for the noise from the environment like friction, wind and so on. Whenever the condition is violated, an emergence brake should take place, which is modeled by $\langle dv = -Bdt\rangle$, where B stands for the full brake deceleration. But

$$\langle ds = vdt, dv = adt + \sigma dW \& P(v > V_{ebi}) > 99.9\%\rangle; \langle dv = -Bdt + \sigma dW \& v \geq 0\rangle$$

is not a correct design of the train, as the continuous evolution of s stops in case the first SDE terminates, but it still continuously evolves obeying $ds = vdt$ during the second SDE being executed in reality. There are two possible ways to avoid such design errors: to refine the modelling language to a small safe subset, or to develop more expressive verification techniques. We consider this as our future work.

3.1. A Running Example

We use SHCSP to model the aircraft position during the flight, which is adapted from [33]. Consider an aircraft that is following a flight path consisting of a sequence of line segments at a fixed altitude. Ideally, the aircraft should fly at a constant velocity v along the nominal path, but due to the wind or cloud disturbance, the deviation of the aircraft from the path may occur. For safety, the aircraft should follow a correction heading to get back to the nominal path as quickly as possible. On the one hand, the correction heading should be orthogonal to the nominal path for the shortest way back, but on the other hand, it should also go ahead to meet the destination. Considering these two objectives, we assume the correction heading is always an acute angle with the nominal path.

Here we model the behavior of the aircraft along one line segment. Without loss of generality, we assume the segment is along x-axis, with $(x_s, 0)$ as the starting point and $(x_e, 0)$ as the ending point. When the aircraft deviates from the segment with a vertical distance greater than λ, we consider it enters a dangerous state. Let (x_s, y_0) be the initial position of the aircraft in this segment, then the future position of the aircraft $(x(t), y(t))$ is governed by the following SDE:

$$\begin{pmatrix} dx(t) \\ dy(t) \end{pmatrix} = v \begin{pmatrix} cos(\theta(t)) \\ sin(\theta(t)) \end{pmatrix} dt + dW(t)$$

where $\theta(t)$ is the correction heading and is defined with a constant degree $\frac{\pi}{4}$ when the aircraft deviates from the nominal path:

$$\theta(t) = \begin{cases} -\frac{\pi}{4} & \text{if } y(t) > 0 \\ 0 & \text{if } y(t) = 0 \\ \frac{\pi}{4} & \text{if } y(t) < 0 \end{cases}$$

Let *PB* be $P(x_s \leq x \leq x_e) = 1$, the movement of the aircraft described above can be modelled by the following SHCSP process P_{Air}:

$$x := x_s; y := y_0; \langle [dx, dy]^T = v[cos(\theta(t)), sin(\theta(t))]^T dt + dW(t) \& \mathit{PB} \rangle$$

which states that, at the beginning, the position of the aircraft is initialized to (x_s, y_0), then the variables x, y standing for the position of the aircraft will evolve by conforming to the given SDE, till the violation of *PB*.

We will apply our approach to specify and verify this example later in the paper.

4. Operational Semantics

Before giving operational semantics, we introduce several auxiliary variables, states, and the operations and functions manipulating states.

Auxiliary Variables In order to define the semantics of SHCSP processes, we need to introduce several auxiliary variables. First of all, we use non-negative reals $\mathbb{R}^+$ to model time, and introduce a global clock now to record the time in the execution of a process.

Secondly, we introduce a variable tr to represent the timed trace accumulated till the current time (recorded by now) of a process. We define a *timed communication* as $\langle ch, b \rangle$, where $ch \in \Sigma$ and $b \in \mathbb{R}^+$, representing that a communication along channel ch occurs at time b. The set $\Sigma \times \mathbb{R}^+$ of all timed communications is denoted by $T\Sigma$. The set of all timed traces is

$$T\Sigma^*_{\leq} = \{\gamma \in T\Sigma^* \mid \text{ if } \langle ch_1, b_1 \rangle \text{ precedes } \langle ch_2, b_2 \rangle \text{ in } \gamma, \text{ then } b_1 \leq b_2\}.$$

If $C \subseteq \Sigma$, $\gamma \restriction_C$ is the projection of γ onto C such that only the timed communications along channels in C of γ are preserved. Given two timed traces γ_1, γ_2, a channel set $C \subseteq \Sigma$, the *alphabetized parallel* of γ_1 and γ_2 over C, denoted by $\gamma_1 \underset{C}{\parallel} \gamma_2$, results in the following set of timed traces

$$\{\gamma \mid \gamma \restriction_{\Sigma - (\Sigma(\gamma_1) \cup \Sigma(\gamma_2))} = \epsilon, \gamma \restriction_{\Sigma(\gamma_1)} = \gamma_1, \gamma \restriction_{\Sigma(\gamma_2)} = \gamma_2 \text{ and } \gamma \restriction_C = \gamma_1 \restriction_C = \gamma_2 \restriction_C\},$$

where $\Sigma(\gamma)$ stands for the set of channels that occur in γ. In this paper, the set C is usually $\Sigma(\gamma_1) \cap \Sigma(\gamma_2)$.

Finally, we introduce a variable rdy to represent the ready set of communication events at current time of a process. To model synchronization of communication events, we need to describe the readiness of them. We define

two forms of readiness: $ch?$, representing that the input event along $ch?$ becomes ready, and $ch!c$, representing that the output event along $ch!$ becomes ready and furthermore as a sender, it will send a value c to the receiver. When both parties along ch become ready, a communication along ch occurs, taking zero time to complete. At a time point, multiple communications may occur. In order to record the execution order of communications occurring at the same time point, we prefix each communication readiness occurrence, with a timed trace that happened before the communication event being ready. So, each *communication readiness* has the form of $\gamma.ch?$ or $\gamma.ch!c$, where $\gamma \in T\Sigma^*_{\leq}$. We denote by ***RDY*** the set of communication readinesses in the sequel.

In the operational semantics, we will define the values of now, tr, and rdy during the process execution. Furthermore, the variables now and tr will be used in defining the proof system, of SHCSP respectively. In what follows, we use $\mathit{Var}(S)^+$ to represent the set of state variables of S, i.e. $\mathit{Var}(S)$, plus the auxiliary variables $\{\mathit{rdy}, \mathit{tr}, \mathit{now}\}$ introduced above. For a parallel process $S_1 \| S_2$, we have the result that $\mathit{Var}(S_1)^+ \cap \mathit{Var}(S_2)^+ = \{\mathit{rdy}, \mathit{tr}, \mathit{now}\}$. The variables in $\mathit{Var}(S)$ will have values in $\mathbb{R}$, *rdy* in ***RDY***, *tr* in $T\Sigma^*_{\leq}$, and *now* in $\mathbb{R}^+$, respectively. We denote the set of all values by *Val*.

Remark 4.1. Notice that the auxiliary variables $\mathit{rdy}, \mathit{tr}, \mathit{now}$ are all at the meta level, which are only used in the definitions of the execution semantics and proof system of a SHCSP process, and never occur in the syntax of any SHCSP process.

States To interpret a process $S \in \mathit{Proc}$, we first define a deterministic state *sd* as a mapping from $\mathit{Var}(S)^+$ to *Val*, and denote by $\mathcal{D}$ the set of such states. Because of stochasticity, we introduce a random variable $\rho : \Omega \to \mathcal{D}$ to describe a *probabilistic state*, i.e. a distribution of all possible states. Notice that the random variable for representing probabilistic states is different from the standard R^n-valued random variables, however, it can be transformed to the standard one by abstracting away the variables $\mathit{Var}(S)^+$ in $\mathcal{D}$ and instead assume each variable has a unique number. This approach was adopted in [30]. In addition,we introduce a stochastic process $H : \mathit{Intv} \times \Omega \to \mathcal{D}$, called *flow*, to represent the continuous flow of process S over the time interval *Intv*, i.e., state distributions on the interval.

To handle conditional and probabilistic choice, we allow the existence of *sub-probabilistic states*, which are mappings from a subset of Ω to $\mathcal{D}$. It can be seen that they will only occur as intermediate states in the operational semantics of SHCSP. In what follows, we will also call the sub-probabilistic states as (probabilistic) states if not stated otherwise.

State Operations In order to define the semantics of probabilistic choice and conditional choice, we need to define the following operations on probabilistic states correspondingly.

Definition 4.1. Assume ρ, ρ_1, ρ_2 are probabilistic states, the conditional $B?\rho$ and the addition $\rho_1 + \rho_2$ are defined as follows (The semantics of formulas B will be given in Sec. 5):

$$B?\rho(\omega) = \begin{cases} \rho(\omega) & \text{if } B \text{ is true under } \rho(\omega) \\ \bot & \text{otherwise} \end{cases}$$

When $\mathrm{dom}(\rho_1)$ and $\mathrm{dom}(\rho_2)$ are disjoint,

$$\rho_1 + \rho_2(\omega) = \begin{cases} \rho_1(\omega) & \text{if } \omega \in \mathrm{dom}(\rho_1) \\ \rho_2(\omega) & \text{if } \omega \in \mathrm{dom}(\rho_2) \\ \bot & \text{otherwise} \end{cases}$$

The state $B?\rho$ is obtained from ρ by removing the samples whose corresponding states do not satisfy B, thus it is partial. The addition $\rho_1 + \rho_2$ is used for the case that the sample domains of ρ_1 and ρ_2 are disjoint, and in such case, it is the union of ρ_1 and ρ_2.

Below we introduce some more functions for manipulating states. Given two states ρ_1 and ρ_2, let A_i be the sample set $\mathrm{dom}(\rho_i)$ for $i = 1, 2$, then we say ρ_1 and ρ_2 are parallelable iff for each $\omega \in A_1 \cap A_2$, $\mathrm{dom}(\rho_1(\omega)) \cap \mathrm{dom}(\rho_2(\omega)) = \{\mathit{rdy}, \mathit{tr}, \mathit{now}\}$, i.e. the state variables of ρ_1 and ρ_1 are disjoint, and $\rho_1(\omega)(\mathit{now}) = \rho_2(\omega)(\mathit{now})$, i.e. the execution time of ρ_1 and ρ_1 is equal. Given two parallelable states ρ_1 and ρ_2, paralleling them over $C \subseteq \Sigma$ results in a set of new states, denoted by $\rho_1 \uplus \rho_2$. For any $\omega \in \mathrm{dom}(\rho_1) \cap \mathrm{dom}(\rho_2)$, any $v \in \mathrm{dom}(\rho_1(\omega)) \cup \mathrm{dom}(\rho_2(\omega))$, each of ρ in $\rho_1 \uplus \rho_2$ satisfies:

$$\rho(\omega)(v) \stackrel{\text{def}}{=} \begin{cases} \rho_1(\omega)(v) & \text{if } v \in \mathrm{dom}(\rho_1(\omega)) \setminus \mathrm{dom}(\rho_2(\omega)), \\ \rho_2(\omega)(v) & \text{if } v \in \mathrm{dom}(\rho_2(\omega)) \setminus \mathrm{dom}(\rho_1(\omega)), \\ \rho_1(\omega)(\mathit{now}) & \text{if } v = \mathit{now}, \\ \gamma, \text{ where } \gamma \in \rho_1(\omega)(\mathit{tr}) \underset{C}{\|} \rho_2(\omega)(\mathit{tr}) & \text{if } v = \mathit{tr}, \\ \rho_1(\omega)(\mathit{rdy}) \cup \rho_2(\omega)(\mathit{rdy}) & \text{if } v = \mathit{rdy}. \end{cases}$$

It makes no sense to distinguish any two states in $\rho_1 \uplus \rho_2$, so hereafter we will use $\rho_1 \uplus \rho_2$ to represent any of its elements. In the operational semantics, $\rho_1 \uplus \rho_2$ will be used to represent the states of parallel processes.

Given a probabilistic state ρ, the update $\rho[v \to e]$ represents a new probabilistic state such that for any $\omega \in \Omega$ and $x \in \mathit{Var}$, $\rho[v \to e](\omega)(x)$ is defined as the value of e under state $\rho(\omega)$ if x is v, and $\rho(\omega)(x)$ otherwise. For simple use, we will write $\rho[tr+\tau]$ as an abbreviation of $\rho[tr \mapsto tr \cdot \langle \tau, now \rangle]$. Given a stochastic process $X : [0, d) \times \Omega \to \mathbb{R}^{dim(s)}$, where $d \in \mathbb{R}^+$, then for any t in the domain, $\rho[s \to X_t]$ is a new probabilistic state such that for any $\omega \in \Omega$ and $x \in \mathit{Var}$, $\rho[s \to X_t](\omega)(x)$ is defined as $X(t, w)$ if x is s, and $\rho(\omega)(x)$ otherwise. Given a stochastic process H over interval $[a, b]$, where $a \leq b$ and $a, b \in \mathbb{R}^+$, $H[v \mapsto e]$ is the stochastic process defined over the same interval such that for any $t \in [a, b]$, for any ω, for any variable $x \in \mathit{Var}$, $H[v \mapsto e](t, \omega)(x)$ is defined as $H(t, \omega)(x)$ if x is not v, otherwise the value of e under the starting state $H(a, \omega)$.

Other Functions In the semantics we will use a family of random variables $\{U_i\}_{1 \leq i \leq N}$ that are distributed uniformly in $[0, 1]$, and assume that for each U_i, it is independent of other U_js and other random variables occurring in the semantics. Moreover, given an arbitrary set G, we define $\mathcal{I}_G$ to represent the characteristic function of G, i.e. $\mathcal{I}_G(x) = 1$ if $x \in G$ and $\mathcal{I}_G(x) = 0$ otherwise.

4.1. Operational Semantics

Each transition relation has the form of $(P, \rho) \xrightarrow{\alpha} (P', \rho', H)$, where P and P' are SHCSP processes, α is an event, ρ, ρ' are probabilistic states, H is a stochastic process (and we will call H a *flow*). It expresses that, starting from initial state ρ, by performing event α, P evolves into P', ends in state ρ', and produces the execution flow H. When the transition is discrete and thus produces a flow on a point interval (i.e. current time now), we will write $(P, \rho) \xrightarrow{\alpha} (P', \rho')$ instead of $(P, \rho) \xrightarrow{\alpha} (P', \rho', \{\rho(now) \mapsto \rho'\})$, without losing any substantial information. The label α represents events, which can be a discrete non-communication event, e.g. skip, assignment, or the evaluation of boolean expressions, uniformly denoted by τ, or an external communication event $ch!c$ or $ch?c$, or an internal communication $ch.c$, or a time delay d, where $c \in \mathbb{R}, d \in \mathbb{R}^+$. When both $ch!c$ and $ch?c$ occur, a communication $ch.c$ occurs. We call the events other than the time delay *discrete events*, and will use β to range over them. We define the dual of $ch?c$ (denoted by $\overline{ch?c}$) as $ch!c$, and vice versa, and define $\mathit{comm}(ch!c, ch?c)$ or $\mathit{comm}(ch?c, ch!c)$ as the communication $ch.c$. In the operational semantics, besides the timed communications, we will also record the τ events that have occurred till now in tr.

