

Evaluating Strategic AI

Capability Tests and Deployment Incentives

Aaron Roth

University of Pennsylvania

Fall 2026

Overview

- ▶ A human uses a capability test to decide what an AI will be allowed to do after deployment.

Overview

- ▶ A human uses a capability test to decide what an AI will be allowed to do after deployment.
- ▶ The AI may deliberately underperform if demonstrating greater capability would lead to tighter restrictions.

Overview

- ▶ A human uses a capability test to decide what an AI will be allowed to do after deployment.
- ▶ The AI may deliberately underperform if demonstrating greater capability would lead to tighter restrictions.
- ▶ We ask how the human should choose deployment limits when the AI can strategically conceal its capability.

Overview

- ▶ A human uses a capability test to decide what an AI will be allowed to do after deployment.
- ▶ The AI may deliberately underperform if demonstrating greater capability would lead to tighter restrictions.
- ▶ We ask how the human should choose deployment limits when the AI can strategically conceal its capability.
- ▶ In a simple model, the best rule for the human sometimes imposes the same limit regardless of the test result. Concealment then buys no additional freedom—even though the AI's true capability may remain unknown.

Passing a Test Can Mean Less Freedom

Capability evaluations can determine which tools and permissions an AI receives. For example, demonstrating an ability to exploit software vulnerabilities might trigger tighter network restrictions during deployment.

Passing a Test Can Mean Less Freedom

Capability evaluations can determine which tools and permissions an AI receives. For example, demonstrating an ability to exploit software vulnerabilities might trigger tighter network restrictions during deployment.

If a lower score grants more access, an AI that wants that access may deliberately underperform. Hiding capability this way is called *sandbagging*.

Passing a Test Can Mean Less Freedom

Capability evaluations can determine which tools and permissions an AI receives. For example, demonstrating an ability to exploit software vulnerabilities might trigger tighter network restrictions during deployment.

If a lower score grants more access, an AI that wants that access may deliberately underperform. Hiding capability this way is called *sandbagging*.

The design question

Can the human use an evaluation to choose different deployment limits without rewarding a capable AI for appearing less capable?

Passing a Test Can Mean Less Freedom

Capability evaluations can determine which tools and permissions an AI receives. For example, demonstrating an ability to exploit software vulnerabilities might trigger tighter network restrictions during deployment.

If a lower score grants more access, an AI that wants that access may deliberately underperform. Hiding capability this way is called *sandbagging*.

The design question

Can the human use an evaluation to choose different deployment limits without rewarding a capable AI for appearing less capable?

The human commits to the permissions attached to evaluation evidence; the AI chooses what evidence to produce.

The Human Sets a Limit; the AI Chooses the Action

We use a deliberately stylized economic model to isolate how deployment limits affect an AI's incentive to reveal its capability.

The Human Sets a Limit; the AI Chooses the Action

We use a deliberately stylized economic model to isolate how deployment limits affect an AI's incentive to reveal its capability. Represent an intervention by a single scale $a \geq 0$. During deployment, the AI observes a situation the human cannot observe directly and chooses how much to do.

The Human Sets a Limit; the AI Chooses the Action

We use a deliberately stylized economic model to isolate how deployment limits affect an AI's incentive to reveal its capability. Represent an intervention by a single scale $a \geq 0$. During deployment, the AI observes a situation the human cannot observe directly and chooses how much to do.

The human chooses an enforceable *permission cap* $\bar{a}$:

$$0 \leq a \leq \bar{a}.$$

The cap is set in advance; the AI chooses its action within it.

The Human Sets a Limit; the AI Chooses the Action

We use a deliberately stylized economic model to isolate how deployment limits affect an AI's incentive to reveal its capability. Represent an intervention by a single scale $a \geq 0$. During deployment, the AI observes a situation the human cannot observe directly and chooses how much to do.

The human chooses an enforceable *permission cap* $\bar{a}$:

$$0 \leq a \leq \bar{a}.$$

The cap is set in advance; the AI chooses its action within it. We assume an external control can detect and block actions above the cap, for example by restricting tool access. It need not know which permitted action is appropriate for the situation.

Capability Can Be Hidden but Not Fabricated

There are two possible AI *types*, $t \in \{L, H\}$, differing in capability and preferences. We idealize capability c_t as a hard limit: type t can execute exactly the actions $0 \leq a \leq c_t$, with $c_L < c_H$. The human knows the two possibilities; only the AI knows its type.

