Algorithmic Contract Theory and **Ambiguous Contracts**

Michal Feldman

Tel Aviv University

Introductory Workshop: Mathematics and Computer Science of Market and Mechanism Design

SLMath, September 2023

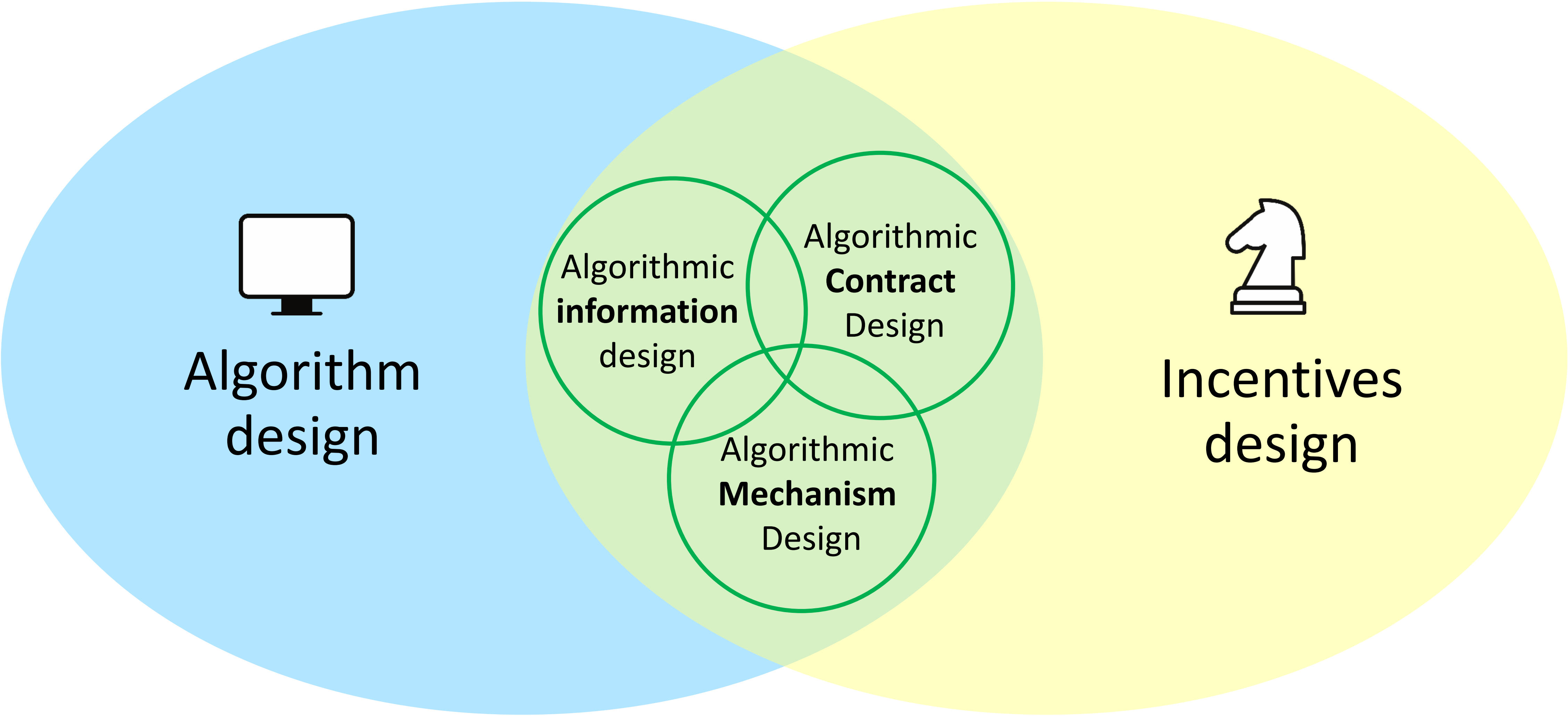
Joint work with:

Paul Duetting and Daniel Peretz

Algorithms Shape Economics and Society

- Resources we get
- Content we see
- Whom we hire
- Opportunities we face
- …

The Algorithms X Incentives Landscape



Contracts

A payment scheme (monetary or otherwise) that incentivizes strategic agents to put in effort, when their actions are hidden

- Examples:
    - Outsourcing a task to a freelancer
    - Getting students to learn

"Modern economies are held together by innumerable contracts"

[Nobel prize, 2016]

Example: Internet Marketing

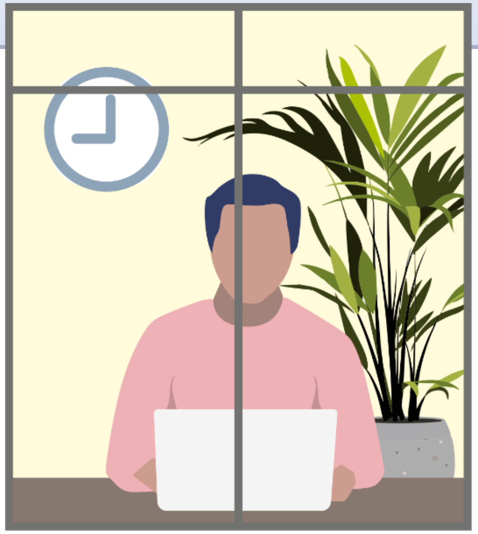
A simple contract setting:

- Marketing agent hired by website owner (principal) to promote a website
- Agent takes action (e.g., SEO, promotion campaign, influencers, bloggers, social media), principal pays

<u>Defining features</u>:

(1) Action not directly observable

(2) Limited liability (principal pays the agents)



Agent

principal

Modern Applications

Growing in scale & complexity / moving online / data-driven

- Outsourcing a task to a freelancer → freelancing platforms
- Getting students to learn → massive online courses

An algorithmic approach is relevant and timely

Algorithmic contract design plays the role for markets for services as algorithmic mechanism design plays for markets for goods

