How to (not) Analyze Cryptographic Protocols using Game Theory

Jesper Buus Nielsen

Main Points

- *Idealizing crypto*: Replace real-life crypto tools by formal objects like term algebras or oracles to make analysis of a protocol easier
- Common in cryptography
  - known to be sound in the usual crazy-versus-stupid models
- Researcher have been idealizing crypto tools for the sake of game theoretic analysis too
  - That is typically *not* sound

Terminology: Computational Solution Concept

- Takes *computation feasibility* into account
  - Examples: Only allow polynomial time computable strategies, price computation via the utility function, discounting, …
- Allows the use of (imperfect) cryptography
  - Example: When your opponent uses encryption the deviation which makes one guess at his secret key and uses the key to break the protocol if the guess is correct gives you a small advantage, so go for $\frac{W}{2}$-NE for negligible small $\frac{W}{2}$ to allow stability
  - Example: Utility of key-guessing smaller than the price of the computation or discounted away

Terminology:
Game Theoretic Solution Concept

- A solution concept which allows arbitrary strategies

Idealizing Crypto

- (Very simple) **idealized signatures**:
  - The world has a global signing oracle *O* which all parties have access to
  - **Sign**: A party P_i can send *sign(m)* to *O* which stores *(i, (i, m))* [read P_i has a signature on *m* from P_i]
  - **Transfer**: If P_k inputs *trans((i,m),n)* to *O* and *(k, (i, m))* is stored in *O*, then *O* stores *(n, (i, m))*
  - **Verify**: If P_k inputs *verify(i,m)* to *O* and *(k, (i, m))* is stored in *O* then *O* outputs *accept* otherwise *reject*
- Possible to show that any cryptographic protocol which is secure when using these idealized signatures is equally secure when they are replaced by real signatures
  - Up to negligible 
  - PKI + unforgeable signatures + UC framework

Why? (1/3)

- A possible *solution heuristic*:
  - Idealize the crypto tools in a protocol and then apply your favorite GT solution concept to the idealized protocol
  - Since the idealized protocol does not rely on computation crypto tools it is free of the deviations with negligibly small advantage which disturb most known GT solution concepts
- Implicit assumption: Guarantees that there are no problems besides key-guessing-like deviations

Why? (2/3)

- Might guide the development of computational solution concepts:
  - Given GT solution concept *X* try to develop a computational version *CX*
  - Then check if *CX* produces solutions similar to the solutions *X* produces for the idealized protocol
- Assumption: The computational version should behave like the pure GT notion

Why? (3/3)

- Modular analysis of complex protocols
- Given a protocol using both signature and encryption:
  - First idealize both primitives and give a hopefully simple analysis of the idealized protocol
  - Show that plugging in real signatures preserves solutions
  - Show that plugging in real encryption preserves solutions
  - Conclude that the real protocol has the same solutions as the ideal protocol

Hope!

- Often a cryptographic analysis (honest parties versus corrupted parties) of an idealized protocol can be proven to give sound conclusions about the real-life protocol
  - Signatures ☺
  - Encryption ☺
  - Zero-knowledge proof of knowledge ☺
  - Zero-knowledge proof of correctness ☹

