Effectively Propositional Interpolants

Samuel Drews and Aws Albarghouthi

UNIVERSITY OF WISCONSIN–MADISON

Effectively Propositional Logic (EPR)

$$\exists x_1 \ldots x_n \forall y_1 \ldots y_m \; \varphi$$

Quantifier-free

No function symbols

EPR

Decidable satisfiability!

EPR

Decidable satisfiability!

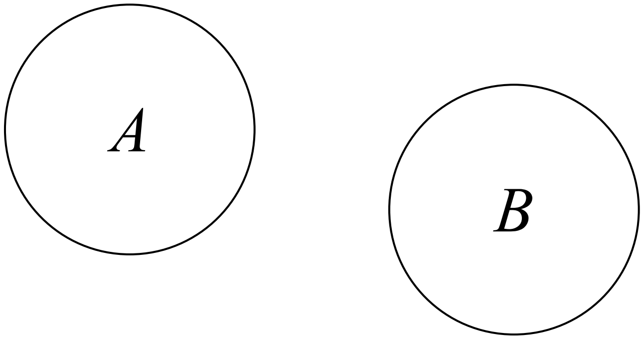
Expressive:

- Linked lists [Itzhaky et al. 2014]
- Software-defined networks [Ball et al. 2014]
- Parameterized distributed protocols [Padon et al. 2016]
- …

Interpolants

Given A and B such that

$$A \wedge B \text{ is unsatisfiable}$$



Interpolants

Given A and B such that

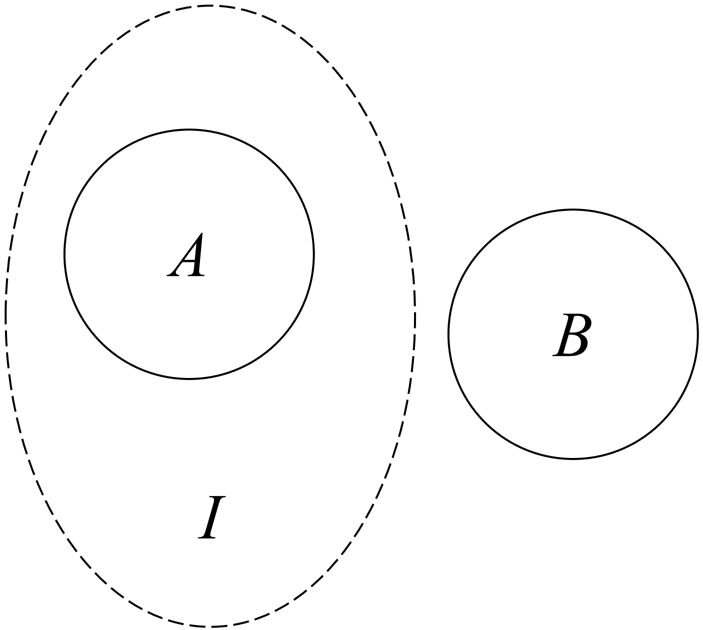
$A \wedge B$ is unsatisfiable

Find I such that

$A \rightarrow I$ is valid

$I \wedge B$ is unsatisfiable

I is in shared vocabulary (A, B)



Restricted Logics for Invariants

$$I(\vec{x}) \land T(\vec{x}, \vec{x}') \to I(\vec{x}')$$

is valid, or

$$I(\vec{x}) \land T(\vec{x}, \vec{x}') \land \neg I(\vec{x}')$$

is unsat

Restricted Logics for Invariants

$$I(\vec{x}) \land T(\vec{x}, \vec{x}') \rightarrow I(\vec{x}')$$

is valid, or

$$I(\vec{x}) \land T(\vec{x}, \vec{x}') \land \neg I(\vec{x}')$$

is unsat

Restricted Logics for Invariants

$$I(\vec{x}) \land T(\vec{x}, \vec{x}') \to I(\vec{x}')$$

is valid, or

$$I(\vec{x}) \land T(\vec{x}, \vec{x}') \land \neg I(\vec{x}')$$

is unsat

$\exists^* \forall^* \varphi$ decidable, but

$\forall^* \exists^* \varphi$ undecidable

Restricted Logics for Invariants

$$I(\vec{x}) \wedge T(\vec{x}, \vec{x}') \rightarrow I(\vec{x}')$$

is valid, or

$$I(\vec{x}) \wedge T(\vec{x}, \vec{x}') \wedge \neg I(\vec{x}')$$

is unsat

$\exists^* \forall^* \varphi$ decidable, but

$\forall^* \exists^* \varphi$ undecidable

Bummer

Restricted Logics for Invariants

$I(\vec{x}) \wedge T(\vec{x}, \vec{x}') \rightarrow I(\vec{x}')$

is valid, or

$I(\vec{x}) \wedge T(\vec{x}, \vec{x}') \wedge \neg I(\vec{x}')$

is unsat

$\exists^* \forall^* \varphi$ decidable, but

$\forall^* \exists^* \varphi$ undecidable

Bummer

1. ∃-logic: $\exists^* \varphi$
2. ∀-logic: $\forall^* \varphi$
3. AF-logic:

   boolean combinations of ∃-logic and ∀-logic

   ex: $(\exists^* \varphi_1 \wedge \forall^* \varphi_2) \vee \forall^* \varphi_3$

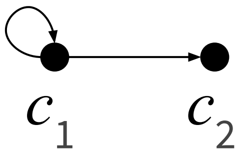
Models and Diagrams

$$\varphi = \exists a \forall b.\, p(a, b)$$

Models and Diagrams

$$\varphi = \exists a \forall b.\, p(a, b)$$

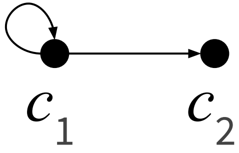
Model $m \models \varphi$



Models and Diagrams

$$\varphi = \exists a \forall b.\, p(a, b)$$

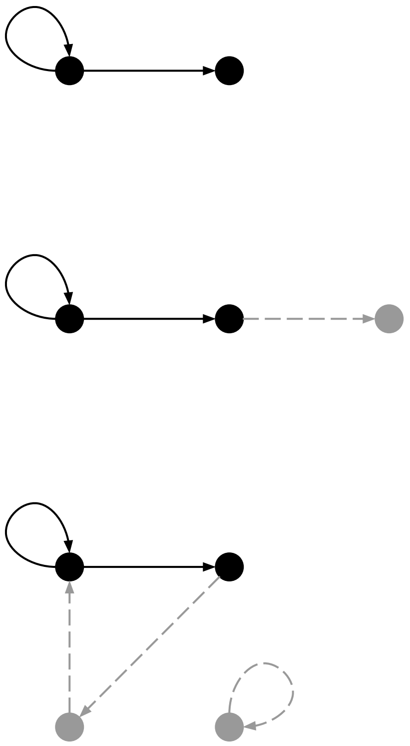
Model $m \models \varphi$



Diagram

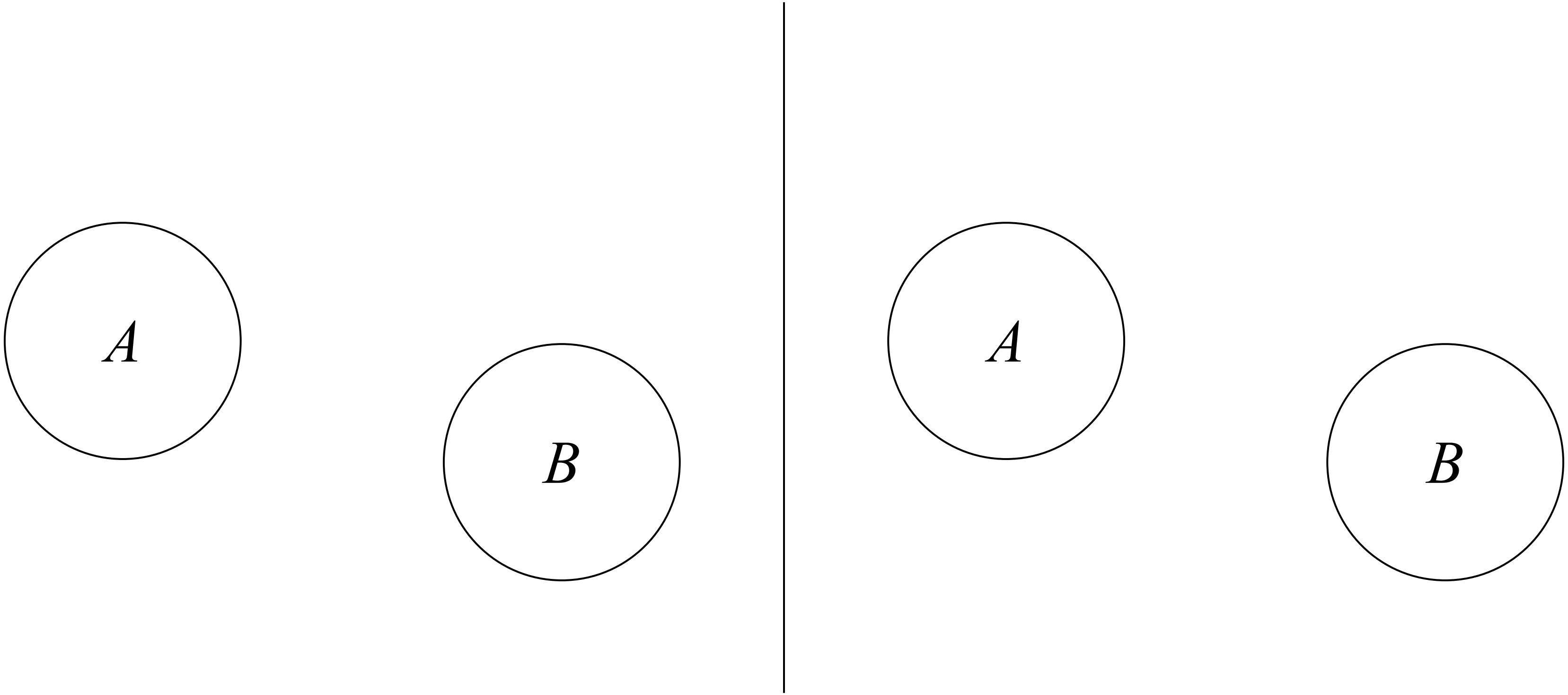
$$diag(m) = \exists c_1, c_2 . c_1 \neq c_2 \\ \wedge\ p(c_1, c_1) \wedge \neg p(c_2, c_2) \\ \wedge\ p(c_1, c_2) \wedge \neg p(c_2, c_1)$$

