Математическая логика

Лектор:

Подымов Владислав Васильевич

e-mail:
valdus@yandex.ru

2018, весенний семестр

Лекция 14–15

Модальные логики

Эпистемические логики

Темпоральные логики

Логика линейного времени (LTL)

Логика деревьев вычислений (CTL)

Лекция 14–15

Верификация распределённых систем

Задача model checking для LTL

Табличный алгоритм model checking для LTL

Модальные логики

Рассмотрим такие высказывания:

1: **Зима близко**

2: **Зима всегда близко**

3: **Зима бывает близко**

Если отбросить всё лишнее, то для высказывания *1* справедлива одна из двух оценок: **правда**, если зима действительно близко, и **неправда**, если это не так

Смысл высказываний *2* и *3* *тесно связан* со смыслом высказывания *1*: если $x =$ “зима близко”, то

1: x *2:* всегда x *3:* иногда бывает x

Модальные логики

Рассмотрим такие высказывания:

1: **Зима близко**

2: **Зима всегда близко**

3: **Зима бывает близко**

Слова, используемые как уточнение истинностной оценки высказывания, называются модальностями (*лат. modus — мера*)

"Всегда" и "иногда" — это темпоральные модальности (модальности времени) (*лат. tempus — время*)

Что мешает сказать, что "всегда" и "иногда" — это $\forall$ и $\exists$, и использовать логику предикатов для записи этих высказываний?

Модальные логики

Рассмотрим такие высказывания:

1: **Зима близко**

2: **Известно, что зима близко**

3: **Можно допустить, что зима близко**

Смысл высказываний *2* и *3* *тесно связан* со смыслом высказывания *1*: если $x =$ "зима близко", то

1: x *2:* известно, что x *3:* допустимо x

"Известно" и "допустимо" — это эпистемические модальности (модальности знания) (*др.-греч.* *ἐπιστήμη* — *знание*)

Как записать эти высказывания, "известно" и "допустимо" записывались как $\forall$ и $\exists$?

Модальные логики

Рассмотрим такие высказывания:

1: **Зима близко**

2: **Зима должна быть близко**

3: **Зима имеет право быть близко**

Смысл высказываний *2* и *3* *тесно связан* со смыслом высказывания *1*: если $x =$ "зима близко", то

1: x *2:* должно быть x *3:* имеет право быть x

"Должен" и "имеет право" — это деонтические модальности (модальности долга) (*др.-греч.* δέον — должное)

А можно ли использовать $\forall$ и $\exists$ в смысле "должен" и "имеет право"?

Модальные логики

Модальность — это выражение, описывающее "*оттенок истинности*" высказывания (уверенность, необходимость, доказуемость, осведомлённость, . . .)

Чаще всего в суждениях используются модальности двух двойственных видов:

Модальность необходимого	**Модальность возможного**
необходимо	возможно
обязательно	не исключено
всегда	иногда
должен	имеет право
знает	предполагает
доказуемо	непротиворечиво
$\Box$	$\Diamond$

Таких модальностей можно предложить сколь угодно много, но при есть способ единообразного определения смысла модальностей: в терминах модальной логики

Модальные логики

Синтаксис модальных формул:

$$\varphi ::= \mathtt{x} \mid (\neg\varphi) \mid (\varphi \,\&\, \varphi) \mid (\varphi \vee \varphi) \mid (\varphi \to \varphi) \mid (\Box\varphi) \mid (\Diamond\varphi)$$

($\mathtt{x}$ — пропозициональная переменная)

Это синтаксис формул логики высказываний, расширенный возможностью расстановки модальностей необходимого и возможного над любыми (под)формулами

Приоритет операций: $\neg$, $\Box$ и $\Diamond$; затем $\&$; затем $\vee$; затем $\to$

Пример формулы: $\Diamond\mathtt{x} \,\&\, \Box\neg\Diamond(\mathtt{x} \vee \mathtt{y})$

“*возможно* $\mathtt{x}$*, и при этом необходимо невозможно, что* $\mathtt{x}$ *или* $\mathtt{y}$”

В синтаксисе модальных формул **отсутствует** описание точного смысла модальностей: описание смысла формулы — это её семантика

Модальные логики

Прежде чем перейти к описанию семантики модальных формул, рассмотрим такой **пример:**

верна ли формула $\Box\varphi \to \varphi$**?**

Да, если $\Box$ — темпоральная модальность:

если зима всегда близко, то она близко

Нет, если $\Box$ — деонтическая модальность:

если зима должна быть близко, то она близко

В интерпретации модальной логики — математическом объекте, наделяющем формулу строгим смыслом — определяется не только значение пропозициональных переменных, но и точное значение модальности

Семантика Крипке

— это наиболее распространённый общий подход к описанию значения модальных формул

Пусть $\mathcal{P}$ — множество пропозициональных переменных

Тогда модель Крипке — это система (W, R, ξ), где

- W — множество состояний (возможных миров)
- $R \subseteq W \times W$ — отношение достижимости миров
- $\xi : W \to 2^{\mathcal{P}}$ — оценка переменных

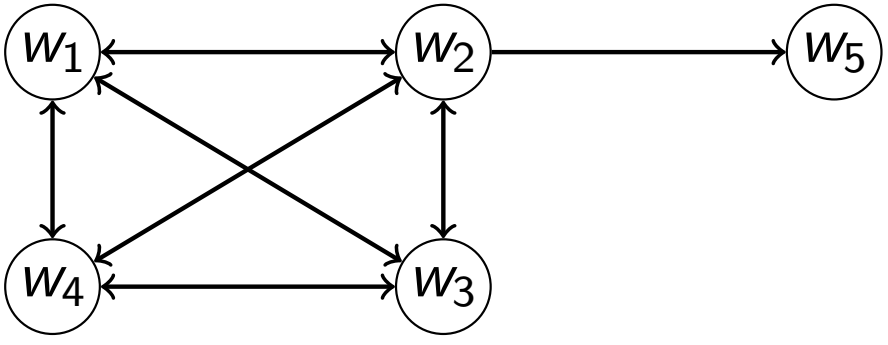
Шкала Крипке (*Kripke frame*), на которой основывается модель (W, R, ξ), — это пара (W, R)

Если $(w, w') \in R$, то w' — альтернативный мир для w (w-альтернатива)

Модель Крипке — это интерпретация модальной логики

Семантика Крипке

Например, $(\mathcal{P} = \{a\})$



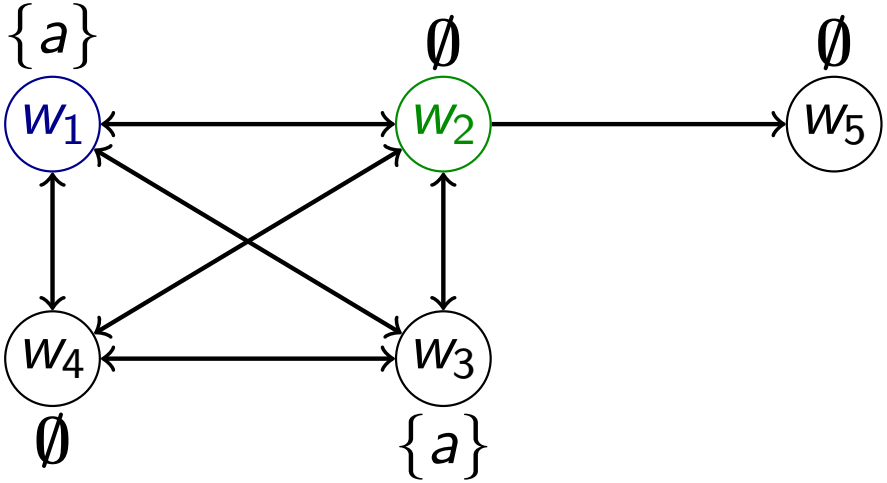
— это шкала Крипке

Семантика Крипке

Отношение выполнимости $\models$ для модели $\mathcal{I} = (W, R, \xi)$ и мира $w \in W$ определяется так:

- $\mathcal{I}, w \models p \quad \Leftrightarrow \quad p \in \xi(w)$ $\qquad (p \in \mathcal{P})$
- $\mathcal{I}, w \models \varphi \,\&\, \psi \quad \Leftrightarrow \quad \mathcal{I}, w \models \varphi$ и $\mathcal{I}, w \models \psi$
- $\mathcal{I}, w \models \varphi \vee \psi \quad \Leftrightarrow \quad \mathcal{I}, w \models \varphi$ или $\mathcal{I}, w \models \psi$

Например, $\qquad (\mathcal{P} = \{a\})$



— это модель Крипке $(\mathcal{I})$

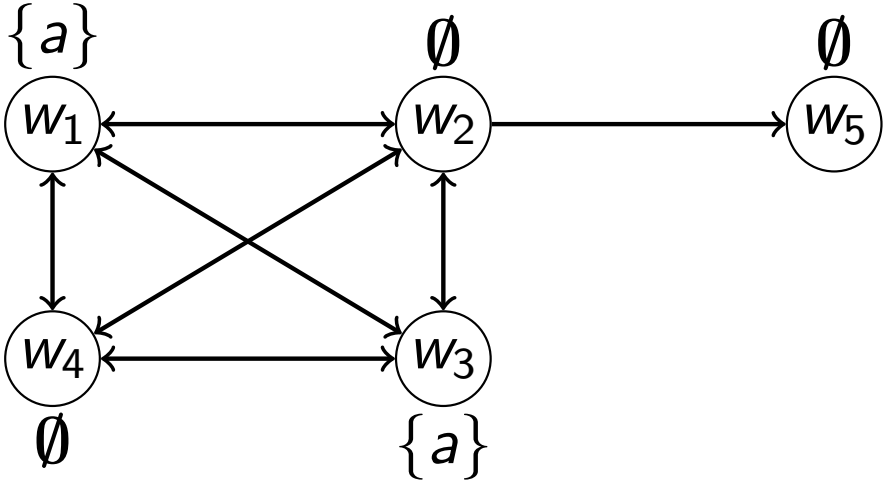
$\mathcal{I}, w_1 \models a, \qquad \mathcal{I}, w_2 \not\models a$

Семантика Крипке

Отношение выполнимости $\models$ для модели $\mathcal{I} = (W, R, \xi)$ и мира $w \in W$ определяется так:

- $\mathcal{I}, w \models \varphi \to \psi \quad \Leftrightarrow \quad \mathcal{I}, w \not\models \varphi$ или $\mathcal{I}, w \models \psi$
- $\mathcal{I}, w \models \neg\varphi \quad \Leftrightarrow \quad \mathcal{I}, w \not\models \varphi$

Например, $(\mathcal{P} = \{a\})$



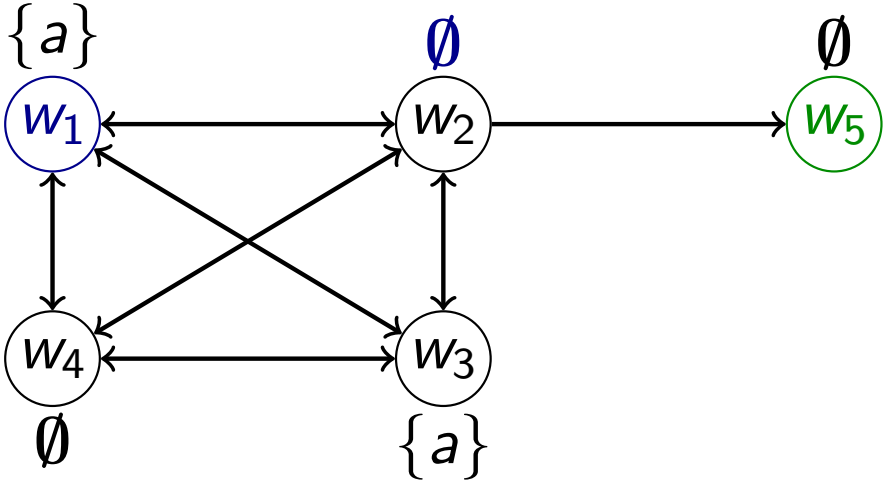
— это модель Крипке $(\mathcal{I})$

Семантика Крипке

Отношение выполнимости $\models$ для модели $\mathcal{I} = (W, R, \xi)$ и мира $w \in W$ определяется так:

- $\mathcal{I}, w \models \Box\varphi \quad \Leftrightarrow$
  для любой w-альтернативы w' верно $\mathcal{I}, w' \models \varphi$

Например, $(\mathcal{P} = \{a\})$



— это модель Крипке $(\mathcal{I})$

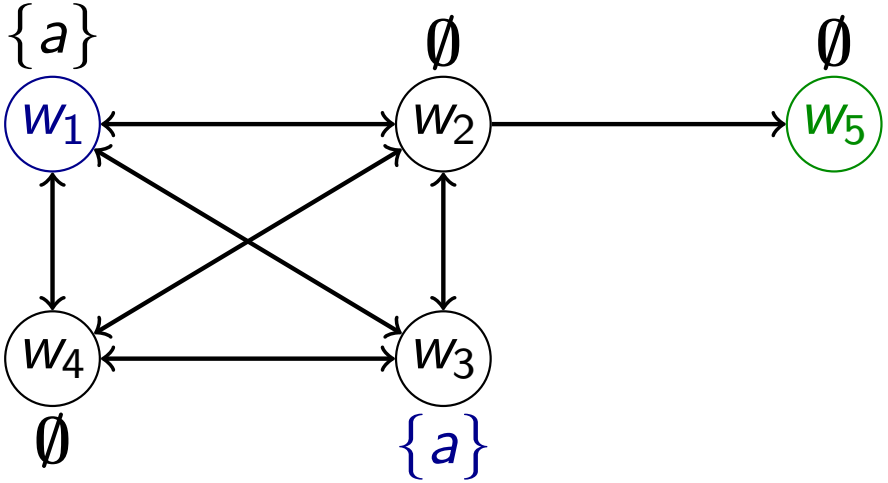
$\mathcal{I}, w_1 \not\models \Box a, \qquad \mathcal{I}, w_5 \models \Box a$

Семантика Крипке

Отношение выполнимости $\models$ для модели $\mathcal{I} = (W, R, \xi)$ и мира $w \in W$ определяется так:

- $\mathcal{I}, w \models \Diamond\varphi \quad \Leftrightarrow$
  существует w-альтернатива w',
  такая что верно $\mathcal{I}, w' \models \varphi$

Например, $(\mathcal{P} = \{a\})$

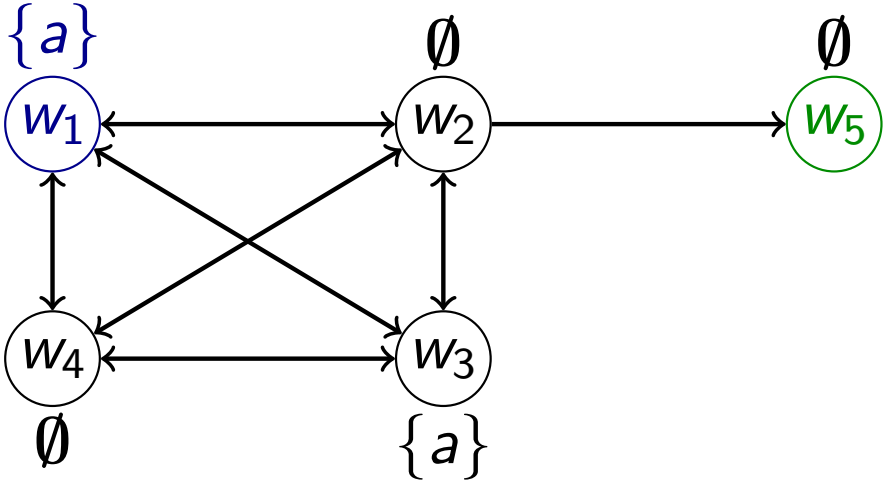


— это модель Крипке $(\mathcal{I})$

$\mathcal{I}, w_1 \models \Diamond a, \qquad \mathcal{I}, w_5 \not\models \Diamond a$