Because of probabilistic and conditional choices, a transition may only occur for some event, i.e. a subset of Ω. We will write $(P, \rho) \xrightarrow{\alpha} (P', \rho', H)$ *on event* A, where $A \subseteq \mathrm{dom}(\rho)$, to mean that $(P, \rho) \xrightarrow{\alpha} (P', \rho'|_A, H|_A)$, i.e. the sample set is reduced to A after the transition is performed. As a consequence, for any transition, the states before and after its execution can be sub-probabilistic states.

The semantics for SHCSP is presented in Table 1 and Table 2. The semantics of **skip** and $x := e$ are defined as usual, except that for each, a τ event occurs. The process ϵ can stay idle for an arbitrarily long time, and then evolves to itself. The rules for input $ch?x$ indicate the following three cases:

- at the very beginning of execution, the input event has to be put in the ready set, and thereafter,
- it may wait for its parter for some time d during which it remains ready. Notice that for different ωs, the waiting time might be different, thus in the second rule, the initial state ρ might be a sub-probabilistic state. The waiting process produces an execution history H^ρ_d, which is a stochastic process such that for any ω, for any $t \in [\rho(\omega)(now), \rho(\omega)(now) + d]$, $H^\rho_d(t, \omega) = \rho[now \mapsto t](\omega)$.
- as soon as its parter becomes ready, a communication occurs immediately and takes zero time to complete, with x being assigned with the value received and tr extended by the timed communication.

The semantics of output $ch!e$ is similarly defined by three rules.

For stochastic continuous evolution $\langle ds = bdt + \sigma dW \& \mathit{PB} \rangle$, suppose $X : [0, \infty) \times \Omega \to \mathbb{R}^{dim(s)}$ is the solution of $ds = bdt + \sigma dW$ with the initial value $X_0 = \rho(s)$, then for any $t \in [0, \infty)$, X_t is a random variable that records the value distribution of s at time t. For any $d > 0$, the stochastic process is able to evolve for d time, if *PB* holds within the period $[0, d)$. In particular, at any time $t \in [0, d)$, the state becomes $\rho[now \mapsto now + t, s \mapsto X_t]$, where now is increased by t and the continuous variable s replaced by the solution X_t, and *PB* holds at time $now + t$ if $[\![\mathit{PB}]\!]^{\rho[now \mapsto now+t, s \mapsto X_t]}_L$ is true. Here L is a random variable for interpreting logical variables in the semantics of SHCSP, but actually, it is not used for the evaluation of *PB*. The semantics of probability formulas is given in next

(Skip) $(\textbf{skip}, \rho) \xrightarrow{\tau} (\epsilon, \rho[tr + \tau])$

(Idle) $(\epsilon, \rho) \xrightarrow{d} (\epsilon, \rho[now \mapsto now + d])$

(Assign) $(x := e, \rho) \xrightarrow{\tau} (\epsilon, \rho[x \mapsto e, tr \mapsto tr \cdot \langle \tau, now \rangle])$

(Input-1)

$$\frac{\rho(tr).ch? \notin \rho(rdy)}{(ch?x, \rho) \xrightarrow{\tau} (ch?x, \rho[rdy \mapsto rdy \cup \{tr.ch?\}])}$$

(Input-2) For any $d > 0$,

$$\frac{\rho(tr).ch? \in \rho(rdy)}{(ch?x, \rho) \xrightarrow{d} \left(ch?x, \rho[now \mapsto now + d], H_d^{\rho}\right)}$$

(Input-3)

$$\frac{\rho(tr).ch? \in \rho(rdy)}{(ch?x, \rho) \xrightarrow{ch?b} \left(\epsilon, \quad \rho[x \mapsto b, tr \mapsto tr \cdot \langle ch, now \rangle] \quad\right)}$$

(Output-1)

$$\frac{\rho(tr).ch! \notin \rho(rdy)}{(ch!e, \rho) \xrightarrow{\tau} (ch!e, \rho[rdy \mapsto rdy \cup \{tr.ch!e\}])}$$

(Output-2) For any $d > 0$,

$$\frac{\rho(tr).ch! \in \rho(rdy)}{(ch!e, \rho) \xrightarrow{d} \left(ch!e, \rho[now \mapsto now + d], H_d^{\rho}\right)}$$

(Output-3)

$$\frac{\rho(tr).ch! \in \rho(rdy)}{(ch!e, \rho) \xrightarrow{ch!e} \left(\epsilon, \quad \rho[tr \mapsto tr \cdot \langle ch, now \rangle] \quad\right)}$$

(SDE-1)

$$\frac{\begin{array}{c} X : [0, \infty) \times \Omega \to \mathbb{R}^{d(s)} \text{ is the solution of } ds = bdt + \sigma dW \\ \text{with } X_0 = \rho(s) \wedge \forall d > 0. \forall t \in [0, d), [\![PB]\!]_L^{\rho[now \mapsto now + t, s \mapsto X_t]} = \textbf{True} \end{array}}{(\langle ds = bdt + \sigma dW \& PB \rangle, \rho) \xrightarrow{d} \left(\begin{array}{l} \langle ds = bdt + \sigma dW \& PB \rangle, \\ \rho[now \mapsto now + d, s \mapsto X_d], H_d^{\rho, s, X} \end{array}\right)}$$

(SDE-2)

$$\frac{[\![cl(\neg PB)]\!]_L^{\rho} = \textbf{True}}{(\langle ds = bdt + \sigma dW \& PB \rangle, \rho) \xrightarrow{\tau} (\epsilon, \rho[tr + \tau])}$$

In the following, denote $\langle ds = bdt + \sigma dW \& PB \rangle \trianglerighteq [\!]_{i \in I} (\omega_i \cdot ch_i * \to Q_i)$ by CI.

(ProbInterrupt-1)

$$\frac{\forall i \in I. \rho(tr).ch_i * \notin \rho(rdy)}{(CI, \rho) \xrightarrow{\tau} (CI, \rho[rdy \mapsto rdy \cup_{i \in I} \{tr.ch_i *\})]}$$

(ProbInterrupt-2)

$$\frac{\begin{array}{c} \forall i \in I. \rho(tr).ch_i * \in \rho(rdy), \\ \{\overline{ch_{i_k} *}\}_{1 \le k \le n} \text{ become ready simultaneously while others not, } i.e. \\ \forall k \in \{1, \cdots, n\}. (ch_{i_k} *, \rho) \xrightarrow{ch_{i_k} *} (\epsilon, \rho_{i_k}, H_{i_k}), \\ U : \Omega \to [0, 1] \text{ is a random variable distributed uniformly in } [0, 1] \end{array}}{\begin{array}{c} (CI, \rho) \xrightarrow{ch_{i_j} *} (Q_{i_j}, \rho_{i_j}, H_{i_j}) \text{ on event} \\ \{(\sum_{k=1}^{j-1} \omega_{i_k} / \sum_{k=1}^{n} \omega_{i_k}) \le U < (\sum_{k=1}^{j} \omega_{i_k} / \sum_{k=1}^{n} \omega_{i_k})\} \text{ for } j \in \{1, \cdots, n\} \end{array}}$$

(ProbInterrupt-3)

$$\frac{\begin{array}{c} \forall i \in I. \rho(tr).ch_i * \in \rho(rdy), \\ \neg \exists i \in I. (ch_i *, \rho) \xrightarrow{ch_i *} (\epsilon, \rho_i, H_i), \\ (\langle ds = bdt + \sigma dW \& PB \rangle, \rho) \xrightarrow{d} (\langle ds = bdt + \sigma dW \& PB \rangle, \rho_s, H_s) \end{array}}{(CI, \rho) \xrightarrow{d} (CI, \rho_s, H_s)}$$

(ProbInterrupt-4)

$$\frac{\begin{array}{c} \forall i \in I. \rho(tr).ch_i * \in \rho(rdy), \\ \neg \exists i \in I. (ch_i *, \rho) \xrightarrow{ch_i *} (\epsilon, \rho_i, H_i), \\ (\langle ds = bdt + \sigma dW \& PB \rangle, \rho) \xrightarrow{\tau} (\epsilon, \rho') \end{array}}{(CI, \rho) \xrightarrow{\tau} (\epsilon, \rho'[rdy \mapsto rdy \setminus (\cup_{i \in I} \{tr.ch_i *\})])}$$

Table 1. The operational semantics for atomic constructs of HCSP

(ProbChoice)

$$\frac{U:\Omega\to[0,1] \text{ is a random variable distributed uniformly in } [0,1]}{(P\sqcup_p Q,\rho)\to\begin{cases}(P,\rho) \text{ on event } \{U\le p\}\\ (Q,\rho) \text{ on event } \{U>p\}\end{cases}}$$

Assume the initial states ρ_1 and ρ_2 are parallelable,

(Parallel-1)

$$\frac{(P_1,\rho_1)\xrightarrow{ch*}(P_1',\rho_1'),\ \text{and}\ (P_2,\rho_2)\xrightarrow{\overline{ch*}}(P_2',\rho_2')}{(P_1\parallel P_2,\rho_1\uplus\rho_2)\xrightarrow{comm(ch*,\overline{ch*})}(P_1'\parallel P_2',\rho_1'\uplus\rho_2')}$$

(Parallel-2)

$$\frac{(P_1,\rho_1)\xrightarrow{\beta}(P_1',\rho_1'),\quad \Sigma(\beta)\notin\Sigma(P_1)\cap\Sigma(P_2)}{(P_1\parallel P_2,\rho_1\uplus\rho_2)\xrightarrow{\beta}(P_1'\parallel P_2,\rho_1'\uplus\rho_2)}$$

(Parallel-3)

$$\frac{(P_i,\rho_i)\xrightarrow{d}(P_i',\rho_i',H_i),\ \text{for } i=1,2}{(P_1\parallel P_2,\rho_1\uplus\rho_2)\xrightarrow{d}(P_1'\parallel P_2',(\rho_1'\uplus\rho_2'),H_1\uplus H_2)}$$

(Parallel-4) $(\epsilon\parallel\epsilon,\rho_1\uplus\rho_2)\xrightarrow{\tau}(\epsilon,\rho_1\uplus\rho_2)$

(Sequential-1)

$$\frac{(P,\rho)\xrightarrow{\alpha}(P',\rho',H)\quad P'\neq\epsilon}{(P;Q,\rho)\xrightarrow{\alpha}(P';Q,\rho',H)}$$

(Sequential-2)

$$\frac{(P,\rho)\xrightarrow{\alpha}(\epsilon,\rho',H)}{(P;Q,\rho)\xrightarrow{\alpha}(Q,\rho',H)}$$

(Conditional)

$$\frac{[\![B]\!]_L^\rho:\Omega\to\{0,1\} \text{ is a random variable}}{(B\to P,\rho)\xrightarrow{\tau}\begin{cases}(P,\rho[tr+\tau]) \text{ on event } \{[\![B]\!]_L^\rho=1\}\\ (\epsilon,\rho[tr+\tau]) \text{ on event } \{[\![B]\!]_L^\rho=0\}\end{cases}}$$

(Reptition-1)

$$\frac{(P,\rho)\xrightarrow{\alpha}(P',\rho',H)\quad P'\neq\epsilon}{(P^*,\rho)\xrightarrow{\alpha}(P';P^*,\rho',H)}$$

(Reptition-2)

$$\frac{(P,\rho)\xrightarrow{\alpha}(\epsilon,\rho',H)}{(P^*,\rho)\xrightarrow{\alpha}(P^*,\rho',H)}$$

(Reptition-3) $(P^*,\rho)\to(\epsilon,\rho)$

Table 2. The operational semantics for compound constructs of HCSP

section. Let *cl*($\neg$*PB*) denote the closure of $\neg$*PB* that includes the boundary of $\neg$*PB*. For example, if *PB* is $P(s>1)\le 0.3$, then $\neg$*PB* is $P(s>1)>0.3$, and *cl*($\neg$*PB*) is $P(s>1)\ge 0.3$. Whenever *cl*($\neg$*PB*) becomes true, the stochastic continuous evolution terminates. The stochastic continuous evolution produces an execution history $H_d^{\rho,s,X}$, which is a stochastic process such that for any ω, for any $t\in[\rho(\omega)(now),\rho(\omega)(now)+d]$, $H_d^{\rho,s,X}(t,\omega)=\rho[now\mapsto t, s\mapsto X_t](\omega)$.

Remark 4.2. Notice that for theoretical simplicity, we assume that the change between different physical principles takes no time, therefore once a stochastic flow breaches the terms of its boundary condition, the process just stops immediately. Similar assumption is normally adopted in dynamical and hybrid systems without noise. But in practice, we have to consider the delay from one physical principle changing to another one, as it may prompt oscillations in otherwise convergently stable feedback loops or vice versa, they can destabilize otherwise stable orbits [38], can

stretch dwell times, may induce residual error that never settles, or can cause transient overshoot into unsafe operational regimes, to name just a few of the various possible effects fundamentally altering system dynamics. Unmodeled delays in a control loop thus have the potential to invalidate any stability or safety certificate obtained on the delay-free model, as delays may significantly deteriorate control performance. In our previous work, some approaches on modelling and verifying delay dynamical and hybrid systems without noise have been proposed [44, 8]. We believe that our previous approaches can be extended to stochastic settings, that will be discussed in another paper.

For probabilistic communication interrupt, there are four cases:

- at the very beginning, all the communication events $\{ch_i*\}_{i\in I}$ are put into the ready set;
- if there are n communications along $\{ch_{i_k}\}_{1\le k\le n}$ ready to occur, then the j-th communication over channel ch_{i_j} followed by Q_{i_j} is chosen to execute, for the samples ω satisfying $\frac{\sum_{k=1}^{j-1}\omega_{i_k}}{\sum_{k=1}^{n}\omega_{i_k}} \le U(\omega) \le \frac{\sum_{k=1}^{j}\omega_{i_k}}{\sum_{k=1}^{n}\omega_{i_k}}$;
- if there is no communication event ready to occur, the process evolves for d time units according to the SDE;
- if there is no communication event ready to occur, as soon as the stochastic continuous evolution terminates, the whole process also terminates by taking a τ event.