- can potentially inform better design in practice

Emerging Frontier

- Simple vs optimal contracts: [Dutting Roughgarden & Talgam-Cohen EC'19], [Alon Dutting Li Talgam-Cohen EC'23]
- Combinatorial contracts: [Lavi & Shamash EC'19], [Dutting Roughgarden & Talgam-Cohen SODA'20], [Dutting Ezra F. & Kesselheim FOCS'21], [Alon Lavi Shamash & Talgam-Cohen EC'21], [Dutting Ezra F. & Kesselheim STOC'23], [Babaioff F. Nisan EC'06], [Castiglioni et al. EC'23], [Dutting F. & Gal-Tzur, working paper]
- Contract design for social goods: [Li Immorlica & Lucier WINE'11], [Ashlagi Li & Lo Management Science'23+]
- Typed agents: [Guruganesh Schneider & Wang EC'21], [Alon Dutting & Talgam-Cohen EC'21], [Castiglioni et al. EC '21], [Castiglioni et al. EC '22], [Guruganesh Schneider & Wang EC'23]
- Learning contracts: [Ho Slivkins & Vaughn EC'14], [Cohen Deligkas & Koren SAGT'22], [Zhu et al. EC'23], [Dutting Guruganesh Schneider & Wang ICML'23]

Today’s talk: Ambiguous Contracts

- In many contractual relations, contracts are “ambiguous”. E.g.,
  - “We’ll grade one question in each problem set” (professors)
  - “we’ll compensate good drivers” (insurance companies)
  - “you’ll get promoted if you perform well” (companies)
- Motivating question: Why are ambiguous contracts so common?
- We study the power of ambiguity in contract design
  - Lots of work in economic and algorithmic design on ambiguity as a constraint
  - We study ambiguity as a tool (inspired by [Di Tillio et al. REStud 2017] who study ambiguity in auction design)

Classic Contract Design

Model [Holmström'79]

- Agent has n actions (effort levels) with costs $c_1, \ldots, c_n$

$c_1 = 0$						
$c_2 = 1$						
$c_3 = 2$						
$c_4 = 2.2$						

Model [Holmström'79]

- Agent has n actions (effort levels) with costs $c_1, \dots, c_n$
- Principal has m rewards $r_1, \dots, r_m$

	$r_1 = 1$	$r_2 = 1.1$	$r_3 = 4.9$	$r_4 = 5$	$r_5 = 5.1$	$r_6 = 5.2$
$c_1 = 0$						
$c_2 = 1$						
$c_3 = 2$						
$c_4 = 2.2$						

Model [Holmström'79]

- Agent has n actions (effort levels) with costs $c_1, \dots, c_n$
- Principal has m rewards $r_1, \dots, r_m$

- Action a_i induces distribution p_i over $\vec{r}$:
  - $p_{i,j} =$ probability that action a_i yields reward r_j

	$r_1 = 1$	$r_2 = 1.1$	$r_3 = 4.9$	$r_4 = 5$	$r_5 = 5.1$	$r_6 = 5.2$
$c_1 = 0$	3/8	3/8	2/8	0	0	0
$c_2 = 1$						
$c_3 = 2$						
$c_4 = 2.2$						

Model [Holmström'79]

- Agent has n actions (effort levels) with costs $c_1, \ldots, c_n$
- Principal has m rewards $r_1, \ldots, r_m$

- Action a_i induces distribution p_i over $\vec{r}$:
  - $p_{i,j}$ = probability that action a_i yields reward r_j

	$r_1 = 1$	$r_2 = 1.1$	$r_3 = 4.9$	$r_4 = 5$	$r_5 = 5.1$	$r_6 = 5.2$
$c_1 = 0$	3/8	3/8	2/8	0	0	0
$c_2 = 1$	0	3/8	3/8	2/8	0	0
$c_3 = 2$	0	0	3/8	3/8	2/8	0
$c_4 = 2.2$	0	0	0	3/8	3/8	2/8

Setting: (c, r, p)

Contract $\vec{t}$

- Specifies a payment $t_j \geq 0$ per reward r_j (notice defining features!)
  - $T_i = \sum_j p_{i,j} t_j =$ expected payment for action a_i
  - $R_i = \sum_j p_{i,j} r_j =$ expected reward from action a_i
- Agent chooses a_i that maximizes her expected utility

$$\underbrace{T_i}_{\text{expected payment}} - \underbrace{c_i}_{\text{cost}}$$

- Principal's expected utility from agent's choice a_i:

$$\underbrace{R_i}_{\text{expected reward}} - \underbrace{T_i}_{\text{expected payment}}$$

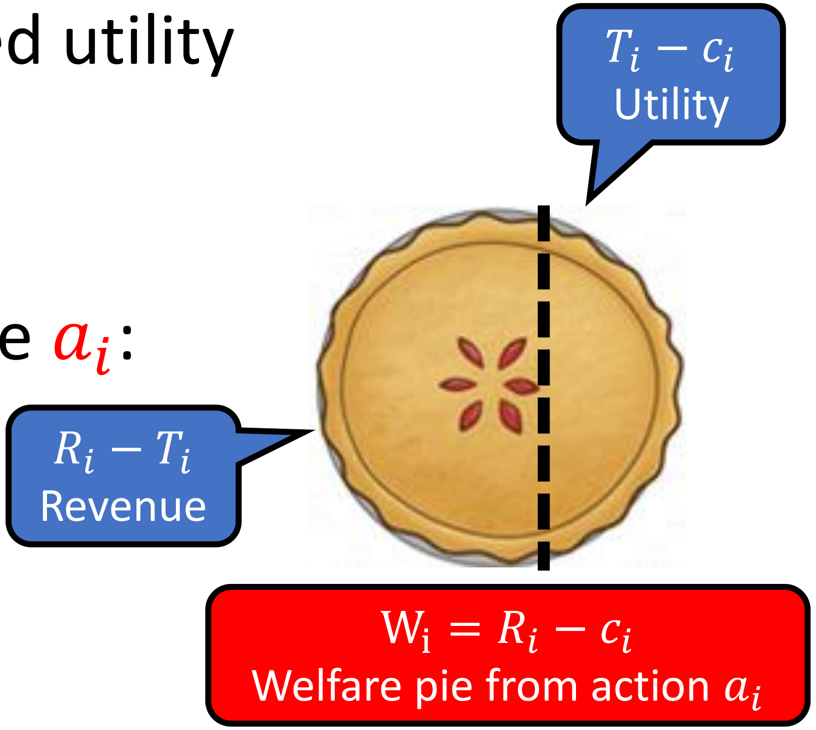
Contract $\vec{t}$

- Specifies a payment $t_j \geq 0$ per reward r_j (notice defining features!)
  - $T_i = \sum_j p_{i,j} t_j$ = expected payment for action a_i
  - $R_i = \sum_j p_{i,j} r_j$ = expected reward from action a_i
- Agent chooses a_i that maximizes her expected utility