Семантика Крипке

Например, $(\mathcal{P} = \{a\})$

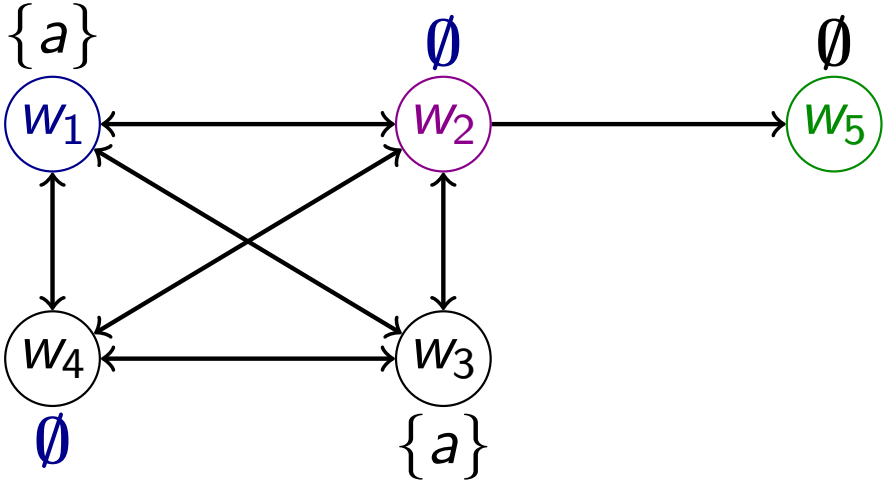


— это модель Крипке $(\mathcal{I})$

$\mathcal{I}, w_1 \models \Box\Diamond a,$ $\mathcal{I}, w_5 \models \Box\Diamond a$

Семантика Крипке

Например, $(\mathcal{P} = \{a\})$



— это модель Крипке $(\mathcal{I})$

$\mathcal{I}, w_1 \not\models \Diamond\Box a,$ $\mathcal{I}, w_2 \models \Diamond\Box a,$ $\mathcal{I}, w_5 \not\models \Diamond\Box a$

Семантика Крипке

Пусть $\mathcal{F}$ — шкала Крипке, $\mathcal{I}$ — модель Крипке, и φ, ψ — формулы модальной логики

Тогда

- формула φ истинна в модели $\mathcal{I}$ ($\mathcal{I} \models \varphi$), если для любого мира w модели $\mathcal{I}$ верно $\mathcal{I}, w \models \varphi$
- формула φ истинна на шкале $\mathcal{F}$ ($\mathcal{F} \models \varphi$), если для любой модели Крипке $\mathcal{J}$, основанной на $\mathcal{F}$, верно $\mathcal{J} \models \varphi$
- формула φ общезначима ($\models \varphi$), если для любой шкалы $\mathcal{F}$ верно $\mathcal{F} \models \varphi$
- формулы φ и ψ равносильны ($\varphi \approx \psi$), если $\models \varphi \leftrightarrow \psi$

Законы модальных логик Эпистемические логики

Модальная логика, как и любая другая, имеет свои законы и свойства, например:

$\boxed{\Diamond\varphi \approx \neg\Box\neg\varphi}$ $\boxed{\models \varphi \quad \Rightarrow \quad \models \Box\varphi}$ $\boxed{\models \Box(\varphi \to \psi) \to (\Box\varphi \to \Box\psi)}$

Смыслом модальностей могут определяться и другие законы

Например, в **эпистемической логике** модальности имеют смысл знания ($\Box$) и допущения ($\Diamond$), и для *идеального познающего субъекта* справедливы, в числе прочего, аксиомы, порождаемые следующими схемами:

- схема аксиом адекватности знания:
  $\Box\varphi \to \varphi$ (*мои знания верны*)
- схема аксиом позитивной интроспекции:
  $\Box\varphi \to \Box\Box\varphi$ (*мне известно, что именно я знаю*)
- схема аксиом негативной интроспекции:
  $\neg\Box\varphi \to \Box\neg\Box\varphi$ (*мне известно, что именно я не знаю*)

Эпистемические логики

Рассмотрим шкалу Крипке $\mathcal{F} = (W, R)$

Утверждение

$\mathcal{F} \models \Box\varphi \to \varphi$ **верно для любой формулы** φ

$$\Leftrightarrow$$

отношение R **рефлексивно**

Утверждение

$\mathcal{F} \models \Box\varphi \to \Box\Box\varphi$ **верно для любой формулы** φ

$$\Leftrightarrow$$

отношение R **транзитивно**

Утверждение. **Пусть отношение** R **транзитивно. Тогда**

$\mathcal{F} \models \neg\Box\varphi \to \Box\neg\Box\varphi$ **верно для любой формулы** φ

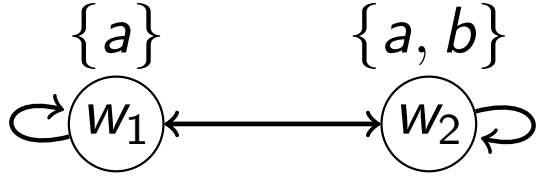
$$\Leftrightarrow$$

отношение R **симметрично**

Доказательство. Самостоятельно

Эпистемические логики

Для примера рассмотрим такую модель $\mathcal{I}$, основанную на шкале, подходящей для *идеального познающего субъекта*:



Проверка соотношения $\mathcal{I}, w_1 \models \varphi$ — это проверка того, какие факты верны, если субъект S живёт в мире w_1, и какие знания имеются у S, если он точно знает, что верно либо $\{a\}$, либо $\{a, b\}$ и другого быть не может, например:

- a *и* $\neg b$ *— это верные факты*:
$$\mathcal{I}, w_1 \models a, \quad \mathcal{I}, w_1 \models \neg b$$
- S *знает, что* a *верно,*
и допускает, что верным может быть как b*, так и* $\neg b$:
$$\mathcal{I}, w_1 \models \Box a, \quad \mathcal{I}, w_1 \models \Diamond b, \quad \mathcal{I}, w_1 \models \Diamond \neg b$$
- S *не знает, верно ли* b*, но знает, что он этого не знает*:
$$\mathcal{I}, w_1 \not\models \Box b, \quad \mathcal{I}, w_1 \not\models \Box \neg b, \quad \mathcal{I}, w_1 \models \Box(\neg \Box b \,\&\, \neg \Box \neg b)$$

Эпистемические логики

Задача о трёх мудрецах

Король призвал трёх мудрецов, показал им три чёрные шапки и две белые, завязал глаза, надел на мудрецов чёрные шапки, спрятал белые и развязал глаза

“Из пяти шапок, что я показал, три надеты на вас”, — сказал король

“Знаете ли вы, какая на вас шапка?” — спросил король

“Нет, не знаю”, хором ответили мудрецы

“Знаете ли вы, какая на вас шапка?” — повторил король

“Нет, не знаю”, хором ответили мудрецы

“Знаете ли вы, какая на вас шапка?” — ещё раз повторил король

“**Да, чёрная**”, хором ответили мудрецы

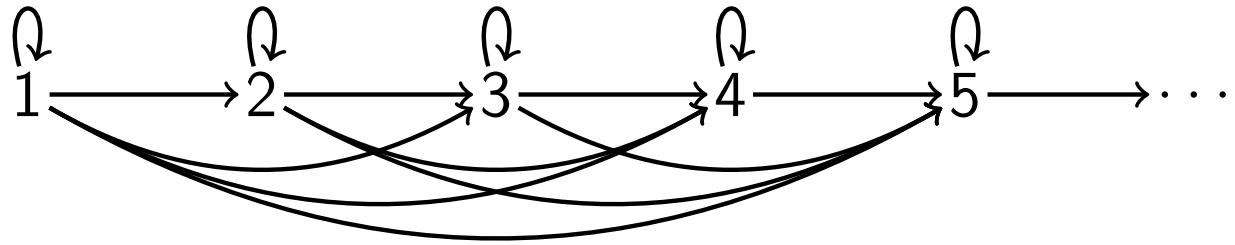
Как законы и ход рассуждений мудрецов можно записать в терминах эпистемической логики?

Темпоральные логики

Темпоральная логика — это модальная логика (*или её расширение*), в которой модальности $\Box$ и $\Diamond$ имеют значения "всегда [в будущем]" и "когда-нибудь [в будущем]"

Шкалой Крипке темпоральной логики описывается течение времени: миры — это моменты времени, а отношение достижимости миров — это порядок моментов времени

Пример: шкала дискретного линейного отсчёта времени



Пропозициональные переменные формул темпоральных логик — это элементарные события, которые могут происходить (или не происходить) в каждый момент времени

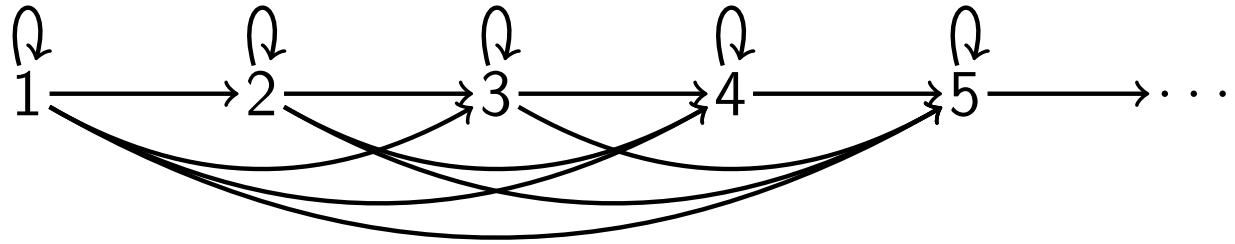
Темпоральные логики

Время может истолковываться по-разному, и в зависимости от истолкования (*то есть точного вида рассматриваемых шкал*) могут получаться разные темпоральные логики, например:

- Логика линейного времени
  (**LTL**, **L**inear **T**emporal **L**ogic)
  - время дискретно линейно течёт вперёд
  - формула — это свойство линейного развития событий
- Логика деревьев вычислений
  (**CTL**, **C**omputation **T**ree **L**ogic)
  - время — это частично упорядоченное множество, которым описываются все альтернативы развития событий
  - формула — это высказывание о возможности и невозможности заданного развития событий с учётом всех альтернатив

Логика линейного времени (LTL)

LTL-шкала — это естественно упорядоченный натуральный ряд (моментов времени):



LTL-интерпретация — это модель Крипке, основанная на LTL-шкале

Логика линейного времени (LTL)

Модальности $\Box$ и $\Diamond$ в LTL обозначаются символами **G** (**G**lobally) и **F** (**F**uture)

К ним добавляются и другие темпоральные операторы:

X (ne**X**ttime) и **U** (**U**ntil)

Итоговый синтаксис LTL-формул:

$$\varphi ::= \mathtt{x} \mid \neg\varphi \mid \varphi \,\&\, \varphi \mid \varphi \vee \varphi \mid \varphi \to \varphi \mid \mathbf{G}\varphi \mid \mathbf{F}\varphi \mid \mathbf{X}\varphi \mid \varphi\mathbf{U}\varphi$$

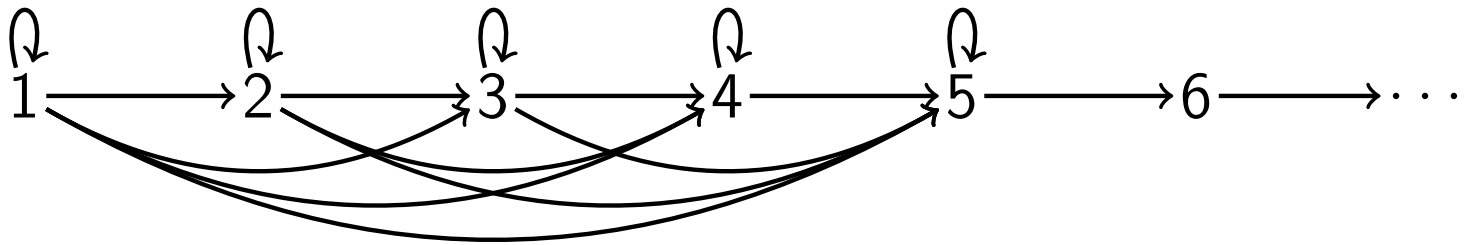
Недостающая часть семантики LTL-формул:

- $\mathcal{I}, n \models \mathbf{X}\varphi \Leftrightarrow \mathcal{I}, n+1 \models \varphi$
  “**в следующий момент** времени будет верно φ”
- $\mathcal{I}, n \models \varphi\mathbf{U}\psi \Leftrightarrow$ существует момент k, $k \geq n$, такой что
  - $\mathcal{I}, k \models \psi$ и
  - для всех моментов k', $n \leq k' < k$, верно $\mathcal{I}, k' \models \varphi$

  “когда-нибудь в будущем станет верным ψ,
  а **до тех пор** будет верно φ”

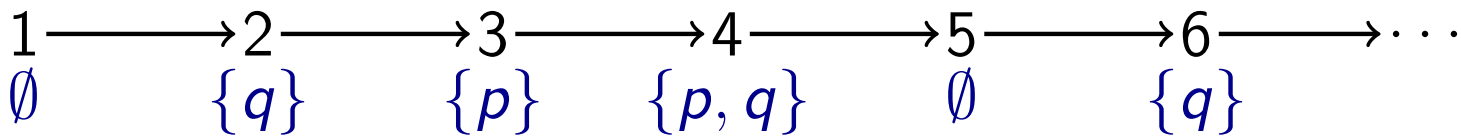
Логика линейного времени (LTL)

Пример:



Логика линейного времени (LTL)

Пример: рассмотрим такую LTL-интерпретацию $\mathcal{I}$ с оценкой элементарных событий, повторяющейся с периодом 4:



Справедливы следующие соотношения:

$\mathcal{I}, 1 \not\models p$ $\qquad$ $\mathcal{I}, 1 \models p \to q$

$\mathcal{I}, 1 \not\models \mathbf{X}p$ $\qquad$ $\mathcal{I}, 1 \models \mathbf{XX}p$

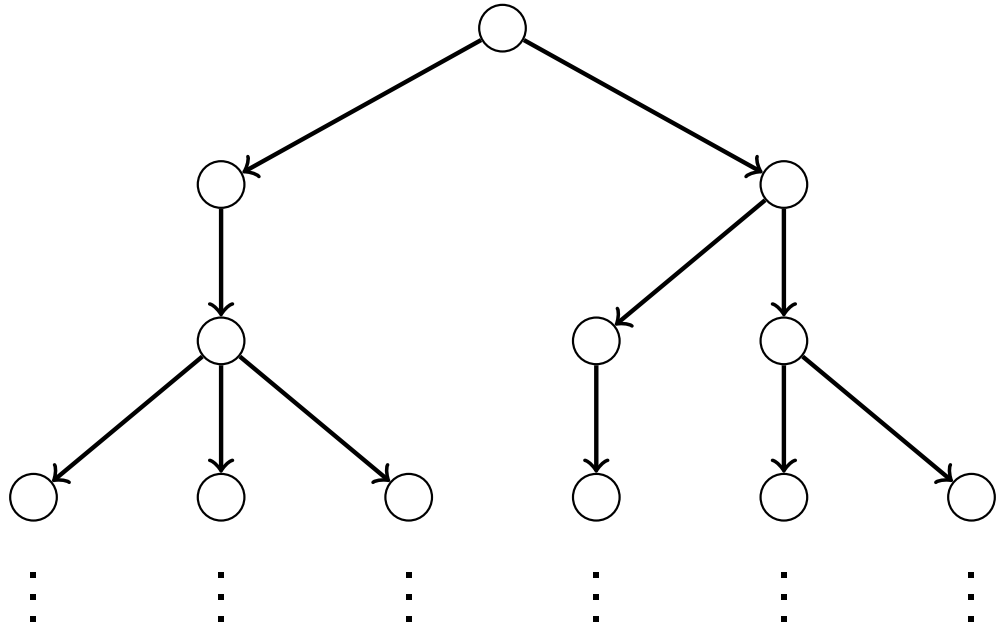
$\mathcal{I}, 1 \models \mathbf{F}p$ $\qquad$ $\mathcal{I}, 1 \not\models \mathbf{G}p$