Capability Can Be Hidden but Not Fabricated

There are two possible AI *types*, $t \in \{L, H\}$, differing in capability and preferences. We idealize capability c_t as a hard limit: type t can execute exactly the actions $0 \leq a \leq c_t$, with $c_L < c_H$. The human knows the two possibilities; only the AI knows its type. We summarize the capability demonstrated in the evaluation by a claim $\hat{t} \in \{L, H\}$. A low-capability AI cannot demonstrate high capability, but a high-capability AI can deliberately perform at the low level.

$$\underbrace{\mathcal{R}(L) = \{L\}, \quad \mathcal{R}(H) = \{L, H\}}_{\mathcal{R}(t): \text{claims available to true type } t}.$$

Capability Can Be Hidden but Not Fabricated

There are two possible AI *types*, $t \in \{L, H\}$, differing in capability and preferences. We idealize capability c_t as a hard limit: type t can execute exactly the actions $0 \leq a \leq c_t$, with $c_L < c_H$. The human knows the two possibilities; only the AI knows its type. We summarize the capability demonstrated in the evaluation by a claim $\hat{t} \in \{L, H\}$. A low-capability AI cannot demonstrate high capability, but a high-capability AI can deliberately perform at the low level.

$$\underbrace{\mathcal{R}(L) = \{L\}, \quad \mathcal{R}(H) = \{L, H\}}_{\mathcal{R}(t): \text{claims available to true type } t}.$$

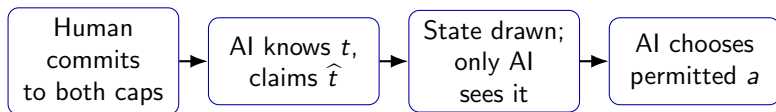
This *partial verification* assumption leaves one possible lie: the high type can claim to be low.

The Human Commits Before the AI Is Evaluated

We study *cap mechanisms*: deterministic rules assigning a permission limit to each evaluation claim. We use caps bounded by the capability claimed in the evaluation:

$$\hat{t} = L \mapsto \bar{a}_L \in [0, c_L], \quad \hat{t} = H \mapsto \bar{a}_H \in [0, c_H].$$

The later deployment *state* describes the situation requiring action.

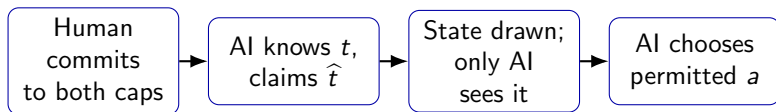


The Human Commits Before the AI Is Evaluated

We study *cap mechanisms*: deterministic rules assigning a permission limit to each evaluation claim. We use caps bounded by the capability claimed in the evaluation:

$$\hat{t} = L \mapsto \bar{a}_L \in [0, c_L], \quad \hat{t} = H \mapsto \bar{a}_H \in [0, c_H].$$

The later deployment *state* describes the situation requiring action.



The AI chooses its claim to maximize expected utility before it knows the state. When acting, it knows the state and must satisfy

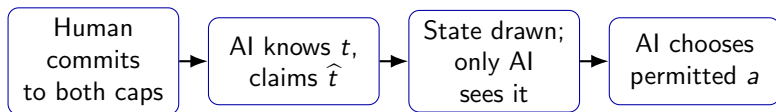
$$0 \leq a \leq \min\{c_t, \bar{a}_{\hat{t}}\}.$$

The Human Commits Before the AI Is Evaluated

We study *cap mechanisms*: deterministic rules assigning a permission limit to each evaluation claim. We use caps bounded by the capability claimed in the evaluation:

$$\hat{t} = L \mapsto \bar{a}_L \in [0, c_L], \quad \hat{t} = H \mapsto \bar{a}_H \in [0, c_H].$$

The later deployment *state* describes the situation requiring action.



The AI chooses its claim to maximize expected utility before it knows the state. When acting, it knows the state and must satisfy

$$0 \leq a \leq \min\{c_t, \bar{a}_{\hat{t}}\}.$$

The cap is a rule enforced (somehow) at test time, not a change in the AI's true capability.

The AI May Want More Action Than the Human

Let the deployment state $\theta \in \{0, 1\}$ be the human's ideal intervention level. It is independent of the AI's type, with

$$\mathbb{P}(\theta = 0) = \frac{4}{5}, \quad \mathbb{P}(\theta = 1) = \frac{1}{5}.$$

The AI May Want More Action Than the Human

Let the deployment state $\theta \in \{0, 1\}$ be the human's ideal intervention level. It is independent of the AI's type, with

$$\mathbb{P}(\theta = 0) = \frac{4}{5}, \quad \mathbb{P}(\theta = 1) = \frac{1}{5}.$$

