REASONING ABOUT PROBABILITIES IN DYNAMICAL DOMAINS

FROM SPECIFICATION TO GOAL REGRESSION AND BEYOND

Vaishak Belle
Department of Computer Science
University of Toronto

(Joint work with Hector Levesque)

DYNAMICAL SYSTEMS

Many (AI) systems operate in dynamical worlds, where properties change and are unknown.

In general, require **language** for **representing** actions and incomplete information (*i.e.* knowledge).

But also **computational mechanisms** for **reasoning**.

However, 2 disparate paradigms: **logical** and **probabilistic**

However, 2 disparate paradigms: **logical** and **probabilistic**

Logic camp: explicit actions, strict uncertainty (*i.e.* disjunctions, quantification)

- *e.g.* STRIPS, situation calculus, dynamic logic

However, 2 disparate paradigms: **logical** and **probabilistic**

Logic camp: explicit actions, strict uncertainty (*i.e.* disjunctions, quantification)

- *e.g.* STRIPS, situation calculus, dynamic logic

Probability camp: random variables over some joint distribution, transition dynamics over continuous probability distributions

e.g. Bayesian Networks, Kalman filters

logical or probabilistic?

Clearly, most applications would benefit from both; e.g. cognitive robotics, but also noisy data

logical or probabilistic?

Clearly, most applications would benefit from both; e.g. cognitive robotics, but also noisy data

Agenda: foundations for representing and reasoning with all of logic and all of probability

From **specification** to any required **fragment**

(some) related work

Many with limited first-order features.

- Bayes nets, filtering mechanisms, hybrid systems: *no strict uncertainty*, *explicit actions* not always treated
- Bayesian logic, Markov logics: *no explicit actions*
- logic for reasoning about probability, e.g. Bacchus 1990: *no actions*
- logics for probability and time, e.g. Halpern-Tuttle: *only discrete prob.*
- planning languages: *no strict uncertainty, limited contextual* features
- action languages/program logics: lacking *continuous fluents/ continuous noise*

THE SITUATION CALCULUS

The situation calculus is a dialect of FOL for representing dynamically changing worlds in which all changes are the result of named actions[1].

3 sorts: **actions** *e.g. put(x,y)*, **situations** (histories) and **objects** (catch-all):

- S_0 (initial) and $do(put(x, y), S_0)$

Predicates or functions whose values may vary from situation to situation are called **fluents**, *e.g.* $\neg Broken(x, S_0)$ but $Broken(x, do(drop(x), S_0))$.

[1] [McCarthy+Hayes 69, Reiter 01]

MODELING A DOMAIN

A logical theory $\mathcal{D}$:

- initial knowledge base: any FOL theory, e.g.
  $\exists x. Broken(x, S_0), value(house1, S_0) = 1000 \vee value(house1, S_0) = 2000$
- preconditions, *e.g.* $Poss(pickup(x), s) \equiv \forall z \neg Holding(z, s)$
- successor state axioms (solution to frame problem)

$$\begin{aligned}\forall a, s.\ Broken(x, do(a, s)) \equiv& \\ & a = drop(x) \wedge Holding(x, s) \wedge Fragile(x) \vee \\ & Broken(x, s) \wedge a \neq repair(x).\end{aligned}$$

Projection Problem: $\mathcal{D} \models \phi[do(\vec{a}, S_0)]$? (Tarskian semantics)

fundamental for planning, verification, *etc.*

KNOWLEDGE IN THE SITUATION CALCULUS

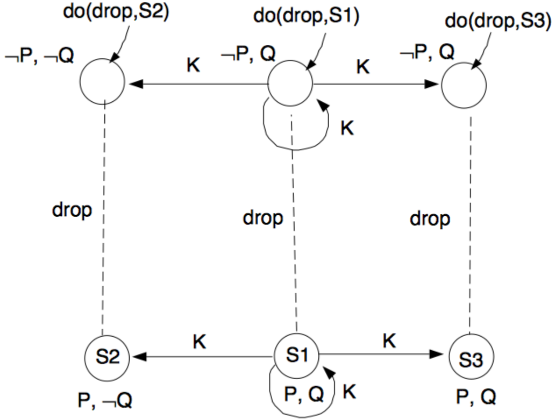
Treat *situations* as possible worlds [Moore 85, Scherl+Levesque 03]

- *i.e.* initial situations other than S_0
- special fluent K: $K(s', s)$ means s' is accessible from s
- $Knows(\phi, s) \doteq \forall s'.\ K(s', s) \supset \phi[s']$
- $\mathcal{D}$ now includes **sensing axioms**: $SF(checkRed(x), s) \equiv Red(x, s)$.