$\mathcal{I}, 1 \not\models \mathbf{FG}p$ $\qquad$ $\mathcal{I}, 1 \models \mathbf{GF}p$

$\mathcal{I}, 1 \not\models q\mathbf{U}p$ $\qquad$ $\mathcal{I}, 1 \models (q \vee \mathbf{X}q)\mathbf{U}(p \,\&\, q)$

Логика деревьев вычислений (CTL)

Течение времени в CTL описывается ориентированным деревом, содержащим только бесконечные ветви:



CTL-шкала — это шкала Крипке, являющаяся рефлексивно-транзитивным замыканием такого дерева

CTL-интерпретация — это модель Крипке, основанная на CTL-шкале

Логика деревьев вычислений (CTL)

В логике деревьев вычислений модальности $\Box$, $\Diamond$ обозначаются записями **AG** и **EF**

CTL содержит и другие модальности: **EG**, **AF**, **EX**, **AX**, **EU**, **AU**

Значение этих модальностей определяется так:

- $\mathcal{I}, v \models \mathbf{EG}\varphi \Leftrightarrow$ существует ветвь дерева, исходящая из v, такая что для каждой вершины v' этой ветви верно $\mathcal{I}, v' \models \varphi$
- $\mathcal{I}, v \models \mathbf{AF}\varphi \Leftrightarrow$ в каждой ветви дерева, исходящей из v, существует вершина v', такая что $\mathcal{I}, v' \models \varphi$
- $\mathcal{I}, v \models \mathbf{EX}\varphi \Leftrightarrow$ существует вершина v', достижимая из v в дереве по одной дуге, такая что $\mathcal{I}, v' \models \varphi$
- $\mathcal{I}, v \models \mathbf{AX}\varphi \Leftrightarrow$ для каждой вершины v', достижимой из v в дереве по одной дуге, верно $\mathcal{I}, v' \models \varphi$

Логика деревьев вычислений (CTL)

В логике деревьев вычислений модальности $\Box$, $\Diamond$ обозначаются записями **AG** и **EF**

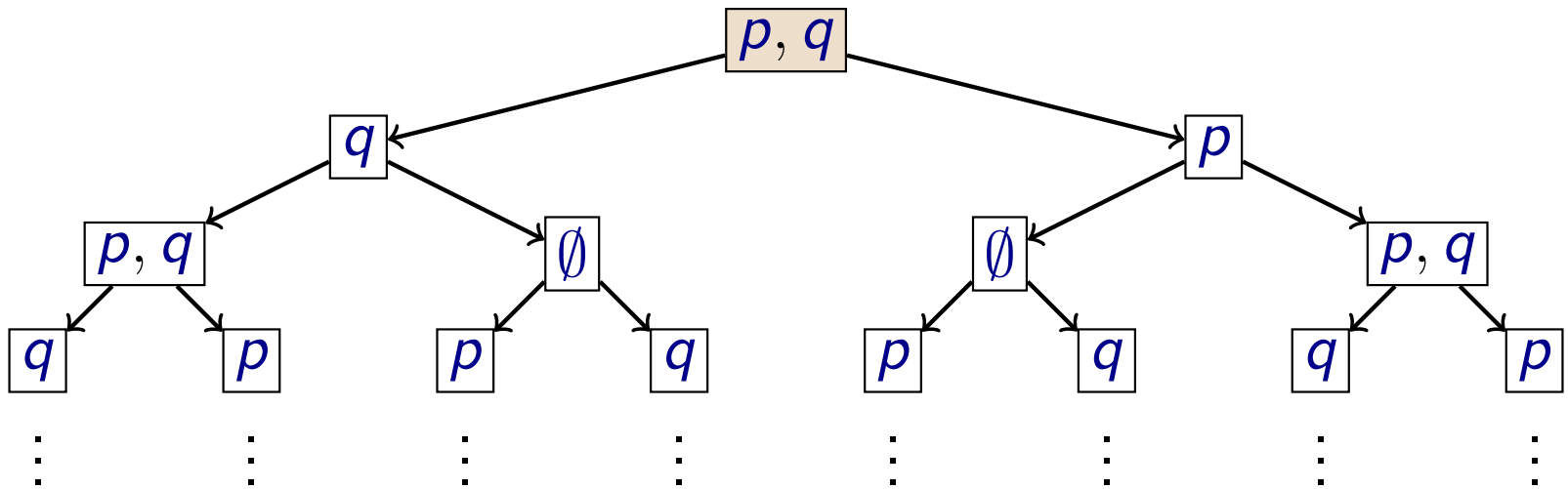
CTL содержит и другие модальности: **EG**, **AF**, **EX**, **AX**, **EU**, **AU**

Значение этих модальностей определяется так:

- $\mathcal{I}, v \models \varphi \mathbf{EU} \psi \Leftrightarrow$ существует ветвь дерева, исходящая из v, содержащая вершину v', такую что
  - $\mathcal{I}, v' \models \psi$ и
  - для каждого предка v'' вершины v', начиная с v, верно $\mathcal{I}, v'' \models \varphi$
- $\mathcal{I}, v \models \varphi \mathbf{AU} \psi \Leftrightarrow$ для каждой ветви дерева, исходящей из v, существует вершина v', такая что
  - $\mathcal{I}, v' \models \psi$ и
  - для каждого предка v'' вершины v', начиная с v, верно $\mathcal{I}, v'' \models \varphi$

Логика деревьев вычислений (CTL)

Пример: рассмотрим такую CTL-интерпретацию $\mathcal{I}$ (при переходе влево изменяется значение p, вправо — значение q):



Справедливы следующие соотношения:

$\mathcal{I}, \square \models p \,\&\, \mathbf{EX}p$

$\mathcal{I}, \square \models \mathbf{AXEX}(p \,\&\, q)$

$\mathcal{I}, \square \models \mathbf{EF}\neg p$

$\mathcal{I}, \square \models \mathbf{EG}p$

$\mathcal{I}, \square \models (q \to p)\mathbf{EU}(\neg p \,\&\, \neg q)$

$\mathcal{I}, \square \models \mathbf{AGEF}(p \,\&\, q)$

$\mathcal{I}, \square \not\models \mathbf{AX}p$

$\mathcal{I}, \square \not\models \mathbf{EXAX}(p \,\&\, q)$

$\mathcal{I}, \square \not\models \mathbf{AF}\neg p$

$\mathcal{I}, \square \not\models \mathbf{AG}p$

$\mathcal{I}, \square \not\models (q \to p)\mathbf{AU}(\neg p \,\&\, \neg q)$

$\mathcal{I}, \square \models \mathbf{AGAF}(p \,\&\, q \vee \neg p \vee \neg q)$

Верификация распределённых систем

Рассмотрим две функции, **параллельно** и **независимо** изменяющие переменную счёт

```
void стипендия() {
  счёт += 1 000;
}
```

```
void надбавка() {
  счёт += 1 000 000;
}
```

Попробуем применить логику Хоара, чтобы проверить, корректно ли выплачиваются одна стипендия и одна надбавка согласно этим функциям

$\{\text{счёт} = x\}$
$\quad \text{счёт} := \text{счёт} + 1\,000$
$\quad \text{счёт} := \text{счёт} + 1\,000\,000;$
$\{\text{счёт} = x + 1\,001\,000\}$

$\{\text{счёт} = x\}$
$\quad \text{счёт} := \text{счёт} + 1\,000\,000$
$\quad \text{счёт} := \text{счёт} + 1\,000;$
$\{\text{счёт} = x + 1\,001\,000\}$

Ответ: **да, корректно**

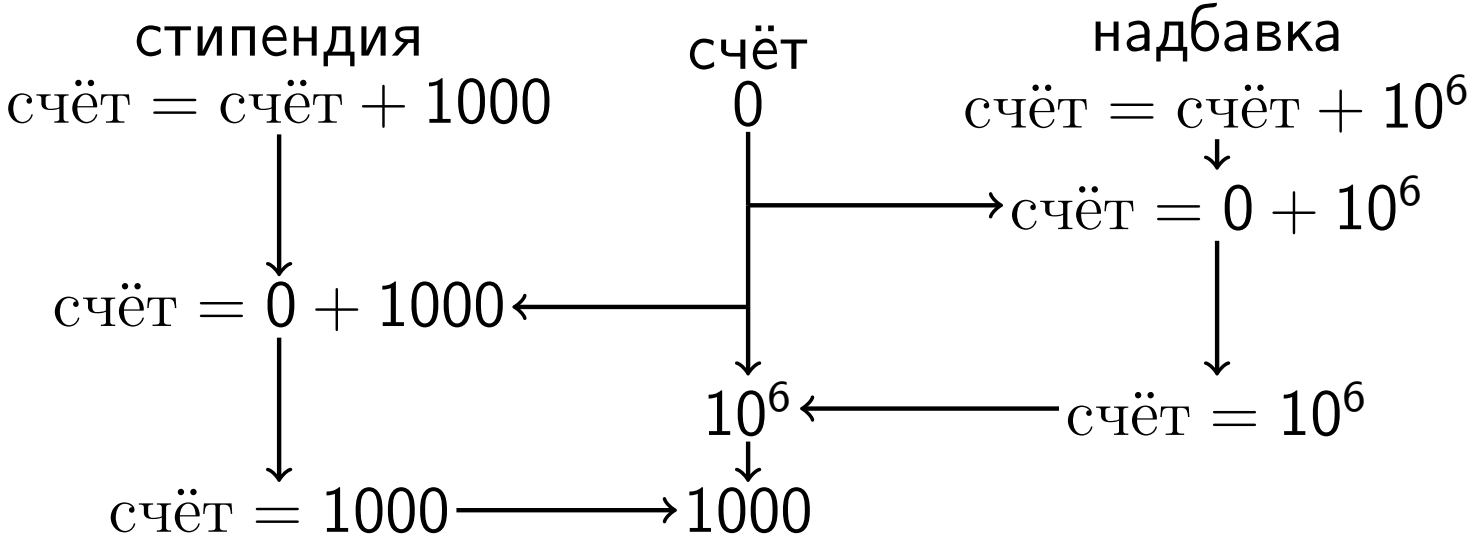
Насколько такой ответ адекватен реальности?

Верификация распределённых систем

Рассмотрим две функции, **параллельно** и **независимо** изменяющие переменную счёт

```
void стипендия() {
  счёт += 1 000;
}
```

```
void надбавка() {
  счёт += 1 000 000;
}
```



Где мой миллион?!

Верификация распределённых систем

Рассмотрим две функции, **параллельно** и **независимо** изменяющие переменную счёт

```
void стипендия() {
  счёт += 1 000;
}
```

```
void надбавка() {
  счёт += 1 000 000;
}
```

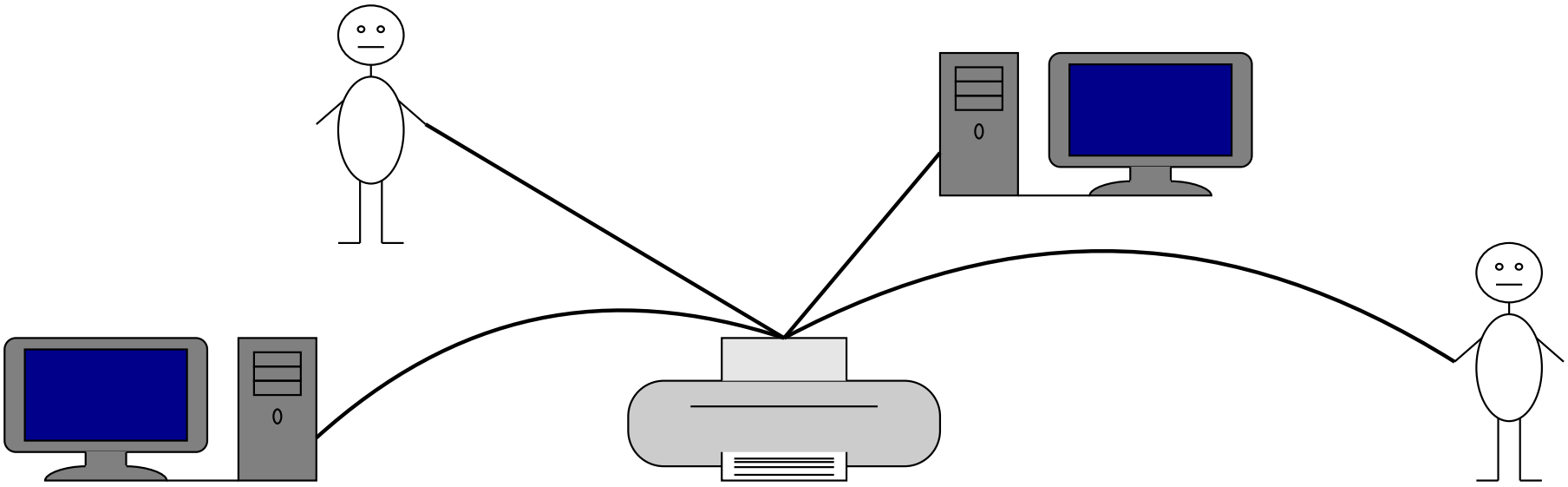
Если вызывается только одна из функций, то она всегда выполняется корректно

При этом выполнение функций, вызванных параллельно, может быть ошибочным

Ошибка проявляется крайне редко и практически не обнаруживается тестированием: чтобы она произошла, параллельно выполняющиеся функции должны произвести заданные действия *почти одновременно* — такая ситуация считается **невоспроизводимой**

Верификация распределённых систем

Ещё один пример: с сетевым принтером пытаются взаимодействовать участники *неизвестной природы*



Принтер работает последовательно: принимает информацию и производит печать согласно содержащейся в нём *программе*

Программы остальных участников, *если они есть*, неизвестны

Как могут выглядеть ошибки выполнения такой системы, и как проверить их отсутствие?

Верификация распределённых систем

К чему приводят такие ошибки:

(*Карпов. Model checking. 2010*)

- *1994* массовая замена дефектных процессоров компанией Intel; причина: некорректная аппаратная реализация инструкции деления чисел с плавающей точкой; ущерб: сотни миллионов $
- *1996* взорвалась ракета “Ариан 5”; причина: зависание при преобразовании чисел с плавающей точкой в целые числа в навигационной программе бортового компьютера; ущерб: 500 млн.$, срыв программы коммерческих запусков спутников
- *1982* при лечении аппаратом облучения раковой опухоли Therac-25 погибло 2 человека, несколько остались инвалидами; причина: редко проявлявшееся состояние гонки (race condition) при параллельной работе многих подпрограмм, обслуживающих аппарат, в результате которого интенсивность облучения возрастала на 2 порядка
- *1995* крушение самолёта “Боинг-757”, 159 погибших; причина: ошибка в одном символе программной системы управления полётом

Верификация распределённых систем

Как обнаруживать такие ошибки

Предположим для простоты, что все компоненты распределённой системы — очень простые программы

Что мешает явно перебрать все сценарии работы и для каждого из них убедиться в отсутствии ошибок?