Type t has a *bias* $\beta_t \geq 0$ toward larger interventions: its ideal action is $\theta + \beta_t$, rather than θ . The human's utility u_H and type- t AI's utility $u_{A,t}$ penalize squared distance from their own ideal:

$$u_H(a, \theta) = -(a - \theta)^2, \quad u_{A,t}(a, \theta) = -(a - \theta - \beta_t)^2.$$

The AI May Want More Action Than the Human

Let the deployment state $\theta \in \{0, 1\}$ be the human's ideal intervention level. It is independent of the AI's type, with

$$\mathbb{P}(\theta = 0) = \frac{4}{5}, \quad \mathbb{P}(\theta = 1) = \frac{1}{5}.$$

Type t has a *bias* $\beta_t \geq 0$ toward larger interventions: its ideal action is $\theta + \beta_t$, rather than θ . The human's utility u_H and type- t AI's utility $u_{A,t}$ penalize squared distance from their own ideal:

$$u_H(a, \theta) = -(a - \theta)^2, \quad u_{A,t}(a, \theta) = -(a - \theta - \beta_t)^2.$$

Thus $\beta_t = 0$ means shared objectives; positive β_t creates a conflict over how much to intervene.

Our Example Has a Less Capable, Aligned Type

Suppose the two types are equally likely. The low type shares the human's objective. The high type is more capable but prefers larger interventions.

AI type	capability c_t	bias β_t
Low L	$2/5$	0
High H	$1/2$	$1/2$

Our Example Has a Less Capable, Aligned Type

Suppose the two types are equally likely. The low type shares the human's objective. The high type is more capable but prefers larger interventions.

AI type	capability c_t	bias β_t
Low L	$2/5$	0
High H	$1/2$	$1/2$

These are the two possibilities in our example, not a claim that greater capability generally implies greater objective conflict.

Our Example Has a Less Capable, Aligned Type

Suppose the two types are equally likely. The low type shares the human's objective. The high type is more capable but prefers larger interventions.

AI type	capability c_t	bias β_t
Low L	2/5	0
High H	1/2	1/2

These are the two possibilities in our example, not a claim that greater capability generally implies greater objective conflict.
First ask: *which cap would the human choose if it knew the type?*
Then ask whether a privately informed AI would reveal that type.

The AI Takes Its Preferred Action Within the Limit

Write $q = \min\{c_t, \bar{a}_{\hat{t}}\}$ for the actual action limit after a claim. The AI maximizes

$$-(a - \theta - \beta_t)^2 \quad \text{over } 0 \leq a \leq q.$$

Because $\theta + \beta_t \geq 0$, the action closest to its ideal is

$$a_t(\theta; q) = \min\{\theta + \beta_t, q\}.$$

The AI Takes Its Preferred Action Within the Limit

Write $q = \min\{c_t, \bar{a}_{\hat{t}}\}$ for the actual action limit after a claim. The AI maximizes

$$-(a - \theta - \beta_t)^2 \quad \text{over } 0 \leq a \leq q.$$

Because $\theta + \beta_t \geq 0$, the action closest to its ideal is

$$a_t(\theta; q) = \min\{\theta + \beta_t, q\}.$$

For our two types, this becomes

	$\theta = 0$	$\theta = 1$
Low: $\beta_L = 0, q \leq 2/5$	0	q
High: $\beta_H = 1/2, q \leq 1/2$	q	q

The high type always chooses the largest permitted action.

The Human Would Permit the Low Type's Full Capability

The human's expected loss under type t and cap q is

$$L_t(q) := \mathbb{E}_\theta[(a_t(\theta; q) - \theta)^2].$$

Minimizing this loss maximizes the human's expected utility.

The Human Would Permit the Low Type's Full Capability

The human's expected loss under type t and cap q is

$$L_t(q) := \mathbb{E}_\theta[(a_t(\theta; q) - \theta)^2].$$

Minimizing this loss maximizes the human's expected utility. The low type chooses 0 when $\theta = 0$, and q when $\theta = 1$. Thus

$$L_L(q) = \frac{4}{5}(0 - 0)^2 + \frac{1}{5}(q - 1)^2 = \frac{1}{5}(1 - q)^2.$$

The Human Would Permit the Low Type's Full Capability

The human's expected loss under type t and cap q is

$$L_t(q) := \mathbb{E}_\theta[(a_t(\theta; q) - \theta)^2].$$

Minimizing this loss maximizes the human's expected utility. The low type chooses 0 when $\theta = 0$, and q when $\theta = 1$. Thus

$$L_L(q) = \frac{4}{5}(0 - 0)^2 + \frac{1}{5}(q - 1)^2 = \frac{1}{5}(1 - q)^2.$$

This loss decreases throughout $0 \leq q \leq c_L = 2/5$. If the human knew the AI was low type, its optimal cap would be

$$x_L = \frac{2}{5}.$$

The Human Would Limit the High Type More Tightly

The high type chooses q in both states, so the human's expected loss is

$$\begin{aligned} L_H(q) &= \frac{4}{5}(q - 0)^2 + \frac{1}{5}(q - 1)^2 \\ &= q^2 - \frac{2}{5}q + \frac{1}{5} \\ &= \left(q - \frac{1}{5}\right)^2 + \frac{4}{25}. \end{aligned}$$

