Independence-friendly logics

Julian Bradfield

Laboratory for Foundations of Computer Science
University of Edinburgh

Introduction

Independence-friendly logic is a variation and extension of first-order logic with a natural(?) concurrent or distributed interpretation of its semantics. Its proponent(s) have argued for its usefulness to various communities – logicians, mathematicians, linguists.

Introduction

Independence-friendly logic is a variation and extension of first-order logic with a natural(?) concurrent or distributed interpretation of its semantics. Its proponent(s) have argued for its usefulness to various communities – logicians, mathematicians, linguists.

Many others have begged to differ.

Talk Structure

- Independence-friendly logics
  - the Henkin ancestor
  - Hintikka's IF logic
  - and issues therewith
  - linguistic applications
- IF modal logic à la Henkin
- and its mu-calculus
- IF modal logic à la Hintikka
  - sequential (Tulenheimo)
  - concurrent (JCB)
  - mu-calculus

Henkin quantifiers

IF logic is another way of presenting the well known Henkin quantifiers.

Henkin (branching) quantifiers allow some quantifiers to be 'independent' of other quantifiers:

$$\begin{matrix} \forall x\, \exists y \\ \forall u\, \exists v \end{matrix} . \phi(x, y, u, v)$$

Henkin quantifiers

IF logic is another way of presenting the well known Henkin quantifiers.

Henkin (branching) quantifiers allow some quantifiers to be 'independent' of other quantifiers:

$$\begin{matrix} \forall x \, \exists y \\ \forall u \, \exists v \end{matrix} . \phi(x, y, u, v)$$

with a semantics via Skolem functions with a limited number of arguments:

$$\exists f, g . \forall x, u . \phi(x, f(x), u, g(u))$$

Henkin quantifiers

IF logic is another way of presenting the well known Henkin quantifiers.

Henkin (branching) quantifiers allow some quantifiers to be 'independent' of other quantifiers:

$$\begin{matrix} \forall x \, \exists y \\ \forall u \, \exists v \end{matrix} . \phi(x, y, u, v)$$

with a semantics via Skolem functions with a limited number of arguments:

$$\exists f, g. \forall x, u. \phi(x, f(x), u, g(u))$$

Well known to have Σ^1_1 power.

Not much used, although have appeared fleetingly in hardware verification.

IF logic

Hintikka–Sandu independence-friendly logic is another way of expressing this:

$$\forall x. \exists y. \forall u / \{ x, y \}. \exists v / \{ x, y \}. \phi$$

IF logic

Hintikka–Sandu independence-friendly logic is another way of expressing this:

$$\forall x.\exists y.\forall u/\{x, y\}.\exists v/\{x, y\}.\phi$$

with a semantics originally via imperfect information games: in the verification game, Eloise has to choose a value for v without knowing (equivalently, uniformly in) the values of x and y.

We'll look at IF logic in a bit more detail:

IF syntax

Atomic formulae are usual first-order relations.

Conjunction and disjunction. If ϕ and ψ are formulae, then $\phi \vee \psi$ and $(\phi \wedge \psi)$ are formulae.

IF syntax

Atomic formulae are usual first-order relations.

Conjunction and disjunction. If ϕ and ψ are formulae, then $\phi \vee \psi$ and $(\phi \wedge \psi)$ are formulae.

Quantifiers. If ϕ is a formula, x a variable, and W a finite set of variables, then $\forall x/W.\phi$ and $\exists x/W.\phi$ are formulae. If W is empty, we write just $\forall x.\phi$ and $\exists x.\phi$.

IF syntax

Atomic formulae are usual first-order relations.

Conjunction and disjunction. If ϕ and ψ are formulae, then $\phi \vee \psi$ and $(\phi \wedge \psi)$ are formulae.

Quantifiers. If ϕ is a formula, x a variable, and W a finite set of variables, then $\forall x/W.\phi$ and $\exists x/W.\phi$ are formulae. If W is empty, we write just $\forall x.\phi$ and $\exists x.\phi$.

Game negation. If ϕ is a formula, so is $\sim\phi$.

Independent connectives. If ϕ and ψ are formulae, and W a finite set of variables, then $\phi \vee/W\, \psi$ and $(\phi \wedge/W\, \psi)$ are formulae.

