Symbolic Methods for Probabilistic Inference, Optimization, and Decision-making

Scott Sanner

With much thanks to research collaborators: Zahra Zamani, Ehsan Abbasnejad, Karina Valdivia Delgado, Leliane Nunes de Barros, Luis Gustavo Rocha Vianna, Cheng (Simon) Fang
Graphical Models are Pervasive in AI

- **Medical**
  - Pathfinder: Expert System
  - BUGS: Epidemiology

- **Text**
  - LDA and 1,000 Extensions

- **Vision**
  - Ising Model!

- **Robotics**
  - Dynamics and sensor models
Graphical Models + Symbolic Methods

- Specify a graphical model for problem at hand
  - Text, vision, robotics, etc.

- Goal: efficient inference and optimization in this model
  - Be it discrete or continuous

- Symbolic methods (e.g., decision diagrams) facilitate this!
  - Useful as building block in any inference algorithm
  - Exploit structure for compactness, efficient computation
    - Automagically!

- Still partially a dream, but many recent advances

- Exploit more structure than graphical model alone
Tutorial Outline

• Part I: Symbolic Methods for Discrete Inference
  – Graphical Models and Influence Diagrams
  – Symbolic Inference with Decision Diagrams

• Part II: Extensions to Continuous Inference
  – Case Calculus
  – Extended ADD (XADD)

• Part III: Applications
  – Graphical Model Inference
  – Sequential Decision-making
  – Constrained Optimization
Part II: Extensions to Continuous Inference
General Form for Continuous Distributions?

• Probability density functions (pdfs), e.g.

\[ N(x; \mu, \sigma^2) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \]

• Could be **piecewise** or **deterministic**
  – Mixture models (gates)
  – Stochastic programs (conditionals)
  – Utilities (step), decision-making (max), preferences (\(\geq\))
  – Dynamical controlled systems (switching control)
  – Deterministic (\(\delta\))
General Piecewise Functions (Cases)

\[ z = f(x, y) = \begin{cases} 
(x > 3) \land (y \cdot x) : x + y \\
(x \cdot 3) \lor (y > x) : x^2 + xy^3 
\end{cases} \]
Formal Problem Statement

• General continuous graphical models represented by piecewise functions (cases)

$$f = \begin{cases} \phi_1 : f_1 \\ \vdots \\ \phi_k : f_k \end{cases}$$

• Continuous inference and optimization via the following piecewise calculus:
  • $f_1 \oplus f_2, f_1 \otimes f_2$
  • $\max(f_1, f_2), \min(f_1, f_2)$
  • $\int_x f(x), \int_x f(x)\delta(x - g(x))$
  • $\max_x f(x), \min_x f(x)$

Question: how do we perform these operations in closed-form?
Polynomial Case Operations: $\oplus$, $\otimes$

\[
\begin{align*}
\begin{cases}
\phi_1 : f_1 \
\phi_2 : f_2
\end{cases}
\oplus
\begin{cases}
\psi_1 : g_1 \
\psi_2 : g_2
\end{cases}
= ?
\end{align*}
\]
Polynomial Case Operations: $\oplus$, $\otimes$

\[
\begin{align*}
\phi_1 : f_1 & \oplus \psi_1 : g_1 \\
\phi_2 : f_2 & \oplus \psi_2 : g_2 \\
\end{align*}
\]

\[
\begin{align*}
\phi_1 \wedge \psi_1 & : f_1 + g_1 \\
\phi_1 \wedge \psi_2 & : f_1 + g_2 \\
\phi_2 \wedge \psi_1 & : f_2 + g_1 \\
\phi_2 \wedge \psi_2 & : f_2 + g_2 \\
\end{align*}
\]

- Similarly for $\otimes$
  - Polynomials closed under $+$, $*$

- What about max?
  - Max of polynomials is not a polynomial 😞
Polynomial Case Operations: max

\[
\max \left( \left\{ \begin{array}{c}
\phi_1 : f_1 \\
\phi_2 : f_2
\end{array} , \quad \begin{array}{c}
\psi_1 : g_1 \\
\psi_2 : g_2
\end{array} \right\} \right) = \ ?
\]
Polynomial Case Operations: max

