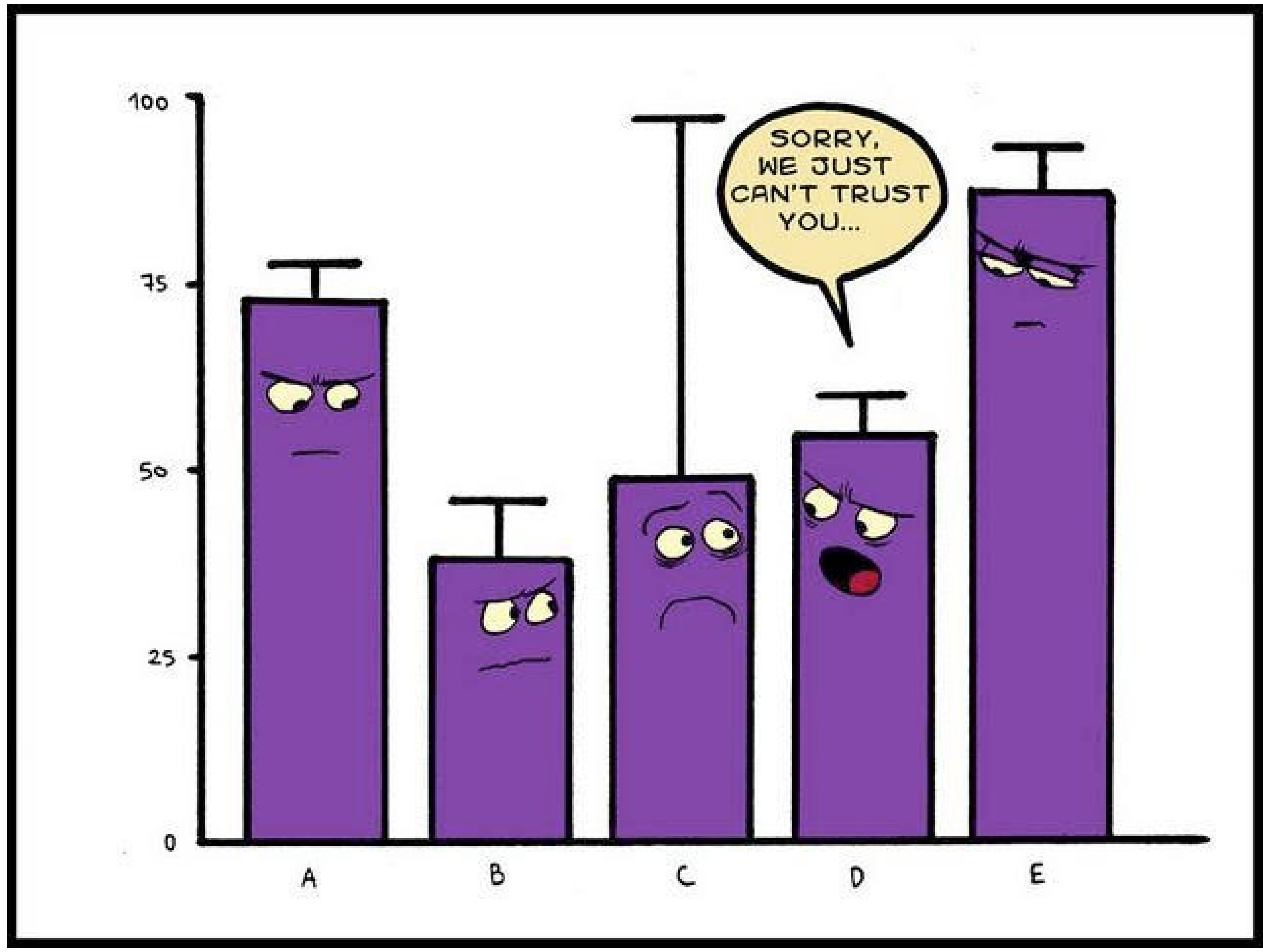
New Directions in Automated Mechanism Design



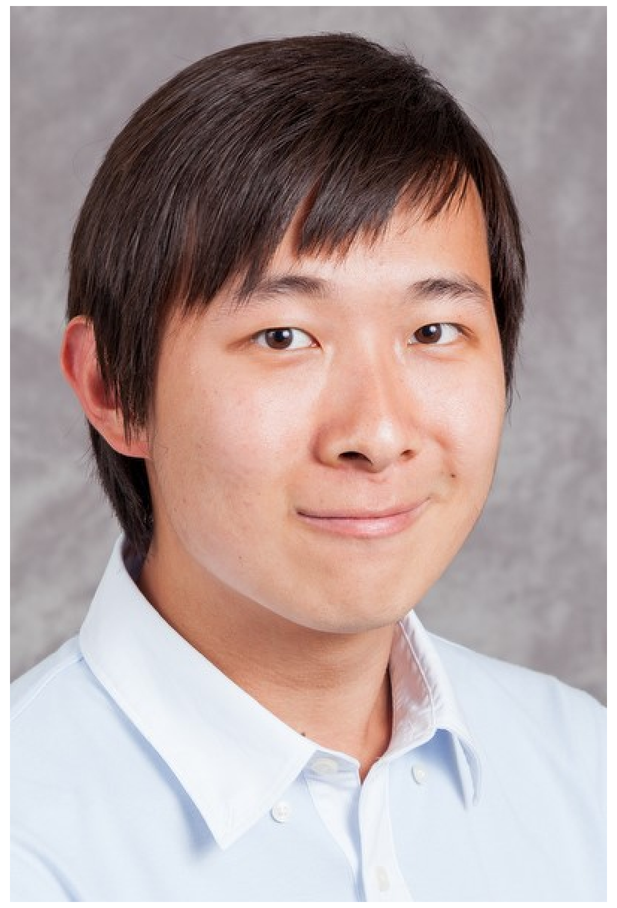
Vincent Conitzer; joint work with:

Michael Albert (Duke → UVA)

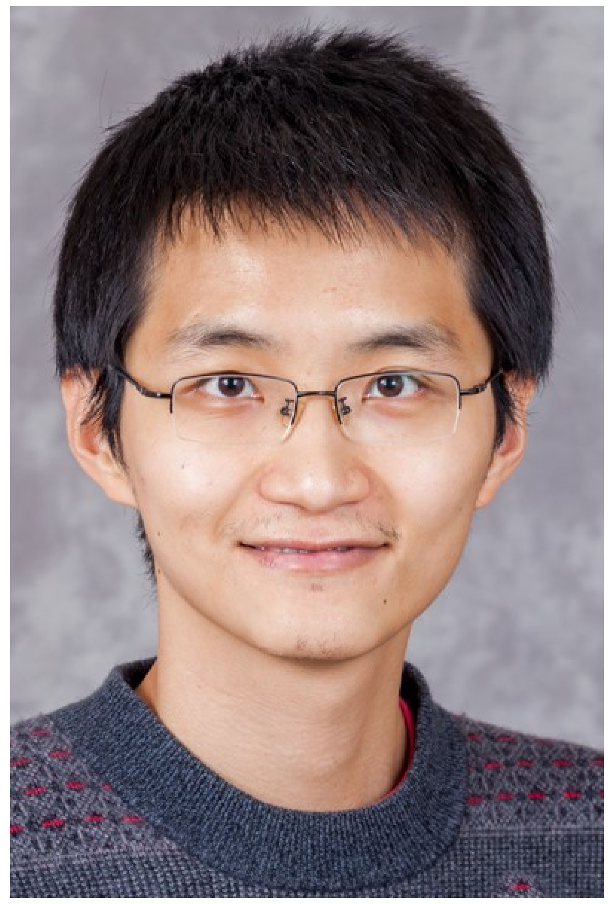
Giuseppe (Pino) Lopomo (Duke)

Peter Stone (UT Austin)

Andrew Kephart (Duke → KeepTruckin)



Hanrui Zhang (Duke)



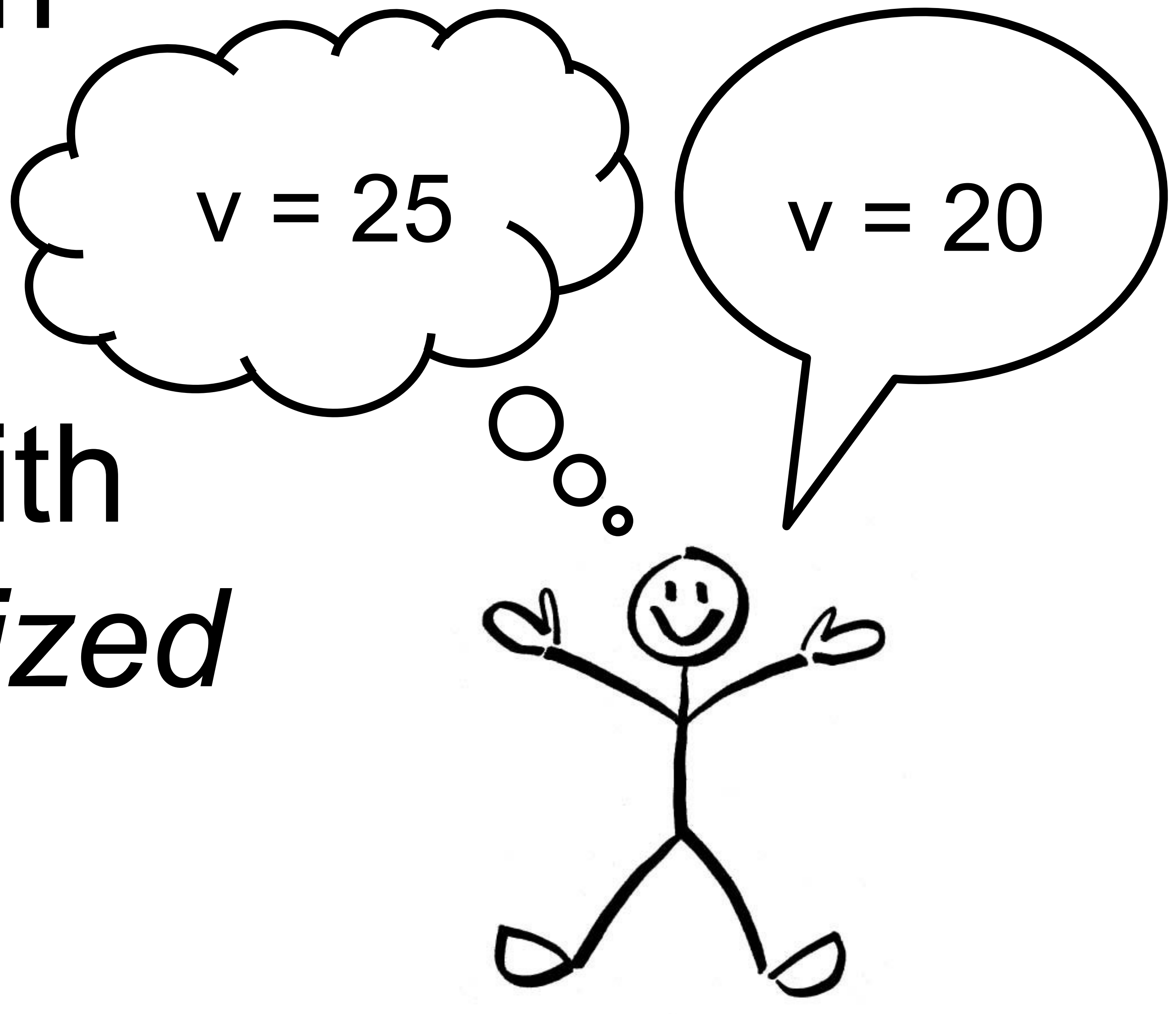
Yu Cheng (Duke → IAS → UIC)

Mechanism design

Make decisions based on the preferences (or other information) of one or more agents (as in social choice)

Focus on *strategic* (game-theoretic) agents with *privately held* information; have to be *incentivized* to reveal it truthfully

Popular approach in design of auctions, matching mechanisms, …



Sealed-bid auctions (on a single item)

Bidder *i* determines how much the item is worth to her (v_i)

Writes a bid (v'_i) on a piece of paper

How would you bid? How much would I make?

