

# Transfer Learning Revisited

Joseph Xu

Work also by Yongjia Wang, Sam Wintermute

University of Michigan

Soar Workshop 28

# Outline

- *Review work from previous year's transfer learning talk*
  - Transfer learning
  - Evaluation environments
  - Indicator heuristic
- Work done after last talk
  - Source-target mapping
  - Results
- Discussion

# Transfer Learning

- Working on a source problem improves learning or performance on target problem
  - Transfer case: work on source problem before working on target
  - No Transfer case: work on target without seeing source

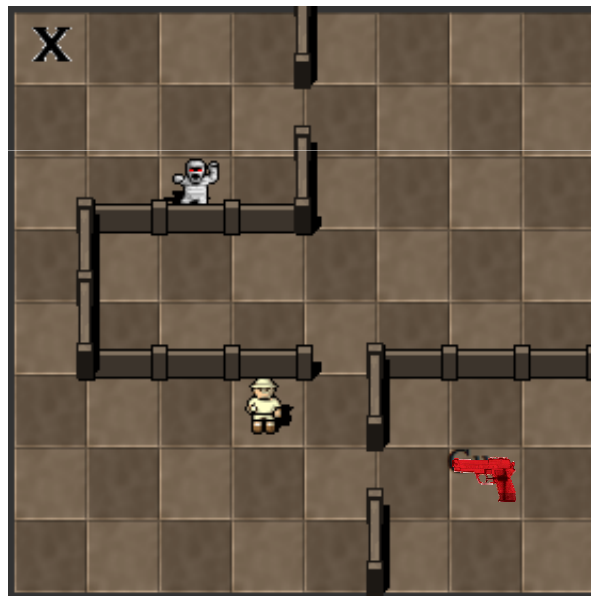
$$\text{regret} = \frac{ntp - tp}{\max(ntp, tp)}$$

