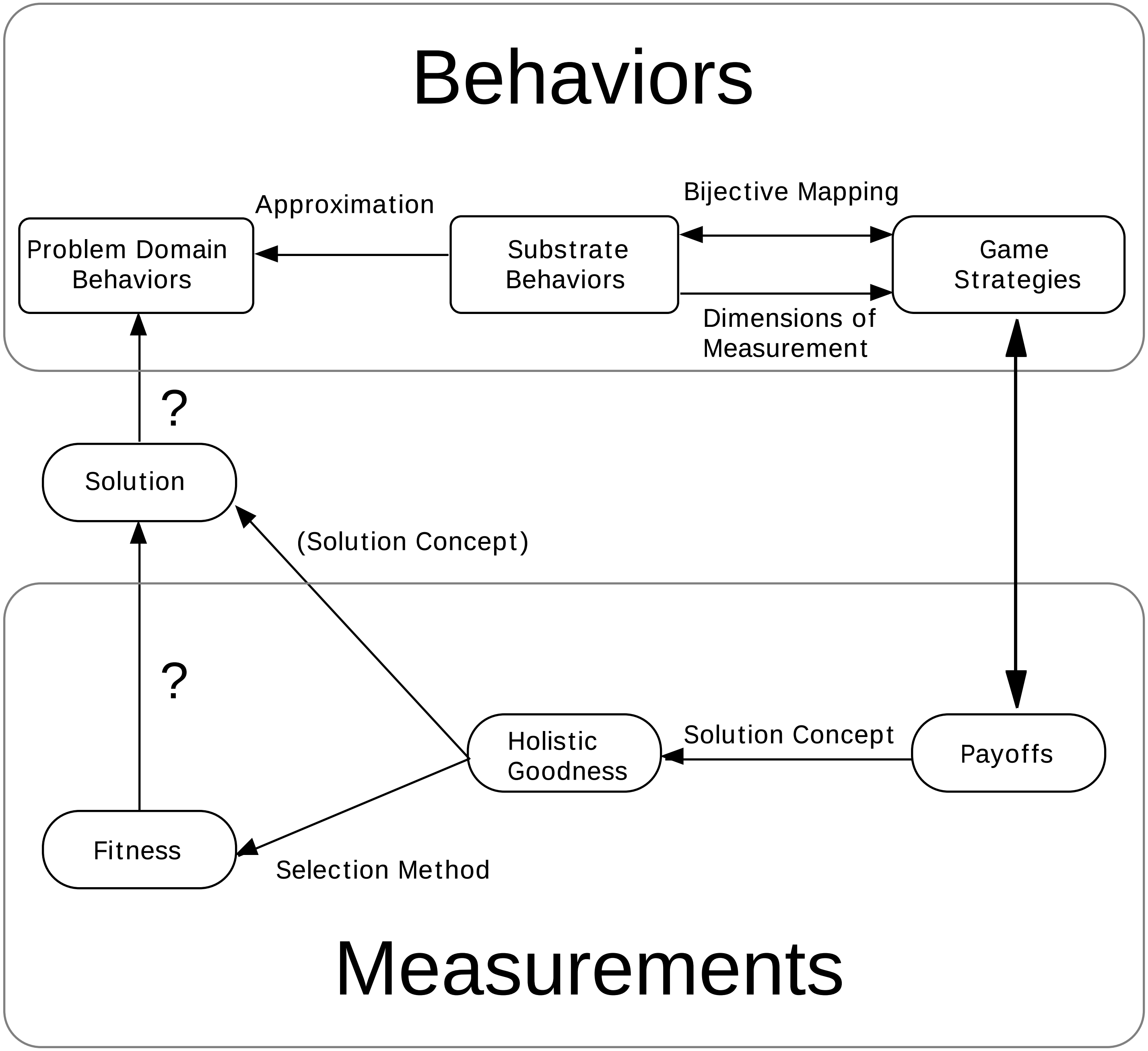
Solution Concepts and Monotonicity

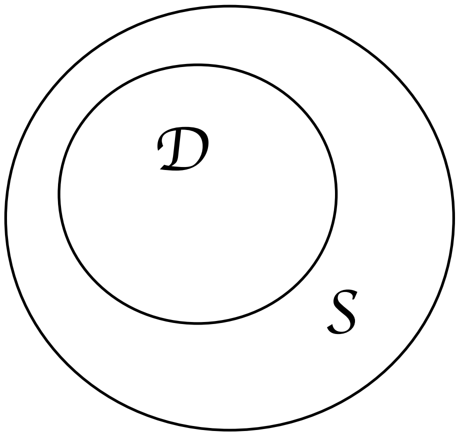
Sevan G. Ficici

Desired Algorithm Operation

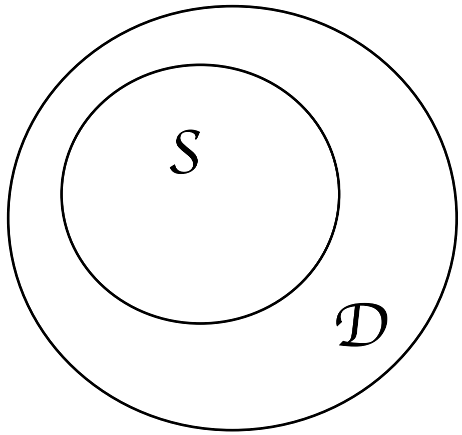
- Assume our coevolutionary algorithm never discards a strategy once discovered
- If we query the algorithm (at “appropriate” points) over time, we desire the solution returned at time t+1 to be no worse than that at time t
- We desire monotonic improvement
- (Elitism in ordinary EA does this)
- In coevolution, whether we get monotonic improvement depends on the solution concept used