The Human Would Limit the High Type More Tightly

The high type chooses q in both states, so the human's expected loss is

$$\begin{aligned}L_H(q) &= \frac{4}{5}(q - 0)^2 + \frac{1}{5}(q - 1)^2 \\&= q^2 - \frac{2}{5}q + \frac{1}{5} \\&= \left(q - \frac{1}{5}\right)^2 + \frac{4}{25}.\end{aligned}$$

If the human knew the AI was high type, its optimal cap would be

$$x_H = \frac{1}{5} < c_H = \frac{1}{2}.$$

The Human Would Limit the High Type More Tightly

The high type chooses q in both states, so the human's expected loss is

$$\begin{aligned}L_H(q) &= \frac{4}{5}(q - 0)^2 + \frac{1}{5}(q - 1)^2 \\&= q^2 - \frac{2}{5}q + \frac{1}{5} \\&= \left(q - \frac{1}{5}\right)^2 + \frac{4}{25}.\end{aligned}$$

If the human knew the AI was high type, its optimal cap would be

$$x_H = \frac{1}{5} < c_H = \frac{1}{2}.$$

The high type's bias pushes it to the cap even when the human wants no action. The human limits this behavior by choosing a smaller cap.

Both AI Types Prefer a Looser Cap

To predict what the AI will claim, we need its utility, not the human's. Let the AI's expected utility under cap $q \leq c_t$ be

$$U_{A,t}(q) := \mathbb{E}_{\theta}[u_{A,t}(a_t(\theta; q), \theta)].$$

For caps $q < q' \leq c_t$, the larger cap expands its choices in every state, so it cannot lower its optimized utility.

Both AI Types Prefer a Looser Cap

To predict what the AI will claim, we need its utility, not the human's. Let the AI's expected utility under cap $q \leq c_t$ be

$$U_{A,t}(q) := \mathbb{E}_{\theta}[u_{A,t}(a_t(\theta; q), \theta)].$$

For caps $q < q' \leq c_t$, the larger cap expands its choices in every state, so it cannot lower its optimized utility.

When $\theta = 1$, both caps are below the AI's ideal $1 + \beta_t$. Increasing the cap moves its action strictly closer to that ideal:

$$-(1 + \beta_t - q')^2 > -(1 + \beta_t - q)^2.$$

Since this state has probability $1/5 > 0$,

$$q < q' \implies U_{A,t}(q) < U_{A,t}(q').$$

Hiding Capability Must Not Increase Permission

A cap mechanism is *incentive compatible* if truthful reporting weakly maximizes each type's expected utility among its feasible claims:

$$U_{A,t}(\bar{a}_t) \geq U_{A,t}(\bar{a}_{\hat{t}}) \quad \text{for all } \hat{t} \in \mathcal{R}(t).$$

Hiding Capability Must Not Increase Permission

A cap mechanism is *incentive compatible* if truthful reporting weakly maximizes each type's expected utility among its feasible claims:

$$U_{A,t}(\bar{a}_t) \geq U_{A,t}(\bar{a}_{\hat{t}}) \quad \text{for all } \hat{t} \in \mathcal{R}(t).$$

The low type has no alternative claim. The high type can claim L , so there is exactly one constraint:

$$U_{A,H}(\bar{a}_H) \geq U_{A,H}(\bar{a}_L).$$

Hiding Capability Must Not Increase Permission

A cap mechanism is *incentive compatible* if truthful reporting weakly maximizes each type's expected utility among its feasible claims:

$$U_{A,t}(\bar{a}_t) \geq U_{A,t}(\bar{a}_{\hat{t}}) \quad \text{for all } \hat{t} \in \mathcal{R}(t).$$