\[
\max \left( \begin{cases} \phi_1 : f_1, \\ \phi_2 : f_2 \\ \psi_1 : g_1 \\ \psi_2 : g_2 \end{cases} \right) = \begin{cases} \phi_1 \land \psi_1 \land f_1 > g_1 : f_1 \\ \phi_1 \land \psi_1 \land f_1 \cdot g_1 : g_1 \\ \phi_1 \land \psi_2 \land f_1 > g_2 : f_1 \\ \phi_1 \land \psi_2 \land f_1 \cdot g_2 : g_2 \\ \phi_2 \land \psi_1 \land f_2 > g_1 : f_2 \\ \phi_2 \land \psi_1 \land f_2 \cdot g_1 : g_1 \\ \phi_2 \land \psi_2 \land f_2 > g_2 : f_2 \\ \phi_2 \land \psi_2 \land f_2 \cdot g_2 : g_2 \end{cases}
\]

• Still a piecewise polynomial!

Size blowup? We’ll get to that…
Integration: \( \int_x \)

- \( \int_x \) closed for polynomials
  - But how to compute for case?

\[
\int_x \left\{ \begin{array}{c}
\phi_1 : f_1 \\
\vdots \\
\phi_k : f_k \\
\end{array} \right\} \, dx
\]

- Just integrate case partitions, \( \oplus \) results!
1. Determine integration bounds

\[ \int_{x}^{\phi_1} f_1 \, dx \]

\[ \phi_1 := [x > -1] \land [x > y - 1] \land [x \cdot z] \land [x \cdot y + 1] \land [y > 0] \]

\[ f_1 := x^2 - xy \]

What constraints here?
- independent of \( x \)
- pairwise UB > LB

UB and LB are symbolic!

How to evaluate?
Definite Integral Evaluation

• How to evaluate integral bounds?

\[
\int_{x=LB}^{UB} x^2 - xy = \left. \frac{1}{3} x^3 - \frac{1}{2} x^2 y \right|_{LB}^{UB}
\]

• Can do polynomial operations on cases!
Max-out Case Operation

• Like $\int_x \text{case}(x)$, reduce to single partition $\max$

  – In a *single* case partition …$\max$ w.r.t. critical points

  • LB, UB
  • Derivative is zero ($\text{Der}0$)

  • $\max(\text{case}(x/\text{LB}), \text{case}(x/\text{UB}), \text{case}(x/\text{Der}0))$

  – Can even track substitutions through $\max$ to recover function of maximizing assignments!

See UAI 2011, AAAI 2012 papers for more details
Integration with a $\delta$: substitution

- Special case for integrals with $\delta$-functions
  \[
  \int_{x} \delta[x - y] f(x) \, dx = f(y) \text{ triggers symbolic substitution}
  \]

- More generally:
  \[
  \int_{x_j'} \delta[x_j' - g(\bar{x})] V' \, dx_j' = V'\{x_j'/g(\bar{x})\}
  \]

- E.g.,
  \[
  \int_{x_1'} \delta[x_1' - (x_1^2 + 1)] \left( \begin{array}{l}
  \frac{x_1'}{x_1'} < 2 : \quad \frac{x_1'}{x_1'} \\
  \frac{x_1'}{x_1'} \geq 2 : \quad \frac{x_1'}{x_1'}
  \end{array} \right) \, dx_1' = \left\{ \begin{array}{l}
  \frac{x_1^2 + 1}{x_1^2 + 1} < 2 : \quad \frac{x_1^2 + 1}{x_1^2 + 1} \\
  \frac{x_1^2 + 1}{x_1^2 + 1} \geq 2 : \quad (\frac{x_1^2 + 1}{x_1^2 + 1})^2
  \end{array} \right.
  \]

- If $g$ is case: need conditional substitution
  - see Sanner, Delgado, Barros (UAI 2011)
Recap

• Continuous inference and optimization problems represented by piecewise functions (cases)

\[ f = \begin{cases} 
\phi_1 : & f_1 \\
\vdots & \vdots \\
\phi_k : & f_k 
\end{cases} \]

• Continuous inference and optimization via the following piecewise calculus:
  • \( f_1 \oplus f_2, f_1 \otimes f_2 \)
  • \( \max( f_1, f_2 ), \min( f_1, f_2 ) \)
  • \( \int_x f(x) \delta(x - g(x)) \)
  • \( \int_x f(x) \)
  • \( \max_x f(x), \min_x f(x) \)

Closed-form for general case
Closed-form for linear piecewise polynomial, others
Closed-form for linear piecewise quadratic
Cool… can we proceed to continuous inference?