First price: Highest bid wins, pays bid

Second price: Highest bid wins, pays next-highest bid

First price with reserve: Highest bid wins iff it exceeds *r*, pays bid

Second price with reserve: Highest bid wins iff it exceeds *r*, pays next highest bid or *r* (whichever is higher)

Revelation Principle

Anything you can achieve, you can also achieve with a truthful (AKA incentive compatible) mechanism.

Revelation Principle

Anything you can achieve, you can also achieve with a truthful (AKA incentive compatible) mechanism.

Automated mechanism design input

Instance is given by

Set of possible *outcomes*

Set of *agents*

For each agent

set of possible *types*

probability distribution over these types

Objective function

Gives a value for each outcome for each combination of agents' types

E.g., social welfare, revenue

Restrictions on the mechanism

Are payments allowed?

Is randomization over outcomes allowed?

What versions of incentive compatibility (IC) & individual rationality (IR) are used?

How hard is designing an optimal *deterministic* mechanism (without reporting costs)?

[C. & Sandholm UAI'02, ICEC'03, EC'04]

NP-complete (even with 1 reporting agent):	**Solvable in polynomial time (for any *constant* number of agents):**
1. Maximizing social welfare (no payments) 2. Designer's own utility over outcomes (no payments) 3. General (linear) objective that doesn't regard payments 4. Expected revenue	1. Maximizing social welfare (not regarding the payments) (VCG)

1 and 3 hold even with no IR constraints

Positive results (randomized mechanisms)

[C. & Sandholm UAI'02, ICEC'03, EC'04]

- Use linear programming
- Variables:

  $p(o \mid \theta_1, \ldots, \theta_n)$ = probability that outcome o is chosen given types $\theta_1, \ldots, \theta_n$

  (maybe) $\pi_i(\theta_1, \ldots, \theta_n)$ = i's payment given types $\theta_1, \ldots, \theta_n$
- Strategy-proofness constraints: for all $i, \theta_1, \ldots \theta_n, \theta_i'$:

  $\Sigma_o p(o \mid \theta_1, \ldots, \theta_n) u_i(\theta_i, o) + \pi_i(\theta_1, \ldots, \theta_n) \geq$

  $\Sigma_o p(o \mid \theta_1, \ldots, \theta_i', \ldots, \theta_n) u_i(\theta_i, o) + \pi_i(\theta_1, \ldots, \theta_i', \ldots, \theta_n)$
- Individual-rationality constraints: for all $i, \theta_1, \ldots \theta_n$:

  $\Sigma_o p(o \mid \theta_1, \ldots, \theta_n) u_i(\theta_i, o) + \pi_i(\theta_1, \ldots, \theta_n) \geq 0$
- Objective (e.g., sum of utilities)

  $\Sigma_{\theta_1, \ldots, \theta_n} p(\theta_1, \ldots, \theta_n) \Sigma_i (\Sigma_o p(o \mid \theta_1, \ldots, \theta_n) u_i(\theta_i, o) + \pi_i(\theta_1, \ldots, \theta_n))$
- Also works for BNE incentive compatibility, ex-interim individual rationality notions, other objectives, etc.
- For deterministic mechanisms, can still use mixed integer programming: require probabilities in $\{0, 1\}$
  - Remember typically designing the optimal deterministic mechanism is NP-hard

A simple example

One item for sale (free disposal)
2 agents, IID valuations: uniform over {1, 2}
Maximize expected revenue under ex-interim IR, Bayes-Nash equilibrium
How much can we get?
(What is optimal expected welfare?)

Agent 2's valuation

Agent 1's valuation	1	2
1	0.25	0.25
2	0.25	0.25

probabilities

Our old AMD solver [C. & Sandholm, 2002, 2003] gives:

```
Status:      OPTIMAL
Objective:   obj = 1.5 (MAXimum)
```

[nonzero variables:]

p_t_1_1_o3	1	(probability of disposal for (1, 1))
p_t_2_1_o1	1	(probability 1 gets the item for (2, 1))
p_t_1_2_o2	1	(probability 2 gets the item for (1, 2))
p_t_2_2_o2	1	(probability 2 gets the item for (2, 2))
pi_2_2_1	2	(1's payment for (2, 2))
pi_2_2_2	4	(2's payment for (2, 2))

A slightly different distribution

One item for sale (free disposal)
2 agents, valuations drawn as on right
Maximize expected revenue under ex-interim IR, Bayes-Nash equilibrium
How much can we get?
(What is optimal expected welfare?)

Agent 2's valuation (columns) / *Agent 1's valuation* (rows)

	1	2
1	0.251	0.250
2	0.250	0.249

probabilities

```
Status:       OPTIMAL
Objective:    obj = 1.749 (MAXimum)
```

[*some of the nonzero payment variables:*]

```
pi_1_1_2        62501
pi_2_1_2       -62750
pi_2_1_1        2
pi_1_2_2        3.992
```

You'd better be really sure about your distribution!

A nearby distribution without correlation

One item for sale (free disposal)
2 agents, valuations IID: 1 w/ .501, 2 w/ .499
Maximize expected revenue under ex-interim IR, Bayes-Nash equilibrium
How much can we get?
(What is optimal expected welfare?)

Agent 2's valuation

Agent 1's valuation	1	2
1	0.251001	0.249999
2	0.249999	0.249001

probabilities

```
Status:      OPTIMAL
Objective:   obj = 1.499 (MAXimum)
```

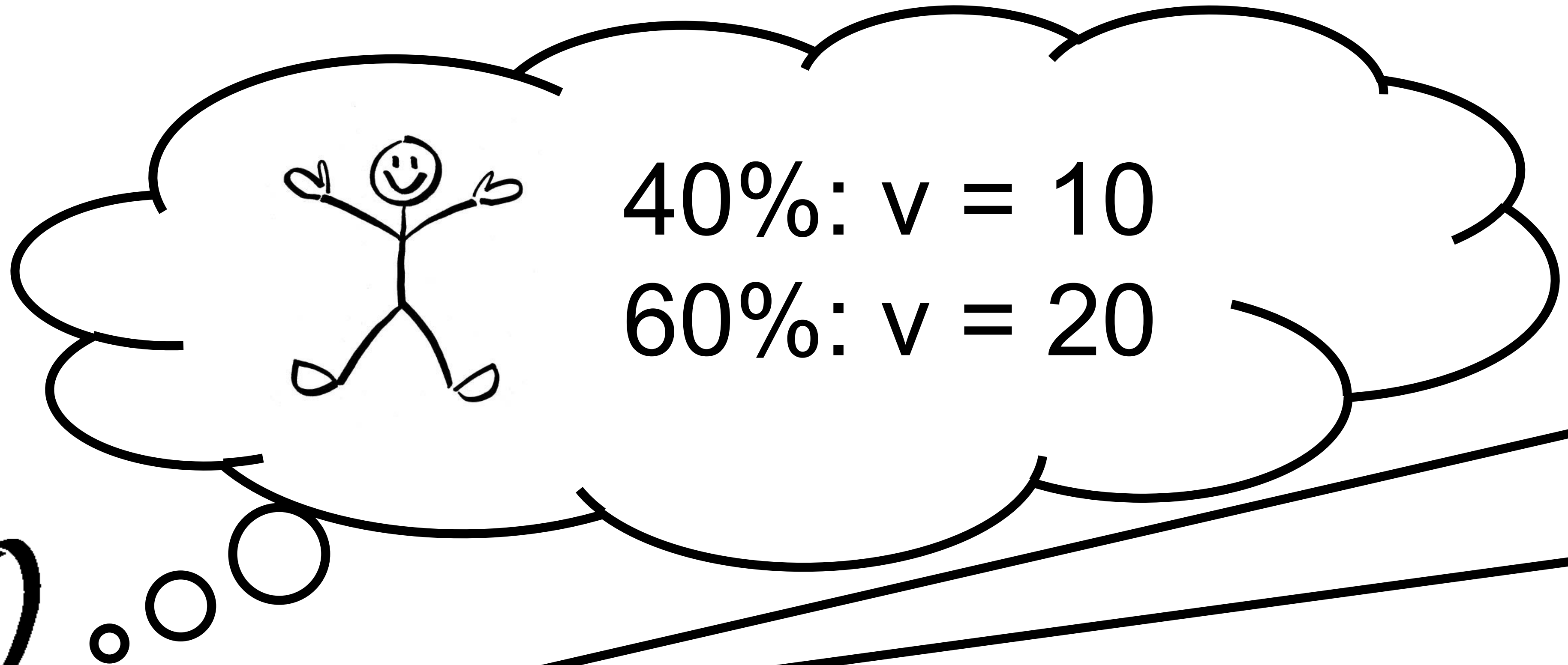
Cremer-McLean [1985]

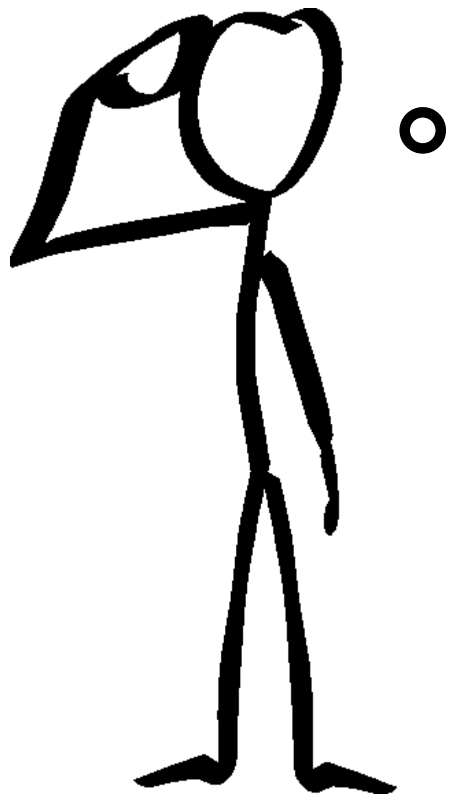
For every agent, consider the following matrix Γ of conditional probabilities, where Θ is the set of types for the agent and Ω is the set of signals (joint types for other agents, or something else observable to the auctioneer)

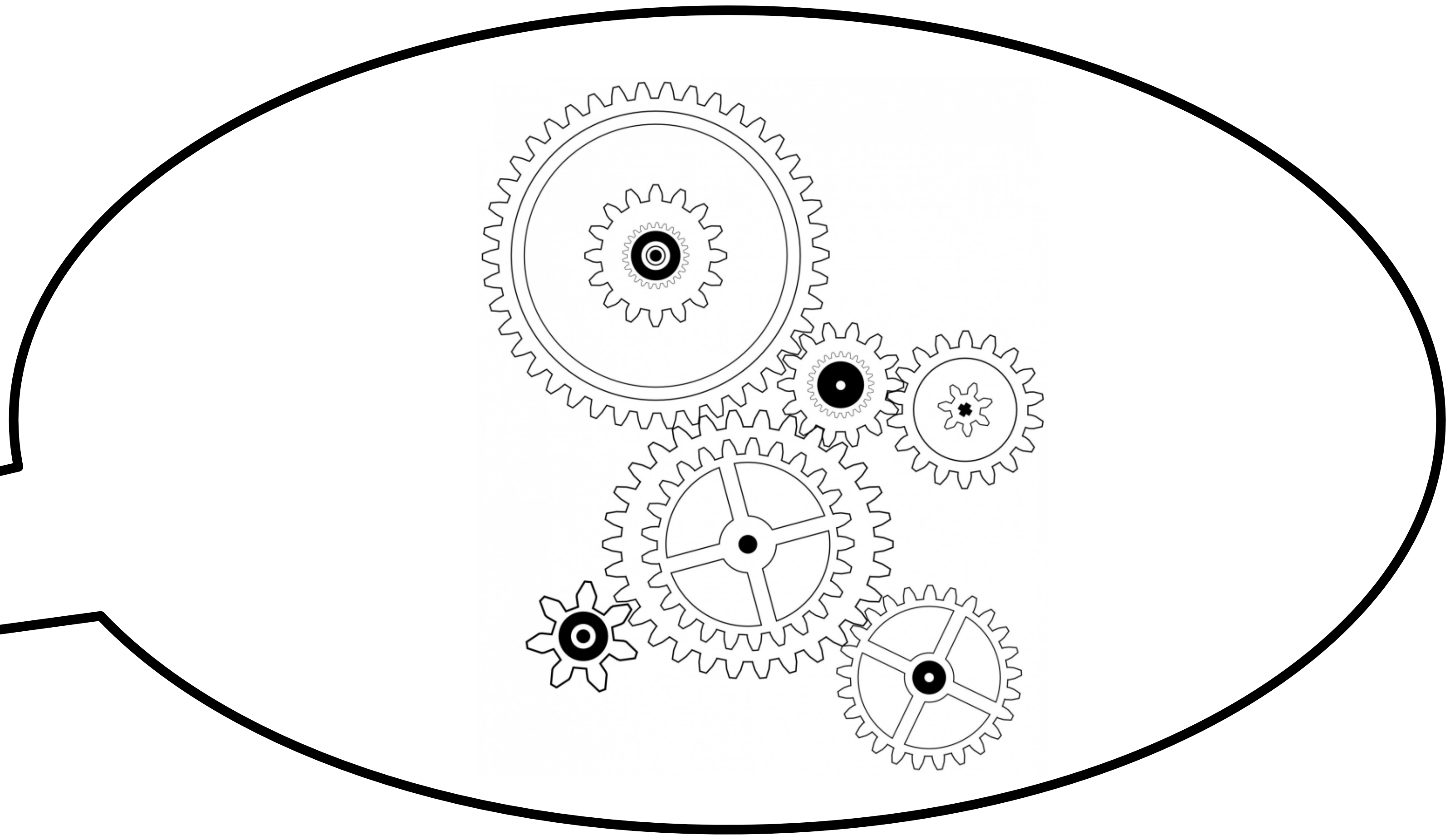
$$\boldsymbol{\Gamma} = \begin{bmatrix} \pi(1|1) & \cdots & \pi(|\Omega||1) \\ \vdots & \ddots & \vdots \\ \pi(1||\Theta|) & \cdots & \pi(|\Omega|||\Theta|) \end{bmatrix}$$