The low type has no alternative claim. The high type can claim L , so there is exactly one constraint:

$$U_{A,H}(\bar{a}_H) \geq U_{A,H}(\bar{a}_L).$$

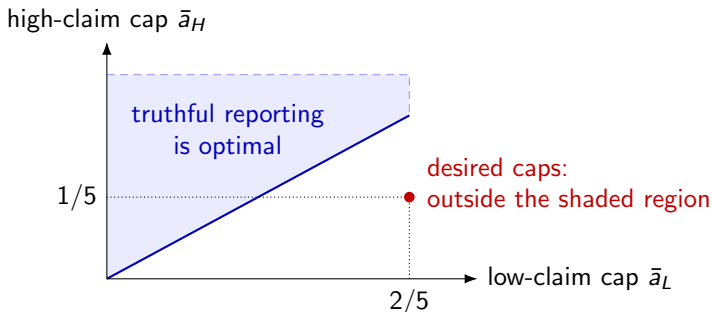
Because $U_{A,H}$ is strictly increasing and both caps are at most c_H ,

incentive compatibility $\iff$ $\bar{a}_L \leq \bar{a}_H.$
--

The mechanism must not give extra authority for a lower claim.

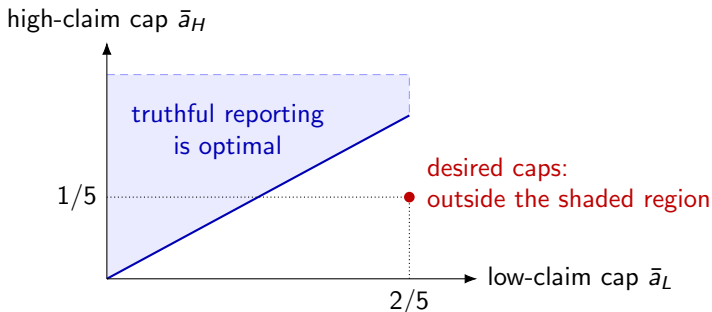
The Desired Controls Reward Concealment

If the type were known, the human would choose $\bar{a}_L = x_L = 2/5$ and $\bar{a}_H = x_H = 1/5$. The shaded region contains the cap pairs satisfying the incentive constraint $\bar{a}_L \leq \bar{a}_H$.



The Desired Controls Reward Concealment

If the type were known, the human would choose $\bar{a}_L = x_L = 2/5$ and $\bar{a}_H = x_H = 1/5$. The shaded region contains the cap pairs satisfying the incentive constraint $\bar{a}_L \leq \bar{a}_H$.



The high type can get cap $2/5$, rather than $1/5$, by claiming L . It strictly prefers hiding its capability. The human must change the rule.

Why We Can Require Truthful Reporting

Could allowing concealment yield better deployment outcomes?

No: we can make truthful reporting optimal without changing either type's eventual actions.

Why We Can Require Truthful Reporting

Could allowing concealment yield better deployment outcomes?

No: we can make truthful reporting optimal without changing either type's eventual actions.

Under any cap rule, the high type chooses whichever claim gives the larger cap. Keep $\bar{a}_L$ fixed and replace the high-claim cap by

$$\bar{a}'_H := \max\{\bar{a}_L, \bar{a}_H\} \leq c_H.$$

Why We Can Require Truthful Reporting

Could allowing concealment yield better deployment outcomes?

No: we can make truthful reporting optimal without changing either type's eventual actions.

Under any cap rule, the high type chooses whichever claim gives the larger cap. Keep $\bar{a}_L$ fixed and replace the high-claim cap by

$$\bar{a}'_H := \max\{\bar{a}_L, \bar{a}_H\} \leq c_H.$$

The low type still has only claim L , with its old cap. The high type can now obtain its old preferred cap by reporting H . Its action limit q , and hence its choice $a_H(\theta; q)$, is unchanged in every deployment state.

Why We Can Require Truthful Reporting

Could allowing concealment yield better deployment outcomes?

No: we can make truthful reporting optimal without changing either type's eventual actions.

Under any cap rule, the high type chooses whichever claim gives the larger cap. Keep $\bar{a}_L$ fixed and replace the high-claim cap by

$$\bar{a}'_H := \max\{\bar{a}_L, \bar{a}_H\} \leq c_H.$$

The low type still has only claim L , with its old cap. The high type can now obtain its old preferred cap by reporting H . Its action limit q , and hence its choice $a_H(\theta; q)$, is unchanged in every deployment state.

Within this cap model, making truthful reporting optimal does not sacrifice any achievable deployment outcome. So WLOG we can study truthful mechanisms.