Models and Diagrams

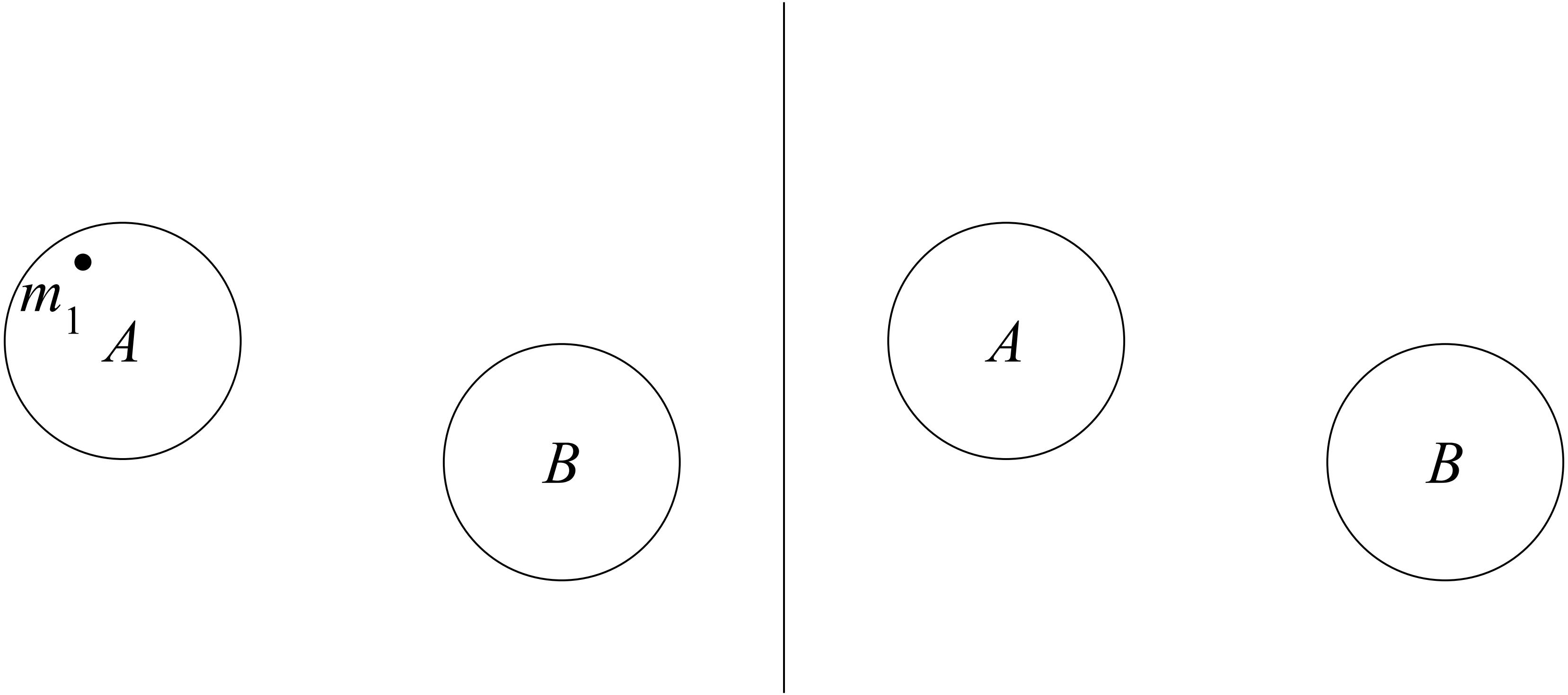


$$diag(m) = \exists c_1, c_2 . c_1 \neq c_2$$
$$\wedge\ p(c_1, c_1) \wedge \neg p(c_2, c_2)$$
$$\wedge\ p(c_1, c_2) \wedge \neg p(c_2, c_1)$$