IF syntax

Atomic formulae are usual first-order relations.

Conjunction and disjunction. If ϕ and ψ are formulae, then $\phi \vee \psi$ and $(\phi \wedge \psi)$ are formulae.

Quantifiers. If ϕ is a formula, x a variable, and W a finite set of variables, then $\forall x/W.\phi$ and $\exists x/W.\phi$ are formulae. If W is empty, we write just $\forall x.\phi$ and $\exists x.\phi$.

Game negation. If ϕ is a formula, so is $\sim\phi$.

Independent connectives. If ϕ and ψ are formulae, and W a finite set of variables, then $\phi \vee/W\ \psi$ and $(\phi \wedge/W\ \psi)$ are formulae.

For most purposes in this talk, forget negation (but remember that it's lurking). Also forget the independent connectives (mainly for simplicity).

IF syntax

Atomic formulae are usual first-order relations.

Conjunction and disjunction. If ϕ and ψ are formulae, then $\phi \vee \psi$ and $(\phi \wedge \psi)$ are formulae.

Quantifiers. If ϕ is a formula, x a variable, and W a finite set of variables, then $\forall x/W.\phi$ and $\exists x/W.\phi$ are formulae. If W is empty, we write just $\forall x.\phi$ and $\exists x.\phi$.

Game negation. If ϕ is a formula, so is $\sim\phi$.

Independent connectives. If ϕ and ψ are formulae, and W a finite set of variables, then $\phi \vee/W\ \psi$ and $(\phi \wedge/W\ \psi)$ are formulae.

For most purposes in this talk, forget negation (but remember that it's lurking). Also forget the independent connectives (mainly for simplicity).

This is not Hintikka's language! He only allowed $\exists x/W.\phi$ where W are universal variables; *and* stated that all previous existential varibles are implicitly included in W.

Game semantics for IF

Standard Hintikka-style game for first-order logic, with the addition of *imperfect information*: at $\exists v/\{x\}$, an Eloise-winning strategy for choosing v must be *uniform* in the possible values of x (because she's not supposed to know what x is).

Game semantics for IF

Standard Hintikka-style game for first-order logic, with the addition of *imperfect information*: at $\exists v/\{x\}$, an Eloise-winning strategy for choosing v must be *uniform* in the possible values of x (because she's not supposed to know what x is).

Then as usual a formula is true iff Eloise has a uniform winning strategy.

A formula is false iff Abelard has a uniform winning strategy.

Game semantics for IF

Standard Hintikka-style game for first-order logic, with the addition of *imperfect information*: at $\exists v/\{x\}$, an Eloise-winning strategy for choosing v must be *uniform* in the possible values of x (because she's not supposed to know what x is).

Then as usual a formula is true iff Eloise has a uniform winning strategy.

A formula is false iff Abelard has a uniform winning strategy.

Tertium datur! A formula is undetermined if neither player has a uniform winning strategy.

Some friendly formulae

The singleton formula:

$$\forall x.\exists y/x.x = y$$

Some friendly formulae

The singleton formula:

$$\forall x. \exists y/x. x = y$$

A valid(?) formula:

$$\exists x. \exists y/x. x = y$$

Some friendly formulae

The singleton formula:

$$\forall x.\exists y/x.x = y$$

A valid(?) formula:

$$\exists x.\exists y/x.x = y$$

The Hintikka infinity formula:

$$\exists c.\forall x.\forall u.\exists y/u.\exists v/x, y.(y = v \Leftrightarrow x = u) \wedge y \neq c$$

Some less friendly formulae

The Hodges signalling formula:

$$\forall x.\exists z.\exists y/x.x = y$$

Some less friendly formulae

The Hodges signalling formula:

$$\forall x.\exists z.\exists y/x.x = y$$

The Caicedo–Krynicki infinity formula

$$\exists c.\forall x.\exists y.\exists z/x.x = z \wedge y \neq c$$

Skolem semantics

Rather than games, use Skolem functions as originally done for Henkin quantifiers.

Any IF sentence can be skolemized to a Σ^1_1 sentence – and vice versa.

The skolemization is either true or false . . . skolem truth iff game truth. (Skolem function = choice function in strategy)

Skolem semantics

Rather than games, use Skolem functions as originally done for Henkin quantifiers.

