Regression

- The formula $\text{regr}_o(\phi)$ represents the set of states from which a state in ϕ is reached by operator o.
- Used in *backward search* in a transition system: from the goal states toward the initial states.
- Regression is powerful because it allows handling huge sets of states (progression: only one state at a time.)
- Handling formulae is more complicated than handling states: many questions about regression are NP-hard.

Regression for simple operators

1. Goals are conjunctions of literals.
2. Operator preconditions are conjunctions of literals.
3. Operators have no conditional effects.

Hence every operator is of the form

$$\langle l_1 \wedge \cdots \wedge l_n, \;\; l'_1 \wedge \cdots \wedge l'_m \rangle$$

where l_i and l'_j are literals. Call this kinds of operators simple.

Regression for simple operators

1. The goal is $l_1 \wedge \cdots \wedge l_n$.
2. Choose an operator that makes some of $l_1, \ldots, l_n$ true and makes none of them false.
3. Form a new goal by removing the fulfilled goal literals and adding the preconditions of the operator.

Regression for simple operators

Given a goal $l''_1 \wedge \cdots \wedge l''_{m'}$, choose an operator

$$\langle l_1 \wedge \cdots \wedge l_n, \;\; l'_1 \wedge \cdots \wedge l'_m \rangle$$

so that

1. $\{l', \ldots, l'_m\} \cap \{\overline{l''_1}, \ldots, \overline{l''_{m'}}\} = \emptyset$
2. $\{l', \ldots, l'_m\} \cap \{l''_1, \ldots, l''_{m'}\} \neq \emptyset$

Regression for simple operators

The *regression* $\text{regr}_o(\phi)$ of $\phi = l''_1 \wedge \cdots \wedge l''_{m'}$ with respect to

$$o = \langle l_1 \wedge \cdots \wedge l_n,\ \ l'_1 \wedge \cdots \wedge l'_m \rangle$$

such that $\{l', \ldots, l'_m\} \cap \{\overline{l''_1}, \ldots, \overline{l''_{m'}}\} = \emptyset$ is the following conjunction of literals.

$$\bigwedge ((\{l''_1, \ldots, l''_{m'}\} \backslash \{l'_1, \ldots, l'_m\}) \cup \{l_1, \cdots, l_n\})$$

Regression for simple operators

$$\begin{aligned}
\text{ABT} &= \langle \textit{AonB} \wedge \textit{Aclear}, \neg \textit{AonB} \wedge \textit{AonT} \wedge \textit{Bclear} \rangle \\
\text{BCA} &= \langle \textit{BonC} \wedge \textit{Bclear} \wedge \textit{Aclear}, \neg \textit{Aclear} \wedge \neg \textit{BonC} \wedge \textit{BonA} \wedge \textit{Cclear} \rangle \\
\text{CTB} &= \langle \textit{ConT} \wedge \textit{Cclear} \wedge \textit{Bclear}, \neg \textit{Bclear} \wedge \neg \textit{ConT} \wedge \textit{ConB} \rangle
\end{aligned}$$

$$\begin{aligned}
G &= \textit{ConB} \wedge \textit{BonA} \\
G_1 &= \text{regr}_{\textit{CTB}}(G) = \textit{BonA} \wedge \textit{ConT} \wedge \textit{Cclear} \wedge \textit{Bclear} \\
G_2 &= \text{regr}_{\textit{BCA}}(G_1) = \textit{ConT} \wedge \textit{Bclear} \wedge \textit{BonC} \wedge \textit{Aclear} \\
G_3 &= \text{regr}_{\textit{ABT}}(G_2) = \textit{ConT} \wedge \textit{BonC} \wedge \textit{Aclear} \wedge \textit{AonB}
\end{aligned}$$

Regression in the conjunctive case

LEMMA A: Let ϕ be a conjunction of literals, o a simple operator, and s and s' be states so that $s' = \text{app}_o(s)$. Then $s \models \text{regr}_o(\phi)$ if and only if $s' \models \phi$.

PROOF: Assume $s \models \text{regr}_o(\phi)$. We show that $s' \models \phi$.

Let l be any literal in ϕ.

If l is an effect of o, then $s' \models l$ by definition of $\text{app}_o(s)$.

If l is not an effect of o, then l is a conjunct of $\text{regr}_o(\phi)$, and hence $s \models l$. By definition of regression, $\bar{l}$ is not an effect of o. Hence by definition of $\text{app}_o(s)$ also $s' \models l$.

Assume that $s' \models \phi$. We show that $s \models \text{regr}_o(\phi)$.

Because $s' = \text{app}_o(s)$, the application of o in s is well-defined, and hence all preconditions of o are true in s.

Let l be one of the remaining literals in $\text{regr}_o(\phi)$, that is, one of the conjuncts of ϕ that are not effects of o.

Because o does not change the value of l and $s' \models \phi$, also $s \models l$.