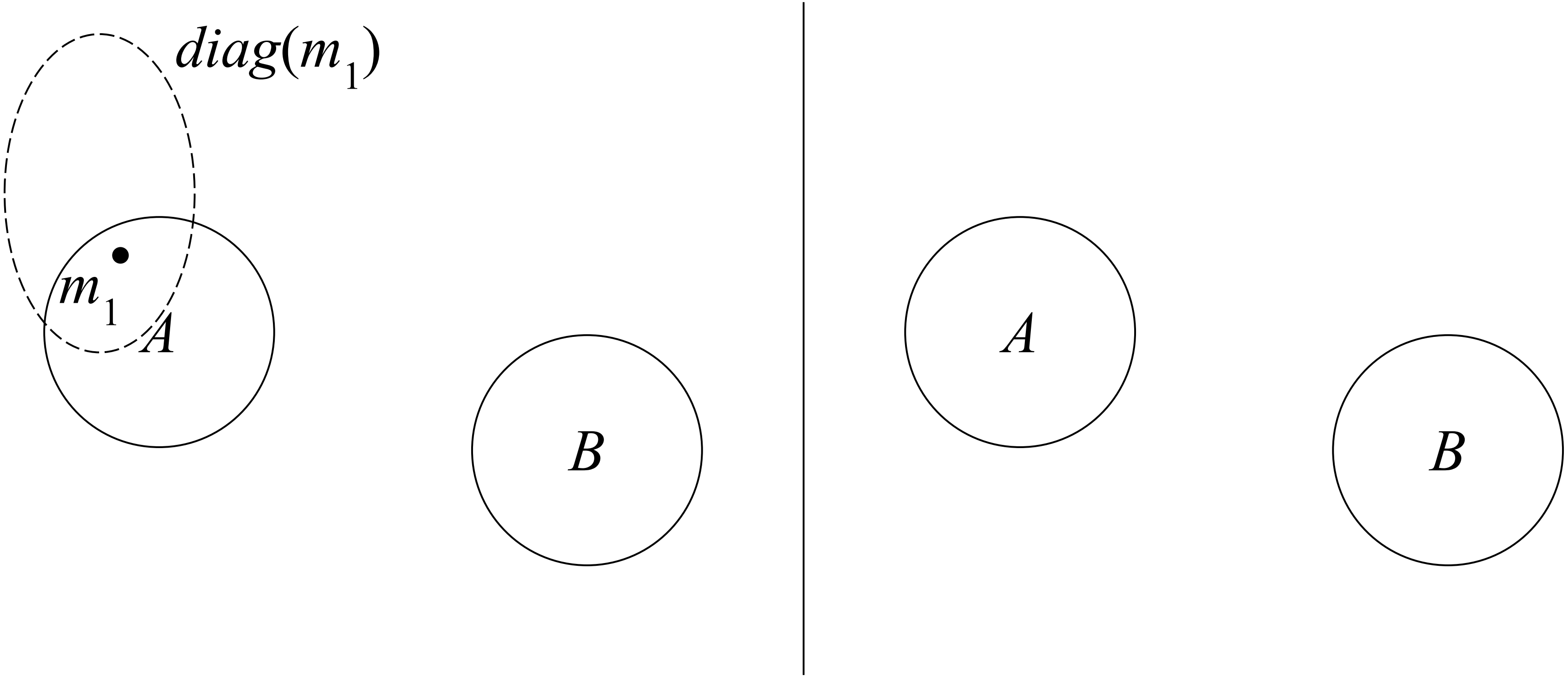
UITP: for ∃-Logic Interpolants



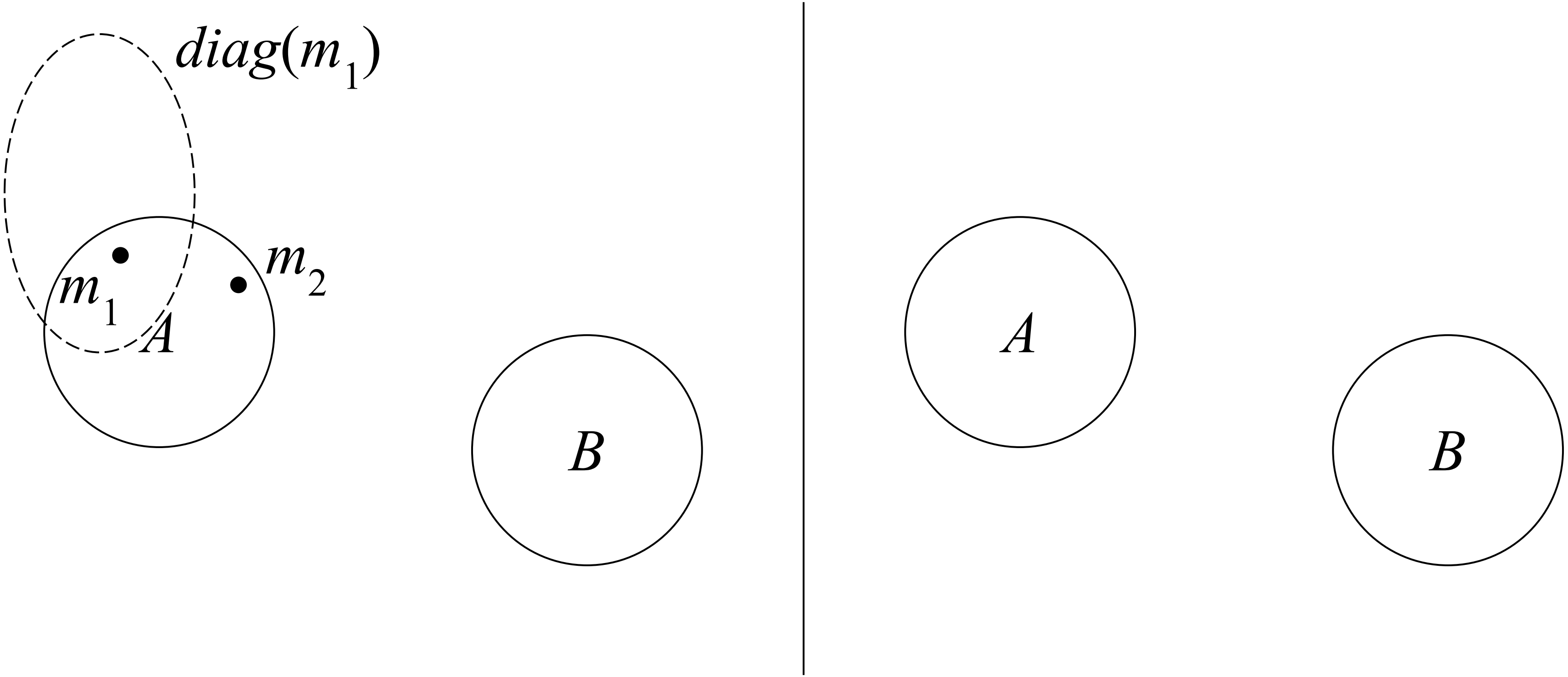
UITP: for ∃-Logic Interpolants



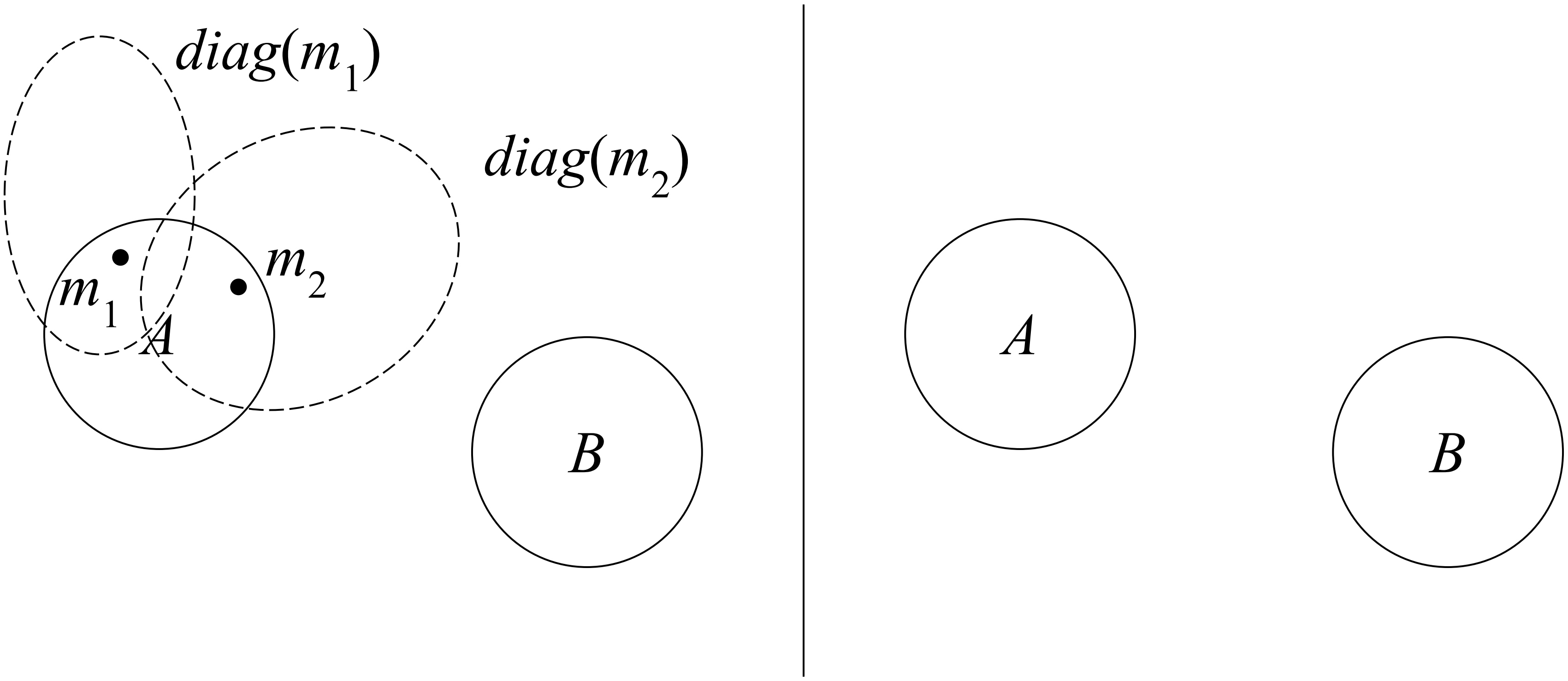
UITP: for ∃-Logic Interpolants



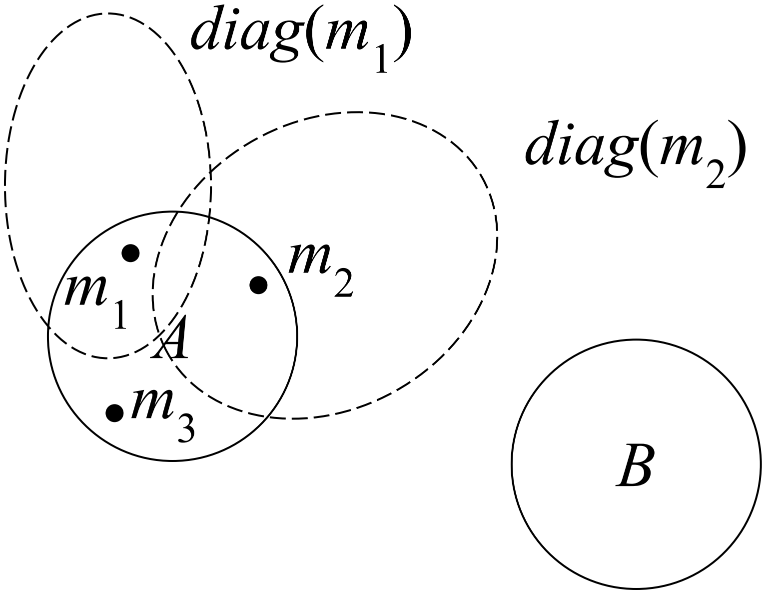
UITP: for ∃-Logic Interpolants

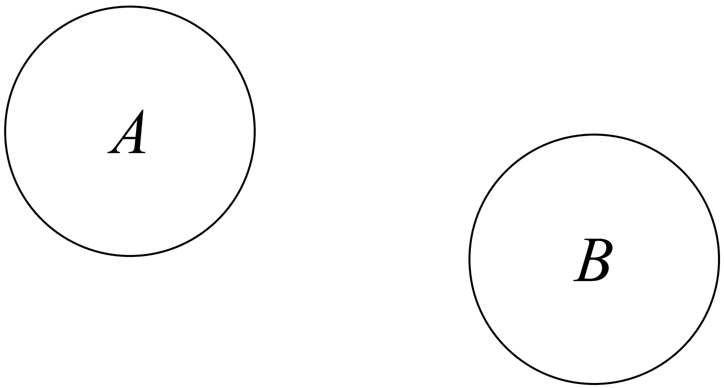


UITP: for ∃-Logic Interpolants

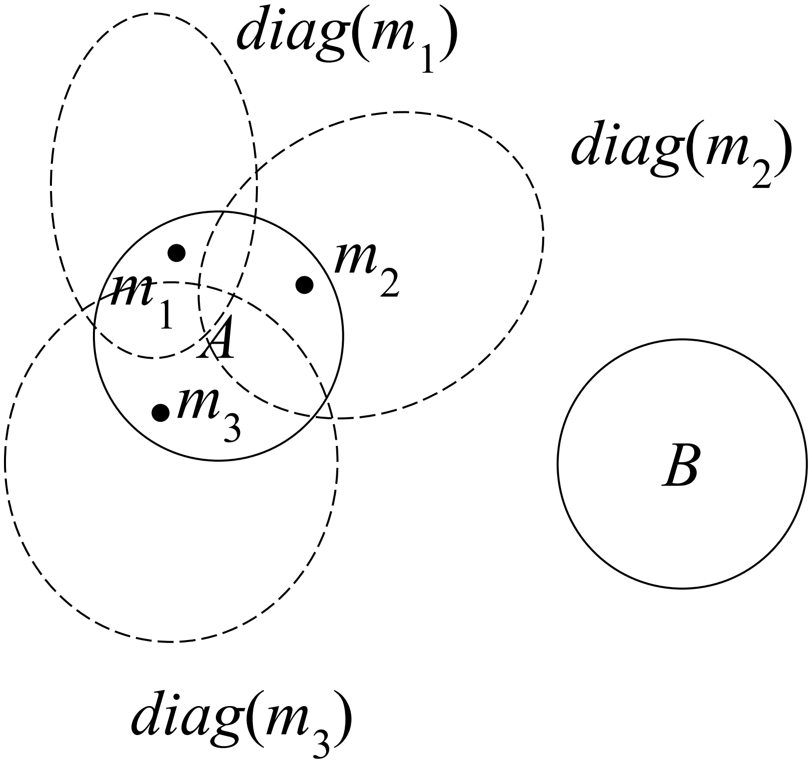


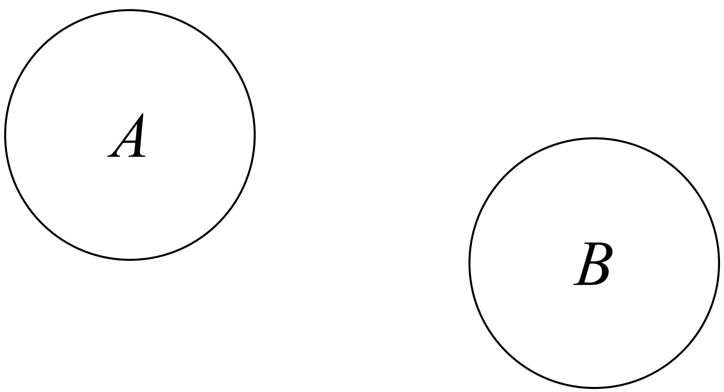
UITP: for ∃-Logic Interpolants





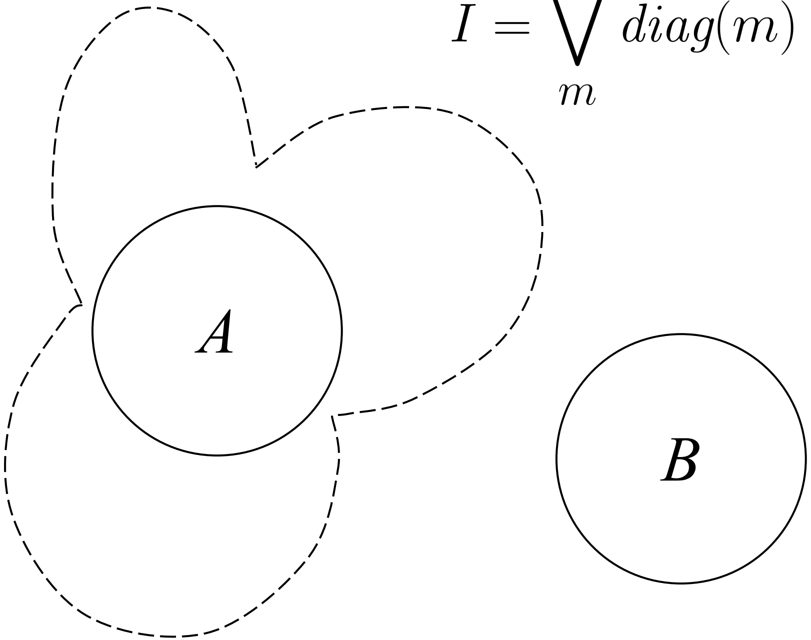
UITP: for ∃-Logic Interpolants

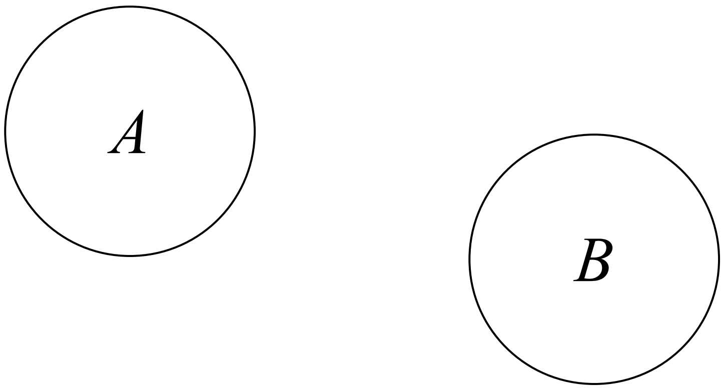




UITP: for ∃-Logic Interpolants

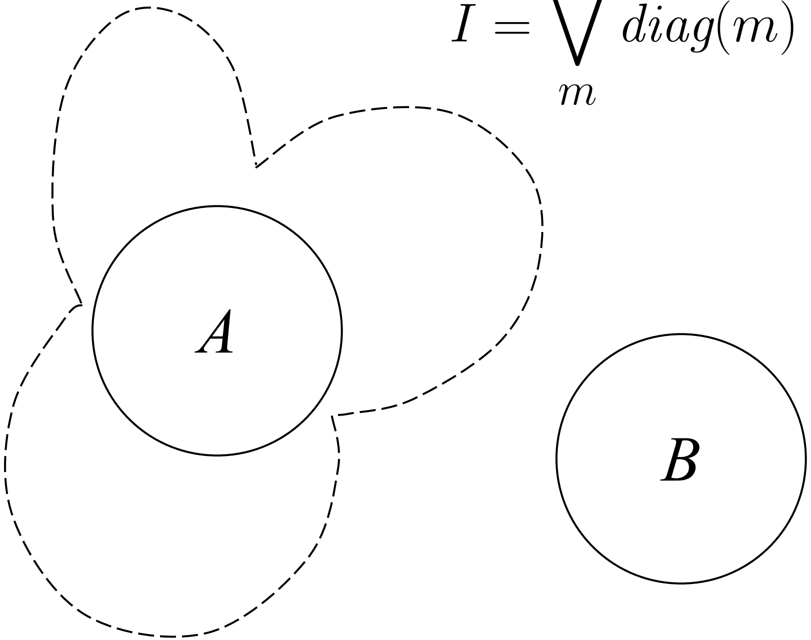
$$I = \bigvee_{m} diag(m)$$