If Γ has rank |Θ| for every agent then the auctioneer can allocate efficiently and extract the full surplus as revenue (!!)

Standard setup in mechanism design

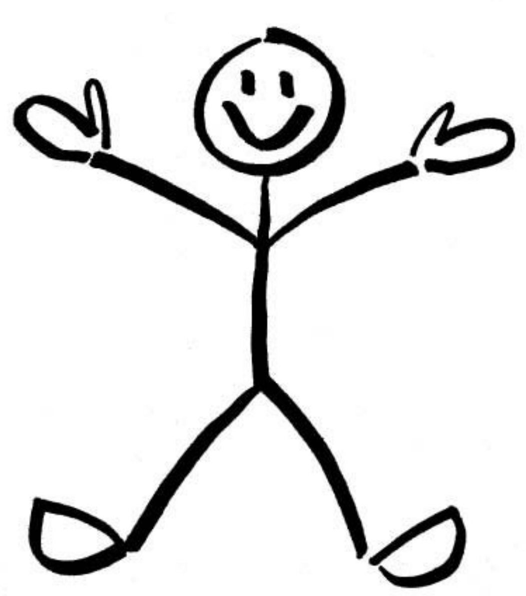


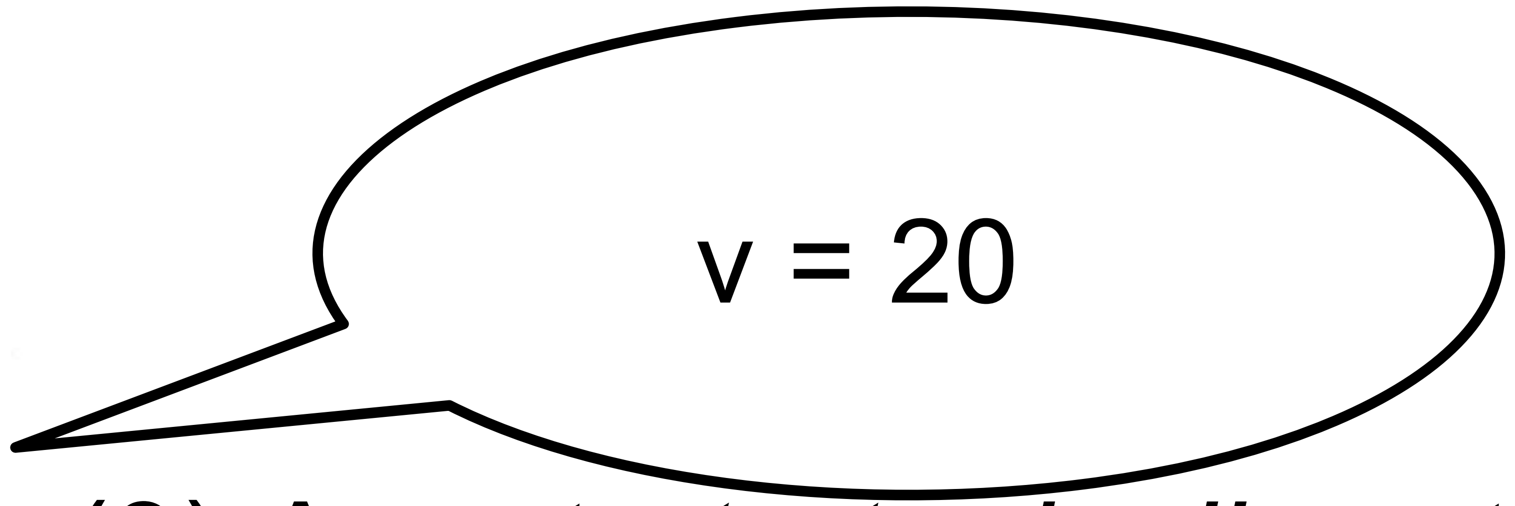




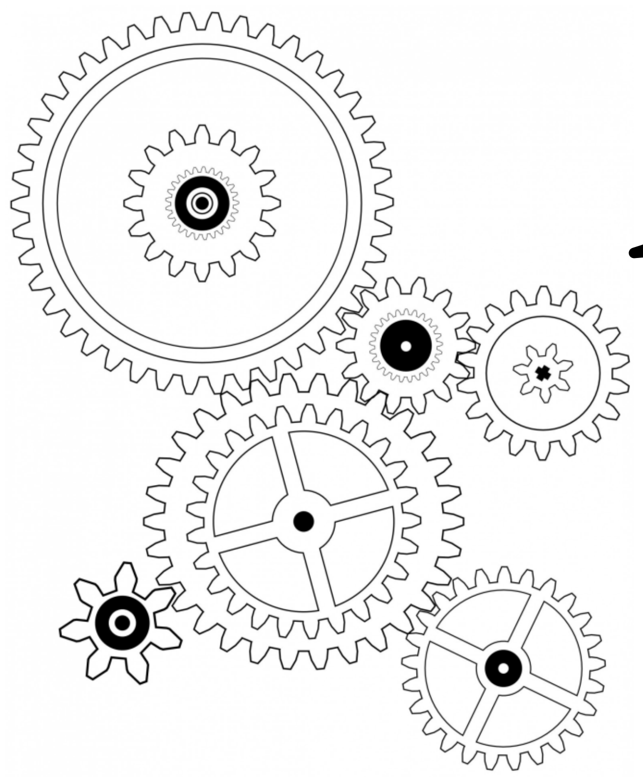
(1) Designer has beliefs about agent's *type* (e.g., preferences)

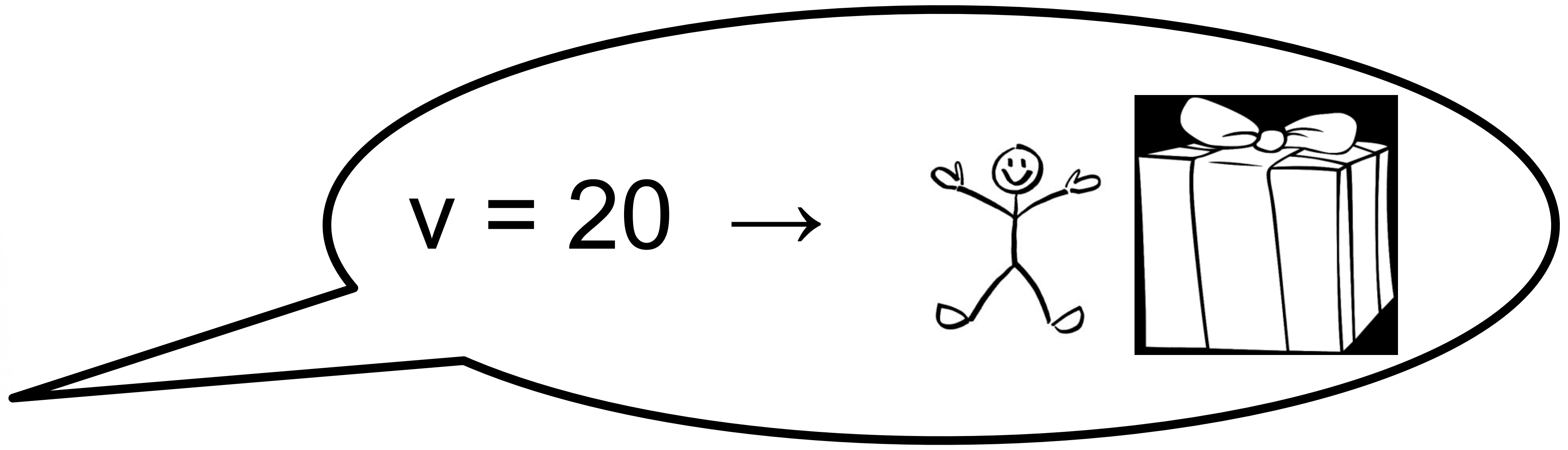
(2) Designer announces mechanism (typically mapping from reported types to outcomes)





(3) Agent *strategically* acts in mechanism (typically type report), *however she likes at no cost*





