Reasoning and Formal Modelling for Forensic Science
Lecture 10

Prof. Dr. Benedikt Löwe

2nd Semester 2010/11

A reminder from Lecture 6.

Adding a temporal dimension.

In many cases, our information changes over time. Further investigation of the situation reveals more values of 'Yes' and 'No', where previously we only had '?'. (Or, preferably not too often, reveals that some of our 'Yes' and 'No' values were false.)

We can see such a course of investigation as a sequence of partial situations where consistency changes values depending on what the current state of information is.

This is a first glimpse of how to include temporal information into the modelling (later in the course).

Revisiting the “Hit and Run” example.

In the document “Semantics for partially controlled situations and the ‘Hit and Run’ example”, we went through the story “Hit and Run” in all detail (available on the webpage).

Revisiting the "Hit and Run" example.

In the document "Semantics for partially controlled situations and the 'Hit and Run' example", we went through the story "Hit and Run" in all detail (available on the webpage).

The story is analysed in six stages, during which the partially controlled situations change based on the new information that we get at the stages of the narrative.

Revisiting the "Hit and Run" example.

In the document "Semantics for partially controlled situations and the 'Hit and Run' example", we went through the story "Hit and Run" in all detail (available on the webpage).

The story is analysed in six stages, during which the partially controlled situations change based on the new information that we get at the stages of the narrative.

> *"In stage 5, something more complicated happens. Moore's new story about the driver switch after the accident forces us to change the setting of the modelling: we now need to have two relations 'driving the car at the time of the accident' and 'being the last driver of the car'."*

Revisiting the "Hit and Run" example.

In the document "Semantics for partially controlled situations and the 'Hit and Run' example", we went through the story "Hit and Run" in all detail (available on the webpage).

The story is analysed in six stages, during which the partially controlled situations change based on the new information that we get at the stages of the narrative.

> *"In stage 5, something more complicated happens. Moore's new story about the driver switch after the accident forces us to change the setting of the modelling: we now need to have two relations 'driving the car at the time of the accident' and 'being the last driver of the car'."*

We replaced the original relation DRIVE by two temporally distinct relations DRIVEACCIDENT and DRIVELAST.

Important methodological note (1).

Important methodological note (1).

The action of replacing one of the original relations from the situations S_1, S_2, S_3, and S_4 is a radical modification of the formal setting.

Important methodological note (1).

The action of replacing one of the original relations from the situations S_1, S_2, S_3, and S_4 is a radical modification of the formal setting.

It is not only reflecting information change on the level of the investigators in the narrative (which would be formally represented by changes in the values of "Yes", "No" or "?" in the situations), but also reflecting an information change for the modeller who has to revise the set-up of the model.

Important methodological note (2).

Important methodological note (2).

- This is a very common situation: if you start building your model based on partial information about the narrative, you are bound to make modelling decisions that will not work in later stages.

Important methodological note (2).

- This is a very common situation: if you start building your model based on partial information about the narrative, you are bound to make modelling decisions that will not work in later stages.
- So, in the actual work of the modeller, this is something to be taken care of.

Important methodological note (2).

- This is a very common situation: if you start building your model based on partial information about the narrative, you are bound to make modelling decisions that will not work in later stages.
- So, in the actual work of the modeller, this is something to be taken care of.
- The disadvantage is that the situations S_1, S_2, S_3 and S_4 are not comparable anymore to S_5 since they are expressed in different formal languages.

Important methodological note (2).

- This is a very common situation: if you start building your model based on partial information about the narrative, you are bound to make modelling decisions that will not work in later stages.
- So, in the actual work of the modeller, this is something to be taken care of.
- The disadvantage is that the situations S_1, S_2, S_3 and S_4 are not comparable anymore to S_5 since they are expressed in different formal languages.
- So, in a cleaned version of the model, we would need to go back to S_1 and change our formal language in order to reflect the additional information we received as modellers in later stages.

Bruner's Spiral Curriculum.

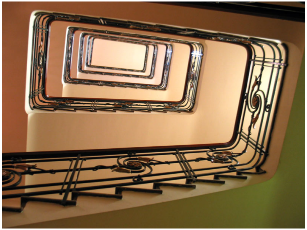
Bruner's Spiral Curriculum.