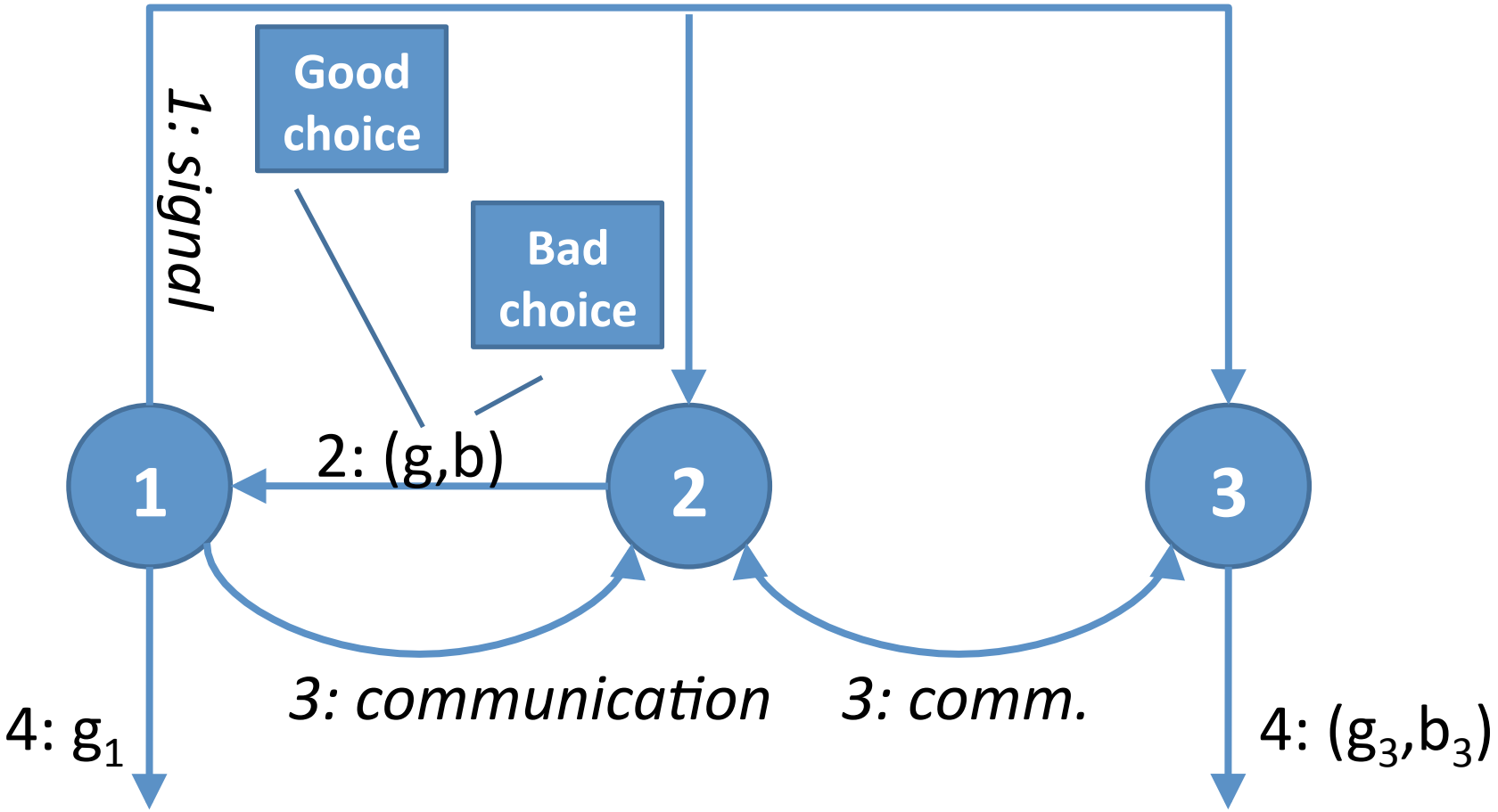
Claims

- The *solution heuristic* is likely to give wrong conclusions
- Comparison to idealization is not a good sanity check for computational solution concepts
- Computational solution concepts must be developed cautiously and have their own computational epistemologies
- After developing good computational solution concepts idealization is possible as a tool for modular analysis

“Proof by Example”

- Will try to argue my point by “solving” a small game in three different settings
- Will see that we get dramatically different solutions depending on whether we idealize crypto or not
- And the solution called by the idealized analysis is arguably the wrong one

Overview


1: signal
Good choice
Bad choice
2: (g,b)
1
2
3
3: communication
3: comm.
4: g_1
4: (g_3,b_3)