Behaviors
Approximation
Bijective Mapping
Problem Domain Behaviors
Substrate Behaviors
Game Strategies
Dimensions of Measurement
?
Solution
(Solution Concept)
?
Holistic Goodness
Solution Concept
Payoffs
Fitness
Selection Method
Measurements

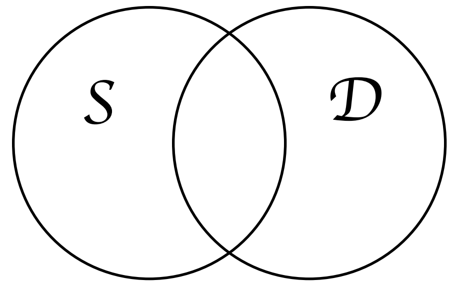
Domain and Substrate Behaviors



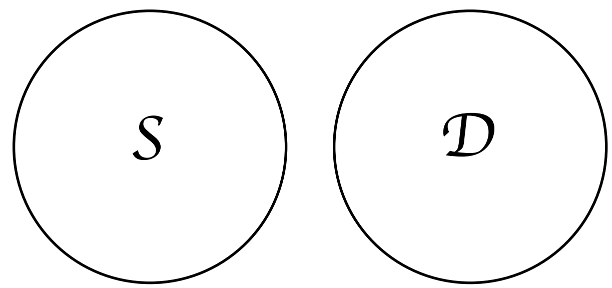
1) illegal strategies



2) solution may be unrepresented

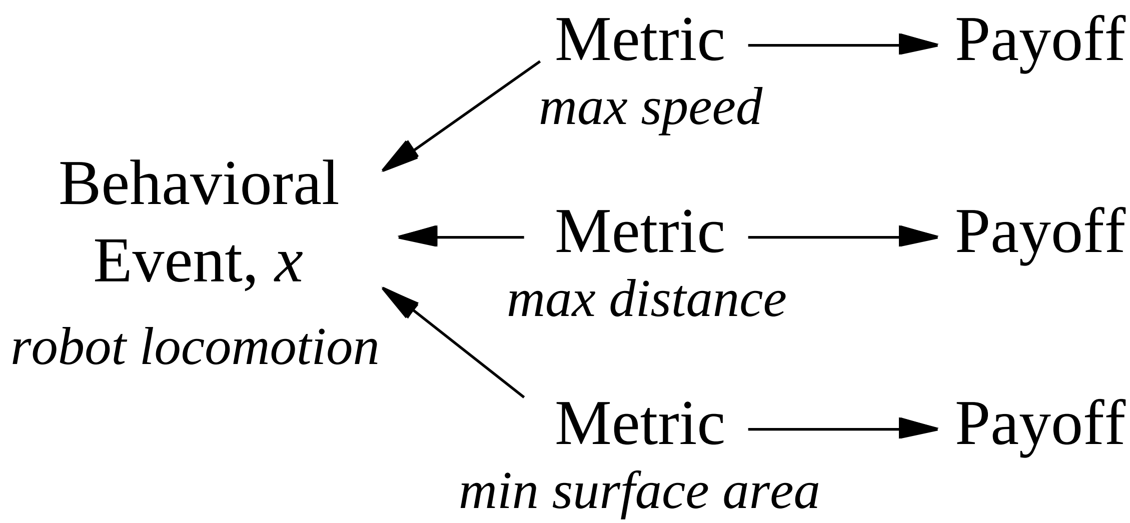


3) features of 1 and 2