Jerome Bruner (b. 1915)

Bruner's Spiral Curriculum.

Jerome Bruner (b. 1915)

We learn by constantly revisiting (and possibly revising) past learning actions:



Spiral Modelling.

Spiral Modelling.

Prof. Dr. Benedikt Löwe

In analogy, when we are modelling, we have to go through the data that we are representing several times, each time rethinking our past decisions, and possibly revising them.

Spiral Modelling.

In analogy, when we are modelling, we have to go through the data that we are representing several times, each time rethinking our past decisions, and possibly revising them.

The final version of the formal representation of a narrative should be phrased in one single language so that you can compare the controlled situations at the various stages.

Two approaches to formal modelling (1).

Two approaches to formal modelling (1).

- “online” and
- “offline”.

Two approaches to formal modelling (1).

- "online" and
- "offline".

"online". If you want to design a system that can deal with changes that happen after you finalize the formal language, you want to have a very rich language that can react to new situations. You would **not** want to fix individuals, properties and relations in advance, because new individuals might show up during your work.

Two approaches to formal modelling (1).

- "online" and
- "offline".

"online". If you want to design a system that can deal with changes that happen after you finalize the formal language, you want to have a very rich language that can react to new situations. You would **not** want to fix individuals, properties and relations in advance, because new individuals might show up during your work.

"Online" systems are needed for software that is general in nature and should apply to many cases, or software that is doing analyses of ongoing cases.

Two approaches to formal modelling (1).

- "online" and
- "offline".

"online". If you want to design a system that can deal with changes that happen after you finalize the formal language, you want to have a very rich language that can react to new situations. You would **not** want to fix individuals, properties and relations in advance, because new individuals might show up during your work.

"Online" systems are needed for software that is general in nature and should apply to many cases, or software that is doing analyses of ongoing cases. They tend to be general and abstract.

Two approaches to formal modelling (1).

- "online" and
- "offline".

"online". If you want to design a system that can deal with changes that happen after you finalize the formal language, you want to have a very rich language that can react to new situations. You would **not** want to fix individuals, properties and relations in advance, because new individuals might show up during your work.

"Online" systems are needed for software that is general in nature and should apply to many cases, or software that is doing analyses of ongoing cases. They tend to be general and abstract.

"offline". If you have the entire story at your disposal, you can do *finite narrative modelling*: you read the entire narrative in advance and design a concrete and specific system that deals with all of the relevant aspects of the narrative.

Two approaches to formal modelling (2).

- “online” and
- “offline”.

Our controlled situations are examples of “offline modelling”. We have all of the information at our disposal and produce a concrete and specific system to represent it.

Two approaches to formal modelling (2).

- "online" and
- "offline".

Our controlled situations are examples of "offline modelling". We have all of the information at our disposal and produce a concrete and specific system to represent it.

However, actually designing such a system is using a bit of both methodologies. While you go along the narrative and make modelling decisions to include elements in your system, you expand your language until you reach the end of the narrative.

Two approaches to formal modelling (2).

- “online” and
- “offline”.

Our controlled situations are examples of “offline modelling”. We have all of the information at our disposal and produce a concrete and specific system to represent it.

However, actually designing such a system is using a bit of both methodologies. While you go along the narrative and make modelling decisions to include elements in your system, you expand your language until you reach the end of the narrative.

In the process of *spiral modelling*, you then go back and re-assess the decisions you made earlier in order to get a homogeneous “offline” language and representation.

Partially Controlled Situation Sequences.

Partially Controlled Situation Sequences.

Prof. Dr. Benedikt Löwe

A partially controlled situation sequence consists of a finite number of *moments* t_1, ..., t_n, a fixed collection of individuals, properties and relations, and for each moment i, a partially controlled situation with relations S_i with these individuals, properties and relations.

Partially Controlled Situation Sequences.

A partially controlled situation sequence consists of a finite number of *moments* t_1, ..., t_n, a fixed collection of individuals, properties and relations, and for each moment i, a partially controlled situation with relations S_i with these individuals, properties and relations.

The semantics at each given moment t_i is the usual semantics for partially controlled situations defining

$$\varphi \text{ is valid in } S_i$$

and

$$\varphi \text{ is invalid in } S_i.$$

Partially Controlled Situation Sequences.