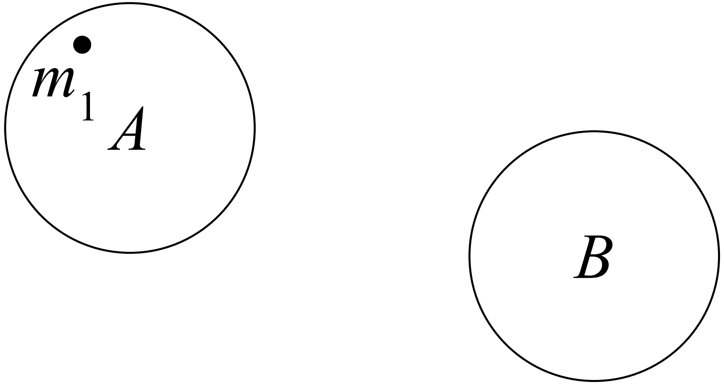




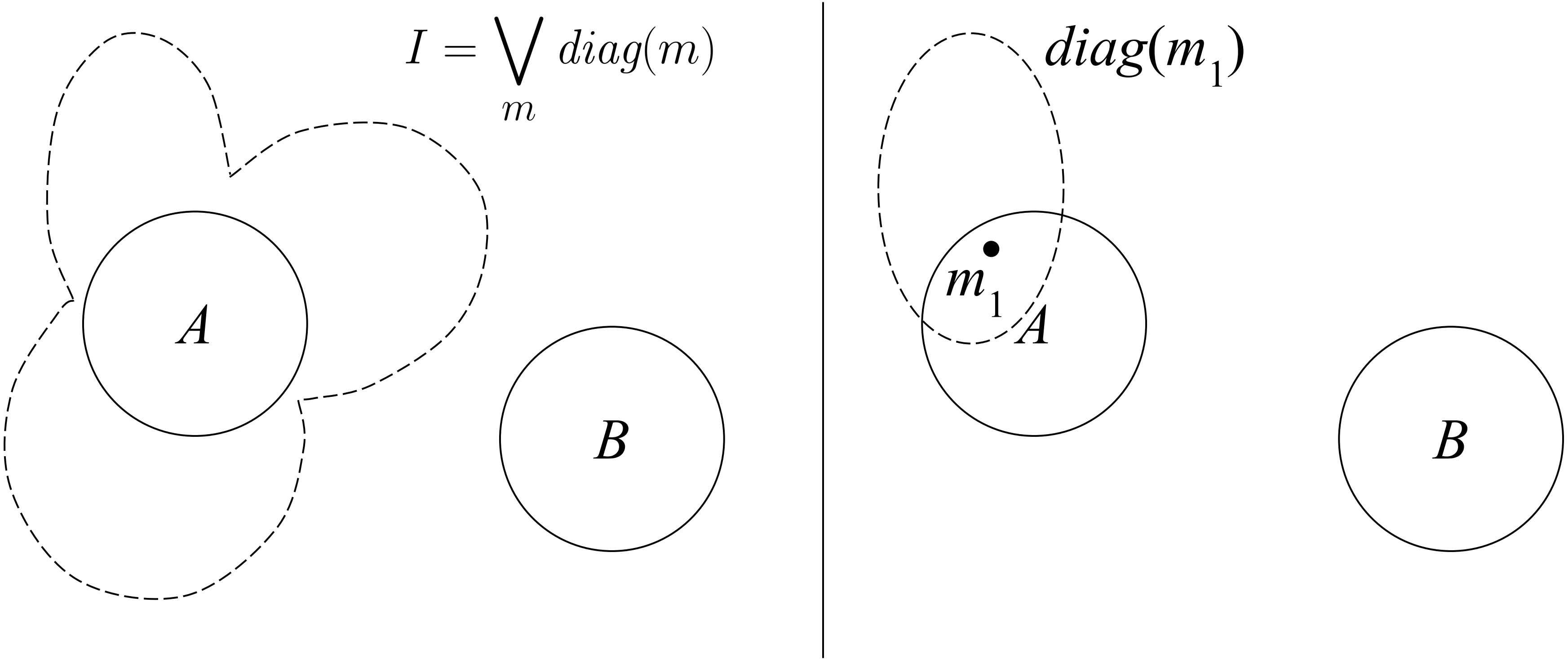
UITP: for ∃-Logic Interpolants

$$I = \bigvee_m diag(m)$$

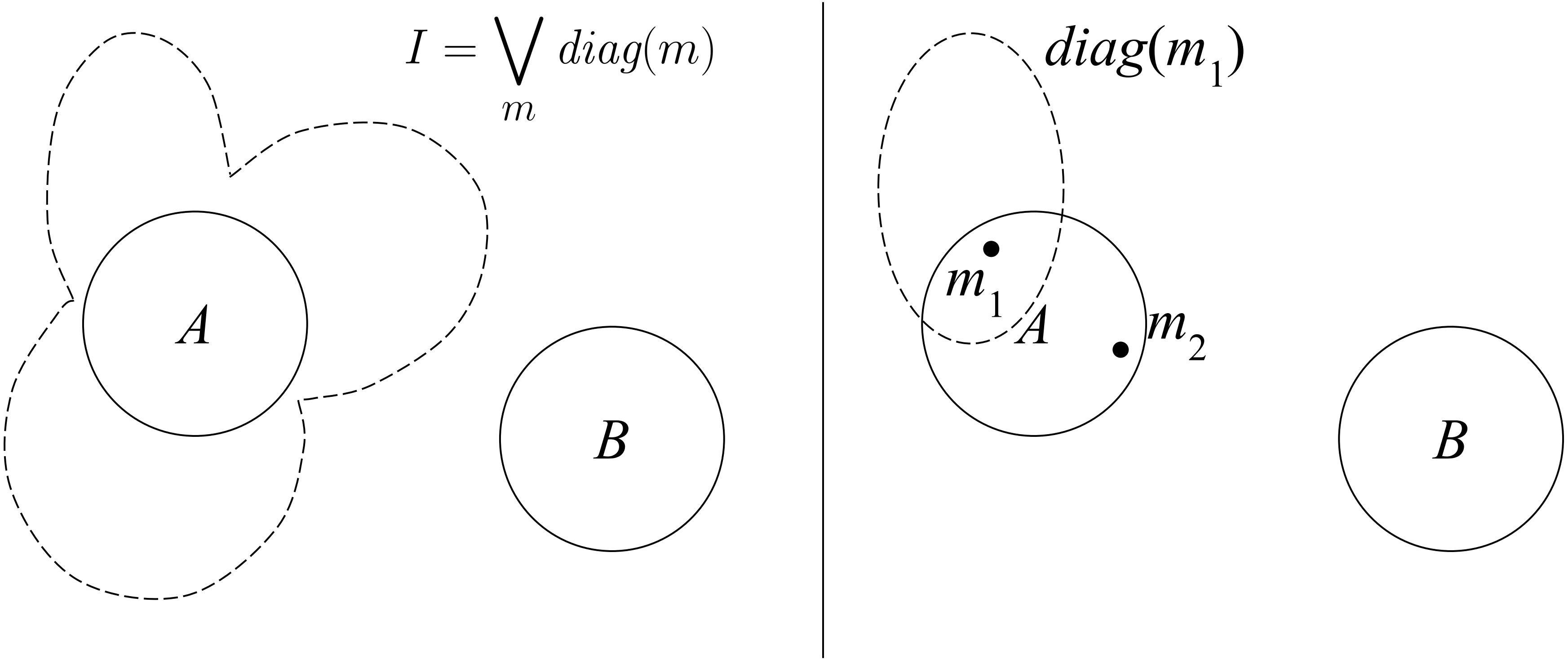




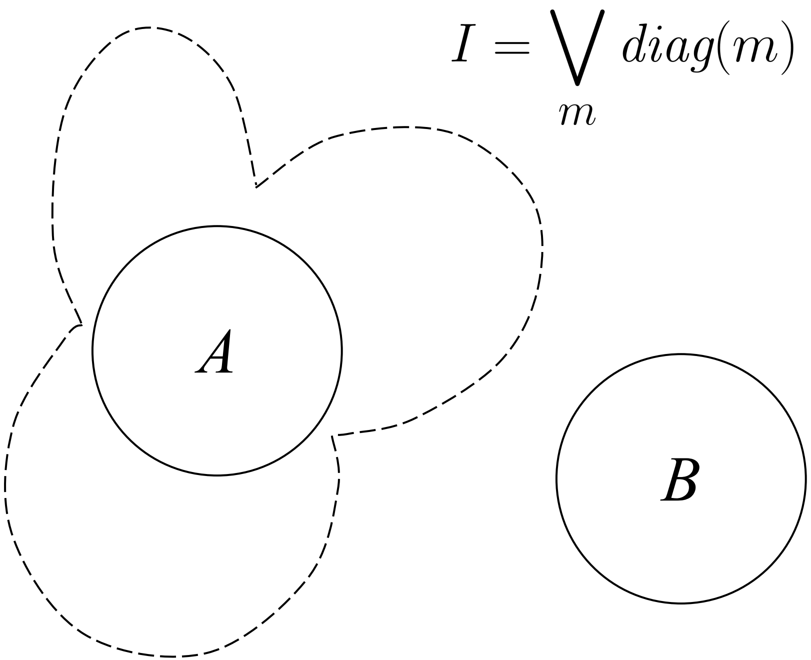
UITP: for ∃-Logic Interpolants

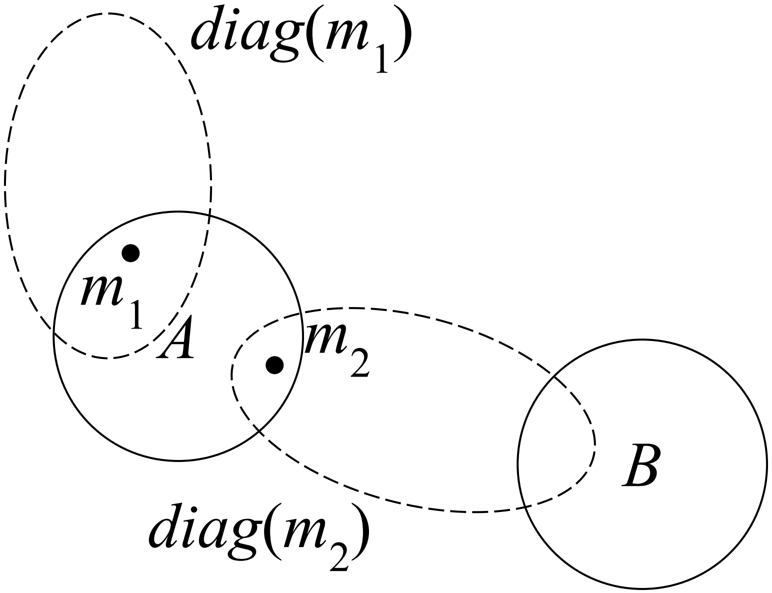


UITP: for ∃-Logic Interpolants

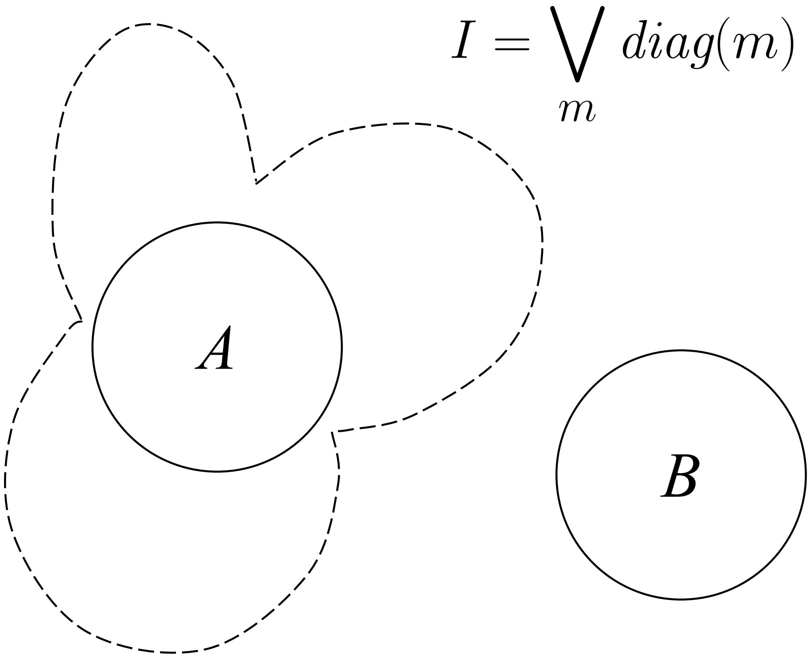


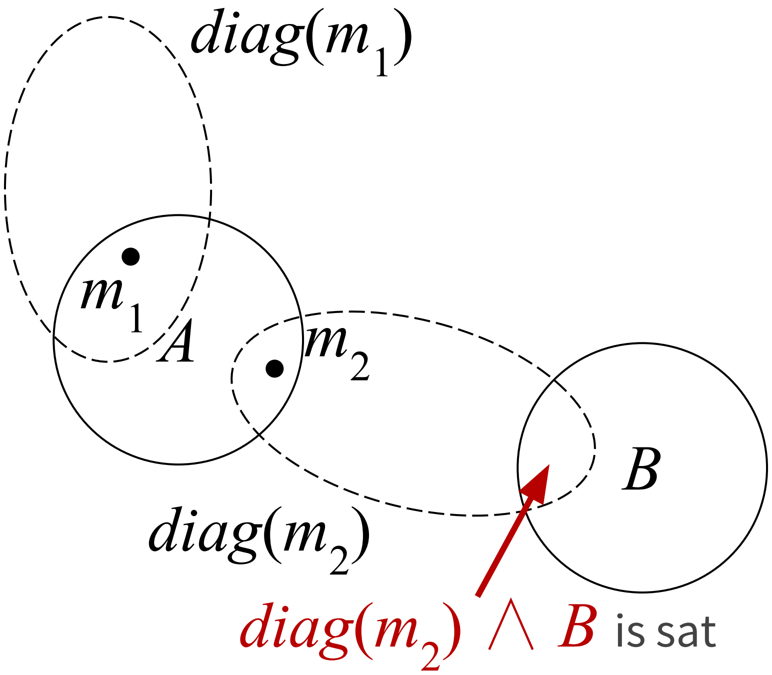
UITP: for ∃-Logic Interpolants



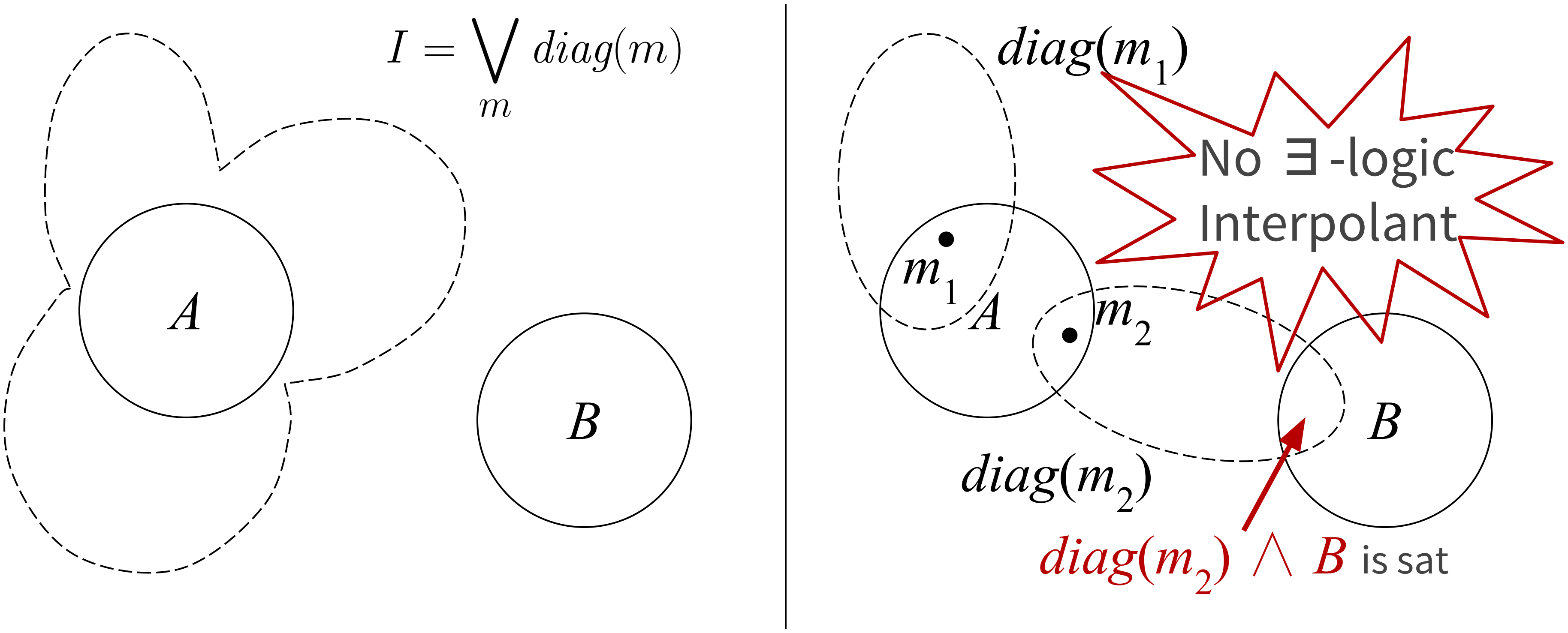


UITP: for ∃-Logic Interpolants





UITP: for ∃-Logic Interpolants

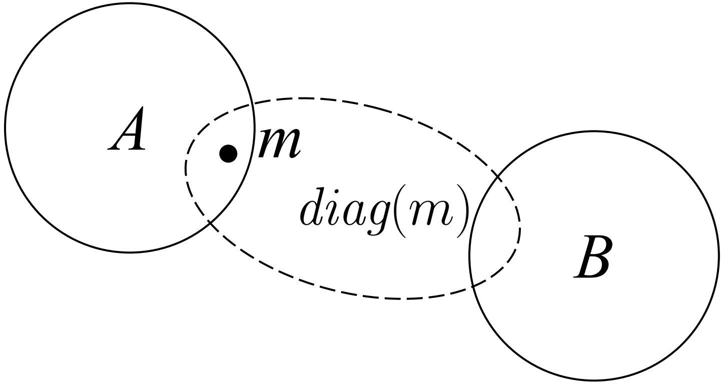


UITP Soundness

- Returning I : interpolant by construction
- Returning $none$ is sound:
