What am I doing?

Tran Cao Son

Current Interests

• Logic programming (Answer Set Programming)
• Planning
Answer Set Programming – Idea

• *Given*: problem P
• *Solution*: Encoded P as program Q(P) such that answer sets of Q(P) are solutions of P

Answer Set Programming

• *Planning*
• *Extension*
• *Debugging and ASP Programming environment*
Answer Set Planning

• Can ASP-based planners be competitive with other approaches? Yes/No
  – How to achieve scalability?
    • Interleaving between grounding and answer set computation (difficult!)
  – How to take advantage of heuristics?
  – Attempts
    • Domain dependent knowledge (encoding cumbersome due to lack of list constructor, e.g. the list a1; a2; a3 needs to be encoded by the set seq(x), seq(y), head(x, a1), tail(x, y), head(y, a2), tail(y, a3))
    • Approximation reasoning (incompleteness)

Extension to ASP

• Goal: making ASP a good knowledge representation language
• What is needed?
  – Aggregates (semantics/implementation)
  – List constructor (or limited list constructor)
Debugging & Programming Environment

• **Goal:**
  – understanding ASP programs
  – providing ASP programmers a tool for testing/debugging their programs

• **Questions:**
  – Why does a program not have answer sets?
  – Why is an atom \( a \) present in/absent from answer set \( S \)?

Planning – Requirement

• **Given:** planning problem \( P \) (action theory \( A \), initial state \( I \), goal \( G \))

• **Goal:** a sequence of actions to change state of the world from \( I \) to \( G \)

• **Problems:**
  – initial state \( I \) might be incomplete
  – actions might have duration
  – goals might have deadline
Planning

• *Goal*: algorithms for planning in
  – Conformant planning
  – Planning in real-world applications (with resources, actions with duration, time constraints)

• *Questions*:
  – Can Graphplan idea be used?
  – What will be a good heuristic for planning domains with static causal laws?
  – Can parallelism be of help?

Graphplan Idea

• Graphplan structure
  – simple
  – can be generated efficiently
  – defined for “simple” planning problems (complete information, actions without duration, no resources, etc.)
  – is the source for deriving heuristics
Planning

• **Goal:** algorithms for planning in
  – Conformant planning
  – Planning in real-world applications (with resources, actions with duration, time constraints)

• **Questions:**
  – Can Graphplan idea be used?
  – What will be a good heuristic for planning domains with static causal laws?
    • There exists no heuristics for this situation
  – Can parallelism be of help?