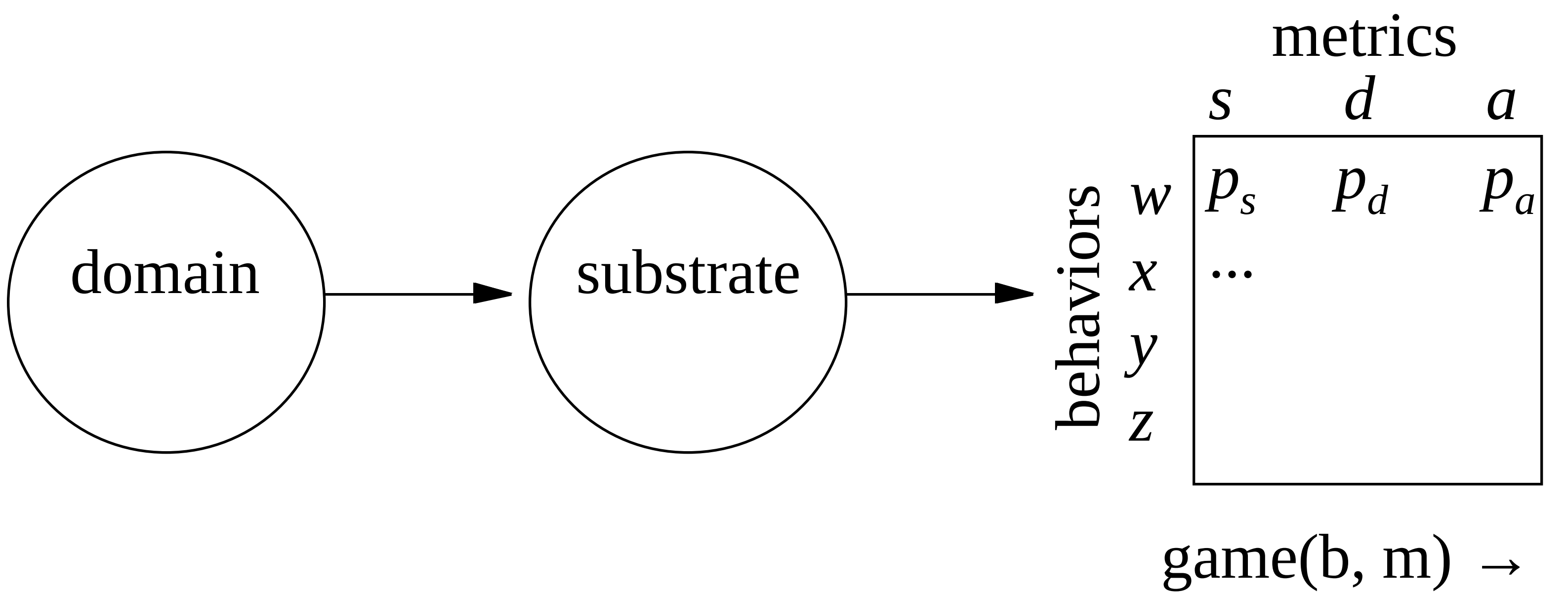
4) robot locomotion and tic-tac-toe

Measurement of Behavior



- *Payoff* indicates success of behavior w.r.t. a single metric of behavior; higher is better
- Payoffs are not utilities; utilities assume common currency of goodness

Game



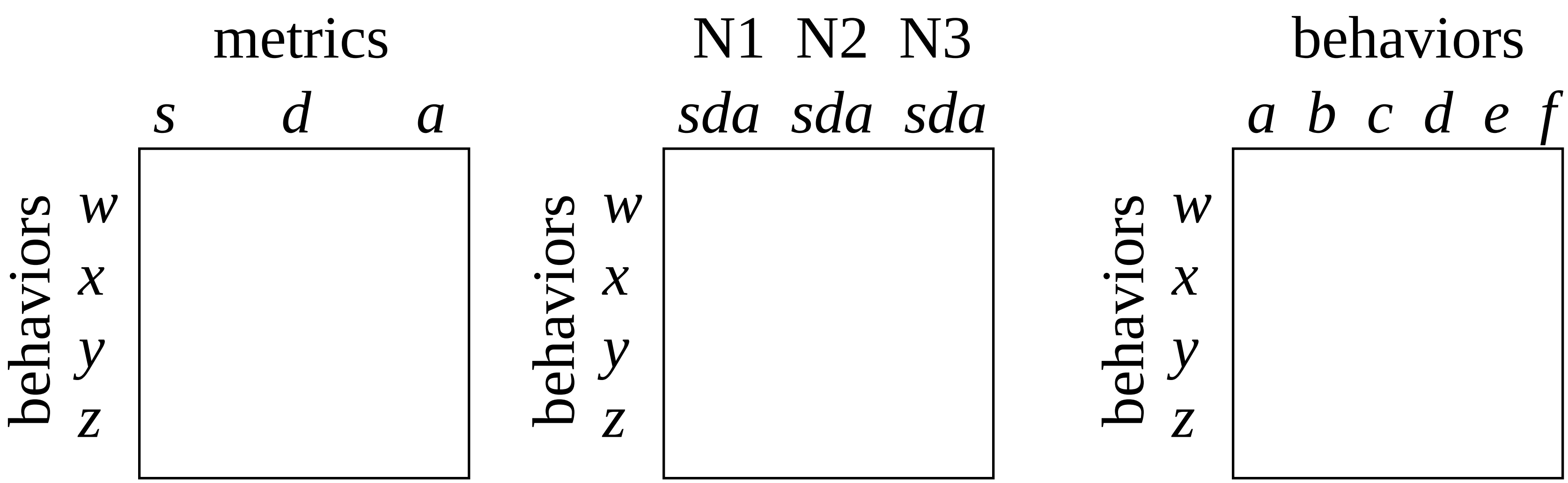
what we care about

what we observe

metrics we apply

- Each behavior is a *pure strategy* of the game, but may be mixed strategy of the domain