Any IF sentence can be skolemized to a Σ^1_1 sentence – and vice versa.

The skolemization is either true or false . . . skolem truth iff game truth. (Skolem function = choice function in strategy)

Do you start skolemization from the outside and work in, or vice versa? It matters – inside–out is a Good Thing (but Hintikka implicitly did outside–in).

Trump semantics of IF

Hodges' Tarski-style semantics for IF logic.

Really just a coding up of strategies.

The meaning of a formula is a *set of sets* of assignments to its free variables. Each set is a *trump*, from which Eloise can win.

Trump semantics of IF

Hodges' Tarski-style semantics for IF logic.

Really just a coding up of strategies.

The meaning of a formula is a *set of sets* of assignments to its free variables. Each set is a *trump*, from which Eloise can win.

Example: the trumps of $\exists y/\{x\}.x = y$ are the singleton sets $\{\{a\}\}$. Hence $\forall x.\exists y/\{x\}.x = y$ has trumps only if the domain has just one value.

Trump semantics of IF

Hodges' Tarski-style semantics for IF logic.

Really just a coding up of strategies.

The meaning of a formula is a *set of sets* of assignments to its free variables. Each set is a *trump*, from which Eloise can win.

Example: the trumps of $\exists y/\{ x \}.x = y$ are the singleton sets $\{ \{ a \} \}$. Hence $\forall x.\exists y/\{ x \}.x = y$ has trumps only if the domain has just one value.

It's easily seen that naive computation of trumps is doubly exponential in formula size. . .

Trump semantics of IF

Hodges' Tarski-style semantics for IF logic.

Really just a coding up of strategies.

The meaning of a formula is a *set of sets* of assignments to its free variables. Each set is a *trump*, from which Eloise can win.

Example: the trumps of $\exists y/\{ x \}.x = y$ are the singleton sets $\{ \{ a \} \}$. Hence $\forall x.\exists y/\{ x \}.x = y$ has trumps only if the domain has just one value.

It's easily seen that naive computation of trumps is doubly exponential in formula size. . .

. . . but IF sentences are ∃SO-expressible, hence NP.

By the way . . .

Lots of things you take for granted in FOL can't be done in IF.

E.g.: Re-use of variables is *not* something trivially removable by renaming.

By the way . . .

Lots of things you take for granted in FOL can't be done in IF.

E.g.: Re-use of variables is *not* something trivially removable by renaming.

We haven't really mentioned open formulae: Hintikka didn't deal with them, Hodges, Caicedo–Krynicki do.

By the way . . .

Lots of things you take for granted in FOL can't be done in IF.

E.g.: Re-use of variables is *not* something trivially removable by renaming.

We haven't really mentioned open formulae: Hintikka didn't deal with them, Hodges, Caicedo–Krynicki do.

Janssen has a different semantics.

Why?

Much of the complexity is because IF games are not just imperfect information, but imperfect recall – players have to forget things, both things they did and things they know.

Why?

Much of the complexity is because IF games are not just imperfect information, but imperfect recall – players have to forget things, both things they did and things they know.

This can be addressed by making Eloise and Abelard into teams (as is natural in the modal logics we'll see later). Väänänen is even developing 'team logic'.

See Dechesne's thesis for more on this.

What is IF logic good for?

According to Hintikka:

- IF logic is a Right Thing, and first-order logic is a Wrong Thing.

What is IF logic good for?

According to Hintikka:

- IF logic is a Right Thing, and first-order logic is a Wrong Thing.
- IF logic is good for mathematics. (E.g. definition of uniform continuity.)

What is IF logic good for?

According to Hintikka:

- IF logic is a Right Thing, and first-order logic is a Wrong Thing.
- IF logic is good for mathematics. (E.g. definition of uniform continuity.)
- IF logic is good for natural language. (E.g. 'Some relative of every townsman knows some friend of every countryman.')

Linguistic applications

Goes back to early 70s work on branching quantifiers in English. 'Some relative of every townsman knows some friend of every countryman' is a Henkin quantified sentence (allegedly).

Linguistic applications

Goes back to early 70s work on branching quantifiers in English. 'Some relative of every townsman knows some friend of every countryman' is a Henkin quantified sentence (allegedly).