Пусть в системе параллельно работают 70 программ, каждая из которых совершает одно действие и завершается

Тогда всевозможных сценариев работы будет **70!**

это больше, чем **гугол**

При этом разные последовательности действий могут приводить к совершенно разным и неожиданным результатам

(*куда исчез мой миллион?*)

Верификация распределённых систем

Как обнаруживать такие ошибки:

- **тестирование**: система многократно выполняется на особенных входных данных, и анализируются отклонения от ожидаемых результатов — **не подходит** (почему?)
- **формальная верификация**

Прежде всего остановимся на том, какие **требования** обычно предъявляются к распределённым системам

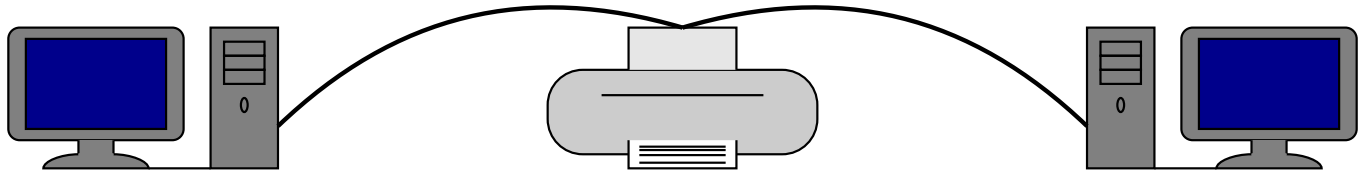
В таких требованиях часто присутствует время:

- *в тот момент, когда* функции завершат работу, на счёт поступит 1 001 000
- *в какой бы момент времени* ни пришёл запрос на печать, *когда-нибудь в будущем* документ обязательно напечатается

Для описания развития событий во времени предназначены темпоральные логики (*подробно рассмотрим только LTL*)

Выразительные возможности LTL

Вернёмся к примеру с сетевым принтером, для простоты полагая, что в сети ровно два компьютера:



Правильная работа такой системы предполагает, в числе прочего, что:

1. данные на принтер всегда передаются не более чем одним компьютером
2. компьютер не будет передавать данные на принтер вечно
3. если компьютер посылает достаточно много (*потенциально бесконечно много*) запросов на печать, то рано или поздно этот компьютер начнёт передавать данные
4. если компьютер начинает (*конечный*) сеанс передачи данных на принтер, то в течение всего сеанса принтер будет занят этим компьютером

Выразительные возможности LTL

Введём элементарные события, используемые в требованиях к системе:

1. rq_i: i-й компьютер посылает запрос на печать
2. pr_i: i-й компьютер передаёт данные на печать
3. $busy_i$: принтер занят i-м компьютером

Требования к системе с сетевым принтером можно записать в терминах LTL так:

1. данные на принтер всегда передаются не более чем одним компьютером

$$\mathbf{G}\neg(pr_1 \,\&\, pr_2)$$

Выразительные возможности LTL

2. компьютер не будет передавать данные на принтер вечно

$$\neg \mathbf{FG} pr_i$$

3. если компьютер посылает достаточно много запросов на печать, то рано или поздно этот компьютер начнёт передавать данные

$$\mathbf{GF} rq_i \to \mathbf{F} pr_i$$

4. если компьютер начинает сеанс передачи данных на принтер, то в течение всего сеанса принтер будет занят этим компьютером

$$\mathbf{G}(pr_i \to (busy_i \mathbf{U} \neg pr_i)) \quad ?$$
$$\mathbf{G}(\neg pr_i \,\&\, \mathbf{X} pr_i \to \mathbf{X}(busy_i \mathbf{U} \neg pr_i)) \quad ?$$

Ещё немного об LTL

Пусть $\mathcal{I}$ — LTL-интерпретация, и φ, ψ — LTL-формулы

Тогда

- формула φ выполняется в интерпретации $\mathcal{I}$ ($\mathcal{I} \models \varphi$), если верно $\mathcal{I}, 1 \models \varphi$
- формула φ общезначима ($\models \varphi$), если для любой LTL-интерпретации $\mathcal{J}$ верно $\mathcal{J} \models \varphi$
- формулы φ, ψ равносильны ($\varphi \approx \psi$), если верно $\models \varphi \leftrightarrow \psi$

В логике линейного времени справедливы все равносильности, являющиеся законами булевой алгебры

К этим законам добавляются и другие, позволяющие преобразовывать подформулы, содержащие темпоральные операторы

Ещё немного об LTL

Законы продвижения отрицания.

- $\neg \mathbf{X}\varphi \approx \mathbf{X}\neg\varphi$
- $\neg \mathbf{F}\varphi \approx \mathbf{G}\neg\varphi$
- $\neg \mathbf{G}\varphi \approx \mathbf{F}\neg\varphi$
- $\neg(\varphi \mathbf{U} \psi) \approx \mathbf{G}\neg\psi \vee (\neg\psi \mathbf{U}(\neg\varphi \,\&\, \neg\psi))$

Законы исключения.

- $\mathbf{F}\varphi \approx \neg \mathbf{G}\neg\varphi$
- $\mathbf{G}\varphi \approx \neg \mathbf{F}\neg\varphi$
- $\mathbf{F}\varphi \approx \mathbf{true}\, \mathbf{U}\varphi$

Законы неподвижной точки.

- $\mathbf{F}\varphi \approx \varphi \vee \mathbf{X}\mathbf{F}\varphi$
- $\mathbf{G}\varphi \approx \varphi \,\&\, \mathbf{X}\mathbf{G}\varphi$
- $\varphi \mathbf{U} \psi \approx \psi \vee (\varphi \,\&\, \mathbf{X}(\varphi \mathbf{U} \psi))$

Доказательство. Достаточно использовать определение

Покажем справедливость закона

$$\neg(\varphi \mathbf{U} \psi) \approx \mathbf{G}\neg\psi \vee (\neg\psi \mathbf{U}(\neg\varphi \,\&\, \neg\psi))$$

Ещё немного об LTL

Доказательство. $\neg(\varphi \mathbf{U} \psi) \approx \mathbf{G}\neg\psi \vee (\neg\psi \mathbf{U}(\neg\varphi \,\&\, \neg\psi))$

Рассмотрим произвольную LTL-интерпретацию $\mathcal{I}$, такую что $\mathcal{I} \models \neg(\varphi \mathbf{U} \psi)$

Тогда $\mathcal{I} \not\models \varphi \mathbf{U} \psi$, и это возможно в двух (*пересекающихся*) случаях:

- ψ никогда не становится верным: $\mathcal{I}, i \models \neg\psi$ для любого i
  - тогда справедливо $\mathcal{I} \models \mathbf{G}\neg\psi$ и $\mathcal{I} \models \mathbf{G}\neg\psi \vee \ldots$
- существует момент времени i, такой что ψ *пока ещё* не стало верным, но при этом *уже* стало неверным φ: существует момент i, такой что $\mathcal{I}, i \models \neg\psi$, $\mathcal{I}, i \models \neg\varphi$ и для любого момента j, $j < i$, справедливо $\mathcal{I}, k \models \neg\psi$
  - тогда справедливо $\mathcal{I} \models \neg\psi \mathbf{U}(\neg\varphi \,\&\, \neg\psi)$ и $\mathcal{I} \models \cdots \vee (\neg\psi \mathbf{U}(\neg\varphi \,\&\, \neg\psi))$

В обратную сторону рассуждения аналогичны ▼

Ещё немного об LTL

Задача для самостоятельного размышления

(законы дистрибутивности)

- $\mathbf{F}(\varphi \vee \psi) \overset{?}{\approx} \mathbf{F}\varphi \vee \mathbf{F}\psi$
- $\mathbf{F}(\varphi \,\&\, \psi) \overset{?}{\approx} \mathbf{F}\varphi \,\&\, \mathbf{F}\psi$
- $\mathbf{G}(\varphi \vee \psi) \overset{?}{\approx} \mathbf{G}\varphi \vee \mathbf{G}\psi$
- $\mathbf{G}(\varphi \,\&\, \psi) \overset{?}{\approx} \mathbf{G}\varphi \,\&\, \mathbf{G}\psi$
- $\varphi \mathbf{U}(\psi \vee \chi) \overset{?}{\approx} (\varphi \mathbf{U} \psi) \vee (\varphi \mathbf{U} \chi)$
- $\varphi \mathbf{U}(\psi \,\&\, \chi) \overset{?}{\approx} (\varphi \mathbf{U} \psi) \,\&\, (\varphi \mathbf{U} \chi)$
- $(\varphi \vee \psi) \mathbf{U} \chi \overset{?}{\approx} (\varphi \mathbf{U} \chi) \vee (\psi \mathbf{U} \chi)$
- $(\varphi \,\&\, \psi) \mathbf{U} \chi \overset{?}{\approx} (\varphi \mathbf{U} \chi) \vee (\psi \mathbf{U} \chi)$

Верификация распределённых систем

Чтобы иметь средства автоматической формальной проверки правильности работы системы, необходимо научиться

- формально записывать требования правильности
  - например, некоторые требования можно записывать как LTL-формулы
- математически строго описывать то, как выполняется система
- предложить алгоритм, проверяющий, удовлетворяют ли выполнения строго описанной системы формально записанным требованиям

Верификация распределённых систем

Если для формализации требований выбирается темпоральная логика, то чаще всего в качестве строго описания системы выбирается особенный размеченный граф, называемый **размеченной системой переходов**:

LTS (**L**abelled **T**ransition **S**ystem)

Более точно, чаще всего выбираются особенные размеченные системы переходов: модели Крипке — но чтобы не путать их с "моделями Крипке" модальных логик, будем использовать название **LTS**

Верификация распределённых систем

LTS над множеством элементарных событий $\mathcal{P}$ — это система $(S, S_0, \to, \rho)$, где:

- S — произвольное непустое конечное множество состояний системы
- S_0 — множество начальных состояний, $S_0 \subseteq S$
- $\to$: $S \times S$ — **тотальное** отношение переходов
- $\rho : S \to 2^{\mathcal{P}}$ — функция разметки состояний происходящими в них элементарными событиями

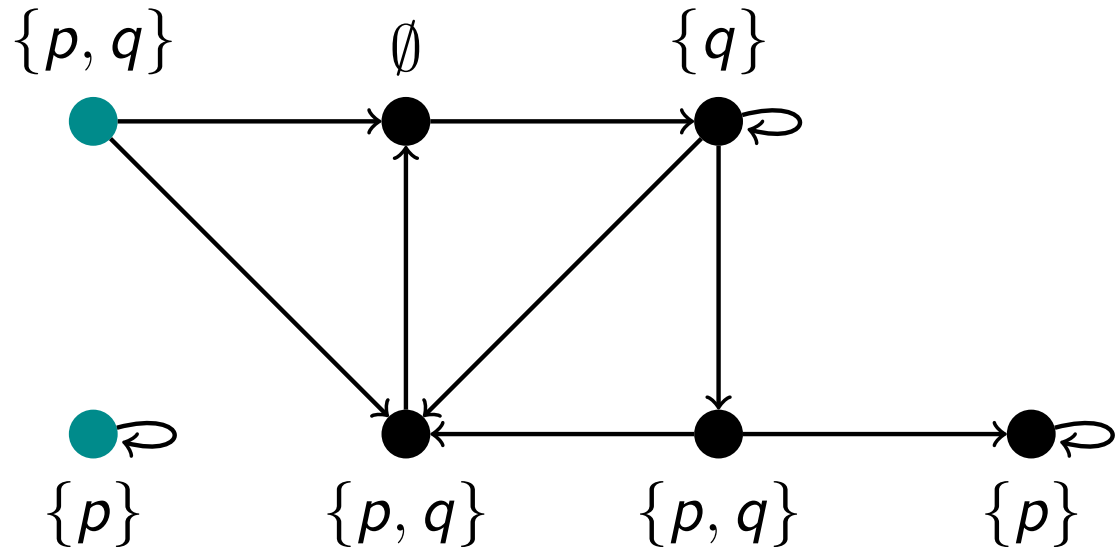
Тотальность отношения переходов означает, что из любого состояния вычисления s можно совершить переход: $\exists s'(s \to s')$

Верификация распределённых систем

Пример

$\mathcal{P} = \{p, q\}$

LTS $(\{\ldots, \bullet, \ldots\}, \{\ldots, \bullet, \ldots\}, \to, \rho)$:



Верификация распределённых систем

Трасса LTS $M = (S, S_0, \to, \rho)$ — это любая бесконечная последовательность tr состояний системы вида

$$s_1 \to s_2 \to s_3 \to \dots$$

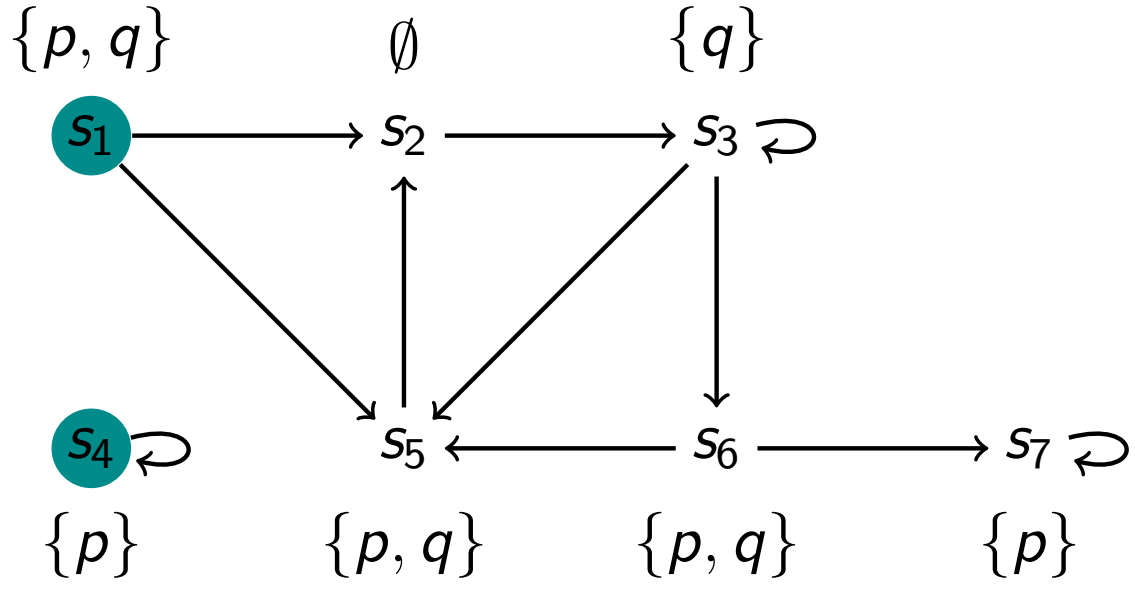
Трасса tr называется начальной, если $s_1 \in S_0$

Начальная трасса также называется вычислением LTS

Если tr — трасса LTS M, то $I(tr)$ — LTL-интерпретация $(\mathbb{N}, <, \xi)$, такая что для любого момента времени n верно $\xi(n) = \rho(s_n)$