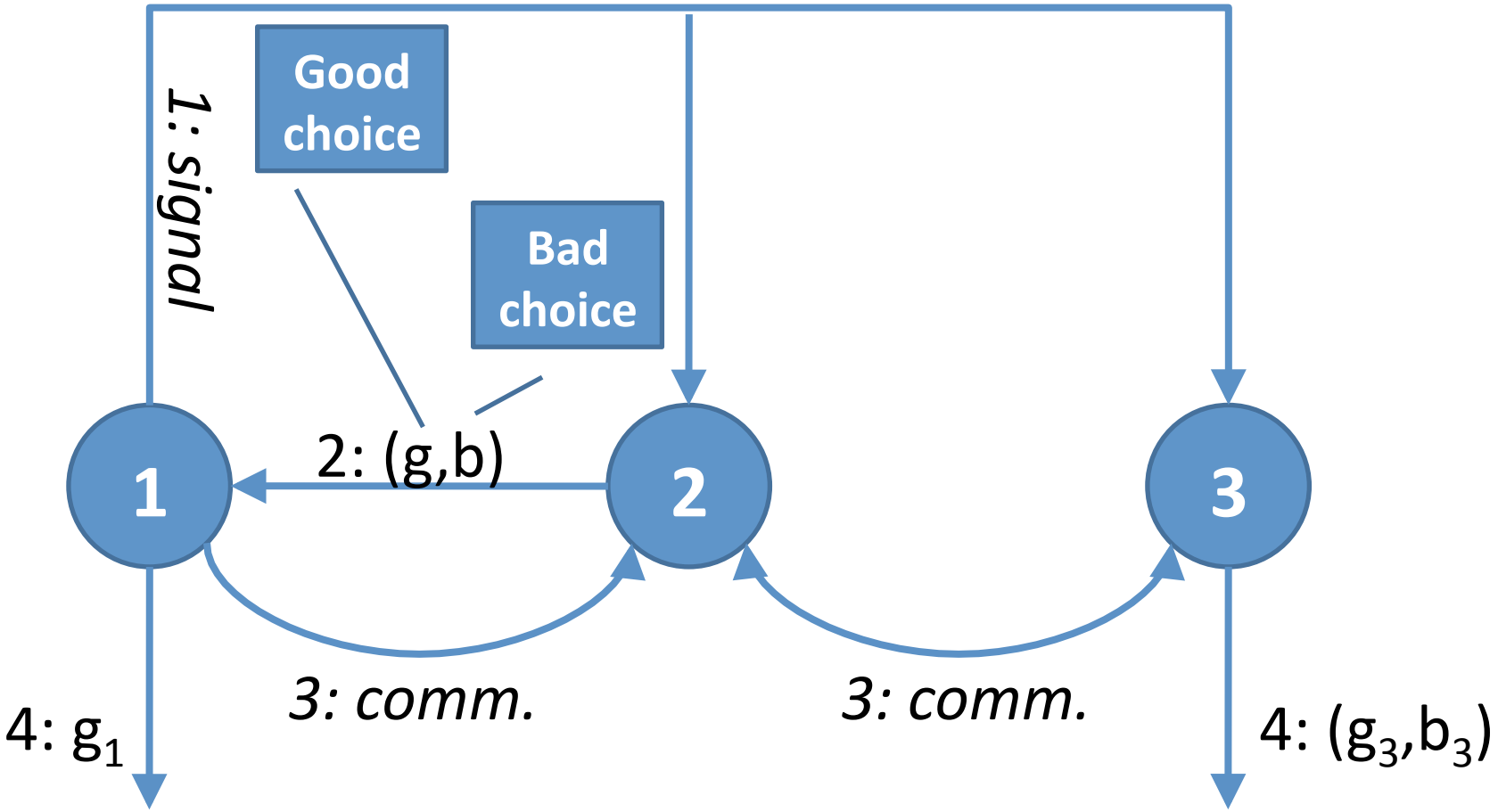
A Few Pennies

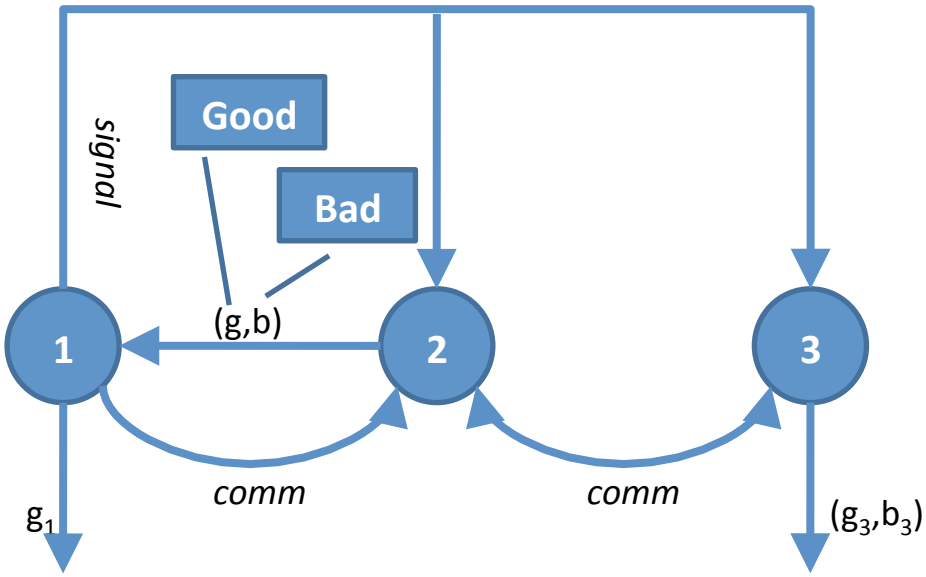
- ***Good&bad:*** P_2 plays $g \in \{1,2,3\}$ and $b \in \{1,2,3\}\backslash\{g\}$
- ***Guess:*** P_1 plays $g_1 \in \{1,2,3\}$
- ***Guess:*** P_3 plays $g_3 \in \{1,2,3,a\}$ and $b_3 \in \{1,2,3\}$
- **Abstain**: If P_3 plays a all parties get utility 0
- **Avoid bad**: If $g_1=b$ or $g_3=b$ then P_1 and P_3 die and P_2 wins the world
- **Know bad**: Same if P_3 does not abstain and $b_3 \in b$
- **Coordinate**: If $g_1,g_3 \in \{1,2,3\} \backslash\{b\}$ and $b_3=b$, then P_1 and P_3 get a positive utility from $g_1=g_3$ *but* P_2 *prefers* $g_1 \in g_3$
  - P_1 has negative utility on $g_1 \in g_3$ but P_3 does not, though he prefers $g_1=g_3$
  - And P_1 prefers to match on g

Played in a Network

- Before P_2 specifies *(g,b)*:
  - P_1 can send a signal to P_3
    - Also seen by P_2
- Then P_1 learns *(g,b)* but P_3 does not
- After P_2 specifies *(g,b)*:
  - P_1 can send a message to P_2
    - Not seen by P_3
  - P_2 and P_3 can communicate with each other
    - Not seen by P_1

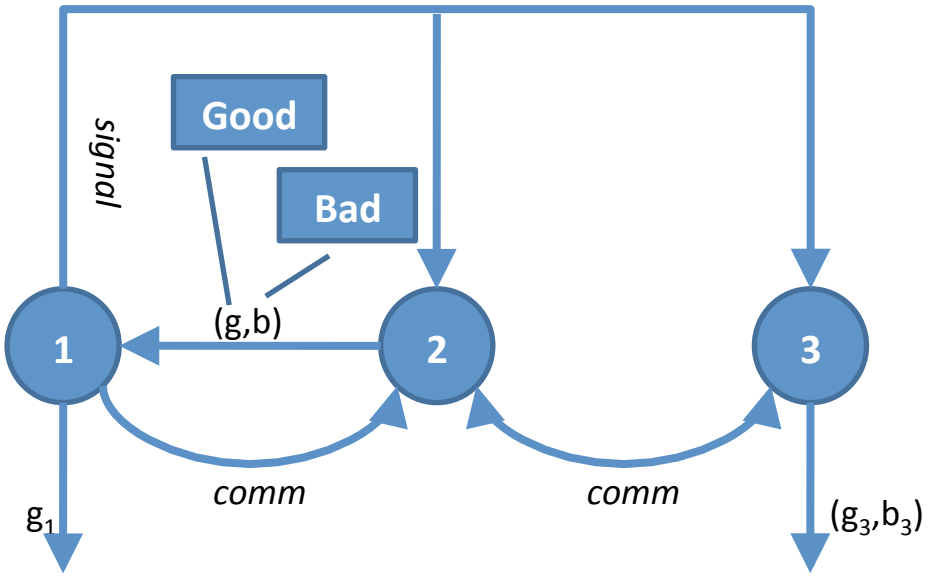
Recap


1: signal
Good choice
Bad choice
2: (g,b)
1
2
3
3: comm.
3: comm.
4: g_1
4: (g_3,b_3)