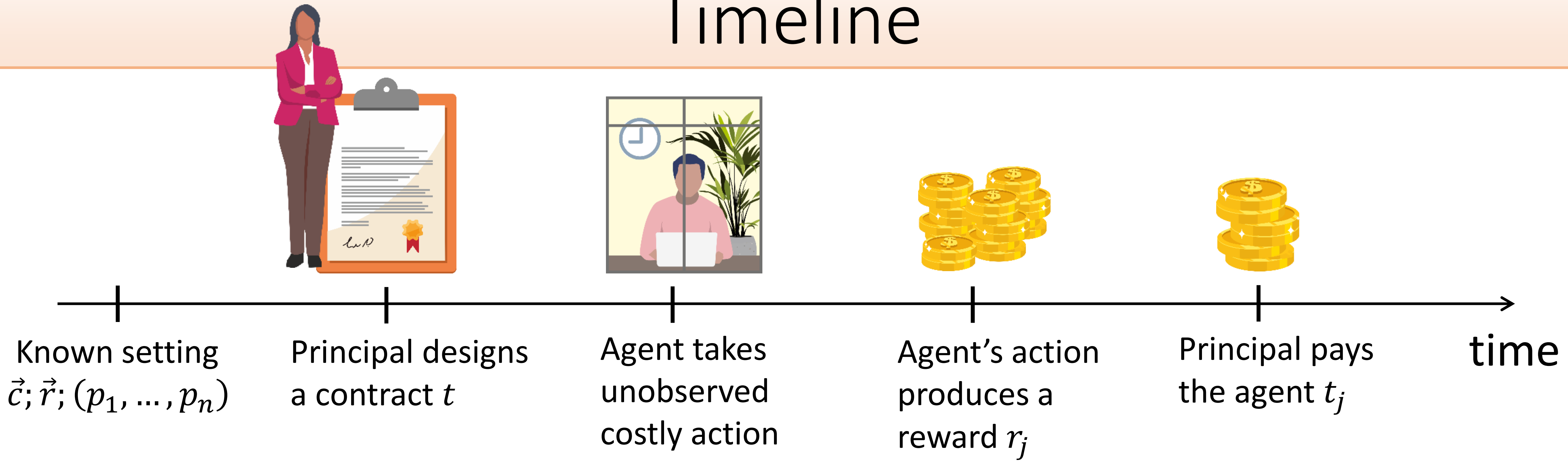
$$\underbrace{T_i}_{\text{expected payment}} - \underbrace{c_i}_{\text{cost}}$$

- Principal's expected utility from agent's choice a_i:

$$\underbrace{R_i}_{\text{expected reward}} - \underbrace{T_i}_{\text{expected payment}}$$