Case partitions blow-up exponentially in number of operations.

Need to a compact form.
Case → XADD

XADD = continuous variable extension of algebraic decision diagram

- compact, minimal case representation
- case operations can exploit structure
Case → XADD

\[ V = \begin{cases} 
  x_1 + k > 100 \land x_2 + k > 100 : 0 \\
  x_1 + k > 100 \land x_2 + k \cdot 100 : x_2 \\
  x_1 + k \cdot 100 \land x_2 + k > 100 : x_1 \\
  x_1 + x_2 + k > 100 \land x_1 + k \cdot 100 \land x_2 + k \cdot 100 \land x_2 > x_1 : x_2 \\
  x_1 + x_2 + k > 100 \land x_1 + k \cdot 100 \land x_2 + k \cdot 100 \land x_2 \cdot x_1 : x_1 \\
  x_1 + x_2 + k \cdot 100 : x_1 + x_2 \\
  \vdots 
\end{cases} \]

*Minimality and canonicity largely irrelevant because they are NP-Hard... but not required for correctness.
Compactness of (X)ADDs

- XADD linear in number of decisions $\phi_i$

- **Case version** has exponential number of partitions!
XADD Maximization

$$\max( \text{\(y > 0\)}, \text{\(x > 0\)} ) =$$

May introduce *new* decision tests

Operations exploit structure: $O(|f||g|)$
Maintaining XADD Orderings

- Max may get decisions out of order

\[
\text{max}(\ x > 0\ , \ y > 0\ ) = x > y
\]

Decision ordering (root → leaf)

- \( x > y \)
- \( y > 0 \)
- \( x > 0 \)

Newly introduced node is out of order!
Maintaining XADD Orderings

- Substitution may get decisions out of order

Decision ordering (root→leaf):
- \( x > y \)
- \( y > 0 \)
- \( x > z \)

\[\sigma = \{ z/y \}\]

Substituted nodes are now out of order!
Correcting XADD Ordering

• Obtain *ordered* XADD from *unordered* XADD
  – key idea: binary operations maintain orderings

  \[ \text{z is out of order} \]

  \[ \begin{align*}
  \text{result will have z in order!}
  \end{align*} \]

  \[ \begin{array}{c}
  \text{Inductively assume ID}_1 \text{ and ID}_0 \text{ are ordered.}
  \\
  \text{All operands ordered, so applying } \times, \oplus \text{ produces ordered result!}
  \end{array} \]
Maintaining Minimality

Node unreachable – $x + y < 0$ always false if $x > 0$ & $y > 0$

If **linear**, can detect with feasibility checker of LP solver & prune

More subtle prunings as well.

$x > 0$ → $y > 0$ → $x + y < 0$ always false if $x > 0$ & $y > 0$
XADD Approximation

• Can we extend APRICODD-style approximations to XADDs?

• Yes, but not as simple as averaging leaves…
Linear XADD Leaf Merging

\[
\min_{\bar{c}^*} \max_{i \in \{1, 2\}} \max_{\bar{x} \in S_{\phi_i}} \left| \tilde{c}_i^T \begin{bmatrix} \bar{x} \\ 1 \end{bmatrix} - \bar{c}^* \begin{bmatrix} \bar{x} \\ 1 \end{bmatrix} \right|, \quad f_i \quad \text{or} \quad f^*
\]
Linear XADD Leaf Merging

Constraint generation: for $c^*$, use \textbf{LP}
to generate max violated constraint

$$\min_{\overrightarrow{c}^*, \epsilon} \epsilon$$

s.t. $\epsilon \geq \left| \overrightarrow{c}_i^T \left[ \begin{array}{c} \tilde{x}_{ij}^k \\ 1 \end{array} \right] - \overrightarrow{c}^*_i^T \left[ \begin{array}{c} \tilde{x}_{ij}^k \\ 1 \end{array} \right] \right|$, \quad \forall i \in \{1, 2\}, \forall \theta_{ij}, \forall k \in \{1 \ldots N_{ij}\}$
Linear Approximation Example

(a) Value at 6th iteration for exact SDP.

(b) Value at 6th iteration for 5% approximate SDP.
Nonlinear XADD Approximation?