*ntp* = No Transfer Case Performance

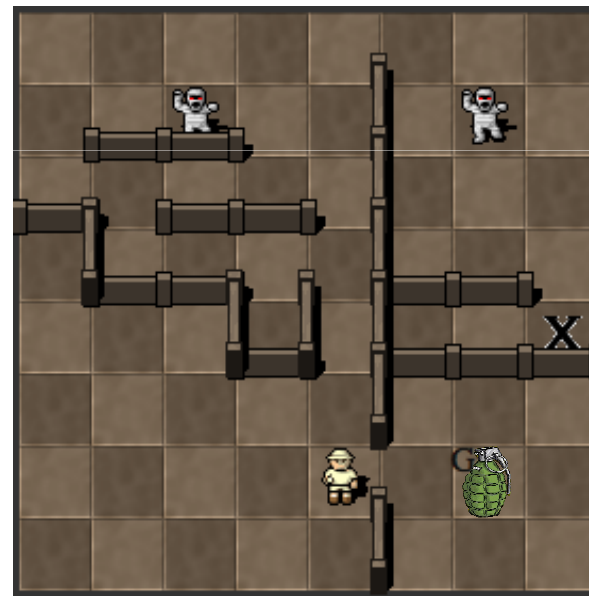
*tp* = Transfer Case Performance

# Environment

- All problems are deterministic, discrete, finite games in General Game Player



Source: Explorer must kill mummy with gun



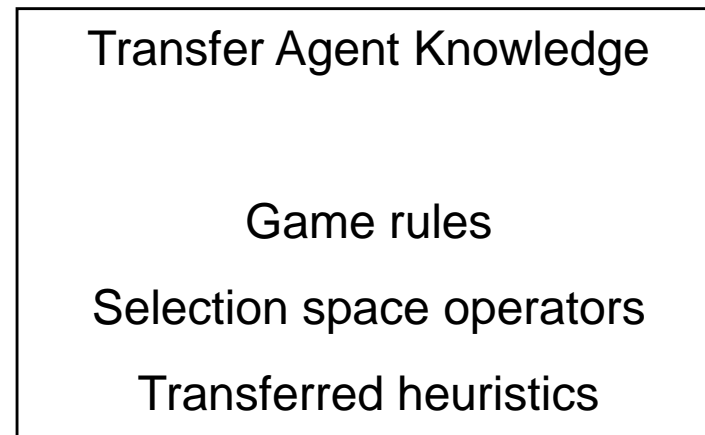
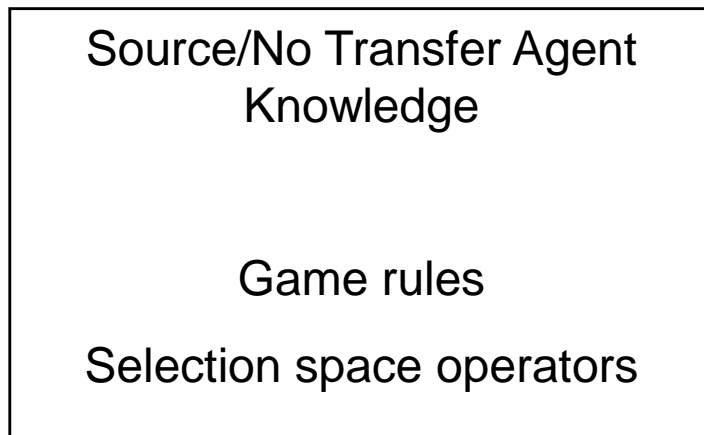
Target: Explorer must kill mummies with grenade

# Environment

- 4 game domains:
  - Escape
  - Wargame
  - Rogue
  - Build
- Sequence of levels categorizing types of transfer
  - Levels 6-9 within-domain transfer
  - Level 9 is automatic obfuscation of symbol names
  - Levels 10, 11 cross-domain transfer

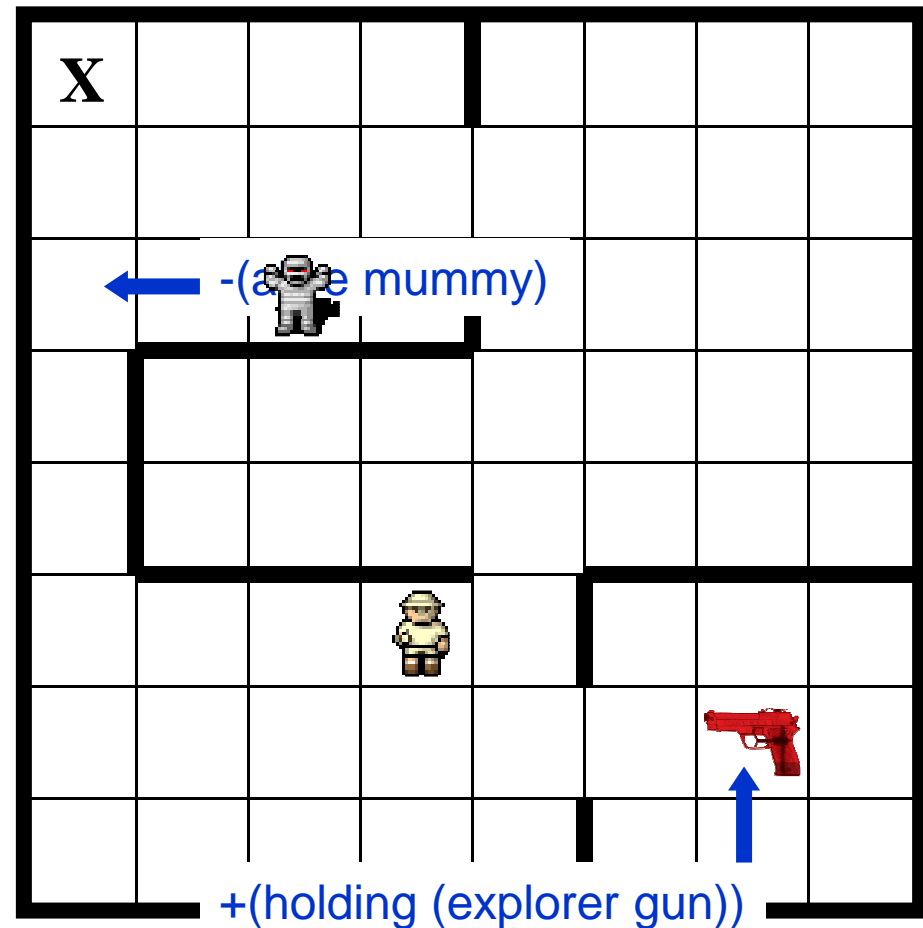
# Transferring Heuristics

- Automatically translate game rules into Soar productions
- Use selection space operators to do look-ahead iterative deepening
- Learn heuristics in source game, transfer to target game



