

Introducing reactive Kripke semantics and arc accessibility

D. Gabbay

Published online: 20 September 2012
© Springer Science+Business Media B.V. 2012

Abstract Ordinary Kripke models are not reactive. When we evaluate (test/measure) a formula A at a model \mathbf{m} , the model does not react, respond or change while we evaluate. The model is static and unchanged. This paper studies Kripke models which react to the evaluation process and change themselves during the process. The additional device we add to Kripke semantics to make it reactive is to allow the accessibility relation to access itself. Thus the accessibility relation \mathcal{R} of a reactive Kripke model contains not only pairs $(a, b) \in \mathcal{R}$ of possible worlds (b is accessible to a , i.e., there is an accessibility *arc* from a to b) but also pairs of the form $(t, (a, b)) \in \mathcal{R}$, meaning that the arc (a, b) is accessible to t , or even connections of the form $((a, b), (c, d)) \in \mathcal{R}$. This new kind of Kripke semantics allows us to characterise more axiomatic modal logics (with one modality \Box) by a class of reactive frames. There are logics which cannot be characterised by ordinary frames but which can be characterised by reactive frames. We also discuss the manifestation of the ‘reactive’ idea in the context of automata theory, where we allow the automaton to react and change its own definition as it responds to input, and in graph theory, where the graph can change under us as we manipulate it.

An earlier version of this paper was presented in CombLog 04, 28–30 July 2004, and published in the proceedings [10]. See www.cs.math.ist.ut.pt/comblog04/talks.html.

D. Gabbay
Bar Ilan University, Ramat Gan, Israel

D. Gabbay (✉)
King’s College London, London, UK
e-mail: dov.gabbay@kcl.ac.uk, gabbayd@macs.biu.ac.il

D. Gabbay
University of Luxembourg, Luxembourg City, Luxembourg

Keywords Modal logic · Other nonclassical logic · Combined logics · Logic in computer science

Mathematics Subject Classification (2010) 03B45

1 Motivation and background

1.1 The reactive idea

Traditional modal logic uses possible world semantics with accessibility relation R . When we evaluate a formula such as $B = \Box p \wedge \Box^2 q$ in a Kripke model $\mathbf{m} = (S, R, a, h)$ (S is the set of possible worlds, $a \in S$, $R \subseteq S^2$ and h is the assignment) the model \mathbf{m} does not change in the course of evaluation of B . We say the model \mathbf{m} is not *reactive*. It stays the same during the process of evaluation.

To make this point absolutely clear, consider the situation in Fig. 1.¹

To evaluate $a \models \Box^2 q$, we have to check $b \models \Box q$. We can also check another formula at b , say, $b \models \Box p$. In either case the world accessible to b are c and d .

We *do not* say that since $b \models \Box q$ started its evaluation at world a as $a \models \Box^2 q$ and continued to $b \models \Box q$, then the accessible worlds to b are now different. In other words the model does not react to our starting the evaluation of $a \models \Box^2 q$ by changing the accessible worlds at b (for example, see the double arrow in Fig. 1, it indicates that we can disconnect the accessibility of node c to b) and therefore allowing us to see a different set of accessible worlds when we continue the evaluation of $b \models \Box q$.

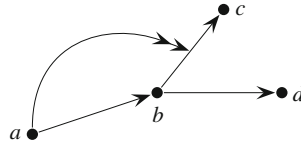
The evaluation of \Box at b *does not* depend, in traditional Kripke semantics, on how we “got” to b .

This paper addresses the case where the semantics does change (or react) under us as we evaluate a formula. This idea makes the evaluation of a wff at a world t dependent on the route leading to t . Thus we get a new kind of semantics, the reactive semantics. This semantics is stronger than ordinary Kripke semantics as there are axiomatic propositional logics with one modality \Box which cannot be characterised by a class of traditional Kripke models but is complete for a class of reactive Kripke models.

The idea of dynamic interaction in logic, classical or non-classical, is not new. Interactive logic has been done in many guises. The most well known example is the game theoretic semantics for the classical quantifiers, and hence for modalities, which has extensive applications in natural language analysis and in theoretical computer science. A related interactive system is a dialogue logic with its vast applications to logic and to argumentation theory. The paradigm of these systems is different from what we are offering here. The interactive paradigm involves two players over a fixed object (a model or a database), who have their respective goals and rules which allow them to manipulate the object and each other in order to win the interaction. Our paradigm is different. As we discuss later in Section 2, we consider systems which react and change because of built-in faults and remedies which get activated while they are used. Applied in this context we can roughly say that we allow the object as

¹Single arrows indicate point-to-point accessibility, double arrows indicate point-to-arc accessibility.

Fig. 1 A reactive Kripke model



well as the rules controlling the players to react to the moves of the players and try to influence them. As the players make their moves the object which they manipulate as well as the game rules governing them can change.²

Meanwhile let us give an example based on Fig. 1, to give the reader some immediate comparison.

Example 1.1 Consider the Kripke model as described in Fig. 1. It has the nodes $S = \{a, b, c, d\}$ with $R = \{(a, b), (b, c), (b, d)\}$. Let the actual world be a .

Let the assignment h for the atom q be $h(q) = \{a, b, d\}$. Thus q is false at world c but holds at worlds a, b and d .

Let us evaluate $a \models_h^1 \Box^2 q$ in a traditional manner (use superscript 1 for traditional)

$$a \models_h^1 \Box^2 q \text{ iff } b \models_h \Box q \text{ iff both } c \models_h^1 q \text{ and } d \models_h^1 q \text{ iff false.}$$

If we allow the model to be reactive (use superscript 2) we get

$$a \models_h^2 \Box^2 q \text{ iff } b \models_h^2 \Box q \text{ iff } d \models_h^2 q \text{ iff true.}$$

under \models^2 , once we passed through node a , c was no longer accessible from b when we came to evaluate $b \models^2 \Box q$.³

The reader might wish to say that we have two models here, one with c accessible to b and one with c not accessible to b .

This view does not work in general. Suppose the reactive model switches $b R c$ on and off each time we make a move. Then whether $b R c$ holds or not depends on how many moves we make to get to b . To illustrate this point, let us modify the model and assume that $a R a$ holds and evaluate $a \models \Box^3 q$.

$$a \models_h^2 \Box^3 q \text{ iff both } (*) a \models_h^2 \Box^2 q \text{ and } (**) b \models_h^2 \Box^2 q \text{ hold.}$$

Since we made only one move, the node c is not accessible to b .

Let us continue:

$$(*) a \models_h^2 \Box^2 q \text{ iff both } (*1) a \models_h^2 \Box q \text{ and } (*2) b \models_h \Box q \text{ hold.}$$

²If the reader insists on the interactive game theoretical point of view then we can regard the reactive paradigm as adding an additional player to the game which is trying to modify and obstruct the two existing players. Whether this point of view is technically correct and/or useful is a matter for investigation towards the end of the paper, after we define our reactive semantics. This additional player may not have any strategy beyond interfering with the game of the other two players. It may be regarded as a faulty environment which changes unpredictably. Michael Gabbay suggested that perhaps we can study a third player whose winning strategy is to deprive the other two players from having a winning strategy.

³Think of node a transmitting a signal along the double arrow and disconnecting the arrow from b to c .

Since we now made two moves, we have that c is again accessible to b and hence $(*2)$ is evaluated at b with $b R c$ active.

On the other hand, $(**)$ is evaluated at b with $b R c$ not active.

We shall see later that this reactive semantics is not reducible to the traditional semantics.

Let us now check how a game theoretical semantics would evaluate $\Box^2 q$ in this model. We have two players **A** and **B**. **A** claims that $\Box^2 q$ holds and **B** claims it does not hold. **B** moves by choosing an accessible point (e.g. b) and challenges **A** to claim that $\Box q$ holds at b . The game goes on and **A** must have a strategy for winning. It is clear that the model is fixed in this game.

We can add a third player **C** who makes changes to the model (and the game rules, if you want) in reaction to the moves made by **A** and **B** and we expect a winning strategy from **A** against both **B** and **C**.

1.2 Examples motivating the reactive idea

Before we continue with more technical material, let us motivate our idea of reactive semantics and consider some case studies.

1.2.1 Airline example

We begin with a very simple and familiar example. Consider Fig. 2.

Figure 2 gives the possible flight routes for the aeroplanes of TUA (Trans Universal Airlines). It is well known that many features of a flight depend on the route. These include the cost of tickets, as well as the right to take passengers at an airport. The right to take passengers at an airport depends on the flight route to that airport and on bilateral agreements between the airlines and governments. Thus, for example, flights to New York originating in London, may take on passengers in London to disembark in New York. However, a flight starting at Paris going to New York through London may not be allowed to pick up passengers in London to go onto New York. It is all a matter of agreements and landing rights. It is quite possible, however, that on the route Prague–Paris–London–New York, the airline is

Fig. 2 Possible flight routes

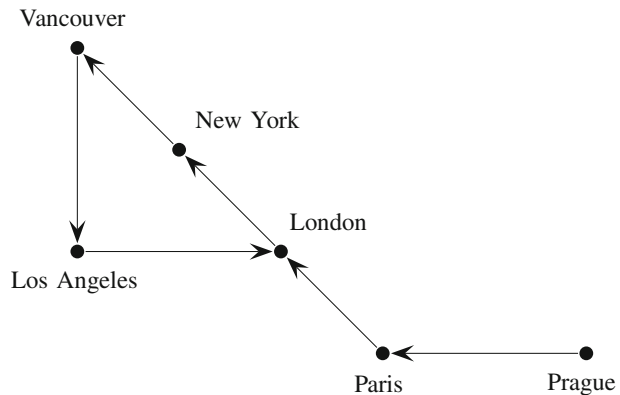
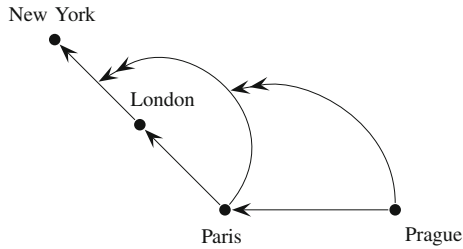


Fig. 3 Flight routes with restrictions



allowed to take passengers in London to disembark in New York. We can describe the above situation in Fig. 3.

The double-headed arrow from Paris to the arc London→New York indicates a cancellation of the ‘passenger’ connection from London to New York. The double-headed arrow from Prague to the double arrow arc emanating from Paris indicates a cancellation of the cancellation.

Figure 3 looks like a typical reactive Kripke model, where we have arcs leading into arcs.

Let us see more examples of this.

1.2.2 Inheritance networks example

This example offers a different point of view of arc semantics, coming from the non-monotonic theory of inheritance networks. Consider Fig. 4.

In Fig. 4, the circular nodes are predicates, such as Fly, Birds, etc. The arrows indicate inheritance, so for example, we have $\forall x(\text{Bird}(x) \rightarrow \text{Fly}(x))$. The arrows with a bar indicate blockage, for example $\forall x(\text{Penguin}(x) \rightarrow \neg\text{Fly}(x))$. The square nodes indicate instantiation, so son of Tweetie is a special penguin.

Figure 4 is the kind of figure one finds in papers on inheritance networks. The figure indicates that Penguins are Birds, that Birds Fly but that Penguins do not Fly. However, special Penguins do Fly and the son of Tweetie, a rare bird, is a special Penguin, and therefore does Fly. The arrow with the bar on it blocks the information from flowing from the Penguin node to the Fly node. The theory of inheritance networks spends a lot of effort on algorithms that allow us to choose between paths in the network so that we can come up with the desired intuitively correct answers. In the case of Fig. 4 we want to get that the son of Tweetie does fly, since we have the

Fig. 4 Inheritance network

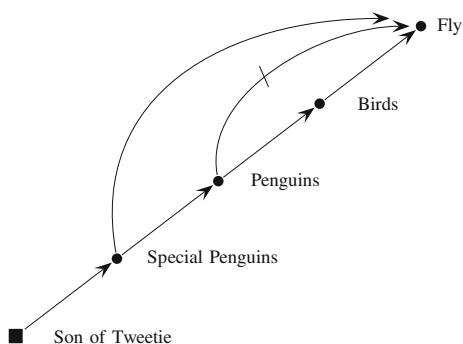
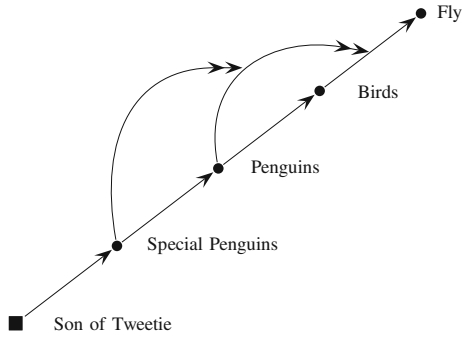


Fig. 5 Figure 4 as a reactive network



most specific information about him. It is not important to us in this paper to take account of how inheritance theory deals with this example. We want to look at the example from our point of view, using our notation, as in Fig. 5.

In Fig. 5 the double headed arrow \rightarrow emanating from Penguins attacks the arrow from Birds to Fly, and the double arrow emanating from Special penguins attacks the double arrow emanating from Penguins and attacking the arrow from Birds to Fly. This is not how inheritance theory would deal with this situation but we are not doing inheritance theory here. Our aim is to motivate our approach and what we need from the inheritance example is just the idea of the algorithmic flow of information during the dynamic evaluation process.⁴

We have already put forward the reactive and dynamic idea of evaluation in earlier papers and lectures (see [18]). A typical example we give is to consider $t \models \Diamond A$. In modal logic this means that there is a possible world s such that we have $s \models A$ we take a more dynamic view of it.

We ask: where is s ? How long does it take to get to it? and how much does it cost to get there?

The reader should recall the way circumscription theory deals with the Tweetie example, see [19, Section 4.1, especially p. 324]. We write

- $Birds(x) \wedge \neg Ab_1(x) \rightarrow Fly(x)$
- $Penguins(x) \rightarrow Birds(x)$
- $Penguins(x) \wedge \neg Ab_2(x) \rightarrow Ab_1(x)$
- $Special\ Penguins(x) \rightarrow Penguins(x)$
- $Special\ Penguins(\text{son of Tweetie})$
- $Special\ Penguins(x) \rightarrow Ab_2(x)$.

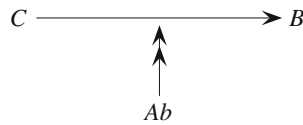
“ $Ab(x)$ ” stands for “ x is abnormal”. If the clause $C(x) \rightarrow B(x)$ represents the arc $C \rightarrow B$ then $C(x) \wedge \neg Ab(x) \rightarrow B(x)$ represents the situation in Fig. 6.

1.2.3 A technical example

It is now time to give a technical example. Consider Fig. 7. This figure displays a past flow of time. The node t is the present moment and a single headed arrow from one

⁴It is our intention to explore whether our idea of double headed arrows cancelling other arrows can simplify inheritance theory algorithms.

Fig. 6 Abnormality as reactivity



node to another, say from s to t , means that t is in the immediate future of s . We use the modality \square to mean ‘always in the immediate past’. Thus the accessibility relation R of Fig. 7 is as follows:

- $tRs, tRb, bRs, sRa, tRt, bRb, sRs$ and aRa .

The double-headed arrows cancel the accessibility relation.

Let us calculate $t \models \square^3 q$ in Fig. 7.

Initial position Starting point is t and all arrows are active.

Step 1 Send double arrow signal from t to all destinations inverting the active/inactive status of all destination arrows. Then go to all accessible worlds (in this case s and b) and evaluate $\square^2 q$ there. If the result is positive 1 and at all nodes, then send ‘success’ back to node t .

Step 2 Evaluate $\square^2 q$ at nodes b and s .

Subcase 2.b Evaluation at b : First we send a double arrow signal from b to all destinations reversing the activation status of these destinations. Thus the single arrow from s to s will be re-activated and we will evaluate $\square q$ at s at the next step 3 (with s accessible to itself). b is not accessible to itself because its arrow has been deactivated by t at step 1.

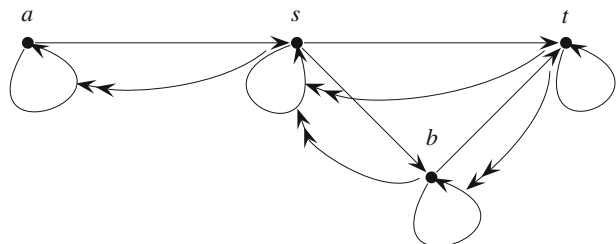
Subcase 2.s Evaluation at s : First we send a double arrow signal to reverse the status arrow from a to a . Then we evaluate $\square^2 q$ at s with s not accessible to itself, since the arrow from s to s was deactivated by t at step 1.

