Models and Languages for Computational Systems Biology

Lecture 6: Branching Time and CTL

Ian Stark

School of Informatics
The University of Edinburgh

Thursday 25 January 2010
Semester 2 Week 3

Linear Temporal Logic

$$\mathrm{P} ::= \mathit{Atom} \mid \mathrm{P} \wedge \mathrm{Q} \mid \mathrm{P} \vee \mathrm{Q} \mid \mathrm{P} \Rightarrow \mathrm{Q} \mid \neg \mathrm{P}$$
$$\mathbf{X}\mathrm{P} \mid \mathbf{F}\mathrm{P} \mid \mathbf{G}\mathrm{P} \mid \mathrm{P}\,\mathbf{U}\,\mathrm{Q} \mid \mathrm{P}\,\mathbf{R}\,\mathrm{Q}$$

Atom	Some basic predicate over system states
$\mathbf{X}\mathrm{P}$	Next: P holds after a single step
$\mathbf{F}\mathrm{P}$	Finally/Future: P holds at some later state
$\mathbf{G}\mathrm{P}$	Globally: P holds at this and all later states
$\mathrm{P}\,\mathbf{U}\,\mathrm{Q}$	Until: Q holds in the future, and P holds until then
$\mathrm{P}\,\mathbf{R}\,\mathrm{Q}$	Release: Q holds indefinitely, or until P also holds

An formula P of LTL is an assertion about a single *run* of a labelled transition system.

Linear Temporal Logic

$$\mathrm{P} ::= \mathit{Atom} \mid \mathrm{P} \wedge \mathrm{Q} \mid \mathrm{P} \vee \mathrm{Q} \mid \mathrm{P} \Rightarrow \mathrm{Q} \mid \neg \mathrm{P}$$
$$\mathbf{X}\mathrm{P} \mid \mathbf{F}\mathrm{P} \mid \mathbf{G}\mathrm{P} \mid \mathrm{P}\,\mathbf{U}\,\mathrm{Q} \mid \mathrm{P}\,\mathbf{R}\,\mathrm{Q}$$

Atom	Some basic predicate over system states
$\mathbf{X}\mathrm{P}$	Next: P holds after a single step
$\mathbf{F}\mathrm{P}$	Finally/Future: P holds at some later state
$\mathbf{G}\mathrm{P}$	Globally: P holds at this and all later states
$\mathrm{P}\,\mathbf{U}\,\mathrm{Q}$	Until: Q holds in the future, and P holds until then
$\mathrm{P}\,\mathbf{R}\,\mathrm{Q}$	Release: Q holds indefinitely, or until P also holds

This interpretation may be extended to all runs from some given start state.

Linear Temporal Logic

$$P ::= \mathit{Atom} \mid P \wedge Q \mid P \vee Q \mid P \Rightarrow Q \mid \neg P$$
$$\mathbf{X}P \mid aP \mid \mathbf{F}P \mid \mathbf{G}P \mid P\,\mathbf{U}\,Q \mid P\,\mathbf{R}\,Q$$

Atom	Some basic predicate over system states
$\mathbf{X}P$	Next: P holds after a single step
aP	Transition: next step is a, after which P holds
$\mathbf{F}P$	Finally/Future: P holds at some later state
$\mathbf{G}P$	Globally: P holds at this and all later states
$P\,\mathbf{U}\,Q$	Until: Q holds in the future, and P holds until then
$P\,\mathbf{R}\,Q$	Release: Q holds indefinitely, or until P also holds

We can add additional forms, such as a modality for each transition.

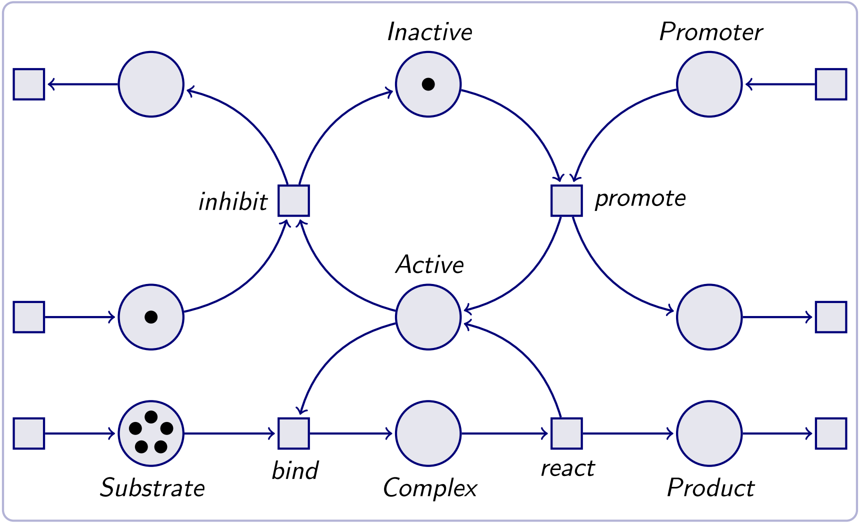
Linear Temporal Logic

$$P ::= \mathit{Atom} \mid P \wedge Q \mid P \vee Q \mid P \Rightarrow Q \mid \neg P$$
$$\mathbf{X}P \mid aP \mid \mathbf{F}P \mid \mathbf{G}P \mid P\,\mathbf{U}\,Q \mid P\,\mathbf{R}\,Q$$