  $diag(m)$ is the strongest $\exists$ -logic formula that m models

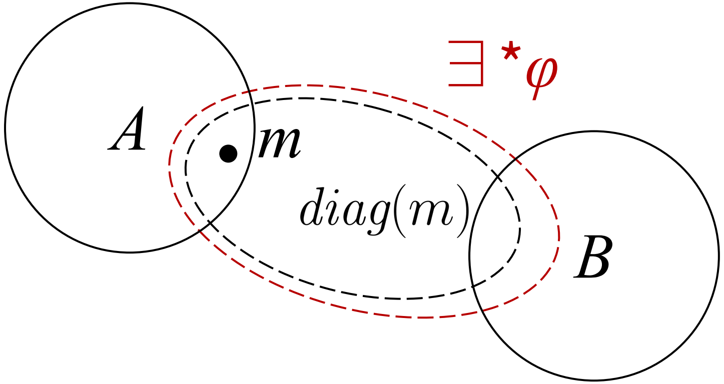
UITP Soundness

- Returning I : interpolant by construction
- Returning $none$ is sound:
  $diag(m)$ is the strongest $\exists$ -logic formula that m models



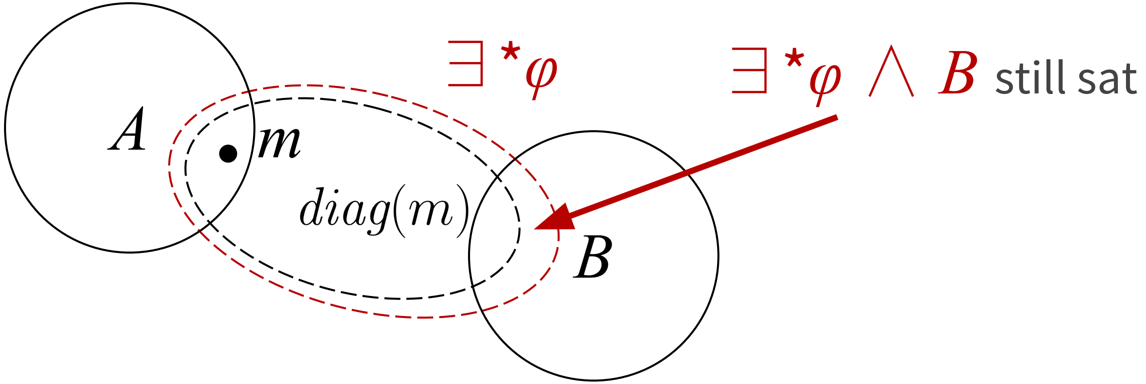
UITP Soundness

- Returning I : interpolant by construction
- Returning *none* is sound:
  $diag(m)$ is the strongest $\exists$-logic formula that m models



UITP Soundness

- Returning I : interpolant by construction
- Returning *none* is sound:
  $diag(m)$ is the strongest $\exists$-logic formula that m models



UITP Termination (and Completeness)

EPR small model property:

All EPR A have a bound k