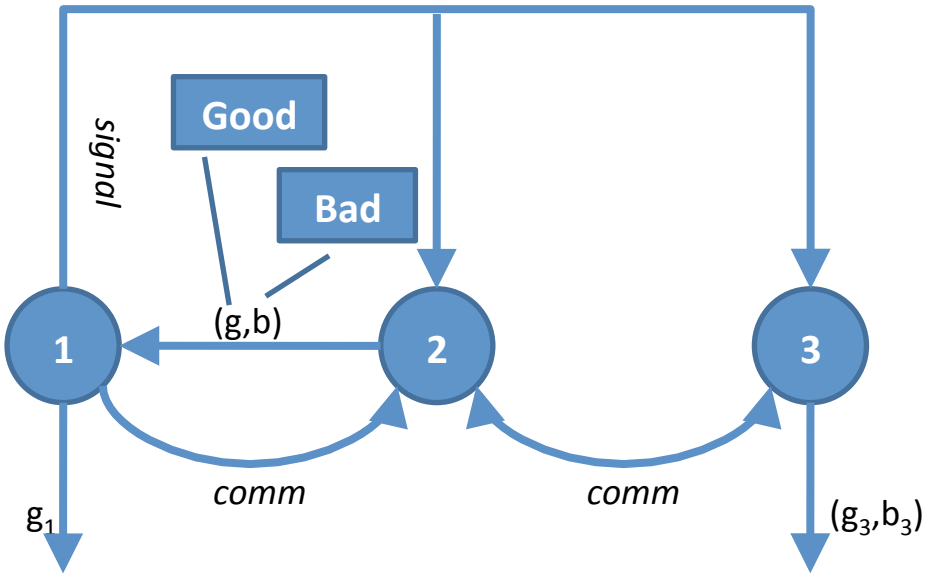


- **Abstain**: $g_3=a$:
  $u_1=u_2=u_3=0$
- **Avoid**: $g_1=b$ or $g_3=b$:
  $u_1=u_3=-$[illegible] , $u_2=$[illegible]
- **Know**: g_3[illegible]a , b_3[illegible]b:
  $u_1=u_3=-$[illegible] , $u_2=$[illegible]
- **Otherwise**:

- g_1[illegible] g_3: $u_1=-2$ $u_2=3$ $u_3=0$
- $g_1=g_3=g$: $u_1=1$ $u_2=1$ $u_3=1$
- $g_1=g_3$[illegible] g: $u_1=0$ $u_2=2$ $u_3=1$

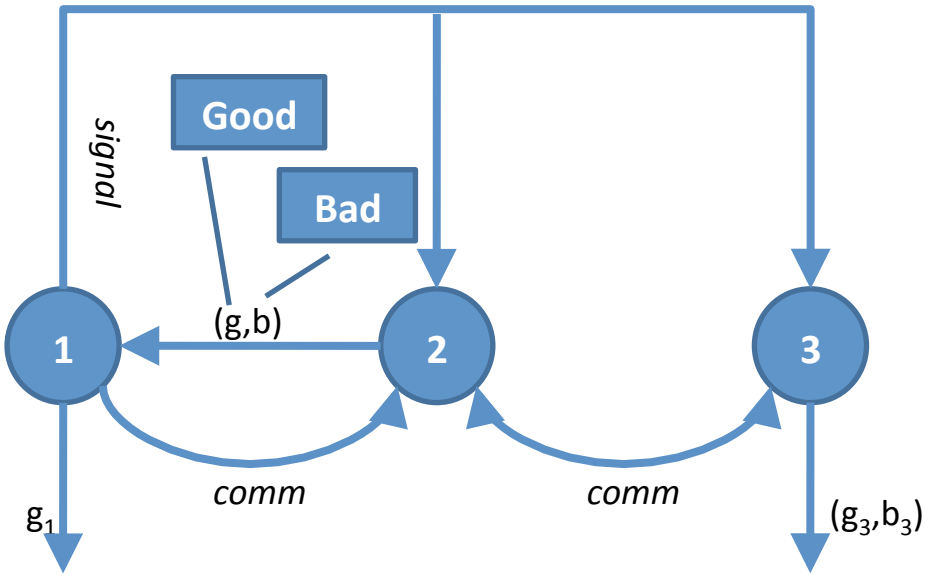


- ***Abstain, Avoid, Know***
- $g_1 \not\sim g_3$: -2 3 0
- $g_1=g_3=g$: 1 1 1
- $g_1=g_3 \not\sim g$: 0 2 1



- ***Abstain, Avoid, Know***
- g_1 ≀ g_3: -2 3 0
- $g_1=g_3=g$: 1 1 1
- $g_1=g_3$ ≀ g: 0 2 1

- Will draw conclusions from this game by informally solving it using "common knowledge of rationality" in the following settings:
  1. Arbitrary strategies
  2. Idealized signatures
  3. Poly-time strategies

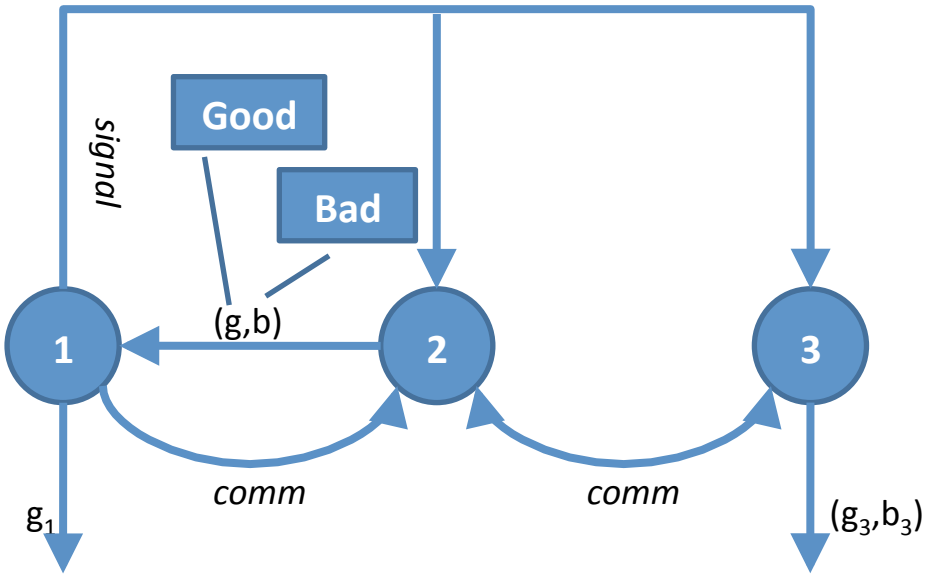


- ***Abstain, Avoid, Know***
- g_1 g_3: -2 3 0
- $g_1=g_3=g$: 1 1 1
- $g_1=g_3$ g: 0 2 1