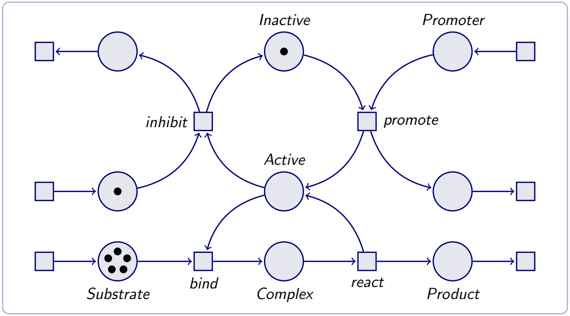
Atom	Some basic predicate over system states
$\mathbf{X}P$	Next: P holds after a single step
aP	Transition: next step is a, after which P holds
$\mathbf{F}P$	Finally/Future: P holds at some later state
$\mathbf{G}P$	Globally: P holds at this and all later states
$P\,\mathbf{U}\,Q$	Until: Q holds in the future, and P holds until then
$P\,\mathbf{R}\,Q$	Release: Q holds indefinitely, or until P also holds

The act of checking whether a formula holds of a system is *model-checking*.

LTL Examples



LTL Examples

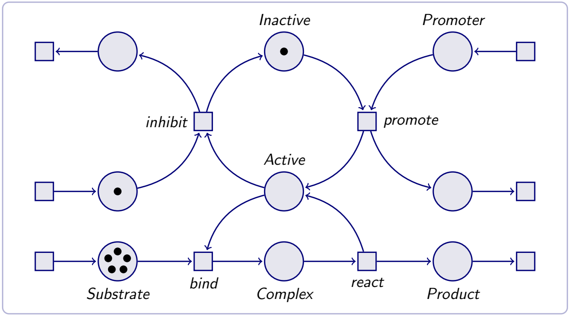


$\mathbf{X}(\mathit{Active})$

$\mathbf{F}(\mathit{Active})$

$\mathbf{G}(\mathit{Inactive} \wedge \mathit{Promoter} \Rightarrow \mathbf{X}((\neg \mathit{Promoter}) \Rightarrow \mathit{Active}))$

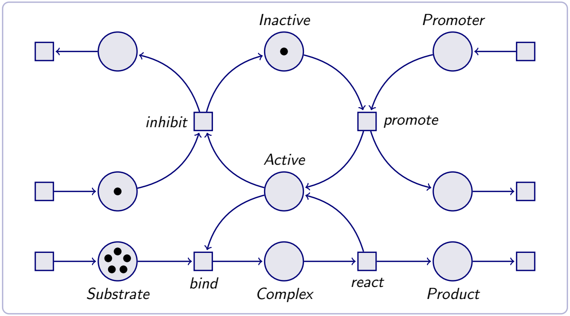
LTL Examples



$$promote.Active$$

$$\neg(promote.(\neg Active))$$

LTL Examples



$\mathbf{GF}(\mathit{Active})$

$\mathbf{F}(\mathit{Product}) \Rightarrow \mathbf{F}(\mathit{Active}))$

$\mathbf{G}(\mathit{Substrate} \wedge \mathbf{F}(\neg \mathit{Substrate}) \Rightarrow \mathbf{F}(\mathit{Active}))$

$\mathbf{G}(\mathit{Substrate} \Rightarrow (\mathit{Substrate}\,\mathbf{U}\,\mathit{Active}))$

$\mathbf{G}(\mathit{Substrate} \Rightarrow (\mathit{Active}\,\mathbf{R}\,\mathit{Substrate}))$

Further Reading

M. Heiner, D. Gilbert, and R. Donaldson.
Petri Nets for systems and synthetic biology.
In *Formal Methods for Computational Systems Biology*, LNCS 5016, pp. 215–264. Springer-Verlag, 2008.
Section 4.2(6) Model Checking of Special Behavioural Properties.

P. T. Monteiro, D. Ropers, R. Mateescu, A. T. Freitas, and H. de Jong.
Temporal logic patterns for querying dynamic models of cellular interaction networks.
Bioinformatics, 24(16):227–233, 2008.

F. Fages and A. Rizk.
On the analysis of numerical data time series in temporal logic.
In *Computational Methods in Systems Biology: Proceedings of the International Conference CMSB 2007*, LNBI 4695, pp. 48–63. Springer-Verlag, 2007.