Верификация распределённых систем

Пример

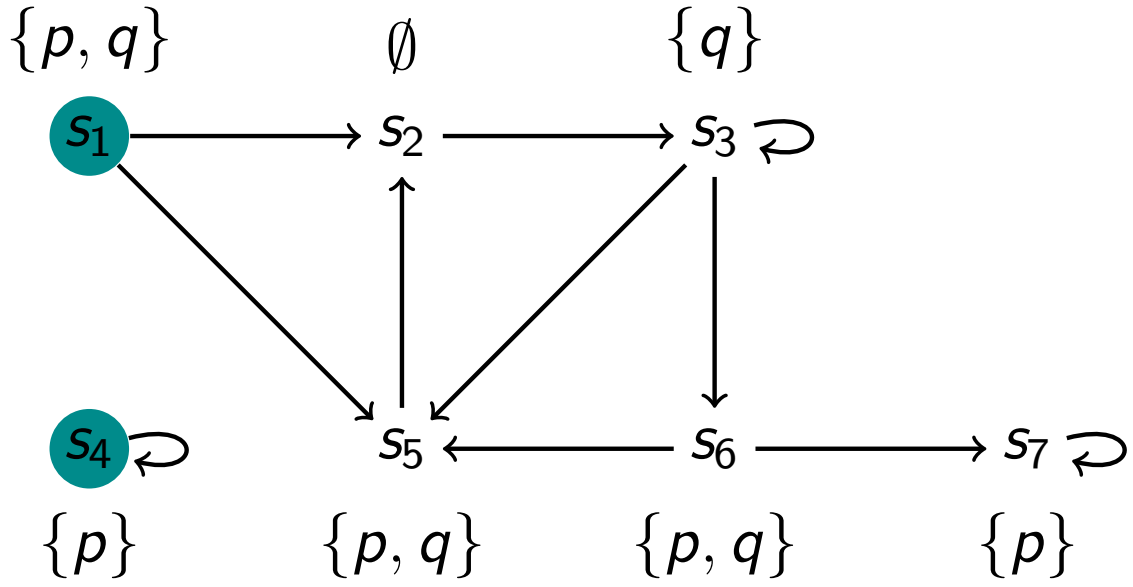


Трасса tr:

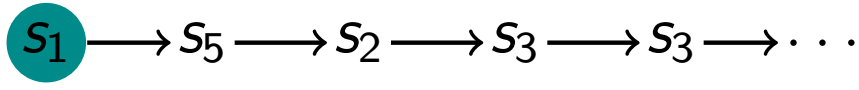
$$s_1 \longrightarrow s_5 \longrightarrow s_2 \longrightarrow s_3 \longrightarrow s_3 \longrightarrow \cdots$$

Верификация распределённых систем

Пример

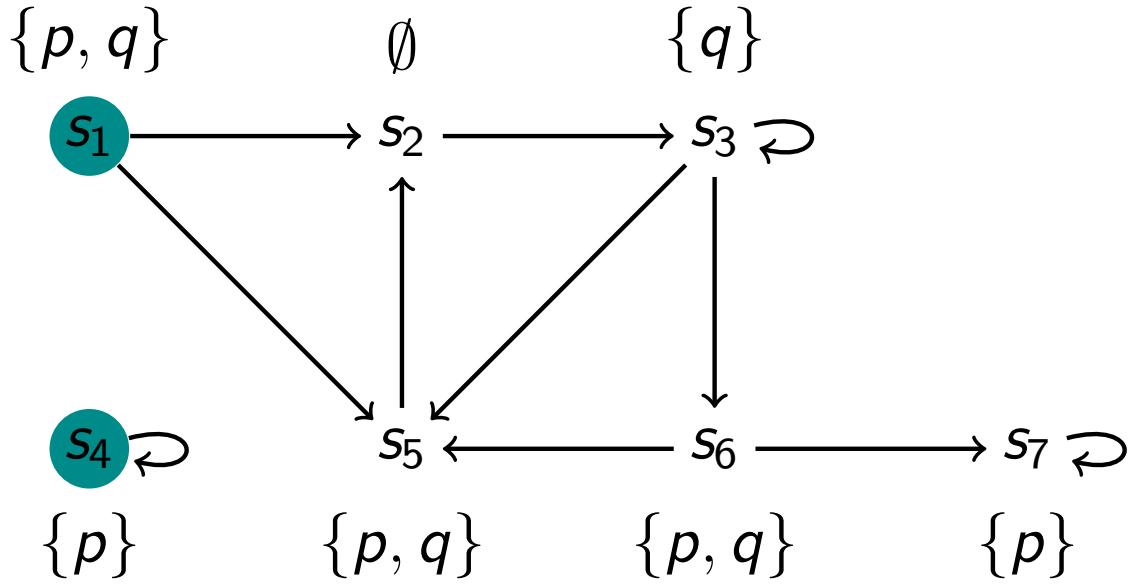


Вычисление tr:



Верификация распределённых систем

Пример



Интерпретация $I(tr)$:

$$\begin{array}{ccccccccccccc} & \{p,q\} & & \{p,q\} & & \emptyset & & \{q\} & & \{q\} & & & \\ 1 & \longrightarrow & 2 & \longrightarrow & 3 & \longrightarrow & 4 & \longrightarrow & 5 & \longrightarrow & \cdots \end{array}$$

Верификация распределённых систем

Как построить LTS, описывающую поведение распределённой системы

Начнём с простого: покажем, как можно построить LTS для последовательной программы π

- состояния LTS — это состояния вычисления программы π
  - (состояние вычисления включает в себя состояние управления и состояние данных)
- начальные состояния LTS — это всевозможные состояния, с которых может начинаться вычисление π
- переход в LTS соответствует шагу вычисления программы π
  - (например, $\langle \pi,\ \theta \rangle \to_I \langle \pi',\ \theta' \rangle$)
- элементарное событие соответствует отношению над значениями переменных и состояния управления программы
- событие включается в метку состояния LTS $\Leftrightarrow$ состояние вычисления удовлетворяет соответствующему отношению

Верификация распределённых систем

Пример

Вернёмся ещё раз к примеру с сетевым принтером

Предположим, что в контроллере принтера есть однобитовый регистр **R**, доступный на чтение и запись всем компьютерам в сети:

$$\mathbf{R} = \mathbf{true} \quad \Leftrightarrow \quad \text{принтер свободен}$$

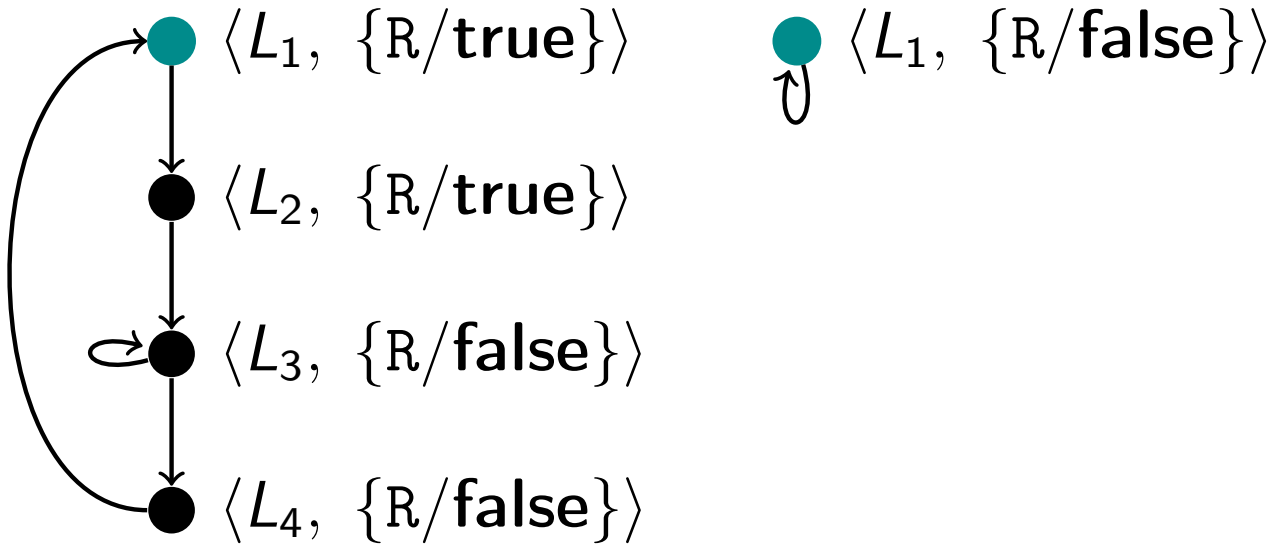
Тогда программа π взаимодействия компьютера с принтером может выглядеть так:

$$\begin{array}{l}
\mathbf{while}\ (\mathbf{true})\ \{ \\
\quad L_1 : \mathbf{while}\ (!R); \\
\quad L_2 : R = \mathbf{false}; \\
\quad L_3 : SEND_DATA \\
\quad L_4 : R = \mathbf{true}; \\
\}
\end{array}$$

Верификация распределённых систем

Пример: LTS для программы π может выглядеть так:

(*функция разметки опущена*)



Верификация распределённых систем

LTS и окружение программы

Программа в распределённой системе может взаимодействовать с другими программами: общие переменные, обмен сообщениями, сигналы, ...

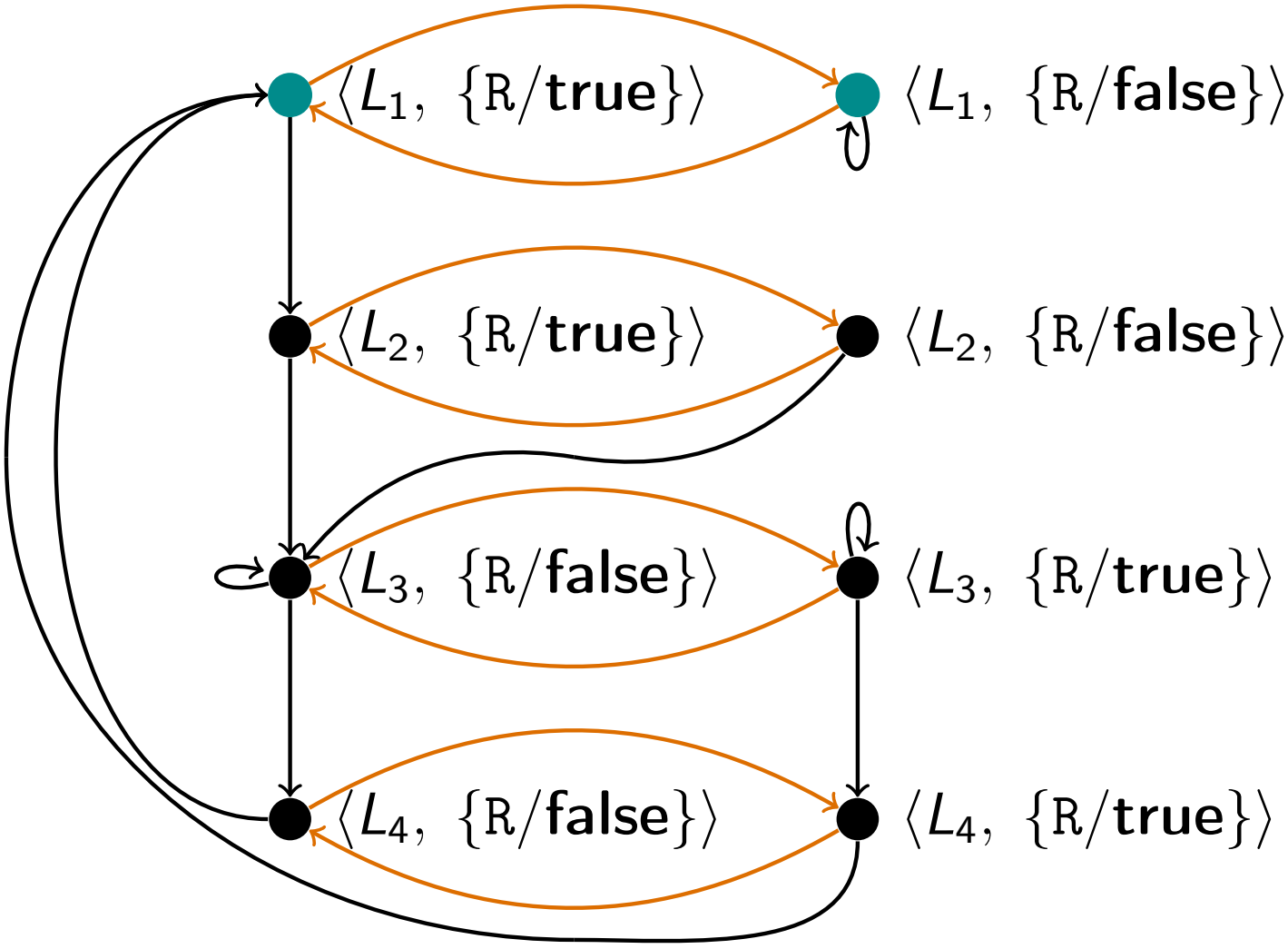
Такое взаимодействие выражается в том, что состояние вычисления программы может измениться (под воздействием её **окружения**)

Например, регистр **R** в последнем примере может быть изменён любым компьютером сети

Чтобы учесть такое изменение, следует добавить в LTS переходы, отвечающие всем возможностям окружения повлиять на состояние вычисления программы

Верификация распределённых систем

Пример: LTS для программы π с окружением, способным произвольно переключать значение регистра **R**



Верификация распределённых систем

LTS и взаимодействие программ

Предположим, что для программ π_1, π_2 построены LTSs и что эти программы взаимодействуют между собой

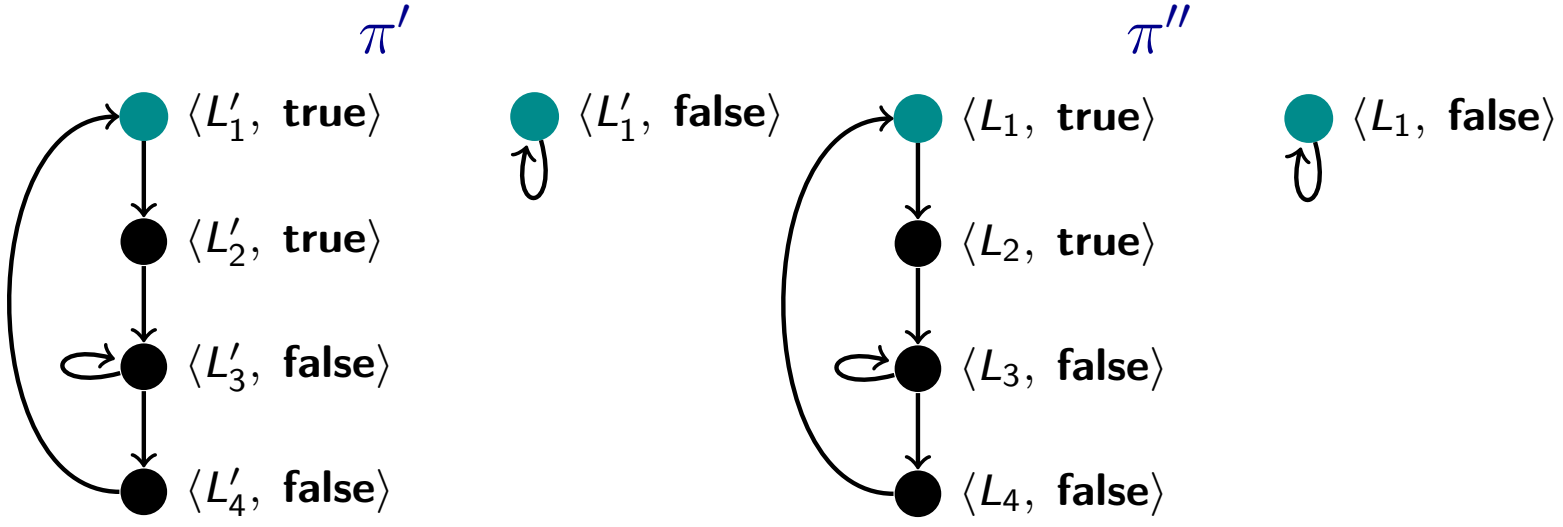
Один из наиболее популярных способов формализации взаимодействия параллельно выполняющихся программ — это семантика чередующихся вычислений: (*interleaving semantics*)