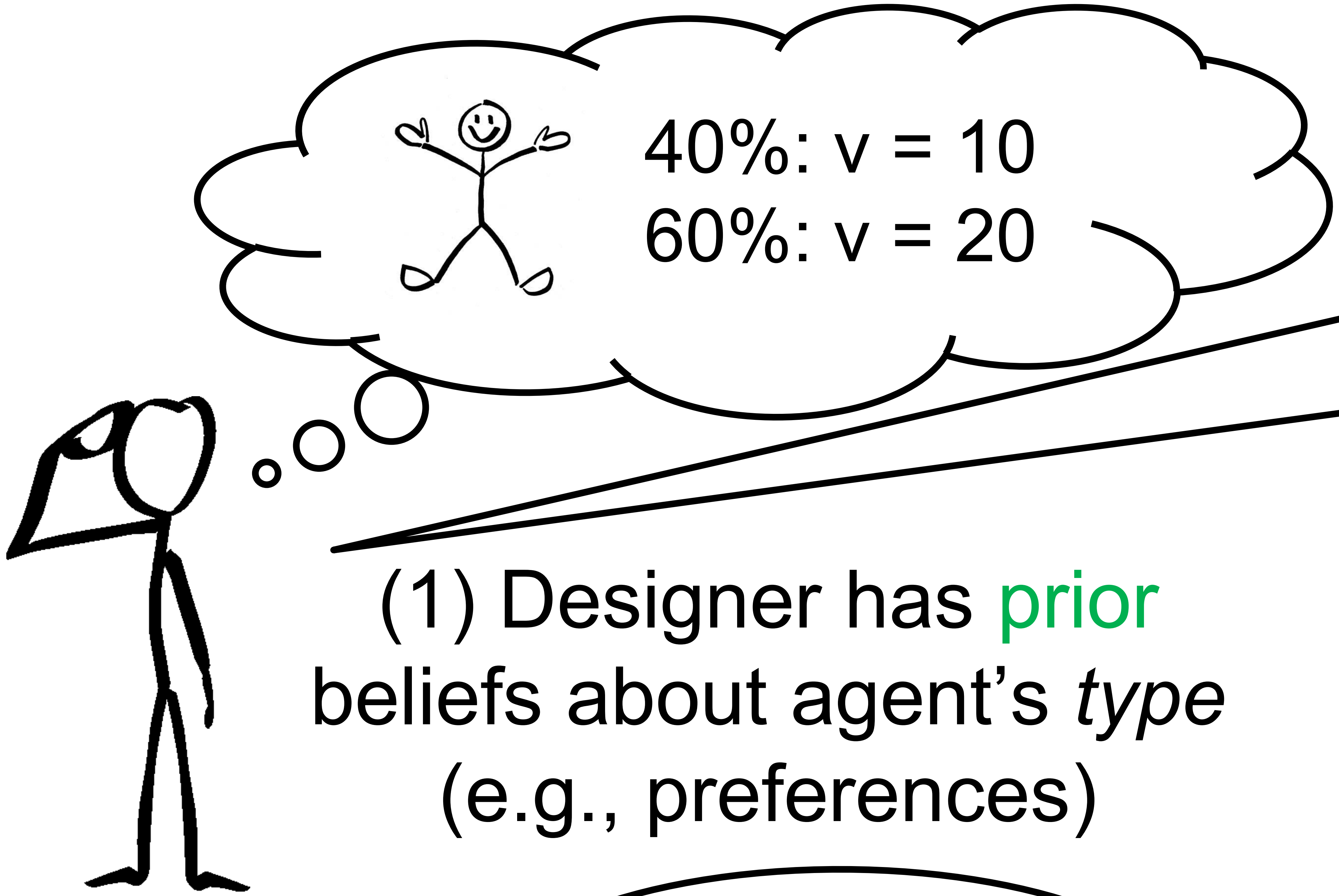
(4) Mechanism functions as specified

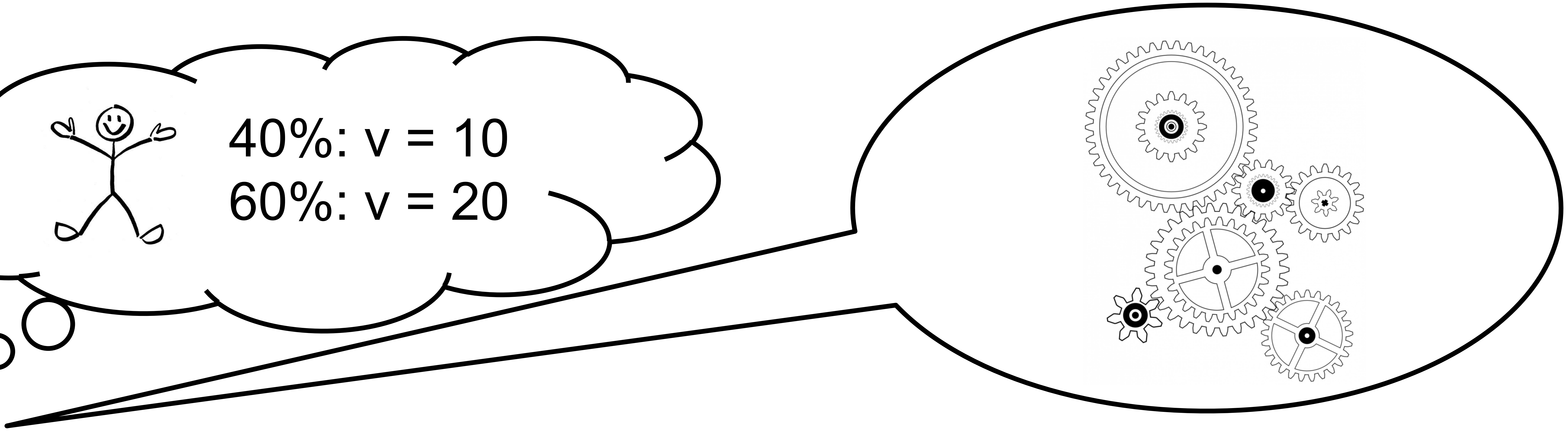
The mechanism may have more information about the specific agent!

application	**information**
online marketplaces	actions taken online
selling insurance	driving record
university admissions	courses taken
webpage ranking	links to page

Attempt 1 at fixing this

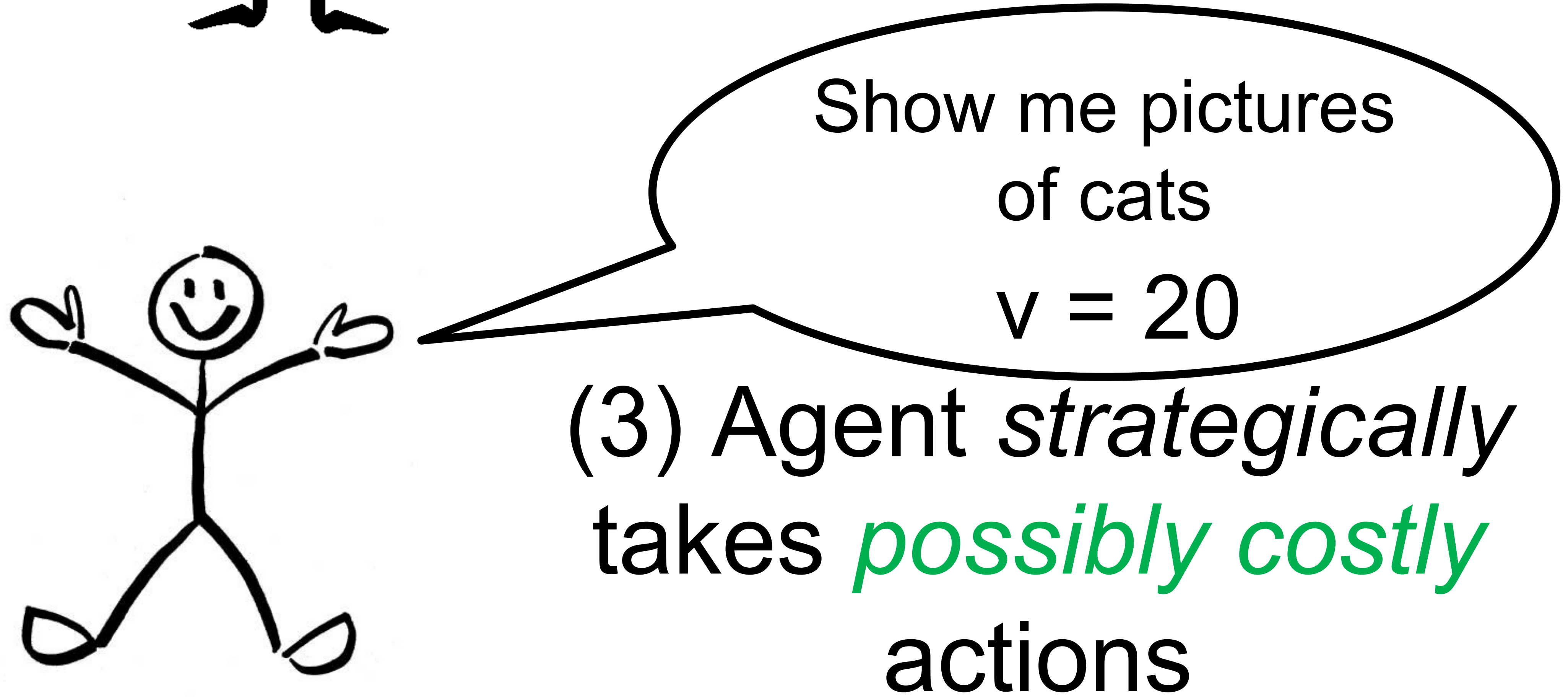
Attempt 2: Sophisticated agent



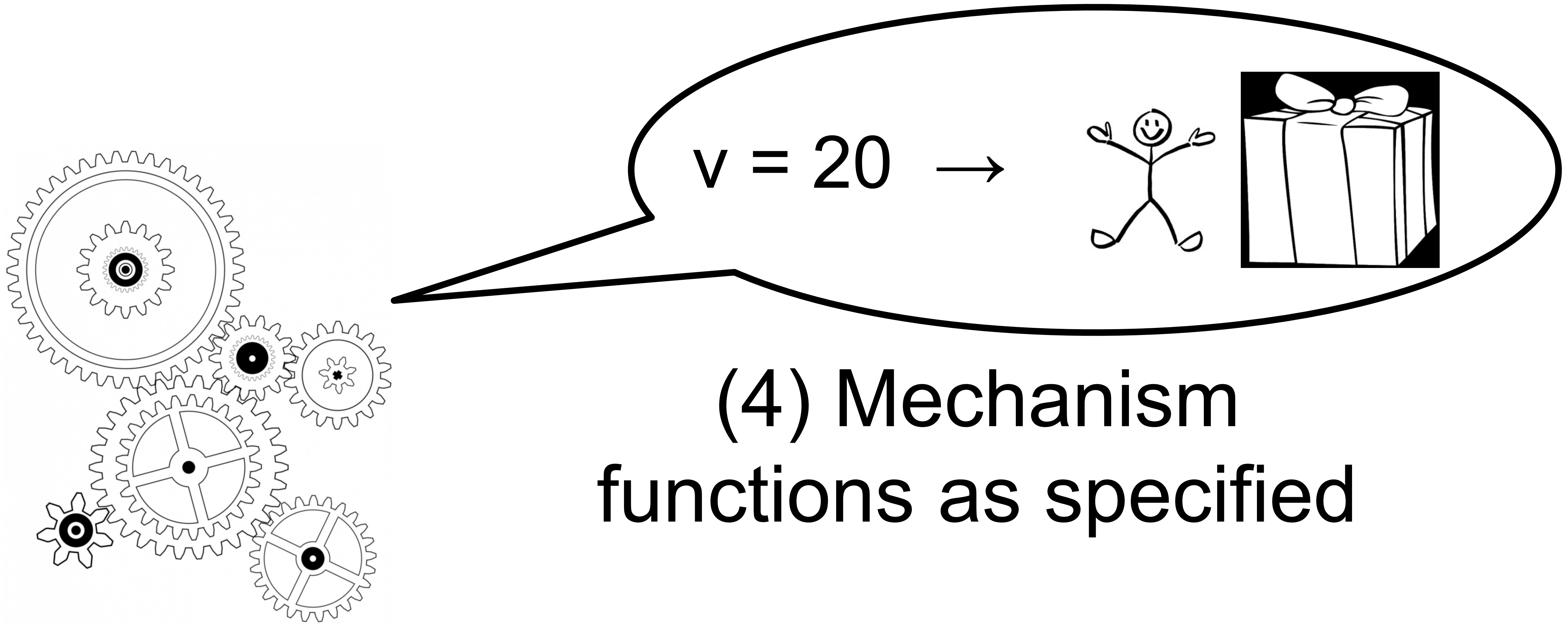


(1) Designer has prior beliefs about agent's *type* (e.g., preferences)

(2) Designer announces mechanism (typically mapping from reported types to outcomes)