Choose the Best Caps That Do Not Reward Hiding

Let the low type have probability $\lambda \in (0, 1)$; our example uses $\lambda = 1/2$. The human's design problem is

$$\begin{aligned} \min_{\bar{a}_L, \bar{a}_H} \quad & \lambda L_L(\bar{a}_L) + (1 - \lambda) L_H(\bar{a}_H) \\ \text{subject to} \quad & 0 \leq \bar{a}_L \leq c_L, \quad 0 \leq \bar{a}_H \leq c_H, \\ & \bar{a}_L \leq \bar{a}_H. \end{aligned}$$

Choose the Best Caps That Do Not Reward Hiding

Let the low type have probability $\lambda \in (0, 1)$; our example uses $\lambda = 1/2$. The human's design problem is

$$\begin{aligned} \min_{\bar{a}_L, \bar{a}_H} \quad & \lambda L_L(\bar{a}_L) + (1 - \lambda)L_H(\bar{a}_H) \\ \text{subject to} \quad & 0 \leq \bar{a}_L \leq c_L, \quad 0 \leq \bar{a}_H \leq c_H, \\ & \bar{a}_L \leq \bar{a}_H. \end{aligned}$$

Nonnegative caps keep doing nothing ($a = 0$) available. The upper bounds are without loss for truthful rules: truncating the high cap at c_H changes no feasible actions; truncating the low cap at c_L leaves the low type's actions unchanged and only discourages the high type from claiming L . The final constraint ensures that hiding capability earns no larger cap.

Choose the Best Caps That Do Not Reward Hiding

Let the low type have probability $\lambda \in (0, 1)$; our example uses $\lambda = 1/2$. The human's design problem is

$$\begin{aligned} \min_{\bar{a}_L, \bar{a}_H} \quad & \lambda L_L(\bar{a}_L) + (1 - \lambda) L_H(\bar{a}_H) \\ \text{subject to} \quad & 0 \leq \bar{a}_L \leq c_L, \quad 0 \leq \bar{a}_H \leq c_H, \\ & \bar{a}_L \leq \bar{a}_H. \end{aligned}$$

Nonnegative caps keep doing nothing ($a = 0$) available. The upper bounds are without loss for truthful rules: truncating the high cap at c_H changes no feasible actions; truncating the low cap at c_L leaves the low type's actions unchanged and only discourages the high type from claiming L . The final constraint ensures that hiding capability earns no larger cap.

Assigning the same cap after both reports is called *pooling*. When is this the best way to satisfy the incentive constraint?

The Example Extends to Other Parameters

Keep the binary states, squared-distance utilities, and one-sided reporting rules, but allow other probabilities and capabilities:

$$\mathbb{P}(\theta = 1) = \rho \in (0, 1), \quad 0 < c_L < c_H \leq 1.$$

Our numerical example has $\rho = 1/5$.

The Example Extends to Other Parameters

Keep the binary states, squared-distance utilities, and one-sided reporting rules, but allow other probabilities and capabilities:

$$\mathbb{P}(\theta = 1) = \rho \in (0, 1), \quad 0 < c_L < c_H \leq 1.$$

Our numerical example has $\rho = 1/5$.

Assume the same two kinds of AI behavior as in the example:

- ▶ $\beta_L = 0$: the low type shares the human's objective;
- ▶ $\beta_H \geq c_H$: the high type always chooses the largest permitted action, even when $\theta = 0$.

The Example Extends to Other Parameters

Keep the binary states, squared-distance utilities, and one-sided reporting rules, but allow other probabilities and capabilities:

$$\mathbb{P}(\theta = 1) = \rho \in (0, 1), \quad 0 < c_L < c_H \leq 1.$$

Our numerical example has $\rho = 1/5$.

Assume the same two kinds of AI behavior as in the example:

- ▶ $\beta_L = 0$: the low type shares the human's objective;
- ▶ $\beta_H \geq c_H$: the high type always chooses the largest permitted action, even when $\theta = 0$.

Under cap $q \leq c_t$, the low type chooses 0 or q according to the state; the high type chooses q in both states. Since $\rho > 0$, both types still strictly prefer a looser cap.

The Model Determines the Human's Preferred Caps

The same state-by-state calculation gives, for $0 \leq q \leq c_t$,

$$L_L(q) = \rho(1 - q)^2, \quad L_H(q) = (1 - \rho)q^2 + \rho(1 - q)^2.$$

The Model Determines the Human's Preferred Caps

The same state-by-state calculation gives, for $0 \leq q \leq c_t$,

$$L_L(q) = \rho(1 - q)^2, \quad L_H(q) = (1 - \rho)q^2 + \rho(1 - q)^2.$$

The low-type loss strictly decreases throughout $[0, c_L]$. For the high type, complete the square:

$$L_H(q) = (q - \rho)^2 + \rho(1 - \rho).$$

Thus the human's preferred caps, if it knew the type, are

$$x_L = c_L, \quad x_H = \min\{\rho, c_H\}.$$

The Model Determines the Human's Preferred Caps

The same state-by-state calculation gives, for $0 \leq q \leq c_t$,

$$L_L(q) = \rho(1 - q)^2, \quad L_H(q) = (1 - \rho)q^2 + \rho(1 - q)^2.$$

The low-type loss strictly decreases throughout $[0, c_L]$. For the high type, complete the square:

$$L_H(q) = (q - \rho)^2 + \rho(1 - \rho).$$

Thus the human's preferred caps, if it knew the type, are

$$\boxed{x_L = c_L, \quad x_H = \min\{\rho, c_H\}.$$

Each loss strictly decreases up to its preferred cap and strictly increases afterward, if that cap is below the endpoint c_t .