This can go on, but we shall not continue as we trust that the reader has got the idea by now.

Note that if we start at t and evaluate $B = \square^3 q \wedge \square^2 q$, we will get that $\square q$ must be evaluated at s in two ways. One with s accessible to itself (coming from b via $t \models \square^3 q$) and once with s not accessible to itself (coming from t via $t \models \square^2 q$).

Let us now calculate $t \models \diamond^2 q$ in Fig. 7.

Fig. 7 Reactive arrows cancel accessibility



- Initial position Starting point is t and all arrows are active.
- Step 1 Send double arrow signal from t to all destinations inverting the active/inactive status of all destination arrows. Then go to one of the accessible worlds (in this case s or b) and evaluate $\diamond^2 q$ there. If the result is positive 1 at this node, then send 'success' back to node t .
- Step 2 Evaluate $\diamond^2 q$ at nodes b or s .
- Subcase 2.b Evaluation at b : First we send a double arrow signal from b to all destinations reversing the activation status of these destinations. Thus the single arrow from s to s will be re-activated and we will evaluate $\diamond q$ at s at the next step 3 (with s accessible to itself). b is not accessible to itself because its arrow has been deactivated by t at step 1.
- Subcase 2.s Evaluation at s : First we send a double arrow signal to reverse the status arrow from a to a . Then we evaluate $\diamond^2 q$ at s with s not accessible to itself, since the arrow from s to s was deactivated by t at step 1.

This can go on, but we shall not continue as we trust that the reader has got the idea by now. For the case of \diamond we make a non-deterministic choice. The model is not sensitive to whether we come to a point because we are evaluating \square or a \diamond . If we want this kind of sensitivity we can have arrows of the form $\rightarrow\square$ and $\rightarrow\diamond$.

Such distinctions may be desirable in dealing with quantifier games, where changes may be different for the cases of \forall and \exists .

1.2.4 Tax example

Having explained the technical side of our reactive (changing) semantics, let us give some real examples.

House prices in London have gone up a great deal. An average upper middle class family is liable to pay inheritance tax on part of the value of their house (if the house is valued over £500,000, for example, then there is tax liability on £250,000). Some parents solved the problem by giving the house as a gift to their children. If at least one of the parents remains alive for seven years after the transaction, then current rules say that there is no tax. Consider therefore the following scenario:

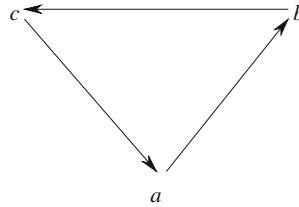
1. current date is April 2004
2. parents gave house as a gift to children in 1996
3. parents continued to live in house as guests of the children

(1)–(3) above imply that (4):

4. if parents both die in March 2004, then no tax is liable.

To continue the story, there were rumours that the tax people were going to change the rules in April 2004, declaring that if parents remain living in the house after it

Fig. 8 Salesman example



was given as a gift, then the gift does not count as such an there is tax liability. The rumours also said that this law is going to apply *retrospectively*.⁵

Thus we have that (5) holds:

- 5. If parents both die on March 2005, then tax is liable.

We assume that (4) still holds even after the new law as we cannot imagine that the UK tax inspector would be opening closed old files and demanding more tax.

The way to represent (4) and (5) is to use two dimensional logic. We write $t \models_s A$ to mean at time t A is true given the point of view proposed or held at time s .

Thus $2005 \models_{2003} \neg(5)$ holds, because from the 2003 laws point of view (before legislation) no tax is liable ((5) says tax is liable). But $2005 \models_{2004} (5)$ also holds, because according to 2004 legislation tax is liable.

So far we have no formal problem and no need for our new semantics, because we can write

- $t \models_t \Box A$ iff for all future $s, s \models_s A$.

In other words we evaluate sentences at time t according to the point of view held at the very same time t .

The problem arises when we want to formalise the following scenario. The parents die in 2003. The lawyer is dealing with the estate. We do not know when he is going to finish. When he submits the paperwork then the tax liability at 2003 is judged according to the time of submission. Now the second index s in $t \models_s A$ behaves like a reactive model as we are evaluating

$$2003 \models_{\text{time lawyer submits}} (4).$$

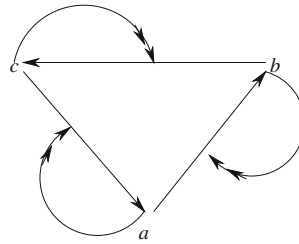
1.2.5 Salesman example

Consider the simple graph of Fig. 8.

A salesman wants to traverse this graph in such a way that he doesn't pass through the same edge twice. Such problems are very common in graph theory. The simplest

⁵Some countries, like Austria, for example, would *never* legislate retrospectively. They regard this as a cultural taboo.

Fig. 9 Salesman example with reactive arrows



way of implementing this restriction is to cancel an edge once it has been used. Figure 9 will do the job.

One cannot always implement the salesman problem in this way. It depends on the graph we deal with. However, it is one more reason for considering arc accessibility.⁶

1.2.6 Resource example

Consider a road system as in Fig. 10.

Assume that whenever we drive through a road we need to pay a fee to pass. Payment must be in cash and it is very expensive. We can withdraw money from cash machines along the way to pay for our passage as follows:

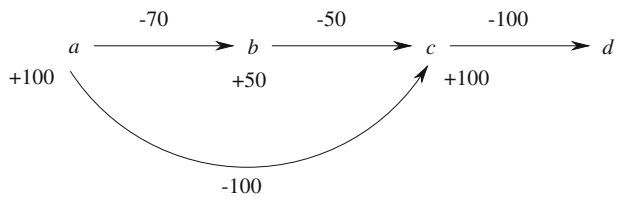
- Point *a* Cash withdrawal \$100 is available.
Road *a* to *b*: costs \$ 70.
- Point *b* Cash withdrawal \$50 available.
Road *b* to *c*: costs \$50
- Point *c* Cash withdrawal \$100 is available.
Road *c* to *d*: costs \$120.

We start with no money at all at point *a*. If we drive from *a* to *c* directly, we can get \$100 from the cash machine at *a*, and this will get us to *c* with no money left and so we cannot get at *c* enough cash to pay passage from *c* to *d*. However, if we pass through *b*, we can withdraw money at *b* and later at *c* and we will have enough to pay for the passage from *c* to *d*.

Figure 10 is essentially an ordinary annotated graph describing the resource situation but the qualitative situation (where money considerations are hidden) can be described in Fig. 11. The upshot of Fig. 11 is that the arc from *a* to *c* sends a signal to cancel the arc from *c* to *d*. This is also the first case where we have double arrows going from arc to arc.

⁶For the salesman example we need accessibility from arcs to arcs. So in Fig. 9, rather than have the double arrow $(b, (a, b))$, we need the double arrow $((a, b), (a, b))$. The arc cancels itself as we go through it.

Fig. 10 Road system



1.2.7 Flow products example

Ordinary products of Kripke frames are defined in a straightforward manner. Given two frames (S_1, R_1) , and (S_2, R_2) , we form the product space $S = S_1 \times S_2$, and define two modalities \Box_1, \Box_2 on pairs as follows:

- $(a, b) \models \Box_1 A$ iff $\forall x(a R_1 x \rightarrow (x, b) \models A)$
- $(a, b) \models \Box_2 A$ iff $\forall y(b R_2 y \rightarrow (a, y) \models A)$.

Figure 12a shows the configuration for the case of $S_1 = S_2 = N$, the set of natural numbers.

\Box_1 shifts the x coordinate, leaving b fixed and \Box_2 shifts the y coordinate, leaving a fixed.

We have, for example, among other things that $\Box_1 \Box_2 = \Box_2 \Box_1$. Products spaces are used whenever we deal with two independent modal or temporal aspects. This is a very active area of many dimensional modal logics, also related to classical predicate logic with a fixed number of variables. See our book [16].

The authors introduce in [18] the concept of flow products. Imagine the x axis is space (measured in kilometers) and the y axis is time measured in hours. Any shift in space will necessarily cause a shift in time. Assuming speed of 1 mile per hour, we get

- $(a, b) \models \Box_1 A$ iff $\forall u((a + u, b + u) \models A)$
- $(a, b) \models \Box_2 A$ iff $\forall u((a, b + u) \models A)$.

When we move in space time shifts.

We can view this as a reactive model (see Fig. 12b). Imagine for any point (a, b) the following double arrows exist as we move from a to $a + 1$, the connection from b to $b + 1$ is switched off. Therefore we get

$$(a, b) \models \Box_1 A$$

in the reactive model iff $\forall u((a + u, b) \models A)$ in the new model after firing the reaction of moving through $a + 1, a + 2, \dots, a + u$, which disconnects b from $b + 1$,

Fig. 11 Reactivity due to resources

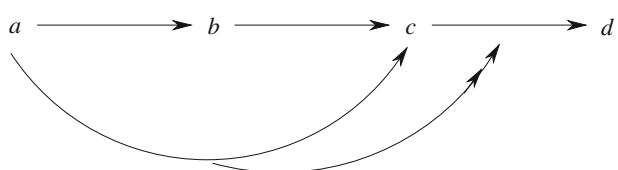
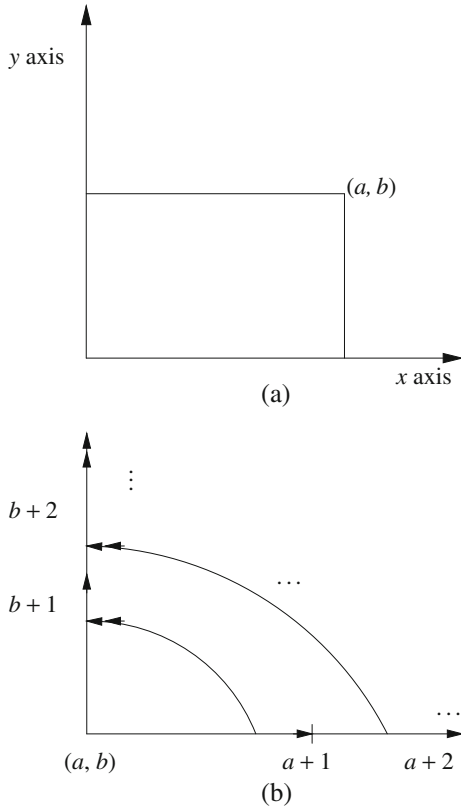


Fig. 12 Flow product: We are at point (a, b) and we move in the x axis.



$b + 2, \dots, b + u$). Hence $(a + u, b) \models \Box_2 B$ is now connected to $(a + u + 1, b + u + 1) \models B$, i.e., it is as if we are at the point $(a + u, b + u)$.

Of course a general flow product model will require a shift function $f(a, b, x)$ on the x -coordinate, satisfying

$$(a, b) \models \Box_1 A \text{ iff } \forall x(aR_1x \rightarrow (x, f(a, b, x)) \models A).$$

$f(a, b, x)$ tells us how much the y coordinate shifts. f must satisfy some additivity properties, such as

- $aR_1x \rightarrow bR_2f(a, b, x)$
- $aR_1x_1 \wedge x_1R_1x_2 \wedge aR_1x_2 \rightarrow f(a, b, x_2) = f(x_1, f(a, b, x_1), x_2)$.

1.3 Plan of this paper

The next section, Section 2 describes the reactive paradigm in general and proceeds to give some specific examples from different application areas. These areas are studied in separate research papers. We also discuss several options for defining reactivity. The rest of the paper should give some technical results in the area of modal logic. There is a lot to do here and the main bulk of the results is postponed to further papers, such as [1, 5, 12–14]. In order to give some formal results in this paper,

we want to show that the class of switch reactive Kripke models can characterise more syntactic modal axiomatic systems than the class of ordinary Kripke models. This is done in Section 4. To prepare the ground for this result we need to devote Section 3 first to hyper-modalities, being a related but not an identical concept to the concept of reactive modality. Section 5 deals with non-deterministic reactive Kripke models and Section 6 compares with fibring. Section 7 concludes with two examples showing the two concepts (hyper-modality and reactive modality) are different.

We now have two options of how to continue this paper:

1.3.1 Option 1

Get more mathematical results for reactive modality. For example, characterise some modal axioms by conditions on models (on R) etc. Investigate complexity, introduce proof theory, etc, etc. See references.

1.3.2 Option 2

Continue with some general conceptual analysis of the concept of reactivity and comparison with relevant literature.

We follow the second option.

Section 5 introduces non-deterministic reactive Kripke models, Section 6 discusses connections with fibring logics and Section 7 discusses how to introduce dedicated reactivity operators.

Each of these topics has enough problems to investigate to merit a separate research paper.

We continue with four appendices which give further results and comparison with the literature.

2 The reactive paradigm in general

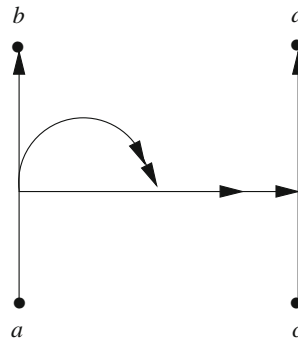
Consider a system \mathcal{S} containing components and some internal connections and procedures governing the interactive behaviour of these components in response to external inputs. We can think for example of \mathcal{S} as a graph (to be traversed or manipulated) or as an automaton which responds to inputs and changes states, or as a Kripke model in which a formula is evaluated, or as a bureaucratic system responding to an application form or even as more familiar daily objects such as a washing machine, a television set or a car.

We can represent such a system as a network containing possibly labelled vertices representing the components and arrows between the vertices representing possible “control flows” within the system of whatever the system internal processes do.

Figure 13 is a typical situation we want to consider.

The arrows from a to b and from c to d indicate that internally whatever the system does in response to input or command, there is a possible path where component a passes the “flow” to component b and similarly maybe in another part of the system, there is a connection between c and d .

So far we have nothing more than a possible network graphic representation of some system.

Fig. 13 Fault example

Now comes the reactive idea. Consider the possibility that the system develops faults due to overuse or stress. The double arrow from the arc (a, b) to the arc (c, d) (which we denote by $((a, b), (c, d))$) indicates how the fault may develop. When we pass from a to b we put internal pressure on the connection from c to d . If significant pressure builds up on the connection from c to d it may fail. For example everyone who drove a car long distance during a very hot day knows that the engine may overheat. This is an example of a fault. In general these faults are predictable and are predetermined by the construction of the system. They are represented in Fig. 13 by the double arrows.

In fact, Fig. 13 also indicates a remedy to this pressure, this is a double arrow from the arc (a, b) to the double arrow $((a, b), (c, d))$, namely $((a, b), ((a, b), (c, d)))$.

The double arrows represent known weaknesses or remedies in the system. We can look at the double arrows as indicating the ways in which the system reacts and adjusts itself under pressure. The simplest model is a once only bridge in Fig. 14.

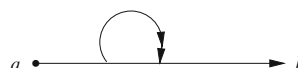
When we cross the bridge from a to b , a signal (representing stress) is sent to the bridge to collapse and so after we cross the bridge the arc from a to b is canceled.

The above is the basic intuition behind the idea of reactivity. Think of it as “fault” reactivity, or better it is “fault-remedy” reactivity.

We now elaborate more about this concept to distinguish it from other somewhat similar concepts.

The “fault” reactivity should be distinguished from the idea of including dynamic metalevel operators in the input. To give example of such metalevel operators, consider the possibility that the system may have several modes of operations say *mode 1*, ..., *mode n*.

The input may contain instructions of the form J_i (jump to mode i) which tells the system how to operate on the input. This is a metalevel instruction inducing a change in the system. It is different from “fault” reactivity, though the latter can be case-by-case represented in it. If we replace double arrows to (c, d) by the metalevel instruction “disconnect(c, d)”, we might be able to simulate the fault induced by an input by adding (interleaving) some “disconnect” instructions to the input to form a new input.

Fig. 14 Once only bridge

This option will be discussed later in the paper on a case-by-case analysis. It is not reactivity but “fibring” of metalevel instructions giving the illusion of reactivity.