Equivalences on effects

$$
\begin{aligned}
c \rhd (e_1 \wedge \cdots \wedge e_n) &\equiv (c \rhd e_1) \wedge \cdots \wedge (c \rhd e_n) && (1)\\
c \rhd (c' \rhd e) &\equiv (c \wedge c') \rhd e && (2)\\
(c_1 \rhd e) \wedge (c_2 \rhd e) &\equiv (c_1 \vee c_2) \rhd e && (3)\\
e \wedge (c \rhd e) &\equiv e && (4)\\
e &\equiv \top \rhd e && (5)\\
e &\equiv \top \wedge e && (6)\\
e \wedge e' &\equiv e' \wedge e && (7)\\
(e_1 \wedge e_2) \wedge e_3 &\equiv e_1 \wedge (e_2 \wedge e_3) && (8)
\end{aligned}
$$

Normal form for operators and effects

DEFINITION: An operator $\langle c, e \rangle$ is in normal form if for all occurrences of $c' \rhd e'$ in e the effect e' is either p or $\neg p$ for some $p \in P$, and e contains at most one occurrence of any atomic effect l.

THEOREM: For every operator there is an equivalent one in normal form.

Proof is constructive: we can transform any operator into normal form by using the equivalences from the previous slide.

Normal form for effects: example

$$
\begin{aligned}
(A &\rhd (B \wedge\\
& \quad (C \rhd (\neg D \wedge E)))) \wedge\\
(\neg B &\rhd E)
\end{aligned}
$$

transformed to normal form is

$$
\begin{aligned}
(A &\rhd B) \wedge\\
((A \wedge C) &\rhd \neg D) \wedge\\
((\neg B \vee (A \wedge C)) &\rhd E)
\end{aligned}
$$

Regression (for all operators)

1. When we have disjunction and conditional effects, things become more tricky. How to define regression e.g. for $A \vee (B \wedge C)$ with $\langle Q, D \rhd B \rangle$?
2. We present a general definition of how to do this.
3. Now we extensively use the idea of *representing sets of states as formulae*.

   The story about goals and subgoals and fulfilling subgoals, as in the conjunctive case, does not work any more.

Auxiliary definition: $\mathit{EPC}_l(e)$

DEFINITION The condition $\mathit{EPC}_l(e)$ of literal l becoming true when effect e is applied is defined as follows.

$$\begin{aligned}
\mathit{EPC}_l(l) &= \top \\
\mathit{EPC}_l(l') &= \bot \text{ when } l \neq l' \text{ (for literals } l') \\
\mathit{EPC}_l(\top) &= \bot \\
\mathit{EPC}_l(e_1 \wedge \cdots \wedge e_n) &= \mathit{EPC}_l(e_1) \vee \cdots \vee \mathit{EPC}_l(e_n) \\
\mathit{EPC}_l(c \rhd e) &= \mathit{EPC}_l(e) \wedge c
\end{aligned}$$

Auxiliary definition: $\mathit{EPC}_l(e)$, examples

$$\begin{aligned}
\mathit{EPC}_A(B \wedge C) &= \bot \vee \bot \equiv \bot \\
\mathit{EPC}_A(A \wedge (B \rhd A)) &= \top \vee (\top \wedge B) \equiv \top \\
\mathit{EPC}_A((C \rhd A) \wedge (B \rhd A)) &= (\top \wedge C) \vee (\top \wedge B) \equiv C \vee B
\end{aligned}$$

Auxiliary definition: $\mathit{EPC}_l(e)$, lemma

LEMMA B: Let s be a state, l a literal and e an effect. Then $l \in [e]_s$ if and only if $s \models \mathit{EPC}_l(e)$.

PROOF: by induction on the structure of the effect e.

Base case 1, $e = \top$: By definition of $[\top]_s$ we have $l \notin [\top]_s = \emptyset$, and by definition of $\mathit{EPC}_l(\top)$ we have $s \not\models \mathit{EPC}_l(\top) = \bot$, so the equivalence holds.

Base case 2, $e = l$: $l \in [l]_s = \{l\}$ by definition, and $s \models \mathit{EPC}_l(l) = \top$ by definition.

Base case 3, $e = l'$ for some literal $l' \neq l$: $l \notin [l']_s = \{'l\}$ by definition, and $s \not\models \mathit{EPC}_l(l') = \bot$ by definition.

Inductive case 1, $e = e_1 \wedge \cdots \wedge e_n$:

$$\begin{aligned}
l \in [e]_s \quad & \text{if and only if } l \in [e']_s \text{ for some } e' \in \{e_1, \ldots, e_n\} \\
& \text{if and only if } s \models \mathit{EPC}_l(e') \text{ for some } e' \in \{e_1, \ldots, e_n\} \\
& \text{if and only if } s \models \mathit{EPC}_l(e_1) \vee \cdots \vee \mathit{EPC}_l(e_n) \\
& \text{if and only if } s \models \mathit{EPC}_l(e_1 \wedge \cdots \wedge e_n).
\end{aligned}$$

Inductive case 2, $e = c \rhd e'$:

$$\begin{aligned}
l \in [c \rhd e']_s \quad & \text{if and only if } l \in [e']_s \text{ and } s \models c \\
& \text{if and only if } s \models \mathit{EPC}_l(e') \text{ and } s \models c \\
& \text{if and only if } s \models \mathit{EPC}_l(c \rhd e').
\end{aligned}$$

Q.E.D.