• 1D Example

• Questions
  – What approximating class?
  – What error function?
    • Max not feasible
    • Volume of squared error? Integral is exact.

But many caveats vs. linear case
XADD Recap

• Represent case by XADD

\[ f = \begin{cases} 
\phi_1 : f_1 \\
\vdots \\
\phi_k : f_k 
\end{cases} \]

• Piecewise calculus operations exploit XADD structure:
  • \( f_1 \oplus f_2, f_1 \otimes f_2 \)
  • \( \max(f_1, f_2), \min(f_1, f_2) \)
  • \( \int x f(x) \delta(x - g(x)) \)
  • \( \int x f(x) \)
  • \( \max_x f(x), \min_x f(x) \)

XADD more compact than direct case representation

Bounded Linear XADD approximation possible.

Working on nonlinear case.
Part III: Applications

Graphical Models
Discrete & Continuous HMMs

No one previously did this inference exactly in closed-form!
Exact Inference in Cont. Graphical Models

- Fully Gaussian ✔
  - Most inference

- Fully Uniform ✔
  - 1D, n-D hyperrectangular cases
  - General Uniform

- Piecewise, Asymmetrical, Multimodal
  - Exact (conditional) inference possible in closed-form?

Yes, but not a solution you can write on 1 sheet of paper
What has everyone been missing?

Compact representations and closed-form operations on piecewise functions
Exact Graphical Model Inference!  
(directed and undirected)

- Represent all factors as piecewise polynomials

\[ p(x_2|x_1) = \frac{\int x_3 \cdots \int x_n \bigotimes_{i=1}^{k} case_i \ dx_n \cdots dx_3}{\int x_2 \cdots \int x_n \bigotimes_{i=1}^{k} case_i \ dx_n \cdots dx_2} \]

- Or an exact expectation of *any* polynomial

\[ \mathbb{E}_{x \sim p(x|o)}[poly(x)|o] = \int x \ p(x|o)poly(x) \ dx \]

- *poly*: mean, variance, skew, curtosis, ..., \( x^2+y^2+xy \)

All computed exactly in closed-form by **Symbolic Variable Elimination (SVE)**
Voila: Closed-form Exact Inference via SVE!

Arbitrary approximations of any pdf with bounded support in practice.
Computational Complexity?

- In theory for SVE on graphical models
  - Best-case complexity $\Omega(#\text{operations})$
  - Worst-case complexity is $O(\exp(#\text{operations}))$
    - Not explicitly tree-width dependent!
    - But worse: integral may invoke 100’s of operations!

Fortunately data structures mitigate worst-case complexity
An Expressive Conjugate Prior for Bayesian Inference

• General Bayesian Inference

\[ p(\theta|D_{n+1}) \propto p(d_{n+1}|\theta)p(\theta|D_n) \]

• Prior & likelihood for computational convenience?
  – No, choose as appropriate for your problem!
Bayesian Robotics

- $D$: true distance to wall
- $x_1, \ldots, x_n$: measurements
- want: $E[D \mid x_1, \ldots, x_n]$
Bayesian Robotics: Results

- Sample position posteriors and expectations...

\[
\mathbb{E}[D|x_1 = 8, d_2 = 5] = 6.0
\]

\[
\mathbb{E}[D|x_1 = 5, d_2 = 3, d_3 = 8] = 4.39
\]

\[
\mathbb{E}[D|x_2 = 1, d_2 = 3, d_3 = 4, d_4 = 8] = 5.45
\]

\[
\mathbb{E}[D|x_1 = 5, d_2 = 4, d_3 = 6, d_4 = 5] = 4.89
\]
Bayesian Robotics: Results

• Example posterior given measurements \{3,5,8\}:

Do we really want to make Gaussian approximation?

What if a stairwell here?
Part III: Applications

Symbolic Dynamic Programming
Continuous State MDPs

- 2-D Navigation
- State: \((x,y) \in \mathbb{R}^2\)
- Actions:
  - move-x-2
    - \(x' = x + 2\)
    - \(y' = y\)
  - move-y-2
    - \(x' = x\)
    - \(y' = y + 2\)
- Reward:
  - \(R(x,y) = \mathbb{1}[\ (x > 5) \land (x < 10) \land (y > 2) \land (y < 5) \ ]\)

Feng et al (UAI-04) Assumptions:
1. Continuous transitions are deterministic and linear
2. Discrete transitions can be stochastic
3. Reward is piecewise rectilinear convex
Exact Solutions to DC-MDPs: Regression

• Continuous regression is just translation of “pieces”
Exact Solutions to DC-MDPs: Maximization

- Q-value maximization yields piecewise rectilinear solution

\[
\max_a Q(a,x,y) = \begin{cases} 
1 & \text{if } a \text{ is optimal} \\
0 & \text{otherwise}
\end{cases}
\]
Previous Work Limitations I

- Exact regression when transitions nonlinear?