# Indicators

- Assume that state changes in solution path are conducive to winning game
- Transfer involves finding analogous state changes in the target
- Increase search depth on paths where indicators occur in target problem solving



# Outline

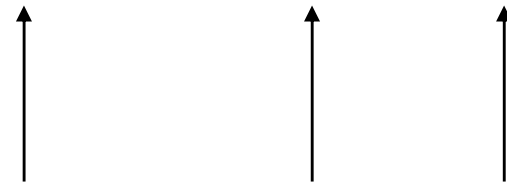
- Review work from previous year's transfer learning talk
  - Transfer learning
  - Evaluation environments
  - Indicator heuristic
- *Work done after last talk*
  - Source-target mapping
  - Results
- Discussion



# Mapping Indicators

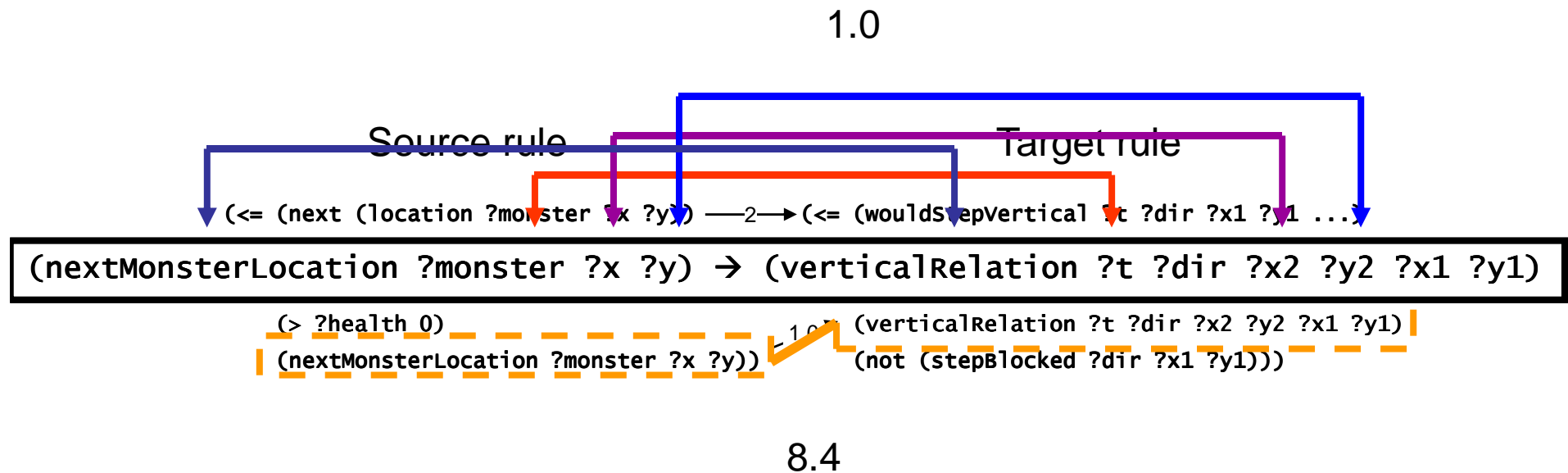
- Hard to determine semantic analogies between source and target symbols
- Simple syntactic “diff”-like alignment
- Break into 2 stages
  - Mapping predicates
  - Mapping constants

(holding explorer gun)