The notion of fault reactivity should also be distinguished from the notion of “sabotage”, independently introduced by J. van Benthem [21] and further developed by P. Rohde [20]. The notion is similar; we envisage a saboteur disconnecting components within the system. There are both conceptual and technical differences between the two notions (“fault” reactivity and “sabotage” reactivity) and this will be analysed in detail in Appendix D. For the moment we note that faults and remedies are built into the system while sabotage is not and also sabotage is presented as a metalevel connective and fault is not. For more detail, see Appendix D.⁷

It turns out that as we apply this idea in various application areas where different non-reactive systems are used we can get new kinds of systems, the reactive systems, which can have better applicability in each respective area.

We give seven immediate examples:

1. Ordinary Kripke models become reactive Kripke models which can change during the process of evaluation, affording a wider class of models for modal logic.
2. Ordinary non-deterministic automata become reactive automata whose transition table changes every time they make a transition, see [5]. Although it is shown in [5] that every reactive automaton is equivalent to another ordinary non-reactive automaton, the reactive automata can perform tasks (recognise inputs) with much less states and represent substantial savings in computational costs!
3. An ordinary graph becomes a reactive graph where there are also arrows from edges to edges, see [17]. This area is very promising and is now under investigation.
4. Reactive grammars and rewrite systems [2]. For reactive rewrite systems we get new hierarchies of languages. Again this area is actively investigated.
5. A proof system becomes a reactive proof system which changes as we advance in the proof process, see [2, 8].
6. Using Reactive arrows we can solve Contrary to duty paradoxes in Deontic logic, as well as model delegation in Talmudic logic , see [1, 12, 15].
7. In argumentation networks reactive arrows become higher level attacks and this allows us to generalise the notions of various networks, see [3, 11, 15].

In each case a new class of models/graphs/automata/proof theory is introduced and some natural questions can be asked.

The following are some sample questions:

1. What is the expressive power/complexity of the new class and what can it do or not do?
2. What do traditional investigations yield when applied to this new class? (E.g. completeness theorems, cut elimination, correspondence theory, axioms, hierarchies of automata, theorems about cycles or classes of graphs, etc. etc.)

⁷Some sabotage may be non-deterministic like water damage running through the system. This can be modelled as a non-deterministic reactive system. The possible options for water damage can be foreseen!

3. How does this concept relate to other dynamic/change concepts already existing in the literature?
4. Can the new models be reduced to known models via traditional interpretations? What is the edge/advantage we have by using the new reactive models?
5. Comparison and evaluation of the options for representing reactivity in the object level vs. metalevel.

Let us give some concrete examples, which will help us get a feel for the notion of reactivity.

Example 2.1 (Modal logic) Here we take the reactive Kripke models of the previous sections. The system is the evaluation process (for a given formula). The reactive parameter is a Kripke model with a point of evaluation t . The environment (or the faults in the system) is realised by the double arrows \twoheadrightarrow , which change the accessibility relation every time we make a move (i.e., every time we continue with the evaluation).

This example generalises slightly, giving it a familiar everyday meaning. Think of a network of nodes representing various components of a system. The arrows represent connections between components and a path through this graph following the arrows represents a way in which the system can be used. Imagine that the system is prone to faults and failures. So the use of one component may send a (double arrow) signal to other components and weaken or influence them.

We know this to be true for many real life systems. So the reactivity of a network is just a measure of the ways it can fail and disappoint you!

Example 2.2 (Quantifier games) Recall the basic game semantics for the quantifiers. Given a classical model \mathbf{m} and a formula, say $\Psi = \forall x \forall y \exists z (\varphi(x, y, z))$ with $\varphi(x, y, z)$ quantifier free, we play a game over the model between two players, **A** (claiming that Ψ holds) and player **B** claiming that Ψ does not hold. At step 0, **A** puts forward Ψ . At step 1 **B** challenges by choosing a in the domain of \mathbf{m} , and continues to choose an element b in the domain of \mathbf{m} . It is the task of player **A** to supply a c such that $\varphi(a, b, c)$ holds. Player **A** has a winning strategy iff he has a function f such that for any a, b the element $c = f(a, b)$ is such that $\mathbf{m} \models \varphi(a, b, f(a, b))$.

In this case the moves are the choices of elements. The reactive environment player **E** has a tinkering function $\tau : (a, \mathbf{m}) \mapsto \mathbf{m}_a^\tau$ which changes any model \mathbf{m} and an element a in the domain of \mathbf{m} into a new model \mathbf{m}_a with the same domain.

Thus in order to win, **A** has to have a winning function $\lambda \tau \lambda x y f_\tau(x, y)$ such that

$$\forall \tau \forall a \forall b \mathbf{m}_{a,b}^\tau \models \varphi(a, b, f_\tau(a, b)).$$

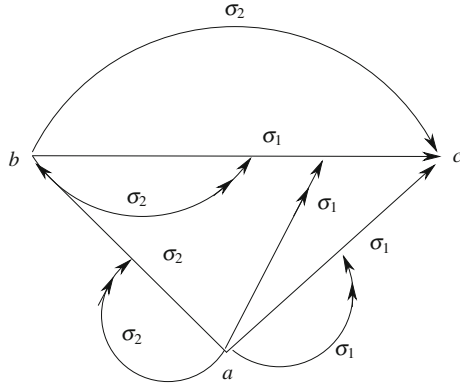
So as we can see, this definition covers the modal logic case, if viewed through its translation to classical logic.

For consider a reactive Kripke model \mathbf{m} with actual world a . Consider the evaluation $a \models \Box^2 \Diamond q$. This is translated into classical logic as

$$\Psi = \forall x \forall y \exists z (aRx \wedge xRy \rightarrow yRz \wedge Q(z)).$$

The function τ for changing the model is governed by the double arrows \twoheadrightarrow and we clearly have a special case of the quantifier games tinkering.

Fig. 15 Reactive automaton



We get in this case the formula Ψ' :

$$\Psi' = \forall x \forall y \exists z [a R x \wedge x R_a y \rightarrow y R_{a,x} z \wedge Q(z)]$$

where R_i denotes the new accessibility relation obtained from R after implementing the double arrows emanating from it.

Example 2.3 (Automata) A non-deterministic automaton $\mathcal{A} = (S, M, a, F, \Sigma)$ is characterised by a set of states S , an initial state $a \in S$, a set $F \subseteq S$ of final states and an alphabet Σ . M is a function giving for each state $t \in S$ and a letter $\sigma \in \Sigma$ a new set of states $M(t, \sigma) \subseteq S$, which are the states that the automaton \mathcal{A} can non-deterministically move to.

Another way to view the automaton is as a multi-modal Kripke model of the form (S, R_σ, a, F) , where R_σ for $\sigma \in \Sigma$ is defined by $x R_\sigma y$ iff $y \in M(x, \sigma)$.

A word of the form $(\sigma_1, \dots, \sigma_n)$ is said to be recognisable by the automaton \mathcal{A} iff there exists a sequence of states x_1, \dots, x_n such that $a R_{\sigma_1} x_1 \wedge \dots \wedge x_{n-1} R_{\sigma_n} x_n$ holds and $x_n \in F$. If the atom q is assigned the set F , then $(\sigma_1, \dots, \sigma_n)$ is recognisable by \mathcal{A} iff $\mathcal{A} \models \diamond_{\sigma_1}, \dots, \diamond_{\sigma_n} q$, where \diamond_σ corresponds to $R_\sigma, \sigma \in \Sigma$.

Thus naturally we can define a reactive automaton by adding double arrows \rightarrow_σ for every $\sigma \in \Sigma$ in the places we want. Figure 15 is an example of such an automaton. Let $\Sigma = \{\sigma_1, \sigma_2\}$, $S = \{a, b, c\}$ and $F = \{c\}$.

We have

$$\begin{aligned} R_{\sigma_1} &= \{(a, c), (b, c)\} \\ R_{\sigma_2} &= \{(a, b), (b, c)\} \end{aligned}$$

Figure 15 indicates the available double arrows. For example, the figure shows that as we move through node a in response to the letter σ_1 we disconnect (a, c) and (b, c) from the R_{σ_1} relation. If we move through node a in response to the letter σ_2 then we disconnect (a, b) from R_{σ_2} .

It is interesting to note that according to this figure, if we move out of state b in response to the letter σ_2 then we disconnect the connection (b, c) in the R_{σ_1} relation but not in the R_{σ_2} relation.

So there can be an interplay between the modalities here.

The reader should note that we can get a new hierarchy of automata here by taking the usual hierarchy and making it reactive. We shall pursue this idea in a subsequent papers, [5, 14].

The next remark, however, lists some possible options for creating reactivity.

Remark 2.4 (Options for the reactive property) We saw that the basic idea of a reactive action is to change the model every time a move is made. To explain our options on how to change the model consider Fig. 16.

From the point t arrows and double arrows emanate. Some may be active and some may not be. Also observe that we have a double arrow emanating from an arrow. Let us list what we have:

1. $t \rightarrow b$
2. $t \Rightarrow (t \rightarrow b)$
3. $a \rightarrow b$
4. $(a \rightarrow b) \Rightarrow (t \rightarrow b)$.

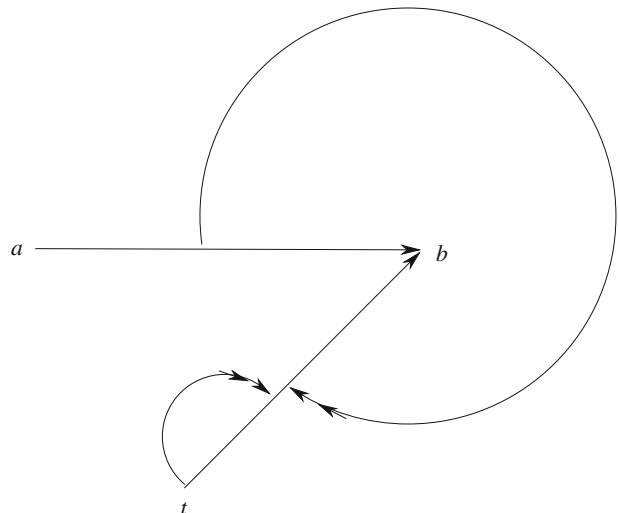
Our first list of options relates to what kind of arrows we allow. Do we want only arrows emanating from a point or do we also allow arrows emanating from arcs?

Items 1–3 above emanate from points and item 4 emanates from arcs.

The second list of options has to do with how we use the arrows to change the model as we pass through a point or an arc.

The first possibility is the switch-like use of the arrows. At any given moment some of the arrows are active (on) and some are dormant (off). When we pass through a point or an arc a signal is sent along the arrows emanating from the said point or arc and reverses the status of the target arrows at the destination. This is a simple switch action. In general, the reactivity can be more intelligent. The second possibility is to allow each node and arc from which arrows emanate to decide, depending on the state (on off) of the target arrows, which of the target arrows to switch on and which

Fig. 16 Reactive arrows change the model



to switch off. Care must be taken to ensure that this decision process is of the same complexity as the original non-reactive model. So for example in the case of reactive finite state automaton of Example 2.3, the decision of each node which double arrows to activate should be done by another automaton.

Consider Fig. 17.

Suppose we start with all arcs being switched on at the starting point a . As we move out of a to b , a switch behaviour will switch off all arcs 1, 2, 3, ... If we move through node a again, these arcs will be switched on again.

A more intelligent option might switch them on selectively one at a time, which each passage through the node a . This would allow us to use them as markers emulating aspects of a stack. Note that the idea is intuitively sound, and is not just a technicality. It makes sense to give nodes some intelligence to decide how to react, based on the situation it ‘sees’.

The above options were deterministic. The most general option was to attach an automaton at each exit point of arrows (of any kind) to decide what to switch on and what to switch off. It is also possible to make these decisions non-deterministic or probabilistic. Consider Fig. 16. We can make all double arrows in this figure non-deterministic. So as we pass, for example, from node a to node b the double arrow to the arc $t \rightarrow b$ (namely $(a \rightarrow b) \Rightarrow (t \rightarrow b)$), if active, may or may not (non-deterministically) send a signal.

Similarly we may attach probabilities to such connection, say 0.7, and so with 0.7 probability a signal will be sent. We shall elaborate more about this option in a subsequent paper [14].

Example 2.5 (Intelligent switches) Previous examples used on and off switches. The present example uses an intelligent switching system. In fact, we build on Example 2.3 (automata) and show in the present example how to simulate a stack automaton. We make sure the intelligent switching process is also done by a finite automaton. Consider the model described in Fig. 18.

This model has two relations R_{σ_1} and R_{σ_2} . It corresponds to an automaton with alphabet $\{q_1^+, q_1^-, q_2^+, q_2^-, t, a\}$ and stack letter α . The initial state is a and the terminal state is t . It is designed to recognise the words of the form $\sigma_i^m \sigma_j^m, i \neq j, m \geq 1$.

The starting state is a . Upon seeing σ_i the automaton moves to state q_i^+ and writes α in the stack. It continues to write α as long as it sees σ_i . When it sees $\sigma_j, j \neq i$ it moves to q_j^- and starts deleting from the stack.

The double arrows are intelligent. The ones emanating from node a or node q^+ activate the first highest non-active arrow or double arrow at α . The ones emanating

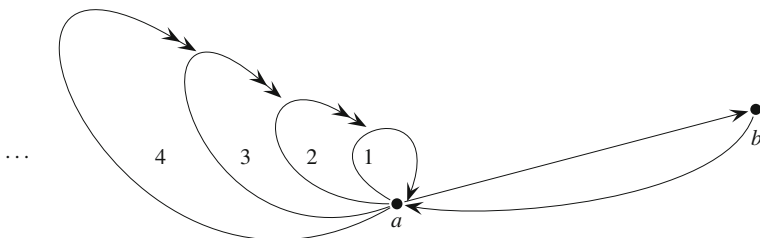


Fig. 17 Complex higher level switch arrows

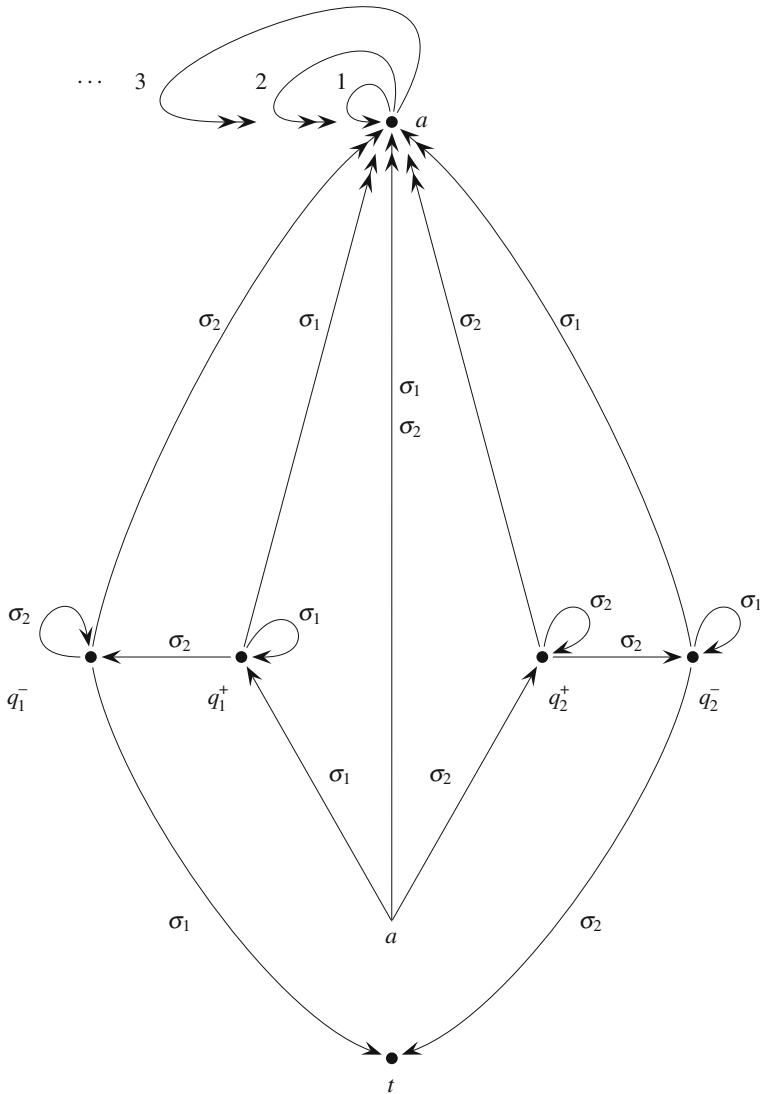


Fig. 18 Stack automaton

from a q^- deactivate the highest active double arrow from α . If q^- sees that all arrows at α are not active it starts to activate arrow 2 and higher. So if arrow 1 is active then q^- deactivates the top arrow and if arrow 1 is not active then q^- activates the first non active arrow above arrow 1. We need to assume that only a finite number of arrows and double arrows from α are active at any given time.