Game Types

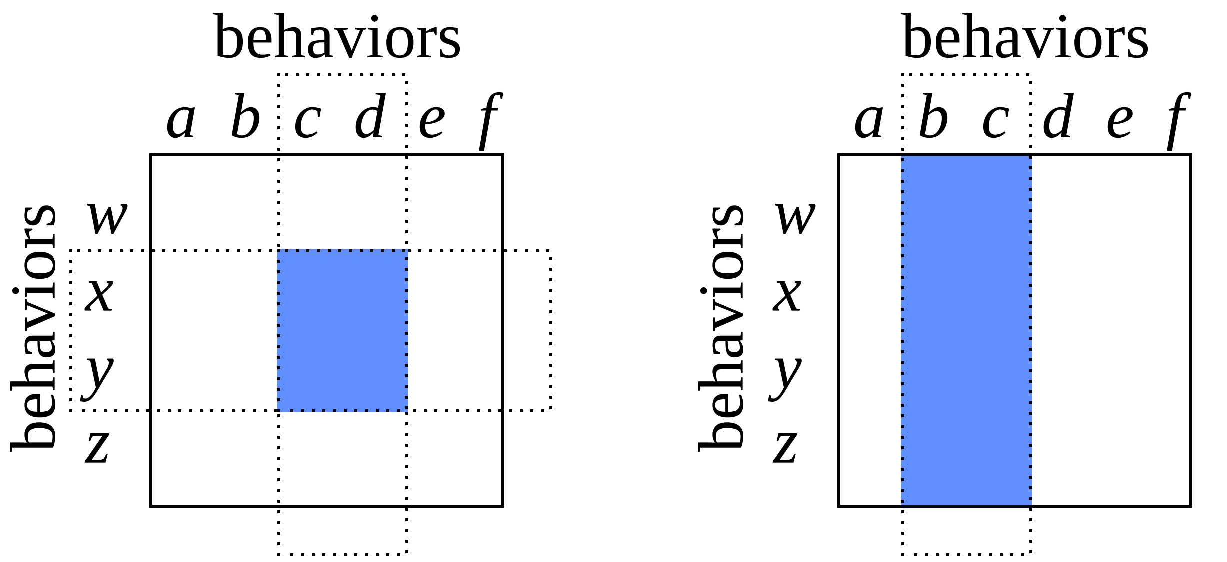


single-player game against nature

against multiple states of nature

multi-player game

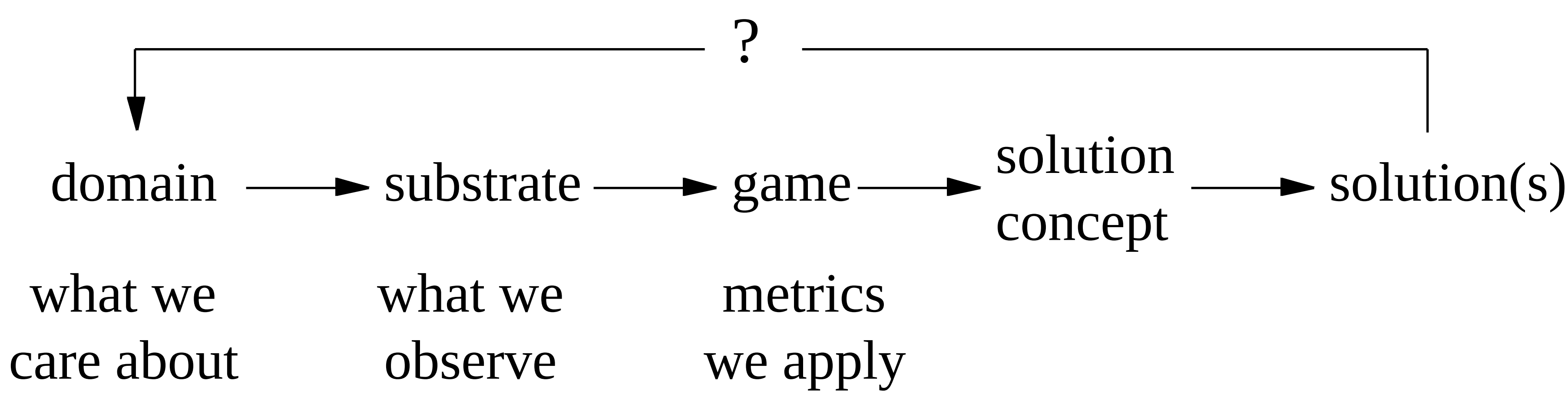
Sub-Game



behaviors
a b c d e f
behaviors
w
x
y
z
behaviors
a b c d e f
behaviors
w
x
y
z

Solution Concept

- Solution concept integrates over multiple metrics of goodness to give a holistic assessment of quality



Configuration and Solution

- Given *n*-player game, a *configuration* represents players' strategy choices:

  $\mathcal{K} = < s_1, s_2, \ldots, s_n >$ (note: s_i may be a set)
- According to solution concept, some configurations are *solutions*, denoted $\mathcal{K}^*$
- *Solution set* is set of solutions for game G according to solution concept O:

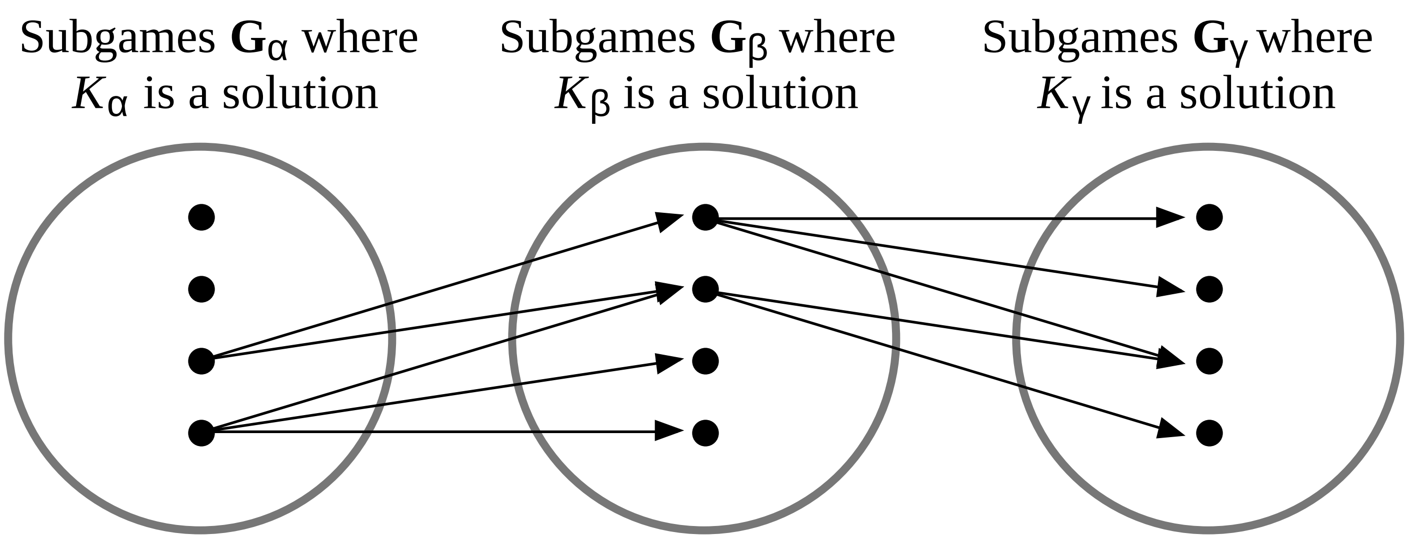
  $\mathcal{S}^*(G, O) = \{ \mathcal{K}^* \mid G, O \}$

Preference Relation I

- For a given game G, we prefer a solution $\mathcal{K}^*$ to a non-solution (configuration) $\mathcal{K}$
- We prefer configuration $\mathcal{K}_\alpha$ to $\mathcal{K}_\beta$ iff:

for each subgame G_β where K_β is the solution,

there exists a game G_α where K_α is the solution, and G_α G_β

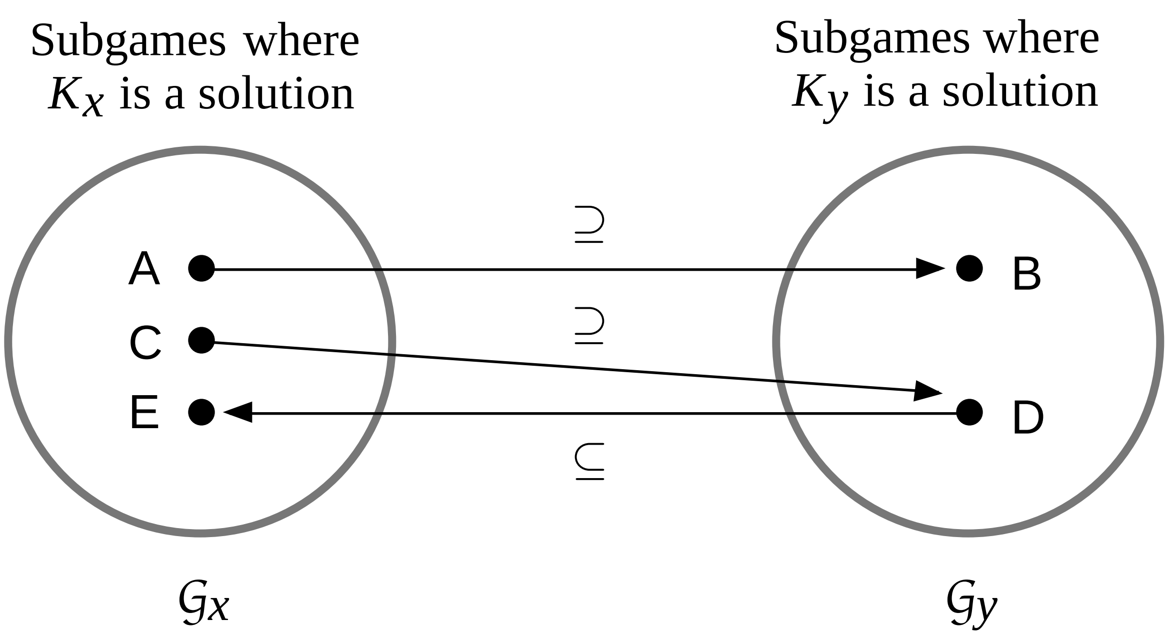


Preference Relation II

- Preference relation is transitive, asymmetric, ~reflexive
- Preference relation gives us a *poset* of configurations