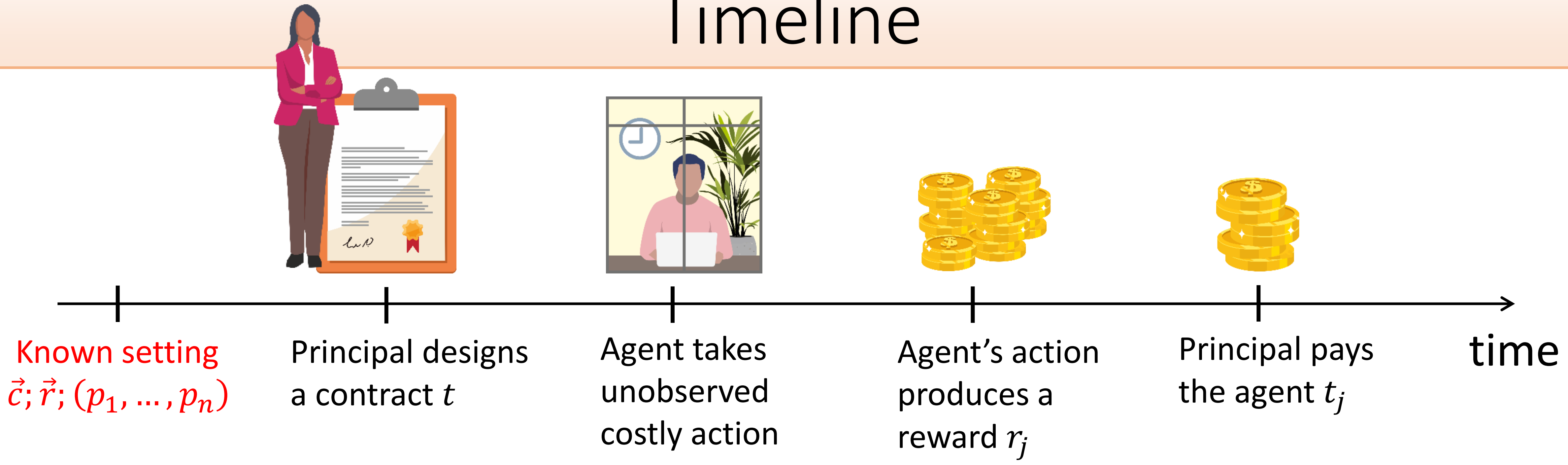


Timeline



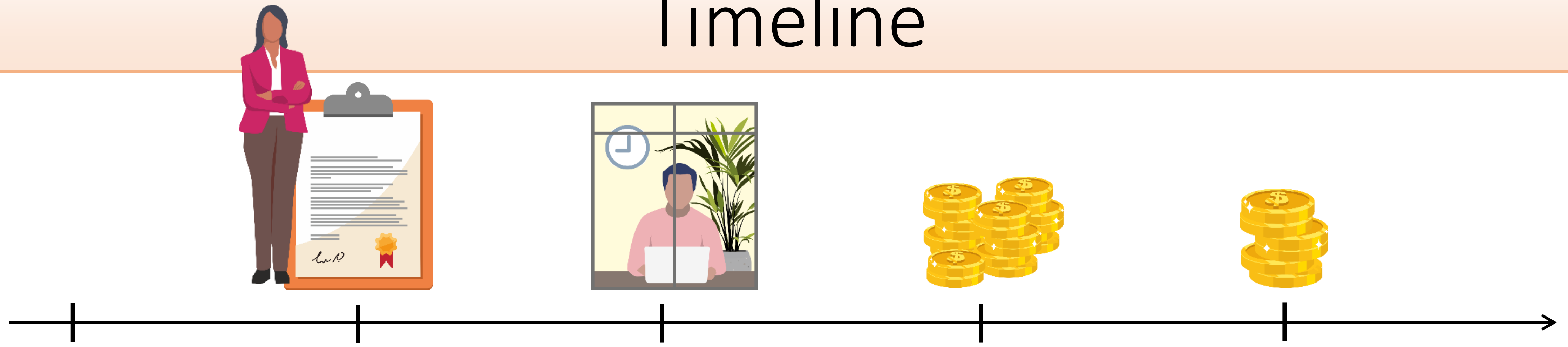
Known setting
$\vec{c}; \vec{r}; (p_1, \dots, p_n)$
Principal designs
a contract t
Agent takes
unobserved
costly action
Agent's action
produces a
reward r_j
Principal pays
the agent t_j
time

Timeline



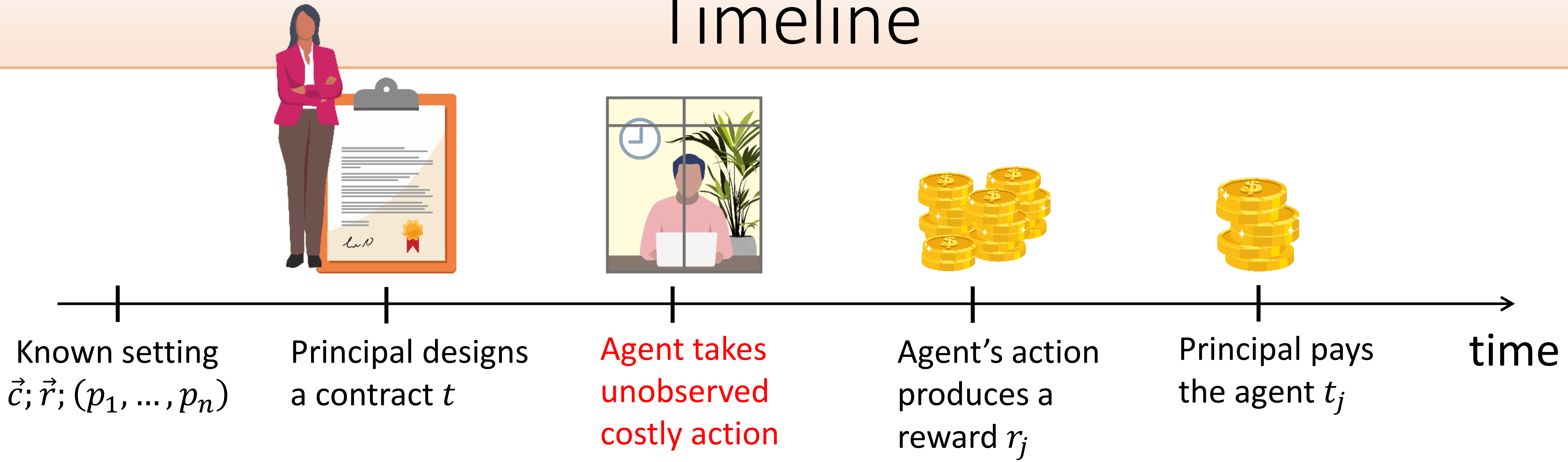
Known setting
$\vec{c}; \vec{r}; (p_1, \ldots, p_n)$
Principal designs a contract t
Agent takes unobserved costly action
Agent's action produces a reward r_j
Principal pays the agent t_j
time