Table 1 describes the moves of the automaton of Fig. 18. Note that only arrows at α may switch. The initial position is state a with all arrows at α not active. The terminal state is t .

So let us simulate an input computation.

Starting state All double arrows and arrows are active except those at α which are not active.

Input step 1 We can assume without loss of generality that we get σ_1 . So we move along R_{σ_1} from node a to node q_1^+ . This move activates arrow 1 at α . The more σ_1 we see the more double arrows at α are activated in sequence. So if we see a total of m σ_1 s, i.e., σ_1^m we get that arrows $1, \dots, m$ are active. The first time we see σ_2 we move from q_1^+ to q_1^- along R_{σ_2} . There is no R_{σ_2} double arrow from q_1^+ to a .

If we continue to get σ_2 at q_1^- we cancel an arrow at α . The minute we get σ_1 again or the input finishes we move to a terminal state t . If the number of σ_2 is equal to σ_1 , we end up stopping at t with no α arrow active. If the number of σ_2 is less than σ_1 , we end up stopping with some α arrows active. If the number of σ_2 is larger than σ_1 we are faced with a situation where the q_1^- automaton sees no arrow connections at α at all and needs to decide what to do. We can tell it to activate arrow 2, leaving arrow 1 not active and to continue to activate arrows 3, 4, ... as long as the input is σ_2 . If the input is empty or q_1^- sees σ_1 again then it stops.

Example 2.6 (Options for interpreting necessity when double arrows emanate from arcs) Consider Fig. 19.

In this figure we have two double arrows emanating from arcs. We ask: What semantic meaning can we give to \square ?

We want to evaluate

$$a \models \square\square\perp$$

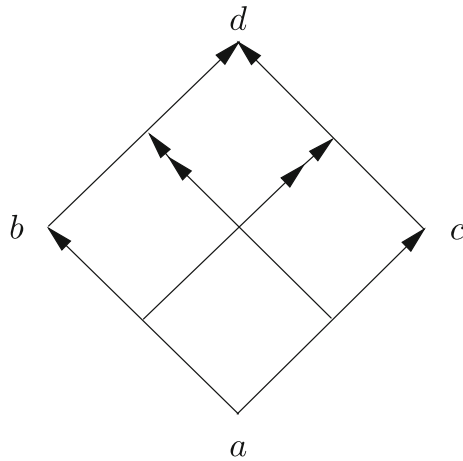
How do we go about it? Do we go to every accessible point one-by-one and check whether $\square\perp$ holds? Or do we go to all points *simultaneously*?

When double arrows emanate from points, there is no difference in the mode of operation, but when they emanate also from arcs, then there is a difference

Table 1 Moves of the automaton of Fig. 18

State	Stack at α	Input letter	Reaction
a	All arrows not active	σ_i	Move to state q_i^+ and activate arrow 1 at α
a	Some arrows active	Any	Do not care. Case will not arise
q_i^+	Arrows $1, \dots, m$ are active $m \geq 1$	σ_i	Stay at q_i^+ and activate arrow $m + 1$
q_i^+	Arrows $1, \dots, m$ are active	$\sigma_j, j \neq i$	Move to q_i^-
q_i^+	Not the above	Any	Don't care. Case will not arise
q_i^-	Arrows $1, \dots, m$ are active $m \geq 1$	$\sigma_j, j \neq i$	Stay at q_i^- and deactivate arrow m .
q_i^-	Arrows $1, \dots, m$ are active, $m \geq 1$	σ_i or no input	Move to t and deactivate arrow m
q_i^-	All arrows at α are not active or arrows $2, \dots, m$ are active $m \geq 2$ and arrow 1 not active	No input or input σ_i	Move to t , activate arrow $m + 1$. If no arrow at α is active then activate arrow 2.
q_i^-	Same as previous	$\sigma_j, j \neq i$	Remain at q_i^- , activate arrow $m + 1$ or if no arrows at α are active then activate arrow 2.
q_i^-	Different from previous	Any	Don't care. Case will not arise.
t	Any	No input. Terminal position	No reaction. Terminal position.

Fig. 19 Two reactive arrows from arcs



Case 1 (Separate evaluation of \Box) When we move from a to c , the arc (c, d) is still connected and so we have that $c \models \neg\Box\perp$. Similarly when we move from a to b , the arc (b, d) is still connected and so we have that $b \models \neg\Box\perp$.

Therefore, according to the separate evaluation of \Box , we get that $a \models \Box\neg\Box\perp$.

Case 2 (Simultaneous evaluation of \Box) If we move to b and to c simultaneously and evaluate at b and at c both arcs (b, d) and (c, d) are disconnected and so we have $b \models \Box\perp$ and $c \models \Box\perp$ and hence $a \models \Box\Box\perp$.

It makes more sense to adopt the separate evaluation of \Box because of the traditional connection of \Box with \Diamond , namely $\Box = \neg\Diamond\neg$ we have:

$$a \models \Diamond A \text{ iff for some accessible point } s, \text{ when we move to } s \text{ we have } s \models A.$$

We also have $a \models \Box A$ iff for all accessible points s when we move to s we have $s \models A$.

To preserve the duality $\Diamond = \neg\Box\neg$ we must adopt the separate evaluation of \Box .

3 Connection with hyper-modalities

In [9], we introduced hyper-modal logics. We showed that such modalities cannot always be characterised by a class of Kripke frames. However, there is hope that our new reactive semantics might provide frames for some of these modalities. This section and the next study the connection.

A hyper-modality \Box is a modality which changes its nature depending on where in a formula it appears. So for example, in the formula $B = \Box^3 q \wedge \Box^2 q$, the inner modalities may not have the same meaning as the outer ones. To illustrate this point consider the arrangement of Fig. 20.

t is *now*, s is in the past of t and so are a and b . We consider two past operators

- $H_{\mathbf{K}}A$ saying A was true at all the immediately past moments of time
- and
- $H_{\mathbf{T}}A$ saying $A \wedge H_{\mathbf{K}}A$

We let \Box alternate between $H_{\mathbf{K}}$ and $H_{\mathbf{T}}$, starting with $H_{\mathbf{T}}$.

Thus $B = \Box^3q \wedge \Box^2q$ reads

$$H_{\mathbf{T}}H_{\mathbf{K}}H_{\mathbf{T}}q \wedge H_{\mathbf{T}}H_{\mathbf{K}}q.$$

Let us evaluate B at t . We write $t \models_{\mathbf{K}} A$ when we are evaluating A at a \mathbf{K} mode and $t \models_{\mathbf{T}} A$ when we are evaluating A at the \mathbf{T} mode.

Writing the above in full we have:

- (*1) $t \models_{\mathbf{K}} \Box A$ iff for all immediately past points s we have $s \models_{\mathbf{T}} A$.
- (*2) $t \models_{\mathbf{T}} \Box A$ iff $t \models_{\mathbf{K}} A$ and for all immediately past points s we have $s \models_{\mathbf{K}} A$.

Let us now evaluate $t \models (\Box^3q \wedge \Box^2q)$ in the flow of Fig. 20. We have (remember we start with $\models_{\mathbf{T}}$):

- $t \models_{\mathbf{T}} \Box^3q$ iff first $s \models_{\mathbf{K}} \Box^2q$ and second $b \models_{\mathbf{K}} \Box^2q$ and third $t \models_{\mathbf{K}} \Box^2q$ iff first $a \models_{\mathbf{T}} \Box q$ and second $s \models_{\mathbf{T}} \Box q$ and third $b \models_{\mathbf{T}} \Box q$ and $s \models_{\mathbf{T}} \Box q$.
- $t \models_{\mathbf{T}} \Box^2q$ iff first $s \models_{\mathbf{K}} \Box q$, and second $b \models_{\mathbf{K}} \Box q$ and third $t \models_{\mathbf{K}} \Box q$.

Since both $t \models_{\mathbf{T}} \Box^3q$ and $t \models_{\mathbf{T}} \Box^2q$ must hold, we see that we need to evaluate both $s \models_{\mathbf{T}} \Box q$ and $s \models_{\mathbf{K}} \Box q$.

This means that we cannot make the evaluation of $\Box q$ at s be dependent solely on the properties of the set $\{y \mid yRs\}$. We do need the dependency on the \mathbf{T} and \mathbf{K} modes.

Indeed, we axiomatise in [9] a modal logic with only the connective \Box with the property that this logic can be characterised by the two \mathbf{K} and \mathbf{T} modes but it cannot be characterised by any class of frames. This shows that mode shifting is a genuinely stronger instrument of defining modal logics than imposing conditions on the accessibility relation R . We will to show in this paper that this logic can be characterised by a class of reactive models.

So much for a short survey of the ideas of [9]. See Appendix A for formal definitions of hyper-modal logics. Let us now proceed to show the connection with the reactive semantics of this paper.

First observe that the mode described above change the meaning of \Box . The modes do not change the semantics. In other words, the geometry of Fig. 20 remained fixed. The model has not changed during the course of evaluation of $t \models B$. We want to show that we can achieve the same effect by changing the semantics as we evaluate.

Fig. 20 Illustrating hyper-modality

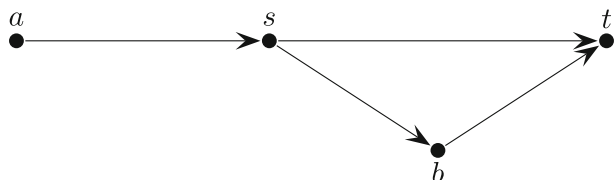
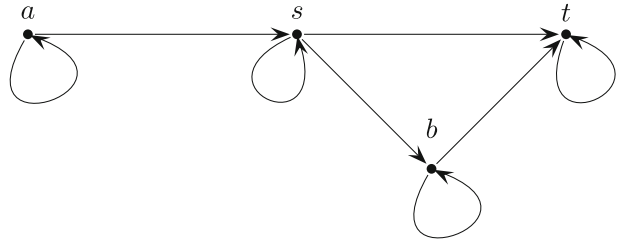


Fig. 21 Figure 20 with time reflexive



Consider Fig. 21.

Figure 21 describes the same flow of time as that of Fig. 20 with the addition of the property that time is reflexive. Now suppose we have two modes of evaluation $\models_{\mathbf{T}}$, where we evaluate \Box in the reflexive mode (i.e., Fig. 21), and $\models_{\mathbf{K}}$, where we evaluate in the irreflexive model (i.e., in Fig. 20).

Let us spell it out clearly:

- (*3) $t \models_{\mathbf{K}} \Box A$ iff first deactivate all reflexive arrows in the accessibility relation R and then ask for $s \models_{\mathbf{T}} A$ to hold at every s which is accessible to t .
- (*4) $t \models_{\mathbf{T}} \Box A$ iff first reactivate all cancelled reflexive arrows in R and then ask for $s \models_{\mathbf{K}} A$ to hold at every s accessible to t .

Clearly the evaluation of $t \models B$ will end up the same whether we view it as shifting the meaning of \Box or shifting the underlying accessibility relation in the model.

Let us view the changing of the semantics as disconnecting or reconnecting arrows (accessibility) in the model.

This we have already done in the technical example of Section 1.3. See Fig. 7.

It is not difficult to see how a general arc model can be constructed in which \Box alternates between a **T** and a **K** modality.

To do this properly, we need first a formal definition of the reactive semantics for modal logic.

4 Switch reactive Kripke models

We now give a definition of what we call switch reactive Kripke models, giving rise to what we call switch arc modal logics. In these models the reactive double arrows are just on/off switches, and the emanate from points.

Definition 4.1 Let S be a set of possible worlds and $a \in S$ is the actual world. An arc-accessibility relation on S is defined as follows:

1. The set of all arcs \mathcal{A} is defined by
 - 1.1. $S \subseteq \mathcal{A}$, these are 0 level arcs.
 - 1.2. If $\alpha \in \mathcal{A}$ is an n level arc and $s \in S$ then $\beta = (s \rightarrow \alpha)$ is an $n + 1$ level arc.
2. A subset $\mathcal{R} \subseteq \mathcal{A}$ is an arc relation.
3. An arc-Kripke model has the form $(S, \mathcal{R}, \mathcal{R}^*, a, h)$, where \mathcal{R} and \mathcal{R}^* are an arc-accessibility relations and $\mathcal{R} \subseteq \mathcal{R}^*$ and h is an assignment giving to each $t \in S$

and each atom q a value $h(t, q) \in \{0, 1\}$. The part of the model $(S, \mathcal{R}, \mathcal{R}^*, a)$ is called the model frame and $a \in S$ is the actual world of the frame.

- Note that in this definition all arcs emanate from points.

Definition 4.2 Let $\mathbf{m} = (S, \mathcal{R}, \mathcal{R}^*, a, h)$ be a model.

Define $a \models A$, by structural induction

- $a \models q$ if $h(a, q) = 1$, for q atomic.
- $a \models A \wedge B$ iff $a \models A$ and $a \models B$.
- $a \models \neg A$ iff $a \not\models A$.
- $a \models \Box A$ iff for all s in S such that $(a, s) \in \mathcal{R}$ we have that $s \models A$ in the model $\mathbf{m}_s = (S, \mathcal{R}_a, \mathcal{R}^*, s, h)$, where \mathcal{R}_a is obtained from \mathcal{R} as follows:

$$\mathcal{R}_a = \mathcal{R} - \{ \alpha \mid (a \rightarrow \alpha) \in \mathcal{R} \wedge \alpha \in \mathcal{R} \} \cup \{ \alpha \mid (a \rightarrow \alpha) \in \mathcal{R} \wedge \alpha \notin \mathcal{R} \wedge \alpha \in \mathcal{R}^* \}.$$

This is a switch satisfaction clause for \Box .

- A model $\mathbf{m} = (S, \mathcal{R}, \mathcal{R}^*, a, h)$ is said to be of level $\leq n, n \geq 1$, if all its arcs in \mathcal{R}^* are of level $\leq n$.

For example, a model of level ≤ 2 can contain either arcs of the form $t \rightarrow s$ or of the form $r \rightarrow (t \rightarrow s)$, where $r, t, s, \in S$.

- Let $\mathbf{K}_A^n, n \geq 1$ be the set of wffs valid in the class of all reactive Kripke models of level $\leq n$. Note that \mathbf{K}_A^1 is ordinary modal \mathbf{K} .

Let \mathbf{K}_A Be the set of wffs valid in the class of all reactive Kripke models. This is the basic reactive analogue of modal \mathbf{K} .

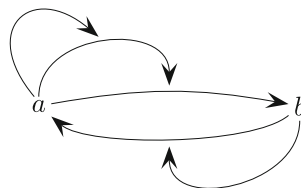
- A modal logic \mathbf{L} is said to be a reactive modal logic if for some class \mathcal{K} of reactive models $\mathbf{L} = \{ A \mid A \text{ is valid in all models of } \mathcal{K} \}$.
- Note that to be more precise these models use only arcs emanating from points and the reactivity acts like a switch.
- Note that the evaluation of $a \models \Box A$ and $a \models \Box B$ for any two wffs A and B is done independently of one another and in parallel, as indicated in item 4 above. We have $\Box(A \wedge B) \leftrightarrow \Box A \wedge \Box B$ and both A and B in this case get evaluated at the model with \mathcal{R}_a .

This gives us scope to introduce the connective “ \wedge ”, where $X \wedge Y$ reads A and then y . So $a \models \Box A \wedge \Box B$ will evaluate $a \models \Box A$ first. In the course of the evaluation of $\Box A$ the relation \mathcal{R} will become \mathcal{R}_a and then we evaluate $a \models \Box B$ in the model $(S, \mathcal{R}_a, \mathcal{R}^*, a, h)$.

Example 4.3 Consider the two point model of Fig. 22 with $S = \{a, b\}$:

Let $\mathcal{R} = \mathcal{R}^* = \{ a \rightarrow b, b \rightarrow a, a \rightarrow (a \rightarrow b), a \rightarrow (a \rightarrow (a \rightarrow b)), b \rightarrow (b \rightarrow a), b \rightarrow (b \rightarrow (b \rightarrow a)) \}$.