- If g g in some NE (with positive probability)
  & ing
  t
  s on
  V $_2$
  had played according to the NE

Common knowledge of rationality

Always abstain

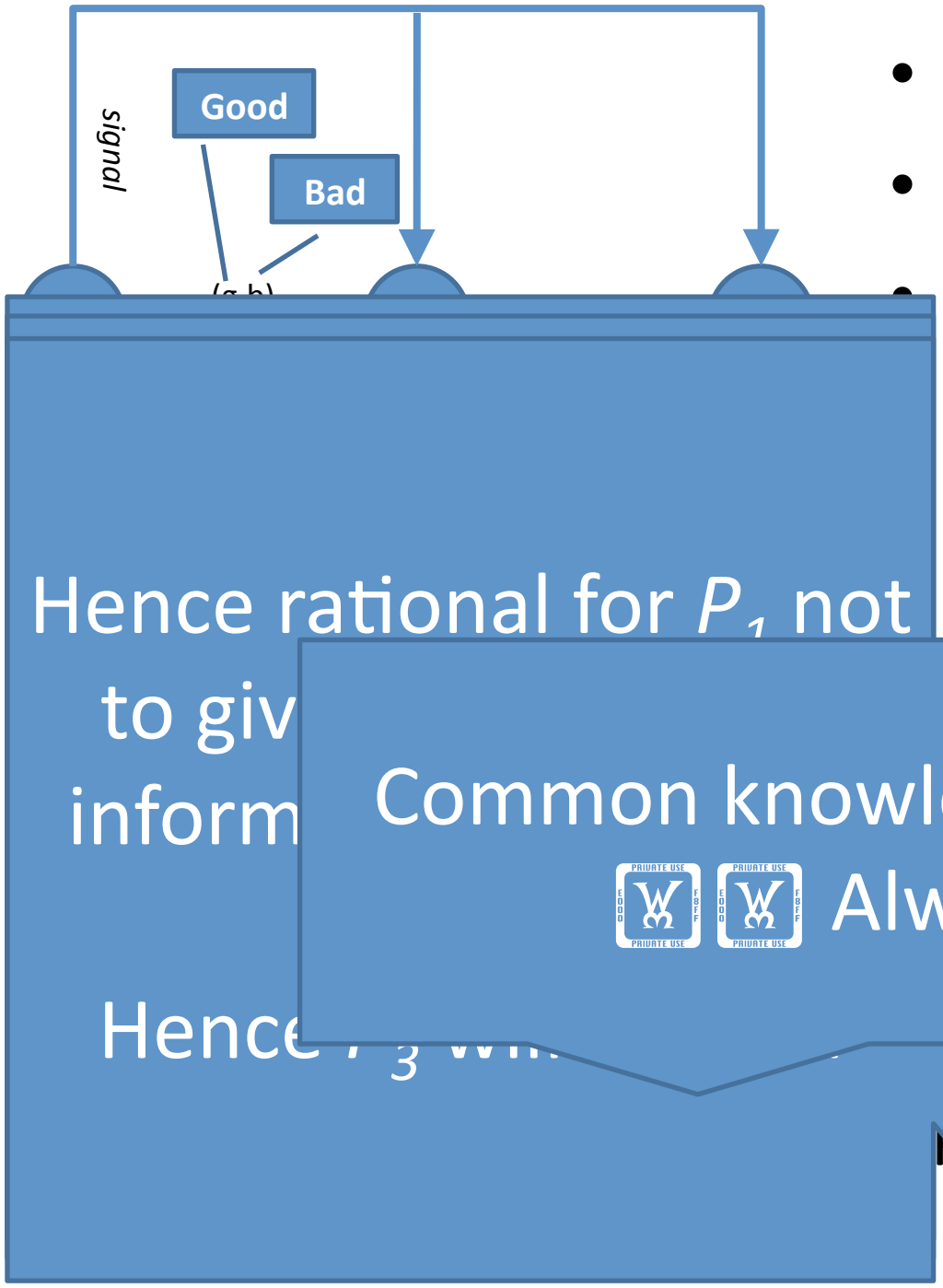


- ***Abstain, Avoid, Know***
- g_1 g_3: -2 3 0
- $g_1=g_3=g$: 1 1 1
- $g_1=g_3$ g: 0 2 1

- "Rationalizable":
- P_1
- P_2
- P_3
- P_3: if $ver_{vk}((g,\cdot),3)=accept$ play $g_3=g$ and $b_3=b$ otherwise $g_3=a$

Common knowledge of rationality

Never abstain



- ***Abstain, Avoid, Know***
- g_1 g_3: -2 3 0
- g_1=g_3=g: 1 1 1
- g_1=g_3 g: 0 2 1

What went Wrong?