Barwise tried such sentences out on people – people don't appear to have the Henkin interpretation.

Linguistic applications

Goes back to early 70s work on branching quantifiers in English. 'Some relative of every townsman knows some friend of every countryman' is a Henkin quantified sentence (allegedly).

Barwise tried such sentences out on people – people don't appear to have the Henkin interpretation.

One class of exception: $\binom{\text{most}}{\text{most}}$, as in 'most of the boys and most of the girls kissed each other'.

Linguistic applications

Goes back to early 70s work on branching quantifiers in English. ‘Some relative of every townsman knows some friend of every countryman’ is a Henkin quantified sentence (allegedly).

Barwise tried such sentences out on people – people don't appear to have the Henkin interpretation.

One class of exception: $\binom{\text{most}}{\text{most}}$, as in ‘most of the boys and most of the girls kissed each other’.

However, Sevenster searched the British National Corpus for such sentences – without success.

(See Sevenster's thesis for more about this.)

A use of IF logic in formal methods?

I claim: IF logic provides a natural meta-language for some natural concurrent temporal logics with obvious applicability.

Concurrent modalities

There are situations in which it seems natural to want to write

$$\begin{matrix}[\,] \\ \langle\rangle\end{matrix}\,\phi$$

Concurrent modalities

There are situations in which it seems natural to want to write

$$\genfrac{}{}{0pt}{}{[\,]}{\langle\,\rangle}\,\phi$$

For example:

- Scissors–Paper–Stone
- Bidding on Ebay . . .
- or in Scottish house-buying.
- Trains entering a section of single-line working.

In all cases, we need to choose an action that gives us the desired results, without knowing what's happening elsewhere.

It's easy to define $\genfrac{}{}{0pt}{}{[\,]}{\langle\,\rangle}$, $\genfrac{}{}{0pt}{}{[\,]}{[\,]}\genfrac{}{}{0pt}{}{\langle\,\rangle}{\langle\,\rangle}$ etc. on systems with two parallel components.

Henkin modal logic

Consider systems of parallel components $T = \|_{1 \leq i \leq n} T_i$ (for some $\|$).

Define our desired modalities $\substack{[\,]\\ \langle\rangle}$, $\substack{[\,]\\ [\,]}\substack{\langle\rangle\\ \langle\rangle}$ etc. via Henkin quantifiers:
e.g. on $T = T_1 \|_{\mathcal{S}} T_2$,

$$(s_1, s_2) \models \begin{matrix} [a_1] \, \langle b_1 \rangle \\ [a_2] \, \langle b_2 \rangle \end{matrix} \phi$$

holds iff

$$\begin{matrix} \forall(a_1, s_1') \; \exists(b_1, s_1'') \\ \forall(a_2, s_2' \;\; \exists(b_2, s_2'') \end{matrix} (s_1, s_2) \xrightarrow{a_1 \otimes a_2} (s_1', s_2') \xrightarrow{b_1 \otimes b_2} (s_1'', s_2'') \wedge (s_1'', s_2'') \models \phi$$

(This is lying a bit . . .)

What use is $\genfrac{}{}{0pt}{}{[\,]}{[\,]}\genfrac{}{}{0pt}{}{\langle\rangle}{\langle\rangle}$?

In contexts where you have cooperating agents with communication problems (e.g. firewalled, on Mars, etc.) and they need to make independent decisions to achieve a common goal.

Think in terms of strategies: $\genfrac{}{}{0pt}{}{[\,]}{[\,]}\genfrac{}{}{0pt}{}{\langle\rangle}{\langle\rangle}\phi$ says each of your teams has a strategy to make a choice in reaction to local events, such that good things happen.

$\genfrac{}{}{0pt}{}{[\,]}{[\,]}\genfrac{}{}{0pt}{}{\langle\rangle}{\langle\rangle}$, like the Henkin quantifier, is quite expressive: it can be NP-complete to verify.

Henkin modal mu-calculus and beyond

$\begin{smallmatrix}[\,]\\[\,]\end{smallmatrix}$ $\begin{smallmatrix}\langle\rangle\\\langle\rangle\end{smallmatrix}$ etc. are just modal operators on global states, and it is trivial to add them to modal mu-calculus to get a temporal logic, which subsumes and extends the ATL of Alur, Henzinger and Kupferman.