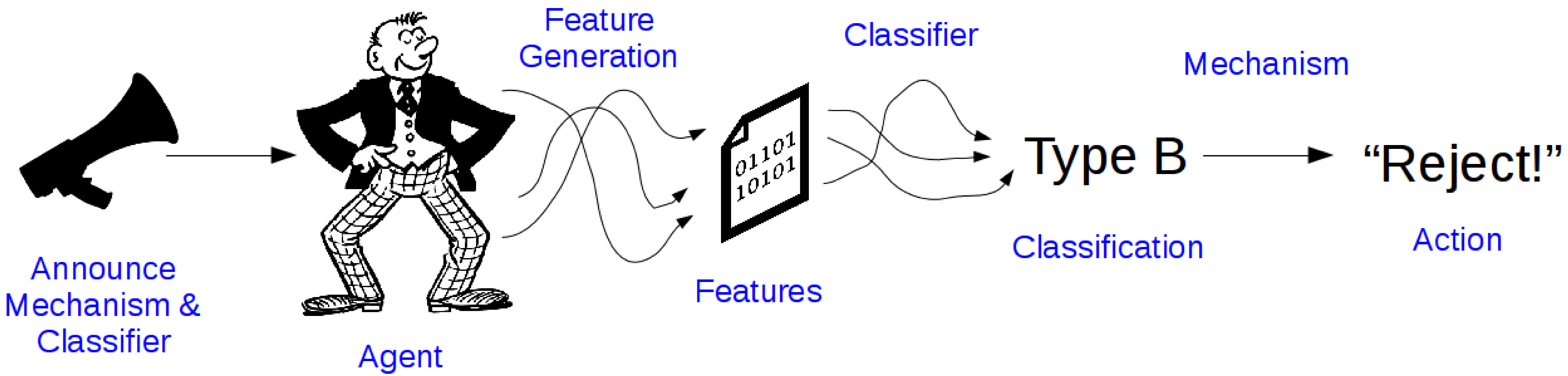


(3) Agent *strategically* takes *possibly costly* actions



(4) Mechanism functions as specified

Machine learning view



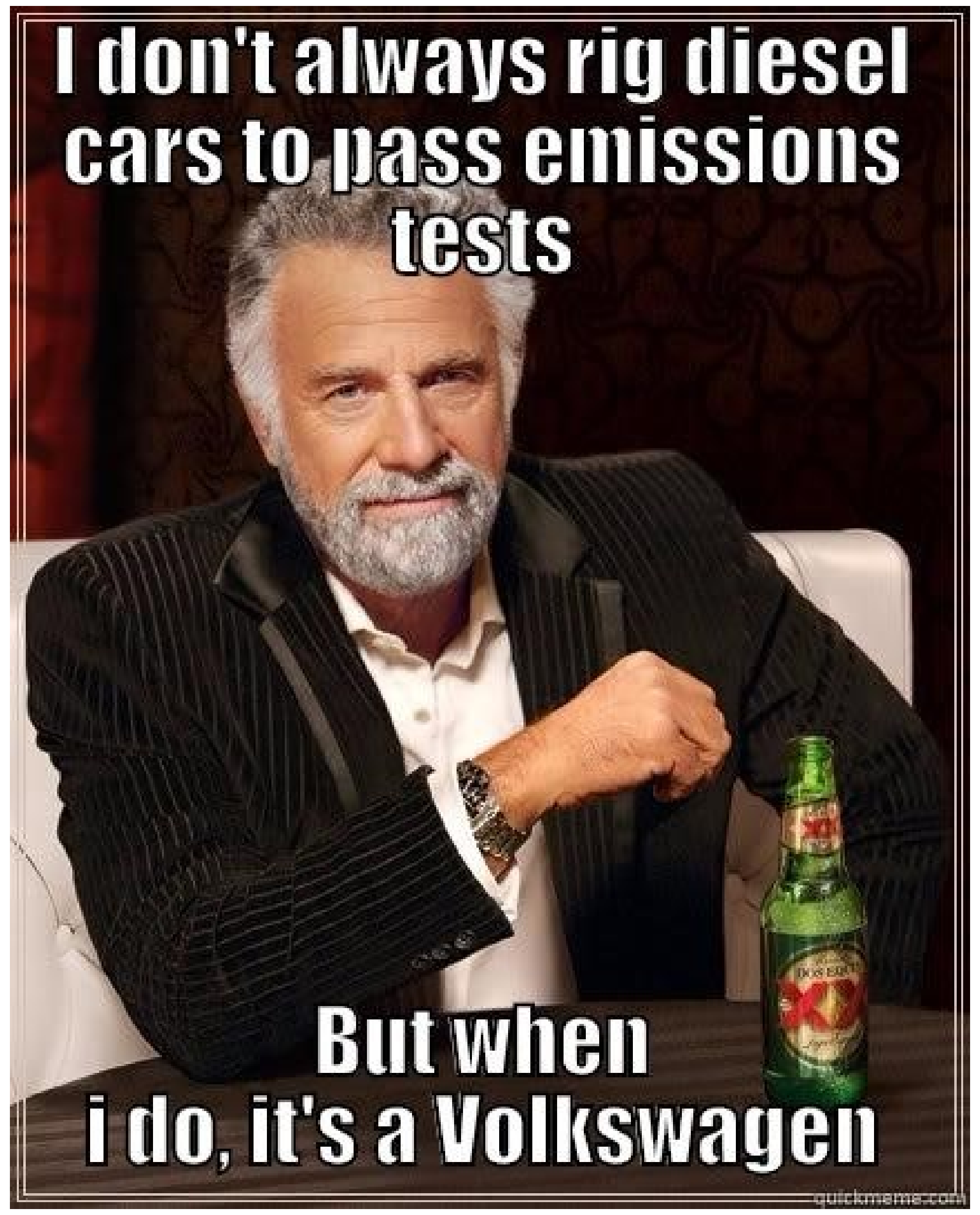
See also later work by Hardt, Megiddo, Papadimitriou, Wootters [2015/2016]