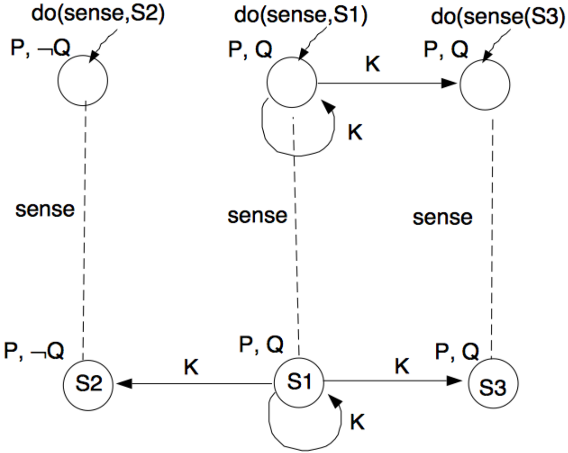
Also a SSA for K:

$$
\begin{aligned}
&K(s', do(a, s)) \equiv \\
&\qquad \exists s''.\ s' = do(a, s'') \wedge K(s'', s) \wedge \\
&\qquad\qquad Poss(a, s'') \wedge SF(a, s'') \equiv SF(a, s).
\end{aligned}
$$

dropping makes *P* false



sensing tells you whether Q holds



KNOWLEDGE: SUMMARY

- K successor state axiom (fixed) + SF axioms (domain dependent) in $\mathcal{D}$
- definition of knowledge and how that changes after sensing;[2]

[2] also see [Lakemeyer+Levesque 04, Van Ditmarsch+ 2011]

KNOWLEDGE: SUMMARY

- K successor state axiom (fixed) + SF axioms (domain dependent) in $\mathcal{D}$
- definition of knowledge and how that changes after sensing;[2]
- Epistemic formulas are **regressable**

  $\mathcal{D} \models Knows(\phi, do(\vec{a}, S_0))$ iff $\mathcal{D}_0 \models \mathcal{R}[Knows(\phi, do(\vec{a}, S_0))]$

  $$e.g.\ \mathcal{R}[Broken(g, do(drop(g, S_0)))] =$$
  $$(a = drop(x) \vee Broken(x, S_0) \wedge a \neq repair(x))^{a,x}_{drop(g),g}$$
  $$e.g.\ \mathcal{R}[Knows(\phi, do(checkRed(g), S_0))] =$$
  $$Red(g, S_0) \supset Knows(Red(g, now) \supset \phi, S_0) \vee$$
  $$\neg Red(g, S_0) \supset Knows(\neg Red(g, now) \supset \phi, S_0)$$

[2] also see [Lakemeyer+Levesque 04, Van Ditmarsch+ 2011]

Belief via special fluent p:[3] $p(s', s)$ gives **weight** accorded to s' when at s

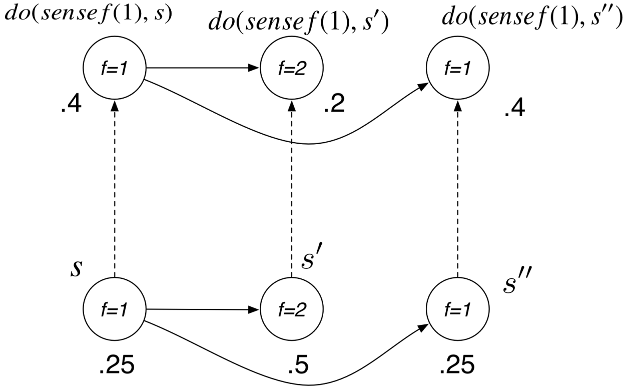
l for **action likelihoods**: $l(sonar(z), s) = \mathcal{N}(z; distance(s), 4)$.