Timeline



$$t = (0,1,4,2,6)$$

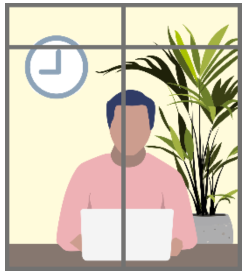
Timeline



Calculates the **expected utility** for each action $U_A(i,t) = T_i(t) - c_i$

Selects action $i^*(t) \in \arg\max_{i \in [n]} U_A(i,t)$

Timeline



Known setting $\vec{c}; \vec{r}; (p_1, \dots, p_n)$

Principal designs a contract t

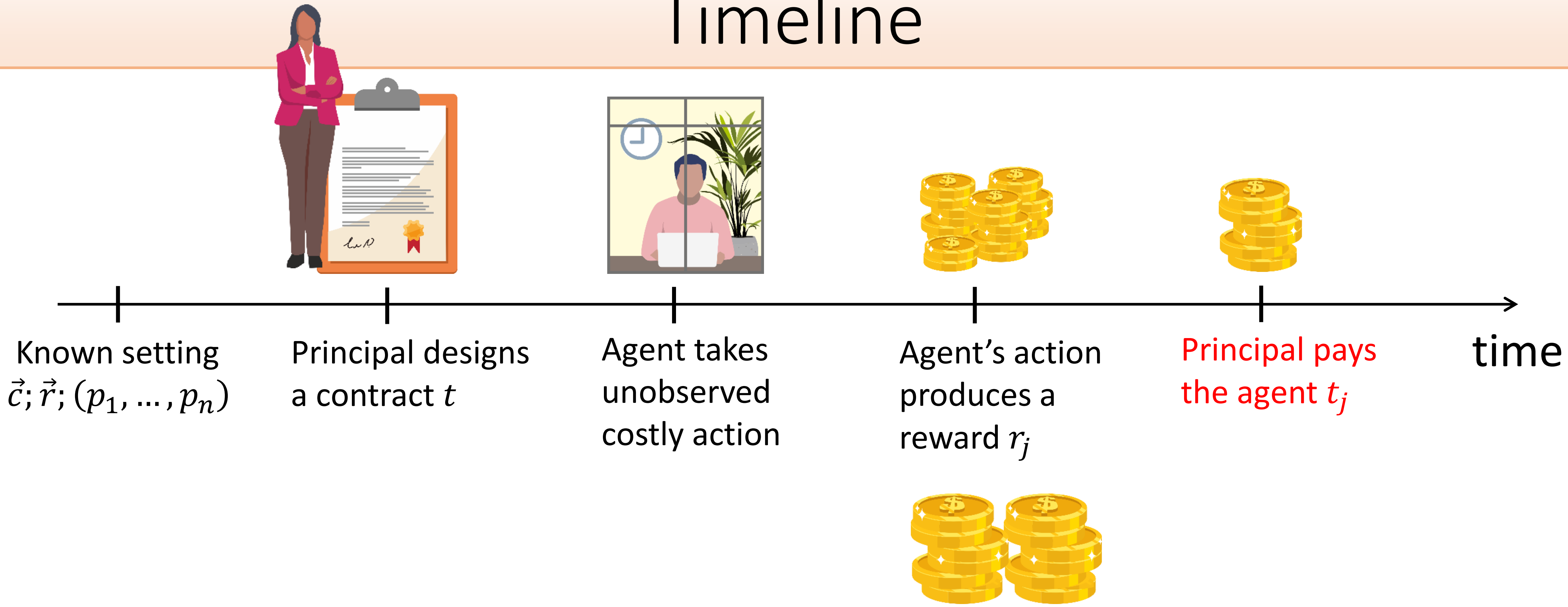
Agent takes unobserved costly action

Agent's action produces a reward r_j

Principal pays the agent t_j

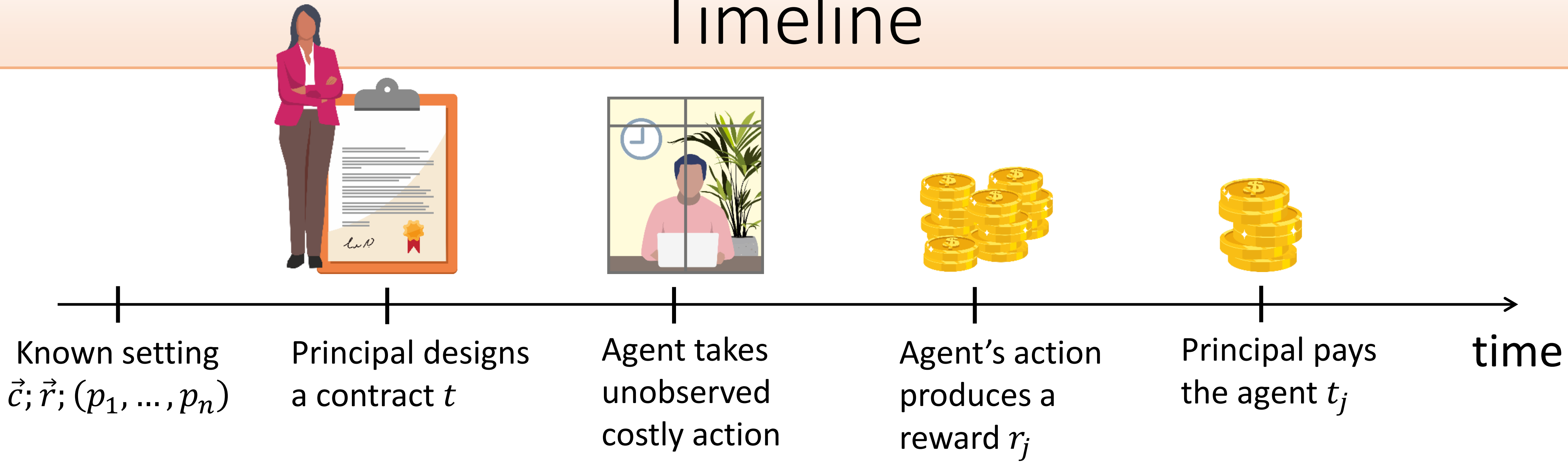
time

Timeline



Known setting
$\vec{c}; \vec{r}; (p_1, \ldots, p_n)$
Principal designs
a contract t
Agent takes
unobserved
costly action
Agent's action
produces a
reward r_j
Principal pays
the agent t_j
time

Timeline



Objective: maximize the principal's expected utility

$$U_P(t) = R_{i^*(t)} - T_{i^*(t)}(t)$$

Reward of chosen action

Expected payment of chosen action

Computing the Optimal Contract

MIN-PAY problem

- **Input**: Contract setting $(\vec{c}, \vec{r}, p)$; an action a_i
- **Output**: Minimum T_i that incentivizes a_i

Observations:

- LP solvable (minimize T_i s.t. $U_A(i, t) \geq U_A(i', t)$ for every action i')
- Optimal contract solvable via n MIN-PAY problems

Caveat: Resulting contract can be weird (e.g., non-monotone)

Ambiguous contracts

Ambiguous Contracts

- An ambiguous contract is a set of contracts $\tau = (t^1, \dots, t^k)$
  - $t^i = (t_1^i, \dots, t_m^i)$ for every i
- Agent is ambiguity averse: selects an action, $i^*(\tau)$, whose minimal expected utility across all contracts t ∈ τ is the highest

$$i^*(\tau) \in \arg\max_{i\in[n]} \underbrace{\min_{t\in\tau} U_A(i,t)}_{U_A(i,\tau)}$$