Henkin modal mu-calculus and beyond

$[]\langle\rangle$ etc. are just modal operators on global states, and it is trivial to add them to modal mu-calculus to get a temporal logic, which subsumes and extends the ATL of Alur, Henzinger and Kupferman.

But this is boring – and not the right sort of expressive. We want to allow all sorts of other things to happen between the $[]$ and the $\langle\rangle$.

Would like to write things like $[a]\langle b\rangle\phi \otimes [a]\langle b\rangle\psi$, or

$$\bigotimes \begin{array}{l} \mu X.P \vee [a]X \\ \nu Y.Q \wedge \langle b\rangle Y \end{array}$$

Henkin modal mu-calculus and beyond

$[\cdot]$ $\langle\rangle$ etc. are just modal operators on global states, and it is trivial to add them to modal mu-calculus to get a temporal logic, which subsumes and extends the ATL of Alur, Henzinger and Kupferman.

But this is boring – and not the right sort of expressive. We want to allow all sorts of other things to happen between the $[]$ and the $\langle\rangle$.

Would like to write things like $[a]\langle b\rangle\phi \otimes [a]\langle b\rangle\psi$, or

$$\otimes \begin{matrix} \mu X.P \vee [a]X \\ \nu Y.Q \wedge \langle b\rangle Y \end{matrix}$$

or even

$$\mu Z.R \vee \left(\otimes \begin{matrix} \mu X.P \vee [a]Z \\ \nu Y.(Q \vee Z) \wedge \langle b\rangle Y \end{matrix} \right)$$

What would this mean? *Back to basics . . .*

Independence-friendly modal logic

To get a firm semantic ground, go back to IF logic à la Hintikka, and use it as a meta-language. Recall that IF logic has $\forall x.\exists y/x.\phi$, meaning y is to be chosen independently of x.

We produce an independence-friendly version of modal logic by tagging modalities to allow the slash notation:

$$[a]_\alpha \langle b \rangle_\beta [c]_{\gamma/\alpha,\beta} \langle d \rangle_{\delta/\alpha,\beta} \Phi$$

Independence-friendly modal logic

To get a firm semantic ground, go back to IF logic à la Hintikka, and use it as a meta-language. Recall that IF logic has $\forall x.\exists y/x.\phi$, meaning y is to be chosen independently of x.

We produce an independence-friendly version of modal logic by tagging modalities to allow the slash notation:

$$[a]_\alpha \langle b \rangle_\beta [c]_{\gamma/\alpha,\beta} \langle d \rangle_{\delta/\alpha,\beta} \Phi$$

What do we mean in a modal logic by requiring a choice at one point to be independent of a choice at an earlier point?

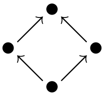
IFML on transition systems

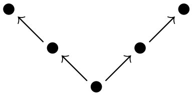
Tero Tulenheimo has studied IFML on transition systems: $[]_\alpha \langle\rangle_{\beta/\alpha}$ means literally the successor state chosen at β must be chosen uniformly in the successor state chosen at α.

IFML on transition systems

Tero Tulenheimo has studied IFML on transition systems: $[]_\alpha \langle\rangle_{\beta/\alpha}$ means literally the successor state chosen at β must be chosen uniformly in the successor state chosen at α.

It turns out that this IFML can express few non-ML properties: essentially only that of a state being a common successor. E.g: $[]_\alpha \langle\rangle_{\beta/\alpha}\mathrm{tt}$ is true of the left-hand model below because the apex of the diamond can be uniformly chosen as a second step regardless of whether the first step went left or right, but not of the bisimilar right-hand model.





This is not very powerful – and no property like $\genfrac{}{}{0pt}{}{[]}{[]}\genfrac{}{}{0pt}{}{\langle\rangle}{\langle\rangle}$ is expressible.

IFML on concurrent systems

In a concurrent or distributed setting, such as Petri nets, process algebras, event structures etc., $[]_\alpha\langle\rangle_{\alpha/\beta}$ makes good sense if an *event* or *local state* is chosen at β and is *concurrent* or *disjoint* from that at α – it is not only possible, but natural to choose β without knowing α.