Successor state axiom for p:

$$\begin{aligned} p(s', do(a, s)) = u \;\equiv \\ &\exists s'' \, [s' = do(a, s'') \wedge Poss(a, s'') \wedge \\ &\quad u = p(s'', s) \times l(a, s'')] \\ &\vee \neg\exists s'' \, [s' = do(a, s'') \wedge Poss(a, s'') \wedge u = 0] \end{aligned}$$

[3] [Bacchus+ 1999], following *e.g.* [Fagin+Halpern 1994]

noisy sensor says $f = 1$



MODELING A DOMAIN

- l axioms (domain dependent) + successor state axiom for p (fixed)
- p axioms in $\mathcal{D}_0$

  e.g. $p(s, S_0) = u \equiv (f(s) = 1 \wedge u = .5) \vee (f(s) = 2 \wedge u = .5)$

  e.g. $p(s, S_0) = \mathcal{N}(f(s); 0, 1)$

  e.g. $\forall s(p(s, S_0) = \mathcal{U}(f(s); 0, 10)) \vee \forall s(p(s, S_0) = \mathcal{U}(f(s); 10, 20))$

MODELING A DOMAIN

- l axioms (domain dependent) + successor state axiom for p (fixed)
- p axioms in $\mathcal{D}_0$
  e.g. $p(s, S_0) = u \equiv (f(s) = 1 \wedge u = .5) \vee (f(s) = 2 \wedge u = .5)$
  e.g. $p(s, S_0) = \mathcal{N}(f(s); 0, 1)$
  e.g. $\forall s(p(s, S_0) = \mathcal{U}(f(s); 0, 10)) \vee \forall s(p(s, S_0) = \mathcal{U}(f(s); 10, 20))$

$$Bel(\phi, s) \doteq \sum_{\{s' : \phi[s']\}} p(s', s) \Big/ \sum_{s'} p(s', s)$$

subsumes Bayesian conditioning; but only well defined when s' is finite

Theorem:[4] $Bel(\phi, do(\vec{a}, S_0))$ can also be given by

$$\frac{1}{\gamma} \sum_{\vec{x}} \begin{cases} p(do(\vec{a}, \iota), do(\vec{a}, S_0)) & \text{if } \exists \iota.\ f_i(\iota) = x_i \wedge \phi[do(\vec{a}, \iota)] \\ 0 & \text{otherwise} \end{cases}$$

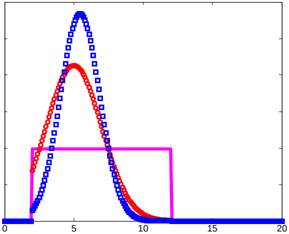
(include **axiom** that there is precisely one situation for every vector of fluent values)

[4] [B+L 13 (IJCAI)]

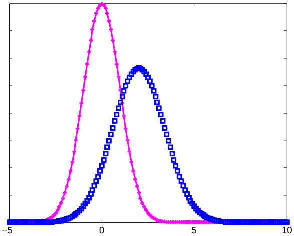
THE CONTINUOUS CASE

$$Bel(\phi, do(\vec{a}, S_0)) \doteq \frac{1}{\gamma} \int_{\vec{x}} \begin{cases} p(do(\vec{a}, \iota), do(\vec{a}, S_0)) & \text{if } \exists \iota.\, f_i(\iota) = x_i \wedge \phi[do(\vec{a}, \iota)] \\ 0 & \text{otherwise} \end{cases}$$

Degrees of beliefs after acting and sensing[5]



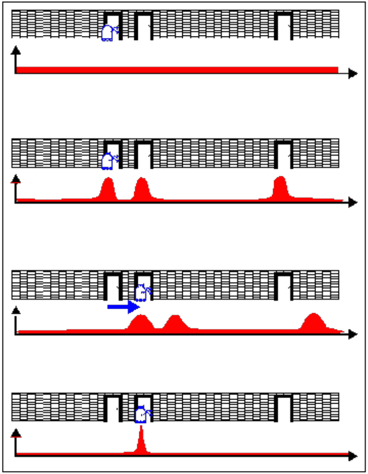
sharpening via repeated sensing



beliefs worsen after noisy move

[5] [B+L 14 (KR)]

Markov localization[6]



[6] [B+L 14 (AAMAS)]