A partially controlled situation sequence consists of a finite number of *moments* t_1, ..., t_n, a fixed collection of individuals, properties and relations, and for each moment i, a partially controlled situation with relations S_i with these individuals, properties and relations.

The semantics at each given moment t_i is the usual semantics for partially controlled situations defining

$$\varphi \text{ is valid in } S_i$$

and

$$\varphi \text{ is invalid in } S_i.$$

Now we are able to express additional temporal information.

Temporal Information (1).

Prof. Dr. Benedikt Löwe

Temporal Information (1).

- (In)valid *at* t_i.

Temporal Information (1).

- (In)valid *at* t_i.
- (In)valid *until* t_i.

Temporal Information (1).

- (In)valid *at* t_i.
- (In)valid *until* t_i.
- (In)valid *before* t_i.

Temporal Information (1).

Prof. Dr. Benedikt Löwe

- (In)valid *at* t_i.
- (In)valid *until* t_i.
- (In)valid *before* t_i.
- (In)valid *since* t_i.

Temporal Information (1).

- (In)valid *at* t_i.
- (In)valid *until* t_i.
- (In)valid *before* t_i.
- (In)valid *since* t_i.
- (In)valid *after* t_i.

Temporal Information (1).

- (In)valid *at* t_i.
- (In)valid *until* t_i.
- (In)valid *before* t_i.
- (In)valid *since* t_i.
- (In)valid *after* t_i.

We can introduce symbols for these: $@_i$, until_i, before_i, since_i, and $\mathrm{after}_\mathrm{i}$.

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* $\mathrm{after}_\mathrm{i}$*:*

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* $\mathrm{after}_\mathrm{i}$*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* $\mathrm{after}_\mathrm{i}$*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

- $@_i\varphi$ is valid in S if φ is valid in S_i.

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* $\mathrm{after_i}$*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

- $@_i\varphi$ is valid in S if φ is valid in S_i.
- $@_i\varphi$ is invalid in S if φ is not valid in S_i.

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* after_i*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

- $@_i\varphi$ is valid in S if φ is valid in S_i.
- $@_i\varphi$ is invalid in S if φ is not valid in S_i.
- $\text{until}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i$.

Temporal Information (2).

Prof. Dr. Benedikt Löwe

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* after_i*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

- $@_i\varphi$ is valid in S if φ is valid in S_i.
- $@_i\varphi$ is invalid in S if φ is not valid in S_i.
- $\text{until}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i$.
- $\text{before}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i - 1$.

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* $\mathrm{after}_\mathrm{i}$*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

- $@_i\varphi$ is valid in S if φ is valid in S_i.
- $@_i\varphi$ is invalid in S if φ is not valid in S_i.
- $\mathrm{until}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i$.
- $\mathrm{before}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i - 1$.
- $\mathrm{until}_i\varphi$ is invalid in S if φ is not valid in S_j for some $j = 1, ..., i$.

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* $\mathrm{after}_\mathrm{i}$*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

- $@_i\varphi$ is valid in S if φ is valid in S_i.
- $@_i\varphi$ is invalid in S if φ is not valid in S_i.
- $\mathrm{until}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i$.
- $\mathrm{before}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i-1$.
- $\mathrm{until}_i\varphi$ is invalid in S if φ is not valid in S_j for some $j = 1, ..., i$.
- $\mathrm{before}_i\varphi$ is invalid in S if φ is not valid in S_j for some $j = 1, ..., i-1$.

Temporal Information (2).

Formal definitions of the semantics of $@_i$, until_i, before_i, since_i, *and* $\mathrm{after}_\mathrm{i}$*:*

We fix a partially controlled situation sequence $S = (S_1, ..., S_n)$.

- $@_i\varphi$ is valid in S if φ is valid in S_i.
- $@_i\varphi$ is invalid in S if φ is not valid in S_i.
- $\mathrm{until}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i$.
- $\mathrm{before}_i\varphi$ is valid in S if φ is valid in S_j for all $j = 1, ..., i - 1$.
- $\mathrm{until}_i\varphi$ is invalid in S if φ is not valid in S_j for some $j = 1, ..., i$.
- $\mathrm{before}_i\varphi$ is invalid in S if φ is not valid in S_j for some $j = 1, ..., i - 1$.
- $\mathrm{since}_i\varphi$ is valid in S if φ is valid in S_j for all $j = i, ..., n$.
- $\mathrm{after}_i\varphi$ is valid in S if φ is valid in S_j for all $j = i + 1, ..., n$.
- $\mathrm{since}_i\varphi$ is invalid in S if φ is not valid in S_j for some $j = i, ..., n$.
- $\mathrm{after}_i\varphi$ is invalid in S if φ is not valid in S_j for some $j = i + 1, ..., n$.