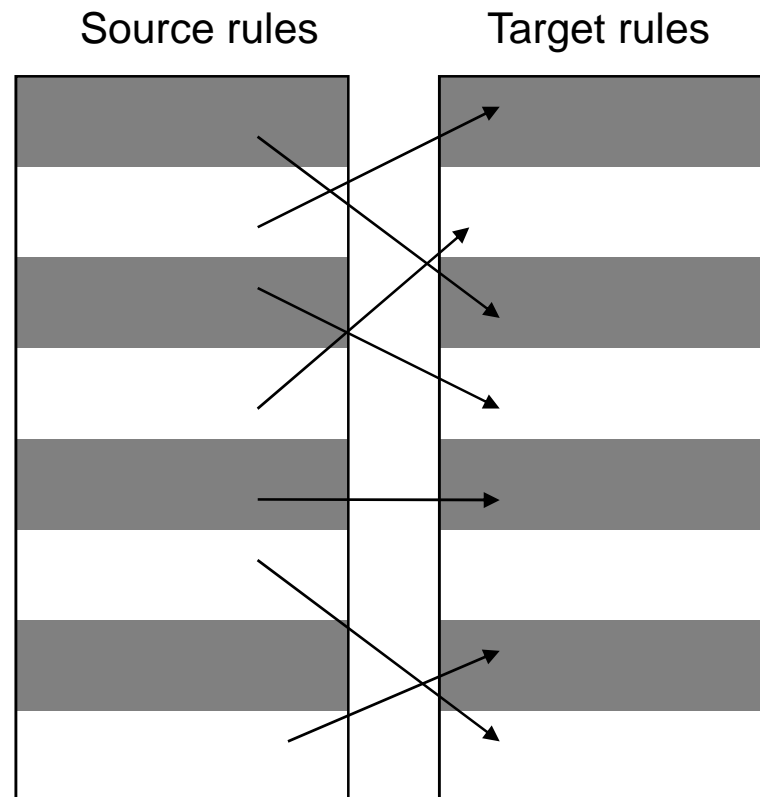
# Mapping Predicates

- Greedy algorithm to maximize structural overlap of rules

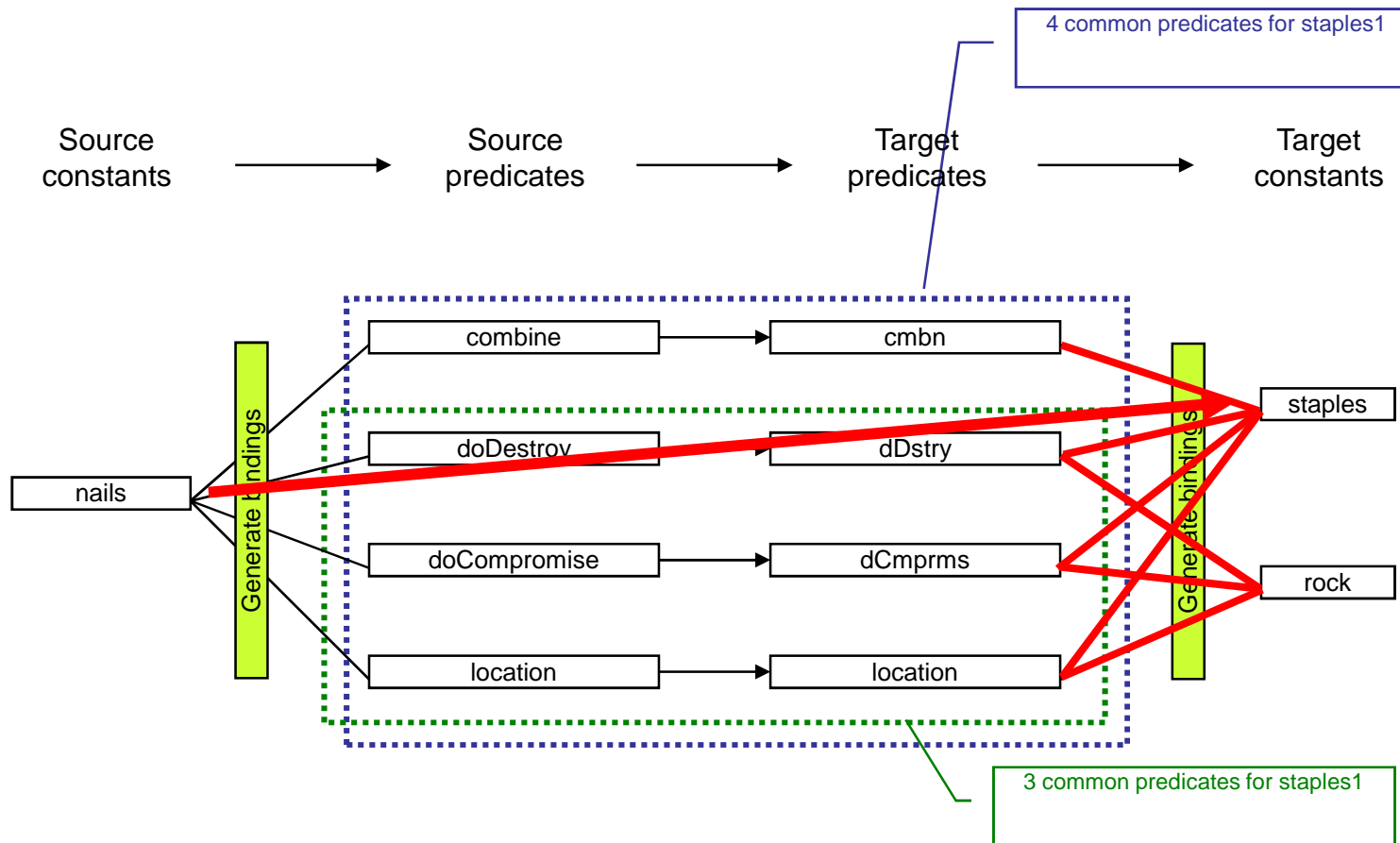


# Mapping Predicates

- Greedy algorithm to maximize structural overlap of rules



# Mapping Constants



# Summary of Approach

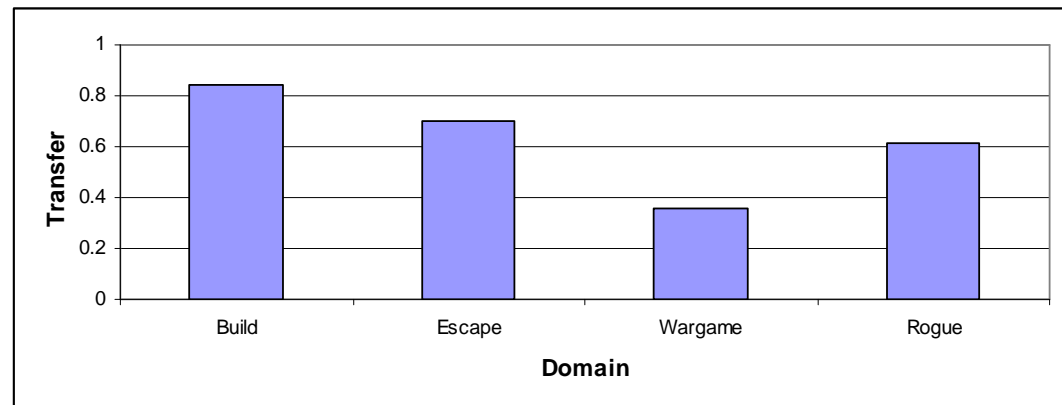
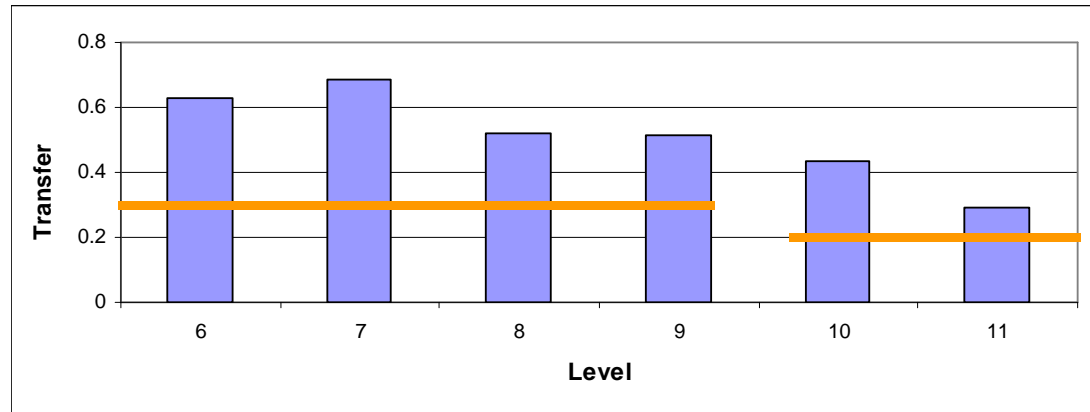
- Transfer case
  1. Source game rules presented
  2. Agent solves source problem with uninformed search
  3. Predicates and constants that experience transitions on solution path saved as indicators
  4. Target game rules presented
  5. Source and target symbols are matched by mapping source rules onto target rules
  6. Agent timed as it solves target problem using transferred indicators as heuristics

