Uniform Interpolantion in Multi-Agent Modal Logics

Liangda Fang, Yongmei Liu and Hans van Ditmarsch

Dept. of Computer Science
Jinan University

May 21, 2016

Accepted at the IJCAI-2016

Introduction

- $\mathcal{P}$: a finite set of propositions;
- $p \in \mathcal{P}$;
- φ: the original formula;
- ψ: a uniform interpolant of φ on $\bar{p}$ ($\mathcal{P} \setminus p$) satisfies the following properties:
  - ψ uses $var(\varphi)$ without p;
  - $\varphi \models \psi$;
  - For any query η which does not contain p, $\varphi \models \eta$ iff $\psi \models \eta$.

How to compute ψ?

A brute-force approach to propositional logic

1. Transform φ into an equivalent principal DNF ψ (a disjunction of full terms using propositions occurred in φ);
2. Obtain ψ^p by eliminating any occurrence of p or $\neg p$, or replacing any occurrence of p or $\neg p$ with $\top$ in ψ.

A brute-force approach to propositional logic

1. Transform φ into an equivalent principal DNF ψ (a disjunction of full terms using propositions occurred in φ);
2. Obtain ψ^p by eliminating any occurrence of p or $\neg p$, or replacing any occurrence of p or $\neg p$ with $\top$ in ψ.

Example

- $\varphi = p \wedge q \vee \neg p \wedge \neg r$;
- $\psi = (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge \neg r)$;
- $\psi^p = (q \wedge r) \vee (q \wedge \neg r) \vee (\neg q \wedge \neg r)$.

A brute-force approach to propositional logic

1. Transform φ into an equivalent principal DNF ψ (a disjunction of full terms using propositions occurred in φ);
2. Obtain ψ^p by eliminating any occurrence of p or $\neg p$, or replacing any occurrence of p or $\neg p$ with $\top$ in ψ.

Example

- $\varphi = p \wedge q \vee \neg p \wedge \neg r$;
- $\psi = (p \wedge q \wedge r) \vee (p \wedge q \wedge \neg r) \vee (\neg p \wedge q \wedge \neg r) \vee (\neg p \wedge \neg q \wedge \neg r)$;
- $\psi^p = (q \wedge r) \vee (q \wedge \neg r) \vee (\neg q \wedge \neg r)$.

How about multi-agent modal logics?

- Modal logic: propositional logic $+$ $\mathbf{K}_i$ operators;
- $\mathbf{K}_i\varphi$: agent i knows φ.

Outline

Outline

Modal language: $\mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$

Definition (Syntax of $\mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$)

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{K}_i\varphi \mid \mathbf{C}\varphi,$$

where $p \in \mathcal{P}$, $i \in \mathcal{A}$, and $\varphi \in \mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$.

Modal language: $\mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$

Definition (Syntax of $\mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$)

$$\varphi ::= p \mid \neg\varphi \mid \varphi \wedge \varphi \mid \mathbf{K}_i\varphi \mid \mathbf{C}\varphi,$$

where $p \in \mathcal{P}$, $i \in \mathcal{A}$, and $\varphi \in \mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$.

Definition (Two sublanguages)

1. $\mathcal{L}_{n}^{\mathbf{K}}$: $\mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$ without $\mathbf{C}$ operator;
2. $\mathcal{L}_{\mathbf{PC}}^{\mathbf{K}}$: $\mathcal{L}_{\mathbf{C}}^{\mathbf{K}}$ with propositional common knowledge, *i.e.*, any φ appearing in $\mathbf{C}\varphi$ must be propositional.

Kripke models

Definition (Kripke models)

A Kripke model is a tuple $\langle S, R, V \rangle$ where

- S: a non-empty set of states,
- R: for each $i \in \mathcal{A}$, $R_i \subseteq S \times S$ is a relation on states.
- V: $S \to 2^{\mathcal{P}}$ is a function assigning to each proposition in a subset of states.

A pair (M, s) is called a pointed model.

Kripke models

Definition (Kripke models)

A Kripke model is a tuple $\langle S, R, V \rangle$ where

- S: a non-empty set of states,
- R: for each $i \in \mathcal{A}$, $R_i \subseteq S \times S$ is a relation on states.
- V: $S \rightarrow 2^{\mathcal{P}}$ is a function assigning to each proposition in a subset of states.

A pair (M, s) is called a pointed model.