Fig. 22 Higher level reactive model



Consider a model

$$\mathbf{m} = (S, \mathcal{R}, \mathcal{R}^*, a, h).$$

Here $\mathcal{R}_a = \{b \rightarrow a, b \rightarrow (b \rightarrow a), a \rightarrow (a \rightarrow (a \rightarrow b))\}$.

$$\begin{aligned} \mathcal{R}_{a,b} &= \{b \rightarrow (b \rightarrow a), a \rightarrow (a \rightarrow (a \rightarrow b))\} \\ \mathcal{R}_{a,b,a} &= \{b \rightarrow (b \rightarrow a), a \rightarrow (a \rightarrow (a \rightarrow b)), a \rightarrow (a \rightarrow b)\}. \end{aligned}$$

Note that $a \rightarrow b$ is not restored until $\mathcal{R}_{a,b,a,b,a}$.

Definition 4.4 Let (S, R) be a set S with a binary relation $R \subseteq S^2$. Let τ be a Horn clause theory in the language with R containing universal clauses of the form (universal closure) $(\bigwedge_{i=1}^n x_i R y_i \rightarrow x R y)$. Let $R^\tau \supseteq R$ be defined as the smallest extension of R such that $(S, R^\tau) \models \tau$. R^τ can be constructed by induction as the closure of R under all instances of τ as follows:

1. Let $R_0 = R$
2. Let $R_{n+1} = \{(a, b) \mid \text{for some clause } L \text{ in } \tau \text{ of the form (universal closure) } (\bigwedge x_i R y_i \rightarrow x R y) \text{ and a substitution } \theta \text{ to the variables of } L \text{ such that } \theta(y_i), \theta(x_i) \in S, \theta(x) = a, \theta(y) = b, \text{ we have that } \bigwedge \theta(x_i) R_n \theta(y_i) \text{ holds.}\}$
3. Let $R_\tau = \bigcup_n R_n$.

Definition 4.5

1. Let \mathbb{N} be the set of natural numbers $\{0, 1, 2, \dots\}$. Let \mathbb{N}^* be the set of all finite sequences of natural numbers including the empty sequence \emptyset . Define $\alpha < \beta$, for $\alpha, \beta \in \mathbb{N}^*$ by

$$\alpha < \beta = (\text{definition}) \text{ for some } m \in \mathbb{N}, \beta = \alpha * (m),$$

where $*$ is concatenation.

Let $<^*$ be the transitive closure of $<$.

2. A tree T is a nonempty subset of \mathbb{N}^* such that if $\beta \in T$ and $\alpha < \beta$ then $\alpha \in T$.
3. Let τ be a Horn theory on $<$. Let $(T, <)$ be a tree and let $(T, <^\tau)$ be its τ -closure. Define \mathcal{R}_τ as follows:

- $(\alpha \rightarrow \beta) \in \mathcal{R}_\tau$ if $(\alpha, \beta) \in <^\tau$.
- $(\gamma \rightarrow (\alpha \rightarrow \beta)) \in \mathcal{R}_\tau$ whenever $\gamma <_*^\tau \alpha$ and $\gamma <_*^\tau \beta$ hold where $<_*^\tau$ is the transitive closure of $<^\tau$.

4. Let \mathcal{K}_τ be the class of all models of the form $(T, <, \mathcal{R}_\tau, \delta, h)$, where T is a tree and $\delta \in T$.
5. Let H_τ be a set of all wffs A such that A holds in any model of \mathcal{K}_τ .

Example 4.6 Let $\tau = \{\forall x(xRx)\}$. Then the models of \mathcal{K}_τ have the form $(T, <, \mathcal{R}_\tau, \delta, h)$, where $\mathcal{R}_\tau = < \cup \{\gamma \rightarrow (\alpha \rightarrow \alpha) \mid \gamma \leq^* \alpha\}$. This is so since $<^\tau$ is $< \cup \{(\alpha, \alpha) \mid \alpha \in T\}$.

It is easy to see that the meaning of \Box alternates between **K** and **T** modalities because \mathcal{R} switches the reflexive arcs on and off. Thus the logic H_τ of our example is the same as the logic H_1^S of Section 3 of [9].

We know from [9] that the following is a Hilbert axiomatisation of H_τ . We make use of our irreflexivity rule, see [6].

Axioms (E, F are wffs without \Box).

1. $A \wedge \Box A$, where A is a substitution instance of a truth functional tautology.
2. $\Box(A \rightarrow B) \rightarrow (\Box A \rightarrow \Box B)$
3. $\Box(\Box(A \rightarrow B) \rightarrow (\Box A \rightarrow \Box B))$
4. $\Diamond \top$
5. $\neg E \wedge \Box^2 E \wedge Y \rightarrow \Diamond(\neg E \wedge Y)$, where $Y = A$ or $Y = \Box A$, for A without \Box .
6. $\neg E \wedge \Box^2 E \wedge \Diamond(\neg E \wedge A) \wedge \Diamond(\neg E \wedge B) \rightarrow \Diamond(\neg E \wedge A \wedge B)$.
7. $\neg E \wedge \Box^2 E \rightarrow \Diamond(\neg E \wedge A) \vee \Diamond(\neg E \wedge \neg A)$
8. $\Box A \wedge \neg E \wedge \Box^2 E \rightarrow \Diamond(\neg E \wedge A)$.
9. $\Box X \wedge \neg E \wedge \Box^2 E \rightarrow \Diamond(\neg E \wedge \Box(\neg \wedge \Box^2 F \rightarrow \Diamond \neg F \wedge X))$
10. $\neg E \wedge \Box^2 E \wedge \Diamond A \wedge \Diamond(\neg E \wedge \Box Y \wedge \Box \Box Y' \wedge \neg A \wedge \Box \Box X) \rightarrow \Diamond(A \wedge X \wedge Y \wedge \Box Y')$ where Y, Y' are without \Box
11. $\neg E \wedge \Box^2 E \wedge \Diamond(C \wedge E \wedge Y) \rightarrow \Diamond(\neg E \wedge \Diamond(Y \wedge E \wedge \Diamond(C \wedge E)))$

Rules

$$\text{MP} \quad \frac{\vdash A; \vdash A \rightarrow B}{\vdash B}$$

$$\text{IRR} \quad \frac{\vdash \neg q \wedge \Box^2 q \rightarrow A}{\vdash A}$$

$$\text{2-necessitation} \quad \frac{\begin{array}{l} \vdash A \\ \text{where } q \text{ is an atom not in } A. \end{array}}{\vdash \Box_n^2 A}$$

$$\text{IRR}^n \quad \frac{\vdash \bigwedge_{m=1}^n \beta_m^m \rightarrow A}{\vdash A}$$

where β_m^m are as defined below and q_j^i are all not in A .

The following defines β_j^i :

Let q_j^i be a double indexed sequence of atoms. Let

1. $\beta_1^i(q_1^i) = \neg q_1^i \wedge \Box^2(q_1^i)$.
2. $\beta_2^i(q_1^i, q_2^i) = \neg q_2^i \wedge \Box^2 q_2^i \wedge \Diamond(\neg q_2^i \wedge \Box \beta_1^i)$.
3. $\beta_{n+1}^i(q_1^i, \dots, q_{n+1}^i) = \neg q_{n+1}^i \wedge \Box^2 q_{n+1}^i \wedge \Diamond(\neg q_{n+1}^i \wedge \Box \beta_n^i)$

Example 4.7 We now give an example of a class of reactive Kripke models characterising a logic which cannot be presented as a hyper-modal logic. Consider one point models of the following form, see Fig. 23.

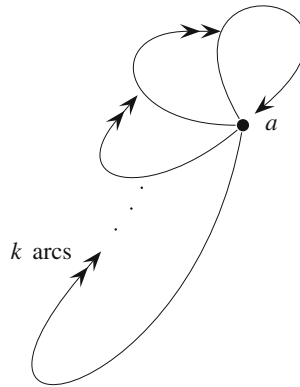
Let

$$\begin{aligned} \text{arc}_0 &= a \rightarrow a \\ &\vdots \\ \text{arc}_{n+1} &= (a \rightarrow \text{arc}_n) \end{aligned}$$

Consider the models \mathbf{m}_k of the form

$$\mathbf{m}_k = (\{a\}, \{\text{arc}_0, \text{arc}_k\}, \{\text{arc}_n \mid n \leq k\}, a)$$

Fig. 23 Kripke models characterising a logic which cannot be presented as a hyper-modal logic



The model \mathbf{m}_k is a one node model with arcs as in Fig. 23, where only the arcs $a \rightarrow a$ and $a \rightarrow (a \rightarrow \dots (a \rightarrow a) \dots)$ (k arrows) are active.

The following points are clear for the class of models $\{\mathbf{m}_k\}$.

1. $\Diamond^n \Box \perp$ is consistent for all n .
2. The following holds in the logic \mathbf{L} of this class

$$\vDash \Diamond^n \Box \perp \rightarrow \bigwedge_{m \leq n} (\Box^m A \Leftrightarrow A)$$

for all wffs A without \Box .

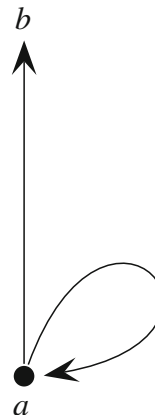
We claim the logic \mathbf{L} cannot be presented as a hyper-modal logic.

For suppose there is a class of traditional Kripke models of the form (S, R, a) and a sequence (Ψ_1, \dots, Ψ_k) of conditions characterising modalities \Box_1, \dots, \Box_k such that the evaluation of \Diamond alternates according to this sequence (see Definition A.1).

Then for a high enough n , the meaning of \Diamond in $\Diamond^n \Box \perp$ starts repeating itself.

Let (S, R, a) be a model of $\Diamond^n \Box \perp$, for n large enough. By property 2, all $\Diamond^m, m \leq n$ satisfy $\vDash_a \Diamond^m A \Leftrightarrow A$, for A without \Box . Thus we must have $\Psi_i(a, R) = \{(a, a)\}$. Therefore, how can we also have $\vDash_a \Diamond^n \Box \perp$?

Fig. 24 A hyper-modal logic which cannot be characterised by a class of finite level reactive models



This example shows that there is a class of reactive models of finite level defining a modal logic **L** which is not a hyper-modal logic.

Example 4.8 We now exhibit a hyper-modal logic which cannot be characterised by a class of finite level reactive models. Consider the situation in Fig. 24.

Let $\Psi_1 = \{(a, a)\}$ and let $\Psi_2 = \{(a, a), (a, b)\}$. Consider the sequence (Ψ_1, Ψ_1, Ψ_2) . This means that \Box is interpreted as seeing only the a node twice and then it can also see b once before repeating. Let **L** be the hyper-modal logic defined by this set up.

To implement this logic by reactive models we need to switch the arc $a \rightarrow b$ on and off by other arcs in the repeating sequence $(-, -, +)$.

Figure 25 can help visualise the situation.

We note that if the connection $a \rightarrow b$ must alternate as $(-, -, +)$ then connection $a \rightarrow (a \rightarrow b)$ (i.e., arc_1 in Fig. 25) must alternate $(-, +, +)$ and arc_2 must alternate $(+, -, +)$ and so on.

However in any finite level model, the highest level arcs cannot alternate. Hence modalities of the form \Box^n for high enough n cannot be implemented in any given model. We conjecture that if we allow models of unbounded level then all reasonable hyper-modalities can be implemented. In our case the sequence $(-, -, +)$ can be implemented by a single model of infinite level.

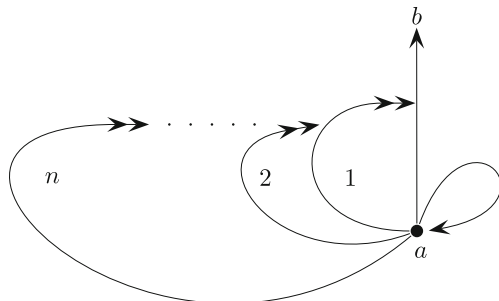
Remark 4.9 Let us give some thought to the problem of how to axiomatise the basic reactive modal logic \mathbf{K}_A (see Definition 4.1, item 6). Let Δ be a consistent complete theory of this logic. Then for some model $\mathbf{m} = (S, \mathcal{R}, \mathcal{R}^*, a, h)$ we have $\Delta = \{A \mid a \models A\}$. So Δ must contain within it enough information to reconstruct a canonical model with a suitable arc accessibility. How this can be is not immediately clear. In ordinary modal logic we can construct accessibility $\Theta R \Theta'$ by letting it be

$$\forall \alpha (\Box \alpha \in \Theta \rightarrow \alpha \in \Theta)$$

We can do the same here and this will give us the arrows but how do we define the double arrows?

Coming to think of it, we still have to check whether \mathcal{K}_A is axiomatisable or is RE? See my paper “Completeness Theorems for Reactive Modal Logics” in this issue.

Fig. 25 Trying to simulate Fig. 24



5 Non-deterministic reactive Kripke models

Let us begin by considering again the basic situation of Fig. 1. This is a simple reactive Kripke model. Let us make it non-deterministic. This would mean that as we move from node a to node b , the double arrow $a \rightarrow (b \rightarrow c)$, may or may not (non-deterministically) fire. So to evaluate $a \models \Box^2 q$, we need to evaluate $b \models \Box q$. If the double arrow does fire, then the connection $b \rightarrow c$ is canceled and only $b \rightarrow d$ is active. If the double arrow does not fire, then $b \rightarrow c$ is also active when we evaluate $b \models \Box q$.

So at this model (the non-deterministic model of Fig. 1), there are two possible non-deterministic valuations of $a \models \Box^2 q$. One with $b \rightarrow c$ active at the time of the evaluation of $b \models \Box q$ and one where it is not active. We say that a non-deterministic model \mathbf{m} can satisfy a wff A if there is at least one non-deterministic valuation of A in the model in which A gets the value \top . We say that A holds in the model if it holds (gets \top) under all non-deterministic valuations in the model.

Example 5.1 Consider the model of Fig. 26. Then as a non-deterministic model we have that $\Diamond^n \Box \perp$ can always be made false, by never activating the double arrow in the evaluation. But it can also be satisfied by activating the double arrow at the $(n - 1)$ st step and then the model becomes irreflexive at point b and $\Box \perp$ then holds at b . Thus we see that such models can satisfy both a wff A and its negation $\neg A$, because different non-deterministic options can come to bear. However, only one of A and $\neg A$ can hold in the model. We can see already that we need to be careful with this concept. Consider the axiom $\Box(A \wedge B) \rightarrow \Box A \wedge \Box B$. This is valid in any reactive Kripke model (provided the evaluations of $\Box A$ and of $\Box B$ are done in parallel, see Definition 4.2). Not so in a non-deterministic model for consider $\Diamond \Box(p \wedge \neg p) \wedge \Diamond \Box(q \wedge \neg q)$ where p and q are different atoms. When we evaluate $a \models \Diamond \Box(p \wedge \neg p)$ in the model of Fig. 26, we may take the non-deterministic choice of not firing the double arrow and thus get a value false. When we do it again for the second copy $\Diamond \Box(q \wedge \neg q)$, we may choose to fire and thus get value true. We are going to have to formulate our definitions carefully here.

The single non-deterministic model of Fig. 26 is equivalent to the two ordinary Kripke models of Fig. 27, \mathbf{m}_1 and \mathbf{m}_2

Fig. 26 Non-deterministic model

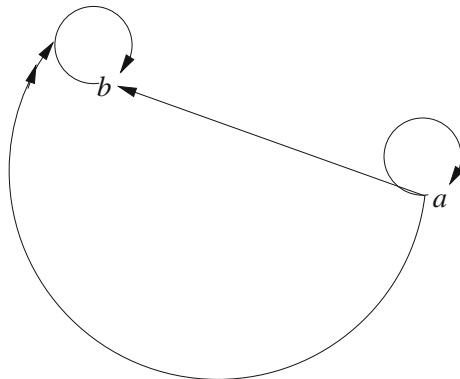
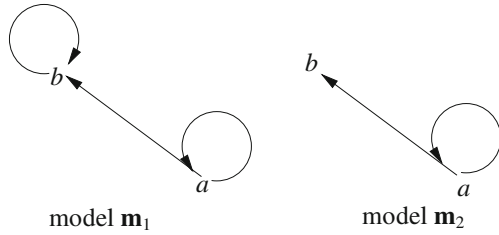


Fig. 27 Two ordinary models equivalent to Fig. 26



During the evaluation at node a we can change our view of whether node b is reflexive or not.