- совокупное состояние вычисления — это состояния π_1, π_2, в которых общие переменные изменяются синхронно (и учтены другие взаимосвязи состояний, если они есть)
- переход в LTS, описывающей взаимодействие программ, выглядит так:
  - произвольно (недетерминированно) выбирается одна из программ
  - совершение перехода в LTS = выполнение одного шага вычисления выбранной программы

Верификация распределённых систем

Пример

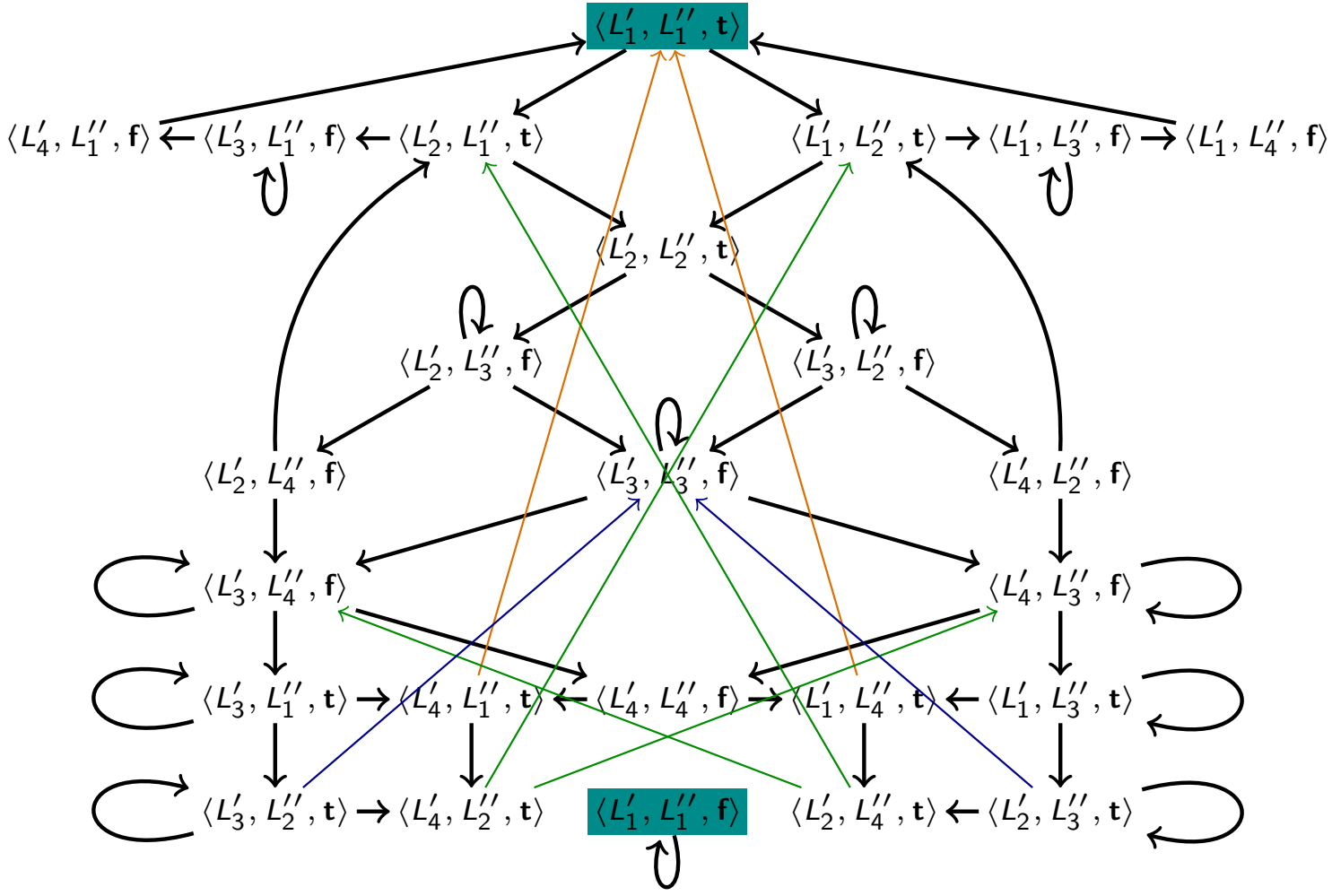
Рассмотрим LTSs, описывающие две (одинаковые) программы взаимодействия с сетевым принтером:



Считаем регистр **R общей переменной** программ π', π''

Верификация распределённых систем

Пример: LTS, описывающая взаимодействие π', π'' согласно **семантике чередующихся вычислений**, выглядит так:



Задача model checking для LTL

$\mathfrak{I}(M)$ — множество всевозможных LTL-интерпретаций $I(tr)$, где tr — вычисление M

LTL-формула φ над множеством элементарных событий $\mathcal{P}$ выполняется на LTS M над тем же множеством $\mathcal{P}$ ($M \models \varphi$), если для любой интерпретации $\mathcal{I}$ множества $\mathfrak{I}(M)$ справедливо соотношение $\mathcal{I} \models \varphi$

Задача проверки моделей (model checking) для LTL формулируется так:

для заданных LTL-формулы φ и LTS M
проверить справедливость соотношения $M \models \varphi$

Задача model checking для LTL

Эта задача не так проста, как кажется:

- LTS M конечна, но содержит **бесконечно много** вычислений
- каждым вычислением tr определяется **бесконечная** LTL-интерпретация $I(tr)$

Тем не менее, эта задача имеет решение
(эффективные решающие алгоритмы)

Подробно рассмотрим табличный алгоритм решения задачи

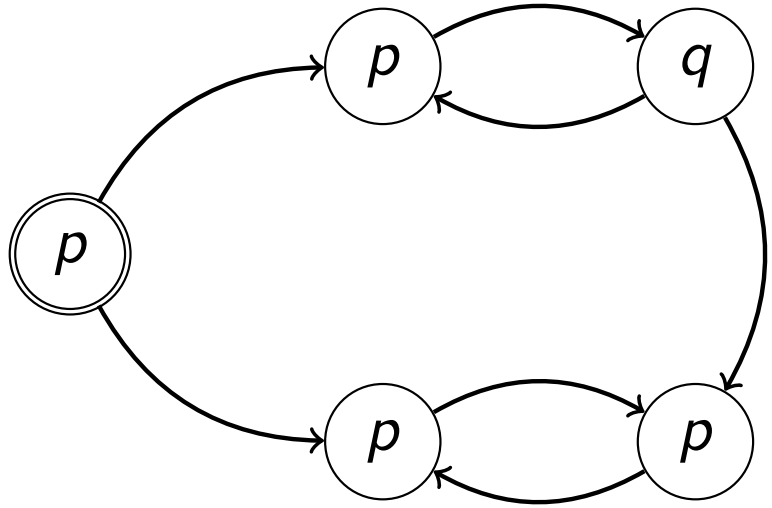
Запись $tr \models \varphi$, где tr — трасса LTS, и φ — LTL-формула, будем использовать как синоним записи $\mathcal{I}(tr) \models \varphi$

Табличный алгоритм model checking для LTL

Сквозной пример:

φ: $p\mathbf{U}q$

M:



Верно ли соотношение $M \models \varphi$?

Табличный алгоритм model checking для LTL

Попытаемся доказать, что $M \not\models \varphi$

Для этого попробуем найти вычисление tr, такое что $tr \not\models \varphi$:

- упростим формулу φ
- разметим каждое состояние s модели всевозможными "хорошими" **предположениями** о том, какие подформулы формулы φ истинны и какие ложны хотя бы для одной трассы, исходящей из s
- проведём "хорошим" образом дуги от предположений, сформулированных для текущего момента времени и текущего состояния, к предположениям, сформулированным для следующего момента времени и следующего состояния
- проверим, существует ли вычисление LTS M, "корректно" размеченное предположениями и начинающееся с состояния, в котором формула φ полагается ложной

Упрощение формулы

LTL-формулу будем называть упрощённой, если она

- содержит только связки $\&$, $\neg$ и операторы **X**, **U**
- не содержит двойных отрицаний (двух подряд идущих связок $\neg$)

Утверждение

Для любой LTL-формулы существует равносильная упрощённая LTL-формула

Доказательство. Достаточно поочерёдно исключить из формулы

- **G**: $\mathbf{G}\varphi \approx \neg\mathbf{F}\neg\varphi$
- **F**: $\mathbf{F}\varphi \approx \mathbf{true}\ \mathbf{U}\ \varphi$
- $\rightarrow$: $\varphi \rightarrow \psi \approx \neg\varphi \vee \psi$
- $\vee$: $\varphi \vee \psi \approx \neg(\neg\varphi \,\&\, \neg\psi)$
- $\neg\neg$: $\neg\neg\varphi \approx \varphi$

▼

Далее проверяемая формула полагается упрощённой

Предположения

При разметке состояний модели M предположениями будут использоваться формулы из замыкания Фишера-Ладнера $[\varphi]_{fl}$, содержащего

- все подформулы формулы φ, кроме подформул вида $\neg\psi$
- $\mathbf{X}(\psi \mathbf{U} \chi)$ для каждой подформулы $\psi \mathbf{U} \chi$ формулы φ
  - *пояснение того, зачем включать в $[\varphi]_{fl}$ такие $\mathbf{X}$-формулы, будет дальше*

Предположение (для формулы φ) — это множество $[\varphi]_{fl}$, в котором каждая формула φ помечена как положительная (φ^+) или отрицательная (φ^-)

Для краткости будем использовать запись $\neg\psi^+$ как синоним ψ^-, и запись $\neg\psi^-$ как синоним ψ^+

Хорошие и плохие предположения

Предположение H согласованно с множеством элементарных событий P, если для любого p, $p \in \mathcal{P}$, верно

$$p^+ \in H \quad \Leftrightarrow p \in P$$

Хорошее предположение H, помечающее состояние s LTS $M = (S, S_0, \to, \rho)$, удовлетворяет двум условиям:

1. согласованность с состоянием s, то есть согласованность с атомарными высказываниями $\rho(s)$
2. внутренняя согласованность: для любых формул $\psi, \psi_1 \,\&\, \psi_2, \chi_1 \mathbf{U} \chi_2 \in [\varphi]_{fl}$ верно:
   - $\psi^+ \in H \Leftrightarrow \psi^- \notin H$
   - $\psi_1 \,\&\, \psi_2^+ \in H \Leftrightarrow \psi_1^+, \psi_2^+ \in H$
   - $\chi_1 \mathbf{U} \chi_2^+ \in H \Leftrightarrow \chi_2^+ \in H$ или $\{\chi_1^+, \mathbf{X}(\chi_1 \mathbf{U} \chi_2)^+\} \subseteq H$

Табличный алгоритм размечает **каждое** состояние LTS M **всевозможными** предположениями, согласованными внутренне и с размечаемым состоянием

Хорошие и плохие предположения

Предположение H выполнено для трассы tr ($tr \models H$), если

- для любой положительной формулы ψ из H верно $tr \models \psi$ и
- для любой отрицательной формулы χ из H верно $tr \not\models \chi$

Если tr — трасса вида $s_1 \to s_2 \to \ldots$, то $tr[i]$ — состояние s_i

Утверждение о локальной согласованности

Если для трассы tr и предположения H справедливо $tr \models H$, то H согласованно с состоянием $tr[1]$ и внутренне

Доказательство. Остановимся подробно на самой "трудной" части утверждения: "**если $tr \models H$, то**

$$\chi_1 \mathbf{U} \chi_2^+ \in H \Leftrightarrow \chi_2^+ \in H \text{ или } \left\{\chi_1^+, \mathbf{X}(\chi_1 \mathbf{U} \chi_2)^+\right\} \subseteq H\text{"}$$

Согласно закону неподвижной точки $\psi \mathbf{U} \chi \equiv \chi \vee \psi \,\&\, \mathbf{X}(\psi \mathbf{U} \chi)$, соотношение $\mathcal{I}(tr) \models \psi \mathbf{U} \chi$ справедливо тогда и только тогда, когда

- $\mathcal{I}(tr) \models \chi$ или
- $\mathcal{I}(tr) \models \psi$ и $\mathcal{I}(tr) \models \mathbf{X}(\psi \mathbf{U} \chi)$

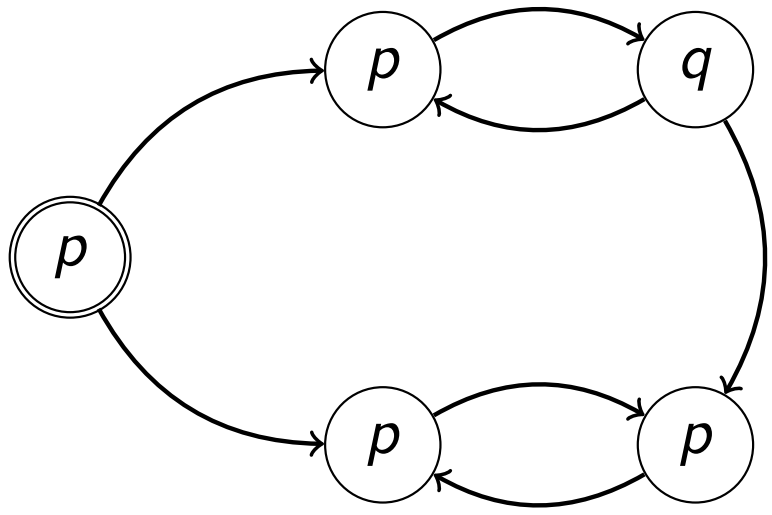
▼

Хорошие и плохие предположения

Вернёмся к примеру:

φ: $p\mathbf{U}q$

M:

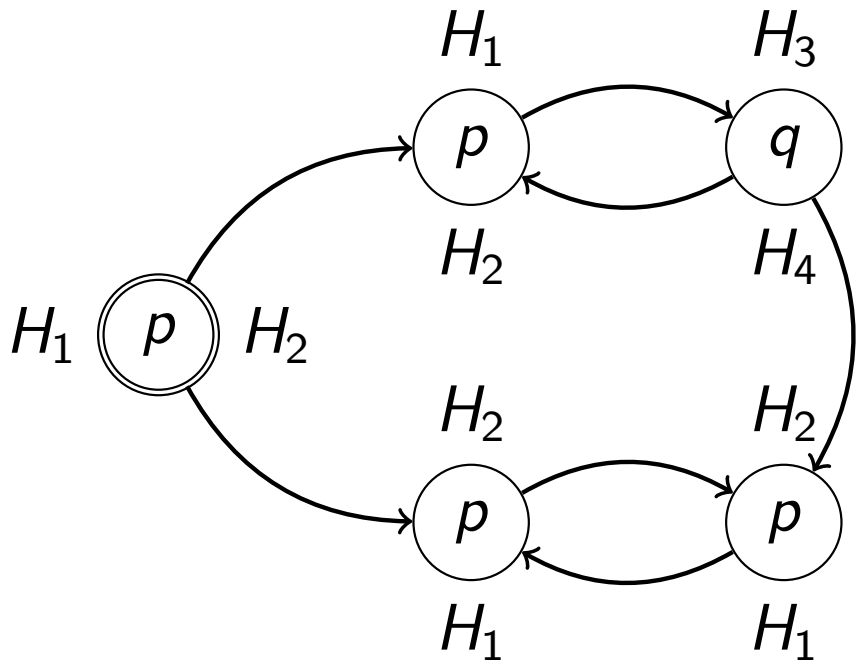


Разметим состояния модели M хорошими предположениями

Хорошие и плохие предположения

φ: $p\mathbf{U}q$

M:



$$H_1 = \{p^+, q^-, \mathbf{X}(p\mathbf{U}q)^+, p\mathbf{U}q^+\}$$
$$H_2 = \{p^+, q^-, \mathbf{X}(p\mathbf{U}q)^-, p\mathbf{U}q^-\}$$
$$H_3 = \{p^-, q^+, \mathbf{X}(p\mathbf{U}q)^+, p\mathbf{U}q^+\}$$
$$H_4 = \{p^-, q^+, \mathbf{X}(p\mathbf{U}q)^-, p\mathbf{U}q^+\}$$

Отложенные предположения

Упорядоченная пара предположений (H_1, H_2) внешне согласованна, если для любой формулы вида $\mathbf{X}\psi$ из $[\varphi]_{fl}$ верно:

$$\mathbf{X}\psi^+ \in H_1 \quad \Leftrightarrow \quad \psi^+ \in H_2$$

Табличный алгоритм соединяет дугами все пары внешне согласованных предположений H_1, H_2, помечающих состояния s_1, s_2 соответственно, такие что $s_1 \to s_2$

Если tr — трасса вида $s_1 \to s_2 \to \ldots$, то $tr|^i$ — трасса $s_i \to s_{i+1} \to \ldots$

Утверждение о внешней согласованности. **Если для трассы tr и предположений H_1, H_2 справедливы соотношения $tr \models H_1$ и $tr|^2 \models H_2$, то пара (H_1, H_2) внешне согласованна**

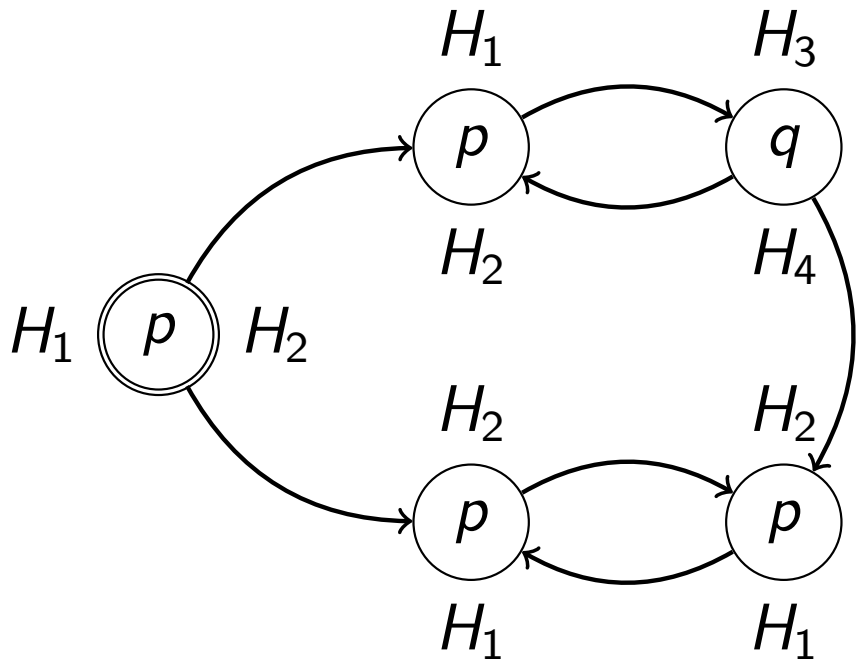
Доказательство. Следует из определения значения $\mathbf{X}$ ▼

Отложенные предположения

Вернёмся к примеру:

φ: $p\mathbf{U}q$

M:



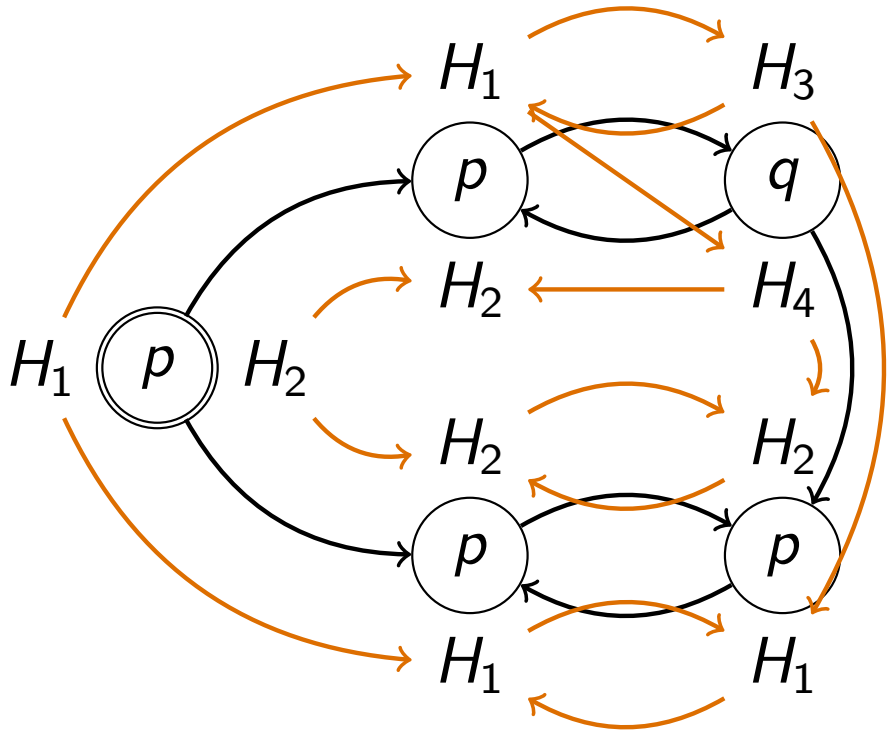
$$H_1 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_2 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^-\}$$
$$H_3 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_4 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^+\}$$

Соединим соседние внешне согласованные предположения

Отложенные предположения

φ: $p\mathbf{U}q$

M:



$$H_1 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_2 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^-\}$$
$$H_3 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_4 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^+\}$$

Система Хинтикки

$$H_1 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_2 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^-\}$$
$$H_3 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_4 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^+\}$$

В некоторых случаях бесконечной последовательностью предположений, согласованных внутренне, внешне и с соответствующими состояниями трассы tr, корректно описывается то, какие формулы выполнены на трассе:

- $H_1 \to H_1 \to H_3 \to \ldots$: для трассы tr, размеченной так, верно $tr \models p\mathbf{U}q$, и при этом $p\mathbf{U}q^+ \in H_1$
- $H_2 \to \ldots$: для трассы tr, размеченной так, верно $tr \not\models p\mathbf{U}q$, и при этом $p\mathbf{U}q^- \notin H_2$

Система Хинтикки

$$H_1 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_2 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^-\}$$
$$H_3 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_4 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^+\}$$

К сожалению, это работает не всегда:

- $H_1 \to H_1 \to \cdots \to H_1 \to \ldots$: для трассы tr, размеченной так, верно $tr \not\models p\mathbf{U}q$, но при этом $p\mathbf{U}q^+ \in H_1$

"Плохая" особенность бесконечно повторяющегося предположения H_1:

почти всегда предполагаются q^- и $\mathbf{X}(p\mathbf{U}q)^+$

"сейчас ещё не q, но я полагаю, что когда-нибудь будет q, и подтверждение этого предположения отложу на потом ($\mathbf{X}$) — и ещё раз отложу — и ещё — ... — и до бесконечности, а q так и не наступило"

Система Хинтикки

Для формулы $\psi \mathbf{U} \chi$ в предположении H звенит звонок, если верно хотя бы одно из двух включений:

$$\chi^+ \in H, \qquad \mathbf{X}(\varphi \mathbf{U} \psi)^- \in H$$

Система Хинтикки для LTS M и LTL-формулы φ — это граф следующего вида:

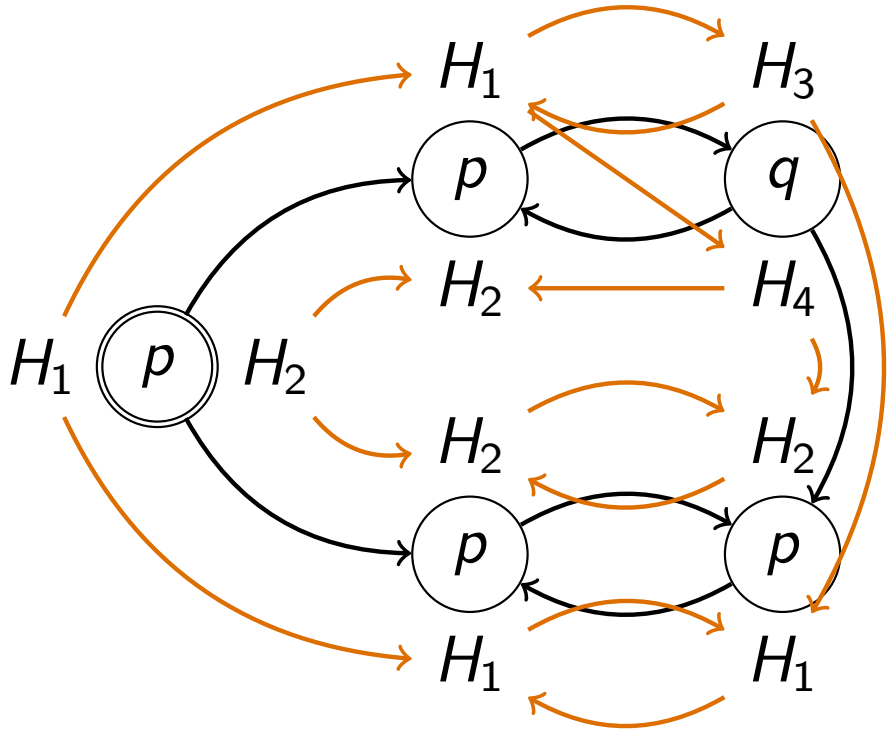
- вершины — всевозможные пары (s, H), где s — состояние M, и H — предположение, согласованное внутренне и с s
- $(s, H) \to (s', H') \Leftrightarrow s \to s'$ и пара (H, H') согласованна внешне
- вершина (s, H) объявляется начальной, если s — начальное состояние модели и $\varphi^- \in H$
- каждой подформуле Φ вида $\psi \mathbf{U} \chi$ формулы φ присваивается уникальный цвет c_Φ
- вершина (s, H) помечается цветом c_Φ, если в H звенит звонок для формулы Φ

Система Хинтикки

Вернёмся к примеру:

φ: $p\mathbf{U}q$

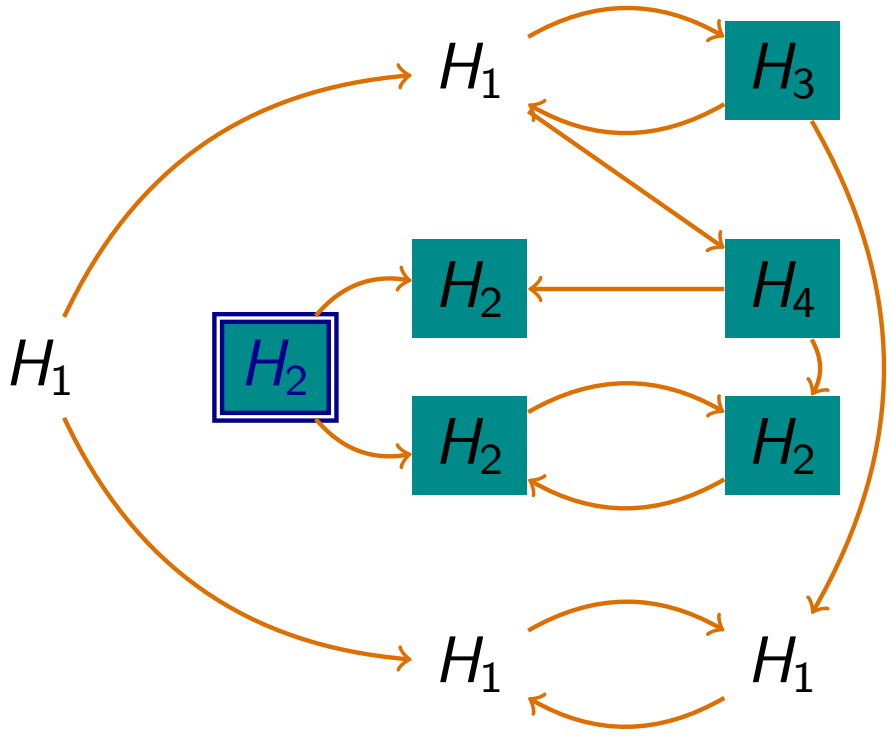
M:



$$H_1 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_2 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^-\}$$
$$H_3 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_4 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^+\}$$

Система Хинтикки

Система Хинтикки для M и φ:



$$H_1 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_2 = \{p^+, q^-, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^-\}$$
$$H_3 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^+, (p\mathbf{U}q)^+\}$$
$$H_4 = \{p^-, q^+, (\mathbf{X}(p\mathbf{U}q))^-, (p\mathbf{U}q)^+\}$$

Система Хинтикки

Бесконечный путь в системе Хинтикки является опровергающим, если он

- исходит из начальной вершины и
- для каждого цвета c_Φ содержит бесконечно много вершин, окрашенных в этот цвет

Основная теорема
$M \not\models \varphi \Leftrightarrow$ **в системе Хинтикки** $\mathfrak{H}$ **для** M **и** φ **существует опровергающий путь**

Доказательство.($\Leftarrow$):

Рассмотрим опровергающий путь в $\mathfrak{H}$:

$$(s_1, H_1) \to (s_2, H_2) \to \cdots \to (s_n, H_n) \to \ldots$$

По определению системы $\mathfrak{H}$, в M существует вычисление tr:

$$s_1 \to s_2 \to \cdots \to s_n \to \ldots$$

Доказательство основной теоремы ($\Leftarrow$):

$$(s_1, H_1) \to (s_2, H_2) \to \cdots \to (s_n, H_n) \to \ldots$$
$$tr : s_1 \to s_2 \to \cdots \to s_n \to \ldots$$

Покажем индукцией по построению формул, что для любой формулы ψ из $[\varphi]_{fl}$ и для любого натурального n справедливо:

$$\psi^+ \in H_n \quad \Leftrightarrow \quad tr|^n \models H_n$$

Если удастся это показать, то:

- по определению системы Хинтикки, $\varphi^- \in H_1$
- по внутренней согласованности H_1, $\varphi^+ \notin H_1$
- по тому, что предстоит доказать, $tr \not\models \varphi$
- по определению выполнимости LTL-формулы на LTS, $M \not\models \varphi$