Monotonicity

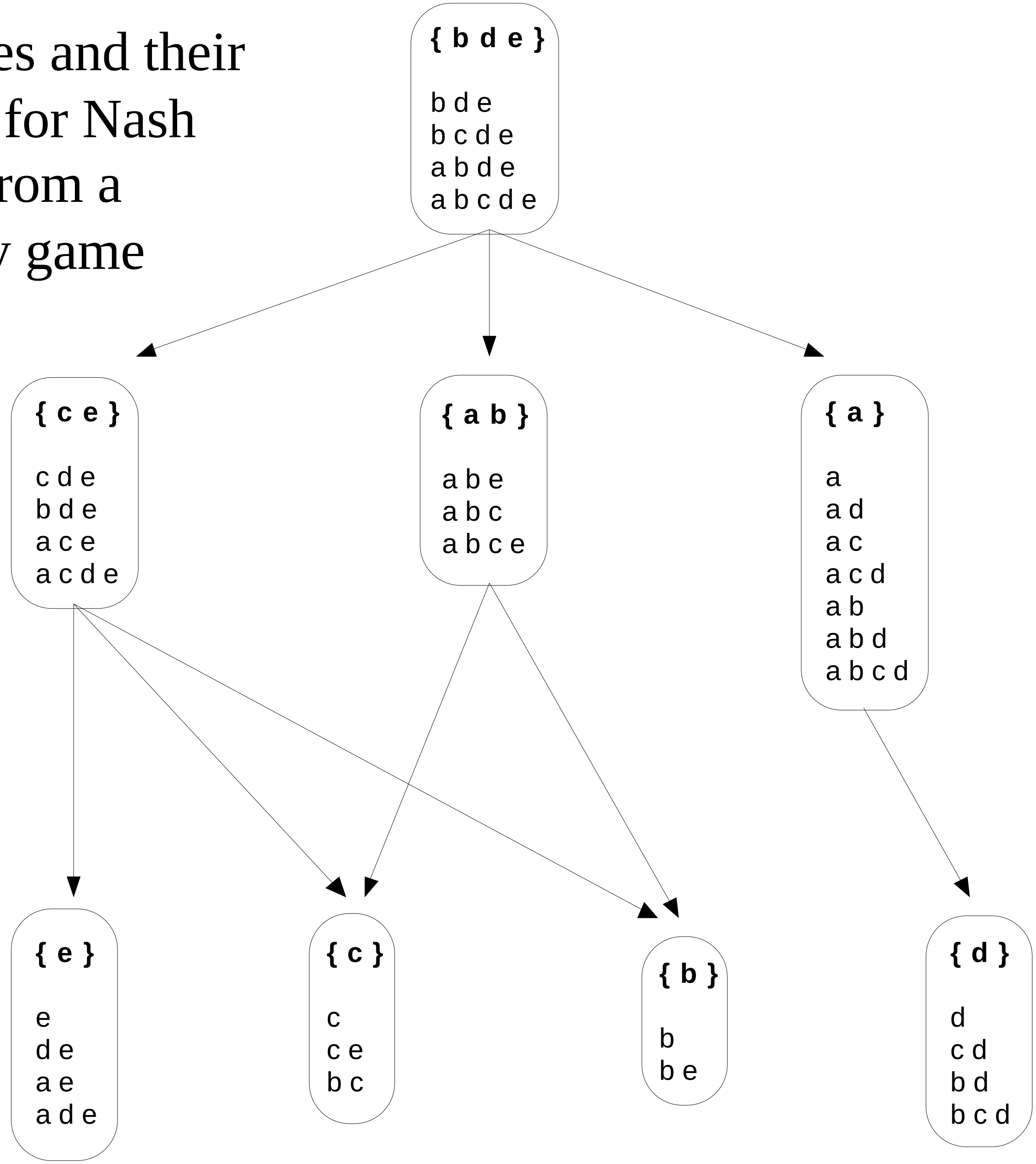
- We prefer $\mathcal{K}x$ to $\mathcal{K}y$, yet E is a subgame of D
- Given subgames α, β, γ, where α β γ
- If $\mathcal{K}$ is solution to α and γ, but not β, then solution concept is *not monotonic*; if all three, then monotonic



Solution Concepts & Monotonicity

- Monotonic concept guarantees that solution improves monotonically with time, when strategies are never discarded
- Nash equilibrium is a monotonic concept
- Non-dominated front is monotonic only if identical appearing strategies are not allowed on front
- “Best-scoring strategy” is not monotonic

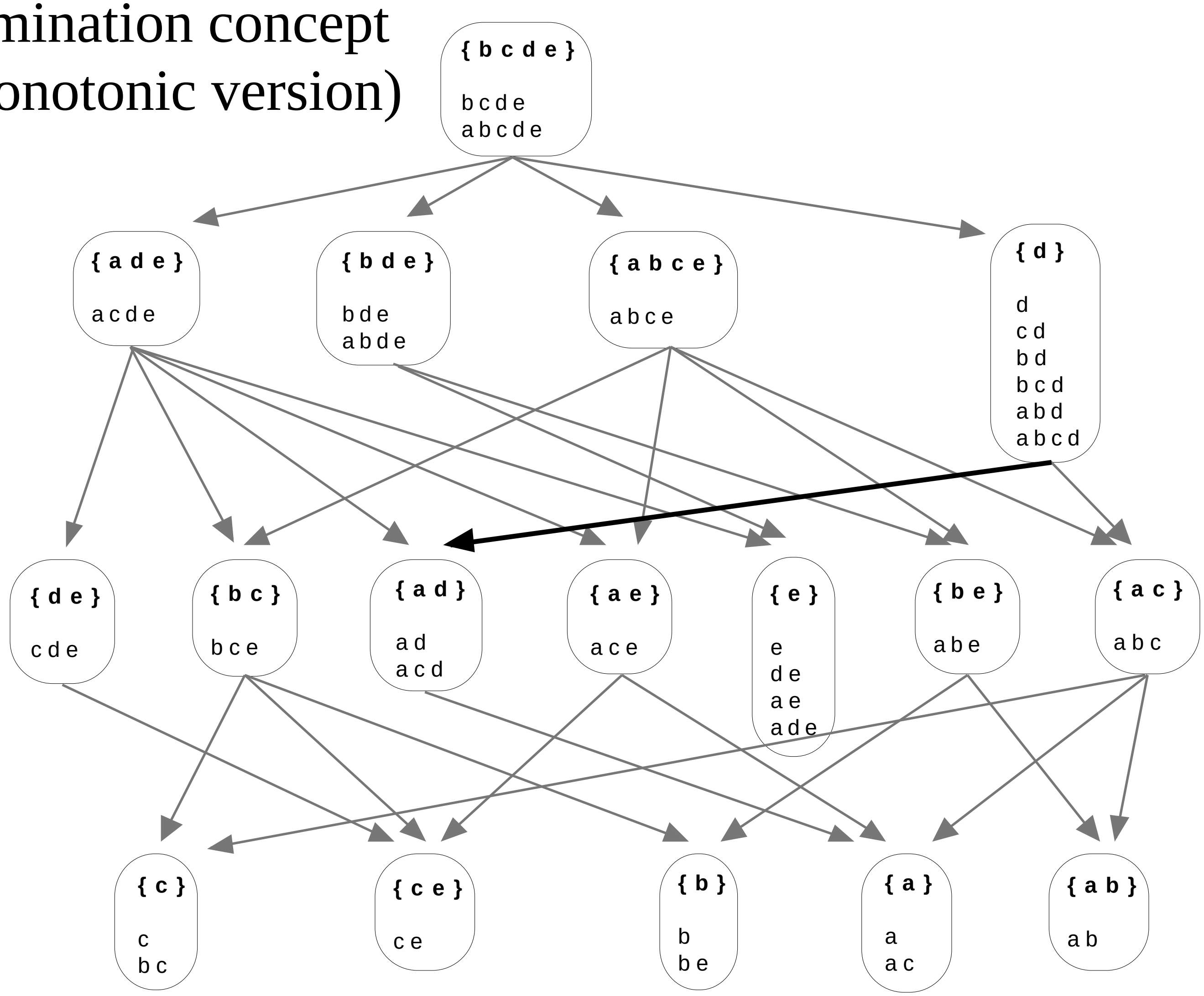
Sub-games and their solutions for Nash concept from a 5-strategy game