Logic systems	Restrictions on R
K	no restrictions
D	Seriality
T	Reflexivity
K45	Transitivity + Euclidean
KD45	Seriality + Transitivity + Euclidean
S5	Reflexivity + Euclidean

Cover modalites

Definition (Cover modalities)

1. $\nabla_i\Phi = \mathbf{K}_i(\bigvee_{\varphi\in\Phi}\varphi) \wedge \bigwedge_{\varphi\in\Phi}\hat{\mathbf{K}}_i\varphi$;
2. $\nabla\Phi = \mathbf{C}(\bigvee_{\varphi\in\Phi}\varphi) \wedge \bigwedge_{\varphi\in\Phi}\hat{\mathbf{C}}\varphi$.

We can use ∇_i (resp. ∇) modality instead of $\mathbf{K}_i$ (resp. $\mathbf{C}$) modality.

Canonical formulas

Definition (Canonical formulas)

Let $P \subseteq \mathcal{P}$ be finite. We inductively define the set E_k^P as follows:

- $E_0^P = \{\bigwedge_{p \in \mathcal{X}} p \wedge \bigwedge_{p \in P \setminus \mathcal{X}} \neg p \mid \mathcal{X} \subseteq P\}$;
- $E_{k+1}^P = \{\delta_0 \wedge \bigwedge_{i \in \mathcal{A}} \nabla_i \Phi_i \mid \delta_0 \in E_0^P$ and $\Phi_i \subseteq E_k^P\}$.

$\delta_k \in E_k$: completely characterizes a Kripke model up to depth k
a full terms in modal logics.

Canonical formulas

Definition (Canonical formulas)

Let $P \subseteq \mathcal{P}$ be finite. We inductively define the set E_k^P as follows:

- $E_0^P = \{\bigwedge_{p \in \mathcal{X}} p \wedge \bigwedge_{p \in P \setminus \mathcal{X}} \neg p \mid \mathcal{X} \subseteq P\}$;
- $E_{k+1}^P = \{\delta_0 \wedge \bigwedge_{i \in \mathcal{A}} \nabla_i \Phi_i \mid \delta_0 \in E_0^P$ and $\Phi_i \subseteq E_k^P\}$.

$\delta_k \in E_k$: completely characterizes a Kripke model up to depth k
a full terms in modal logics.

Proposition

Any formula in $\mathcal{L}_n^{\mathbf{K}}$ can be equivalently transformed into a disjunction of satisfiable canonical formulas.

Canonical formulas

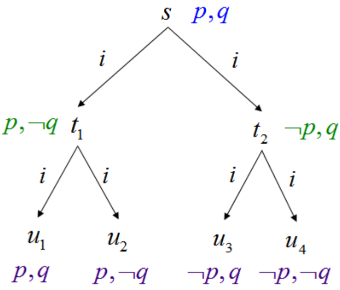

Figure: Kripke model

1. $p \wedge q$;
2. $p \wedge q \wedge \nabla_i \{p \wedge \neg q, \neg p \wedge q\}$;
3. $p \wedge q \wedge$
   $\nabla_i \{p \wedge \neg q \wedge \nabla_i \{p \wedge q, p \wedge \neg q\} \wedge$
   $\neg p \wedge q \wedge \nabla_i \{\neg p \wedge q, \neg p \wedge \neg q\}\}$

Outline

Computation of uniform interpolants

1. Transform φ to a disjunction of satisfiable canonical formulas $\bigvee_{\delta\in\Phi}\delta$;
2. Obtain δ^p by eliminating any occurrence of p or $\neg p$, or replacing any occurrence of p or $\neg p$ with $\top$ in δ;
3. $\bigvee_{\delta\in\Phi}\delta^p$ is a uniform interpolation of φ on $\bar{p}$.

Computation of uniform interpolants

1. Transform φ to a disjunction of satisfiable canonical formulas $\bigvee_{\delta \in \Phi} \delta$;
2. Obtain δ^p by eliminating any occurrence of p or $\neg p$, or replacing any occurrence of p or $\neg p$ with $\top$ in δ;
3. $\bigvee_{\delta \in \Phi} \delta^p$ is a uniform interpolation of φ on $\bar{p}$.

Example

- $\varphi = \hat{\mathbf{K}}_i p \wedge \hat{\mathbf{K}}_i \neg p$;
- $\varphi \equiv \delta_1 \vee \delta_2$;
- $\delta_1 = p \wedge \nabla_i \{p, \neg p\}$;
- $\delta_2 = \neg p \wedge \nabla_i \{p, \neg p\}$;
- $\delta_1^p \vee \delta_2^p \equiv \top \wedge \nabla_i \{\top\} \equiv \hat{\mathbf{K}}_i \top$.