The semantics for probabilistic choice is defined by two cases, depending on the value of $U(\omega)$ for each ω : if $U(\omega) \le p$, then P is taken, otherwise, Q is taken. For $P_1 \| P_2$, we always assume that the initial states ρ_1 and ρ_2 are parallelable. There are four rules:

- P_1 and P_2 together perform a synchronized communication. Especially, for the case of synchronization, the state $\rho_1' \uplus \rho_2'$ produced after the communication is taken, guarantees that it is well defined for the common sample set of ρ_1' and ρ_2'.
- P_1 may progress separately on internal events or external communication events, and the symmetric case can be defined similarly (omitted here);
- both P_1 and P_2 evolve for d time units in case they can progress for d time units respectively;
- $P_1 \| P_2$ terminates when both P_1 and P_2 terminate. The rules of parallel composition obey the priority property.

At last, the semantics for sequential, internal choice, and repetition is defined as usual. For conditional choice $B \to P$, for samples ω, if B holds, then P will execute, otherwise it terminates.

Flow of a Process Given two flows H_1 and H_2 defined on $[r_1, r_2]$ and $[r_2, r_3]$ (or $[r_2, \infty)$) respectively, we define the concatenation $H_1^\frown H_2$ as the flow defined on $[r_1, r_3]$ (or $[r_1, \infty)$) such that $H_1^\frown H_2(t)$ is equal to $H_1(t)$ if $t \in [r_1, r_2)$, otherwise $H_2(t)$. Given a process P and an initial state ρ, if we have the following sequence of transitions:

$$(P,\rho) \xrightarrow{\alpha_0} (P_1,\rho_1,H_1) \quad (P_1,\rho_1) \xrightarrow{\alpha_1} (P_2,\rho_2,H_2)\ldots \quad (P_{n-1},\rho_{n-1}) \xrightarrow{\alpha_{n-1}} (P_n,\rho_n,H_n)$$

then we define $H_1^\frown \ldots^\frown H_n$ as the *flow* from P to P_n with respect to the initial state ρ, and furthermore, write $(P,\rho) \xrightarrow{\alpha_0\cdots\alpha_{n-1}} (P_n,\rho_n,H_1^\frown \ldots^\frown H_n)$ to represent the whole transition sequence (and for simplicity, the label sequence can be omitted sometimes). When P_n is ϵ, we call $H_1^\frown \ldots^\frown H_n$ a *complete flow* of P with respect to ρ.

The following theorem indicates that the semantics of SHCSP is well defined.

Theorem 4.1. If we have $(P,\rho) \to (\epsilon,\rho',H)$, then H is an almost surely $càdlàg$ process and adapted to the filtration $(\mathcal{F}_t)_{t\ge 0}$ generated by ρ, the Brownian motion $(W_s)_{s\le t}$, and uniform U processes, and furthermore, the execution time of P, denoted by $\Delta(P)$, is a Markov time with the filtration $(\mathcal{F}_t)_{t\ge 0}$.

Proof. We will prove the $càdlàg$, adaptedness and Markov time properties by induction on the structure of SHCSP P.

- Cases **skip**, ϵ and $x := e$: Deterministic times $\Delta(\textbf{skip}) = \Delta(x = e) = 0$, and $\Delta(\epsilon) = d$ are trivial Markov times. For all of them, H is adapted to the filtration generated by ρ. For **skip** and $x := e$, H is trivially $càdlàg$ as the time domain is $\{0\}$. For ϵ, all variables are preserved the same but now is progressing with derivative 1, H is obviously $càdlàg$.
- Case $ch?x$: According to the semantics, for each $\omega \in \Omega$, there must exist some $d > 0$ such that $\Delta(ch?x)(\omega) = d$ for the ω. For any $t \ge 0$, consider the set $\{\omega : \Delta(ch?x)(\omega) \le t\}$, denoted by Ω_t. Intuitively, Ω_t includes the samples for which the communications have occurred by time t, and does not include the samples for which the communications occur after time t in the future. Obviously, $\Omega_t \in \mathcal{F}_t$, thus, $\Delta(ch?x)$ is a Markov time. H combines the waiting stage and the communication at the termination, obviously it is $càdlàg$ and adapted to the filtration generated by ρ.

- The case for output can be proved similarly.
- Case $\langle ds = bdt + \sigma dW \& \boldsymbol{PB}\rangle$: By Theorem 2.1, H is adapted to the filtration generated by $(W_s)_{s\le t}$ and ρ, and it is continuous. By Theorem 5.1, ***PB*** is well defined, and there exists some d such that ***PB*** turns 0 for the first time and $\Delta(\langle ds = bdt + \sigma dW \& \boldsymbol{PB}\rangle) = d$, which is a Markov time.
- Case $B \to P$: If B is true, $B \to P$ executes P, otherwise, terminates immediately. By induction hypothesis, $\Delta(P)$ is a Markov time and H_P is *càdlàg* and adapted. According to the semantics, $\Delta(B \to P) = B?\Delta(P) + \neg B?0$, for which $B?$ is 1 for the samples that make B true, and $\neg B?$ is 1 for the samples that make B false. Thus, the sum is a Markov time. Moreover, we also have $H = B?H_P + \neg B?H_\rho$, where H_ρ stands for the singleton process with ρ. By induction hypothesis, H is *càdlàg*, and H is also adapted, where B generates the filtration.
- Case $P \sqcup_p Q$: By induction hypothesis, $\Delta(P)$ and $\Delta(Q)$ are both Markov times. According to the semantics, $\Delta(P \sqcup_p Q) = \mathcal{I}_{U\le p}\Delta(P) + \mathcal{I}_{U>p}\Delta(Q)$. The characteristic functions $\mathcal{I}_{U\le p}$ and $\mathcal{I}_{U>p}$ both have two values 0 and 1, thus $\mathcal{I}_{U\le p}\Delta(P)$ and $\mathcal{I}_{U>p}\Delta(Q)$ are Markov times. The sum of two Markov times, $\Delta(P \sqcup_p Q)$, is also a Markov time. By induction hypothesis, H_P for P and H_Q for Q are both *càdlàg*. Because *càdlàg* functions form an algebra, $H = \mathcal{I}_{U\le p}H_P + \mathcal{I}_{U>p}\Delta(Q)H_Q$ is also *càdlàg*. H is adapted, because H_P and H_Q are adapted and the choice $\sqcup_p$ generates the filtration.
- Case $P; Q$: Suppose $(P; Q, \rho) \xrightarrow{\alpha} (Q, \rho', H')$ and $(Q, \rho') \xrightarrow{\alpha} (\epsilon, \rho'', H'')$. By induction hypothesis, $\Delta(P)$ is a Markov time and H' is *càdlàg* and adapted to $(\mathcal{F'}_t)_{t\ge 0}$ generated by ρ and the constituent Brownian motion and uniform processes during P, and especially ρ' is a random variable. By induction hypothesis, $\Delta(Q)$ is a Markov time and H'' is *càdlàg* and adapted to $(\mathcal{F''}_{t-\Delta(P)})_{t\ge\Delta(P)}$ generated by ρ' and the constituent Brownian motion and uniform processes during Q. Obviously, $\Delta(P; Q) = \Delta(P) + \Delta(Q)$ is a Markov time. $H = H' \frown H''$ is adapted to $(\mathcal{F}_t)_{t\ge 0}$, which includes the two parts $(\mathcal{F'}_t)_{t\ge 0}$ and $(\mathcal{F''}_{t-\Delta(P)})_{t\ge\Delta(P)}$, since the two parts H', H'' are adapted respectively. By induction hypothesis, H is *càdlàg* on $[0, \Delta(P))$ and on $(\Delta(P), \Delta(P; Q))$, because the constituent fragments are, and at $\Delta(P)$, H is *càdlàg* by construction.
- Case $\langle ds = bdt + \sigma dW \& \boldsymbol{PB}\rangle \trianglerighteq [\!]_{i\in I}(\omega_i \cdot ch_i* \to Q_i)$: If the evolution of $\langle ds = bdt + \sigma dW \& \boldsymbol{PB}\rangle$ terminates before any communication occurs, this case is same as $\langle ds = bdt + \sigma dW \& B\rangle$. Otherwise, some communication is chosen to occur and the corresponding Q_i is executed. By induction hypothesis and the case for sequential composition, the results can be proved obviously. Here we omit the details.
- Case $P \| Q$: Suppose $(P_1 \parallel P_2, \rho_1 \uplus \rho_2) \to (\epsilon \parallel \epsilon, \rho_1' \uplus \rho_2', H_1 \uplus H_2)$. Because the processes P and Q don't share variables, by induction hypothesis, $H = H_1 \uplus H_2$ is *càdlàg* and adapted to the filtration generated by $\rho_1 \uplus \rho_2$, $(W_s)_{s\le t}$ and uniform processes during P and Q. $\Delta(P\|Q) = \max(\Delta(P), \Delta(Q))$ is a Markov time.

□

5. Assertions and Specifications

In this section, we define a specification logic for reasoning about SHCSP programs. We will first present the assertions including syntax and semantics, and then the specifications based on Hoare triples. The proof system will be given in next section.

5.1. Assertion Language

The assertion language is essentially defined by a first-order logic with emphasis on the notion of explicit time and the addition of several specific predicates on occurrence of communication traces and events. Before giving the syntax of assertions, we introduce three kinds of expression first.

$$\begin{array}{lcl} \boldsymbol{Ce} & ::= & h \mid \varepsilon \mid \langle ch, T\rangle \mid \boldsymbol{Ce} \cdot \boldsymbol{Ce} \mid \boldsymbol{Ce}^* \mid tr \\ \boldsymbol{Ve} & ::= & v \mid b \mid x \mid f^k(E_1, ..., E_k) \\ \boldsymbol{Te} & ::= & o \mid d \mid now \mid u^l(T_1, ..., T_l) \end{array}$$

Ce defines trace expressions, including a logical trace variable h, an empty trace ε, a communication pair $\langle ch, T\rangle$ representing that a communication along channel ch has occurred at time T, a concatenation of two traces $\boldsymbol{Ce}_1 \cdot \boldsymbol{Ce}_2$, and the system variable tr for representing the communication trace of the system. ***Ve*** defines value expressions, including a logical value variable v, a value constant b, a variable x, or an arithmetic function application. ***Te*** defines

time expressions, including a time logical variable o, a time constant d, the system variable now, or arithmetic time expressions.

The categories of the assertion language include terms, denoted by θ, θ_1 *etc.*, state formulas, denoted by S, S_1 *etc.*, probabilistic state formulas, denoted by PS, PS_1, and probabilistic formulas, denoted by PF, PF_1 *etc.*, which are given by the following syntax:

$$\begin{array}{lcl} \theta & ::= & Ve \mid Te \mid Ce \\ S & ::= & \bot \mid R^n(\theta_1, ..., \theta_n) \mid h.ch? \mid h.ch!G(\cdot) \mid \neg S \mid S_1 \vee S_2 \mid \forall v.S \\ PS & ::= & \bot \mid P(S) \bowtie p \mid \neg PS \mid PS \vee PS \mid \forall v.PS \mid B?PS \mid PS + PS \\ PF & ::= & \bot \mid PS \text{ at } T \mid \neg PF \mid PF \vee PF \mid \forall v.PF \mid \forall t.PF \end{array}$$

The terms θ include value, time and trace expressions. The state expressions S include false (denoted by $\bot$), truth-valued relation R^n on terms, readiness, and logical combinations of state formulas. Notice that the connectives $\wedge$, $\rightarrow, \exists$ can be derived from the existing ones. In particular, the readiness $h.ch?$ represents that the communication event over $ch?$ is enabled, and prior to it, the sequence of communications recorded in h has occurred; and $h.ch!G(\cdot)$ has similar meaning, except that the communication event over $ch!$ will send a value satisfying property $G(\cdot)$. For the property $G(\cdot)$, when it is instantiated with an arbitrary variable x, $G(x)$ is a first-order logical formula. Both the terms θ and S are defined for characterizing deterministic states in set $\mathcal{D}$.

The probabilistic state formulas are defined for characterizing the probabilistic states in $\Omega \rightarrow \mathcal{D}$. The intuitive explanation of the probabilistic state formulas are given below:

- $P(S) \bowtie p$, where $\bowtie \in \{<, =, >\}$ and $p \in [0, 1]$, asserts that S holds at the considered state with probability satisfying $\bowtie p$;
- The traditional logical combinations $\neg$, $\vee$, $\forall$ can be understood as usual;
- $B?PS$ performs conditional test. It holds for a state if it is a conditional version of some state satisfying PS;
- $PS + PS'$ performs addition. It holds for a state if it can be split into two parts satisfying PS and PS' respectively;

The probabilistic formulas PF add real time to the probabilistic state formulas. It include false, a primitive PS at T representing that PS holds at time T, and logical combinations of formulas (v, t represent logical variables for values and time resp.). For time primitive, we have the following two new axioms:

$$(PS_1 \text{ at } T \wedge PS_2 \text{ at } T) \Leftrightarrow (PS_1 \wedge PS_2) \text{ at } T$$
$$(PS_1 \text{ at } T \vee PS_2 \text{ at } T) \Leftrightarrow (PS_1 \vee PS_2) \text{ at } T$$

In the sequel, we use the logical abbreviation PS dr $[T_1, T_2]$ to represent that PS holds during the time interval $[T_1, T_2]$, with the following definition:

$$PS \text{ dr } [T_1, T_2] \stackrel{\text{def}}{=} \forall t.(T_1 \leq t \leq T_2) \Rightarrow PS \text{ at } t$$

Interpretation In the following, we will use a random variable $Z : \Omega \rightarrow (Var \rightarrow Val)$ to describe the current state and a stochastic process $\mathcal{H} : [0, +\infty) \times \Omega \rightarrow (Var \rightarrow Val)$ to represent the whole evolution. Moreover, let $LVar$ denote the set of all logical variables introduced in the formulas, we use an interpretation $L : \Omega \rightarrow (LVar \rightarrow Val)$ to record the assignment of the logical variables.