such that $m \models A \rightarrow \exists m_{small}$:

- $m_{small} \models A$
- $m_{small} \subseteq m$
- $|m_{small}| \leq k$

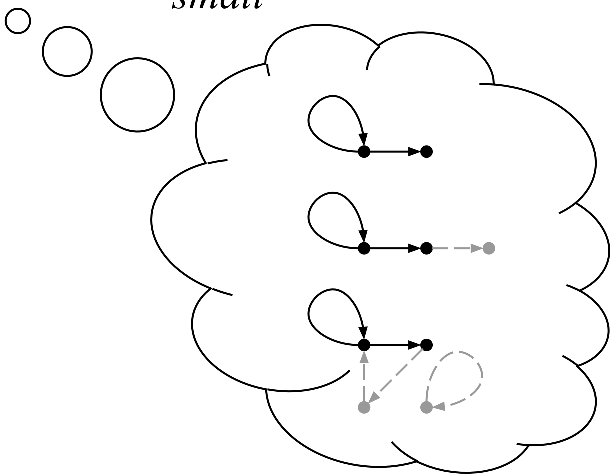
UITP Termination (and Completeness)

EPR small model property:

All EPR A have a bound k such that $m \models A \rightarrow \exists m_{small}$:

- $m_{small} \models A$
- $m_{small} \subseteq m$
- $|m_{small}| \leq k$

So $m \models diag(m_{small})$



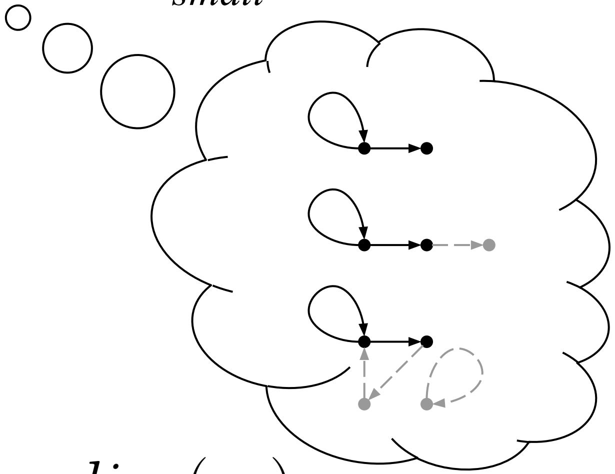
UITP Termination (and Completeness)

EPR small model property:

All EPR A have a bound k such that $m \models A \rightarrow \exists m_{small}$:

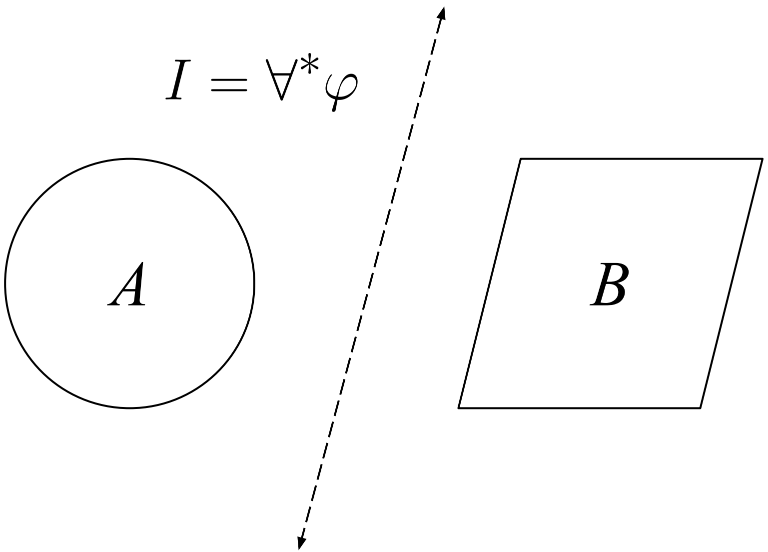
- $m_{small} \models A$
- $m_{small} \subseteq m$
- $|m_{small}| \leq k$

