Safety Logics I: Absolute Safety

Zhisheng Huang and John Bell*
Applied Logic Group
Department of Computer Science
Queen Mary and Westfield College
University of London
email: {huang, jb}@dcs.qmw.ac.uk

Abstract

In this paper we distinguish between absolute safety and normative safety, and develop a formal theory of absolute safety. We start with the idea of a *disastrous state* (a state in which some disastrous fact is true) and define a *disastrous action* to be an action which always leads to a disastrous state. We then define *dangerous actions* to be actions which may lead to disastrous states, and *dangerous states* to be states in which every possible action is dangerous. Finally, *safe actions* (*states*) are defined to be actions (states) which are not dangerous. Using Dynamic Logic as a basis, we develop a formal possible-worlds semantics for our theory of safety, and give the basic safety logic **SL** and several extensions of it. We then give an example of reasoning about safety in nuclear power stations, and conclude with an indication of how we are building on this work to produce a formal account of normative safety.

1 Introduction

The task of planning for rational agents is to find plans which will achieve their goals, given an appropriate theory of action and change. In safety-critical systems, agents have to consider not only how they can achieve their goals, they also have to consider how they can achieve them safely. One solution is constraint-based planning, however the constraints have to be *specified* in advance, and there is no principled attempt to *reason* about them. As a result, the limitations of this approach will make themselves felt as the the application domains increase in complexity. The alternative is to develop formal logics which agents can reason about safety. This

*This research forms part of the RED project, *Decision Support Systems, Safety and Liability*, and is supported by the United Kingdom Engineering and Physical Sciences Research Council under grant number GR/H 12348.

is a complex task, as reasoning about safety may involve commonsense reasoning about actions and change, time, beliefs, goals, rationality, etc. We begin the task in this paper by abstracting away from many of these features, and developing a formal theory of what we consider to be the essential features of safety, and which we will subsequently extend.

We start with disasters and the idea of a *disastrous state*. A disastrous state is one in which some fact, which the agent considers to be disastrous, is true. It is assumed that such states are abhorrent to the agent and that the agent tries to avoid them if at all possible. For instance, a typical medical safety-critical system would consider states in which the patient dies as a result of treatment to be disastrous, and would try to avoid them at all costs. We further assume that the application domain is such that there is no question as to which states the agent considers to be disastrous. Of course, what counts as a disastrous state may be context-dependent. For example, the agent might consider a state in which it has missed a plane to be disastrous if it has a non-transferrable ticket, but it need not do so if there is another flight and the ticket is transferrable. However, in some domains, disasters may not by context dependent. We will refer to disastrous states of this kind as *stable* disastrous states. However, in most applications disasters are *unstable*. Furthermore, as will become clear in our formal analysis, stable disastrous states can be viewed as a special case. Consequently, when talking about disastrous states we will usually take them to be unstable. A *disastrous action* can now be defined to be an action which *always* leads to a disastrous state.

We then define a *dangerous action* to be an action which *may* lead to a disastrous state, and a *dangerous state* to be state in which every action which the agent can perform – every action which is open to the agent – is dangerous. Note that disastrous actions (states) are thus dangerous actions (states).

Finally, we define a *safe action* (a *safe state*) to be an action (a state) which is not dangerous.

As an example of these definitions consider an agent who is learning to swim, and suppose that the agent is alone in the pool. The agent's goal is to enjoy its swim, and, naturally, the agent considers that its death by drowning is a disaster; that is, the agent takes states in which it has drowned to be disastrous states. The agent enjoys swimming, regardless of whether it is swimming in shallow or deep water. Swimming is shallow water is safe; the agent is never in danger of drowning because it can always stand up. However, if the agent is in deep water this action is no longer open to it and all its attempts to stay afloat amount to helpless floundering. Consequently it is in danger of drowning no matter what action it takes, and will do so unless someone comes to its aid. Swimming in deep water is therefore dangerous (or unsafe) for the agent. Note that being in deep water is only unsafe because *all* of the actions open to the agent are dangerous. If, for example, the agent could capture a buoy whenever it wanted – that is, if a safe action were available to the agent – then swimming in the deep end would be safe.

We have defined an *absolute* notion of safety, in which an action is safe only if there is absolutely no possibility of it leading to a dangerous state. In practice, safety is a *normative* concept. That is, in everyday life almost all actions are dangerous, as there is always *some* (perhaps remote) possibility that the action will lead to a disastrous state. So in practice we are interested in which actions are *normally* safe; that is, in actions which do not normally lead to disastrous states. For example, it is not absolutely safe to fly by major airlines (because, there is always the possibility, however remote, of a crash) but it is normally safe to do so (as flights do not normally end in disaster). However, it is useful to start with the formalisation of absolute safety for two reasons. Firstly, the formalisation of reasoning about absolute safety is central to the development of safety-critical systems. Systems of this kind involve domains in which many of the dangers of daily life are removed, or are not considered to be possibilities. So it makes sense to define (absolutely) safe actions and states, and then to aim to maintain safety by trying to ensure that only safe actions are performed. The second reason for starting with absolute safety is methodological: the formalisation of absolute safety provides a conceptual and theoretical basis on which further work, for example on the formalization of normative safety can build.