# Summary of Approach

- No-Transfer case
  1. Target game rules presented
  2. Agent timed as it solves target problem with uninformed search

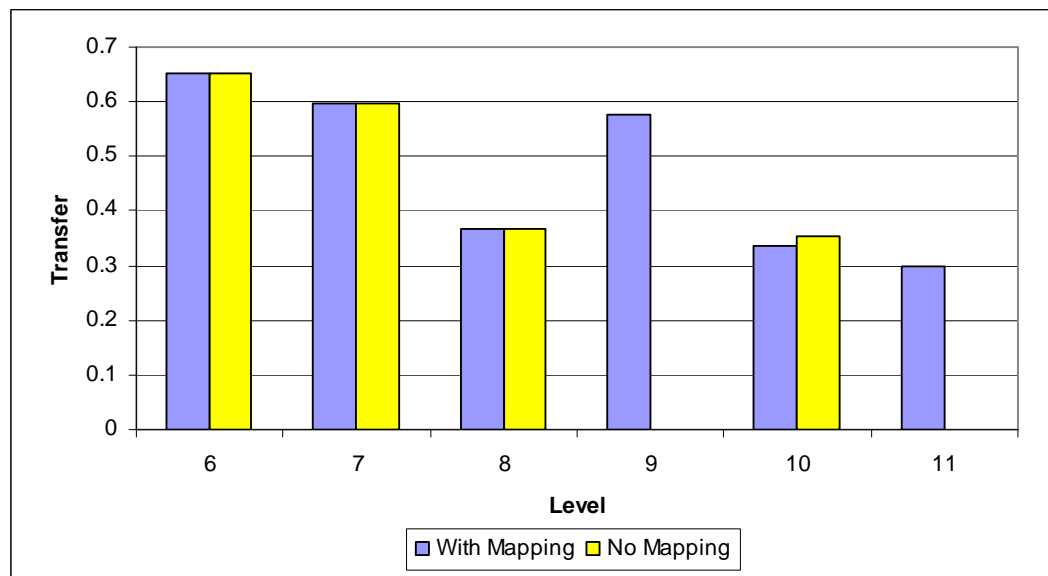
# Results

- 40 games spread over levels and domains
- Measured in Soar decisions



# Credit Assignment

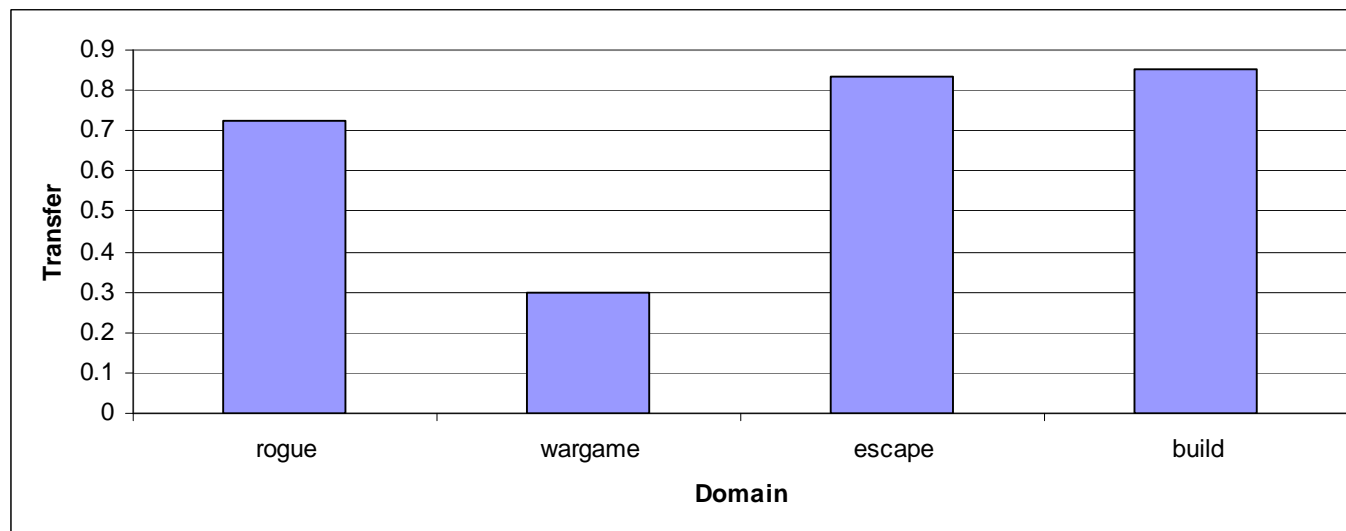
- Mapping
  - Many source-target pairs were similar enough such that mapping did not have to be accurate
  - Mostly perfect mapping of level 9
  - Hard to gauge success on other levels





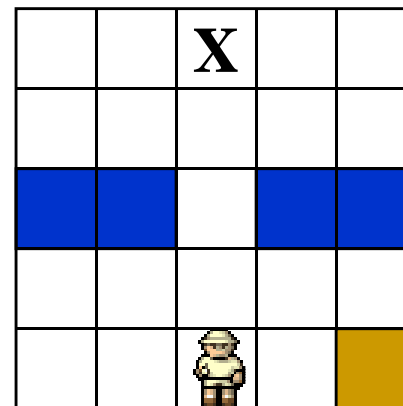
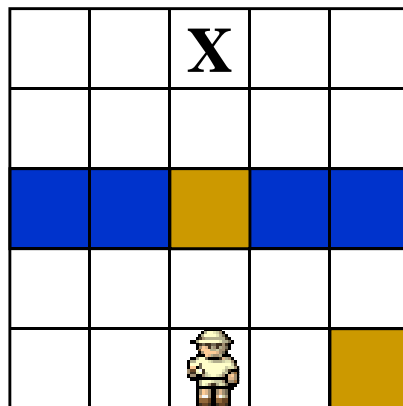
# Credit Assignment

- Heuristic
  - Level 9 source-target pairs with perfect mapping
  - No-transfer-case agent had bare minimum knowledge
  - Game rules + selection space search + very basic heuristics
  - Any transferred knowledge gets big improvements



# Generalizability

- Source knowledge that requires search is useful to transfer
- Cannot determine correctness of transferred knowledge without first solving target game
- No formal guarantee of correctness/desirability that can be generalized over a class of games



# Conclusion

- Nuggets
  - Developed agent that performed well on evaluations
  - Soar can solve any GGP game
- Coals
  - Can't make any strong claims about generalizability of results

# Hindsight

- GGP is too general
  - No formal relationships between sources and targets
  - Implicit constraints set on evaluation domains were ad hoc and hazy
  - No guarantees about generalizability of results beyond game domains tested
- To make meaningful progress
  - Change the source-target paired evaluation paradigm
  - Focus on formally constrained problems
  - Test over problem distributions instead of single instances

# Engineering Challenges

- Up to ...
  - 60000 WMEs – many multi-value attributes
  - 600 Rules
  - 2 million decisions
- Over 100 automatically generated agents