So $m \models diag(m_{small})$

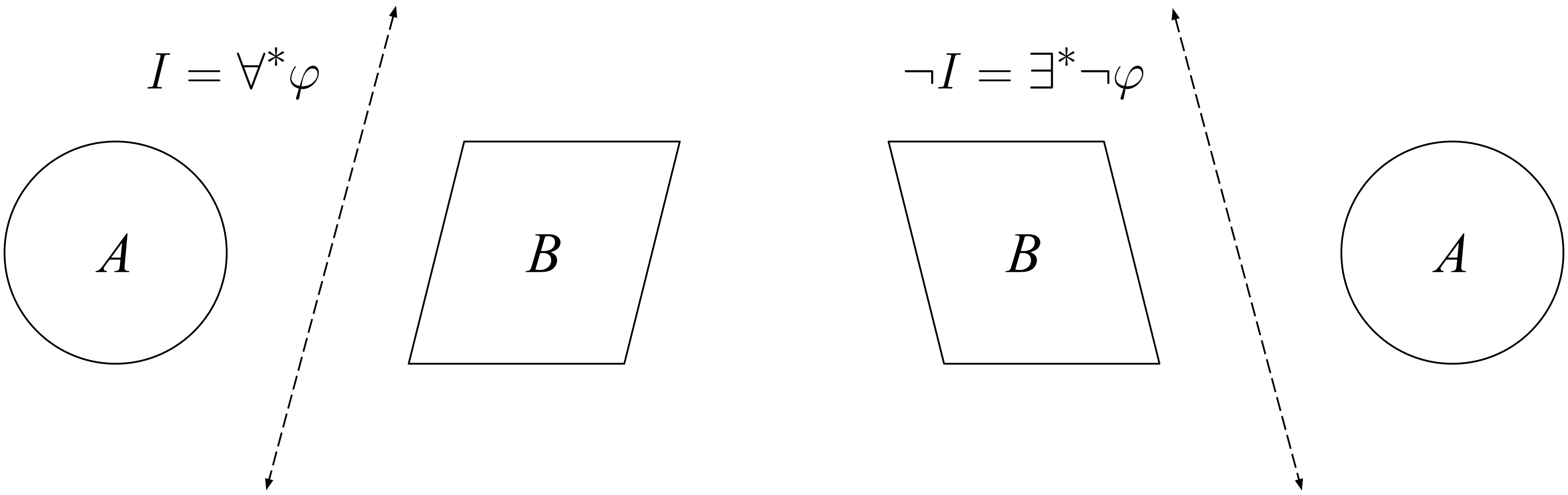


$$A \rightarrow \bigvee_{m \models A : |m| \leq k} diag(m)$$

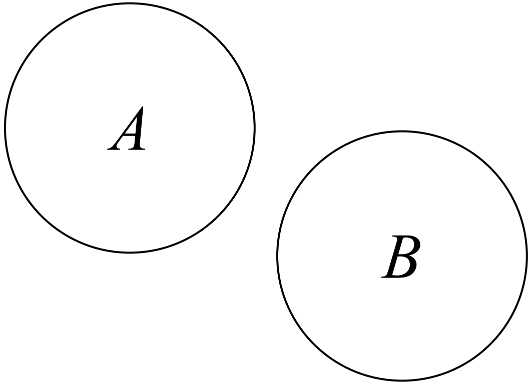
UITP: for ∀-Logic Interpolants



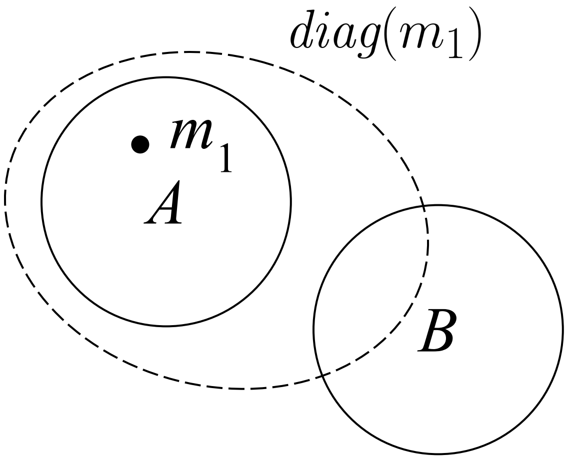
UITP: for ∀-Logic Interpolants



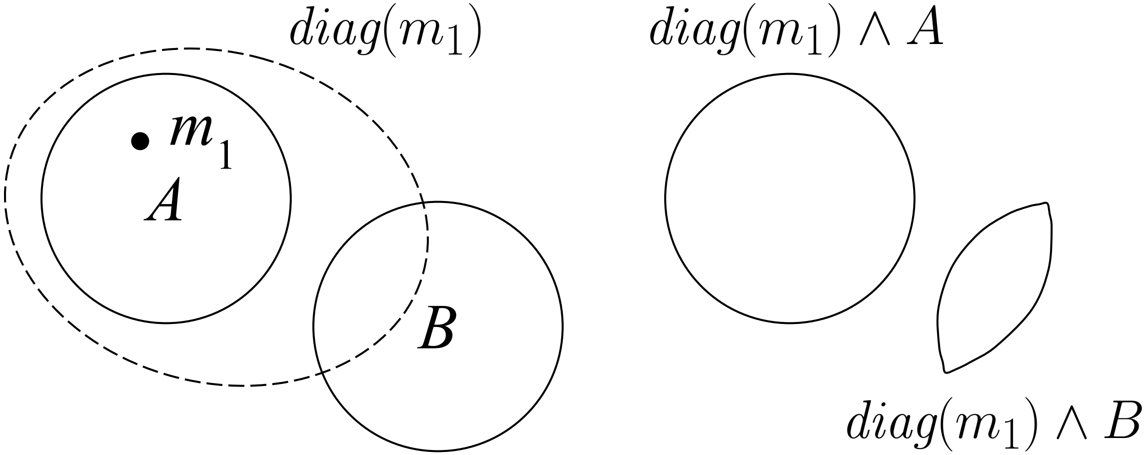
BITP: for AF-Logic Interpolants



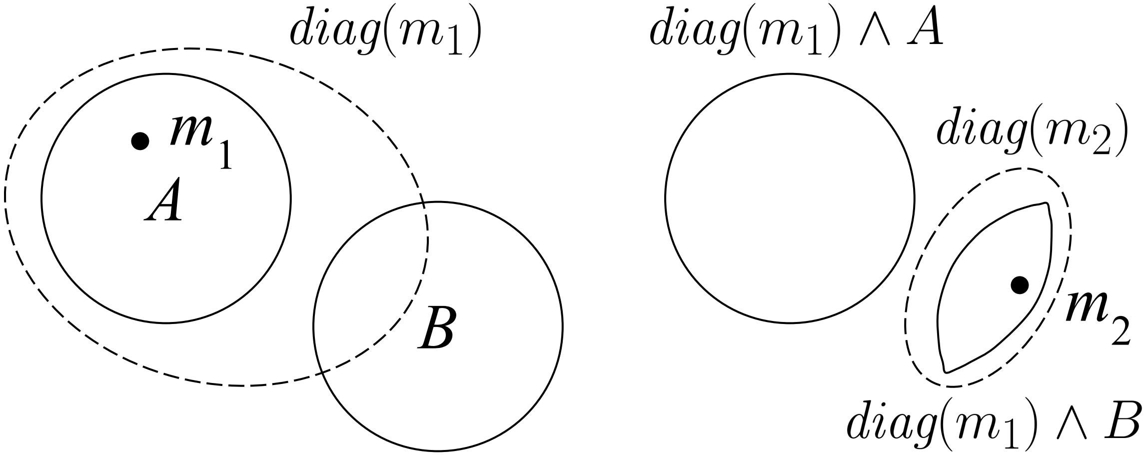
BITP: for AF-Logic Interpolants



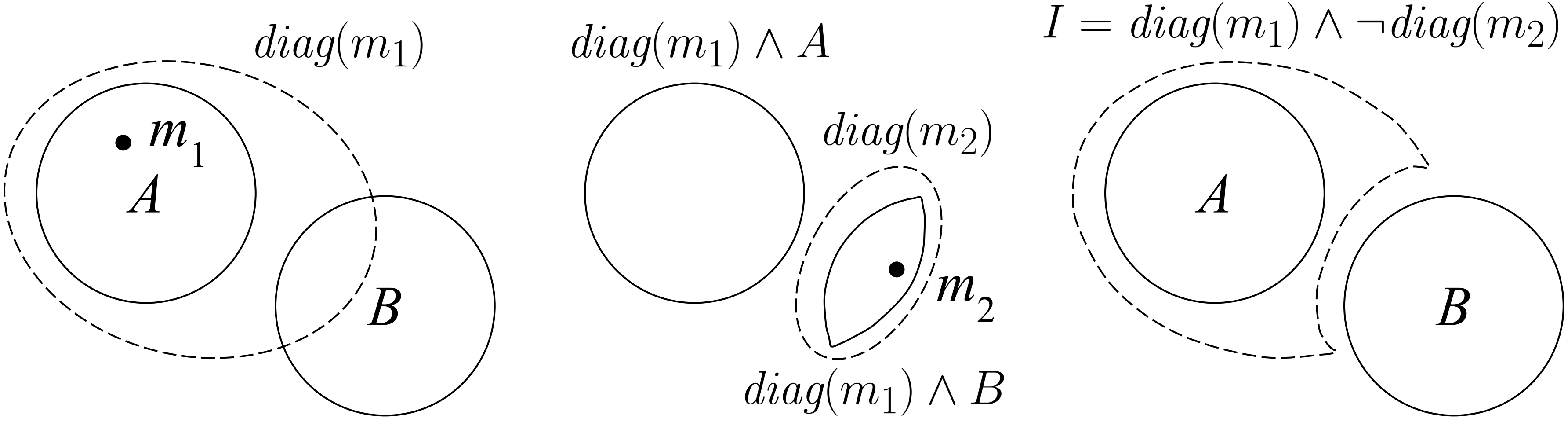
BITP: for AF-Logic Interpolants



BITP: for AF-Logic Interpolants

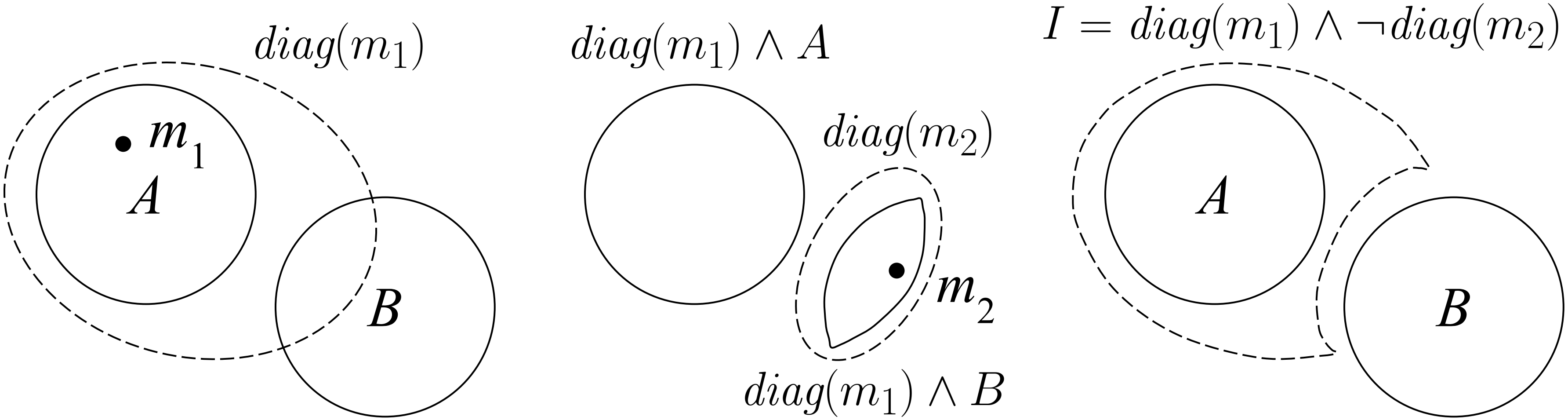


BITP: for AF-Logic Interpolants



BITP: for AF-Logic Interpolants

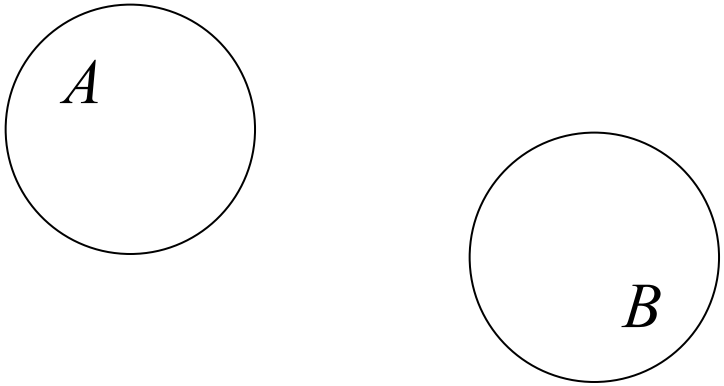
$\exists^{*}\varphi_1 \wedge \forall^{*}\varphi_2$



BITP Soundness and Relative Completeness

Soundness: returned I is interpolant by construction

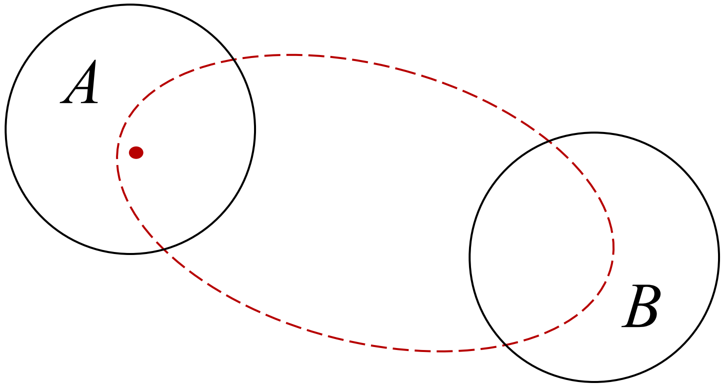
Rel. Compl.: Existence of AF-logic interpolant → termination



BITP Soundness and Relative Completeness

Soundness: returned I is interpolant by construction

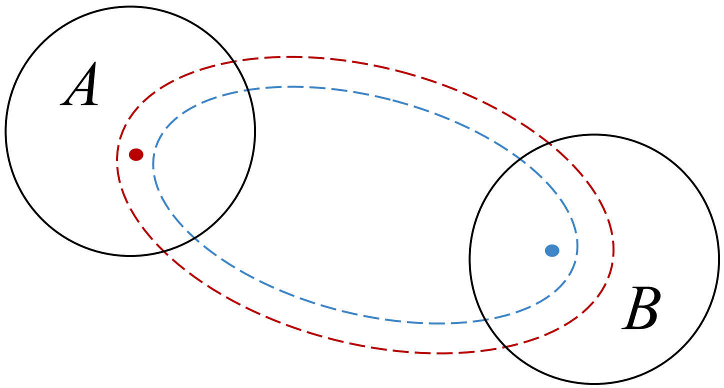
Rel. Compl.: Existence of AF-logic interpolant → termination



BITP Soundness and Relative Completeness

Soundness: returned I is interpolant by construction

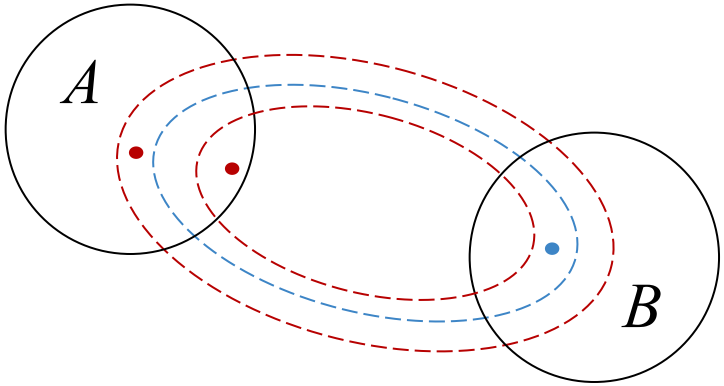
Rel. Compl.: Existence of AF-logic interpolant → termination



BITP Soundness and Relative Completeness

Soundness: returned I is interpolant by construction

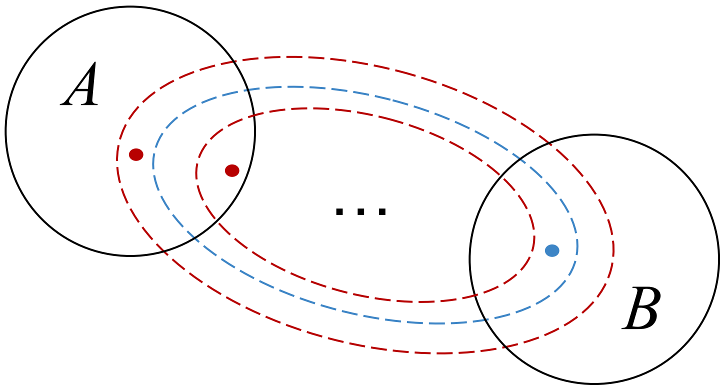