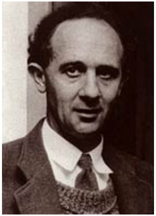
Modern history of temporal logic.

Modern history of temporal logic.

Considerations of temporal logic go back to Aristotle (next lecture),

Modern history of temporal logic.

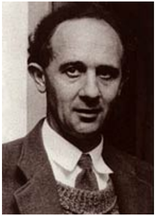
Considerations of temporal logic go back to Aristotle (next lecture), but modern developments started with



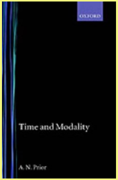
Arthur Prior (1914–1969)

Modern history of temporal logic.

Considerations of temporal logic go back to Aristotle (next lecture), but modern developments started with



Arthur Prior (1914–1969)



A. Prior, Time and Modality, Oxford University Press 1957

Hybrid Logic

Prof. Dr. Benedikt Löwe

Hybrid Logic

Torben Braüner (2008). "Hybrid Logic". Stanford Encyclopedia of Philosophy.

Hybrid Logic

Torben Braüner (2008). "Hybrid Logic". Stanford Encyclopedia of Philosophy.

Original ideas go back to Prior, and have been formalized in

Bull, R. 1970. An Approach to Tense Logic. *Theoria*, 36: 282–300.

Hybrid Logic

Torben Braüner (2008). "Hybrid Logic". Stanford Encyclopedia of Philosophy.

Original ideas go back to Prior, and have been formalized in

Bull, R. 1970. An Approach to Tense Logic. *Theoria*, 36: 282–300.

Independently discovered by the Bulgarian school (Passy and Tinchev), and developed by Goranko, Blackburn and others.

Hybrid Logic

Torben Braüner (2008). "Hybrid Logic". Stanford Encyclopedia of Philosophy.

Original ideas go back to Prior, and have been formalized in

Bull, R. 1970. An Approach to Tense Logic. *Theoria*, 36: 282–300.

Independently discovered by the Bulgarian school (Passy and Tinchev), and developed by Goranko, Blackburn and others.

ten Cate, B. 2004. Model Theory for Extended Modal Languages. Ph.D. thesis, Institute for Logic, Language and Computation, University of Amsterdam.

Revisiting "Hit and Run" (1).

Revisiting "Hit and Run" (1).

When "something more complicated" happened, we realized that we need a temporal component that has at least two moments: the time before the accident t_{before} and the time after the accident t_{after}.

Revisiting "Hit and Run" (1).

When "something more complicated" happened, we realized that we need a temporal component that has at least two moments: the time before the accident t_{before} and the time after the accident t_{after}.

After spiralling once, we realize (during the modelling of stage 5) that we made a mistake by not including temporal information. We fix this mistake now:

Revisiting "Hit and Run" (1).

When "something more complicated" happened, we realized that we need a temporal component that has at least two moments: the time before the accident t_{before} and the time after the accident t_{after}.

After spiralling once, we realize (during the modelling of stage 5) that we made a mistake by not including temporal information. We fix this mistake now:

We had started with

Situation S_1 consists of the individuals m (Charles Moore), c (the car), and u (an unknown driver). We include the unknown driver in order to be able to express that someone else drove Moore's car. We use the properties STOLEN *and* KILLER *and the relation* DRIVE, *standing for "was stolen", "is the killer of the girl", and "was driving at the time of the accident". The semantics of this partially controlled situation is given by:*

	STOLEN	KILLER
m	*No*	*?*
c	*?*	*No*
u	*No*	*?*

DRIVE	*m*	*c*	*u*
m	*No*	*?*	*No*
c	*No*	*No*	*No*
u	*No*	*?*	*No*

Revisiting "Hit and Run" (2).