We have chosen to formalize absolute safety in Dynamic Logic [3], as this offers a powerful tool for formalizing actions using classical logic and Kripke semantics. In particular, states are represented as possible worlds, and actions are represented as accessibility relations between worlds. The crucial distinction between the necessary and the possible consequences of actions in our informal analysis is thus readily and naturally formalized. In the next section we define $\mathcal{SL}$, our language of safety, and give formal semantics for it. We then turn to logics of safety and give our basic safety logic **SL** and discuss several extensions of it. An example of reasoning about safe formulas in nuclear power stations is then given. Finally, we give a brief indication of ongoing work on formalising normative safety and of how this builds on the present work.

2 The Language $\mathcal{SL}$

Let Φ_0 be a set of primitive propositions, and PA be a set of primitive actions. We will use the Roman letters $a, b, \ldots$ (with or without subscripts or superscripts) to denote actions, and lower case Greek letters $\phi, \psi, \ldots$ (with or without subscripts or superscripts) to denote formulas.

Definition 1 (Actions) *The set of (composite) actions $ACTION$ is, as usual, defined to be the smallest set which is closed under the following syntactic rules:*

- *If $a \in PA$ then $a \in ACTION$*
- *If $a, b \in ACTION$ then $(a \cup b), (a; b) \in ACTION$*

Here $\cup$ is the non-deterministic choice operator (so $a \cup b$ means "do either a or b non-deterministically"), and ';' is the sequence operator (so $a; b$ means "do a and then do b"). The definition can be extended to include further action operators, such as the "Kleene star", '$*$', and the '**skip**' operator. However, their inclusion is not required for present purposes, and is thus left for future work.

Definition 2 (The language $\mathcal{SL}$) *The language $\mathcal{SL}$ is the minimal set of formulas which satisfies the following conditions:*

- *If $p \in \Phi_0$ then $p \in \mathcal{SL}$.*
- *If $\phi, \psi \in \mathcal{SL}$ then $\neg\phi \in \mathcal{SL}$ and $\phi \wedge \psi \in \mathcal{SL}$.*
- *If $a \in ACTION$ and $\phi \in \mathcal{SL}$ then $\langle a \rangle \phi \in \mathcal{SL}$.*
- *If $\phi \in \mathcal{SL}$ then $\mathbf{Dis}\phi \in \mathcal{SL}$ and $\mathbf{Dan}\phi \in \mathcal{SL}$.*
- *If $a \in ACTION$ then $\mathbf{Dis}(a) \in \mathcal{SL}$ and $\mathbf{Dan}(a) \in \mathcal{SL}$.*

Informally, $\langle a \rangle \phi$ states that doing a makes ϕ possible, that is ϕ *may* be true after action a is taken. Then $[a]\phi$ is defined as $\neg\langle a \rangle \neg\phi$. Thus $[a]\phi$ states that doing a makes ϕ inevitable; that is, ϕ *will* be true after action a is taken. A sentence of the form $\mathbf{Dis}\phi$ ($\mathbf{Dan}\phi$) means that ϕ is disastrous (dangerous). Similarly $\mathbf{Dis}(a)$ ($\mathbf{Dan}(a)$) means that action a is disastrous (dangerous). The safety operators are then introduced by definition as follows:

- $\mathbf{Safe}(a) \overset{\text{def}}{\iff} \neg\mathbf{Dan}(a)$
- $\mathbf{Safe}\phi \overset{\text{def}}{\iff} \neg\mathbf{Dan}\phi$

The falsum '$\perp$' and the remaining Boolean connectives '$\vee$', '$\rightarrow$' and '$\leftrightarrow$' are defined using the negation and conjunction operators in the usual way.

Recall that our analysis of safety started with disastrous states. We therefore extend the standard Kripke models of propositional Dynamic Logic by adding the component D in order to represent disastrous states. In the simplest case D can just be a set of worlds. This would give a model with *stable* disastrous states; that is, it would give a model in which what counts as a disaster is independent of any particular world. However, as the airline-ticket example illustrated, disasters are typically unstable. Consequently, we prefer to model unstable disasters, and to treat stable disasters as a special case; see Section 5.

Definition 3 ($\mathcal{SL}$-models) *A* model for $\mathcal{SL}$ *is a tuple $M = \langle W, \{R^a\}_{a \in PA}, D, V \rangle$, where*

- *W is a set of possible worlds,*
- *$R^a \subseteq W \times W$ is a binary accessibility relation for each primitive action $a \in PA$,*
- *$D : W \rightarrow \mathcal{P}(W)$ assigns to each world w the set $D(w)$ of worlds which are disastrous with respect to w, and*
- *$V : \Phi_0 \rightarrow \mathcal{P}(W)$ is the usual valuation function.*

As usual, we extend the accessibility relations for primitive actions to complex actions as follows:

- $R^a_+ = R^a$ *(where a is primitive)*
- $R^{a\cup b}_+ = R^a_+ \cup R^b_+$
- $R^{a;b}_+ = R^a_+ \circ R^b_+$ *(composition of relations)*

We will usually write R^a rather than R^a_+ when there is no danger of ambiguity.