Definition 5.2 A non-deterministic choice of switches in a model $\mathbf{m} = (S, \mathcal{R}, \mathcal{R}^*, a, h)$ is a pair of sets ρ_a^+, ρ_a^- for each $b \in S$ satisfying the following:

$$\begin{aligned} \rho_b^+ &\subseteq \{\alpha \mid (b \rightarrow \alpha) \in \mathcal{R} \wedge \alpha \in \mathcal{R}^*\} \\ \rho_b^- &\subseteq \{\alpha \mid (b \rightarrow \alpha) \in \mathcal{R} \wedge \alpha \notin \mathcal{R}^* \wedge \alpha \in \mathcal{R}^*\} \end{aligned}$$

Definition 5.3 (Non-deterministic switch Kripke models)

1. By a model $\mathbf{m} = (S, \mathcal{R}, \mathcal{R}^*, a, h)$ we mean a model as defined in Definition 4.1.
2. By a sequence of selections of non-deterministic choices of switches in a model \mathbf{m} we mean a sequence of the form $\rho_{a,k}^\pm, k = 0, 1, \dots$ where for each $k, \rho_{a,k}^\pm$ is a non-deterministic choice as defined in the previous definition.
3. Given a wff A and a node a in a model \mathbf{m} , we define the universal non-deterministic evaluation $a \models_k A$ in parallel to Definition 4.2 as follows (where k is a natural number representing a universal ticking clock meaning that this is the k th time that a \square is evaluated at the node a).

- (a) $a \models_k q$ if $h(a, q) = 1$, for q atomic and any $k \geq 0$.
- (b) $a \models_k A \wedge B$ iff $a \models_k A$ and $a \models_k B$
- (c) $a \models_k \neg A$ iff $A \not\models_k A$
- (d) $a \models_k \square A$ iff for all $s \in S$ such that $(a, s) \in \mathcal{R}$, we have that $s \models_{k+1} A$ in the model \mathbf{m}_s^k , where

$$\mathbf{m}_s^k = (S, \mathcal{R}_a^k, \mathcal{R}^*, s, h)$$

and where

$$\mathcal{R}_a^k = (\mathcal{R} - \rho_{a,k}^+) \cup \rho_{a,k}^-.$$

- (e) A formula A is said to be satisfiable in a model \mathbf{m} if for some non-deterministic choice sequence $\rho_{t,k}^\pm, t \in S, k = 0, 1, 2, \dots$ We have $[(f)]$ that $a \models_0 A$. Note that it is possible to have that both $a \models_0 A$ and $a \models_0 \neg A$.
- (g) A formula is said to hold in a model \mathbf{m} if its negation is not satisfiable in \mathbf{m} .
- (h) A modal logic \mathbf{L} is said to be a non-deterministically reactive if for some class of reactive Kripke models we have

$$\mathbf{L} = \{A \mid A \text{ is non-deterministically valid in all models in the class}\}$$

Example 5.4 This example explains the ‘ticking clock’ of the previous definition.

Consider the frame of Fig. 26, and consider $a \models \Diamond \Box (p \wedge \neg p)$ and again $a \models \Diamond \Box (q \wedge \neg q)$.

In the first evaluation we activate the double arrow immediately and get true and in the second evaluation we do not activate it and get false. To avoid such a situation, we introduce a universal counting clock, counting the number of nested modalities. We now write

$$a \models_0 \Diamond \Box (P \wedge \neg p) \text{ iff } a \models_1 \Box (p \wedge \neg p) \text{ or } b \models_1 \Box (p \wedge \neg p)$$

We require always the same non-deterministic choice for any evaluation of the form $t \models_k \Box$, with the same t and k .

If we have this we get that $\Box(A \wedge B) \rightarrow \Box A \wedge \Box B$ holds

6 Connection with fibring logics

This section gives some methodological remarks and points out a connection with fibring logics, see [8, 14].

6.1 Methodological considerations

Consider two modal logics \mathbf{L}_1 and \mathbf{L}_2 . Assume \mathbf{L}_1 has a \mathbf{K} modality, \Box_1 and let \mathbf{L}_2 have a $\mathbf{K4}$ modality, \Box_2 . We know that \mathbf{L}_1 is complete for all Kripke frames of the form (S_1, R_1, a_1) , with $a_1 \in S_1, R_1 \subseteq S_1^2$, and \mathbf{L}_2 is complete for all Kripke frames of the form (S_2, R_2, a_2) with $a_2 \in S_2, R_2 \subseteq S_2^2$ and R_2 is a transitive (irreflexive) relation. When we combine these two logics in a way which is generally called *fusion* (or *dovetailing* in [8]), we get completeness for the class of all Kripke models of the form (S, R_1, R_2, a) with R_2 transitive irreflexive and where the truth table for \Box_i is defined as follows:

$$t \models \Box_i q \text{ iff } \forall s (tR_i s \rightarrow s \models q) \text{ for } i = 1, 2. \tag{*1}$$

The above combination of \mathbf{L}_1 and \mathbf{L}_2 assumes that the languages \mathbf{L}_1 and \mathbf{L}_2 are disjoint, and to the extent that they share the classical connectives, they agree on these connectives.

Assume now that \mathbf{L}_1 and \mathbf{L}_2 contain an additional modality \Box , i.e., $\mathbf{L}_1 = \mathbf{L}_2(\Box, \Box_1)$ and $\mathbf{L}_2 = \mathbf{L}_2(\Box, \Box_2)$, where \Box is a \mathbf{T} modality. Now the models for \mathbf{L}_i have the form (S_i, R, R_i, a_i) where $R \subseteq S_i^2$ and R is reflexive. Obviously \mathbf{L}_1 and \mathbf{L}_2 which contain the *same* symbol for modality \Box , must agree on the nature of this modality (namely that it is a \mathbf{T} modality). We can therefore safely combine them (in the same way as before) and get a logic \mathbf{L}_{1+2} with \Box, \Box_2, \Box_1 . This logic is complete for models of the form (S, R, R_1, R_2, a) where $R \subseteq S^2$ is reflexive and $R_2 \subseteq S^2$ is irreflexive and transitive, and $R_1 \subseteq S^2$.

The question arises what can we do if \mathbf{L}_1 and \mathbf{L}_2 do not agree on the modality \Box ? What if \mathbf{L}_1 has \Box as \mathbf{K} modality while \mathbf{L}_2 has \Box as a \mathbf{T} modality? How do we do this? To put it bluntly, can we combine two systems for the same modality \Box , where one says \Box is a \mathbf{K} modality and the other one says that \Box is a \mathbf{T} modality?

To make the problem more acute we can have modalities with contradictory conditions on R . For example, \mathbf{L}_1 can ask $\forall x \exists y xRy$ (the axiom $\Box \Diamond \top$) and \mathbf{L}_2 can ask $\exists x \forall y \neg xRy$ (the axiom $\Diamond \Box \perp$).

One way to combine two different systems for the same symbols is to timeshare the symbol, i.e., \Box can sometimes be an \mathbf{L}_1 modality and sometimes an \mathbf{L}_2 modality. This is the way processes share the same resource; they timeshare according to a certain protocol μ .

So let us examine modal logics with a single modality \Box , where the meaning of \Box is timeshared between, say a \mathbf{K} and a \mathbf{T} modality.

The questions we ask are the following:

- (Q1) How do we define and execute the timeshare?
- (Q2) What is the semantics corresponding to (Q1)?

Let us start by choosing something simple. Let the meaning of \Box alternate between a \mathbf{T} and a \mathbf{K} modality. Thus a wff of the form

$$\alpha = \Box(\Box q \wedge \Box\Box p)$$

will read as

$$\beta = \Box_{\mathbf{T}}(\Box_{\mathbf{K}} q \wedge \Box_{\mathbf{K}}\Box_{\mathbf{T}}\Box_{\mathbf{K}} p).$$

Thus the meaning of \Box alternates according to its nesting depth in the formula.

So now we know how to read α , namely we read it as β , but what is the semantics for β ? Being a mono-modal logic, we expect a Kripke semantics with a single accessibility relation R , i.e., models of the form (S, R, a) . So how can R be and not be reflexive alternatively? This is impossible!

To this end we need a new point of view on how we see our semantics. Let us consider the notion of a *mode of search* for a possible world. Namely let $\Psi(x, y)$ be (a possibly higher order) binary wff on the classical model (S, R, a) . Let

$$t \models_{\Psi} \Box q \text{ iff for all } s \text{ such that } \Psi(t, s) \text{ holds we have } s \models q. \tag{*2}$$

Ψ is the mode of search for worlds s . The following are examples:

- For \mathbf{K} modality $\Psi_{\mathbf{K}}(x, y) \equiv xRy$
 - For \mathbf{T} modality $\Psi_{\mathbf{T}}(x, y) \equiv x = y \vee xRy$
 - For $\mathbf{K4}$ modality $\Psi_{\mathbf{K4}}(x, y) \equiv$ for some $n \geq 0$ we have $xR^n y$.
- Note that $\Psi_{\mathbf{K4}}$ is not first-order.

The truth condition for a system with alternating \mathbf{K} and \mathbf{T} modality can be written as follows:

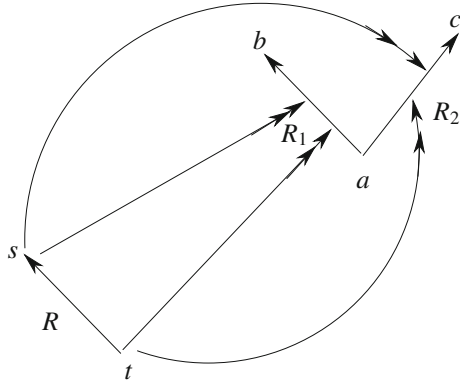
$$\begin{aligned} \bullet t \models_{\mathbf{K}} \Box q \text{ iff for all } s, \Psi_{\mathbf{K}}(t, s) \rightarrow s \models_{\mathbf{T}} q. \\ \bullet t \models_{\mathbf{T}} \Box q \text{ iff for all } s, \Psi_{\mathbf{T}}(t, s) \rightarrow s \models_{\mathbf{K}} q. \end{aligned} \tag{*3}$$

The logic with (S, R, a) where \Box alternates in meaning as above can be seen as one way of fibring disagreeing modalities via a timesharing protocol.

Let us use the following terminology:

- use the term *C-fibring* for *Compromise Fibring* when we combine two logics which do not agree on the common part.
- Use the words μ -*Protocol* for any procedures μ for timesharing the common symbols.
- A modality which changes its meaning in the sentence is called a *hypermodality*, see [9].

Fig. 28 Illustrating the basic reactive mechanism



We now continue and discuss another ‘twist’ in our way of thinking. Obviously when \Box is interpreted as \Box_T , we use Ψ_T . How about modalities complete for classes of models whose relations cannot be nicely defined via some Ψ (even second order)? We certainly can form models of the form (S, R_1, R_2, a) and let the meaning of \Box alternate between R_1 and R_2 . However, this is not neat, because with one symbol \Box we expect semantics with one symbol R or some generalisation of it.

We need some inspiration at this junction. So here it is:

We talk about *reactive models*. These are models whose accessibility relation changes as we go along. Figure 28 illustrates the basic mechanism:

We evaluate $t \models \Box q$ at moment t . We assume that at each moment, either all the R_1 or all the R_2 (but not both) connections are on. When we move from $t \models \Box q$ to s , i.e., to $s \models q$, we send a signal from $t (\rightarrow)$ to all other R_1 and R_2 connections. If a connection receives the signal and it is on, then it turns off. If it is off then it turns on. So when we come to s , we switch from R_1 to R_2 or from R_2 to R_1 depending which one is active. In general, we deal with reactive models as in Section 2.

Now in order to be able to compromise-fibre two logics with the same modality but different reactive Kripke semantics, we need to allow arrow \rightarrow to cancel other arrows up to arbitrary finite iterations. This yields Definitions 4.1 and 4.2 of Section 4.

We have now completed our methodological motivation for reactive Kripke models. For more on compromise fibring, see [14].

6.2 Formal compromise fibring

It is now time to give a formal definition of C -fibring.

Definition 6.1

1. Let (S, R, a) be a Kripke frame. By a *finite path* we mean a sequence of elements of the form $\vec{S} = (a, a_1, \dots, a_n)$ such that $aRa_1 \wedge a_1Ra_2 \wedge \dots \wedge a_{n-1}Ra_n$ holds. We also include the empty sequence as a path.
2. Let $t \in S$, we define the model $(S_{[t]}, R_{[t]}, t)$ as follows.
 - a. Let $S_{[t]}$ be the smallest subset $S' \subseteq S$ such that $t \in S'$ and if $x \in S'$ and xRy then $y \in S'$.
 - b. Let $R_{[t]}$ be $R \upharpoonright S_{[t]}$.

Definition 6.2 Let $\mathcal{K}_1, \dots, \mathcal{K}_n$ be n classes of Kripke frames. Let $\mathbf{L}_1, \dots, \mathbf{L}_n$ be the logics they define as frames in a language with the syntactical modality \Box .

We define a C-fibred general model for \Box as follows:

1. (S, R, a, \mathbf{f}) is a C-fibred model for $\mathbf{L}_1 + \mathbf{L}_2 + \dots + \mathbf{L}_n$ iff \mathbf{f} is a function giving for each finite path $\vec{t} = (a, a_1, \dots, a_n)$ a value $\mathbf{f}(\vec{t}) = R_t^{\mathbf{f}} \subseteq R_{[a_n]}$ such that for some $m \leq n$, the model $(S_{[a_m]}, R_t^{\mathbf{f}}, a_m)$ is in \mathcal{K}_m .
2. We define satisfaction in (S, R, a, \mathbf{f}) as follows (under an assignment h to the atoms) for $\vec{t} = (a, a_1, \dots, a_n)$:
 - $a_n \models_{\vec{t}} \Box q$ iff for all $s \in S_{[t]}$ such that $tR_t^{\mathbf{f}}s$ we have that $s \models_{t*(s)} q$, where $*$ is concatenation of sequences.
3. Note that the condition $R_t^{\mathbf{f}} \subseteq R_{[a_n]}$ is not restrictive because we can always start with $R = S^2$.
4. Let $\mathbf{L}_1 + \dots + \mathbf{L}_n$ be the logic defined by all models (S, R, a, \mathbf{f}) .

Remark 6.3 Note that ordinary fibring of two modalities \Box_1 and \Box_2 can be captured by this semantics. Let us pretend that $\Box_1 = \Box_2 = \Box$ and that \mathbf{L}_1 reads \Box as \Box_1 for a class of models \mathcal{K}_1 and similarly \mathbf{L}_2 for \mathcal{K}_2 . Consider all models with frames (S, R_1, R_2, a) where $(S, R_1, t) \in \mathcal{K}_1$ and $(S, R_2, t) \in \mathcal{K}_2$ for all $t \in S$. Let R be defined as $R_1 \cup R_2$. Let $\mathbf{f}(a, a_1, \dots, a_n) = R_1$ if n is odd and R_2 if n is even. Then \Box_1 and \Box_2 can be translated in a *context sensitive way* as seen by the following example.

Let $\alpha = \Box_1(\Box_1 q \wedge \Box_2 \Box_1 p)$. Then α is translated into β .

$$\beta = \Box(\Box^2 q \wedge \Box \Box p).$$

We translate each \Box_i from outside in. We translate each \Box_i as either \Box or $\Box \Box$ depending whether it gives us an odd or even nesting of boxes as required by the index i . **This translation is not faithful. More work needs to be done on this idea.**

7 Dedicated reactivity connectives

Previous sections dealt with a modality \Box that both sent the evaluation to accessible worlds and at the same time activated all reactivity double arrows. Let us check and

Fig. 29 Illustrating the Jump operation

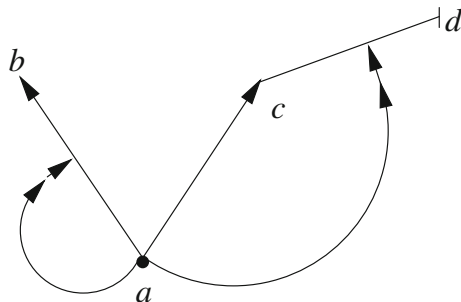
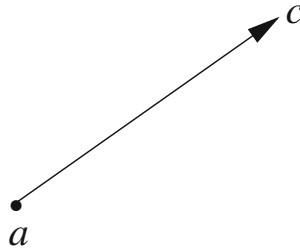


Fig. 30 Jumping from Fig. 29



see how we can separate these two functionalities using two separate connectives. Consider the simple model in Fig. 29.

