

CPS 296.1

# Repeated games

Vincent Conitzer  
conitzer@cs.duke.edu

# Repeated games

- In a (typical) repeated game,
  - players play a normal-form game (aka. the **stage game**),
  - then they see what happened (and get the utilities),
  - then they play again,
  - etc.
- Can be repeated finitely or infinitely many times
- Really, an extensive form game
  - Would like to find subgame-perfect equilibria
- One subgame-perfect equilibrium: keep repeating some Nash equilibrium of the stage game
- But are there other equilibria?

# Finely repeated Prisoner's Dilemma

- Two players play the Prisoner's Dilemma  $k$  times

	cooperate	defect
cooperate	2, 2	0, 3
defect	3, 0	1, 1

- In the last round, it is dominant to **defect**
- Hence, in the second-to-last round, there is no way to influence what will happen
- So, it is optimal to defect in this round as well
- Etc.
- So the only equilibrium is to always defect

# Modified Prisoner's Dilemma

- Suppose the following game is played twice

	cooperate	defect <sub>1</sub>	defect <sub>2</sub>
cooperate	5, 5	0, 6	0, 6
defect <sub>1</sub>	6, 0	4, 4	1, 1
defect <sub>2</sub>	6, 0	1, 1	2, 2

- Consider the following strategy:
  - In the first round, cooperate;
  - In the second round, if someone defected in the first round, play defect<sub>2</sub>; otherwise, play defect<sub>1</sub>
- If both players play this, is that a subgame perfect equilibrium?

# Another modified Prisoner's Dilemma

- Suppose the following game is played twice

	cooperate	defect	crazy
cooperate	5, 5	0, 6	1, 0
defect	6, 0	4, 4	1, 0
crazy	0, 1	0, 1	0, 0

- What are the subgame perfect equilibria?
- Consider the following strategy:
  - In the first round, cooperate;
  - In the second round, if someone played defect or crazy in the first round, play crazy; otherwise, play defect
- Is this a Nash equilibrium (**not** subgame perfect)?

# Infinitely repeated games

- First problem: are we just going to add up the utilities over infinitely many rounds?
  - Everyone gets infinity!
- (Limit of) **average** payoff:  $\lim_{n \rightarrow \infty} \frac{1}{n} \sum_{1 \leq t \leq n} u(t)$ 
  - Limit may not exist...
- **Discounted** payoff:  $\sum_t \delta^t u(t)$  for some  $\delta < 1$

# Infinitely repeated Prisoner's Dilemma

	cooperate	defect
cooperate	2, 2	0, 3
defect	3, 0	1, 1

- **Tit-for-tat** strategy:
  - Cooperate the first round,
  - In every later round, do the same thing as the other player did in the **previous** round
- Is both players playing this a Nash/subgame-perfect equilibrium? Does it depend on  $\delta$ ?
- **Trigger** strategy:
  - Cooperate as long as everyone cooperates
  - Once a player defects, defect **forever**
- Is both players playing this a subgame-perfect equilibrium?
- What about one player playing tit-for-tat and the other playing trigger?

# Folk theorem(s)

- Can we somehow characterize the equilibria of infinitely repeated games?
  - Subgame perfect or not?
  - Averaged utilities or discounted?
- Easiest case: averaged utilities, no subgame perfection
- We will characterize what (averaged) **utilities** ( $u_1, u_2, \dots, u_n$ ) the agents can get in equilibrium
- The utilities must be **feasible**: there must be outcomes of the game such that the agents, on average, get these utilities
- They must also be **enforceable**: deviation should lead to punishment that outweighs the benefits of deviation
- **Folk theorem**: a utility vector can be realized by some Nash equilibrium if and only if it is both feasible and enforceable



# Feasibility

2, 2	0, 3
3, 0	1, 1

- The utility vector (2, 2) is feasible because it is one of the outcomes of the game
- The utility vector (1, 2.5) is also feasible, because the agents could **alternate** between (2, 2) and (0, 3)
- What about (.5, 2.75)?
- What about (3, 0.1)?
- In general, **convex combinations** of the outcomes of the game are feasible
  - $p_1a_1 + p_2a_2 + \dots + p_na_n$  is a convex combination of the  $a_i$  if the  $p_i$  sum to 1 and are nonnegative