The semantics of terms θ, denoted by $[\![\theta]\!]^Z_L$, returns a value of type $\Omega \rightarrow Val$. We define the semantics for the three kinds of expressions respectively. The semantics of value expressions $[\![Ve]\!]^Z_L$ is defined as follows:

$$[\![v]\!]^Z_L = X \text{ where } X(\omega) = L(\omega)(v) \text{ for } \omega \in \Omega$$
$$[\![b]\!]^Z_L = b$$
$$[\![x]\!]^Z_L = Y \text{ where } Y(\omega) = Z(\omega)(x) \text{ for } \omega \in \Omega$$
$$[\![f^k(E_1, ..., E_k)]\!]^Z_L = f^k([\![E_1]\!]^Z_L, ..., [\![E_k]\!]^Z_L)$$

The semantics of time expressions $[\![Te]\!]_L^Z$ is defined as follows:

$$[\![o]\!]_L^Z = X \text{ where } X(\omega) = L(\omega)(o) \text{ for } \omega \in \Omega$$
$$[\![d]\!]_L^Z = d$$
$$[\![now]\!]_L^Z = Y \text{ where } Y(\omega) = Z(\omega)(now) \text{ for } \omega \in \Omega$$
$$[\![u^l(T_1, ..., T_l)]\!]_L^Z = u^l([\![T_1]\!]_L^Z, ..., [\![T_l]\!]_L^Z)$$

The semantics of trace expressions $[\![Ce]\!]_L^Z$ is defined as follows:

$$[\![h]\!]_L^Z = X \text{ where } X(\omega) = L(\omega)(h) \text{ for } \omega \in \Omega$$
$$[\![\varepsilon]\!]_L^Z = \varepsilon$$
$$[\![\langle ch, T\rangle]\!]_L^Z = \langle ch, [\![T]\!]_L^Z\rangle$$
$$[\![Ce_1 \cdot Ce_2]\!]_L^Z = [\![Ce_1]\!]_L^Z \cdot [\![Ce_2]\!]_L^Z$$
$$[\![Ce^*]\!]_L^Z = ([\![Ce]\!]_L^Z)^*$$
$$[\![tr]\!]_L^Z = Y \text{ where } Y(\omega) = Z(\omega)(tr) \text{ for } \omega \in \Omega$$

The semantics of state formulas S, denoted by $[\![S]\!]_L^Z$, returns a value of type $\Omega \to \{0, 1\}$. The definition is given as follows:

$$[\![\bot]\!]_L^Z = 0$$
$$[\![R^n(\theta_1, \ldots, \theta_n)]\!]_L^Z = R^n([\![\theta_1]\!]_L^Z, \ldots, [\![\theta_n]\!]_L^Z)$$
$$\text{where } R^n([\![\theta_1]\!]_L^Z, \ldots, [\![\theta_n]\!]_L^Z)(\omega) = R^n([\![\theta_1]\!]_L^Z(\omega), \ldots, [\![\theta_n]\!]_L^Z(\omega))$$
$$[\![h.ch?]\!]_L^Z = \mathcal{I}_{\{\omega \in \Omega | [\![h]\!]_L^Z(\omega).ch?|_{ch} \in Z(\omega)(rdy)|_{ch}\}}$$
$$[\![h.ch!G(\cdot)]\!]_L^Z = \exists c.G(c) \wedge \mathcal{I}_{\{\omega \in \Omega | [\![h]\!]_L^Z(\omega).ch!c|_{ch} \in Z(\omega)(rdy)|_{ch}\}}$$
$$[\![\neg S]\!]_L^Z = 1 - [\![S]\!]_L^Z$$
$$[\![S_1 \vee S_2]\!]_L^Z = [\![S_1]\!]_L^Z + [\![S_2]\!]_L^Z - [\![S_1]\!]_L^Z * [\![S_2]\!]_L^Z$$
$$[\![\forall v.S]\!]_L^Z = \forall c : \mathbb{R}.[\![S[c/v]]\!]_L^Z$$

In particular, $[\![h.ch?]\!]_L^Z(\omega) = 1$ denotes that, there exists a sequence in the rdy set of ω such that it has the same projection to ch as $h.ch?$, and $[\![h.ch!G(\cdot)]\!]_L^Z(\omega) = 1$ denotes that, there exists a sequence in the rdy set of ω, say $h'.ch!c$, such that it has the same projection to ch as $h.ch!G()$, and furthermore $G(c)$ holds, i.e. $G(\cdot)$ is the property of the value received from the sender. Notice that we use the projection to ch instead of the history trace h itself, because we only care about the history of the communications that have occurred over channel ch in the past.

The semantics of probabilistic state formulas *PS*, denoted by $[\![PS]\!]_L^Z$, returns a boolean value, defined as follows:

$$[\![P(S) \bowtie p]\!]_L^Z = (P([\![S]\!]_L^Z = 1) \bowtie p) = (P(\{\omega \in \Omega : [\![S]\!]_L^Z(\omega) = 1\}) \bowtie p)$$
$$[\![B?PS]\!]_L^Z = \exists Z_1.Z = B?Z_1 \wedge [\![PS]\!]_L^{Z_1}$$
$$[\![PS + PS']\!]_L^Z = \exists Z_1, Z_2.Z = Z_1 + Z_2 \wedge [\![PS]\!]_L^{Z_1} \wedge [\![PS']\!]_L^{Z_2}$$

where $Z = B?Z_1$ means that $Z(\omega) = Z_1(\omega)$ iff $[\![B]\!]_L^{Z_1}(\omega) = 1$. The connectives $\neg$, $\vee$ and $\forall$ can be defined as usual.

The semantics of formula *PF*, denoted by $[\![PF]\!]_L^{\mathcal{H},Z}$, is interpreted over a stochastic process, an initial random variable and the logical interpretation, and it returns a boolean value. The definition of the time primitive formula is given below:

$$[\![PS \text{ at } T]\!]_L^{\mathcal{H},Z} = [\![PS]\!]^{\mathcal{H}([\![T]\!]_L^Z)}$$

The others can be defined as usual.

We have proved that the terms and state formulas of the assertion language are measurable, stated by the following theorem:

Theorem 5.1 (Measurability). For any random variable Z and any stochastic process $\mathcal{H}$, the semantics of $[\![\theta]\!]_L^Z$ and $[\![S]\!]_L^Z$ are random variables (i.e. measurable).

Proof. We will prove this fact by induction on the structure of θ and S.

We list several cases for the proof of $[\![\theta]\!]_L^Z$ in the following, and the rest can be proved similarly.

- $[\![c]\!]_L^Z = c$ is a random variable trivially.
- $[\![x]\!]_L^Z = Y$ is a random variable, because $Y(\omega) = Z(\omega)(x)$ for each $\omega \in \Omega$ and Z is measurable.
- $[\![f^k(E_1, ..., E_k)]\!]_L^Z = f^k([\![E_1]\!]_L^Z, ..., [\![E_k]\!]_L^Z)$ is a random variable, because $[\![E_1]\!]_L^Z$, ..., $[\![E_k]\!]_L^Z$ are measurable and f^k is Borel-measurable. Thus, the composition $f^k([\![E_1]\!]_L^Z, ..., [\![E_k]\!]_L^Z)$ is measurable (the σ-algebras in the composition are compatible).
- $[\![h_1 \cdot h_2]\!]_L^Z = [\![h_1]\!]_L^Z \cdot [\![h_2]\!]_L^Z$ is a product. It is also measurable by induction hypothesis (measurable functions form an algebra).

We list several cases for the proof of $[\![S]\!]_L^Z$ in the following, and the rest can be proved similarly.

- $[\![h.ch!G(\cdot)]\!]_L^Z = \exists c.G(c) \wedge \mathcal{I}_{\{\omega \in \Omega | [\![h]\!]_L^Z(\omega).ch!c|_{ch} \in Z(\omega)(rdy)|_{ch}\}}$ is measurable, because it is defined by a quantified formula, which is a conjunction of first-order formula G and a characteristic function. Both of them are measurable.
- $[\![\neg S]\!]_L^Z = 1 - [\![S]\!]_L^Z$ is measurable, because $[\![S]\!]_L^Z$ is measurable by induction hypothesis.

$[\![R^n(\theta_1, \ldots, \theta_n)]\!]_L^Z$, $[\![h.ch!]\!]_L^Z$ and $[\![S_1 \vee S_2]\!]_L^Z$ can be proved similarly.

□

As a consequence of Theorem 5.1, the probabilistic state formulas *PS* are well defined, i.e. the probability with which some state formula *S* holds or not is definable. By Theorem 4.1, the flow of a process is adapted and $càdlàg$, thus the probabilistic formulas *PF* are also well defined.

5.2. Specifications

Based on the assertion language, the specification for an SHCSP process P is defined as a Hoare triple of the form $\{A; E\}\, P\, \{R; C\}$, where A, E, R, C are probability formulas. A and R are *precondition* and *postcondition*, which specify the initial state and the terminating state of P respectively. For both of them, the formulas *PF* occurring in them have the special form *PS* at now, and we will always write *PS* for short. E is called an *assumption* of P, which expresses the timed occurrence of the dual of communication events provided by the environment. C is called a *commitment* of P, which expresses the timed occurrence of communication events, and the real-time properties of P.

Definition 5.1 (Validity). We say a Hoare triple $\{A; E\}\, P\, \{R; C\}$ is *valid*, denoted by $\models \{A; E\}\, P\, \{R; C\}$, iff for any process Q, any initial states ρ_1 and ρ_2, if P terminates, i.e. $(P \| Q, \rho_1 \uplus \rho_2) \xrightarrow{\alpha^*} (\epsilon \| Q', \rho_1' \uplus \rho_2', \mathcal{H})$ then $[\![A]\!]^{\rho_1}$ and $[\![E]\!]^{\mathcal{H}, \rho_2}$ imply $[\![R]\!]^{\rho_1'}$ and $[\![C]\!]^{\mathcal{H}, \rho_1'}$, where $\mathcal{H}$ is the stochastic process of the evolution.

In this paper, we only consider partial correctness of a SHCSP process. As most stochastic hybrid systems are reactive, the termination of the systems is less important compared to safety. Some hybrid systems even do not terminate at all.

6. Proof System

Before giving the proof system, we introduce the notion of local and global invariants, for handing the SDEs and loop repetition respectively.

6.1. Local and Global Invariants

Definition 6.1 (Local Differential Invariant). The probabilistic formula Inv_{sde} is a local differential invariant of $\langle ds = bdt + \sigma dW \& PB \rangle$ with respect to the precondition *Init*, iff the following formulas hold:

- $Init \wedge PB \rightarrow Inv_{sde}$, representing that the initial states satisfy the invariant;

- $\mathit{Inv}_{sde} \rightarrow [\langle ds = bdt + \sigma dW \& \mathit{PB}\rangle]\mathit{Inv}_{sde}$, representing that starting from a state satisfying the invariant, the execution of the SDE preserves the invariant.

Definition 6.2 (Global Loop Invariant). The probabilistic formula Inv_{loop} is a global differential invariant of P^* with respect to the precondition *Init* and the environment assumption *Env*, iff the following formulas hold:

- $\mathit{Init} \rightarrow \mathit{Inv}_{loop}$, representing that the initial states satisfy the invariant;
- $\mathit{Inv}_{loop} \rightarrow [P, \mathit{Env}]\mathit{Inv}_{loop}$, representing that starting from a state satisfying the invariant, the execution of the loop body under the given environment *Env* preserves the invariant.

6.2. Axioms and Inference Rules of SHCSP

We present a proof system for reasoning about all valid Hoare triples for SHCSP processes. First we axiomatize SHCSP language by defining the axioms and inference rules for all the primitive and compound constructs, and then the general rules and axioms that are applicable to all processes.

Skip The rule for **skip** is very simple. Indicated by $\top$, the skip process as an internal action requires nothing from the environment; and it takes no time to complete, thus guarantees nothing to the history. The execution of **skip** produces a τ event. For simplicity, we abbreviate $h \cdot \langle \tau, now\rangle$ as $h + \tau$ in the rest of this paper.

$$\{A \wedge tr = h; \top\}\ \mathbf{skip}\ \{A[h/tr] \wedge tr = h + \tau; \top\}$$

Assignment The assignment $x := e$ changes nothing but assigns x to e and extends the trace by τ in the final state.

$$\{A \wedge x = v \wedge tr = h; \top\}\, x := e\, \{A[x, h/v, tr] \wedge x = e \wedge tr = h + \tau; \top\}$$

Input For input $ch?x$, we use logical variables o to denote the starting time, h the initial trace, and v the initial value of x respectively, in the precondition. The assumption indicates that no compatible output event is ready during $[o, o_1)$, and at time o_1, with probability p, a compatible output event with a value satisfying $F(\cdot)$ transmitted becomes ready. As a consequence of the assumption, during the whole interval $[o, o_1]$, the input event keeps waiting and ready, as indicated by the commitment. At time o_1, the communication occurs and terminates immediately. As indicated by the postcondition, with probability equal or greater than p, now is increased to o_1 (this is because, there may exist a compatible $ch!$ event occurring at o_1 while not satisfying $F(\cdot)$, but it can still make the communication over ch complete at o_1), and furthermore, with probability p, $F(x)$ holds and the trace is augmented by the new pair $\langle ch, o_1\rangle$. Assume o_1 is finite (and this assumption will be adopted for the rest of the paper). The rule is presented as follows:

$$\begin{aligned}&\{A \wedge now = o \wedge tr = h \wedge x = v; \neg h.ch!\ \mathsf{dr}\ [o, o_1) \wedge P(h.ch!F(\cdot)) = p\ \mathsf{at}\ o_1\} ch?x\\ &\quad \{A[o, h, v/now, tr, x] \wedge P(now = o_1) \geq p \wedge P(F(x) \wedge tr = h \cdot \langle ch, o_1\rangle) = p; h.ch?\ \mathsf{dr}\ [o, o_1]\}\end{aligned}$$

If such finite o_1 does not exist, i.e., the compatible output event will never become available. As a consequence, the input event will keep waiting forever, as shown by the following rule:

$$\{A \wedge now = o \wedge tr = h; \neg h.ch!\ \mathsf{dr}\ [o, \infty)\}\ ch?x\ \{\{A[o/now] \wedge now = \infty; h.ch?\ \mathsf{dr}\ [o, \infty)\}\}$$

Output Similarly, for output $ch!e$, we have one rule for the case when the compatible input event becomes ready in finite time. Thus the communication occurs successfully. Let $G(v) \stackrel{\text{def}}{=} (v = e)$ for all v.

$$\begin{aligned}&\{A \wedge now = o \wedge tr = h; \neg h.ch?\ \mathsf{dr}\ [o, o_1) \wedge P(h.ch?) = p\ \mathsf{at}\ o_1\} ch!e\\ &\quad \{A[o, h/now, tr] \wedge P(now = o_1) \geq p \wedge P(tr = h \cdot \langle ch, o_1\rangle) = p, h.ch!G(\cdot)\ \mathsf{dr}\ [o, o_1]\}\end{aligned}$$

We also have another rule for the case when the compatible input event will never become ready.

$$\{A \wedge now = o \wedge tr = h; (\neg h.ch?)\ \mathsf{dr}\ [o, \infty)\}\ ch!e\ \{\{A[o/now] \wedge now = \infty; h.ch!\ \mathsf{dr}\ [o, \infty)\}\}$$

Stochastic Continuous Evolution Based on the notion of stochastic differential invariant, we have the following rule for $\langle ds = bdt + \sigma dW \& \mathit{PB}\rangle$.

$$\frac{\mathit{Inv}_{sde} \text{ is a stochastic differential invariant of } \langle ds = bdt + \sigma dW \& \mathit{PB}\rangle \text{ with respect to } A}{\begin{aligned}&\{A \wedge s = s_0 \wedge now = o \wedge tr = h; \top\}\langle ds = bdt + \sigma dW \& \mathit{PB}\rangle\\ &\quad \{A[s_0, o, h/s, now, tr] \wedge tr = h + \tau \wedge \mathit{cl}(\neg \mathit{PB}) \wedge \mathit{Inv}_{sde}; (\mathit{PB} \wedge \mathit{Inv}_{sde})\ \mathsf{dr}\ [o, now)\}\end{aligned}}$$

where o, s_0 are logical variables denoting the starting time and the initial value of s resp., and $cl()$ returns the closure of a formula, e.g. $cl(P(x < 2) = p) \stackrel{\text{def}}{=} P(x \leq 2) = p$. The rule states that, throughout the whole evolution except for the escaping point, the domain ***PB*** and the invariant $\mathit{Inv}_{\mathit{sde}}$ hold almost surely, and furthermore, at the termination, the closure of $\neg$***PB*** holds almost surely.