- Idealization of signatures have been proven sound in cryptography, so what went wrong?
- P_2 can prove to P_3 that P_1 sent b while hiding g and thus renegotiate P_3 into a strategy which is an advantage for P_2
- Cryptography has a centralized adversary who controls and coordinates all corrupted parties, hence the use of cryptography “internal to the deviation” does not give extra power to the adversary compared to the idealized case

Conclusion 1

- The heuristic solution concept can easily give "very" wrong solutions
  - A three-party, simultaneous mutual conflict/ mutual advantage of cooperation setting, like the one used, can arise in many settings and might even be subtly hidden
- Seems hard to judge whether a protocol can be soundly analyzed using the heuristic, so better just abstain from doing it

Conclusion 2

- It does not seem as a way out to make more involved idealizations which, e.g., allows "spliting" of signatures as we did in the example
  - The idealization would probably end up being more complicated than the real-life tool
  - The idealization would have to be head on: allow all possible uses and misuses and nothing else to hope for soundness

Conclusion 3

- Comparison to how GT solution concepts behave on idealized protocols is *not* a good sanity check for proposed computational solution concepts
  - In our case the computational notion should exactly give another solution

Conclusion 4

- There does not seem to be a way around cautiously developing computational solution concepts and try to give epistemic models based on bounded rationality

The Good News

- Modular analysis via idealization is possible for Computational Nash Equilibrium (CNE)
  - Only reasons via single agent deviation
  - Hence crypto cannot be used to facilitate deviations
- In [Peter Bro Miltersen, Jesper Buus Nielsen, Nikos Triandopoulos: Privacy-Enhancing Auctions Using Rational Cryptography. CRYPTO 2009] we show a cryptographic auction protocol to be a CNE via a sound idealizing of the crypto and a game theoretic analyzing of the idealized protocol

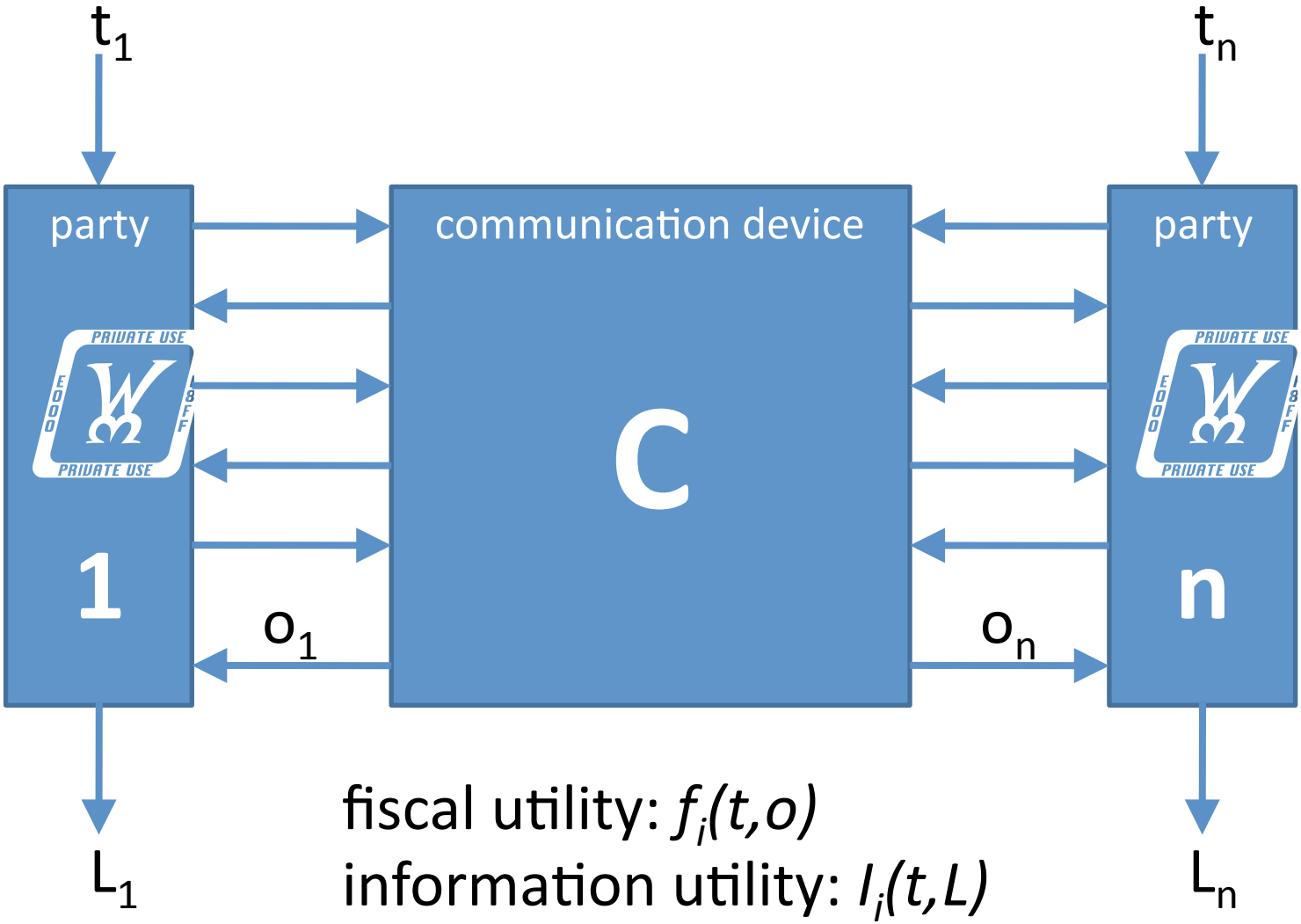
Setting

- The goal in [MNT09] was to give a game-theoretic analysis of a protocol which *n* parties can run among themselves on the Internet to emulate a trusted mediator
  - They should end up having signed contracts from all other parties on their outcomes to avoid disputes after the game is over
  - The parties are allowed to have privacy concerns, e.g., to prefer to keep their type secret over leaking it