Finally, we require that the global accessibility relation $R = \{\langle w, w'\rangle : wR^a_+w'$ for some $a\}$ is a chronology – that is, that R is a partial order which is required to be backwards-linear (or anti-convergent).

Intuitively wR^aw' means that world w' is one possible outcome of doing action a in world w.

The states of our informal analysis are thus represented by worlds, and actions are represented by accessibility relations amongst them. In order to capture the idea of the consequences of an action *leading to* a disastrous state, we define the *trace* of action each a from each world w. The trace of a from w, is a tree rooted at w each branch of which represents a possible sequence of events resulting from doing a at w; the trace of a from w is thus the tree of all states that doing a at w can lead to. Thus a leads to a disastrous world (state) if there is such a world on some branch of the trace of a from w. Formally, let $R^aw = \{w' \in W : \langle w, w'\rangle \in R^a_+\}$ be the set of all possible outcomes of doing the (possibly compound) action a at w. Then $Trace(a, w)$ is defined as follows.

- $Trace(a, w) = \{\langle w, w\rangle\} \cup \{\langle w, w'\rangle : w' \in R^aw\}$ where a is primitive
- $Trace((a \cup b), w) = Trace(a, w) \cup Trace(b, w)$
- $Trace((a; b), w) = Trace(a, w) \cup \bigcup_{w' \in R^aw} Trace(b, w') \cup$
  $(Trace(a, w) \circ \bigcup_{w' \in R^aw} Trace(b, w'))$

For brevity's sake we will write $w_1R^aww_2$ if $\langle w_1, w_2\rangle \in Trace(a, w)$; thus, $w_1R^aww_2$ states that w_1 occurs no later than w_2 on a branch in the trace of a from w.

Definition 4 (Truth conditions for $\mathcal{SL}$) *Let $M = \langle W, \{R^a\}_{a\in PA}, D, V\rangle$ be an $\mathcal{SL}$-model. Then a sentence ϕ is true at a world w in M (written $M, w \models \phi$, or, equivalently, $w \in [\![\phi]\!]^M$) as follows.*

$$
\begin{array}{lll}
M, w \models p & \textit{iff} & w \in V(p) \textit{ where } p \textit{ primitive} \\
M, w \models \neg\psi & \textit{iff} & M, w \not\models \psi \\
M, w \models \psi \wedge \chi & \textit{iff} & M, w \models \psi \textit{ and } M, w \models \chi \\
M, w \models \langle a\rangle\psi & \textit{iff} & \exists w'(w' \in R^aw \textit{ and } w' \in [\![\psi]\!]^M) \\
M, w \models \mathbf{Dis}\psi & \textit{iff} & [\![\psi]\!]^M \subseteq D(w) \\
M, w \models \mathbf{Dis}(a) & \textit{iff} & \forall w_1(wR^aww_1 \Rightarrow \\
 & & \exists w_2((w_1R^aww_2 \textit{ or } w_2R^aww_1) \textit{ and } w_2 \in D(w))) \\
M, w \models \mathbf{Dan}(a) & \textit{iff} & \exists w'(wR^aww' \textit{ and } w' \in D(w)) \\
M, w \models \mathbf{Dan}\psi & \textit{iff} & \forall w_1(w_1 \in [\![\psi]\!]^M \Rightarrow \forall a(R^aw_1 \neq \emptyset \Rightarrow \\
 & & \exists w_2(w_1R^aw_1w_2 \textit{ and } w_2 \in D(w))))
\end{array}
$$

As usual, a sentence ϕ is said to be true in a model M (written $M \models \phi$) if $M, w \models \phi$ for all worlds w in M, and ϕ is said to be valid (written $\models \phi$) if ϕ is true in all models.

The truth condition for dangerous states is a bit tricky. The condition $R^a w_1 \neq \emptyset$ means that action a *can* be taken (is open) at world w_1; a new CAN operator is therefore not necessary. Let $Rw = \{a \in ACTION : R^a w \neq \emptyset\}$ be the set of actions which are open to the agent at w. For a set of possible worlds X, we define the *zone* of X as follows:

$Zone(X) = \{w : \forall a \in Rw \exists w' (wR^a ww' \text{ and } w' \in X)\};$

thus $w \in Zone(X)$ if every possible action in w might lead to a world in X. The truth condition for dangerous states can then be re-stated as:

$M, w \models \mathbf{Dan}\phi$ iff $[\![\phi]\!]^M \subseteq Zone(D(w))$.

We can then define the three kinds of worlds as follows.

- w' is a disastrous world with respect to w iff $w' \in D(w)$.
- w' is a dangerous world with respect to w iff $w' \in Zone(D(w))$.
- w' is a safe world with respect to w iff $w' \in W/Zone(D(w))$.