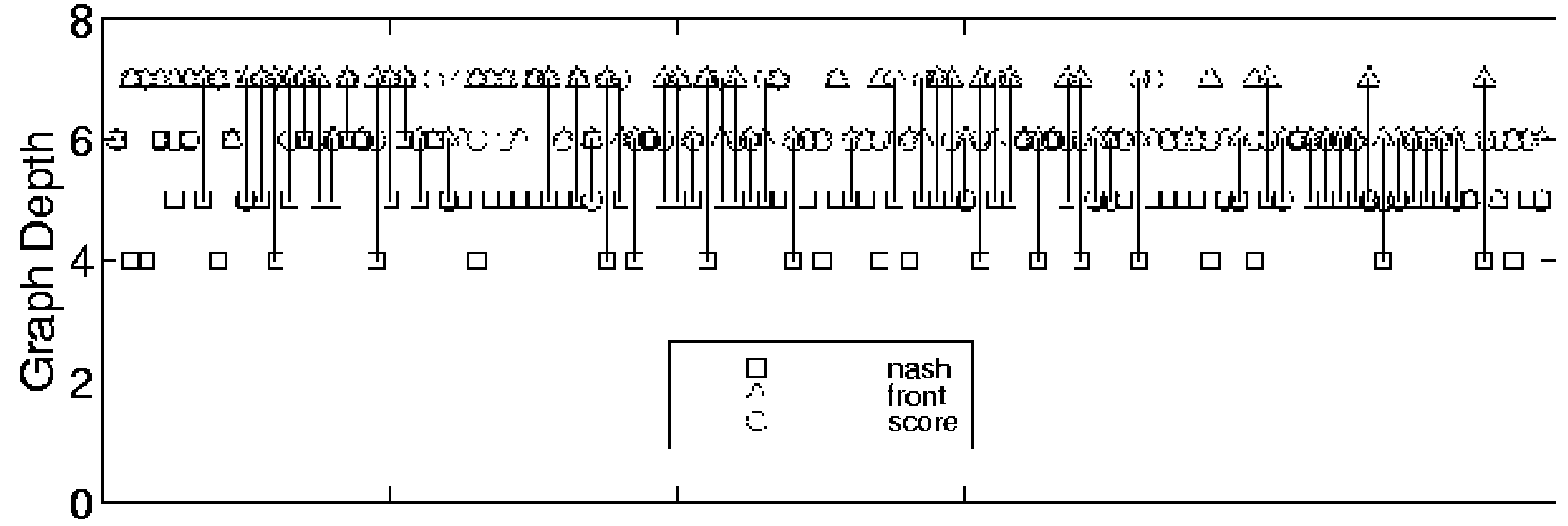


Monotonicity and Open-Endedness

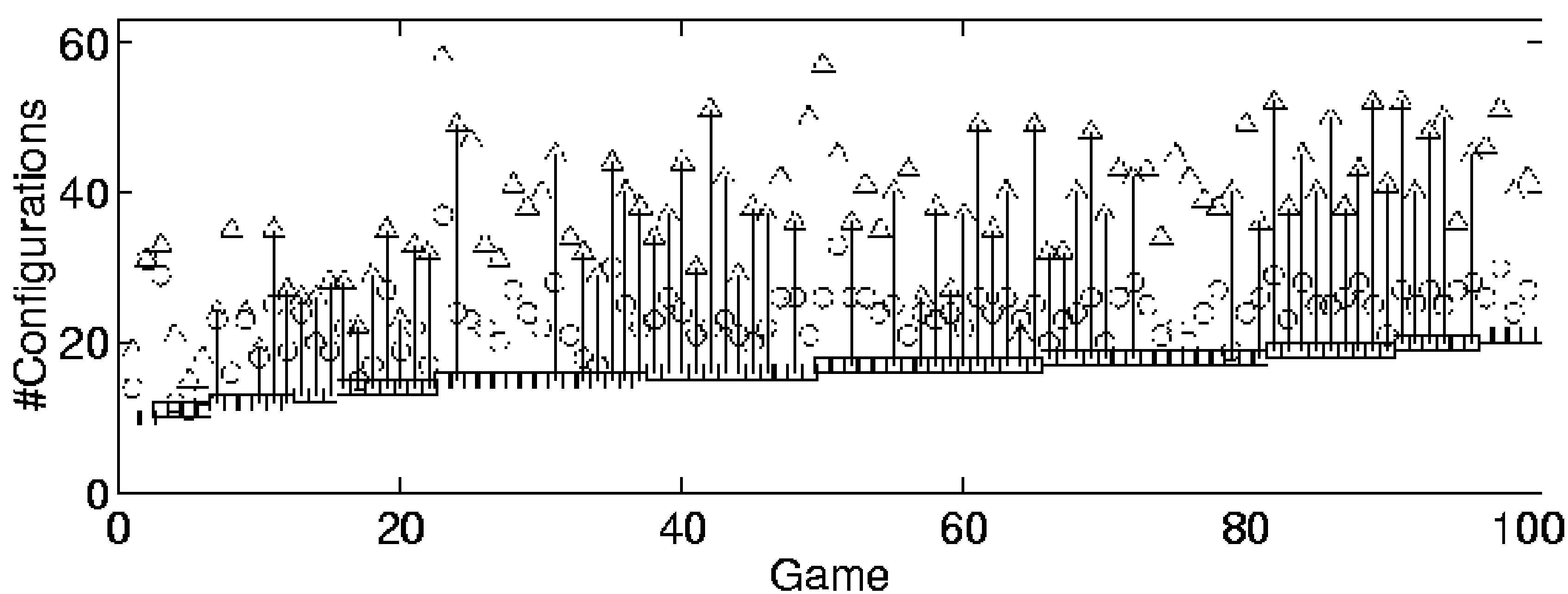
- Open-ended arms race requires:
  - proper substrate (working on this...)
  - proper algorithm (we’ve learned a lot about this)
  - *monotonic solution concept*