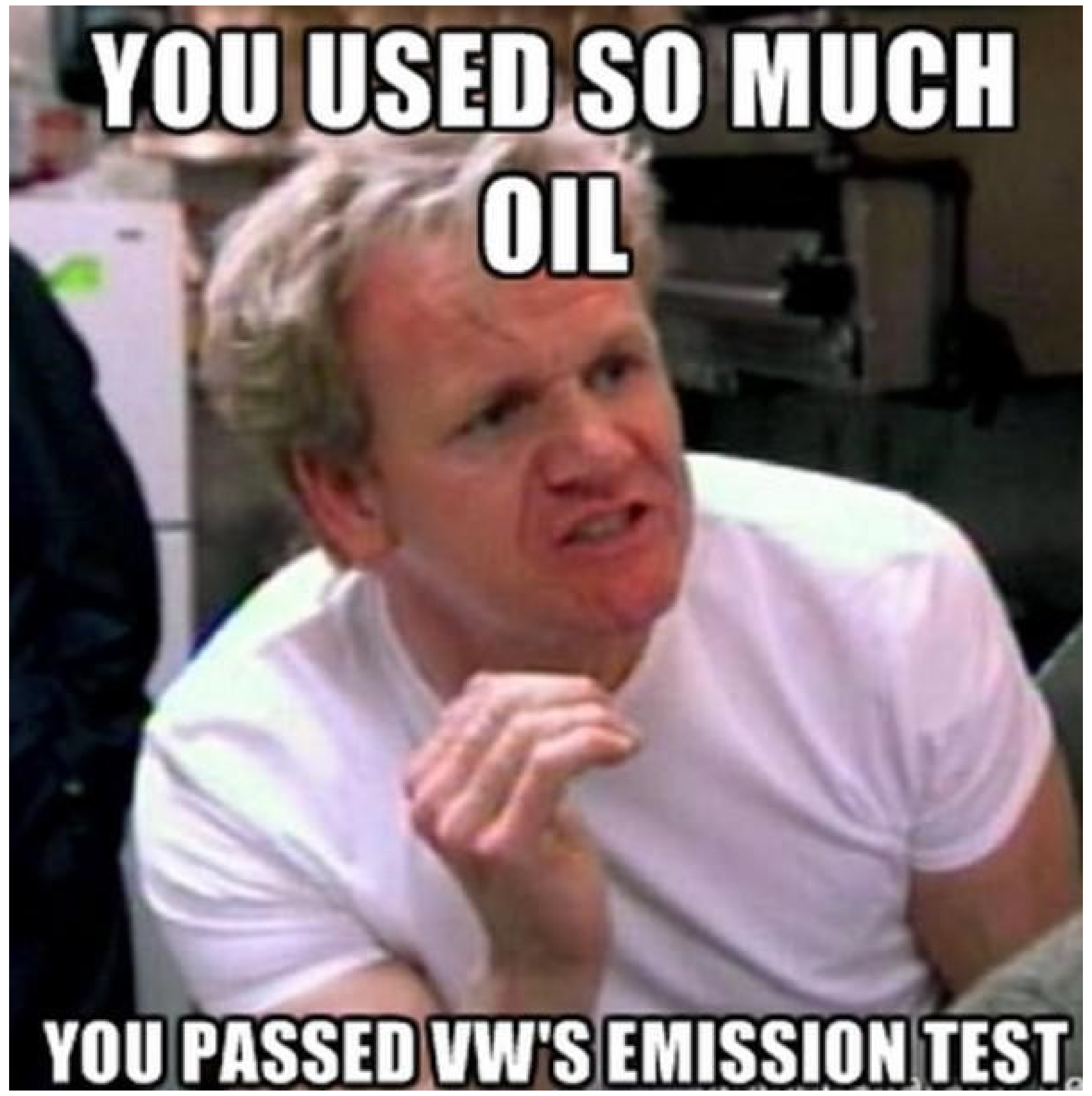
From Ancient Times…

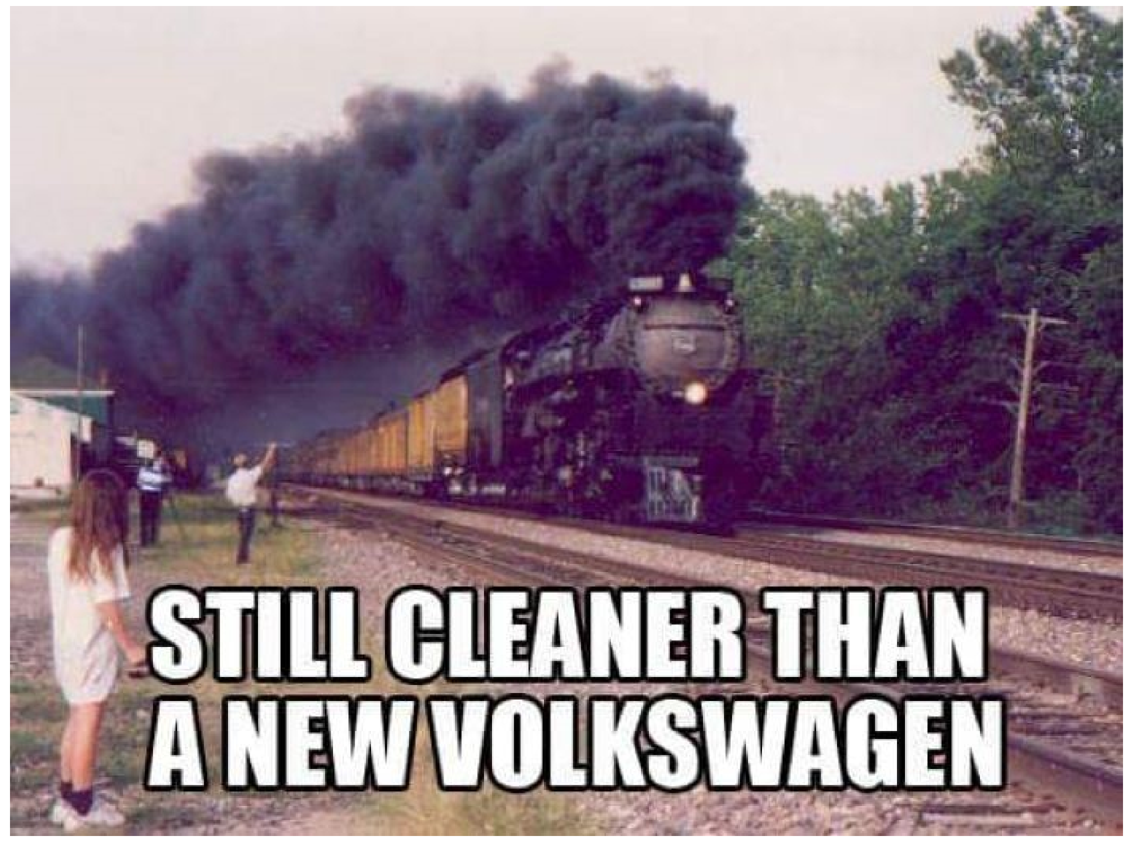
Jacob and Esau

Trojan Horse

… to Modern Times







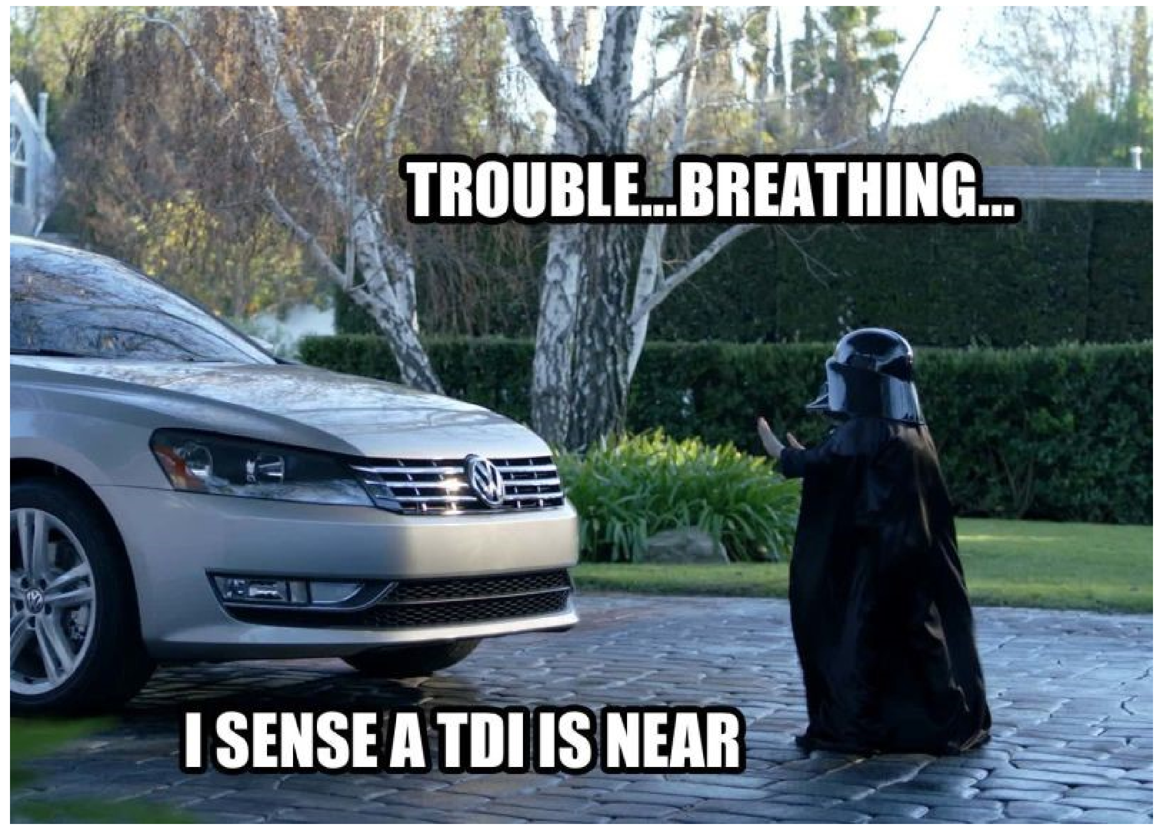


Illustration: Barbara Buying Fish From Fred

Types $(t \in T)$	**Actions** $(a \in A)$	**Choice Function** $(F : T \to A)$
fresh	*accept*	$fresh \to accept$
ok	*reject*	$ok \to accept$
rotten		$rotten \to reject$

Classifications $(\hat{t} \in \hat{T})$**:** $\widehat{fresh}$, $\widehat{ok}$, $\widehat{rotten}$

... continued

Effort Function $(E : T \times \hat{T} \to \mathbb{R})$**:**

	$\hat{fresh}$	$\hat{ok}$	$\hat{rotten}$
$fresh$	0	0	0
ok	10	0	0
$rotten$	30	10	0

Valuation Function $(V : T \times A \to \mathbb{R})$**:**

$$V(\cdot, accept) = 20,\ V(\cdot, reject) = 0$$

Mechanism $M : \hat{T} \to A$

First Try: $M = \hat{fresh} \to accept,\ \hat{ok} \to accept,\ \hat{rotten} \to reject$

... continued

Effort Function$(E : T \times \hat{T} \to \mathbb{R})$**:**

	$\hat{fresh}$	$\hat{ok}$	$\hat{rotten}$
$fresh$	0	0	0
ok	10	0	0
$rotten$	30	10	0

Valuation Function $(V : T \times A \to \mathbb{R})$**:**

$$V(\cdot, accept) = 20, \; V(\cdot, reject) = 0$$

Mechanism $M : \hat{T} \to A$

First Try: $M = \hat{fresh} \to accept, \; \hat{ok} \to accept, \; \hat{rotten} \to reject$

Better: $M^* = \hat{fresh} \to accept, \; \hat{ok} \to reject, \; \hat{rotten} \to reject.$

Comparison With Other Models

	$\hat{fresh}$	$\hat{ok}$	$\hat{rotten}$
fresh	0	0	0
ok	0	0	0
rotten	0	0	0

Standard Mechanism Design

	$\hat{fresh}$	$\hat{ok}$	$\hat{rotten}$
fresh	0	∞	0
ok	∞	0	0
rotten	0	0	0

Mechanism Design with Partial Verification

	$\hat{fresh}$	$\hat{ok}$	$\hat{rotten}$
fresh	∞	0	0
ok	1.2	5	$-\infty$
rotten	-3	0	0

Mechanism Design with Signaling Costs

Green and Laffont. Partially verifiable information and mechanism design. RES 1986
Auletta, Penna, Persiano, Ventre. Alternatives to truthfulness are hard to recognize. AAMAS 2011

Question

Given:

Types $(t \in T)$	**Actions** $(a \in A)$	**Choice Function** $(F : T \to A)$
fresh	*accept*	*fresh* $\to$ *accept*
ok	*reject*	*ok* $\to$ *accept*
rotten		*rotten* $\to$ *reject*

Classifications $(\hat{t} \in \hat{T})$: $\widehat{fresh}$, $\widehat{ok}$, $\widehat{rotten}$

Effort Function $(E : T \times \hat{T} \to \mathbb{R})$:

	$\widehat{fresh}$	$\widehat{ok}$	$\widehat{rotten}$
fresh	0	0	0
ok	10	0	0
rotten	30	10	0

Valuation Function $(V : T \times A \to \mathbb{R})$:

$$V(\cdot, accept) = 20, \ V(\cdot, reject) = 0$$

Then:

Does there exist a

Mechanism $M : \hat{T} \to A$

which implements the choice function?

NP-complete!

Auletta, Penna, Persiano, Ventre. Alternatives to truthfulness are hard to recognize. AAMAS 2011

Results

with Andrew Kephart (AAMAS 2015)

		Transfers (T)		No Transfers (NT)	
		Two Outcomes (TO)	*Injective SCF (FI)*	*Two Outcomes (TO)*	*Injective SCF (FI)*
Free Utilities (FU)	*Unrestricted Costs (U)*	NP-c	**NP-c**	NP-c	**NP-c**
	$\{0, \infty\}$ *Costs (ZI)*	NP-c	**NP-c**	NP-c	**P**
Targeted Utilities (TU)	*Unrestricted Costs (U)*	**NP-c**	**P**	NP-c	**P**
	$\{0, \infty\}$ *Costs (ZI)*	**NP-c**	**P**	NP-c	**P**

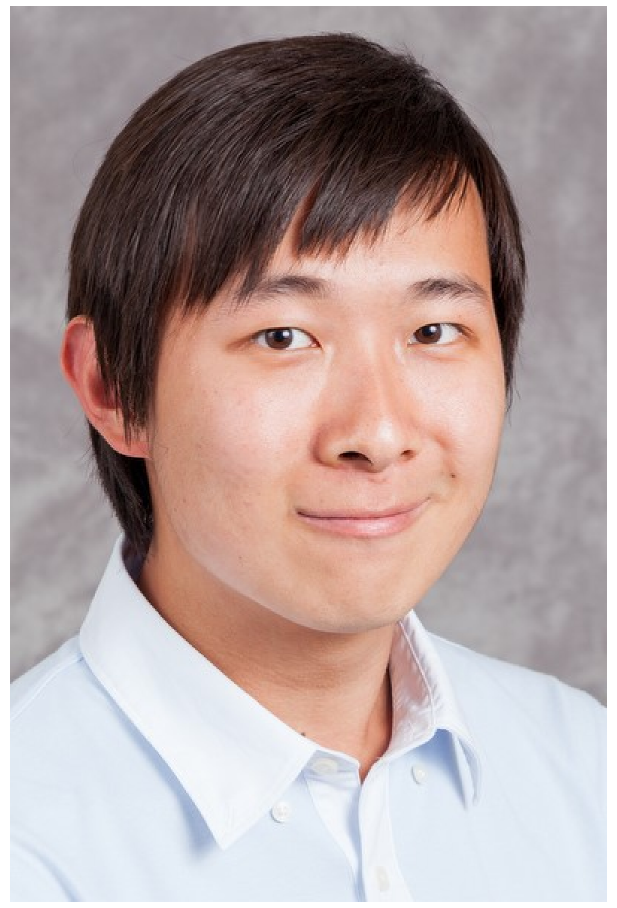
Non-bolded results are from:
Auletta, Penna, Persiano, Ventre. Alternatives to truthfulness are hard to recognize. AAMAS 2011

Hardness results fundamentally rely on revelation principle failing – conditions under which revelation principle still holds in Green & Laffont '86 and Yu '11 (partial verification), and Kephart & C. EC'16 (costly signaling).

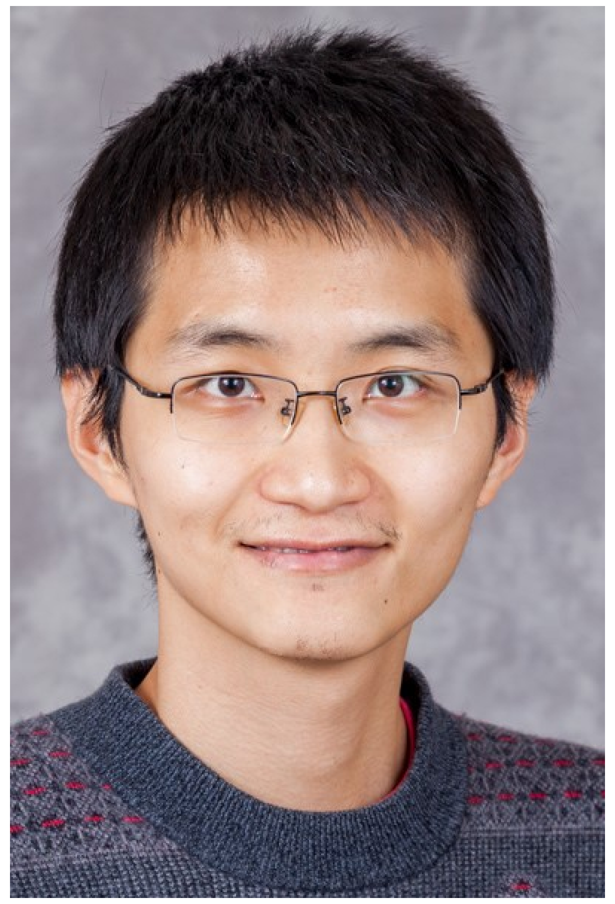
When Samples Are Strategically Selected

Bob, Professor of Rocket Science

ICML 2019, with



Hanrui Zhang (Duke)



Yu Cheng (Duke → IAS → UIC)

Academic hiring…

Charlie, Bob's student

Academic hiring…

I NEED TO CHOOSE THE BEST 3 PAPERS TO CONVINCE BOB, SO THAT HE WILL HIRE ALICE.

CHARLIE WILL DEFINITELY PICK THE BEST 3 PAPERS BY ALICE, AND I NEED TO CALIBRATE FOR THAT.