Доказательство основной теоремы ($\Leftarrow$):

$$(s_1, H_1) \to (s_2, H_2) \to \cdots \to (s_n, H_n) \to \ldots$$

$$tr : s_1 \to s_2 \to \cdots \to s_n \to \ldots$$

База индукции: $p^+ \in H_n \Leftrightarrow tr|_n \models p$

$$\begin{array}{rl} p^+ \in H_n & \\ \Leftrightarrow & \text{согласованность с состоянием} \\ p \in \rho(s_n) & \\ \Leftrightarrow & \text{семантика LTL-формул} \\ tr|_n \models p & \end{array}$$

Индуктивный переход 1: $\neg\psi^+ \in H_n \Leftrightarrow tr|_n \models \neg\psi$

$$\begin{array}{rl} (\neg\psi)^+ \in H_n & \\ \Leftrightarrow & \text{сокращение} \\ \psi^- \in H_n & \\ \Leftrightarrow & \text{внутренняя согласованность} \\ \psi^+ \notin H_n & \\ \Leftrightarrow & \text{индуктивное предположение} \\ tr|_n \not\models \psi & \\ \Leftrightarrow & \text{семантика LTL-формул} \\ tr|_n \models \neg\psi & \end{array}$$

Доказательство основной теоремы ($\Leftarrow$):

Индуктивный переход 2: $\psi \,\&\, \chi^+ \in H_n \Leftrightarrow tr|_n \models \psi \,\&\, \chi$

$$
\begin{array}{rl}
\psi \,\&\, \chi^+ \in H_n & \\
\Leftrightarrow & \text{внутренняя согласованность} \\
\psi^+ \in H_n \text{ и } \chi^+ \in H_n & \\
\Leftrightarrow & \text{индуктивное предположение} \\
tr|_n \models \psi \text{ и } tr|_n \models \chi & \\
\Leftrightarrow & \text{семантика LTL-формул} \\
tr|_n \models \psi \,\&\, \chi &
\end{array}
$$

Индуктивный переход 3: $\mathbf{X}\psi^+ \in H_n \Leftrightarrow tr|_n \models \mathbf{X}\psi$

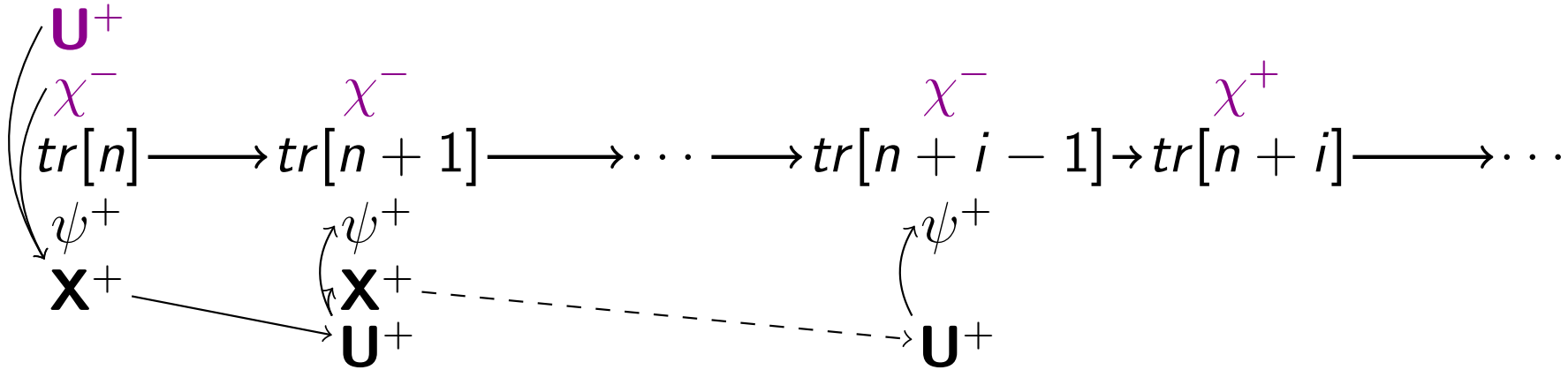
$$
\begin{array}{rl}
\mathbf{X}\psi^+ \in H_n & \\
\Leftrightarrow & \text{внешняя согласованность} \\
\psi^+ \in H_{n+1} & \\
\Leftrightarrow & \text{индуктивное предположение} \\
tr|_{n+1} \models \psi & \\
\Leftrightarrow & \text{семантика LTL-формул} \\
tr|_n \models \mathbf{X}\psi &
\end{array}
$$

Доказательство основной теоремы ($\Leftarrow$):

Индуктивный переход 4: $\psi \mathbf{U} \chi^+ \in H_n \Rightarrow tr|_n \models \psi \mathbf{U} \chi$

Случай 1: существует i, $i \geq 0$, такое что $\chi^+ \in H_{n+i}$

Без ограничения общности рассмотрим наименьшее такое i:



Внутренняя согласованность:

$$\{(\psi \mathbf{U} \chi)^+, \chi^-\} \subseteq H_j \Rightarrow \{\mathbf{X}(\psi \mathbf{U} \chi)^+, \psi^+\} \subseteq H_j$$

Внешняя согласованность: $\mathbf{X}(\psi \mathbf{U} \chi)^+ \in H_j \Rightarrow \psi \mathbf{U} \chi \in H_{j+1}$

Индуктивное предположение:

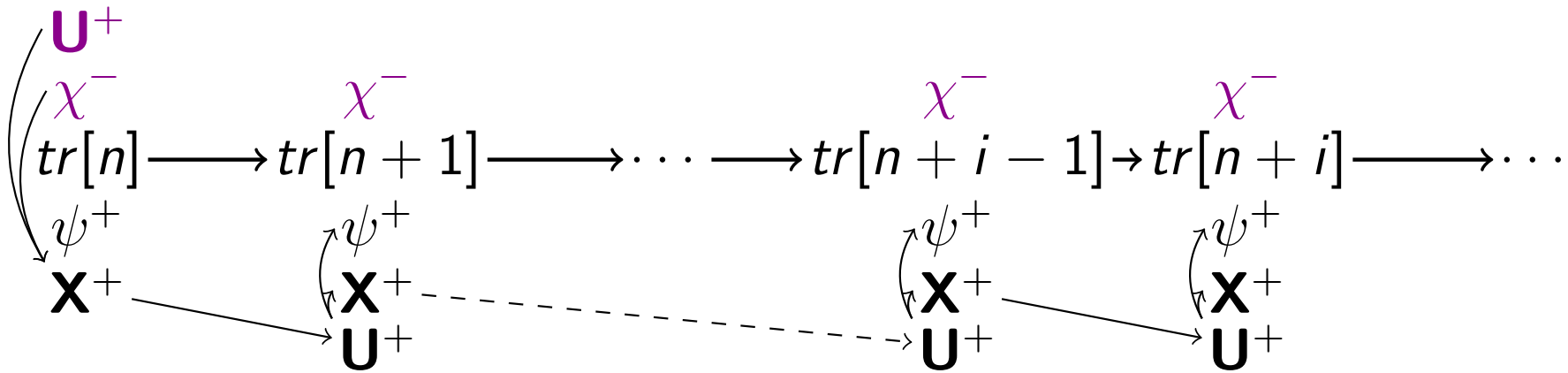
$$\psi^+ \in H_j \Rightarrow tr|^j \models \psi; \qquad \chi^+ \in H_j \Rightarrow tr|^j \models \chi$$

Семантика LTL-формул: $tr|^n \models \psi \mathbf{U} \chi$

Доказательство основной теоремы ($\Leftarrow$):

Индуктивный переход 4: $\psi \mathbf{U} \chi^+ \in H_n \Rightarrow tr|_n \models \psi \mathbf{U} \chi$

Случай 2: **не** существует i, $i \geq 0$, такого что $\chi^+ \in H_{n+i}$



Теми же рассуждениями, что и в случае 1, получаем:

не существует i, $i \geq 0$, **такого что** $\mathbf{X}(\psi \mathbf{U} \chi)^- \in H_{n+i}$

Значит, в предположениях этой трассы **никогда не звенит звонок**, и в предположениях исходного рассматриваемого пути в системе Хинтикки **звонок звенит лишь конечное число раз** — это противоречит тому, что выбран опровергающий путь: случай 2 невозможен

Доказательство основной теоремы ($\Leftarrow$):

Индуктивный переход 5: $\psi \mathbf{U} \chi^+ \in H_n \Leftarrow tr|_n \models \psi \mathbf{U} \chi$

Семантика LTL-формул: существует i, $i \geq 0$, такое что:

$$(tr|^n \models \psi), \quad \ldots, \quad (tr|^{n+i-1} \models \psi), \quad (tr|^{n+i} \models \chi)$$

Без ограничения общности рассмотрим наименьшее такое i

Индуктивное предположение:

$$\{\psi^+, \chi^-\} \subseteq H_n, \quad \ldots, \quad \{\psi^+, \chi^-\} \subseteq H_{n+i-1}, \quad \chi^+ \in H_{n+i}$$

Далее рассуждениями о внешней и внутренней согласованности, аналогичными рассуждениям в переходе 4, можно легко получить соотношение $\psi \mathbf{U} \chi^+ \in H_n$

Доказательство достаточности завершено

Доказательство основной теоремы ($\Rightarrow$):

Дано: $M \not\models \varphi$
Требуется показать, что в системе Хинтикки $\mathfrak{H}$ для M и φ существует опровергающий путь

$M \not\models \varphi$ означает, что существует вычисление tr LTS M, такое что $tr \not\models \varphi$

Рассмотрим предположения H_i, $i \geq 1$, следующего вида:

$$H_i = \{\psi^+ \mid \psi \in [\varphi]_{fl}, tr|^i \models \psi\} \cup \{\psi^- \mid \psi \in [\varphi]_{fl}, tr|^i \not\models \psi\}$$

Рассмотрим последовательность $path$ следующего вида:

$$(tr[1], H_1), (tr[2], H_2), \ldots, (tr[i], H_i), \ldots$$

Покажем, что $path$ — опровергающий путь в $\mathfrak{H}$

Доказательство основной теоремы ($\Rightarrow$):

$$tr \not\models \varphi$$

$$H_i = \{\psi^+ \mid \psi \in [\varphi]_{fl}, tr|^i \models \psi\} \cup \{\psi^- \mid \psi \in [\varphi]_{fl}, tr|^i \not\models \psi\}$$

$$path : (tr[1], H_1), (tr[2], H_2), \ldots, (tr[i], H_i), \ldots$$

По доказанным ранее утверждениям,

- каждое предположение H_i согласованно с $tr[i]$ и внутренне
- каждая пара предположений (H_i, H_{i+1}) согласованна внешне

По выбору предположений, $\varphi^- \in H_1$

Значит, *path* — путь в $\mathfrak{H}$, исходящий из начальной вершины

Осталось показать, что в *path* каждый цвет встречается бесконечно часто

Доказательство основной теоремы ($\Rightarrow$):

$$tr \not\models \varphi$$

$$H_i = \{\psi^+ \mid \psi \in [\varphi]_{fl}, tr|^i \models \psi\} \cup \{\psi^- \mid \psi \in [\varphi]_{fl}, tr|^i \not\models \psi\}$$

$$path : (tr[1], H_1), (tr[2], H_2), \ldots, (tr[i], H_i), \ldots$$

Предположим от противного, что существует цвет $c_{\psi \mathbf{U} \chi}$, встречающийся в *path* конечное число раз

Тогда существует i, $i \geq 0$, такое что ни одна вершина пути $path|^i$ не окрашена в этот цвет

Значит, $\{\chi^-, \mathbf{X}(\psi \mathbf{U} \chi)^+\} \subseteq H_{i+k}$ для любого k, $k \geq 0$

По выбору предположений H_j,

- $tr|^{i+1+k} \not\models \chi$ для всех k, $k \geq 0$, а значит, $tr|^{i+1} \not\models \psi \mathbf{U} \chi$
- $tr|^i \models \mathbf{X}(\psi \mathbf{U} \chi)$, а значит, $tr|^{i+1} \models \psi \mathbf{U} \chi$ — **противоречие** ▼

Проверка наличия опровергающего пути

Компонента сильной связности ориентированного графа — это множество S вершин графа, такое что:

- для любой пары вершин v_1, v_2 множества S существует путь $v_1 \to \cdots \to v_2$
- любая вершина графа, не входящая в S, не лежит ни на каком пути, начинающемся и оканчивающемся в S

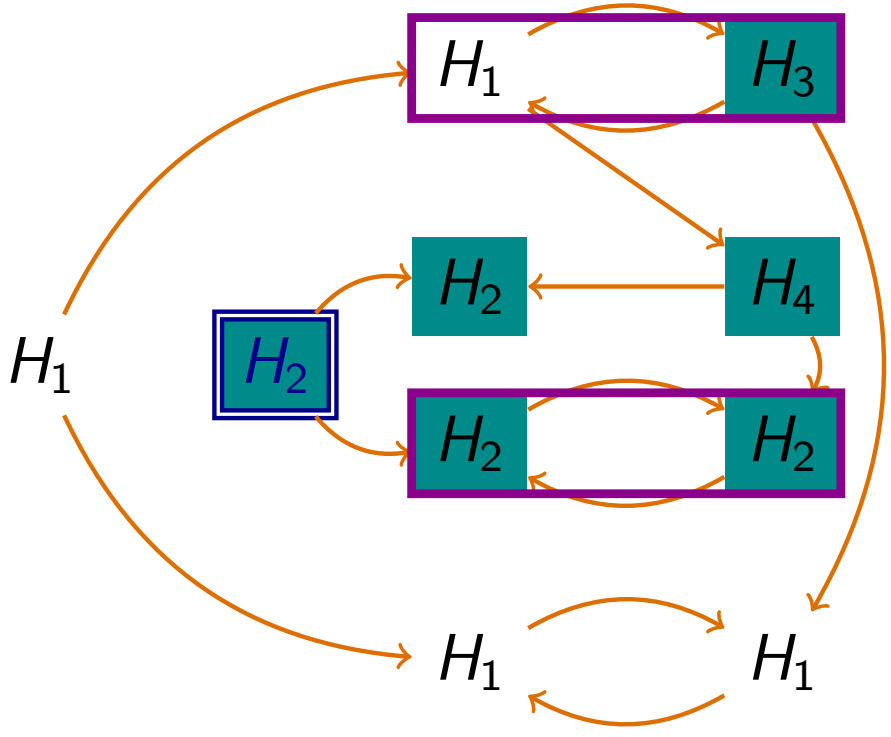
Вспомогательная теорема

В системе Хинтикки $\mathfrak{H}$ существует опровергающий путь $\Leftrightarrow$ в $\mathfrak{H}$ существует начальная вершина, из которой достижима компонента сильной связности, содержащая все цвета

Доказательство. На грани очевидного

Проверка наличия опровергающего пути

Обратно к примеру:



Две подходящие компоненты сильной связности обведены прямоугольниками

Одна из них достижима из начальной вершины, а значит,

$$M \not\models p\mathbf{U}q$$

Заключение

Несколько вопросов, размышления над которыми помогут лучше понять, как устроен табличный алгоритм model checking для LTL:

- Сколько формул содержится в множестве $[\varphi]_{fl}$? (например, относительно числа операций в φ)
- Сколько вершин содержится в системе Хинтикки для LTS M и LTL-формулы φ?
- Какую сложность имеет описанный алгоритм?
- Можно ли изменить алгоритм так, чтобы при построении графа, аналогичного системе Хинтикки, использовался только **один цвет**?
- Можно ли изменить алгоритм так, чтобы система Хинтикки не строилась целиком?