Situation S_1 consists of the individuals m (Charles Moore), c (the car), and u (an unknown driver). We include the unknown driver in order to be able to express that someone else drove Moore's car. We use the properties STOLEN *and* KILLER *and the relation* DRIVE, *standing for "was stolen", "is the killer of the girl", and "was driving at the time of the accident". The semantics of this partially controlled situation is given by:*

	STOLEN	KILLER
m	*No*	*?*
c	*?*	*No*
u	*No*	*?*

DRIVE	*m*	*c*	*u*
m	*No*	*?*	*No*
c	*No*	*No*	*No*
u	*No*	*?*	*No*

Revisiting "Hit and Run" (2).

Situation S_1 consists of the individuals m (Charles Moore), c (the car), and u (an unknown driver). We include the unknown driver in order to be able to express that someone else drove Moore's car. We use the properties STOLEN *and* KILLER *and the relation* DRIVE, *standing for "was stolen", "is the killer of the girl", and "was driving at the time of the accident". The semantics of this partially controlled situation is given by:*

	STOLEN	KILLER
m	*No*	*?*
c	*?*	*No*
u	*No*	*?*

DRIVE	*m*	*c*	*u*
m	*No*	*?*	*No*
c	*No*	*No*	*No*
u	*No*	*?*	*No*

And we ran into trouble in stage 5 with

"In stage 5, something more complicated happens. Moore's new story about the driver switch after the accident forces us to change the setting of the modelling: we now need to have two relations 'driving the car at the time of the accident' and 'being the last driver of the car'." ... Also, we can now get rid of the individual u, since we know that this is about James.

Revisiting "Hit and Run" (2).

Situation S_1 consists of the individuals m (Charles Moore), c (the car), and u (an unknown driver). We include the unknown driver in order to be able to express that someone else drove Moore's car. We use the properties STOLEN *and* KILLER *and the relation* DRIVE, *standing for "was stolen", "is the killer of the girl", and "was driving at the time of the accident". The semantics of this partially controlled situation is given by:*

	STOLEN	KILLER
m	*No*	*?*
c	*?*	*No*
u	*No*	*?*

DRIVE	*m*	*c*	*u*
m	*No*	*?*	*No*
c	*No*	*No*	*No*
u	*No*	*?*	*No*

And we ran into trouble in stage 5 with

"In stage 5, something more complicated happens. Moore's new story about the driver switch after the accident forces us to change the setting of the modelling: we now need to have two relations 'driving the car at the time of the accident' and 'being the last driver of the car'." ... Also, we can now get rid of the individual u, since we know that this is about James.

We use this information in our second round of the spiral to replace u by j throughout the stages, and by introducing the temporal component with the two moments t_{before} and t_{after}. With this information, we spiral back to S_1.

Revisiting “Hit and Run” (3).

Revisiting "Hit and Run" (3).

We now have to give two situations, S_1^{before} and S_1^{after}.

Revisiting "Hit and Run" (3).

We now have to give two situations, S_1^{before} and S_1^{after}. In both situations, we have the individuals m (Charles Moore), c (the car), and j (James Moore). We use the properties STOLEN and KILLER and the relation DRIVE.

Revisiting "Hit and Run" (3).

We now have to give two situations, S_1^{before} and S_1^{after}. In both situations, we have the individuals m (Charles Moore), c (the car), and j (James Moore). We use the properties STOLEN and KILLER and the relation DRIVE. The semantics of this partially controlled situation is given by:

before	STOLEN	KILLER
m	No	No
c	?	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	?	No
j	No	?

Revisiting "Hit and Run" (3).

We now have to give two situations, S_1^{before} and S_1^{after}. In both situations, we have the individuals m (Charles Moore), c (the car), and j (James Moore). We use the properties STOLEN and KILLER and the relation DRIVE. The semantics of this partially controlled situation is given by:

before	STOLEN	KILLER
m	No	No
c	?	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	?	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

Revisiting "Hit and Run" (3).

We now have to give two situations, S_1^{before} and S_1^{after}. In both situations, we have the individuals m (Charles Moore), c (the car), and j (James Moore). We use the properties STOLEN and KILLER and the relation DRIVE. The semantics of this partially controlled situation is given by:

before	STOLEN	KILLER
m	No	No
c	?	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	?	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\text{before}} \text{DRIVE}(y, c) \wedge \exists z @_{\text{after}} \text{DRIVE}(z, c)$.