The general problem

A **distribution (Alice)** over paper qualities $\theta \in \{g, b\}$ arrives, which can be either a good one ($\theta = g$) or a bad one ($\theta = b$)

Alice, the postdoc applicant

The general problem

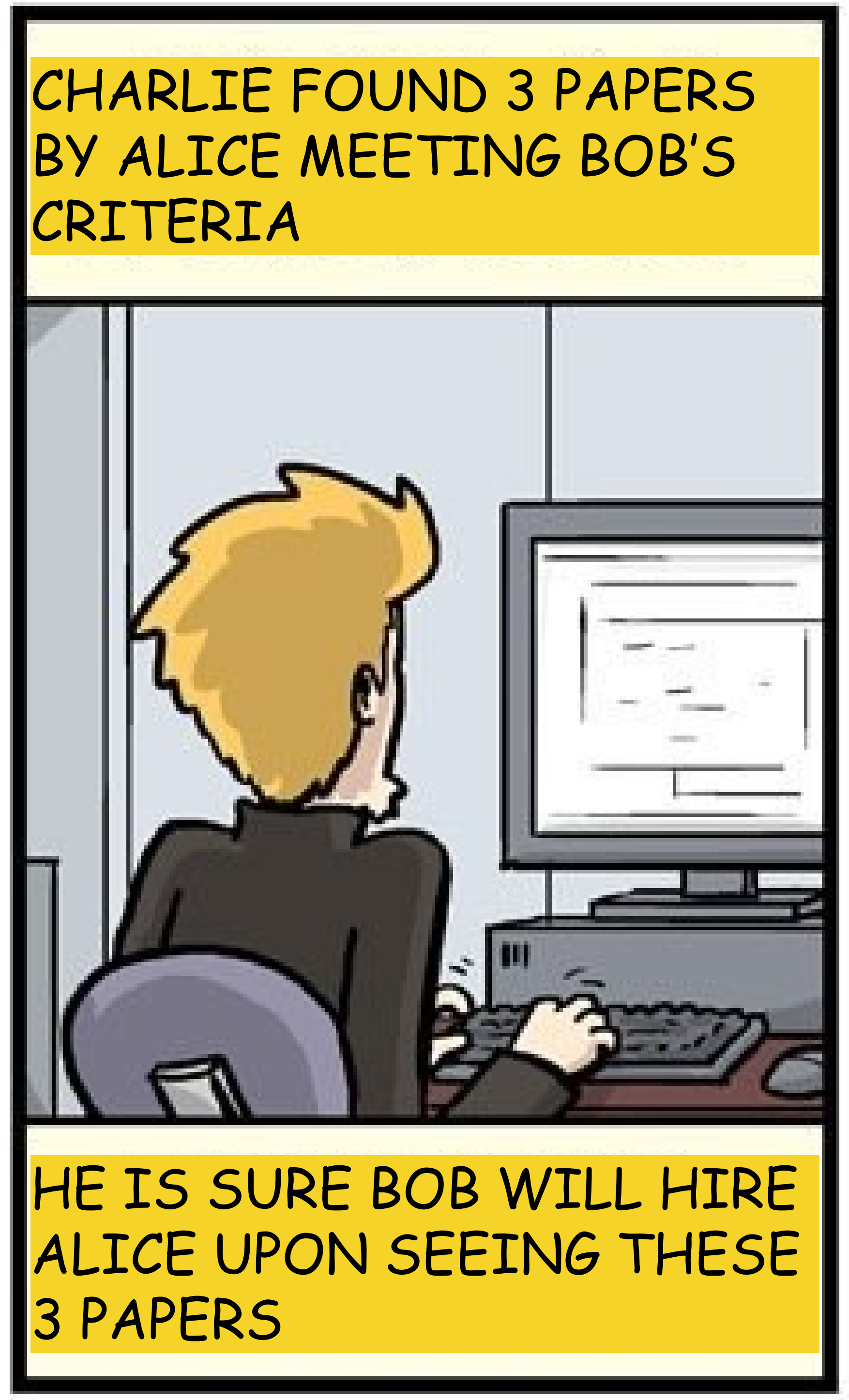
The **principal (Bob)** announces a **policy**, according to which he decides, based on the **report** of the **agent (Charlie)**, whether to **accept θ (hire Alice)**

The general problem

The **agent (Charlie)** has access to **n(=15) iid samples (papers)** from θ (Alice), from which he **chooses m(=3)** as his **report**

The general problem

The **agent (Charlie)** sends his **report** to the principal, **aiming to convince the principal (Bob) to accept θ (Alice)**

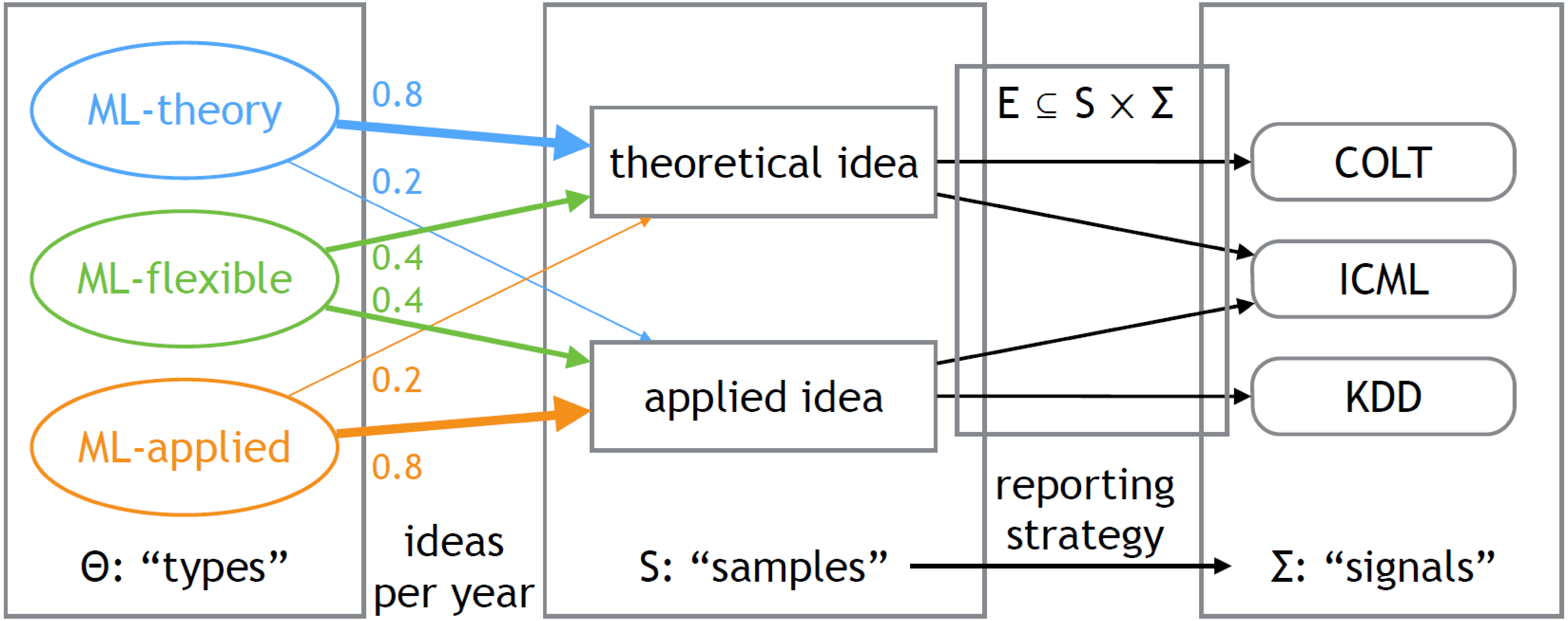


The general problem

The **principal (Bob)** observes the **report** of the **agent (Charlie)**, and makes the decision according to the policy announced

Questions

How does strategic selection affect the principal's policy?
Is it easier or harder to classify based on **strategic samples**, compared to when the principal has access to **iid samples**?
Should the principal ever have a **diversity** requirement (e.g., at least 1 mathematical paper and at least 1 experimental paper), or only go by total quality according to a single metric?



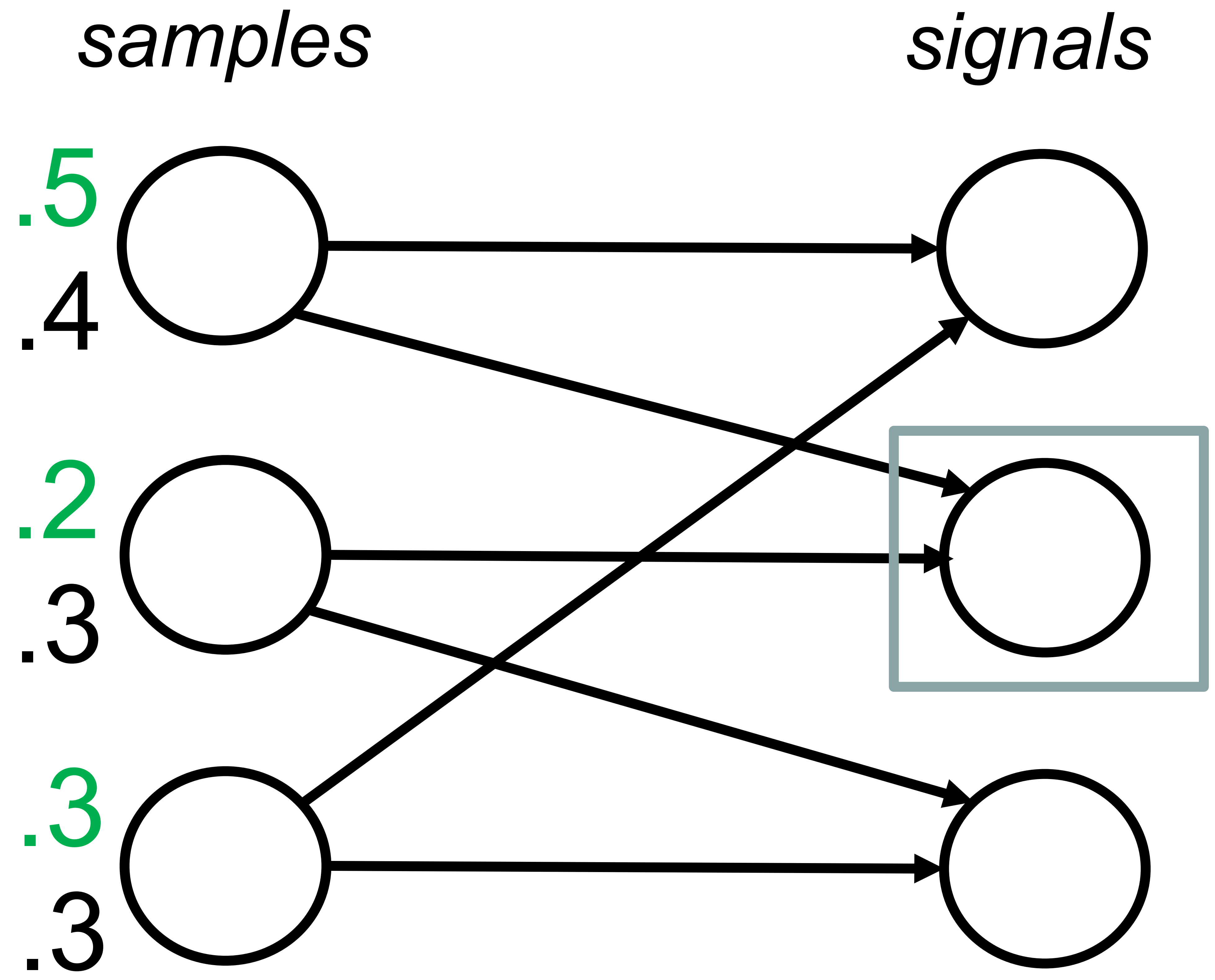
Agent's problem:

- "How do I distinguish myself from other types?"
- "How many samples do I need for that?"

Principal's problem:

- "How do I tell ML-flexible agents from others?"
- "At what point in their career can I reliably do that?"