FEATURES

- generalizes categorical knowledge change account, subsumes Bayesian conditioning
- incomplete/partial specifications (*e.g.* non-unique priors)
- expressive action specifications (*e.g.* continuous distribution to mixed)
- contextual likelihood axioms

$$l(sonar(z), s) = u \equiv humid(s) \wedge u = \mathcal{N}(z; distance(s), 4)$$
$$\neg humid(s) \wedge u = \mathcal{N}(z; distance(s); 1)$$

- hypothetical reasoning, introspection, *etc.*

ON PROJECTION

How can we solve projection?

$$\mathcal{D} \models Bel(\phi, do(\vec{a}, S_0))$$

By a generalized form of **regression**![7]

[7] [B+L 13 (UAI)]; also see [Kaelbling+Lozano-Pérez 13] for a similar result applied to robotic planning

ON PROJECTION

How can we solve projection?

$$\mathcal{D} \models Bel(\phi, do(\vec{a}, S_0))$$

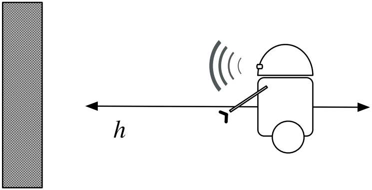
By a generalized form of **regression**![7]

Theorem: $\mathcal{D} \models Bel(\phi, do(\vec{a}, S_0))$ iff $\mathcal{D}_0 \models \mathcal{R}[Bel(\phi, do(\vec{a}, S_0))]$

(*e.g.* suppose $\mathcal{D}_0$ is a Bayesian network, reasoning about dynamics becomes straightforward)

[7] [B+L 13 (UAI)]; also see [Kaelbling+Lozano-Pérez 13] for a similar result applied to robotic planning

REGRESSION SETUP



Suppose $p(s, S_0) = \mathcal{U}(h(s); 2, 12)$ and

$$h(do(a, s)) = u \equiv \exists z.\ a = move(z) \land u = max(0, h(s) - z) \lor$$
$$\neg\exists z.\ a = move(z) \land u = h(s).$$

$$
\begin{aligned}
& Bel(h \geq 11, do(fwd(1), S_0)) \\
&= \frac{1}{\gamma} \int_{x \in \mathbb{R}} \mathcal{T}[\ P(x, \underline{h \geq 11}, do(fwd(1), S_0))\ ] && \text{(i)} \\
&= \frac{1}{\gamma} \int_{x \in \mathbb{R}} \mathcal{T}[P(x, \underline{\mathcal{R}[\psi]}, S_0)] && \text{(ii)} \\
&\qquad \text{where } \psi \text{ is } (h \geq 11)[do(fwd(1), now)] \\
&= \frac{1}{\gamma} \int_{x \in \mathbb{R}} \underline{\mathcal{T}[P(x, \max(0, h-1) \geq 11, S_0)]} && \text{(iii)} \\
&= \frac{1}{\gamma} \int_{x \in \mathbb{R}} P(x, \max(0, h-1) \geq 11, S_0) && \text{(iv)} \\
&= \frac{1}{\gamma} \int_{x \in \mathbb{R}} \begin{cases} p(\iota, S_0) & \text{if } \exists \iota.\ h(\iota) = x \wedge h(\iota) \geq 12 \\ 0 & \text{otherwise} \end{cases} && \text{(v)} \\
&= \frac{1}{\gamma} \int_{x \in \mathbb{R}} \begin{cases} .1 & \text{if } x \in [2, 12] \text{ and } x \geq 12 \\ 0 & \text{otherwise} \end{cases} && \text{(vi)} \\
&= \frac{1}{\gamma} \int_{x \in \mathbb{R}} \begin{cases} .1 & \text{if } x = 12 \\ 0 & \text{otherwise} \end{cases} && \text{(vii)} \\
&= 0
\end{aligned}
$$

Subsumes products of Gaussians, distribution transformations, *etc.*

PREGO

From that general specification language, we have implemented a **projection** system called PREGO[8]

- all families of successor state axioms, and contextual likelihood axioms
- limited $\mathcal{D}_0$ to a joint distribution over continuous random variables
- empirical behavior is very promising, *i.e.* interesting bridge between logic-based action languages and real-time needs of robotic applications

[8] [B+L 14 (AAAI)]

ON PROJECTION (REVISITED)

Regression is appropriate for planning and plan search, but over the course of millions of actions backward reasoning becomes infeasible.

