

The Pareto Frontier for Random Mechanisms

Timo Mennle & Sven Seuken,
University of Zurich, Switzerland

Meeting of COST Action on Computational Social Choice,
Istanbul Bilgi University, Turkey,
November 2, 2015

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

The screenshot shows the Doodle interface for a 'CERG-Outing' event. The event is scheduled for November 2015, with seven possible dates: Do 5, Mo 9, Di 10, Mi 11, Do 12, Mi 18, and Do 19. Seven participants are listed: Sven Seuken, Steffen Schuldenzuel, Dmitry Moor, Timo Mennle, Ludwig Dierks, Gianluca Brero, and Mike Shann. Each participant's response is indicated by a colored circle: green for 'good', yellow for 'acceptable', and red for 'unacceptable'. A summary table at the bottom shows the number of 'Ja' and 'Nein' responses for each date, along with the total number of participants who responded 'Ja' or 'Nein'.

		November 2015						
		Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer		18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
Sven Seuken		🟢	🟢	🟢	🟡	🟢	🟢	🟢
Steffen Schuldenzuel		🟢	🟢	🟡	🟢	🟢	🟡	🟡
Dmitry Moor		🟢	🟡	🟡	🟢	🟢	🟢	🟢
Timo Mennle		🔴	🟢	🟢	🟢	🟡	🟢	🟡
Ludwig Dierks		🟡	🟢	🟢	🟡	🟡	🟡	🟡
Gianluca Brero		🟡	🟢	🟢	🟢	🟡	🟢	🟢
Mike Shann		🟢	🟢	🟡	🟢	🟡	🟡	🟢
Ihr Name		Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein
Ja	3	6	4	5	3	4	3	
Wennsseinmuss	3	1	3	2	4	3	4	
Nein	1	0	0	0	0	0	0	

Buttons: Ich kann nicht, Speichern

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

Doodle Funktionen Preise Konto erstellen Einloggen

Gemeinsam einen Termin finden
Geben Sie Ihren Namen im Eingabefeld unten ein, und wählen Sie die Termine, an denen Sie Zeit haben.

CERG-Outing

Umfrage von Sven Seuken | 7 | 1 | vor weniger als einer Minute

Dinner (<http://www.oepfelchammer.ch/>) + Movie (James Bond: Spectre)

Tabellen-Ansicht Kalender-Ansicht

	November 2015						
	Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
Sven Seuken	👍	👍	👍	👎	👍	👍	👍
Steffen Schuldenzuel	👍	👍	👎	👍	👍	👎	👎
Dmitry Moor	👍	👎	👎	👍	👍	👍	👍
Timo Mennle	👎	👍	👍	👍	👎	👍	👎
Ludwig Dierks	👎	👍	👍	👎	👎	👎	👎
Gianluca Brero	👎	👍	👍	👍	👎	👍	👍
Mike Shann	👍	👍	👎	👍	👎	👍	👍
Ihr Name	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein
Ja	3	6	4	5	3	4	3
Wennsseinmuss	3	1	3	2	4	3	4
Nein	1	0	0	0	0	0	0

Ich kann nicht Speichern

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

The screenshot shows the Doodle interface for a "CERG-Outing" event. The event is organized by Sven Seuken, with 7 participants and 8 alternatives. The interface is in German and shows a calendar view for November 2015. The alternatives are: Dinner (http://www.oepfelchammer.ch) + Movie (James Bond: Spectre). The calendar view shows the following data:

	Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
Sven Seuken	✓	✓	✓	✓	✓	✓	✓
Steffen Schuldenzuel	✓	✓	✗	✓	✓	✗	✗
Dmitry Moor	✓	✗	✗	✓	✓	✓	✓
Timo Mennle	✗	✓	✓	✓	✗	✓	✗
Ludwig Dierks	✗	✓	✓	✗	✗	✗	✗
Gianluca Brero	✗	✓	✓	✓	✗	✓	✓
Mike Shann	✓	✓	✗	✓	✗	✗	✓
Ihr Name	Ja (Ja) / Nein	Ja (Ja) / Nein	Ja (Ja) / Nein	Ja (Ja) / Nein	Ja (Ja) / Nein	Ja (Ja) / Nein	Ja (Ja) / Nein
Ja	3	6	4	5	3	4	3
Wahrscheinlichkeit	3	0	3	2	4	3	4
Nein	1	1	0	0	0	0	0

Buttons: Ich kann nicht, Speichern

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

Doodle Funktionen Preise Konto erstellen Einloggen

Gemeinsam einen Termin finden
Geben Sie Ihren Namen im Eingabefeld unten ein, und wählen Sie die Termine, an denen Sie Zeit haben.