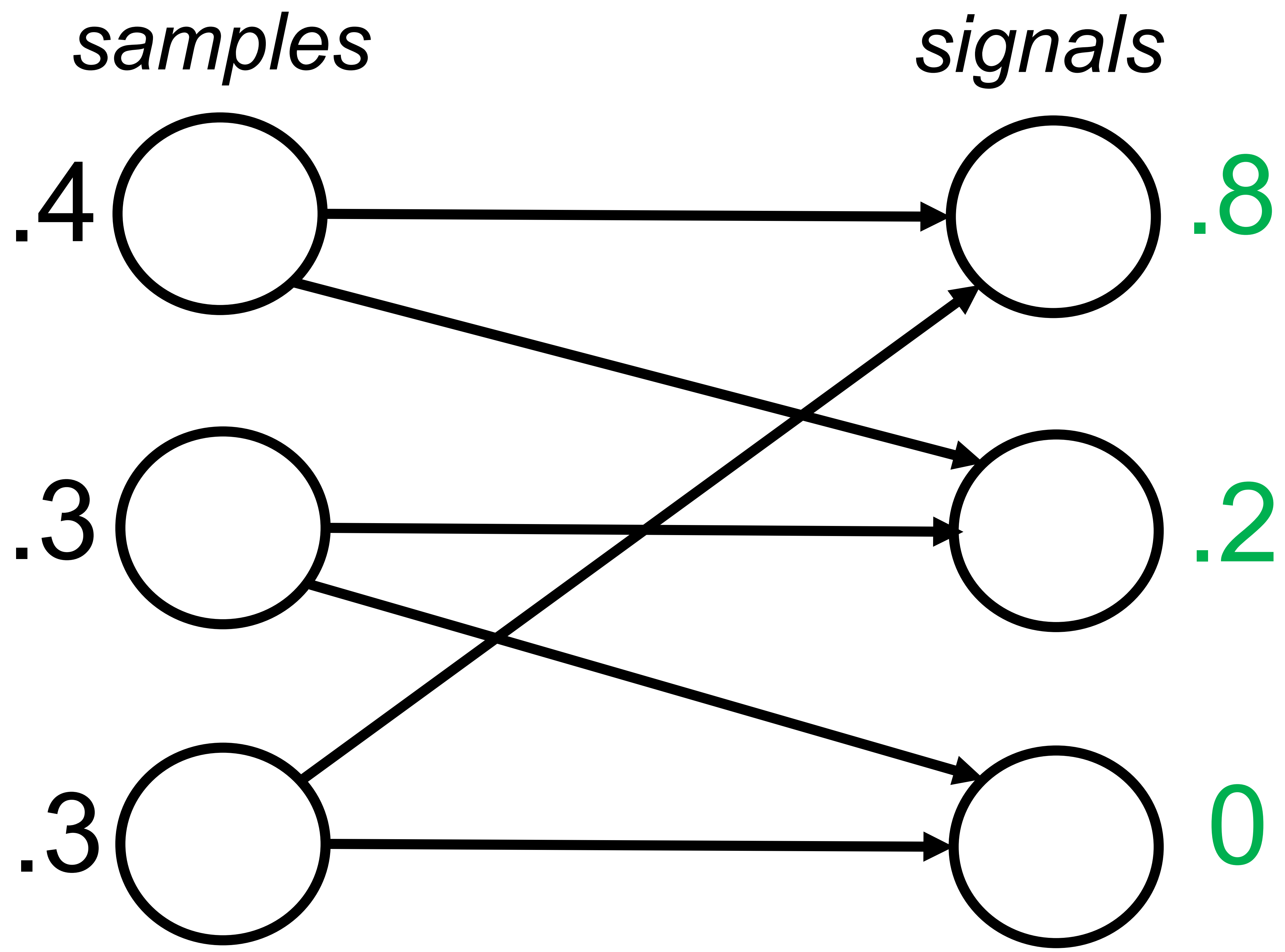
One **good** and one **bad** distribution



This subset covers .5+.2=.7 good mass and .4+.3=.7 bad mass, so it doesn't work. (What does?)

Pick a subset of the right-hand side (to accept) that maximizes
(green mass covered - black mass covered)
If positive, can (eventually) distinguish; otherwise not.
NP-hard in general.

But if we know the strategy for the good distribution (revelation principle holds):



Can place good mass on the signals side because we know the strategy

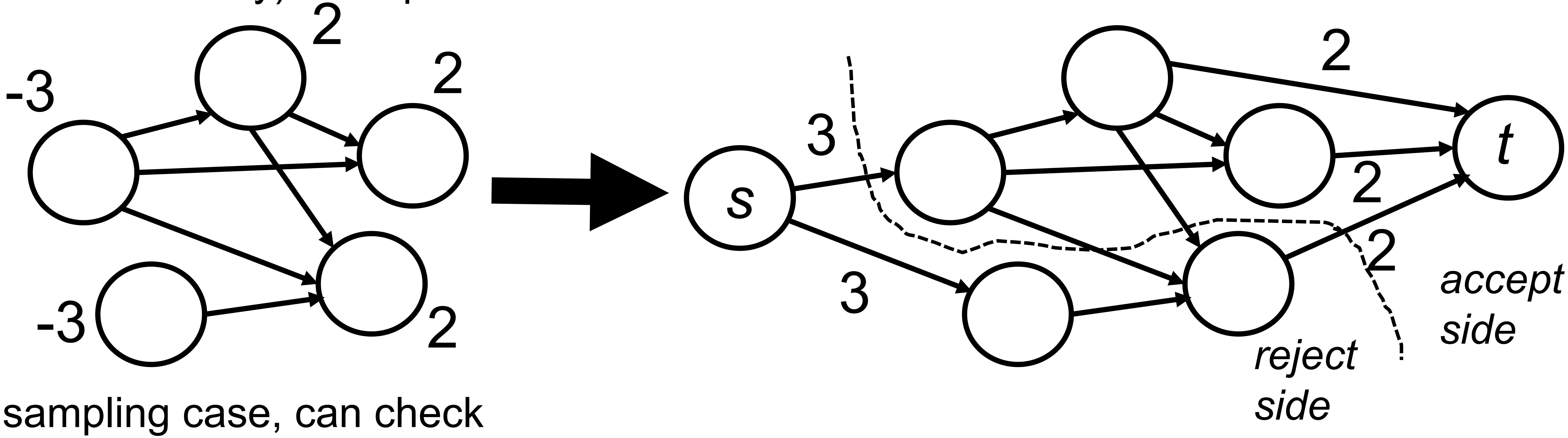
Solve as maximum flow/matching from left to right with capacities on vertices
Duality gives set of signals to accept (~Hall's marriage theorem)

Optimization: reduction to min cut

(when revelation principle holds)

types are vertices; edges imply ability to (cost-effectively) misreport

edges between types have capacity ∞



In sampling case, can check existence of edges with previous technique

Values are P(type)*value(type)

Can be generalized to more outcomes than accept/reject, if types have the same utility over them.

Conclusion

First part:

When considering correlation, small changes can have a huge effect

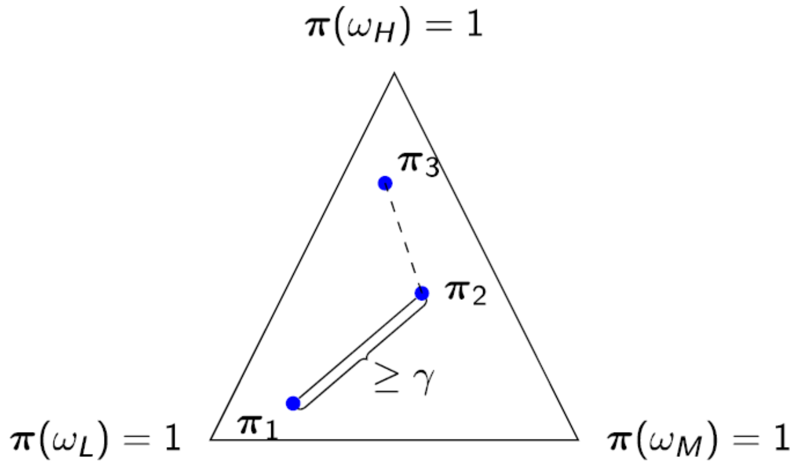
Automatically designing robust mechanisms addresses this

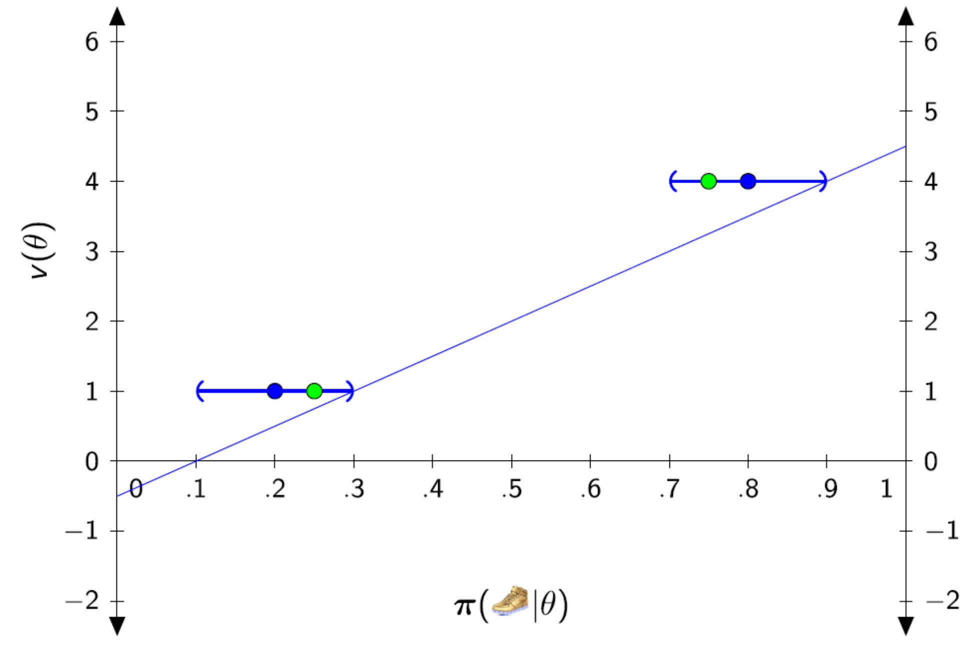
Combines well with learning (under some conditions)

0.251	0.250
0.250	0.249

v.

0.251001	0.249999
0.249999	0.249001





Second part:

With costly or limited misreporting, revelation principle can fail

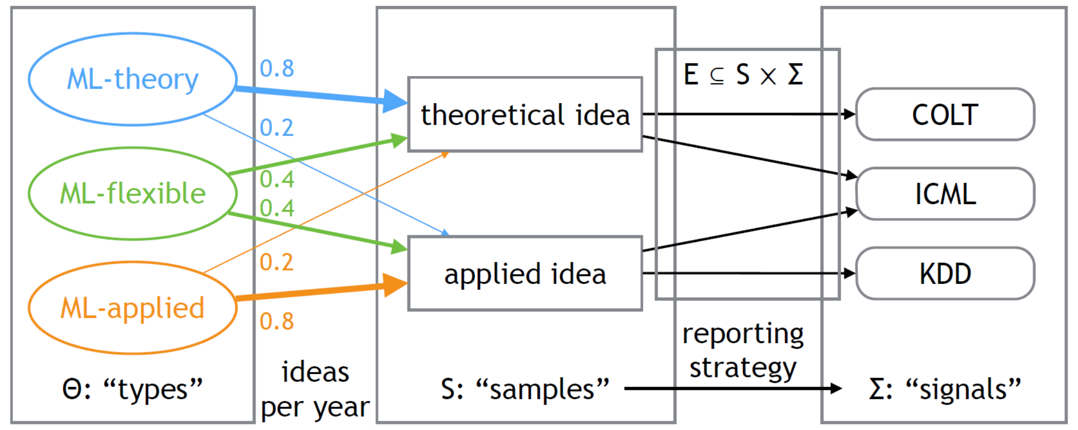
Causes computational hardness in general

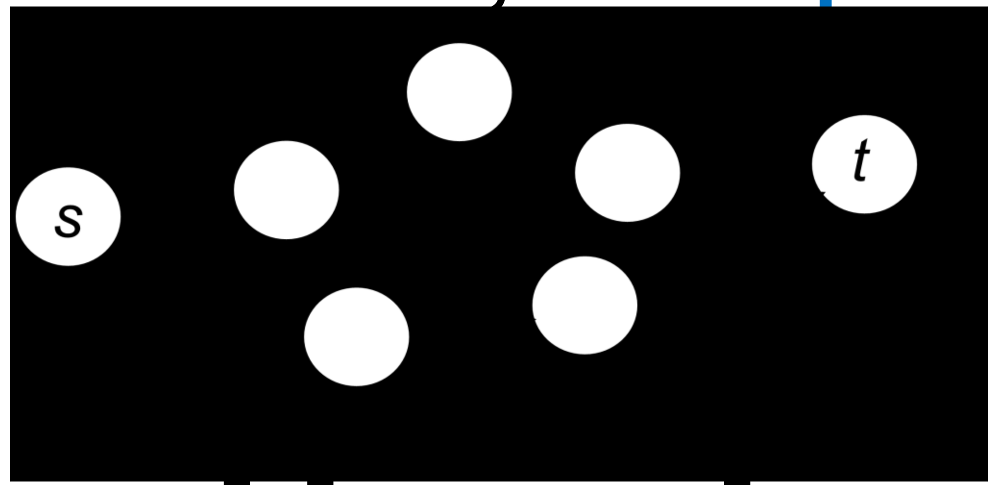
Sometimes agents report based on their samples

Some efficient algorithms for the infinite limit case; sample bounds

Effort Function$(E : T \times \hat{T} \to \mathbb{R})$**:**

	$\hat{fresh}$	$\hat{ok}$	$\hat{rotten}$
fresh	0	0	0
ok	10	0	0
rotten	30	10	0





Thank you for your attention!