The possibility operator $\langle\cdot\rangle$ can be extended to the more general operator $\langle\cdot\rangle^*$. Informally, $\langle a\rangle^*\phi$ means that ϕ is possible after some sub-action of action a. The new operator is defined as follows:

- $\langle a\rangle^*\phi \overset{\text{def}}{\Longleftrightarrow} \langle a\rangle\phi \vee \phi$ where a is primitive.
- $\langle a \cup b\rangle^*\phi \overset{\text{def}}{\Longleftrightarrow} \langle a \cup b\rangle\phi \vee (\langle a\rangle^*\phi \vee \langle b\rangle^*\phi) \vee \phi$
- $\langle a; b\rangle^*\phi \overset{\text{def}}{\Longleftrightarrow} \langle a; b\rangle\phi \vee \langle a\rangle^*\phi \vee \langle a\rangle\langle b\rangle^*\phi \vee \phi$

3 The Safety Logic SL

The logic of safety, **SL**, consists of the following axioms, definitions and inference rules.[1]

Axioms

(PC) All propositional tautologies
(A1) $\neg\langle a\rangle\bot$

[1] Some of the axioms may be redundant. We leave this question for further work.

(A2) $\langle a\rangle(\phi \vee \psi) \rightarrow \langle a\rangle\phi \vee \langle a\rangle\psi$
(A3) $\langle a;b\rangle\phi \leftrightarrow \langle a\rangle\langle b\rangle\phi$
(A4) $\langle a \cup b\rangle\phi \leftrightarrow \langle a\rangle\phi \vee \langle b\rangle\phi$

(D1) $\mathbf{Dis}\bot$
(D2) $\mathbf{Dis}(\phi \vee \psi) \rightarrow \mathbf{Dis}\phi \vee \mathbf{Dis}\psi$
(D3) $\mathbf{Dis}\phi \wedge \mathbf{Dis}\psi \rightarrow \mathbf{Dis}(\phi \vee \psi)$
(D4) $\mathbf{Dis}(a \cup b) \leftrightarrow \mathbf{Dis}(a) \wedge \mathbf{Dis}(b)$

(DA1) $\mathbf{Dan}\phi \wedge \mathbf{Dan}\psi \rightarrow \mathbf{Dan}(\phi \wedge \psi)$
(DA2) $\langle a\rangle\phi \wedge \mathbf{Dis}\phi \rightarrow \mathbf{Dan}(a)$
(DA3) $\mathbf{Dan}(a \cup b) \leftrightarrow \mathbf{Dan}(a) \vee \mathbf{Dan}(b)$
(DA4) $\mathbf{Dan}(a) \vee (\langle a;b\rangle\phi \wedge \mathbf{Dan}\phi) \rightarrow \mathbf{Dan}(a;b)$
(DA5) $\mathbf{Dan}\phi \wedge \phi \wedge \neg[a]\bot \rightarrow \mathbf{Dan}(a)$

(DD1) $\mathbf{Dis}\phi \rightarrow \mathbf{Dan}\phi$
(DD2) $\mathbf{Dis}(a) \rightarrow \mathbf{Dan}(a)$

Definitions

(A5) $\langle a\rangle^*\phi \leftrightarrow \langle a\rangle\phi \vee \phi$ (where a is primitive)
(A6) $\langle a \cup b\rangle^*\phi \leftrightarrow \langle a \cup b\rangle\phi \vee (\langle a\rangle^*\phi \vee \langle b\rangle^*\phi) \vee \phi$
(A7) $\langle a;b\rangle^*\phi \leftrightarrow \langle a;b\rangle\phi \vee \langle a\rangle^*\phi \vee \langle a\rangle\langle b\rangle^*\phi \vee \phi$
(A8) $[a]\phi \leftrightarrow \neg\langle a\rangle\neg\phi$

(SAdf) $\mathbf{Safe}(a) \leftrightarrow \neg\mathbf{Dan}(a)$
(SSdf) $\mathbf{Safe}\phi \leftrightarrow \neg\mathbf{Dan}\phi$

Inference Rules

(MP) From ϕ and $\phi \rightarrow \psi$ infer ψ
(NECA) From ϕ infer $[a]\phi$
(MONA) From $\langle a\rangle\phi$ and $\phi \rightarrow \psi$ infer $\langle a\rangle\psi$
(SPD) From $\psi \rightarrow \phi$ infer $\mathbf{Dis}\phi \rightarrow \mathbf{Dis}\psi$
(SPDS) From $\psi \rightarrow \phi$ infer $\mathbf{Dan}\phi \rightarrow \mathbf{Dan}\psi$

Theorems and Derived Rules

(SPS) From $\phi \rightarrow \psi$ infer $\mathbf{Safe}\phi \rightarrow \mathbf{Safe}\psi$
(SS1) $\neg\mathbf{Safe}\bot$
(SS2) $\mathbf{Safe}(\phi \wedge \psi) \leftrightarrow \mathbf{Safe}\phi \wedge \mathbf{Safe}\psi$
(SS3) $\mathbf{Safe}(\phi \wedge \psi) \rightarrow \mathbf{Safe}(\phi) \vee \mathbf{Safe}(\psi)$
(SA1) $\mathbf{Safe}(a \cup b) \leftrightarrow \mathbf{Safe}(a) \wedge \mathbf{Safe}(b)$
(SA2) $\mathbf{Safe}(a;b) \rightarrow \mathbf{Safe}(a)$
(SA3) $\mathbf{Safe}(a) \wedge \langle a\rangle^*\phi \rightarrow \neg\mathbf{Dis}\phi$