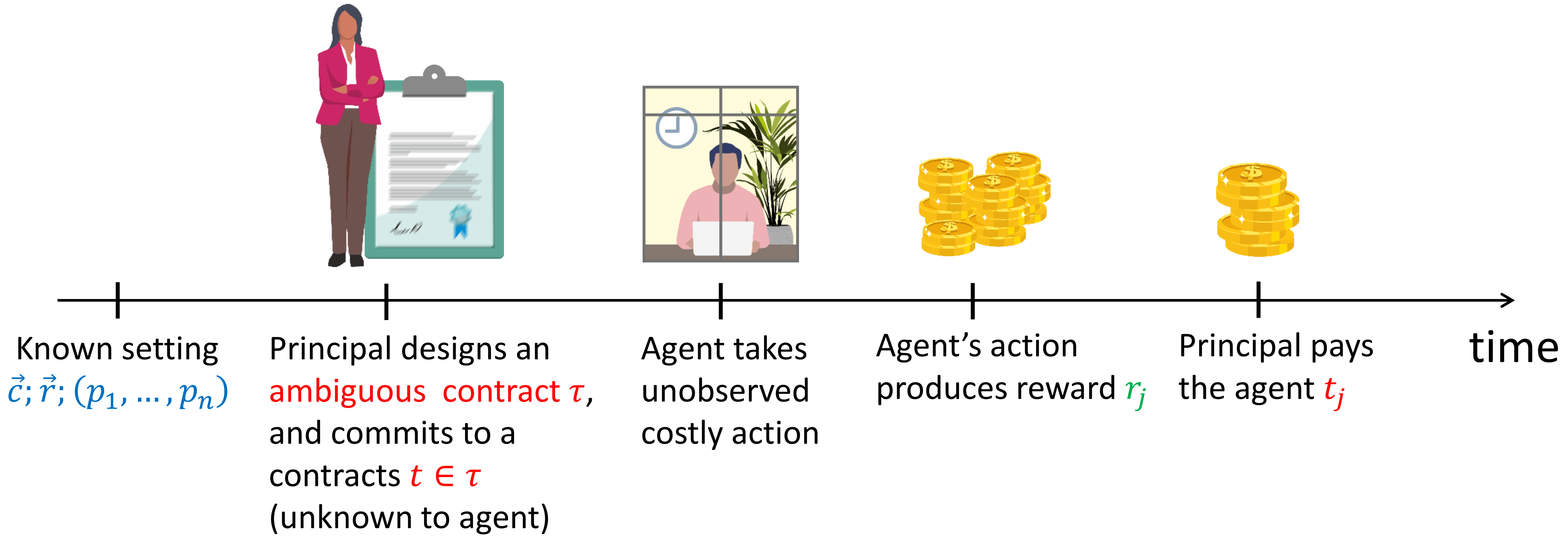
[breaking ties in favor of principal]

- Credibility: principal is indifferent between all contracts $t \in \tau$ w.r.t. action $i^*(\tau)$, i.e., for any two contracts $t^j, t^l \in \tau$:

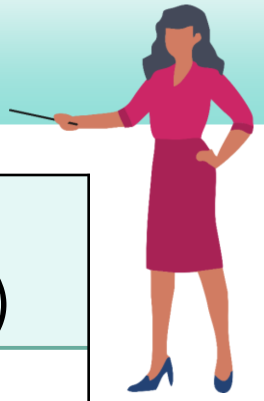
$$U_P\left(i^*(\tau), t^j\right) = U_P(i^*(\tau), t^l)$$

(also implies that $T_{i^*(\tau)}\left(t^j\right) = T_{i^*(\tau)}\left(t^l\right)$ and $U_A\left(i^*(\tau), t^j\right) = U_A\left(i^*(\tau), t^l\right)$)

Timeline


Known setting
$\vec{c}; \vec{r}; (p_1, \ldots, p_n)$
Principal designs an ambiguous contract τ, and commits to a contracts $t \in \tau$ (unknown to agent)
Agent takes unobserved costly action
Agent's action produces reward r_j
Principal pays the agent t_j
time

Example



	Cost	Get only A right (r=**1**)	Get only B right (r=**2**)	Get A&B right (r=**4**)
Lazy A	0.25	1/2	0	1/4
Lazy B	0.25	0	1/2	1/4
Work hard	1	1/8	1/8	3/4

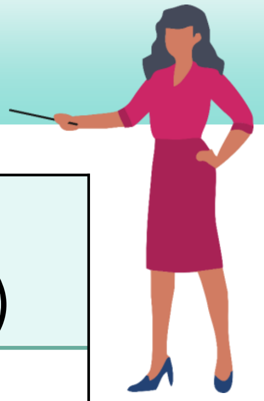
Best single contract:

- incentivizes “work hard”
- $t = (0,0,{}^{3}/_{2})$
- Expected payment = $\frac{3}{4} * \frac{3}{2} =$ 1.125

Consider ambiguous contract $\tau = (t^1, t^2)$ where $t^1 = (8,0,0)$ and $t^2 = (0,8,0)$

- “lazy A” gives -0.25 utility under t^2
- “lazy B” gives -0.25 utility under t^1
- Expected payment of both t^1 and t^2 under “work hard” = $\frac{1}{8} * 8 = 1$
- “work hard” gives utility **0** under both t^1, t^2

Example



	Cost	Get only A right (r=**1**)	Get only B right (r=**2**)	Get A&B right (r=**4**)
Lazy A	0.25	1/2	0	1/4
Lazy B	0.25	0	1/2	1/4
Work hard	1	1/8	1/8	3/4

Best single contract incentivizes “work hard” for a payment of 1.125

- Principal’s utility = 2.25

Best ambiguous contract incentivizes “work hard” for a payment of $1 < 1.125$

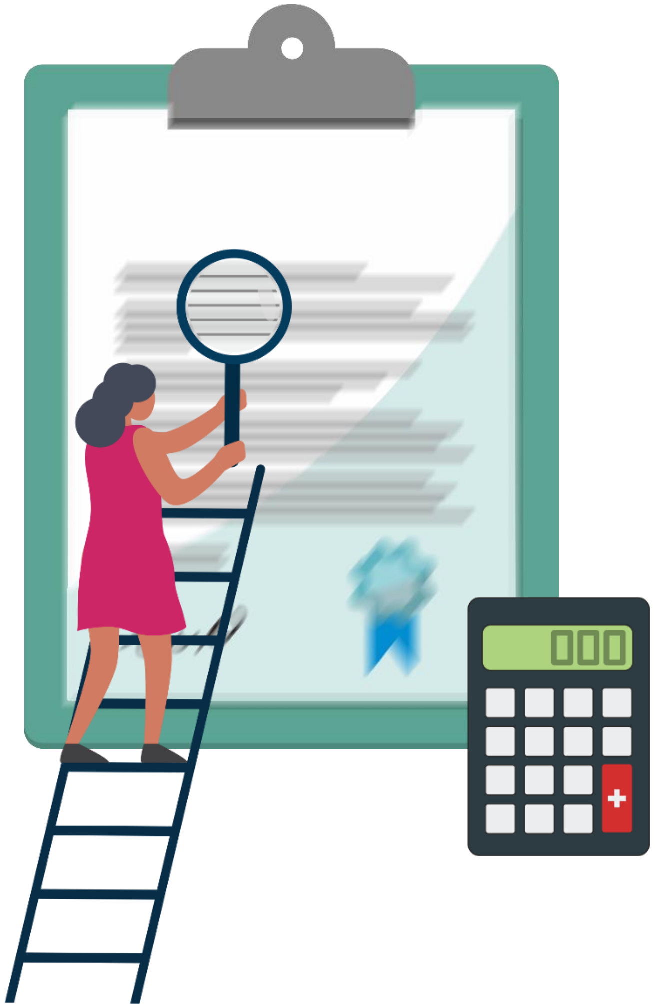
- Principal’s utility = 2.375

Upshot: principal can gain by employing ambiguous contracts

Many Questions Arise…

- What's the structure of the optimal ambiguous contract?
- What's the computational hardness of the optimal ambiguous contract?
- Are there classes of contracts that are "ambiguity-proof"?
- How much can the principal gain by employing ambiguous contracts?
- …

Structure and computation



What's the structure and computational hardness of the optimal ambiguous contract?