Sequential Composition For $P; Q$, we use o to denote the starting time, and o_1 the termination time of P, if P terminates, which is also the starting time of Q. The first rule is for the case when P terminates.

$$\frac{\{A \wedge now = o; E\}\, P\, \{R_1 \wedge now = o_1; C_1\}\ \ \{R_1 \wedge now = o_1; E[o/now]\}\, Q\, \{R; C\}}{\{A; E\}\, P; Q\, \{R; C_1[o_1/now] \wedge C\}}$$

On the other hand, if P does not terminate, the effect of executing $P; Q$ is same to that of executing P itself.

$$\frac{\{A \wedge now = o; E\}\, P\, \{R \wedge now = \infty; C\}}{\{A \wedge now = o; E\}\, P; Q\, \{R \wedge now = \infty; C\}}$$

Conditional There are two cases depending on whether B holds or not. For both cases, the evaluation of B produces a τ event, as indicated by the following rule.

$$\frac{\{B?A[h/tr] \wedge tr = h + \tau; E\}\, P\, \{R; C\}}{\{A \wedge now = o \wedge tr = h; E\}\, B \to P\, \{R + (\neg B?A[h/tr] \wedge tr = h + \tau); C + (now = o \ \mathsf{at}\ now)\}}$$

Probabilistic Choice The rule for $P \sqcup_p Q$ is defined as follows:

$$\frac{\begin{array}{c}\{A; E\}\, P\, \{P(S) \bowtie_1 p_1; P(S') \bowtie_2 p_2 \ \mathsf{at}\ T\} \\ \{A; E\}\, Q\, \{P(S) \bowtie_1 q_1; P(S') \bowtie_2 q_2 \ \mathsf{at}\ T\}\end{array}}{\{A; E\}\ \ P \sqcup_p Q\ \ \{P(S) \bowtie_1 pp_1 + (1-p)q_1; P(S') \bowtie_2 pp_2 + (1-p)q_2 \ \mathsf{at}\ T\}}$$

where $\bowtie_1, \bowtie_2$ are two relational operators. The final postcondition indicates that, if after P terminates S holds with probability $\bowtie_1 p_1$, and after Q terminates S holds with probability $\bowtie_1 q_1$, then after $P \sqcup_p Q$ terminates, S holds with probability $\bowtie_1 pp_1 + (1-p)q_1$; The history formula can be understood similarly.

Communication Interrupt We use a logical variable o_F to denote the execution time of the SDE. The premise of the first rule indicates that the compatible events of $\{ch_i*\}_{i \in I}$ are not ready after the continuous terminates. For this case, the effect of executing the whole process is thus equivalent to that of executing the SDE.

$$\frac{\begin{array}{c}\{A \wedge now = o; E\}\langle ds = bdt + \sigma dW \,\&\, \mathbf{PB}\rangle\{R \wedge now = o + o_F; C\} \\ \forall i \in I. A \wedge now = o \wedge tr = h \wedge E \Rightarrow (\neg h.\overline{ch_i*} \ \mathsf{dr}\ [o, o + o_F])\end{array}}{\{A \wedge now = o \wedge tr = h; E\}\ \langle ds = bdt + \sigma dW \,\&\, \mathbf{PB}\rangle \trianglerighteq [\!]_{i \in I}(\omega_i \cdot ch_i* \to Q_i)\ \{R \wedge now = o + o_F; C\}}$$

By contrast, when some compatible events become ready before the continuous evolution terminates, the continuous evolution will be interrupted by one of the communications with the corresponding probability indicated by the weight of it.

$$\frac{\begin{array}{c}\{A \wedge now = o; E\}\langle ds = bdt + \sigma dW \,\&\, \mathbf{PB}\rangle\{R \wedge now = o + o_F; C\} \\ (A \wedge now = o \wedge tr = h \wedge E) \Rightarrow (\wedge_{1 \leq j \leq n} h.\overline{ch_{i_j}*} \ \mathsf{at}\ (o + o_1) \wedge o_1 \leq o_F) \\ \forall 1 \leq j \leq n. \{now = o \wedge tr = h; E\}\, ch_{i_j}*; Q_{i_j}\ \{P(S) \bowtie_1 p_{i_j}; P(S') \bowtie_2 q_{i_j} \ \mathsf{at}\ T\}\end{array}}{\begin{array}{l}\{A \wedge now = o \wedge s = s_0 \wedge tr = h; E\}\ \langle ds = bdt + \sigma dW \,\&\, \mathbf{PB}\rangle \trianglerighteq [\!]_{i \in I}(\omega_i \cdot ch_i* \to Q_i) \\ \{P(S) \bowtie_1 \bigoplus_{1 \leq j \leq n} \frac{\omega_{i_j}}{\sum_{j=1}^{n} \omega_{i_j}} \cdot p_{i_j}; (\mathbf{PB} \wedge \mathit{Inv}_{\mathit{sde}}) \ \mathsf{dr}\ [o, o + o_1) \wedge \mathbf{AfterC}\}\end{array}}$$

where ***AfterC*** is defined as $P(S') \bowtie_2 \sum_{1 \leq j \leq n} (\frac{\omega_{i_j}}{\sum_{j=1}^{n} \omega_{i_j}} \cdot q_{i_j})$ $\mathsf{at}\ T$ if $T \geq o + o_1$, and $(\bigoplus_{1 \leq j \leq n} P(S') \bowtie_2 q_{i_j})$ $\mathsf{at}\ T$ if $T < o + o_1$. $\bigoplus_{1 \leq k \leq n} F_k$ is defined as F_1 if n is 1, otherwise $\bigoplus_{1 \leq k \leq (n-1)} F_k + F_n$ if n is greater than 1. In the first o_1 time, the domain ***PB*** and the differential invariant $\mathit{Inv}_{\mathit{sde}}$ of the SDE hold.

Parallel Composition For $P \| Q$, let X be $X_1 \cap X_2$ where $X_1 = \Sigma(P)$ and $X_2 = \Sigma(Q)$. For each parallel process, we assume it starts from time 0 and an empty trace. Suppose A defines the initial values of state variables of P and Q,

then

$$\frac{\begin{array}{c}A \Rightarrow A_1 \wedge A_2, \quad \{A_1 \wedge now = 0 \wedge tr = \varepsilon; E_1\}\, P\, \{R_1 \wedge now = o_1; C_1\} \\ \{A_2 \wedge now = 0 \wedge tr = \varepsilon; E_2\}\, Q\, \{R_2 \wedge now = o_2; C_2\} \\ \forall ch \in X.(C_1[o_1/now] \restriction_{ch} \Rightarrow E_2 \restriction_{ch}) \wedge (C_2[o_2/now] \restriction_{ch} \Rightarrow E_1 \restriction_{ch}) \\ \forall dh \in X_1 \setminus X.E \restriction_{dh} \Rightarrow E_1 \restriction_{dh} \quad \forall dh' \in X_2 \setminus X.E \restriction_{dh'} \Rightarrow E_2 \restriction_{dh'}\end{array}}{\{A \wedge now = 0 \wedge tr = \varepsilon; E\}\, P \| Q\, \{R; C_1' \wedge C_2'\}}$$

$$R \stackrel{\text{def}}{=} R_1[\gamma_1/tr, o_1/now] \wedge R_2[\gamma_2/tr, o_2/now] \wedge now = o_m \wedge \gamma_1 \restriction_X = \gamma_2 \restriction_X \wedge tr = (\gamma_1 \underset{C}{\|} \gamma_2)$$

$$C_i' \stackrel{\text{def}}{=} C_i[o_i/now] \wedge R_i'[o_i/now]\ \mathsf{dr}\ [o_i, now) \text{ for } i = 1, 2$$

where A_1 is a probabilistic formula of P (i.e., it only contains state variables of P), A_2 a probabilistic formula of Q, o_1 and o_2, γ_1 and γ_2 logical variables representing the termination time and the accumulated traces of P and Q respectively, o_m the $\max\{o_1, o_2\}$, for $i = 1, 2$, $R_i \Rightarrow R_i'$ but $tr \notin R_i'$.

The compatibility check is performed between P and Q: the environmental assumption of P (resp. Q), i.e. E_1 (resp. E_2), is guaranteed by the commitment of Q (resp. P) and the overall environment E of $P \| Q$. At termination of $P \| Q$, the time will be the maximum of the termination time of P and Q, thus it is the maximum of o_1 and o_2; and the trace is the alphabetized parallel of the traces after P and Q, which is $(\gamma_1 \underset{C}{\|} \gamma_2)$. In C_1' and C_2', we specify that none of variables of P and Q except for now and tr will change after their termination.

Repetition For P^*, let Inv_{loop} be the global loop invariant of P, then

$$\frac{A \Rightarrow Inv_{loop} \quad \{Inv_{loop} \wedge now = o_l; E[o/now]\}\ P\ \{Inv_{loop}; Inv_{loop}\ \mathsf{dr}\ [o_l, now)\}}{\{A \wedge now = o; E\}\ P^*\ \{A[\mathbf{v}_o/\mathbf{v}] \wedge Inv_{loop}; Inv_{loop}\ \mathsf{dr}\ [o, now)\}}$$

where the logical variable o denotes the starting time of P^*, o_l the starting time of each iteration of P, and the variable list $\mathbf{v}$ represents the set of modified variables in $Var^+(P)$ and $\mathbf{v}_o$ their initial values.

The general rules that are applicable to all processes, such as Monotonicity, Case Analysis, and so on, are similar to the classical Hoare Logic.

An Example

We present a SHCSP example below and show how to apply the inference system to specify and verify its property. First, given a time $T \in \mathbb{R}^+$, define wait T to be an abbreviation for $t := 0; \langle \dot{t} = 1 \& t < T \rangle$, i.e. time makes progress for T time units. We then define the processes P_1, P_2 and P_3 as follows:

$$\begin{array}{lcl} P_1 & \stackrel{\text{def}}{=} & \langle ds = b(s,a)dt + \sigma(s,a)dW \rangle \trianglerighteq [\!]_{i \in \{1,2\}} (\omega_i \cdot ch_i?a \rightarrow \textbf{skip}) \\ P_2 & \stackrel{\text{def}}{=} & \text{wait } T_1; ch_1!1 \\ P_3 & \stackrel{\text{def}}{=} & \text{wait } T_2; ch_2!(-1) \end{array}$$

where s is a continuous variable, a is a discrete variable, $\omega_1 = 2, \omega_2 = 3$, $T_1, T_2 \in \mathbb{R}^+$, and ch_1, ch_2 are channels.

Consider the parallel process $P_1 \| P_2 \| P_3$. The SDE in P_1 receives a value from P_2 and P_3 with different weights 2 and 3 respectively, and then assigns it to the control variable a, if both P_2 and P_3 are available to communicate with P_1. Otherwise, P_1 receives a value from P_2 or P_3 depending on which is ready first. Below we denote the precondition, assumption, postcondition, and commitment of process S in $\{P_1, P_2, P_3, P_1 \| P_2 \| P_3\}$ by $S.Pre, S.Env, S.Post$ and $S.Comt$ respectively. First of all, assume we have the following preconditions and environment assumptions for the processes:

$$\begin{array}{lcl} (P_1 \| P_2 \| P_3).Pre & \stackrel{\text{def}}{=} & s = s_0 \wedge now = 0 \wedge tr = \varepsilon, \text{ for } i = 1, 2, 3 \\ P_1.Env & \stackrel{\text{def}}{=} & \neg ch_1!\ \mathsf{dr}\ [0, T_1) \wedge ch_1!1\ \mathsf{at}\ T_1 \wedge \neg ch_2!\ \mathsf{dr}\ [0, T_2) \wedge ch_2!(-1)\ \mathsf{at}\ T_2 \\ P_j.Env & \stackrel{\text{def}}{=} & P(ch_{j-1}?) = \frac{\omega_{j-1}}{5}\ \mathsf{at}\ T_{j-1}, \text{ for } j = 2, 3 \end{array}$$

The precondition indicates that, at the beginning of the execution of $P_1 \| P_2 \| P_3$, s is initialized as s_0, now as 0, and the trace tr as empty. The environmental assumption of P_1 indicates that the partner communication event $ch_1!$ is not ready until T_1, and $ch_2!$ is not ready until T_2. The environmental assumption of P_2 indicates that the partner communication

event $ch_1?$ is ready at time T_1 with probability $\frac{2}{5}$. The environmental assumption of P_3 can be explained similarly. Secondly, suppose for the SDE, we have the following specification:

$$\{P_1.\mathit{Pre}; P_1.\mathit{Env}\}\ \langle ds = b(s,a)dt + \sigma(s,a)dW\rangle\ \{R_F \wedge now = o_F; C_F\}$$

and meanwhile, we assume $o_F > \max(T_1, T_2)$. As a result, according to the semantics of the communication interrupt, the SDE will be interrupted by the communication along channel ch_1 or channel ch_2, depending on the values of T_1 and T_2. In the following deduction, we will omit the details of the specification of the SDE, and focus on mainly the communication interaction between the processes and especially the effect of the probabilistic aspects brought by the communication interrupt.