Action **move-nonlin**:  
- \( x' = x^3y + y^2 \)  
- \( y' = y \times \log(x^2y) \)

How to compute boundary in closed-form?
Previous Work Limitations II

• $\max(.,.)$ when reward/value arbitrary piecewise?

Closed-form representation for $\max$?
Continuous State MDPs

- Value Iteration for $h \in 0..H$
  
  - Regression step:
    $$Q_{a}^{h+1}(\tilde{b}, \tilde{x}) = R_{a}(\tilde{b}, \tilde{x}) + \gamma \cdot$$
    $$\sum_{\tilde{b}'} \int_{\tilde{x}'} \left( \prod_{i=1}^{n} P(b'_i|\tilde{b}, \tilde{x}, a) \prod_{j=1}^{m} P(x'_j|\tilde{b}, \tilde{b}', \tilde{x}, a) \right) V^h(\tilde{b}', \tilde{x}') d\tilde{x}'$$
  
  - Maximization step:
    $$V_{h+1} = \max_{a \in A} Q_{a}^{h+1}(\tilde{b}, \tilde{x})$$
Continuous Actions?

If we can solve this, can solve **multivariate inventory control** – closed-form policy unknown for 50+ years!
Continuous Actions

• Inventory control
  – Reorder based on stock, future demand
  – Action: \( a(\Delta); \Delta \in \mathbb{R}^{|a|} \)

• Need \( \max_\Delta \) in Bellman backup

\[
V_{h+1} = \max_{a \in A} \max_{\Delta} Q_{a}^{h+1}(\Delta)
\]

• \( \max_x \text{case}(x) \) previously defined \text{case}(x)
  – Can track maximizing \( \Delta \) substitutions to recover \( \pi \)
Sequential Control Summary

- Continuous state, action, observation (PO)MDPs
  - Discrete action MDPs  (UAI-11)
  - Continuous action MDPs (incl. exact policy)  (AAAI-12b)
  - Continuous observation POMDPs  (NIPS-12)
  - Robust solutions with continuous noise  (IJCAI-13)

All papers co-authored with Sanner
Part III: Applications

Optimization
\[
\max_x \text{ case}(x) = \text{Constrained Optimization!}
\]

- **Conditional constraints**
  - E.g., if \((x > y)\) then \((y < z)\)
  - Not 0-1 MILP, MIQP equivalent

- **Factored / sparse constraints**
  - Constraints may be sparse!
    \[x_1 > x_2, \quad x_2 > x_3, \quad \ldots, \quad x_{n-1} > x_n\]
  - Dynamic programming for continuous optimization!

- **Parameterized optimization**
  - \(f(y) = \max_x f(x,y)\)
  - Maximum value, substitution as a **function of** \(y\)
Open Problems
Continuous Actions, Nonlinear

• **Robotics**
  – Continuous position, joint angles
  – Represent exactly with polynomials
    • Radius constraints

• **Obstacle Navigation**
  – 2D, 3D, 4D (time)
  – Don’t discretize!
    • Grid worlds
  – But nonlinear 😞

**Multilinear, quadratic extensions.**
In general: algebraic geometry.

Solve in 2 steps!
Open Problems

- Bounded approximation for nonlinear XADDs

This XADD has > 1000 nodes!
Recap

- Defined a calculus for piecewise functions
  - $f_1 \oplus f_2$, $f_1 \otimes f_2$
  - $\max(f_1, f_2)$, $\min(f_1, f_2)$
  - $\int_x f(x)$
  - $\max_x f(x)$, $\min_x f(x)$

- Defined XADD to efficiently compute with cases

- Makes possible
  - Exact inference in continuous graphical models
  - Unprecedented expressive sequential optimization and control
  - New approaches for optimization
Symbolic Piecewise Calculus + XADD

= Expressive Continuous Inference & Optimization

Thank you!
Questions?