(TA1) $\phi \to \langle a \rangle^* \phi$

The axioms (A1)-(A4) and the inference rules (NECA) and (MONA) are those of Dynamic Logic. Axiom (D1) states that logical inconsistency is a disaster. (D2) states that disasters are decomposable under disjunction. (D3) states that if two states are disastrous (simultaneously), then one of them must be disastrous. (D4) states that action $a \cup b$ is disastrous exactly when actions a and b are disastrous. (DA1) states that dangerous states are closed under conjunction. (DA2) states that if one possible outcome of the action is disastrous, then the action is dangerous. (DA3) reiterates that dangerous actions are closed under disjunction. (DA4) states that if a sub-action is dangerous, then the whole sequence of the action is dangerous as well. (DA5) says that all actions which can be taken in a dangerous state are dangerous. (DD1) and (DD2) state that all disastrous states and actions are dangerous. The inference rule (SPD) states that any state which implies a disastrous state is also disastrous, and (SPDS) states that the same applies to dangerous states. Some of the properties of safe actions and sates are listed as theorems. (SA2) and (SA3) are of particular interest. (SA2) states that if an action sequence $a;b$ is safe, then action a is safe. It does *not* generally follow that it is safe to do b; as doing a may lead to dangers which are not otherwise apparent. This question is considered further in Section 5.2. Similarly, (SA3) states that a safe action can never lead to a disastrous state; although it may lead to a dangerous one.

Theorem 1 (Soundness of SL) *The logic* **SL** *is sound for the class of* $\mathcal{SL}$*-models.*

PROOF: The proofs for most axioms and inference rules are straightforward from the definitions. For example, proof for (DD1) is as follows. Suppose that $M, w \models \mathbf{Dis}\phi$, then, by the truth condition, we have $[\![\phi]\!]^M \subseteq D(w)$. For any $w_1 \in [\![\phi]\!]^M$ and any action a, we have $w_1 R a w_1 w_1$ by the definition of a trace. So $w_1 R a w_1 w_1$ and $w_1 \in D(w)$. So, for any $w_1 \in [\![\phi]\!]^M$ and any action a, there exists a world w_1 such that $w_1 R a w_1 w_1$ and $w_1 \in D(w)$. So, by the truth condition for dangerous states, we have $M, w \models \mathbf{Dan}\phi$. □

We have not yet proved the completeness of **SL**. There are two difficulties, which arise because the language $\mathcal{SL}$ lacks the required expressive power.

1. From the truth condition for dangerous actions, we know that if an action a is dangerous, then there exists a state ψ such that ψ is accessible via the action a (written $\langle a \rangle^* \psi$), and ψ is disastrous, namely, the following "axiom" holds:

   (DA$^+$) $\mathbf{Dan}(a) \to \langle a \rangle^* \psi \wedge \mathbf{Dis}\psi$ for some formula ψ.

   However, the second-order existential quantification in this axiom cannot be expressed in $\mathcal{SL}$, or in a sensible rule schema.

2. The truth condition for dangerous states involves universal quantification over actions. However, $\mathcal{SL}$ does not permit this.

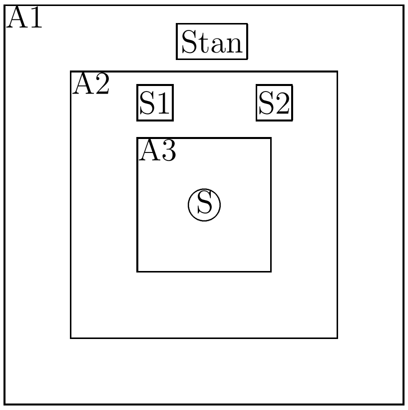

Figure 1: Power station Example

4 Reasoning about Safety: An Example

In order to illustrate how **SL** can be used to reason about safety, this section discusses an example of safety-critical reasoning in the imaginary nuclear power station pictured in Figure 1. The area $A1$ represents those parts of the power station which should be kept free of radiation. This area surrounds the area $A3$ which contains the nuclear reactor, and which is thus an area of high radioactivity. The area $A2$ acts as a "radiation lock" between $A1$ and $A3$; it provides access between the two areas, and is designed to prevent radioactivity spreading from area $A3$ to area $A1$. Area $A2$ contains safety suits $S1$ and $S2$ which are designed to protect against radioactivity. If an agent is wearing one of these suits, the agent can go into area $A3$ safely. But if the agent goes into area $A3$ without a suit, the agent will be exposed to a lethal dose of radiation. Agents who are wearing safety suits are not allowed to enter area $A1$, as the residual radiation on the suits would contaminate area $A1$. An accident has occurred, the reactor is out of control and is heading towards melt-down. The only remaining way of disabling it is by means of switch S in area $A3$. Our agent, *Stan*, who is currently located in area $A1$, has to form and execute a plan for doing this safely.

The initial state can be described as follows;

$(Init)$ $Agent(Stan) \wedge Suit(S1) \wedge Suit(S2) \wedge Switch(S) \wedge Object(S1) \wedge Object(S2) \wedge Object(S) \wedge Area(A1) \wedge Area(A2) \wedge Area(A3) \wedge In(Stan, A1) \wedge In(S1, A2) \wedge In(S2, A2) \wedge In(S, A3) \wedge \neg Radiated(Stan) \wedge \neg Radioactive(A1)$

The safety constraints can be represented as follows:

$(C1)$ $In(Stan, A3) \wedge \neg(Wearing(Stan, S1) \vee Wearing(Stan, S2)) \rightarrow Radiated(Stan)$

$(C2)$ $In(S1, A1) \vee In(S2, A1) \rightarrow Radioactive(A1)$

Thus $Stan$ gets a lethal dose of radiation if he enters $A3$ without a suit, and area $A1$ becomes radioactive if a suit is brought into it.