- Because of monotonicity, we will never return to a configuration once we leave it; we will never return to offspring configurations

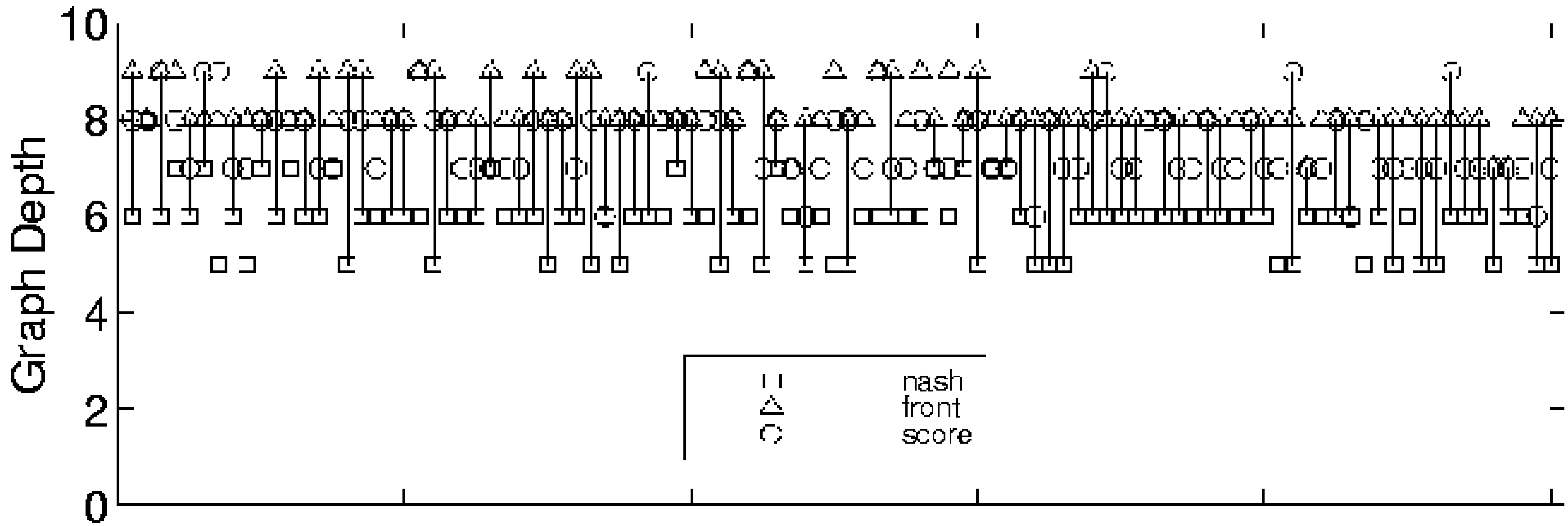
Same 5-strategy game, non-domination concept (non-monotonic version)





Random seven-strategy games





Random nine-strategy games