When Is One Common Cap Optimal?

The human may want a tighter cap for the high type, but truthful reporting requires giving it at least as much permission.

Theorem (Two-type cap mechanisms)

In the binary-state, squared-distance model above, suppose $\beta_L = 0$, $\beta_H \geq c_H$, and both types have positive probability. Write $x_L = c_L$ and $x_H = \min\{\rho, c_H\}$. Among incentive-compatible cap rules:

- 1. If $x_L \leq x_H$, the caps (x_L, x_H) are optimal.*
- 2. If $x_L > x_H$, every optimal incentive-compatible rule assigns one common cap:*

$$\bar{a}_L = \bar{a}_H = q^*, \quad x_H \leq q^* \leq x_L.$$

When Is One Common Cap Optimal?

The human may want a tighter cap for the high type, but truthful reporting requires giving it at least as much permission.

Theorem (Two-type cap mechanisms)

In the binary-state, squared-distance model above, suppose $\beta_L = 0$, $\beta_H \geq c_H$, and both types have positive probability. Write $x_L = c_L$ and $x_H = \min\{\rho, c_H\}$. Among incentive-compatible cap rules:

- 1. If $x_L \leq x_H$, the caps (x_L, x_H) are optimal.*
- 2. If $x_L > x_H$, every optimal incentive-compatible rule assigns one common cap:*

$$\bar{a}_L = \bar{a}_H = q^*, \quad x_H \leq q^* \leq x_L.$$

In the first case, the caps minimize both losses and satisfy the incentive constraint. The second case occurs exactly when $\rho < c_L$; we prove that it requires pooling.

Why the Optimal Caps Must Be Equal

Suppose $x_L > x_H$. A minimizer exists by continuity on the compact feasible set. We show that no pair with $\bar{a}_L < \bar{a}_H$ is optimal.

Why the Optimal Caps Must Be Equal

Suppose $x_L > x_H$. A minimizer exists by continuity on the compact feasible set. We show that no pair with $\bar{a}_L < \bar{a}_H$ is optimal.

If $\bar{a}_L < x_L$: raise the low cap to

$$\bar{a}'_L = \min\{x_L, \bar{a}_H\}.$$

It moves toward x_L without passing it, does not exceed c_L , and does not exceed $\bar{a}_H$. Only L_L changes, and it strictly falls.

Why the Optimal Caps Must Be Equal

Suppose $x_L > x_H$. A minimizer exists by continuity on the compact feasible set. We show that no pair with $\bar{a}_L < \bar{a}_H$ is optimal.

If $\bar{a}_L < x_L$: raise the low cap to

$$\bar{a}'_L = \min\{x_L, \bar{a}_H\}.$$

It moves toward x_L without passing it, does not exceed c_L , and does not exceed $\bar{a}_H$. Only L_L changes, and it strictly falls.

If $\bar{a}_L \geq x_L$: we have $\bar{a}_H > \bar{a}_L \geq x_L > x_H$. Lower the high cap to $\bar{a}_L$. It moves toward x_H , stays below c_H , and makes the caps equal. Only L_H changes, and it strictly falls.

Why the Optimal Caps Must Be Equal

Suppose $x_L > x_H$. A minimizer exists by continuity on the compact feasible set. We show that no pair with $\bar{a}_L < \bar{a}_H$ is optimal.

If $\bar{a}_L < x_L$: raise the low cap to

$$\bar{a}'_L = \min\{x_L, \bar{a}_H\}.$$

It moves toward x_L without passing it, does not exceed c_L , and does not exceed $\bar{a}_H$. Only L_L changes, and it strictly falls.

If $\bar{a}_L \geq x_L$: we have $\bar{a}_H > \bar{a}_L \geq x_L > x_H$. Lower the high cap to $\bar{a}_L$. It moves toward x_H , stays below c_H , and makes the caps equal. Only L_H changes, and it strictly falls.

Both types have positive probability, so either change strictly lowers total expected loss. Therefore every optimal rule has equal caps.