We will thus say that $Stan$'s actions are safe if they do not violate $(C1)$ and $(C2)$.

$$(SA) \quad \mathbf{Safe}(a) \leftrightarrow \neg\langle a\rangle^* Radiated(Stan) \wedge \neg\langle a\rangle^* Radioactive(A1)$$

And a safe shutdown state is defined as follows:

$$(SS) \quad \mathbf{Safe}Shutdown \leftrightarrow Shutdown \wedge In(Stan, A1) \wedge \neg Radiated(Stan) \wedge \neg Radioactive(A1)$$

In order to reason about action and change, we will, for present purposes, adopt the extended Strips-approach used in Sipe [2, 5]. The available actions (operators) are as follows. An agent can move between areas, or move to an object in an area, or move away from an object in an area.

- $Move(a, a1, a2)$. Preconditions: $Agent(a) \wedge Area(a1) \wedge Area(a2) \wedge In(a, a1) \wedge \neg\exists x NextTo(a, x)$. Additions: $In(a, a2)$. Deletions: $In(a, a1)$.
- $Move(a, a1, x)$. Preconds: $Agent(a) \wedge Area(a1) \wedge Object(x) \wedge In(a, a1) \wedge In(x, a1)$. Additions: $NextTo(a, x)$.
- $Move(a, x, a1)$. Preconds: $Agent(a) \wedge Object(x) \wedge Area(a1) \wedge In(a, a1) \wedge In(x, a1)$. Deletions: $NextTo(a, x)$.

An agent can also put on or take off a protective suit.

- $Enrobe(a, s)$. Preconds: $Agent(a) \wedge Suit(s) \wedge NextTo(a, s)$. Additions: $Wearing(a, s)$. Deletions: $NextTo(a, s)$.
- $Disrobe(a, s)$. Preconds: $Agent(a) \wedge Suit(s) \wedge Wearing(a, s)$. Additions: $NextTo(a, s)$. Delete list: $Wearing(a, s)$.

Finally, an agent can disable the reactor.

- $Disable(a, s)$. Preconds: $Agent(a) \wedge Switch(s) \wedge NextTo(a, s)$. Additions: $Shutdown$.

In Strips, an action a occurs in a state Σ. If the preconditions of a are true when a occurs, then a succeeds. The resulting state Σ' is obtained by adding the additions of a to Σ and removing the deletions of a from Σ. In Sipe, Σ may, in effect, also contain domain rules; such as the safety constraints $(C1)$ and $(C2)$. Indeed, in the present example we also need the following domain rules to describe a ramification of the move-area action:

$$(C3) \quad In(a, a1) \wedge Wearing(a, s) \rightarrow In(s, a1)$$
$$(C4) \quad Object(o) \wedge In(o, a1) \wedge a1 \neq a2 \rightarrow \neg In(o, a2)$$

Thus, if an agent is wearing a safety suit when it moves from one area to another, then the suit should go with the agent.[2] The presence of domain rules complicates the update process; as it is necessary, after the earlier revisions, to add the consequents of any domain rules that are applicable, and then delete any literals which were in Σ and whose complements have been added by the domain rules. We will refer to this process as a "SIPE update".

Let us assume that *Stan* produces the following non-linear plan:

$(\pi)\quad Move(Stan, A1, A2); (\pi_1 \cup \pi_2); Move(Stan, A2, A1)$

where

$\pi_1 = Move(Stan, A2, S1); Enrobe(Stan, S1);$
$Move(Stan, A2, A3); Move(Stan, A3, S);$
$Disable(Stan, S); Move(Stan, S, A3);$
$Move(Stan, A3, A2); Disrobe(Stan, S1);$
$Move(Stan, S1, A2)$

$\pi_2 = Move(Stan, A2, S2); Enrobe(Stan, S2);$
$Move(Stan, A2, A3); Move(Stan, A3, S);$
$Disable(Stan, S); Move(Stan, S, A3);$
$Move(Stan, A3, A2); Disrobe(Stan, S2);$
$Move(Stan, S2, A2)$

Our task is to formalise this plan and to show that it is safe; that is, that it satisfies (SA) and (SS). We can represent an action a in $\mathcal{SL}$ as the sentence $\Sigma \rightarrow [a]\Sigma'$; where Σ is a conjunction of sentences of $\mathcal{SL}$ which describes the state in which a occurs and which implies the preconditions of a, and Σ' is the appropriate SIPE-update of Σ. We will not attempt to formalise the inference process which produces the SIPE-update. We will thus assume, $(SIPE)$, that for each state Σ which is produced by the plan and which implies the preconditions of an action a in the plan, we have the sentence $\Sigma \rightarrow [a]\Sigma'$; in this example it is clear what Σ' is in each case. Given Σ, a can thus be applied, $(SIPE)$ and (MP), to give $[a]\Sigma'$. For conciseness, we will abbreviate $\Sigma \rightarrow [a]\Sigma'$ to $\Sigma[a]\Sigma'$. Our task is thus to prove that

$(SAS)\quad \mathbf{Safe}(\pi) \wedge \Sigma[\pi]\Sigma' \wedge (\Sigma' \rightarrow \mathbf{Safe}Shutdown)$

where Σ is the initial state description $(Init)$ conjoined with the domain rules $(C1) - (C4)$.