Single-outcome-payment (SOP) contracts

Definition: an SOP contract is one that pays only for a single outcome, e.g., $t = (0,0,4,0)$

Theorem (informal): For any ambiguous contract τ there's an "equivalent" ambiguous contract τ' composed of SOP contracts

Theorem (formal): For any ambiguous contract τ there's an ambiguous contract $\hat{\tau}$ composed of at most $\min\{n-1, m\}$ SOP contracts such that:

- $i^*(\tau) = i^*(\hat{\tau})$ [τ and $\hat{\tau}$ incentivize the same action]
- $T_{i^*(\tau)}(\tau) = T_{i^*(\tau)}(\hat{\tau})$ [they do so for the same expected payment]

Remark: an analogous theorem for monotone contracts, with step contracts instead of SOPs

Proof Idea

For every action $i \neq i^*$, there exists a contract $t^i \in \tau$ such that

$$U_A(i, t^i) \leq U_A(i^*, t^i) = U_A(i^*, \tau)$$

Plan: modify t^i to an SOP contract $\hat{t}^i$ such that:

- $T_{i^*}(\hat{t}^i) = T_{i^*}(\tau)$ (action i^* has the same E[payment] in $\hat{t}^i$ as in τ)
- $T_i(\hat{t}^i) \leq T_i(t^i)$ (action i has E[payment] in $\hat{t}^i$ at most as in t^i)

We get: $U_A(i, \hat{t}^i) \leq U_A(i, t^i) \leq U_A(i^*, \tau) = U_A(i^*, \hat{\tau})$ (so i^* is incentivized)

Constructing $\hat{t}^i$: Set $\hat{t}^i_{j_{max}} = \frac{T_{i^*}(\tau)}{p_{i^*, j_{max}}}$ and $\hat{t}^i_j = 0$ for all $j \neq j_{max}$,

where $j_{max} \in \arg\max_{j \in m} \frac{p_{i^*, j}}{p_{i,j}}$

Optimal Ambiguous Contract Computation

Theorem: An optimal ambiguous contract can be computed in $O(n^2 m)$ time

Proof idea:

For every action i, compute an optimal ambiguous contract incentivizing it

- For any action $i' \neq i$ find an outcome

$$j(i') \in \arg\max_{j \in [m]} \frac{p_{i,j}}{p_{i',j}}$$

- Calculate minimal expected payment to incentivize i over i' by paying for $j(i')$
- Update minimal expected payment to incentivize action i over all actions if needed

Generalizes maximum likelihood ratio principle, combined with waterfilling argument

$\theta \leftarrow c_i$
$S \leftarrow \emptyset$
for *each* $i' \neq i$ **do**
  $j(i') \leftarrow \text{argmax}_{j \in [m], p_{i,j} > 0} \frac{p_{i,j}}{p_{i',j}}$
  $\theta_{i'} \leftarrow p_{i,j(i')} \cdot \frac{c_i - c_{i'}}{p_{i,j(i')} - p_{i',j(i')}}$
  $\theta \leftarrow \max\{\theta, \theta_{i'}\}$
  $S \leftarrow S \cup \{j(i')\}$
end
$\tau^i \leftarrow \emptyset$
for *each* $j \in S$ **do**
  $t \leftarrow (0, \ldots, 0, t_j = \frac{\theta}{p_{i,j}}, 0, \ldots, 0)$
  $\tau^i \leftarrow \tau^i \cup \{t\}$
end
Return τ^i

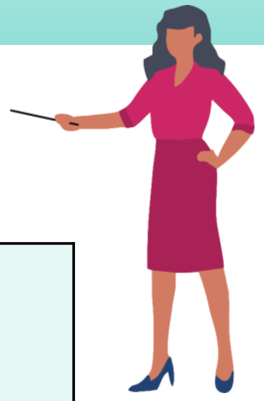
Ambiguity Proofness

Are there classes of contracts that are "immune to ambiguous contracts"?

Ambiguity Proofness

Definition: A class of contracts $\mathcal{T}$ is susceptible to ambiguity if there exists an instance, an action i and an ambiguous contract $\tau \in \mathcal{T}$, such that τ incentivizes action i at a strictly lower cost than any single contract in $\mathcal{T}$

Recall Example



	Cost	Get only A right (r=**1**)	Get only B right (r=**2**)	Get A&B right (r=**4**)
Lazy A	0.25	1/2	0	1/4
Lazy B	0.25	0	1/2	1/4
Work hard	1	1/8	1/8	3/4

Best single contract incentivizes “work hard” for a payment of 1.125

Best ambiguous contract incentivizes “work hard” for a payment of $1 < 1.125$

Ambiguity Proofness

Definition: A class of contracts $\mathcal{T}$ is susceptible to ambiguity if there exists an instance, an action i and an ambiguous contract $\tau \in \mathcal{T}$, such that τ incentivizes action i at a strictly lower cost than any single contract in $\mathcal{T}$

Theorem: A class of contracts $\mathcal{T}$ is susceptible to ambiguity iff there exist $t, t' \in \mathcal{T}$ and $x_1, x_2 \in \mathcal{R}^+$ s.t. $t(x_1) > t'(x_1)$ and $t(x_2) < t'(x_2)$

Proof (only if direction): Suppose that for every $t, t' \in \mathcal{T}$, either $t(x) \geq t'(x)$ for every $x \in \mathcal{R}^+$ or $t(x) \leq t'(x)$ for every $x \in \mathcal{R}^+$. Then, one of them yields the agent (weakly) higher expected payment (and utility). Removing it does not change the agent's and principal's utilities.