Main theorem

Theorem

Let L *be* $\mathrm{K_n}$, $\mathrm{D_n}$, $\mathrm{T_n}$, $\mathrm{K45_n}$, $\mathrm{KD45_n}$ *or* $\mathrm{S5_n}$.
Let δ *be a pc-canonical formula satisfiable in* L.
Then, δ^p *is a uniform interpolant of* δ *on* $\bar{p}$.

Main theorem

Theorem

Let L *be* $\mathsf{K_n}$, $\mathsf{D_n}$, $\mathsf{T_n}$, $\mathsf{K45_n}$, $\mathsf{KD45_n}$ *or* $\mathsf{S5_n}$.
Let δ *be a pc-canonical formula satisfiable in* L.
Then, δ^p *is a uniform interpolant of* δ *on* $\bar{p}$.

Definition (Uniform interpolation)

A logic L has uniform interpolation: for any L-formula φ and any proposition p, there exists an L-formula ψ which is a uniform interpolant of φ on $\bar{p}$.

Main theorem

Theorem

Let L *be* $\mathrm{K_n}$, $\mathrm{D_n}$, $\mathrm{T_n}$, $\mathrm{K45_n}$, $\mathrm{KD45_n}$ *or* $\mathrm{S5_n}$.
Let δ *be a pc-canonical formula satisfiable in* L.
Then, δ^p *is a uniform interpolant of* δ *on* $\bar{p}$.

Definition (Uniform interpolation)

A logic L has uniform interpolation: for any L-formula φ and any proposition p, there exists an L-formula ψ which is a uniform interpolant of φ on $\bar{p}$.

Corollary

$\mathrm{K_n}$, $\mathrm{D_n}$, $\mathrm{T_n}$, $\mathrm{K45_n}$, $\mathrm{KD45_n}$ *and* $\mathrm{S5_n}$ *have uniform interpolation.*

Common knowledge case

- Negative result: KC does not have uniform interpolation. [Studer, 2009]
- We consider the propositional common knowledge case, *i.e.*, $\mathcal{L}^{\mathbf{K}}_{\mathbf{PC}}$ where any φ appearing in $\mathbf{C}\varphi$ must be propositional.

Pc-canonical formulas

Definition (Pc-canonical formulas)

Let $P \subseteq \mathcal{P}$ be finite. We inductively define the set C_k^P as follows:

- $C_0^P = \{\theta \wedge \nabla\Phi_{\mathcal{A}} \mid \theta \in E_0^P$ and $\Phi_{\mathcal{A}} \subseteq E_0^P\}$;
- $C_{k+1}^P = \{\theta \wedge (\bigwedge_{i\in\mathcal{A}} \nabla_i\Phi_i) \wedge \nabla\Phi_{\mathcal{A}} \mid \theta \in E_0^P, \Phi_i \subseteq C_k^P$ and $\Phi_{\mathcal{A}} \subseteq E_0^P\}$.

Pc-canonical formulas

Definition (Pc-canonical formulas)

Let $P \subseteq \mathcal{P}$ be finite. We inductively define the set C_k^P as follows:

- $C_0^P = \{\theta \wedge \nabla\Phi_{\mathcal{A}} \mid \theta \in E_0^P \text{ and } \Phi_{\mathcal{A}} \subseteq E_0^P\}$;
- $C_{k+1}^P = \{\theta \wedge (\bigwedge_{i \in \mathcal{A}} \nabla_i \Phi_i) \wedge \nabla\Phi_{\mathcal{A}} \mid \theta \in E_0^P,\ \Phi_i \subseteq C_k^P \text{ and } \Phi_{\mathcal{A}} \subseteq E_0^P\}$.

Proposition

Any formula in $\mathcal{L}_{\mathbf{PC}}^{\mathbf{K}}$ can be equivalently transformed into a disjunction of satisfiable pc-canonical formulas.

Main theorem 2

Theorem

Let L *be* KC, DC, TC, K45C, KD45C *or* S5C.
Let δ *be a pc-canonical formula satisfiable in* L.
Then, δ^p *is a uniform interpolant of* δ *on* $\bar{p}$.

Corollary

KPC, DPC, TPC, K45PC, KD45PC *and* S5PC *have uniform interpolation.*

Conclusions

- Prove that K_n, D_n, T_n, $K45_n$, $KD45_n$ and $S5_n$ have uniform interpolation.
- Extend the above results to propositional common knowledge case.

Future work

- A practical approach for computing uniform interpolant;
- More general cases of common knowledge;
- Distributed knowledge;
- Progression and diagnose in multi-agent settings.

Thank you!