When we evaluate $a \models \Box q$, we move to the evaluation of $b \models q$ and $c \models q$. The node a also sends a signal along the double arrows at a and as a result the arcs (a, b) and (c, d) are disconnected.

Let us separate these two operations. Let us look at the fault disconnection signal separately from the ordinary modality operation. Let us have an ordinary modality $\Box \vec{A}$ and a reactive modality $\vec{\Box} A$ operating as follows:

1. $a \models \vec{\Box} A$ iff for all y such that aRy we have $y \models A$. This is an ordinary modality moving along the single arrows.
2. $a \models \Box \vec{A}$ iff $a \models A$ in the model obtained by letting all reactive double arrows do their job!

So this is a modality which changes the model (set of possible worlds and its accessibility relation). It is like a **Jump** operator of the form

$$a \models_{\text{model 1}} \mathbf{Jump} A \text{ iff } a \models_{\text{model 2}} A.$$

The move from one model to the other is done by activating the double arrows emanating from a .

So, for the situation in Fig. 29 we have that

$$\begin{aligned} a \models \vec{\Box} A \text{ in the model of Fig. 29} \\ \text{iff} \\ a \models A \text{ in the model of Fig. 30} \end{aligned}$$

Having isolated the reactive operation, we would like to check the inter-definability properties of \Box , $\vec{\Box}$ and $\vec{\vec{\Box}}$.

We will see that matters are not so simple.

We would like to have maybe that

$$\Box = \vec{\Box} \vec{\Box}$$

or

$$\Box = \vec{\vec{\Box}}$$

or some combination

$$\Box = \text{combination } (\vec{\Box}, \vec{\vec{\Box}})$$

However this is not so simple because \Box fires both $\vec{\Box}$ and $\overrightarrow{\Box}$ simultaneously. If we do $\vec{\Box}$ first, we no longer can disconnect the (c, d) arc. If we do $\overrightarrow{\Box}$ first, we no longer can get to b .

If we use $a \models \Box\top$, we do not get the effect of $\vec{\Box}$ because we move to the accessible worlds b and c .

Thus, no matter what we do, we have a problem interdefining these connectives.

In fact $\vec{\Box}$ is a metalevel connective, operating on the model (disconnecting accessibility arrows) and is different in nature from the evaluation of the ordinary connective $\vec{\Box}$. Being action metalevel connective we can talk naturally about the simultaneity of several actions. Let “ \parallel ” indicate simultaneous operation of actions, being another metalevel connective.

Then we can write

$$\Box = \vec{\Box} \parallel \overrightarrow{\Box}$$

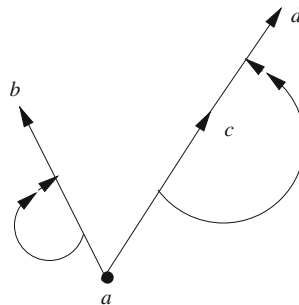
The difference between the connectives is even more serious. Consider Fig. 31.

Figure 31 is like Fig. 29 except the “fault” double arrows emanate from the arcs and not from the point. Do we want $\vec{\Box}$ to fire all double arrows from arcs or only one of them? Note that when we write $a \models \diamond A$, we either go to $c \models A$ in which case only (c, d) is disconnected or we go to $b \models A$ in which case only (a, b) is disconnected.

It looks like we need, in addition to the modalities $\vec{\Box}$ and $\overrightarrow{\Box} = \neg \vec{\Box} \neg$ six additional fault connectives to be able to express all our options:

1. $a \models \vec{\Box}_{\text{point}} A$ activates all double arrows emanating from the point a .
2. $a \models \vec{\Box}_{\text{arc}} A$ activates all double arrows emanating from all arcs beginning at a .
3. $a \models \overrightarrow{\diamond}_{\text{point}} A$ activates at least one double arrow emanating from a
4. $a \models \overrightarrow{\diamond}_{\text{arc}}^{\text{one}} A$ activates at least one double arrow emanating from one arc coming out of a
5. $a \models \overrightarrow{\diamond}_{\text{arc}}^{\text{all}} A$ activates *all* double arrows emanating from at least one arc coming out of a .
6. $a \models (X \parallel Y)A$. Execute XA and YA in parallel, simultaneously.

Fig. 31 A variation of Fig. 29



Thus we have

- $\overset{\rightarrow}{\Box} = \overset{\rightarrow}{\Box} \parallel \overset{\rightarrow}{\Box}_{\text{point}} \parallel \overset{\rightarrow}{\Box}_{\text{arc}}$
- $\overset{\rightarrow}{\Diamond} = \overset{\rightarrow}{\Diamond} \parallel \overset{\rightarrow}{\Box}_{\text{point}} \parallel \overset{\rightarrow}{\Diamond}_{\text{arc}}^{\text{all}}$

Note that we can use \parallel also as a connective and write $(\overset{\rightarrow}{\Box}A) \parallel (\overset{\rightarrow}{\Box}B)$, which means evaluate $\overset{\rightarrow}{\Box}A$ and $\overset{\rightarrow}{\Box}B$ in parallel at the same time so they will interfere with each other. Compare with $\overset{\rightarrow}{\Box}A \wedge \overset{\rightarrow}{\Box}B$ which do not interfere.

These are metalevel deletion operators brought into the object level by the use of connectives. Their detailed logical study is a big story on its own and needs to be done within a wider context. Deletion appears in abduction in logic and in handling databases as well as deletion of elements from models. See our anti-formula and anti-element papers [4, 5].

Acknowledgements I am grateful to David Makinson and to the referee for valuable comments.

Appendices: Further Topics

A Hypermodalities

Since we are comparing in this paper the notions of hyper-modal logics and reactive modal logics, we need to give here the exact definition of a hyper-modal logic.

Our starting point is a general Kripke model of the form $\mathbf{m} = (S, R, a, h)$. R is an arbitrary binary relation on S .

We are going to introduce evaluation modes into such semantics. It is convenient to regard the relation xRy as a classical formula $\Psi_{\mathbf{K}}(x, R, a, y)$ in the language of the relation R , the individual variables x, y and actual world constant a as follows

- $\Psi_{\mathbf{K}}(x, R, a, y) =_{\text{def}} xRy$

we have

- $t \models \overset{\rightarrow}{\Box}A$ iff $\forall s(\Psi_{\mathbf{K}}(t, s) \rightarrow s \models A)$.

We can refer to $\Psi_{\mathbf{K}}$ as the mode of evaluation for $\overset{\rightarrow}{\Box}$. It is fixed in the semantics and does not change. Intuitively it tells us, for a world x , how to evaluate $\overset{\rightarrow}{\Box}A$ at x , namely where to look for worlds y where $y \models A$ must hold. The subscript \mathbf{K} indicates that this formula is used in the case of \mathbf{K} modality.

We can think of different formulas Ψ for the mode. Consider for example:

- $\Psi_{\mathbf{T}}(x, R, a, y) =_{\text{def}} xRy \vee x = y$
- $\Psi_{\mathbf{K4}}(x, R, a, y) =_{\text{def}} (\exists n \geq 1)xR^n y$.
Where $xR^n y$ is defined by induction as:
 - $xR^0 y$ iff $x = y$
 - $xR^{n+1} y$ iff $\exists z(xRz \wedge zR^n y)$.
- $\Psi_{\mathbf{KB}}(x, R, a, y) =_{\text{def}} (xRy \vee yRx)$.

Clearly $\Psi_{\mathbf{K4}}(x, R, a, y)$ is not a first-order formula. It defines the transitive closure of R .

One can think of Ψ as changing the accessibility relation from R to $\lambda x \lambda y \Psi(x, y)$. Another way of looking at Ψ is that it gives us a new mode of how to use R in evaluating the truth value of $\Box A$. The latter view is more convenient to use because we will be shifting modes during the evaluation.

Let us write \vDash_i , to mean that the mode Ψ_i is used in the evaluation. Then $\vDash_{\mathbf{K}}$ for arbitrary frames (S, R, a) yields the logic \mathbf{K} , $\vDash_{\mathbf{T}}$ yields the logic \mathbf{T} , $\vDash_{\mathbf{KB}}$ yields the logic \mathbf{KB} and $\vDash_{\mathbf{K4}}$ yields the logic $\mathbf{K4}$.

Note that our starting point is a frame (S, R, a) with an arbitrary R . We define $\Psi_i(x, R, a, y)$ as a binary relation and use it to evaluate \Box . Thus in traditional terms the frame we are using is (S, Ψ_i, a) not (S, R, a) . When we shift modalities, i.e., change from $t \vDash_i \Box A$ to $s \vDash_j \Box A$ it is like shifting from (S, Ψ_i, a) to (S, Ψ_j, a) . We now give a formal definition of hypermodality.

We treat the simple case is where the number of modes is a finite set μ and there is a function ε for shifting modes. This case is given in the next definition.

Definition A.1 (Mode shifting) Let $\mu = \{\Psi_0, \dots, \Psi_k\}$ be a set of modes and let ε be a function assigning to each $0 \leq i \leq k$ a value $0 \leq \varepsilon(i) \leq k$.

Let (S, R, a, h) be a Kripke model. We define the following (μ, ε) satisfaction in the model

- $t \vDash_i \Box A$ iff $\forall s (\Psi_i(t, s) \rightarrow s \vDash_{\varepsilon(i)} A)$.
- We say A is true in the model if $a \vDash_0 A$.

Definition A.2

1. Let \mathcal{K} be a class of models of the form (S, R, a, h) . Let (μ, ε) be a mode system. We write $\mathcal{K} \vDash_{(\mu, \varepsilon)} A$ iff for every model (S, R, a, h) in \mathcal{K} we have $a \vDash_0 A$.
2. Let \mathbf{L} be a logic complete for a class \mathcal{K} of Kripke models of the form (S, R, a, h) . Let $\mathcal{K}[\mu, \varepsilon]$ be $\{A \mid \mathcal{K} \vDash_{(\mu, \varepsilon)} A\}$. We sometimes write $\mathbf{L}[\mu, \varepsilon]$ for $\mathcal{K}[\mu, \varepsilon]$, when the implicit dependence on \mathcal{K} is clear.

Obviously the nature of hypermodal logic depends on (μ, ε) and its abstract properties and also on the class \mathcal{K} of models chosen.

B Traversing a graph

Definition B.1 By a graph G we mean a set S with a binary relation $R \subseteq S^2$. Let $a \in S$ be the starting point. We write $G = (S, R, a)$.

Definition B.2

1. By a Horn closure condition in the language of R we mean a clauses of the form

$$C : \bigwedge_i x_i R y_i \wedge \bigwedge_j u_j \neq v_j \rightarrow x R y.$$

2. A substitution θ from $Z = \{x_i, y_i, u_j, v_j, x, y\}$ into S is a function θ assigning values $\theta(z) \in S$ for each $z \in Z$.

3. We say a relation $R^* \subseteq S^2$ satisfies the clause C if for all θ , if $\theta(x_i)R^*\theta(y_i)$ holds and $\theta(u_j) \neq \theta(v_j)$ holds then $\theta(x)R^*\theta(y)$ also holds. We write $G^* = (S, R^*, a) \models C$.
Let τ be a set of clauses. We say $G \models \tau$ iff $G \models C$ for all $C \in \tau$.

Lemma B.3 *Let $G = (S, R, a)$ be a graph and τ a set of clauses. Then there exists the smaller $R^* \supseteq R$ such that $G^* = (S, R^*, a) \models \tau$.*

Definition B.4 Let $G = (S, R, a)$ be a graph and let $\mu = (\tau_1, \dots, \tau_k)$ be a sequence of sets of clauses. Let R_1^*, \dots, R_k^* be the closures of R under τ_i resp. Define a μ -path H through S as follows.

The first element of H is $a_0 = a$. The next element of H is a_1 such that $a_0 R_1^* a_1$ holds. We say a_1 is a τ_1 choice.

Assume a_n is a τ_n choice then a_{n+1} is such that $a_n R_{n+1}^* a_{n+1}$ where $R_{n+1}^* = R_{i+1}^*$ if a_n is a τ_i choice, $i < k$ and $R_{n+1}^* = R_k^*$ if a_n is a τ_k choice.

Definition B.5 (A reactive graph) Let $G = (S, R, a)$ be a graph. Write $R = \{(x, y) \mid xRy \text{ holds}\}$. Define W as follows:

$$\begin{aligned} W_0 &= S \times S \quad \text{connections of level 0} \\ W_{n+1} &= S \times W_n \quad \text{connections of level } n + 1 \\ W &= \bigcup_n W_n \end{aligned}$$

We consider any set of connections of level ≥ 1 as a *switch*.

A subset \bar{R} of W can be used to describe S paths H as follows.

1. initial element of H is $a_0 = a$ initial set of $\bar{R}_1 = \bar{R}$.
2. Assume R_n is defined and $\bar{R}_n \subseteq \bar{R}$.
3. We define \bar{R}_{n+1} .
Let $R_{n+1} = \bar{R}_n = \{\alpha \mid (a_n, \alpha) \in R_n\} \cup \{\beta \mid (a_n, \beta) \in (\bar{R}) - R_n\}$.
We assume (a_n, a_{n+1}) is in \bar{R}_n .

Conjecture *Let H_μ be the all possible paths defined on $G = (S, R, a)$ using $\mu = (\tau_1, \dots, \tau_k)$. Then there exists an $\bar{R} \subseteq W$ that yields the same paths.*

Definition B.6

1. A multigraph has the form $G = (V, E, s, r)$ where V is a set of vertices, E a set of edges, and $s, r : E \rightarrow V$ are maps describing the source and range of edges.
2. A family of sets of edges, $R_H = R_h = (E_h, s_h, r_h)$, is called compatible if:
for

$$e \in E_h \cap E_{h'}$$

then we have

$$s_h(e) = s_{h'}(e) \text{ and } r_h(e) = r_{h'}(e) .$$

So it make sense considering a set of edges that is the union of the elements R_H as

$$\bigcup R_H = \left(\bigcup_{h \in H} E_h, s, r \right)$$

where $s(e) = s_h(e)$ and $r(e) = r_h(e)$ for any $h \in H$. In a similar way we define intersection of sets of edges and the predicate \in to edges and sets of edges of this form.

Definition B.7 A reactive multigraph has the form $G = (V, R_H)$ where $R_H = \{R_h = (E_h, s_h, r_h) \mid h \in H\}$ is a compatible family of set of edges based on V indexed on H and (V, R_h, a) is a multigraph for all h . The elements of H are sequences of edges and it satisfies:

$$\varepsilon \in H \text{ the empty path, corresponding to the empty sequence of edges}$$

$$p = e_1 \dots e_n \in H \text{ iff } e_1 \dots e_{n-1} \in H \text{ and } r_{e_1 \dots e_{n-1}}(e_{n-1}) = s_{e_1 \dots e_{n-1}}(e_n)$$

H is called the set of reactive paths of G .

Definition B.8 (Switch reactivity)

- Let $G = (V, R)$ be a multigraph and $R = (E, s, r)$. Define the total set of possible switches W_R as follows:

$$W_0 = E$$

$$\vdots$$

$$W_{n+1} = \{(e, \alpha) \mid e \in E, \alpha \in W_n\}$$

$$\vdots$$

$$W_R = \bigcup_n W_n.$$

- A switch graph \mathbf{G} has the form $(V, R, \mathbf{R}_\varepsilon, \mathbf{R})$ where (V, R) is a multigraph, which set of edges "possible in the graph", $\mathbf{R}_\varepsilon \subseteq \mathbf{R} \subseteq W_R$, \mathbf{R}_ε is the initial configuration of the switch graph.
- Let $\mathbf{R}' \subseteq \mathbf{R}$ and let $e \in E$. Define \mathbf{R}'_e as follows:

$$\mathbf{R}'_e = \mathbf{R}' - \{\alpha \mid (e, \alpha) \in \mathbf{R}'\} \cup \{\beta \mid (e, \beta) \in \mathbf{R} \wedge (e, \beta) \notin \mathbf{R}'\}.$$

We say α is *deactivated* and β is *activated* by the passing through the node x .