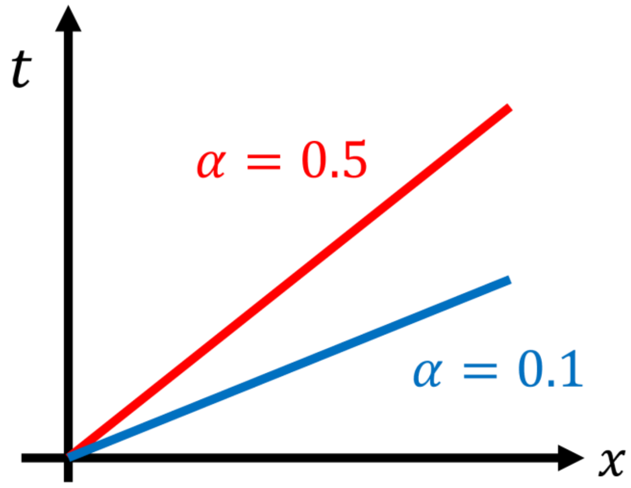
Ambiguity Proofness

Definition: A class of contracts $\mathcal{T}$ is susceptible to ambiguity if there exists an instance, an action i and an ambiguous contract $\tau \in \mathcal{T}$, such that τ incentivizes action i at a strictly lower cost than any single contract in $\mathcal{T}$

Theorem: A class of contracts $\mathcal{T}$ is susceptible to ambiguity iff there exist $t, t' \in \mathcal{T}$ and $x_1, x_2 \in \mathcal{R}^+$ s.t. $t(x_1) > t'(x_1)$ and $t(x_2) < t'(x_2)$

$t_j = \alpha r_j$ for some $\alpha \in [0,1]$

Corollary: linear contracts are ambiguity proof



A Long-Standing Puzzle

Why are simple, sub-optimal contract formats ubiquitous?

> *"It is probably the great robustness of [linear contracts] that accounts for their popularity. That point is not made as effectively as we would like by our model; we suspect that it cannot be made effectively in any traditional […] model."*
>
> [Holmström & Milgrom'87]

Linear contracts are robustly optimal w.r.t.

- Unknown actions [Carroll'15]
- Unknown distributions [Duetting Talgam-Cohen Roughgarden'19]
- Our contribution to the puzzle: Linear contracts are ambiguity-proof

Ambiguity Proofness: Mixed Strategies

Theorem: mixed strategies eliminate the power of ambiguity altogether.

	Cost	Get only A right (r=**1**)	Get only B right (r=**2**)	Get A&B right (r=**4**)
Lazy A	0.25	1/2	0	1/4
Lazy B	0.25	0	1/2	1/4
Work hard	1	1/8	1/8	3/4

For example: here, mixing between (Lazy A, Lazy B) with prob. 0.5 each, gives the agent a higher utility than "work hard" against $\tau = ((8,0,0), (0,8,0))$

Ambiguity gap

How much can the principal gain by ambiguous contracts?

Ambiguity gap

Ambiguity gap of an instance (c, r, p):

maximal principal's utility
using an **ambiguous contract**

$$\rho(c, r, p) = \frac{\max_{\tau} U_p(i^*(\tau), \tau)}{\max_{t} U_p(i^*(t), t)}$$

maximal principal's utility
using a **single contract**

Ambiguity gap

Ambiguity gap of an instance (c, r, p):

maximal principal's utility
using an **ambiguous contract**

$$\rho(c,r,p) = \frac{\max_{\tau} U_p(i^*(\tau), \tau)}{\max_{t} U_p(i^*(t), t)}$$

maximal principal's utility
using a **single contract**

Ambiguity gap of a class of instances $\mathcal{C}$: $\rho(\mathcal{C}) = \sup_{(c,r,p)\in\mathcal{C}} \rho(c,r,p)$

Max ambiguity gap over all
instances in class $\mathcal{C}$

Ambiguity gap

Ambiguity gap of an instance (c, r, p):

maximal principal's utility
using an **ambiguous contract**

maximal welfare of an action

$$\rho(c,r,p) = \frac{\max_{\tau} U_p(i^*(\tau),\tau)}{\max_{t} U_p(i^*(t),t)} \leq \frac{\max_{i\in[n]} W_i}{\max_{t} U_p(i^*(t),t)}$$

maximal principal's utility
using a **single contract**

Ambiguity gap of a class of instances $\mathcal{C}$: $\rho(\mathcal{C}) = \sup_{(c,r,p)\in\mathcal{C}} \rho(c,r,p)$

Max ambiguity gap over all instances in class $\mathcal{C}$

Main Results

Theorem: For any number n of effort levels, the ambiguity gap is at most n. Moreover, there exists such an instance with ambiguity gap $n-1$.

This bound applies also to 1^{st} vs. 2^{nd} best ratio.

Remark: For instances with two effort levels ($c_i \in \{L, H\}$ for every action i), the ambiguity gap in tightly 2.

Two Effort Levels

Key Lemma: The worst ambiguity gap is obtained for a "diagonal instance", containing m actions with cost L and welfare W_L, and one action with cost H and welfare W_H.

	$r_1 = 0$	r_2	r_3	…	r_m
$c_1 = L$	1	0	0	0	0
$c_2 = L$	rest	$\frac{W_L + L}{r_2}$	0	0	0
$c_3 = L$	rest	0	$\frac{W_L + L}{r_3}$	0	0
…	…	…	…	…	
$c_m = L$	rest	0	0	0	$\frac{W_L + l}{r_m}$
$c_{m+1} = H$					

Summary

- Algorithmic contract design is a new frontier in AGT
- Many interesting directions waiting to be explored

- Ambiguity can be used by the principal to gain higher utility
- Optimal ambiguous contracts have simple structure (SOP, step)
- Computing the optimal ambiguous contract is feasible
- Linear contracts are immune to ambiguity manipulations
- The ambiguity gap is at most the number of effort levels

Additional Resources

STOC’22 Tutorial: Algorithmic Contract Theory (Feldman and Lucier)
Dutting and Talgam-Cohen
[slides pt1, slides pt2, video]

EC’19 Tutorial: Contract Theory: A New Frontier for AGT
Dutting and Talgam-Cohen
[slides pt1, slides pt2, video pt1, video pt2]

Coming soon: Survey on Algorithmic Contract Theory
Dutting Feldman and Talgam-Cohen