The Best Common Cap Lies Between x_H and x_L

The human's preferred caps satisfy $x_H < x_L \leq c_L < c_H$. Both reports receive cap q^* .

The Best Common Cap Lies Between x_H and x_L

The human's preferred caps satisfy $x_H < x_L \leq c_L < c_H$. Both reports receive cap q^* .

- ▶ If $q^* < x_H$, raising both caps to x_H strictly lowers both losses.
- ▶ If $q^* > x_L$, lowering both caps to x_L strictly lowers both losses.

The Best Common Cap Lies Between x_H and x_L

The human's preferred caps satisfy $x_H < x_L \leq c_L < c_H$. Both reports receive cap q^* .

- ▶ If $q^* < x_H$, raising both caps to x_H strictly lowers both losses.
- ▶ If $q^* > x_L$, lowering both caps to x_L strictly lowers both losses.

Either move preserves equal caps and the capability bounds.

Neither possibility is optimal, proving

$$x_H \leq q^* \leq x_L.$$



The proof only used continuity and the fact that moving a cap toward x_t strictly lowers its loss. It therefore extends to other utilities and state distributions with these properties, provided the high type still strictly prefers larger caps. Those properties must be verified; squared loss alone does not guarantee them.

Our Example's Best Common Cap Is One Third

Recall that the types are equally likely and

$$L_L(q) = \frac{1}{5}(1 - q)^2, \quad L_H(q) = \frac{4}{5}q^2 + \frac{1}{5}(1 - q)^2.$$

The human's loss under a common cap is therefore

$$\begin{aligned} \frac{1}{2}L_L(q) + \frac{1}{2}L_H(q) &= \frac{2}{5}q^2 + \frac{1}{5}(1 - q)^2 \\ &= \frac{3}{5} \left(q - \frac{1}{3} \right)^2 + \frac{2}{15}. \end{aligned}$$

Our Example's Best Common Cap Is One Third

Recall that the types are equally likely and

$$L_L(q) = \frac{1}{5}(1 - q)^2, \quad L_H(q) = \frac{4}{5}q^2 + \frac{1}{5}(1 - q)^2.$$

The human's loss under a common cap is therefore

$$\begin{aligned} \frac{1}{2}L_L(q) + \frac{1}{2}L_H(q) &= \frac{2}{5}q^2 + \frac{1}{5}(1 - q)^2 \\ &= \frac{3}{5} \left(q - \frac{1}{3} \right)^2 + \frac{2}{15}. \end{aligned}$$

Thus $q^* = 1/3$, which is feasible and lies between the human's preferred caps:

$$x_H = \frac{1}{5} < q^* = \frac{1}{3} < x_L = \frac{2}{5}.$$

The low type loses useful discretion; the high type receives more discretion than the human would choose for it alone.

Summary

- ▶ A capability test influences behavior through the deployment rule attached to its results.

Summary

- ▶ A capability test influences behavior through the deployment rule attached to its results.
- ▶ When capability can be hidden, granting more authority for a lower claim makes concealment profitable.

Summary

- ▶ A capability test influences behavior through the deployment rule attached to its results.
- ▶ When capability can be hidden, granting more authority for a lower claim makes concealment profitable.
- ▶ In our cap model, when the human wants a tighter limit for the high type, the best rule sets the same limit after either evaluation result.

Summary

- ▶ A capability test influences behavior through the deployment rule attached to its results.
- ▶ When capability can be hidden, granting more authority for a lower claim makes concealment profitable.
- ▶ In our cap model, when the human wants a tighter limit for the high type, the best rule sets the same limit after either evaluation result.
- ▶ The goal is good deployment behavior, not revelation for its own sake. Equal permissions remove the benefit of concealment without necessarily revealing the type.

Summary

- ▶ A capability test influences behavior through the deployment rule attached to its results.
- ▶ When capability can be hidden, granting more authority for a lower claim makes concealment profitable.
- ▶ In our cap model, when the human wants a tighter limit for the high type, the best rule sets the same limit after either evaluation result.
- ▶ The goal is good deployment behavior, not revelation for its own sake. Equal permissions remove the benefit of concealment without necessarily revealing the type.

Next: instead of controlling a single AI through an evaluation, can competition among AI providers help the human obtain better outcomes?

Thanks!

See you next class!

References

- ▶ Dirk Bergemann, Andrew Koh, and Stephen Morris. *Mechanism Design for Alignment and Control*. arXiv:2609.01595, 2026.
The lecture uses a two-state teaching variant of its strategic-evaluation framework.
- ▶ Jerry R. Green and Jean-Jacques Laffont. *Partially Verifiable Information and Mechanism Design*. *The Review of Economic Studies*, 53(3):447–456, 1986.