By applying the rules for output and sequential composition, we can obtain the postconditions and commitments for P_2 and P_3 as follows:

$$\begin{array}{lcl} P_2.\mathit{Post} & = & P(now = T_1) \geq \frac{2}{5} \wedge P(tr = \langle ch_1, T_1\rangle) = \frac{2}{5} \\ P_2.\mathit{Comt} & = & ch_1!1\ \mathsf{at}\ T_1 \\ P_3.\mathit{Post} & = & P(now = T_2) \geq \frac{3}{5} \wedge P(tr = \langle ch_2, T_2\rangle) = \frac{3}{5} \\ P_3.\mathit{Comt} & = & ch_2!(-1)\ \mathsf{at}\ T_2 \end{array}$$

By applying the rule for input, we prove the following specifications:

$$\{now = 0 \wedge tr = \varepsilon; P_1.\mathit{Env}\}\ ch_1?a; \mathbf{skip}\ \{now = T_1 \wedge a = 1 \wedge tr = \langle ch_1, T_1\rangle; ch_1?\ \mathsf{dr}\ [0, T_1]\}$$
$$\{now = 0 \wedge tr = \varepsilon; P_1.\mathit{Env}\}\ ch_2?a; \mathbf{skip}\ \{now = T_2 \wedge a = -1 \wedge tr = \langle ch_2, T_2\rangle; ch_2?\ \mathsf{dr}\ [0, T_2]\}$$

For further use, denote the above two postconditions by R_1 and R_2 respectively. If $T_1 = T_2$ holds, then P_1 will communicate with both P_2 and P_3 with probabilities $\frac{2}{5}$ and $\frac{3}{5}$ respectively. By applying the second rule for communication interrupt, we have the following results:

$$\begin{array}{lcl} P_1.\mathit{Post} & = & now = T_1 \wedge P(a = 1) = \frac{2}{5} \wedge P(a = -1) = \frac{3}{5} \\ & & \wedge P(tr = \langle ch_1, T_1\rangle) = \frac{2}{5} \wedge P(tr = \langle ch_1, T_1\rangle) = \frac{3}{5} \\ P_1.\mathit{Comt} & = & ch_1?\ \mathsf{dr}\ [0, T_1) \wedge P(ch_1?) = \frac{2}{5}\ \mathsf{at}\ T_1 \wedge ch_2?\ \mathsf{dr}\ [0, T_2) \wedge P(ch_2?) = \frac{3}{5}\ \mathsf{at}\ T_2 \end{array}$$

Finally, consider the parallel composition $P_1 \| P_2 \| P_3$. We check the compatibility between the processes and prove the following facts:

$$P_1.\mathit{Comt}\restriction_{ch_1} \Rightarrow P_2.\mathit{Env}\restriction_{ch_1}, P_1.\mathit{Comt}\restriction_{ch_2} \Rightarrow P_3.\mathit{Env}\restriction_{ch_2}, P_2.\mathit{Comt}\restriction_{ch_1} \Rightarrow P_1.\mathit{Env}\restriction_{ch_1}, P_3.\mathit{Comt}\restriction_{ch_2} \Rightarrow P_1.\mathit{Env}\restriction_{ch_2}$$

By applying the rule for parallel composition, we have

$$\begin{array}{l} \{P_1.\mathit{Pre}; \top\} P_1 \| P_2 \| P_3 \{now = T_1 \wedge P(a = 1) = \frac{2}{5} \wedge P(a = -1) = \frac{3}{5} \\ \qquad \wedge P(\gamma_1 = \langle ch_1, T_1\rangle) = \frac{2}{5} \wedge P(\gamma_1 = \langle ch_1, T_1\rangle) = \frac{3}{5}; P_1.\mathit{Comt} \wedge P_2.\mathit{Comt} \wedge P_3.\mathit{Comt}\} \end{array}$$

which indicates that, after $P_1 \| P_2 \| P_3$ terminates, the execution time is T_1 (which is also T_2), and for process P_1, the control variable a and the trace are 1 and $\langle ch_1, T_1\rangle$ with probability $\frac{2}{5}$, and are -1 and $\langle ch_2, T_2\rangle$ with probability $\frac{3}{5}$ respectively.

For this example, we can also consider the cases when T_1 and T_2 are not equal, for each of which, P_1 will only communicate with one of them respectively.

6.3. Properties

Definition 6.3 (Theorem). If $\{A; E\}\ P\ \{R; C\}$ is derivable from the inference system of SHCSP defined above, then it is called a *theorem*, denoted by $\vdash \{A; E\}\ P\ \{R; C\}$.

We have proved the following theorem for the soundness of the inference system.

Theorem 6.1 (Soundness). If $\vdash \{A; E\}\ P\ \{R; C\}$, then $\models \{A; E\}\ P\ \{R; C\}$, i.e. every theorem of the proof system is valid.

Proof. To prove soundness, we need to show that the axioms are valid, and that every inference rule in the proof system preserves validity. That is, if every premise of the rule is valid, then the conclusion is also valid. We will prove the soundness theorem by induction on the structure of Stochastic HCSP processes P. In the following proof, we always assume P executes in parallel with its environment PE, and $(P \| PE, \rho_1 \uplus \rho_2) \xrightarrow{\alpha^*} (\epsilon \| PE', \rho_1' \uplus \rho_2', \mathcal{H}), \mathcal{H}$

is the stochastic process of the evolution. Then what we need to prove is that, $[\![A]\!]_L^{\rho_1}$ and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$ imply $[\![R]\!]_L^{\rho_1'}$ and $[\![C]\!]_L^{\rho_1',\mathcal{H}}$.

Assume $T_0 = \rho_1(now)$ for simplicity. Notice that T_0 introduced here is a random variable of type $\Omega \to \mathbb{R}$. Throughout the proof, we will introduce some random variables for assistance, and when there is no confusion, we will not explain them specially.

- Case **skip**: $\rho_1' = \rho_1[tr+\tau]$. We need to prove $[\![A \wedge tr = h]\!]_L^{\rho_1} \Rightarrow [\![A[h/tr] \wedge tr = h+\tau]\!]_L^{\rho_1'}$, which holds obviously.
- Case Assignment $x := e$: From the operational semantics, we have $\rho_1' = \rho_1[x \mapsto e, tr \mapsto tr \cdot \langle \tau, now \rangle)]$. Assume $[\![A \wedge tr = h \wedge x = v]\!]_L^{\rho_1}$, we need to prove $[\![A[v, h/x, tr] \wedge x = e \wedge tr = h + \tau]\!]_L^{\rho_1'}$. Obviously this holds.

- Case Input $ch?x$: From the operational semantics, we have $\rho_1' = \rho_1[now \mapsto T_0 + d, x \mapsto b, tr \mapsto tr \cdot \langle ch, T_0 + d \rangle]$ for some random variables d and b; and for any $\omega \in \Omega$ and any $t \in [T_0(\omega), T_0(\omega) + d(\omega))$, $\rho_1(\omega)(tr).ch! \restriction_{ch} \notin \mathcal{H}(t,\omega)(rdy) \restriction_{ch}$, and $\rho_1(\omega)(tr).ch!b(\omega) \restriction_{ch} \in \mathcal{H}(T_0(\omega) + d(\omega), \omega)(rdy) \restriction_{ch}$; and for any $t \in [T_0(\omega), T_0(\omega) + d(\omega)]$, $\rho_1(\omega)(tr).ch? \in \mathcal{H}(t,\omega)(rdy)$. Assume $[\![A \wedge now = o \wedge tr = h \wedge x = v]\!]_L^{\rho_1}$ and $[\![\neg h.ch! \ \mathsf{dr}\ [o, o_1) \wedge P(h.ch!F(\cdot)) = p \ \mathsf{at}\ o_1]\!]_L^{\rho_2,\mathcal{H}}$, we need to prove that $[\![A[o, h, v/now, tr, x] \wedge P(now = o_1) \geq p \wedge P(F(x) \wedge tr = h \cdot \langle ch, o_1 \rangle) = p]\!]_L^{\rho_1'}$ and $[\![h.ch? \ \mathsf{dr}\ [o, o_1)]\!]_L^{\rho_1',\mathcal{H}}$.

  First from $[\![A \wedge tr = h \wedge now = o \wedge x = v]\!]_L^{\rho_1}$, we have $[\![A[o, h, v/now, tr, x]]\!]_L^{\rho_1}$. Compare ρ_1' with ρ_1, we can find that only variables tr, now, and x are changed. We obtain $[\![A[o, h, v/now, tr, x]]\!]_L^{\rho_1'}$.

  From the assumption $\rho_2, \mathcal{H} \models \neg h.ch! \ \mathsf{dr}\ [o, o_1) \wedge P(h.ch!F(\cdot)) = p \ \mathsf{at}\ o_1$, we can get the facts that

  $$P(\{\omega : \forall t \in [L(\omega)(o), L(\omega)(o_1)).L(\omega)(h).ch! \restriction_{ch} \notin \mathcal{H}(t,\omega)(rdy) \restriction_{ch}\}) = 1$$
  $$P(\{\omega : \exists c.F(c) \wedge L(\omega)(h).ch!c \restriction_{ch} \in \mathcal{H}(L(\omega)(o_1), \omega)(rdy) \restriction_{ch}\}) = p$$

  For simplicity, we denote the second sample set above as Ω_1. From $\rho_1 \models tr = h \wedge now = o$, then for all ω, $\rho_1(\omega)(tr) = L(\omega)(h)$ and $\rho_1(\omega)(now) = L(\omega)(o) = T_0(\omega)$. Moreover, we have that, for any $\omega \in \Omega'$, $[\![now]\!]_L^{\rho_1'}(\omega) = T_0(\omega) + d(\omega) = L(\omega)(o_1)$. Thus, $P(\{\omega : [\![now = o_1]\!]_L^{\rho_1'}(\omega)\}) \geq p$, $P(now = o_1) \geq p$ is proved. For any $\omega \in \Omega'$, we have $[\![F(x) \wedge tr = h \cdot \langle ch, o_1 \rangle]\!]_L^{\rho_1'}(\omega)$. Thus, $P(F(x) \wedge tr = h \cdot \langle ch, o_1 \rangle) \geq p$ is proved. On the other hand, for any ω satisfying $[\![F(x) \wedge tr = h \cdot \langle ch, o_1 \rangle]\!]_L^{\rho_1'}(\omega)$, we have $\omega \in \Omega'$ by letting c be $b(\omega)$, thus $P(F(x) \wedge tr = h \cdot \langle ch, o_1 \rangle) = p$ is proved.

  Finally, from the assumptions, o_1 is less or equal than $T_0 + d$ almost surely. As a result, $[\![h.ch? \ \mathsf{dr}\ [o, o_1)]\!]_L^{\rho_1',\mathcal{H}}$ is proved.
- Case Output $ch!e$: The fact can be proved similarly to $ch?x$.
- Case Continuous $\langle ds = bdt + \sigma dW \& PB \rangle$: From the operational semantics, there must exist a random variable $d : \Omega \to \mathbb{R}$ such that $\rho_1' = \rho_1[now \mapsto now + d, s \mapsto X_d][tr \mapsto tr \cdot \langle \tau, now \rangle]$ and $\mathcal{H} = H_d^{\rho_1,s,X}$, where $X : [0,d) \times \Omega \to \mathbb{R}^{d(s)}$ is the solution of $ds = bdt + \sigma dW$ and $\forall t \in [0,d)$, $[\![PB]\!]_L^{\rho_1[now \mapsto now+t, s \mapsto X_t]} = \mathbf{True}$, and $[\![cl(\neg PB)]\!]_L^{\rho_1'} = \mathbf{False}$. Assume $[\![A \wedge s = s_0 \wedge now = o \wedge tr = h]\!]_L^{\rho_1}$, we need to prove $[\![A[s_0, o, h/s, now, tr] \wedge tr = h + \tau \wedge cl(\neg PB) \wedge Inv_{sde}]\!]_L^{\rho_1'}$ and $[\![(PB \wedge Inv_{sde}) \ \mathsf{dr}\ [o, now)]\!]_L^{\rho_1',\mathcal{H}}$. From $[\![A \wedge s = s_0 \wedge now = o \wedge tr = h]\!]_L^{\rho_1}$, we have $[\![A[s_0, o, h/s, now, tr]]\!]_L^{\rho_1}$. Since only now, s, tr are changed in ρ_1', thus $[\![A[s_0, o, h/s, now, tr]]\!]_L^{\rho_1'}$. $[\![tr = h + \tau \wedge cl(\neg PB)]\!]_L^{\rho_1'}$ and $[\![PB \ \mathsf{dr}\ [o, now)]\!]_L^{\rho_1',\mathcal{H}}$ hold obviously. From the definition of local differential invariant Inv_{sde} in Def. 6.1, $[\![Inv_{sde}]\!]_L^{\rho_1'}$ and $[\![Inv_{sde} \ \mathsf{dr}\ [o, now)]\!]_L^{\rho_1',\mathcal{H}}$ can be proved directly.
- Case Sequential Composition $P; Q$: We assume the intermediate state at termination of P is ρ_1'' (thus Q will start from ρ_1''), and the behaviors of P and Q are $\mathcal{H}_1$ and $\mathcal{H}_2$ respectively, whose concatenation is exactly $\mathcal{H}$. Assume we have $[\![A \wedge now = o]\!]_L^{\rho_1}$ and $[\![E]\!]_L^{\rho_2,H}$, we need to prove that $[\![R]\!]_L^{\rho_1'}$ and $[\![C_1[o_1/now] \wedge C]\!]_L^{\rho_1',H}$, where $\{A \wedge now = o; E\}\, P\, \{R_1 \wedge now = o_1; C_1\}$ and $\{R_1 \wedge now = o_1; E[o/now]\}\, Q\, \{R; C\}$ as in the rule for sequential composition.

  According to the inference rules, from $\{A \wedge now = o; E\}\, P\, \{R_1 \wedge now = o_1 \wedge tr = h_1; C_1\}$, we can get $\{A \wedge now = o; E \restriction_{\leq o_1}\}\, P\, \{R_1 \wedge now = o_1; C_1\}$, where $E \restriction_{\leq o_1}$ only addresses the behavior of environment before or equal time o_1. Then the proof is given as follows: First, from $[\![E]\!]_L^{\rho_2,H}$, we have $[\![E \restriction_{\leq o_1}]\!]_L^{\rho_2,H_1} \rho_1, \mathcal{H}_1 \models E \restriction_{\leq o_1}$, then by induction hypothesis, for P, we have $[\![R_1 \wedge now = o_1]\!]_L^{\rho_1''}$ and $[\![C_1]\!]_L^{\rho_1'',\mathcal{H}_1}$. Similarly, by induction hypothesis

again for Q, we have $[\![R]\!]_L^{\rho_1'}$ and $[\![C]\!]_L^{\rho_1',\mathcal{H}_2}$, then $[\![C]\!]_L^{\rho_1',\mathcal{H}}$. From $[\![C_1]\!]_L^{\rho_1'',\mathcal{H}_1}$, we have $[\![C_1[o_1/now]]\!]_L^{\rho_1',\mathcal{H}}$. The result is proved finally.