Here is an outline of the proof. The plan π in effect consists of a plan involving suit $S1$ and a plan involving suit $S2$:

[2]For convenience we have occasionally used first-order formulae in the prepresentation of the problem. These are easily translated into suitable, if long-winded, sentences of $\mathcal{SL}$.

$\pi_{S1} = Move(Stan, A1, A2); \pi_1; Move(Stan, A2, A1)$
$\pi_{S2} = Move(Stan, A1, A2); \pi_2; Move(Stan, A2, A1)$

It is sufficient to prove that π_{S1} and π_{S2} both satisfy (SAS); as it then follows from the theorem (SA1) and the theorem:

$(SA4) \quad [a]\phi \wedge [b]\phi \rightarrow [a \cup b]\phi$

that π satisfies (SAS). As the two plans are similar, we will outline the proof of the case for π_{S1} only.

We have the following action-application rule:

(App) From Σ and $\Sigma[a]\Sigma'$ infer $[a]\Sigma'$

And, by means of (MONA), (A3), (A8) and propositional reasoning, we can derive the following action-sequence rule:

(Seq) From $\Sigma[a]\Sigma'$ and $\Sigma'[b]\Sigma''$ infer $\Sigma[a;b]\Sigma''$.

In order to show that $\mathbf{Safe}(\pi_{S1})$, we can use (MONA), (App), (Seq), $(SIPE)$ and Σ to successively show that:

$[Move(Stan, A1, A2)](\neg Radiated(Stan) \wedge \neg Radioactive(A1))$
$[Move(Stan, A1, A2); Move(Stan, A2, S1)](\neg Radiated(Stan) \wedge \neg Radioactive(A1))$
...
$[\pi_{S1}](\neg Radiated(Stan) \wedge \neg Radioactive(A1))$.

Then, by (MONA), (A5)-(A8) and propositional reasoning, we can obtain the desired conclusion.

$\neg\langle\pi_{S1}\rangle^* Radiated(Stan) \wedge \neg\langle\pi_{S1}\rangle^* Radiated(A1)$.

To complete the proof for π_{S1}, we can repeatedly use (Seq), and $(SIPE)$ to show that in $\Sigma[\pi_{S1}]\Sigma'$, Σ' is $\Sigma \wedge Shutdown$. And so, by propositional reasoning and (SS), we have $\Sigma' \rightarrow \mathbf{Safe}Shutdown$ as required.

5 Extensions of SL

Imposing various conditions on $\mathcal{SL}$-models results in different safety logics which include **SL**. Some examples are given in this section.

5.1 Stable disastrous states

An $\mathcal{SL}$-model $M = \langle W, \{R^a\}_{a \in PA}, D, V\rangle$ has stable disastrous states if D satisfies the condition $\forall w \forall w' (D(w) = D(w'))$. The corresponding logic is obtained by adding the following axioms to **SL**.

(Ds1) $\mathbf{Dis}\phi \leftrightarrow [a]\mathbf{Dis}\phi$
(Ds2) $\mathbf{Dan}\phi \leftrightarrow [a]\mathbf{Dan}\phi$

(Ds1) and (Ds2) state that disastrous (dangerous) states are not changed no matter what action is taken. These yield the following extra properties.

(SD) $\mathbf{Safe}\phi \leftrightarrow [a]\mathbf{Safe}\phi$

5.2 Locally stable disastrous states

The condition is: $\forall w \forall w' \forall a (w' \in R^a w \Rightarrow D(w') \subseteq D(w))$, and the corresponding axiom is $\langle a \rangle \mathbf{Dan}(b) \rightarrow \mathbf{Dan}(a; b)$, giving the theorem $\mathbf{Safe}(a; b) \rightarrow [a]\mathbf{Safe}(b)$.

The condition means that there would be no more new disasters which can be created by taking any action. Therefore, the corresponding axiom says that if an action $a; b$ is safe (with respect to the current situation), then the the action b always is safe with respect to the situations after taking action a.

A typical example for this axiom is: if, having drunk alcohol, it is dangerous to drive, then it is dangerous to drink alcohol and then drive.

5.3 Logical inconsistency is the only disaster

Adding the condition $\forall w (D(w) = \emptyset)$ to $\mathcal{SL}$-models gives the corresponding axiom $\mathbf{Dis}\phi \rightarrow (\phi \leftrightarrow \bot)$, which states that logical inconsistency is the only disaster.

5.4 Disasters are other-world oriented

Adding the condition $\forall w (w \notin D(w))$ to $\mathcal{SL}$-models gives the corresponding axiom $\mathbf{Dis}\phi \rightarrow \neg\phi$, which states that disasters occur only in other worlds.