Analytic Technique

- We use a notion of protocol game, which allows to model both a trusted mediator and the Internet in a unified manner
- We then relate the properties of the real-life protocol to the mediated case and conclude that the real-life protocol is as stable as the mediated case and gives the same utility profile
  - Implies that it leaks no more information, as the utility associated to information loss/collection is captured in the utility functions

Protocol Games

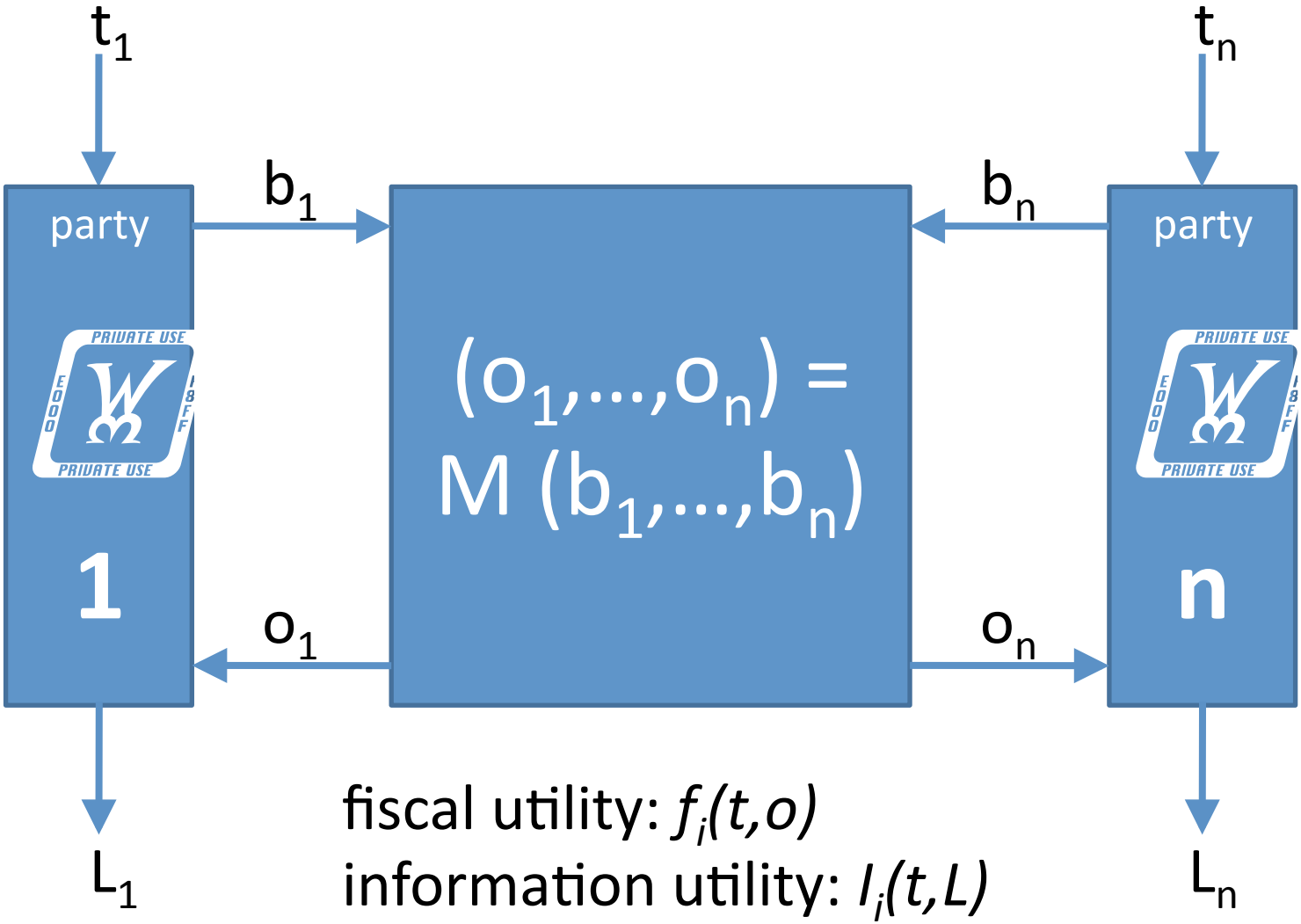


fiscal utility: $f_i(t,o)$

information utility: $I_i(t,L)$

utility: $u_i(t,o,L) = f_i(t,o)+I_i(t,L)$

Mediation

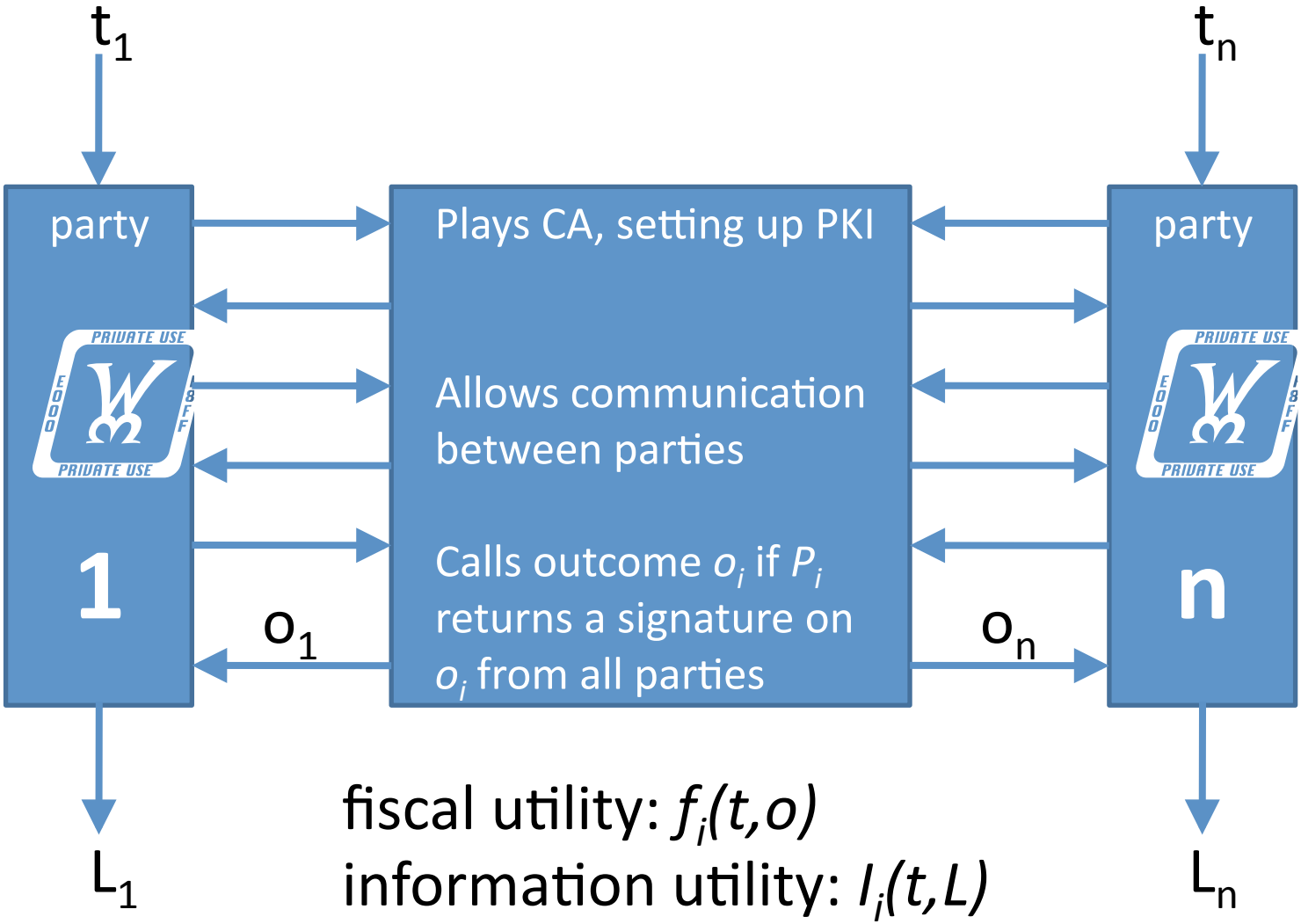


fiscal utility: $f_i(t,o)$

information utility: $I_i(t,L)$

utility: $u_i(t,o,L) = f_i(t,o)+I_i(t,L)$

Internet Contract Games



fiscal utility: $f_i(t,o)$

information utility: $I_i(t,L)$

utility: $u_i(t,o,L) = f_i(t,o)+I_i(t,L)$

Important Design Choices

- Same type profile *T* makes sense in all settings
- Outcome is called by the device as last round of outputs, so well-defined in all settings
- Local information is output be the parties, so well-defined in all settings
- So, same *u=f+I* makes sense in all settings
- We can keep types and utilities fixed and relate different strategies in different settings
  - We can talk about whether it is better to play some given strategy in the real-life setting than it is to play some other strategy in the ideal setting

Nash Implementation

- Fix T and $f = (f_1,\ldots,f_n)$
- We say that (C,Γ) is a ***t-resilient*** **privacy-enhanced Nash implementation** of (D,Γ), *written* (C,Γ) $\Gamma_{t,T,r}$ (D,Γ), if for all admissible I and $u = f+I$ it holds that:
- **No less utility**: For all P_i: $u_i(T,C,\Gamma)$ Γ $u_i(T,D,\Gamma)$ - Γ