Auxiliary definition: $EPC_l(e)$ vs. normal form

Notice that in terms of $EPC_p(e)$ any operator $\langle c, e\rangle$ can be expressed in normal form as

$$\left\langle c, \bigwedge_{p\in P} (EPC_p(e) \rhd p) \wedge (EPC_{\neg p}(e) \rhd \neg p)\right\rangle.$$

Regression: definition for literals

The formula $(p \wedge \neg EPC_{\neg p}(e)) \vee EPC_p(e)$ expresses the truth-value of $p \in P$ after applying o in terms of truth-values of formulae before applying o: either

- p was true before and did not become false, or
- p became true.

Regression: definition for literals, examples

Let $e = (B \rhd A) \wedge (C \rhd \neg A) \wedge B \wedge \neg D$.

p	$(p \wedge \neg EPC_{\neg p}(e)) \vee EPC_p(e)$
A	$(A \wedge \neg C) \vee B$
B	$(B \wedge \neg\bot) \vee \top \equiv \top$
C	$(C \wedge \neg\bot) \vee \bot \equiv C$
D	$(D \wedge \neg\top) \vee \bot \equiv \bot$

Regression: definition for literals

LEMMA C: Let p be a state variable and $o = \langle c, e\rangle \in O$ and operator. Let s be a state and $s' = \mathrm{app}_o(s)$. Then $s \models (p \wedge \neg EPC_{\neg p}(e)) \vee EPC_p(e)$ if and only if $s' \models p$.

PROOF: Assume $s \models (p \wedge \neg EPC_{\neg p}(e)) \vee EPC_p(e)$. Do a case analysis on the two disjuncts.

Case 1: Assume that $s \models p \wedge \neg EPC_{\neg p}(e)$. By Lemma B $\neg p \notin [e]_s$. Hence p remains true in s'.

Case 2: Assume that $s \models EPC_p(e)$. By Lemma B $p \in [e]_s$, and hence $s' \models p$.

For the other half of the equivalence, assume that $s \not\models (p \land \neg \mathit{EPC}_{\neg p}(e)) \lor \mathit{EPC}_p(e)$.

Hence $s \models (\neg p \lor \mathit{EPC}_{\neg p}(e)) \land \neg \mathit{EPC}_p(e)$.

Assume that $s \models p$. Now $s \models \mathit{EPC}_{\neg p}(e)$ because $s \models \neg p \lor \mathit{EPC}_{\neg p}(e)$, and hence by Lemma B $\neg p \in [e]_s$ and hence $s' \not\models p$.

Assume that $s \not\models p$. Because $s \models \neg \mathit{EPC}_p(e)$, by Lemma B $p \notin [e]_s$ and hence $s' \not\models p$.

Therefore it must be that $s' \not\models p$. Q.E.D.

Regression: definition for formulae

The formula $\mathit{EPC}_l(o)$ can now be used in defining regression for operators o.

DEFINITION Let ϕ be a propositional formula. Let $\langle z, e \rangle$ be an operator. The *regression* of ϕ with respect to $o = \langle z, e \rangle$ is $\mathrm{regr}_e(\phi) = \phi_r \land z \land f$ where ϕ_r is obtained from ϕ by replacing $p \in P$ by $(p \land \neg \mathit{EPC}_{\neg p}(e)) \lor \mathit{EPC}_p(e)$, and $f = \bigwedge_{p \in P} \neg(\mathit{EPC}_p(e) \land \mathit{EPC}_{\neg p}(e))$.

The conjuncts of f say that no state variable may become both true and false.

Regression: examples

$\mathrm{regr}_{\langle a,b \rangle}(b) = (((b \land \neg\bot) \lor \top) \land a) \equiv a$

$\mathrm{regr}_{\langle a,b \rangle}(b \land c \land d) = (((b \land \neg\bot) \lor \top) \land ((c \land \neg\bot) \lor \bot) \land ((d \land \neg\bot) \lor \bot) \land a) \equiv c \land d \land a$

$\mathrm{regr}_{\langle a,c \rhd b \rangle}(b) = (((b \land \neg\bot) \lor c) \land a) \equiv (b \lor c) \land a$

$\mathrm{regr}_{\langle a,(c \rhd b) \land (b \rhd \neg b) \rangle}(b) = (((b \land \neg b) \lor c) \land a \land \neg(c \land b)) \equiv c \land a \land \neg b$

$\mathrm{regr}_{\langle a,(c \rhd b) \land (d \rhd \neg b) \rangle}(b) = (((b \land \neg d) \lor c) \land a \land \neg(c \land d)) \equiv (b \lor c) \land (\neg d \lor c) \land a \land (\neg c \lor d)$