5.5 Dangerous world system

A dangerous world system is one in which all actions can be taken are dangerous. The condition is $\forall w \forall a (R^a w \neq \emptyset \Rightarrow w \in D(w))$, and the corresponding axiom is: $\neg[a]\bot \rightarrow \mathbf{Dan}(a)$.

5.6 Miserable world system

A miserable world system is one in which all worlds are disastrous. The condition is $\forall w(D(w) = W)$, and the corresponding axiom is $\mathbf{Dis}\top$.

6 Normative Safety

In the introduction we distinguished between absolute safety and normative safety, and have subsequently concentrated on formalising absolute safety. In a companion paper [1] we extend the formal machinery developed here to give a theory of normative safety, and the remainder of this section gives an indication of how this is done. The intuitive idea is that an action (state) is safe if it is not *normally* dangerous; where, for example, an action is not normally dangerous if it does not normally lead to a disastrous state. In order to formalise this, we add an abnormal-action operator to $\mathcal{SL}$. Informally, for action a, $Ab(a)$ states that action a has ended abnormally. For example, if we want to say that a flight always ends abnormally if the plane crashes, we can write:

$[fly](Crash \rightarrow Ab(fly))$,

In order to formalise this, we add an abnormal-action operator to $\mathcal{SL}$. Informally, for action a, $Ab(a)$ states that action a has ended abnormally. For example, if we want to say that a flight always ends abnormally if the plane crashes, we can write:

$[fly](Crash \rightarrow Ab(fly))$,

Formally, $\mathcal{SL}$-models are extended by adding a function $AB : W \rightarrow \mathcal{P}(ACTION)$ which assigns to each world w the set $AB(w)$ of all actions which end abnormally at w. Thus:

$M, w \models Ab(a)$ *iff* $a \in AB(w)$.

The Ab-operator can be used to define a "normal outcome" operator $[\cdot]_N$ as follows:

$$[a]_N\phi \overset{\text{def}}{\iff} [a](\neg Ab(a) \rightarrow \phi)$$

Thus $[a]_N\phi$ states that ϕ holds in all *normal* outcomes of a. Formal semantics can be given for this operator and for the safety-related operators using the idea of the *normal trace* of an action. Informally, $NTrace(a, w)$ is the sub-tree of $Trace(a, w)$ consisting of all branches of $Trace(a, w)$ in which events unfold normally, that is, of all branches along which contain no worlds in which an action ends abnormally. Reasoning in the resulting normative safety logic $\mathbf{SL_N}$ can be illustrated by the flying example discussed in the introduction. Let Θ be the theory:

(1) $\mathbf{Safe_N}(fly) \leftrightarrow [fly]_N \neg Crash$
(2) $\mathbf{Safe}(fly) \leftrightarrow [fly] \neg Crash$
(3) $\langle fly \rangle Crash$
(4) $[fly](Crash \rightarrow Ab(fly))$

Thus, (1) it is normally safe to fly iff the plane doesn't normally crash, (2) it is absolutely safe to fly iff the plane never crashes, (3) it is possible that the plane will crash, and (4) the flight will end abnormally if the plane crashes. Then clearly $\Theta \models \mathbf{Safe_N}(fly) \wedge \neg \mathbf{Safe}(fly)$.

In order to be able to conclude that actions end normally, we define a preference order on models. Intuitively, the preferred models are those in which abnormalities occur as late as possible, that is, the preferred models are those which are chronologically most normal. We then define a preferential relation in the usual way [4]; a set of sentences Θ preferentially entails a sentence ϕ (written $\Theta \mathrel{|\!\approx} \phi$) iff ϕ is true in all preferred models of Θ. In the example we then have $\Theta' \mathrel{|\!\approx} \neg [fly]_N Crash$ as required.

7 Concluding Remarks

In this paper we distinguished between absolute safety and normative safety, and developed a formal theory of absolute safety. We started with the idea of a disastrous state (a state in which some disastrous fact is true) and defined a disastrous action to be an action which always leads to a disastrous state. We then defined dangerous actions to be actions which may lead to disastrous states, and dangerous states to be states in which every possible action is dangerous. Finally, safe actions (states) were defined to be actions (states) which are not dangerous. Using Dynamic Logic as a basis, we developed a formal possible-worlds semantics for our theory of safety, and defined the basic safety logic **SL** and several extensions of it. We then gave an example of reasoning about safety in nuclear power stations, and concluded with an indication of how we are building on this work to produce a formal account of normative safety.

References

[1] John Bell and Zhisheng Huang, Safety Logics II: Normative Safety. Proceedings of ECAI-96, 1996.

[2] Fikes, R., and Nilsson, N., STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving, *Artificial Intelligence* 2, 1971, pp. 189-209.

[3] Harel, D., Dynamic Logic, in: D. Gabbay and F. Guenthner, (eds.), *Handbook of Philosophical Logic*, Vol.II, (D. Reidel publishing company, 1984), 497-604.

[4] Shoham, Y. *Reasoning About Change*, M.I.T. Press, Cambridge, Massachusetts, 1988.

[5] D. Wilkins. *Practical Planning*. Morgan Kaufmann, San Mateo, California, 1988.