- Case Conditional $B \to P$: Starting from $B?\rho_1[tr+\tau]$, under the same environment, assume the terminating state of P is ρ_P and the flow of P is $\mathcal{H}_P$ respectively. Then according to the operational semantics, we have $\rho_1' = \rho_P + \neg B?\rho_1[tr+\tau]$, and moreover $\mathcal{H} = \mathcal{H}_P + \neg B?\{\rho_1(now) \mapsto \rho_1[tr+\tau]\}$. Assume $[\![A \wedge now = o \wedge tr = h]\!]_L^{\rho_1}$ and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, then we need to prove $[\![R + \neg B?A]\!]_L^{\rho_1'}$ and $[\![C + (now = o \text{ at } now)]\!]_L^{\rho_1',\mathcal{H}}$.

  From $[\![A \wedge tr = h]\!]_L^{\rho_1}$, we can get $[\![B?A[h/tr] \wedge tr = h + \tau]\!]_L^{B?\rho_1[tr+\tau]}$. By induction hypothesis, we obtain the following results: $[\![R]\!]_L^{\rho_P}$ and $[\![C]\!]_L^{\rho_P,\mathcal{H}_P}$. On the other hand, from the definitions of probabilistic states and assertions, we can prove $[\![\neg B?A[h/tr] \wedge tr = h+\tau]\!]_L^{\neg B?\rho_1[tr+\tau]}$ and $[\![now = o \text{ at } now]\!]_L^{\neg B?\rho_1[tr+\tau],\neg B?\{\rho_1(now)\mapsto\rho_1[tr+\tau]\}}$. The final result is proved by combining the above facts.
- Case Probabilistic Choice $P \sqcup_p Q$: Starting from ρ_1, under the same environment, assume the terminating states of P and Q are ρ_P and ρ_Q, and the flow of P and Q are $\mathcal{H}_P$ and $\mathcal{H}_Q$, respectively. Then from the operational semantics, we have $\rho_1' = \mathcal{I}_{U\le p}?\rho_P + \mathcal{I}_{U>p}?\rho_Q$, and moreover $\mathcal{H} = \mathcal{I}_{U\le p}?\mathcal{H}_P + \mathcal{I}_{U>p}?\mathcal{H}_Q$. Assume $[\![A]\!]_L^{\rho_1}$ and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, then we need to prove $[\![P(S) \bowtie_1 pp_1 + (1-p)q_1]\!]_L^{\rho_1'}$ and $[\![P(S') \bowtie_2 pp_2 + (1-p)q_2 \text{ at } T]\!]_L^{\rho_1',\mathcal{H}}$. Below for simplicity, denote the postconditions and commitments of P and Q by R_i and C_i, where $i = 1, 2$, respectively.

  Consider the sample subset $\Omega_P = \{\omega : U(\omega) \le p\}$. For all $\omega \in \Omega_P$, the execution of $P \sqcup_p Q$ turns into the execution of P, and it will result in the state $\mathcal{I}_{U\le p}?\rho_P$ and flow $\mathcal{I}_{U\le p}?\mathcal{H}_P$. From $\{A; E\}\ P\ \{R_1; C_1\}$, we can prove easily $\{\mathcal{I}_{U\le p}?A; \mathcal{I}_{U\le p}?E\}\ P\ \{\mathcal{I}_{U\le p}?R_1; \mathcal{I}_{U\le p}?C_1\}$. From the assumption $[\![A]\!]_L^{\rho_1}$ and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, we can get the facts that $[\![\mathcal{I}_{U\le p}?A]\!]_L^{\mathcal{I}_{U\le p}?\rho_1}$ and $[\![\mathcal{I}_{U\le p}?E]\!]_L^{\mathcal{I}_{U\le p}?\rho_2,\mathcal{I}_{U\le p}?\mathcal{H}}$, the second is exactly $[\![\mathcal{I}_{U\le p}?E]\!]_L^{\mathcal{I}_{U\le p}?\rho_2,\mathcal{I}_{U\le p}?\mathcal{H}_P}$. By induction hypothesis, we obtain the facts $[\![\mathcal{I}_{U\le p}?R_1]\!]_L^{\mathcal{I}_{U\le p}?\rho_P}$ and $[\![\mathcal{I}_{U\le p}?C_1]\!]_L^{\mathcal{I}_{U\le p}?\rho_P,\mathcal{I}_{U\le p}?\mathcal{H}_P}$. Similarly, we can prove for Q that $[\![\mathcal{I}_{U>p}?R_2]\!]_L^{\mathcal{I}_{U>p}?\rho_Q}$ and $[\![\mathcal{I}_{U>p}?C_2]\!]_L^{\mathcal{I}_{U>p}?\rho_Q,\mathcal{I}_{U>p}?\mathcal{H}_Q}$. Combining the two results, we have $[\![\mathcal{I}_{U\le p}?R_1 + \mathcal{I}_{U>p}?R_2]\!]_L^{\rho_1'}$ and $[\![\mathcal{I}_{U\le p}?C_1 + \mathcal{I}_{U>p}?C_2]\!]_L^{\rho_1',\mathcal{H}}$. From the fact that the uniformly distributed random variable U is independent from all the other random variables, thus $[\![P(S) \bowtie_1 pp_1 + (1-p)q_1]\!]_L^{\rho_1'}$ and $[\![P(S') \bowtie_2 pp_2 + (1-p)q_2 \text{ at } T]\!]_L^{\rho_1',\mathcal{H}}$ are easily proved.
- Case Communication Interrupt: Assume $[\![A \wedge now = o]\!]_L^{\rho_1}$, and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$. For the first rule, from the hypothesis of the second line, the continuous evolution terminates without any communication occurring. Thus, we have the fact that ρ_1' and $\mathcal{H}$ are also the terminating state and the flow of $\langle ds = bdt + \sigma dW \& B\rangle$ (except that the ready set of ρ is larger by including the ch_i*s). By induction hypothesis, we obtain $[\![R \wedge o + o_F]\!]_L^{\rho_1'}$ and $[\![C]\!]_L^{\rho_1',\mathcal{H}}$ easily.

  For the second case, the SDE is executing from the beginning, and meanwhile, all the $\{ch_i*\}$s become ready and start to wait. At some time, here $o + o_1$, some communication occurs with the probability decided by w_is. We record the state and the flow till time $o + o_1$ by ρ_m and $\mathcal{H}_m$ respectively. From the semantics of the SDE, we have $[\![(PB \wedge Inv_{sde}) \text{ dr } [o, o + o_1)]\!]_L^{\rho_m,\mathcal{H}_m}$, thus $[\![(PB \wedge Inv_{sde}) \text{ dr } [o, o + o_1)]\!]_L^{\rho_1',\mathcal{H}_m}$. On the other hand, for the ready events $\{ch_{i_j}*\}_{1\le j\le n}$, during the waiting time, $A[o, s_0/now, s] \wedge PB$ always holds. Thus, similar to the probabilistic choice, we can prove the final facts.
- Case Parallel Composition $P \| Q$: From the operational semantics, there must exist ρ_{11} and ρ_{11}' , ρ_{12} and ρ_{12}' for initial states and terminating states of P and Q respectively, which satisfy: $\rho_1 = \rho_{11} \uplus \rho_{12}$ and $\rho_1' = \rho_{11}' \uplus \rho_{12}'$; $\rho_{11}'(\cdot)(tr) \restriction_X = \rho_{12}' \restriction_X$ (assuming P and Q terminate at the same time here, which will be generalized in the following proof). Assume we have $[\![A \wedge now = 0 \wedge tr = \varepsilon]\!]_L^{\rho_1}$, and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, we need to prove $[\![R]\!]_L^{\rho_1'}$ and $[\![C_1' \wedge C_2']\!]_L^{\rho_1',\mathcal{H}}$. The proof is given by the following steps.

  First of all, we prove that $[\![C_1]\!]_L^{\rho_{11}',\mathcal{H}}$ and $[\![C_2]\!]_L^{\rho_{12}',\mathcal{H}}$. In the following proof, we always consider a specific sample ω when this is not pointed out specially. If the two do not hold, assume C_1 fails to hold not later than C_2, and the first time for which C_1 does not hold is t_1 (when it exists), then for all $t < t_1$, C_2 holds. There are three kinds of formulas at time t_1 in C_1: if the formula is for internal variables or internal communication (between P and Q) non-readiness, then it will not depend on Q or E, according to the fact that C_1 holds before time t_1, it must hold at t_1; if the formula is for external communication readiness, first from compatibility check, for any channel $dh \in X_1 \setminus X$, it does not occur in C_2, then we have $E \restriction_{dh} \Rightarrow E_1 \restriction_{dh}$, where $E \restriction_{dh}$ extracts formulas related to communications along dh from E. Then from $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, we have $[\![E_1 \restriction_{dh}]\!]_L^{\rho_2,\mathcal{H}}$, and thus $[\![E_1 \restriction_{dh}]\!]_L^{\rho_{12}\uplus\rho_2,\mathcal{H}}$. By induction hypothesis, the formula considered must hold at t_1; if the formula is for internal communication readiness, then there must exist an open interval (t_0, t_1) during which it is not satisfied. From the assumption, C_2

holds in the interval (t_0, t_1), thus $E_1 \restriction_X$ holds in the interval (t_0, t_1). By induction, the internal communication readiness assertions in C_1 hold in the interval (t_0, t_1). We thus get a contradiction. Therefore, we can get the fact that, both $[\![C_1]\!]_L^{\rho'_{11},\mathcal{H}}$ and $[\![C_2]\!]_L^{\rho'_{12},\mathcal{H}}$ hold. On the other hand, if such t_1 does not exist, there must exist an open interval (t_2, t_3) such that for all $t \leq t_2$, C_1 and C_2 hold, while C_1 does not hold in (t_2, t_3). The proof is very similar to the above case. We omit it here for avoiding repetition.

Based on the above facts, from $[\![A_1 \wedge now = 0 \wedge tr = \varepsilon]\!]_L^{\rho_1}$ and $[\![E]\!]_L^{\rho_1,\mathcal{H}}$, and compatibility check, we have therefore $[\![E_1]\!]_L^{\rho_{12} \uplus \rho_2,\mathcal{H}}$. Similarly, we can get for another process Q that $[\![A_2 \wedge now = 0 \wedge tr = \varepsilon]\!]_L^{\rho_{12}}$, and $[\![E_2]\!]_L^{\rho_{11} \uplus \rho_2,\mathcal{H}}$. Then, by induction on P and Q, we have $[\![R_1 \wedge now = o_1]\!]_L^{\rho'_{11}}$ and $[\![C_1]\!]_L^{\rho'_{11},\mathcal{H}}$, $[\![R_2 \wedge now = o_2]\!]_L^{\rho'_{12}}$ and $[\![C_2]\!]_L^{\rho'_{12},\mathcal{H}}$ respectively.

Notice that $\rho'_{11} \uplus \rho'_{12}$, i.e. ρ'_1, only redefines the values of tr and now, where the communications are arranged in the order according to their occurring time, and variable now takes the greater value between $\rho'_{11}(\omega)(now)$ and $\rho'_{12}(\omega)(now)$ for all ω. Obviously, we have $[\![R_1[\gamma_1/tr, o_1/now] \wedge R_2[\gamma_2/tr, o_2/now] \wedge now = o_m]\!]_L^{\rho'_1}$. And, $[\![\gamma_1 \restriction_X = \gamma_2 \restriction_X]\!]_L^{\rho'_1}$ holds because of synchronization. From the definition of $\uplus$, $\rho'_1(tr)(t) \in \rho'_{11}(tr)(t) \| \rho'_{12}(tr)(t)$, we can easily get the fact $[\![tr = (\gamma_1 \underset{C}{\parallel} \gamma_2))]\!]_L^{\rho'_1}$. Thus the postcondition of $P||Q$ holds for the final state.

From $[\![C_1]\!]_L^{\rho'_{11},\mathcal{H}}$ and $[\![C_2]\!]_L^{\rho'_{12},\mathcal{H}}$, considering that only now changes and matters, we have $[\![C_1[o_1/now] \wedge C_2[o_2/now]]\!]_L^{\rho'_1,\mathcal{H}}$. After P or Q terminates, only rdy, tr and now may change, plus the fact that R_1 and R_2 do not contain readiness, $R_1 \Rightarrow R'_1$, $R_2 \Rightarrow R'_2$, and the assumption that R'_1, R'_2 do not contain tr, we have $[\![R'_1[o_1/now] \ \mathsf{dr}\ [o_1, now)]\!]_L^{\rho'_1,\mathcal{H}}$ and $[\![R'_2[o_2/now] \ \mathsf{dr}\ [o_2, now)]\!]_L^{\rho'_1,\mathcal{H}}$. Thus the history formula of $P||Q$ holds for the final flow.

- Case Repetition P^*: From the operational semantics, there must exist a finite integer $n \geq 0$, and $\rho_{11}, ..., \rho_{1n}$ such that P executes for n times iteratively, and $(P \| PE_0, \rho_{10} \uplus \rho_{20}) \to (\epsilon \| PE_1, \rho_{11} \uplus \rho_{21}, \mathcal{H}_1) \ldots (P \| PE_{n-1}, \rho_{1n-1} \uplus \rho_{2n-1}) \to (\epsilon \| PE_n, \rho_{1n} \uplus \rho_{2n}, \mathcal{H}_n)$ where $\rho_{10} = \rho_1, \rho_{20} = \rho_2, PE_0 = PE, \rho_{1n} = \rho'_1, \rho_{2n} = \rho'_2$ and $\mathcal{H} = \mathcal{H}_1^\frown \cdots^\frown \mathcal{H}_n$. Assume $[\![A \wedge now = o]\!]_L^{\rho_1}$ and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, we need to prove that $[\![A[\mathbf{v}_o/\mathbf{v}] \wedge \mathit{Inv}_{loop}]\!]_L^{\rho'_1}$ and $[\![\mathit{Inv}_{loop} \ \mathsf{dr}\ [o, now)]\!]_L^{\rho'_1,\mathcal{H}}$.