- **No more incentive to deviate**: For all $C\Gamma\{1,\ldots,n\}$ with $|C|\Gamma t$ and all $\Gamma_C{}^*$ there exists $\Gamma_C{}^*$ such that $u_i(T,D,(\Gamma_C{}^*,\Gamma_{-C}))$ Γ $u_i(T,C,(\Gamma_C{}^*,\Gamma_{-C}))$ - Γ for all $i\Gamma C$

The Result in the Paper

- We construct for each mechanism *M* a contract game for the Internet which is an *(n-1)*-resilient privacy-enhanced Nash implementation of the ideally mediated setting for *M* if all parties have *ex interim* strict rationality

Property 1 of Nash Implementation

- If *(C,* $\mathcal{W}$*)* is an $\mathcal{W}$-NE (tolerating collusions of size *t*) and *(C,* $\mathcal{W}$*)* $\mathcal{W}_{t,T,r}$ *(D,* $\mathcal{W}$*)* then *(D,* $\mathcal{W}$*)* is an $\mathcal{W}$-NE (tolerating collusions of size *t*)
  - Allows to lift analysis from an ideal setting to a real-life setting
- So, any $\mathcal{W}$-NE for the mediated setting (with *ex interim* strict rationality) is also a $\mathcal{W}$-NE in the Internet contract game

Property 2 of Nash Implementation

- If $(C, \heartsuit) \heartsuit_{t,T,r} (D, \heartsuit)$ and
  $(D, \heartsuit) \heartsuit_{t,T,r} (E, \heartsuit)$
  then $(C, \heartsuit) \heartsuit_{t,T,r} (E, \heartsuit)$
- This allows a modular analysis going from the mediated setting to the Internet setting via gradually more refined settings (introducing, e.g., one crypto primitive at a time)

…

- The notion of Nash implementation is a trivial adoption of the notion NE from intra-game analysis to inter-game analysis
- Yet it allows to do modular analysis with much the same flavor as modular analysis in crypto via idealization
- There is justified hope that other good computational solution concepts will allow similar lifting to inter-game analysis and hence allow modular analysis
- We just need some good computational solution concepts…