# Enforceability

2, 2	0, 3
3, 0	1, 1

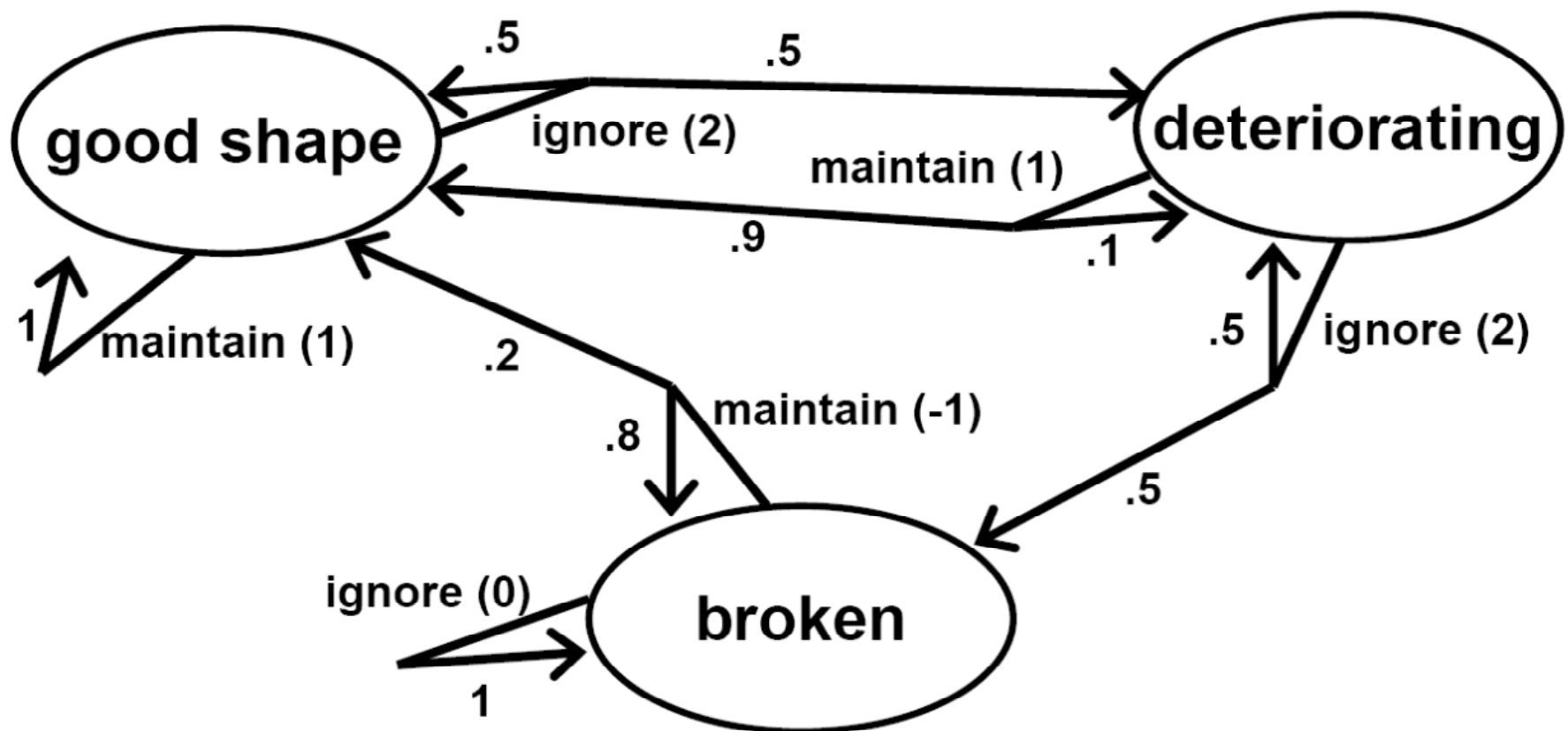
- A utility for an agent is **not enforceable** if the agent can guarantee herself a higher utility
- E.g. a utility of .5 for player 1 is not enforceable, because she can guarantee herself a utility of 1 by defecting
- A utility of 1.2 for player 1 is enforceable, because player 2 can guarantee player 1 a utility of at most 1 by defecting
- What is the relationship to minimax strategies & values?

# Computing a Nash equilibrium in a 2-player repeated game using folk theorem

- Average payoff, no subgame perfection
- Can be done in polynomial time:
  - Compute minimum enforceable utility for each agent
    - I.e., compute maxmin values & strategies
  - Find a feasible point where both players receive at least this utility
    - E.g., both players playing their maxmin strategies
  - Players play feasible point (by rotating through the outcomes), unless the other deviates, in which case they punish the other player by playing minmax strategy forever
    - Minmax strategy easy to compute
- A more complicated (and earlier) algorithm by [Littman & Stone \[04\]](#) computes a “nicer” and subgame-perfect equilibrium