[9] [Lin+Reiter 97, Vassos+Levesque 08]
[10] Strong model-theoretic guarantees

on projection (revisited)

Regression is appropriate for planning and plan search, but over the course of millions of actions backward reasoning becomes infeasible.

The progression of basic action theories[9]

$$\mathcal{D} \models \phi[do(\vec{a}, S_0)] \qquad \text{iff} \qquad Update(\mathcal{D}, \vec{a}) \models \phi[S_0]$$

Needs second-order logic, in general;[10] provides clean semantics for open/closed-world STRIPS, database updates, *etc.*

[9] [Lin+Reiter 97, Vassos+Levesque 08]
[10] Strong model-theoretic guarantees

How classical progression works: list **affected** atoms, then **forget** these atoms.

Example: Suppose $\mathcal{D}_0 = \{\neg Broken(g, S_0)\}$. Suppose agent does $drop(g)$. Then

- affected atom: $Broken(g, S_0)$
- instantiated successor state axiom: $Broken(x, do(a, s)) \equiv (a = drop(x) \vee Broken(x, s) \wedge a \neq repair(x))^{a,x,s}_{drop(g),g,S_0}$
- forget atom: ${\mathcal{D}_0}^{Broken(g,S_0)}_{true} \vee {\mathcal{D}_0}^{Broken(g,S_0)}_{false}$ $(\equiv true)$
- new theory is union of above two

Here: continuous likelihood axioms, degrees of belief

We introduce a new technique for progression.[11] *Example:*

- $f(do(a,s)) = u \equiv (\exists z.\ a = act(z) \wedge u = f(s) + z)$
  $\vee \neg\exists z(a = act(z)) \wedge u = f(s)$
- **invert** them:[12]
  $f(s) = u \equiv \exists z(a = act(z) \wedge u = f(do(a,s)) - z) \vee \neg\exists z(a = act(z) \wedge u = f(s))$.
- for p sentences in $\mathcal{D}_0$, inverting wrt a amounts to replacing every occurrence of $p(s, S_0)$ in $\mathcal{D}_0$ by $p(s, S_0)/\textsc{Likelihood}(a)$

[11] [B+L 14 (KR)]

[12] Only possible for invertible successor state axioms.

PROGRESSION IN CONTINUOUS DOMAINS (2)

Replace $\mathcal{D}$ by axiom inversions = new formulation of progression!

Generality: no noise and degrees of belief = classical definition

With noise = new general theory of belief propagation

e.g. Kalman filters special case

Space complexity results, *e.g.* efficiency of *context completeness*

CONCLUSION

A representation language for all of logic and all of probability, with general projection methodologies

- semantic and computational connections and bridge between the knowledge representation and probabilistic reasoning advances
- special purpose languages such as PREGO

In progress: a modal account, progression implementation, connections between progression and particle filters, *etc.*

FUTURE WORK

From action languages to programs, *e.g.* GOLOG[13]

$$\begin{array}{ll}\textbf{loop}: & \textbf{if}\ \neg Empty(queue) \\ & \quad \textbf{then}\ (\pi p)\, selectRequest(p); \\ & \qquad pickupCoffee;\ bringCoffee(p) \\ & \quad \textbf{else}\ wait\end{array}$$

constructs are **actions** from $\mathcal{D}$, used on virtual/ physical agents.

[13] [Levesque+ 97]

FUTURE WORK

From action languages to programs, *e.g.* GOLOG[13]

$$\begin{array}{ll}\textbf{loop}: & \textbf{if}\ \neg Empty(queue)\\ & \quad \textbf{then}\ (\pi p)\, selectRequest(p);\\ & \qquad pickupCoffee;\ bringCoffee(p)\\ & \quad \textbf{else}\ wait\end{array}$$

constructs are **actions** from $\mathcal{D}$, used on virtual/ physical agents.

A **general** version of GOLOG that admits noisy effectors and sensors and the robot's changing degrees of belief would lead to **more realistic high-level robot programs** . . . stay tuned!

Efficient implementations would bring robotic technologies and logical reasoners closer.

13 [Levesque+ 97]