Rel. Compl.: Existence of AF-logic interpolant → termination



BITP Soundness and Relative Completeness

Soundness: returned I is interpolant by construction

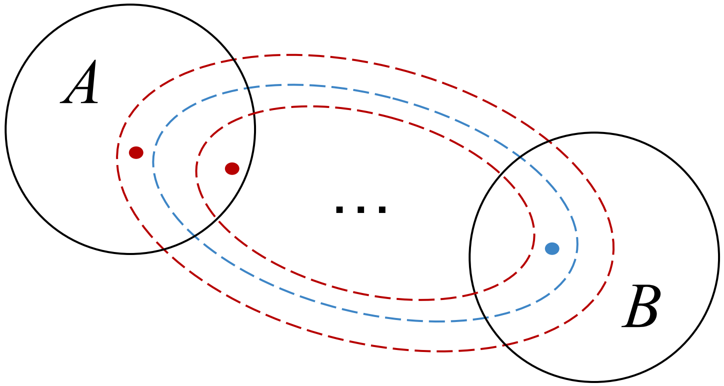
Rel. Compl.: Existence of AF-logic interpolant → termination



BITP Soundness and Relative Completeness

Soundness: returned I is interpolant by construction

Rel. Compl.: Existence of AF-logic interpolant → termination



If $\varphi \in$ AF-logic

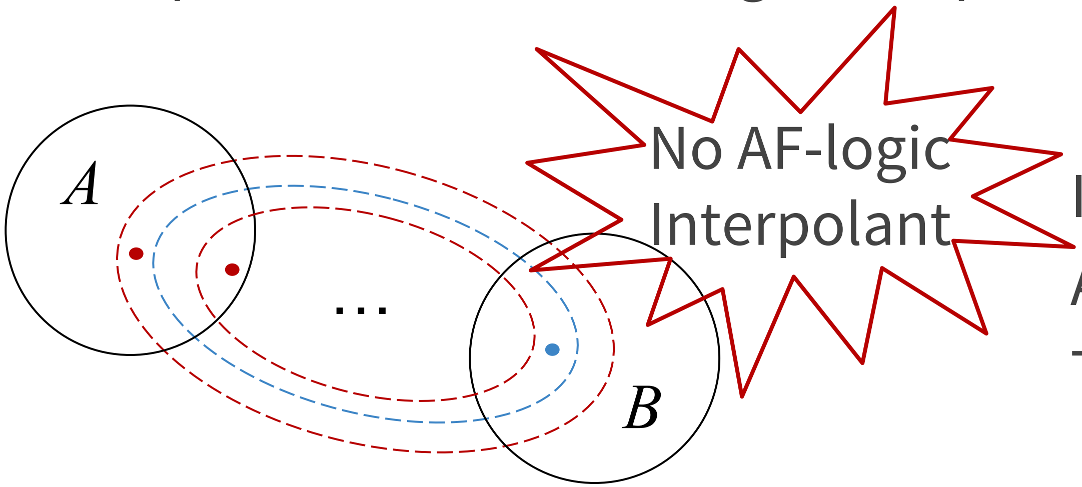
And $A \rightarrow \varphi$

Then $\varphi \wedge B$ is sat

BITP Soundness and Relative Completeness

Soundness: returned I is interpolant by construction

Rel. Compl.: Existence of AF-logic interpolant → termination



If $\varphi \in$ AF-logic

And $A \rightarrow \varphi$

Then $\varphi \wedge B$ is sat

Experiments

ITPV: an interpolation-based verifier

Compared to $PDR_\forall$ [Itzhaky et al., 2014] on linked-list programs

Experiments

ITPV: an interpolation-based verifier

Compared to $PDR_{\forall}$ [Itzhaky et al., 2014] on linked-list programs

Mostly comparable in finding $\forall$-logic invariants

ITPV can find AF-logic invariants

Experiments

ITPV: an interpolation-based verifier

Compared to $\text{PDR}_{\forall}$ [Itzhaky et al., 2014] on linked-list programs

Mostly comparable in finding $\forall$-logic invariants

ITPV can find AF-logic invariants

Conclusion

UITP and BITP interpolate EPR formulae

UITP: sound/complete finding interpolants in ∃- and ∀-logic

BITP: sound/rel.comp. finding interpolants in AF-logic