IFML on concurrent systems

In a concurrent or distributed setting, such as Petri nets, process algebras, event structures etc., $[]_\alpha \langle\rangle_{\alpha/\beta}$ makes good sense if an *event* or *local state* is chosen at β and is *concurrent* or *disjoint* from that at α – it is not only possible, but natural to choose β without knowing α.

Has NP power – $[]_\alpha \langle\rangle_\beta []_\gamma \langle\rangle_{\delta/\alpha,\beta}$ is equivalent to $\begin{smallmatrix} [] & \langle\rangle \\ [] & \langle\rangle \end{smallmatrix}$, so can code Henkin quantifier.

IFML on concurrent systems

In a concurrent or distributed setting, such as Petri nets, process algebras, event structures etc., $[]_\alpha \langle\rangle_{\alpha/\beta}$ makes good sense if an *event* or *local state* is chosen at β and is *concurrent* or *disjoint* from that at α – it is not only possible, but natural to choose β without knowing α.

Has NP power – $[]_\alpha \langle\rangle_\beta []_\gamma \langle\rangle_{\delta/\alpha,\beta}$ is equivalent to $\begin{smallmatrix}[] & \langle\rangle \\ [] & \langle\rangle\end{smallmatrix}$, so can code Henkin quantifier.

This gives a semantics to IFML which encompasses Henkin ML – and extends it.

IF modal mu-calculus

Recall the trump semantics for IF logic. This is a normal Tarskian semantics over a lattice. So we can invent a fixpoint extension IF-LFP of IF logic. (A talk in itself.)

IF modal mu-calculus

Recall the trump semantics for IF logic. This is a normal Tarskian semantics over a lattice. So we can invent a fixpoint extension IF-LFP of IF logic. (A talk in itself.)

Then we can use IF-LFP to give a semantics to IF modal mu-calculus.

IF modal mu-calculus

Recall the trump semantics for IF logic. This is a normal Tarskian semantics over a lattice. So we can invent a fixpoint extension IF-LFP of IF logic. (A talk in itself.)

Then we can use IF-LFP to give a semantics to IF modal mu-calculus.

Formulae like

$$[]_\alpha \mu Z.\langle\rangle_{/\alpha} Z \vee \langle\rangle \phi$$

now make sense ('whatever you do, I can get to ϕ, though I may first have to do a bunch of things that don't depend on what you did'); or

IF modal mu-calculus

Recall the trump semantics for IF logic. This is a normal Tarskian semantics over a lattice. So we can invent a fixpoint extension IF-LFP of IF logic. (A talk in itself.)

Then we can use IF-LFP to give a semantics to IF modal mu-calculus.

Formulae like

$$[]_\alpha \mu Z. \langle\rangle_{/\alpha} Z \vee \langle\rangle \phi$$

now make sense ('whatever you do, I can get to ϕ, though I may first have to do a bunch of things that don't depend on what you did'); or

$$\nu Z(\alpha).[]_\alpha Z(\alpha) \wedge \langle\rangle_{/\alpha} \phi$$

'I can always get to ϕ with the same action, regardless of what you do'.

IF modal mu-calculus is in its infancy. It makes most sense interpreted on true concurrent structures such as coherent event structures.

IF-LFP is horrifically powerful – how expressive is IF-μ? (But they're still decidable to verify on finite systems (phew!).)

IF modal mu-calculus is in its infancy. It makes most sense interpreted on true concurrent structures such as coherent event structures.

IF-LFP is horrifically powerful – how expressive is IF-μ? (But they're still decidable to verify on finite systems (phew!).)

More importantly: are there real uses for it? I've suggested it has a natural link with distributed systems such as MAS. Does it?

Thanks

to all those who have talked with me (or let me talk at them) about this and related areas - in particular, in reverse chronological order:

Francien Dechesne, Merlijn Sevenster, Tero Tulenheimo, Carla Delgado, Stephan Kreutzer, Gabriel Sandu, Juliana Küster Filipe, Mike Wooldridge, Michael Fisher, Sibylle Fröschle.

For more on many aspects, see Ph.D. theses of Tulenheimo, Dechesne and Sevenster (and of Ahti Pietarinen).