"Domain-independent" planning and "Domain-dependent" planning

Le Meilleur est l'ennemi du bien.

The Best is the enemy of the good.

Voltaire

Thanks to Dana Nau….

Domain dependent planners?

- For many applications, domain dependent techniques are still critical
  - Exploit domain features for efficiency
  - Avoid the limitations of PDDL
  - Control the types of plan output
- Planning is at least partly an engineering discipline, and domain-independent planning isn't (yet) the best solution for most domains
- Some examples
  - NASA Europa and Aspen applications
  - Uninhabited Air Vehicles
  - Amada sheet metal bending
  - Bridge Baron

Domain dependent planners?

- Many "domain dependent planners" aren't
  - TLPlan, TALplan
  - SHOP2
  - Europa
  - Aspen
- Techniques used in these planners diverge

How do we incorporate domain-dependent techniques?

- We need to be able to study domain-dependent techniques.
- How do we empirically study these planners and their heuristics?
  - Comparison and ablation studies
- A certain amount of apples and oranges comparison is unavoidable.
  - Including comparing these techniques against domain-independent planners.

Why should domain-independent researchers care?

- Performance of domain-dependent techniques provides challenges to improve performance of d-i planners
- Identify semi-specific techniques for classes of problems
  - E.g., heuristics that are useful for domains involving motion in 2-space…
- Provide new challenges for expressive power of PDDL
  - State features, optimization criteria, etc.

PDDL is not a natural phenomenon & the IPC is not an application

- Domain independent planners claim not to use domain knowledge.
- But it takes a great deal of domain knowledge and cleverness about planners to write a good PDDL domain definition.
  - E.g., J. Hoffmann, *et al.* (2006) ``Engineering Benchmarks for Planning: the Domains Used in the Deterministic Part of IPC-4", *JAIR*, 26, 453--541.
- We don't trip over PDDL domains in the wild.
- Many are reverse-engineered from applications
- The distinction between domain-dependent and independent is fuzzier than one might think….