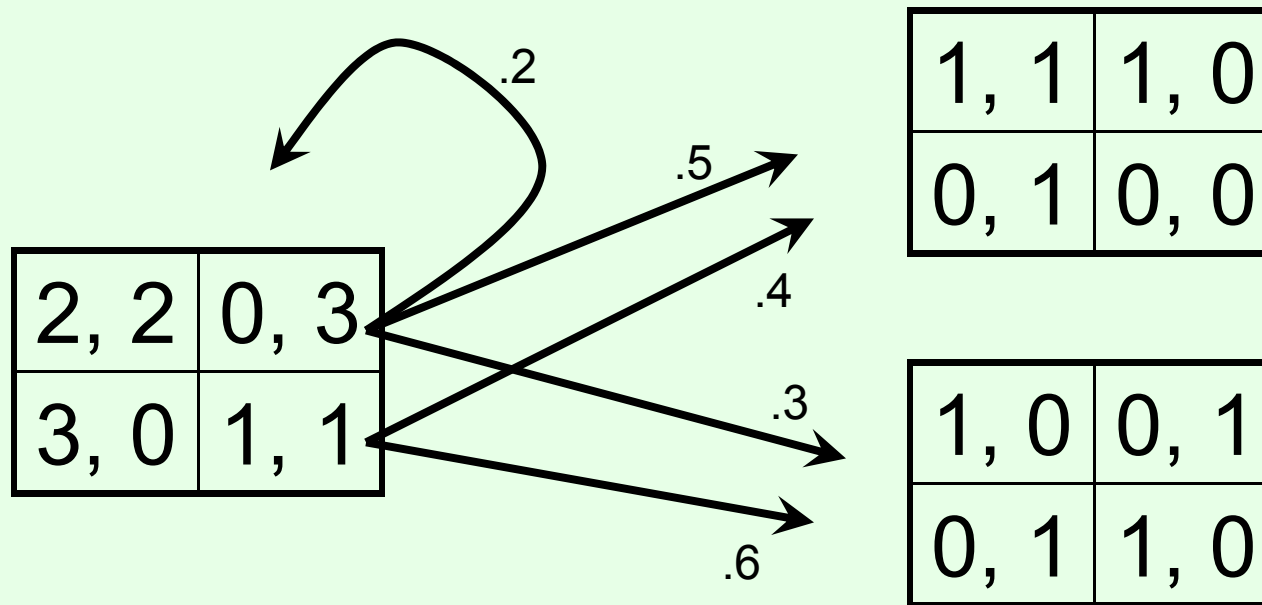
# Example Markov Decision Process (MDP)

- Machine can be in one of three states: good, deteriorating, broken
- Can take two actions: maintain, ignore



# Stochastic games

- A stochastic game has multiple **states** that it can be in
- Each state corresponds to a normal-form game
- After a round, the game randomly **transitions** to another state
- Transition probabilities depend on state and actions taken
- Typically utilities are discounted over time



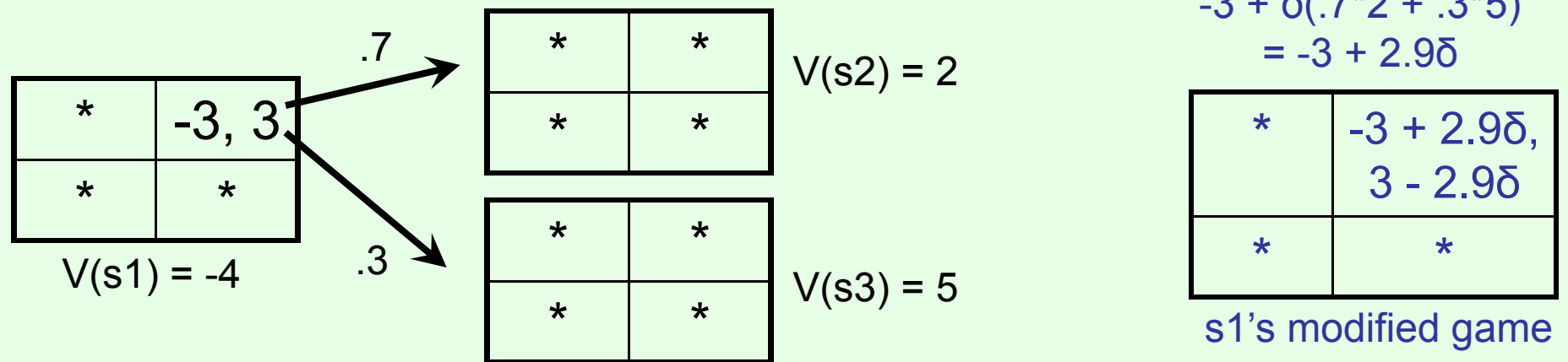
- 1-state stochastic game = (infinitely) repeated game
- 1-agent stochastic game = Markov Decision Process (MDP)

# Stationary strategies

- A **stationary strategy** specifies a mixed strategy for each state
  - Strategy does **not** depend on history
  - E.g., in a repeated game, stationary strategy = always playing the same mixed strategy
- An equilibrium in stationary strategies always exists [Fink 64]
- Each player will have a **value** for being in each state

# Shapley's [1953] algorithm for 2-player zero-sum stochastic games (~value iteration)

- Each state  $s$  is arbitrarily given a value  $V(s)$ 
  - Player 1's utility for being in state  $s$
- Now, for each state, compute a normal-form game that takes these (discounted) values into account



- Solve for the value of the modified game (using LP)
- Make this the new value of  $s_1$
- Do this for all states, repeat until convergence
- Similarly, analogs of policy iteration [Pollatschek & Avi-Itzhak] and Q-Learning [Littman 94, Hu & Wellman 98] exist