Research Ideas

October 10, 2017

1 Ideas

1. Learn B-automata ala Angluin.
2. Angluin learning for graph grammars. Applications to NLP?
3. Find useful quantitative LTL formalism and extend to SL. e.g, Orna's quality, Demri's constraint LTL
4. Give logic to talk about what agents believe/assume about other agent strategies. e.g., x believes that y will make a best-effort. think in terms of classes of strategies (defined in some expressive logic).
5. Reinvent SL... restrict to certain structures, e.g., transition models from planning (i.e., states are evals of vars), and reachability objectives.
6. iCGS with public actions and CGS are similar. Should be able to stretch all computational results from CGS to iCGS with public actions. E.g., results about randomised strategies, about computing NE, etc.
7. Find logic that talks about non-idealised agents. E.g., agents in the wild behave in certain ways, they have certain behaviours, and perhaps these behaviours, once learned, can be talked about.
8. Recall dfn of "Action-LTL" and do synthesis. The idea is simply to be able to express actions more naturally. E.g., can one now express prescriptive actions?
9. can logic help learn agent behaviours?
10. LTL[F] finite traces? functions in arbitrary algebras? $f : A^k \to A$, atoms map to constants... WHAT IS THE USE? process mining? runtime verification? PTIME
11. MSO(T) to LTL(T)?
12. alternating automata with classes of boolean functions (rather than all boolean functions); e.g., sat formula? "exists $x_1, x_2 : phi(x_1, x_2, q_1, ..., q_n)$?
13. the following problems are very similar
    - control flow synthesis (lustig et al)
    - regular open APIs

- single entry exit petri nets = components = transducers with exit states
- read "Network-Formation Games with Regular Objectives" by Avni-Kupferman-Tamir, 2014

14. study modal automata where the modal operators have different meanings. e.g., satisfy different axioms, "there exists a path st", "there exists a coloring of the subtree such that...", "for some $\sim$-related node" (bastien)...
15. modal automata on graphs.
16. Have a SL whose MC procedure returns memoryless strategies if these exist. e.g., $ATL << A >> G << Ag >> Fp$ does not necc. return a memless strategy. In SL, at least you get a finite state strategy. Looks like the usual MC procedure for (iterate fixpoints) mu-calculus gives a memless str.

2 Wild Ideas

1. network of majority games... equilibria
2. nested robots
3. logic of neural networks
4. distributed synthesis with message loss
5. better complexity for BC systems; logic of BC systems (axiomatising epistemic formulas)
6. An evolutionarily stable strategy (ESS) is a strategy which, if adopted by a population in a given environment, cannot be invaded by any alternative strategy that is initially rare.
7. SL which returns simplest strategies
8. language for expressing strategies explicitly. principled way to combine strategies.
9. model iterated eliminitation of dominated strategies by PACGS
10. iCGS with public actions and CGS are similar. Should be able to stretch all computational results from CGS to iCGS with public actions. E.g., results about randomised strategies, about computing NE, etc.
11. Find logic that talks about non-idealised agents. E.g., agents in the wild behave in certain ways, they have certain behaviours, and perhaps these behaviours, once learned, can be talked about. can logic help learn agent behaviours? See Nick Jenning's work, e.g.,
12. LTL[F] finite traces? functions in arbitrary algebras? $f : A^k \to A$, atoms map to constants... WHAT IS THE USE? process mining? runtime verification? PTIME

13. MSO(T) to LTL(T)?

14. alternating automata with classes of boolean functions (rather than all boolean functions); e.g., sat formula? "exists $x_1, x_2 : phi(x_1, x_2, q_1, ..., q_n)$?

15. the following problems are very similar

    - control flow synthesis (lustig et al)
    - regular open APIs
    - single entry exit petri nets = components = transducers with exit states
    - read "Network-Formation Games with Regular Objectives" by Avni-Kupferman-Tamir, 2014

16. study modal automata where the modal operators have different meanings. e.g., satisfy different axioms, "there exists a path st", "there exists a coloring of the subtree such that...", "for some $\sim$-related node" (bastien)...

17. modal automata on graphs.

18. Have a SL whose MC procedure returns memoryless strategies if these exist. e.g., $ATL << A >> G << Ag >> Fp$ does not necc. return a memless strategy. In SL, at least you get a finite state strategy. Looks like the usual MC procedure for (iterate fixpoints) mu-calculus gives a memless str.