  If $n = 0$, then we have $\rho_1 = \rho'_1$. From $[\![A]\!]_L^{\rho_1}$ and $A \Rightarrow \mathit{Inv}_{loop}$, $[\![\mathit{Inv}_{loop}]\!]_L^{\rho_1}$ holds. The fact is proved directly. Suppose the fact holds for the case $n = k > 1$, then we prove the case for $n = k+1$. From $[\![A \wedge now = o]\!]_L^{\rho_1}$, $A \Rightarrow \mathit{Inv}_{loop}$, and $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, then by induction hypothesis, we have $[\![\mathit{Inv}_{loop}]\!]_L^{\rho_{1k}}$, and $[\![\mathit{Inv}_{loop} \ \mathsf{dr}\ [o, now)]\!]_L^{\rho_{1k},\mathcal{H}_1^\frown \cdots^\frown \mathcal{H}_k}$. Let o_l denote the starting time of the $k+1$-th iteration, $[\![now = o_l]\!]_L^{\rho_{1k}}$. On the other hand, from $[\![E]\!]_L^{\rho_2,\mathcal{H}}$, we have $[\![E[o/now]]\!]_L^{\rho_{2k},\mathcal{H}}$. By induction hypothesis, the facts $[\![\mathit{Inv}_{loop}]\!]_L^{\rho_{1k+1}}$, and $[\![\mathit{Inv}_{loop} \ \mathsf{dr}\ [o_l, now)]\!]_L^{\rho_{1k+1},\mathcal{H}_{k+1}}$ hold. $[\![\mathit{Inv}_{loop}]\!]_L^{\rho_{1n}}$, and $[\![\mathit{Inv}_{loop} \ \mathsf{dr}\ [o, now)]\!]_L^{\rho_{1n},\mathcal{H}}$ are proved finally.

□

6.4. Discovery of Differential Invariants for SDE

The most challenging problem for the deductive verification of stochastic hybrid systems is to discover the local differential invariants of SDE. In [32], Prajna et al. extended differential invariant generation approach based on barrier certificates for traditional hybrid systems to stochastic hybrid systems. Their approach requires all expressions occurring in a template of invariant to be a super-martingale w.r.t the considered SDE, and then exploits semi-definite programming to solve the constraints on the parameters in the template. Their approach is very restrictive, as only very few invariants can be synthesized. In [30], Platzer proposes a special form of differential invariants for a SDE under some given conditions, which is represented by the following theorem.

Theorem 6.2 (Differential Invariant of SDE [30]). Assume the SDE is $\langle ds = bdt + \sigma dW \& PB \rangle$, where *PB* is a boolean formula (that holds almost surely), and the precondition is A. Let $\lambda > 0, p \geq 0$ be real values. If $f(s) \in C^2(\mathbb{R}^n, \mathbb{R})$ has compact support on *PB*, and if the following conditions hold:

$$A \to PB \to f \leq \lambda p \quad B \to f \geq 0 \wedge \mathcal{L}f \leq 0$$

where $\mathcal{L}f$ represents the Lie derivative of f, then the probabilistic formula $P(f(s) \geq \lambda) \leq p$ is a differential invariant of the SDE $\langle ds = bdt + \sigma dW \& PB \rangle$.

The theorem states that, if the initial state of the SDE satisfies $f \leq \lambda p$, and inside the domain PB, f is always non-negative and $\mathcal{L}f$ is non-positive, then during the whole evolution of the SDE, the probability of $f(s) \geq \lambda$ is less than or equal to p. By applying the result, the safety property of the aircraft example can be proved.

Example 6.1. For the aircraft example, define $f(x, y)$ as $|y|$, assume $f(x_s, y_0) = |y_0| \leq \lambda p$, where $p \in [0, 1]$. Obviously, $B \to (f \geq 0) \wedge (\mathcal{L}f \leq 0)$ holds. By applying the inference rule of SDE and Theorem 6.2, we obtain the following result:

$$\{now = o; \mathbf{True}\}\, P_{Air}\, \{\ (x \leq x_s \vee x \geq x_e) \wedge P(f(s) \geq \lambda) \leq p; (PB \wedge P(f(s) \geq \lambda) \leq p)\ \mathsf{dr}\ [o, now)\ \}$$

which shows that, the probability of the aircraft entering the dangerous state is always less than or equal to p during the flight. Thus, to guarantee the safety of the aircraft, p should be as little as possible. For instance, if the safety factor of the aircraft is required to be 99.98%, then p should be less than or equal to 0.0002, and in correspondence, $|y_0| \leq \frac{\lambda}{5000}$ should be satisfied.

However, because of the restrictive conditions, Theorem 6.2 has some limitations for solving SDEs. In our recent work, we present a method for reachability analysis for SDE using the existing partial differential equation (PDE) solvers. The differential invariant problem of SDE is one of our future work.

The global loop invariants for stochastic hybrid systems can be computed according to the traditional approach. In particular, the problem on computing global invariants of hybrid systems has been investigated in [31, 41].

7. Conclusion

This paper presents stochastic HCSP (SHCSP) for modelling hybrid systems with probability and stochasticity. SHCSP is expressive by combining interacting discrete, continuous and stochastic dynamics. We have defined the operational semantics of stochastic HCSP based on probabilistic states and related operations, and proved that the semantics is well-defined with respect to stochasticity. We propose an assertion language for specifying time-related and probability-related properties based on (timed) probabilistic formulas, and have proved the assertions are well-defined with respect to stochasticity. For handling SDEs and repetition, we propose the notions of local stochastic differential invariants and global loop invariants. To the end, we define a compositional Hoare Logic for specifying and verifying SHCSP processes, which is able to reason about how the probability of a property changes with respect to the execution of a process. To illustrate our approach, we model and verify a case study on a flight planing problem throughout the paper. Our future work includes the investigation of the local stochastic differential invariant for SDEs, and the application of our framework to more interesting case studies.

Acknowledgements. We are indebted Mr. Yu Peng for his contribution on the earlier conference version. We also thank the anonymous referees for their constructive comments that help to improve this paper very much.

The first two authors are supported partly by "973 Program" under grant No. 2014CB340701, by NSFC under grant No. 91418204. The second author is also supported partly by the Outstanding Youth Funding of NSFC under grant No. 61625206. The second and the third authors are supported partly by CDZ project CAP (GZ 1023), and by the CAS/SAFEA International Partnership Program for Creative Research Teams. The third author is also supported partly by NSFC under grant No. 61532019 and 61472473.

References

[1] A. Abate, M. Prandini, J. Lygeros, and S. Sastry. Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11):2724–2734, 2008.

[2] E. Altman and V. Gaitsgory. Asymptotic optimization of a nonlinear hybrid system governed by a Markov decision process. *SIAM Journal of Control and Optimization*, 35(6):2070–2085, 1997.

[3] R. Banach, M. Butler, S. Qin, N. Verma, and H. Zhu. Core Hybrid Event-B I: Single Hybrid Event-B machines. *Science of Computer Programming*, 105:92–123, 2015.

[4] M. L. Bujorianu. Extended stochastic hybrid systems and their reachability problem. In *HSCC'04*, volume 2993 of *LNCS*, pages 234–249, 2004.

[5] M. L. Bujorianu and J. Lygeros. Toward a general theory of stochastic hybrid systems. *Lecture Notes in Control and Information Sciences (LNCIS)*, 337:3–30, 2006.

[6] M. L. Bujorianu, J. Lygeros, and M. C. Bujorianu. Bisimulation for general stochastic hybrid systems. In *HSCC'05*, volume 3414 of *LNCS*, pages 198–214, 2005.

[7] E. Castelan and J. Hennet. On invariant polyhedra of continuous-time linear systems. *IEEE Trans. Autom. Control*, 38(11):1680–1685, 1993.

[8] M. Chen, M. Fränzle, Y. Li, P. N. Mosaad, and N. Zhan. Validated simulation-based verification of delayed differential dynamics. In *FM'16*, volume 9995 of *LNCS*, pages 137–154, 2016.

[9] M. Fränzle, E. M. Hahn, H. Hermanns, N. Wolovick, and L. Zhang. Measurability and safety verification for stochastic hybrid systems. In *HSCC'11*, pages 43–52. ACM, 2011.

[10] E. Goubault, J.-H. Jourdan, S. Putot, and S. Sankaranarayanan. Finding non-polynomial positive invariants and Lyapunov functions for polynomial systems through Darboux polynomials. In *ACC 2014*, pages 3571–3578, 2014.

[11] F. Gretz, J.-P. Katoen, and A. McIver. Operational versus weakest pre-expectation semantics for the probabilistic guarded command language. *Performance Evaluation*, 73:110 – 132, 2014.

[12] S. Gulwani and A. Tiwari. Constraint-based approach for analysis of hybrid systems. In A. Gupta and S. Malik, editors, *CAV'08*, volume 5123 of *LNCS*, pages 190–203. Springer Berlin Heidelberg, 2008.

[13] E. M. Hahn, A. Hartmanns, H. Hermanns, and J. Katoen. A compositional modelling and analysis framework for stochastic hybrid systems. *Formal Methods in System Design*, 43(2):191–232, 2013.

[14] E. M. Hahn, H. Hermanns, B. Wachter, and L. Zhang. PASS: abstraction refinement for infinite probabilistic models. In *TACAS'10*, volume 6015 of *LNCS*, pages 353–357, 2010.

[15] J. I. Hartog. Verifying probabilistic programs using a hoare like logic. In *ASIAN 1999*, volume 1742 of *LNCS*, pages 113–125, 1999.

[16] J. He. From CSP to hybrid systems. In *A Classical Mind, Essays in Honour of C.A.R. Hoare*, pages 171–189. Prentice Hall International (UK) Ltd., 1994.

[17] T. A. Henzinger. The theory of hybrid automata. In *LICS'96*, pages 278–292, July 1996.

[18] C. A. R. Hoare. An axiomatic basis for computer programming. *Commun. ACM*, 12(10):576–580, 1969.

[19] C. A. R. Hoare. *Communicating Sequential Processes*. Prentice-Hall, 1985.

[20] J. Hu, J. Lygeros, and S. Sastry. Towards a theory of stochastic hybrid systems. In *HSCC'02*, volume 1790 of *LNCS*, pages 160–173, 2002.

[21] M. Kwiatkowska, G. Norman, D. Parker, and H. Qu. Assume-guarantee verification for probabilistic systems. In *TACAS 2010*, volume 6015 of *LNCS*, pages 23–37, 2010.

[22] J. Liu, J. Lv, Z. Quan, N. Zhan, H. Zhao, C. Zhou, and L. Zou. A calculus for hybrid CSP. In *APLAS'10*, volume 6461 of *LNCS*, pages 1–15. 2010.

[23] J. Liu, N. Zhan, H. Zhao, and L. Zou. Abstraction of elementary hybrid systems by variable transformation. In *FM 2015*, volume 9109 of *LNCS*, pages 360–377. Springer International Publishing, 2015.

[24] J. Meseguer and R. Sharykin. Specification and analysis of distributed object-based stochastic hybrid systems. In *HSCC'06*, volume 3927 of *LNCS*, pages 460–475, 2006.

[25] C. Morgan, A. McIver, and K. Seidel. Probabilistic predicate transformers. *ACM Transactions on Programming Languages and Systems*, 18(3):325–353, 1996.

[26] C. Morgan, A. McIver, K. Seidel, and J. W. Sanders. Refinement-oriented probability for CSP. *Formal Asp. Comput.*, 8(6):617–647, 1996.

[27] B. Øksendal. *Stochastic Differential Equations: An Introduction with Applications*. Springer, 2007.

[28] Y. Peng, S. Wang, N. Zhan, and L. Zhang. Extending hybrid CSP with probability and stochasticity. In *SETTA'15*, volume 9409 of *LNCS*, pages 87–102, 2015.

[29] A. Platzer. Differential-algebraic dynamic logic for differential-algebraic programs. *J. Log. and Comput.*, 20(1):309–352, Feb. 2010.

[30] A. Platzer. Stochastic differential dynamic logic for stochastic hybrid programs. In *CADE'11*, volume 6803 of *LNCS*, pages 446–460. 2011.

[31] A. Platzer and E. M. Clarke. Computing differential invariants of hybrid systems as fixedpoints. In *CAV 2008*, volume 5123 of *LNCS*, pages 176–189. 2008.

[32] S. Prajna, A. Jadbabaie, and G. Pappas. A framework for worst-case and stochastic safety verification using barrier certificates. *IEEE Transactions on Automatic Control*, 52(8):1415–1428, 2007.

[33] M. Prandini and J. Hu. Application of reachability analysis for stochastic hybrid systems to aircraft conflict prediction. In *47th IEEE Conference on Decision and Control (CDC)*, pages 4036 – 4041. IEEE, 2008.

[34] R. Rebiha, N. Matringe, and A. V. Moura. Transcendental inductive invariants generation for non-linear differential and hybrid systems. In *HSCC 2012*, pages 25–34, New York, NY, USA, 2012. ACM.

[35] S. Sankaranarayanan. Automatic invariant generation for hybrid systems using ideal fixed points. In *HSCC'10*, pages 221–230, New York, NY, USA, 2010. ACM.

[36] S. Sankaranarayanan, H. B. Sipma, and Z. Manna. Constructing invariants for hybrid systems. In R. Alur and G. J. Pappas, editors, *HSCC'04*, volume 2993 of *LNCS*, pages 539–554. 2004.

[37] J. Sproston. Decidable model checking of probabilistic hybrid automata. In *Formal Techniques in Real-Time and Fault-Tolerant Systems*, volume 1926 of *LNCS*, pages 31–45. 2000.

[38] X. Tang and X. Zou. Global attractivity in a predator-prey system with pure delays. *Proc. Edinburgh Math. Soc.*, 51:495–508, 2008.

[39] S. Wang, N. Zhan, and D. Guelev. An assume/guarantee based compositional calculus for hybrid CSP. In M. Agrawal, S. Cooper, and A. Li, editors, *TAMC 2012*, volume 7287 of *LNCS*, pages 72–83. Springer Berlin Heidelberg, 2012.

[40] Z. Yang, W. Lin, and M. Wu. Exact safety verification of hybrid systems based on bilinear SOS representation. *ACM Trans. Embed. Comput. Syst.*, 14(1):16:1–16:19, Jan. 2015.

[41] N. Zhan, S. Wang, and H. Zhao. Formal modelling, analysis and verification of hybrid systems. In *Unifying Theories of Programming and Formal Engineering Methods*, volume 8050 of *LNCS*, pages 207–281. 2013.

[42] L. Zhang, Z. She, S. Ratschan, H. Hermanns, and E. M. Hahn. Safety verification for probabilistic hybrid systems. In *CAV'10*, volume 6174 of *LNCS*, pages 196–211, 2010.

[43] C. Zhou, J. Wang, and A. P. Ravn. A formal description of hybrid systems. In *Hybrid Systems III*, volume 1066 of *LNCS*, pages 511–530. 1996.

[44] L. Zou, M. Fränzle, N. Zhan, and P. N. Mosaad. Automatic verification of stability and safety for delay differential equations. In *CAV'15*, volume 9207 of *Lecture Notes in Computer Science*, pages 338–355, 2015.

[45] P. Zuliani, A. Platzer, and E. M. Clarke. Bayesian statistical model checking with application to stateflow/simulink verification. *Formal Methods in System Design*, 43(2):338–367, 2013.