- Let $H = (e_1 \dots e_n)$. We let \mathbf{R}_H be defined as follows:

$$\mathbf{R}_{e_1} = (\mathbf{R}_\varepsilon)_{e_1}$$

$$\vdots$$

$$\mathbf{R}_{e_1 \dots e_{m+1}} = (\mathbf{R}_{e_1 \dots e_m})_{e_{m+1}}$$

Here we see $e \in E$ as edge $(e, s|_e, r|_e)$.

- A switch graph has level n of reactivity if $\mathbf{R} \subseteq \bigcup_{i \leq n} W_i$.

A switch graph is a particular case of a reactive graph, with:

$$R_h = \mathbf{R}_h \cap R$$

Problem B.9 Can every reactive multigraph be represented by a switch graph?

C Reactive proof theory

We give an example of how to do reactive proof theory. A full analysis will be done in [14]. The proper environment for developing reactive proof theory is the methodology of Labelled Deductive Systems, see [7]. See also [4] where reactive rules were independently introduced.

The usual natural deduction propositional system has elimination rules and introduction rules. In principle an elimination rule has the form

$$\boxed{\text{ER} \quad \frac{A_1, \dots, A_n}{B} \text{ where } A_1, \dots, A_n, B, \text{ are well formed formulas.}}$$

and an introduction rule has the form of a subproof:

$$\boxed{\text{IR: To show } \varphi(A_1, \dots, A_n, B), \text{ start a subproof with } A_1, \dots, A_n \text{ as additional assumptions and conclude the subproof successfully with obtaining } B, \text{ where } \varphi, A_1, \dots, A_n, B \text{ are well formed formulas. } \varphi \text{ is built up from } A_1, \dots, A_n, B.}}$$

Well known examples of such rules are the implicational rules:

$$\boxed{\Rightarrow \text{E} \quad \frac{A, A \Rightarrow B}{B}}$$

$$\boxed{\Rightarrow \text{I} \quad \text{To show } A \Rightarrow B, \text{ assume } A \text{ and prove } B}$$

Definition C.1 (Proofs of level $\leq n$) Let **S** be a set of proof rules.

1. A line (in a proof) is a sequence of the form

$$l : A, J, \rho, \alpha$$

where l is a line reference (line number), A is the formula of the line, J is the justification of the line, and ρ is the set of proof rules active at this line and α is a set of line references accessible at the line. ρ changes only in a reactive proof system. In an ordinary proof system ρ is always the full set of rules of the logic, and α contains all previous line references.

The justification in $l : B, J, \rho, \alpha$ can be either the word “assumption” or the phrase “ A is obtained using the elimination rule $\text{ER} : \frac{A_1, \dots, A_n}{A}$, where A_i are formulas obtained in lines $\ell_i, i = 1, \dots, n$ and ℓ_i are accessible”, or the phrase “ $A = \varphi(A_1, \dots, A_n, B)$ where A is a formula of an introduction rule IR and the justification is a subproof π whose assumptions are A_1, \dots, A_n together with whatever is accessible at ℓ ”.

2. A (non-reactive) proof π of level 0 is a sequence of numbered lines containing formulas and justifications of the form

$$\text{line number: wff, Justification}$$

such that the beginning of the sequence contains formulas justified as “assumptions” and any subsequent line has the form $\ell : B, \mathbb{R}$ where \mathbb{R} is an elimination rule of the form $\frac{A_1, \dots, A_n}{B}$ here $\ell_i : A_i$ Justification are previous lines in the sequence. We say that the sequence is a proof of the formula of the last line from the initial sequence of assumptions.

3. A proof of level $\leq n + 1$ is defined as a sequence π of lines containing formulas and justifications where the initial elements of the sequence are all justified as “assumptions” and each subsequent formula in line ℓ is either justified from previous lines using an elimination rule, as described in (2) above, or is a formula of the form $\varphi(A_1, \dots, A_n, B)$ appearing in an introduction rule, justified by a proof π_φ of level $\leq n$, of B from the assumptions A_1, \dots, A_n and whatever is accessible to ℓ .

If the last line of π contains the formula E and the initial sequence of the proof are the assumptions D_1, \dots, D_k , then we say that π is a level $\leq n + 1$ proof of E from D_1, \dots, D_k .

Definition C.2 (Reactive proof system) A reactive system has reactive rules of the form

$$\mathbf{R}_n : (\mathbb{R}, r_i^+, r_j^-) \ i = 1, \dots, k^+, \ j = 1, \dots, k^-, \ n = 1, \dots, m, \ r_i^\pm \leq m.$$

where \mathbb{R} is a rule (elimination or introduction). The reading of \mathbf{R}_n is that if we use rule \mathbb{R} then the rules $\mathbf{R}_{r_i^+}, i = 1, \dots, k^+$ should be activated and rules $\mathbf{R}_{r_j^-}, j = 1, \dots, k^-$ should be deactivated.

The notion of a proof is modified as follows.

At the start of the proof (line 1) we add a third component ρ indicating which rules are active and a fourth component α indicating which previous lines are accessible for the purpose of justification. The general nature of the logic will have procedures for telling us given a line ℓ in the proof and its α , what will be the α of the next line? For example if in line ℓ , the justification is “assumption”, then line ℓ is available to be used in justification of the next line or for example if line ℓ is justified by an elimination rule of the form $\frac{A_1, \dots, A_n}{B}$, where A_i are from accessible lines ℓ_i , resp. then lines ℓ_i are no longer accessible at the next line (line $\ell + 1$). Lines which are assumptions do not change which rules are active and which are not. A line which uses a rule \mathbf{R}_n as justification (either elimination or introduction) activates or deactivates the other rules as indicated in the rule itself and the result of the change are the rules ρ which are available for the next line of the proof. The changes in accessibility α can depend on the logic at hand. For example as we have mentioned, a resource logic may make assumptions not accessible once they are used in a justification.

Example C.3 Take the ordinary natural deduction rules for classical logic including

$$\frac{A, A \Rightarrow B}{B}$$

and

$$\frac{\perp}{B}$$

and

To show $A \Rightarrow B$ assume A and prove B

Assume that using modus ponens deactivates the negation rule.

Thus $A, A \Rightarrow \perp \not\vdash C$ by modus ponens alone using level 0 proofs because once we use modus ponens to get \perp we do not have the negation rule anymore. We could work our way around this in this case, using level 1 proofs as follows:

1. A , assumption
2. $A \Rightarrow \perp$, assumption
3. $\perp \Rightarrow C$
using \Rightarrow introduction rule,
 - 3.1. \perp , assumption
 - 3.2. C from 3.1 and the negation rule
 - 3.3. Exit with $\perp \Rightarrow C$ proved
4. \perp , from 1 and 2 using modus ponens. Note that the negation rule which was used in the subproof at 3.2 does not cancel the modus ponens in the main proof.
5. C from 4 and 5 using modus ponens again.

D J. van Benthem's sabotage modal logic

J. van Benthem introduced sabotage modal logic in [21]. The idea was put forward by van Benthem (2002) and is based on games played on graphs where one player is trying to traverse the graph while another tries to delete edges to make the first player fail.

This is another instance of reactive behaviour, conceptually conceived as sabotage. In the context of Kripke models (S, R, a, h) , van Benthem introduces two modalities, the usual evaluation modality (denoted in our notation by $\vec{\square}$) and a sabotage modality denoted by \diamond .

There are several versions for \diamond ⁸

$a \models \diamond_{\text{point}} A$ iff $a \models A$ in a new model in which some point $s \in S, s \neq a$ is deleted.

We can use the convenient notation $a \models_s A$.

$a \models \diamond_{\text{arc}} A$ iff $a \models A$ in a new model in which some arc $(t, s) \in R$ is deleted. Again we can use the notation $a \models_{(t,s)} A$.

Philip Rohde studied properties of these logics, with regard to complexity.

Another recent variation recommended by Benedikt Löwe, is to have the sabotage modality localised by turning the system two dimensional.

⁸van Benthem has modalities indexed also by actions, but this is irrelevant to the system's conceptual aspects.

$a, t \models \Diamond_{\text{arc}} A$ iff for some s such that tRs we have $a, s \models_{(t,s)} A$.⁹

We now compare sabotage modality with our “fault-remedy” modality to give the reader a view of what is going on.

It is obvious that Professor van Benthem and myself share is the idea of reactivity, of working against a system which changes under you whether by built-in weaknesses and remedies or by sabotage. The actual modal logics produced emerge from the original point of view taken.¹⁰

I also adopt the point of view that deletion is a metalevel notion and that there should be a logical discipline and stylised machinery for bringing it into the object level. This should be seen in a wider context of dynamic evaluations and dynamic operators on which there is a lot of literature. See [22] for a recent application, and the references there.

The rest of this comparison is simply technical, comparing formal options between teh fault-remedy modality and the sabotage modality.

1. Fault modality can have arrow to arrow deletions of many levels. This comes from the original point of view.
2. Switch procedures also come naturally in this context.

⁹To clarify the notation, let

- $\sharp(z_1, \dots, z_n) \equiv \bigwedge_{i \neq j} z_i \neq z_j$
- $\sharp((x_1, y_1), \dots, (x_m, y_m)) = \bigwedge_{i \neq j} (x_i \neq x_j \vee y_i \neq y_j)$

Let $t \models_{z_1, \dots, z_n} A$ mean that A holds at t , in the model where all the different points z_1, \dots, z_n have been deleted.

Similarly, $t \models_{(x_1, y_1), \dots, (x_n, y_n)} A$ means that A holds at t in the model where all the different arcs $(x_1, y_1), \dots, (x_n, y_n)$ have been deleted.

Thus for point sabotage we have

- $t \models_{z_1, \dots, z_n} \Diamond A$ iff $\exists s \exists y (tRs \wedge \sharp(t, s, z_1, \dots, z_n, y)$ and $s \models_{z_1, \dots, z_n, y} A$).

For arc sabotage we have:

- $t \models_{(x_1, y_1), \dots, (x_n, y_n)} \Diamond A$ iff $\exists s \exists u \exists v (tRs \wedge uRv \wedge \sharp((x_1, y_1), \dots, (x_n, y_n), (t, s)) \wedge \sharp((x_1, y_1), \dots, (x_n, y_n), (u, v)))$ and $s \models_{(x_1, y_1), \dots, (x_n, y_n), (u, v)} A$.

¹⁰van Benthem says in [21] (this paper was already written in 2002) as follows:

“In particular, the logical model-checking angle suggests a study of evaluation of first-order logic on structures *which change under evaluation*. E.g. an object might become unavailable once drawn from a domain, or a fact might change when inspected (think of measurement in quantum mechanics)”.

In comparison, I say in [6]

“This paper addresses the case where the semantics does change (or react) under us as we evaluate a formula. This idea makes the evaluation of a wff at a world t dependent on the route leading to t . Thus we get a new kind of semantics, the reactive semantics.”

and later in the paper, I continue to say

“we have put forward the reactive and dynamic idea of evaluation in earlier papers and lectures. A typical example we give is to consider $t \models \Diamond A$. In modal logic this means that there is a possible world s such that $s \models A$. We take a more dynamic view of it. We ask: where is s ? How long does it take to get to it? and how much does it cost to get there?”

3. Fault modality is a simultaneous parallel combination of metalevel changes and evaluation. This makes it possibly technically different from the logic where the functionalities are separated.
We can simulate sabotage by double arrows and non-determinism. The operator $\hat{\Diamond}_{\text{arc}}$, deleting an arc, can be simulated, for example, by looking at double arrows going from each node to all arcs and non-deterministically activating one double arrow every time we make a move.
4. Conceptually the fault-remedy concept is universal, while the notion of sabotage is agent specific. Consider a case of one agent sabotaging the work of several other agents. Additional special effort is required by the saboteur to make it look like a fault (accident).¹¹

References

1. Abraham, M., Belfer, I., Gabbay, D., Schild, U.: Delegation, count as and security in Talmudic logic, a preliminary study. In: Béziau, J.-Y., Coniglio, M. E. (eds.) *Logic Without Frontiers: Festschrift for Walter Alexandre Carnielli on the Occasion of his 60th Birthday*. Tribute Series, vol. 17, pp. 73–96. College Publications, London (2011)
2. Barringer, H., Gabbay, D., Rydeheard, D.: Reactive grammars. In: Dershowitz, N. (ed.) *LNCS Volume in Honour of Yakov Choueka*. Springer, Berlin (2012, to appear)
3. Barringer, H., Gabbay, D., Woods, J.: Temporal, numerical and metalevel dynamics in argumentation networks. *Argum. Comput.* **3**(2–3), 143–202 (2012)
4. Barringer, H., Rydeheard, D., Havelund, K.: Rule systems for run-time monitoring: from Eagle to RuleR. *J Logic Computation* **20**(3), 675–706 (2010)
5. Crochemore, M., Gabbay, D.: Reactive automata. *Inf. Comput.* **209**(4), 692–704 (2011). doi:[10.1016/j.ic.2011.01.002](https://doi.org/10.1016/j.ic.2011.01.002)
6. Gabbay, D.M.: An irreflexivity lemma with applications to axiomatizations of conditions on tense frames. In: Monnich, U. (ed.) *Aspects of Philosophical Logic*, pp. 67–89. Reidel, Amsterdam (1981)
7. Gabbay, D.M.: *Labelled Deductive Systems*. Oxford University Press, London (1996)
8. Gabbay, D.M.: *Fibring Logics*. Oxford University Press, London (1998)
9. Gabbay, D.M.: A theory of hypermodal logics: mode shifting in modal logic. *J. Philos. Logic* **31**, 211–243 (2002)
10. Gabbay, D.M.: Reactive Kripke semantics and arc accessibility. In: Carnielli, W., F.M. Dionesio, Mateus, P. (eds.) *Proceedings of CombLog04*, pp. 7–20. Centre for Logic and Computation, University of Lisbon (2004)
11. Gabbay, D.: Semantics for higher level attacks in extended argumentation frames. Part 1: Overview. *Stud. Log.* **93**, 355–379 (2009)
12. Gabbay, D.: Temporal denotic logic for the generalised Chisholm set of contrary to duty obligations. In: Agotnes, T., Broersen, J., Elgesem, D. (eds.) *DEON 2012, LNAI 7393*, pp. 91–107. Springer, Heidelberg (2012)
13. Gabbay, D.M., Hogger, C.J., Robinson, J.A. eds.: *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 3: Nonmonotonic Reasoning and Uncertain Reasoning. Oxford University Press (1994)
14. Gabbay, D.M.: Reactive Kripke models and contrary-to-duty obligations. Expanded version, original version 2008, revised 2012 into two parts (Part A: Semantics, to appear in *Journal of Applied Logic*; Part B: Proof Theory, submitted to *Journal of Applied Logic*)
15. Gabbay, D.: Bipolar argumentation frames and contrary to duty obligations, a position paper. In: Fisher, M. (eds.) *LNAI 7486*, pp. 1–24. Springer, Berlin (2012)
16. Gabbay, D.M., Kurucz, A., Wolter, F., Zakharyashev, M.: *Many Dimensional Modal Logics*. Elsevier, Amsterdam (2004)

¹¹Think of the numerous detective novels where the murderer tries to make the murder look like an accident, and how the detective can show it was not.

17. Gabbay, D., Marcelino, S.: Modal logics of reactive frames. *Stud. Log.* **93**, 403–444 (2009)
18. Gabbay, D.M., Shehtman, V.: Flow products of modal logics. Draft (2000). Incorporated in [18]. Expanded version in draft
19. Lifschitz, V.: Circumscription. In [13], pp. 297–352 (1994)
20. Rohde, P.: Moving in a crumbling network: the balanced case. In: Marcinkowski, J., Tarlecki, A. (eds.) *CSL 2004, LNCS*, pp. 1–25. Springer, Berlin (2004)
21. van Benthem, J.: An essay on sabotage and obstruction. In: Hutter, D., Werner, S. (eds.) *Mechanising Mathematical Reasoning. Festschrift in honour of Professor J. Siekmann, LNAI 2605*, pp. 268–276. Springer, Berlin (2005)
22. van Benthem, J., van Eijck, J., Kooi, B.: Logics of communication and change. *Inf. Comput.* **204**(11), 1620–1662 (2006)