Revisiting "Hit and Run" (3).

We now have to give two situations, S_1^{before} and S_1^{after}. In both situations, we have the individuals m (Charles Moore), c (the car), and j (James Moore). We use the properties STOLEN and KILLER and the relation DRIVE. The semantics of this partially controlled situation is given by:

before	STOLEN	KILLER
m	No	No
c	?	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	?	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\mathrm{before}}\mathrm{DRIVE}(y, c) \wedge \exists z @_{\mathrm{after}}\mathrm{DRIVE}(z, c)$.

ϱ_1 $@_{\mathrm{before}}\mathrm{STOLEN}(c) \rightarrow \neg @_{\mathrm{before}}\mathrm{DRIVE}(m, c)$.

Revisiting "Hit and Run" (3).

We now have to give two situations, S_1^{before} and S_1^{after}. In both situations, we have the individuals m (Charles Moore), c (the car), and j (James Moore). We use the properties STOLEN and KILLER and the relation DRIVE. The semantics of this partially controlled situation is given by:

before	STOLEN	KILLER
m	No	No
c	?	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	?	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\text{before}}\text{DRIVE}(y, c) \wedge \exists z @_{\text{after}}\text{DRIVE}(z, c)$.

ϱ_1 $@_{\text{before}}\text{STOLEN}(c) \rightarrow \neg @_{\text{before}}\text{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\text{before}}\text{DRIVE}(x, c) \rightarrow @_{\text{after}}\text{KILLER}(x)$.

Revisiting "Hit and Run" (4).

before	STOLEN	KILLER
m	No	No
c	?	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	?	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\text{before}} \text{DRIVE}(y, c) \wedge \exists z @_{\text{after}} \text{DRIVE}(z, c)$.

ϱ_1 $@_{\text{before}} \text{STOLEN}(c) \rightarrow \neg @_{\text{before}} \text{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\text{before}} \text{DRIVE}(x, c) \rightarrow @_{\text{after}} \text{KILLER}(x)$.

Revisiting “Hit and Run” (4).

before	STOLEN	KILLER
m	No	No
c	?	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	?	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\mathrm{before}} \mathrm{DRIVE}(y, c) \wedge \exists z @_{\mathrm{after}} \mathrm{DRIVE}(z, c)$.

ϱ_1 $@_{\mathrm{before}} \mathrm{STOLEN}(c) \rightarrow \neg @_{\mathrm{before}} \mathrm{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\mathrm{before}} \mathrm{DRIVE}(x, c) \rightarrow @_{\mathrm{after}} \mathrm{KILLER}(x)$.

These rules will be in place during the entire investigation and act as our consistency check.

Revisiting "Hit and Run" (4).

before	STOLEN	KILLER
m	No	No
c	No	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	No	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\text{before}} \text{DRIVE}(y, c) \wedge \exists z @_{\text{after}} \text{DRIVE}(z, c)$.

ϱ_1 $@_{\text{before}} \text{STOLEN}(c) \to \neg @_{\text{before}} \text{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\text{before}} \text{DRIVE}(x, c) \to @_{\text{after}} \text{KILLER}(x)$.

These rules will be in place during the entire investigation and act as our consistency check.

When we move to stage 2, we change the values of STOLEN for both S_2^{before} and S_2^{after}.

Revisiting "Hit and Run" (4).

before	STOLEN	KILLER
m	No	No
c	No	No
j	No	No

after	STOLEN	KILLER
m	No	Yes
c	No	No
j	No	No

before

DRIVE	m	c	j
m	No	Yes	No
c	No	No	No
j	No	No	No

after

DRIVE	m	c	j
m	No	Yes	No
c	No	No	No
j	No	No	No

ϱ_0 $\exists y @_{\mathrm{before}} \mathrm{DRIVE}(y, c) \wedge \exists z @_{\mathrm{after}} \mathrm{DRIVE}(z, c)$.

ϱ_1 $@_{\mathrm{before}} \mathrm{STOLEN}(c) \rightarrow \neg @_{\mathrm{before}} \mathrm{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\mathrm{before}} \mathrm{DRIVE}(x, c) \rightarrow @_{\mathrm{after}} \mathrm{KILLER}(x)$.

These rules will be in place during the entire investigation and act as our consistency check.

When we move to stage 2, we change the values of STOLEN for both S_2^{before} and S_2^{after}.

Similarly, when we move to stage 3, we remove all of the question marks.

Revisiting "Hit and Run" (4).

before	STOLEN	KILLER
m	No	No
c	No	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	No	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\text{before}} \text{DRIVE}(y, c) \wedge \exists z @_{\text{after}} \text{DRIVE}(z, c)$.

ϱ_1 $@_{\text{before}} \text{STOLEN}(c) \rightarrow \neg @_{\text{before}} \text{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\text{before}} \text{DRIVE}(x, c) \rightarrow @_{\text{after}} \text{KILLER}(x)$.

These rules will be in place during the entire investigation and act as our consistency check.

When we move to stage 2, we change the values of STOLEN for both S_2^{before} and S_2^{after}.

Similarly, when we move to stage 3, we remove all of the question marks. Stage 4 is just stage 2 again.

Revisiting "Hit and Run" (5).

before	STOLEN	KILLER
m	No	No
c	No	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	No	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

ϱ_0 $\exists y @_{\text{before}} \text{DRIVE}(y, c) \wedge \exists z @_{\text{after}} \text{DRIVE}(z, c)$.

ϱ_1 $@_{\text{before}} \text{STOLEN}(c) \rightarrow \neg @_{\text{before}} \text{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\text{before}} \text{DRIVE}(x, c) \rightarrow @_{\text{after}} \text{KILLER}(x)$.

Revisiting "Hit and Run" (5).

before	STOLEN	KILLER
m	No	No
c	No	No
j	No	No

after	STOLEN	KILLER
m	No	?
c	No	No
j	No	?

before

DRIVE	m	c	j
m	No	?	No
c	No	No	No
j	No	?	No

after

DRIVE	m	c	j
m	No	No	No
c	No	No	No
j	No	Yes	No

ϱ_0 $\exists y @_{\text{before}} \text{DRIVE}(y, c) \wedge \exists z @_{\text{after}} \text{DRIVE}(z, c)$.

ϱ_1 $@_{\text{before}} \text{STOLEN}(c) \rightarrow \neg @_{\text{before}} \text{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\text{before}} \text{DRIVE}(x, c) \rightarrow @_{\text{after}} \text{KILLER}(x)$.

In stage 5, we finally use the temporal structure in a meaningful way. We learn change the after values of DRIVE. Still, rule ϱ_2 is consistent with Charles Moore being the killer.

Revisiting "Hit and Run" (5).

before	STOLEN	KILLER
m	No	No
c	No	No
j	No	No

after	STOLEN	KILLER
m	No	No
c	No	No
j	No	Yes

before

DRIVE	m	c	j
m	No	No	No
c	No	No	No
j	No	Yes	No

after

DRIVE	m	c	j
m	No	No	No
c	No	No	No
j	No	Yes	No

ϱ_0 $\exists y @_{\mathrm{before}} \mathrm{DRIVE}(y, c) \wedge \exists z @_{\mathrm{after}} \mathrm{DRIVE}(z, c)$.

ϱ_1 $@_{\mathrm{before}} \mathrm{STOLEN}(c) \rightarrow \neg @_{\mathrm{before}} \mathrm{DRIVE}(m, c)$.

ϱ_2 $\forall x @_{\mathrm{before}} \mathrm{DRIVE}(x, c) \rightarrow @_{\mathrm{after}} \mathrm{KILLER}(x)$.

In stage 5, we finally use the temporal structure in a meaningful way. We learn change the after values of DRIVE. Still, rule ϱ_2 is consistent with Charles Moore being the killer.

Finally, in stage 6, every question mark is resolved.

Another example.

Stage 1. *The police find Jean Bartington dead in her office with a knife in her back. The investigation of the crime scene shows that the murderer must have had a key to her office. There are only two people (except for Jean) who have a key: her secretary Paul and the building administrator Sheila. Paul was the person who found the body.*

Stage 2. *The forensic investigation finds fingerprints of Paul and Sheila on the knife.*

Stage 3. *The investigation shows that Paul's fingerprints resulted from him touching the knife when he found the body.*