CERG-Outing

Umfrage von Sven Seuken | 7 | 1 | vor weniger als einer Minute

Dinner (<http://www.oepfelchammer.ch/>) + Movie (James Bond: Spectre)

Tabellen-Ansicht Kalender-Ansicht

	November 2015						
	Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
Sven Seuken	✓	✓	✓	✓	✓	✓	✓
Steffen Schuldenzuel	✓	✓	✗	✓	✓	✗	✗
Dmitry Moor	✓	✗	✗	✓	✓	✓	✓
Timo Mennle	✗	✓	✓	✓	✗	✓	✗
Ludwig Dierks	✗	✓	✓	✗	✗	✗	✗
Gianluca Brero	✓	✓	✓	✓	✓	✓	✓
Mike Shann	✓	✓	✗	✓	✗	✗	✓
Ihr Name	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein
Ja	3	6	4	5	3	4	3
Wahrscheinlich	3	0	3	2	4	3	4
Nein	1	1	0	0	0	0	0

Ich kann nicht Speichern

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

The screenshot shows the Doodle interface for a 'CERG-Outing' event. The event is scheduled for November 2015, with dates Do 5, Mo 9, Di 10, Mi 11, Do 12, Mi 18, and Do 19. The time slot is 18:00 – 23:00. There are 7 participants: Sven Seuken, Steffen Schuldenzuel, Dmitry Moor, Timo Mennle, Ludwig Dierks, Gianluca Brero, and Mike Shann. The interface shows a grid of responses for each participant across the dates. The bottom of the grid shows the number of 'Ja' and 'Nein' responses for each date.

		November 2015						
		Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer		18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
👤	Sven Seuken	👍	👍	👍	👍	👍	👍	👍
👤	Steffen Schuldenzuel	👍	👍	👎	👍	👍	👎	👎
👤	Dmitry Moor	👍	👎	👎	👍	👍	👍	👍
👤	Timo Mennle	👎	👍	👍	👍	👎	👍	👎
👤	Ludwig Dierks	👎	👍	👍	👎	👎	👎	👎
👤	Gianluca Brero	👎	👍	👍	👍	👎	👍	👍
👤	Mike Shann	👍	👍	👎	👍	👎	👍	👍
👤	Ihr Name	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein
	Ja	3	6	4	5	3	4	3
	Wahrscheinlich	3	1	3	2	4	3	4
	Nein	1	0	0	0	0	0	0

Buttons: Ich kann nicht, Speichern

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

Doodle Funktionen Preise Konto erstellen Einloggen

Gemeinsam einen Termin finden
Geben Sie Ihren Namen im Eingabefeld unten ein, und wählen Sie die Termine, an denen Sie Zeit haben.

CERG-Outing

Umfrage von Sven Seuken | 7 | 1 | vor weniger als einer Minute

Dinner (<http://www.oepfelchammer.ch/>) + Movie (James Bond: Spectre)

Tabellen-Ansicht Kalender-Ansicht

		November 2015						
		Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer		18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
Sven Seuken		✓	✓	✓	✓	✓	✓	✓
Steffen Schuldenzuel		✓	✓	✓	✓	✓	✓	✓
Dmitry Moor		✓	✓	✓	✓	✓	✓	✓
Timo Mennle		✓	✓	✓	✓	✓	✓	✓
Ludwig Dierks		✓	✓	✓	✓	✓	✓	✓
Gianluca Brero		✓	✓	✓	✓	✓	✓	✓
Mike Shann		✓	✓	✓	✓	✓	✓	✓
Ihr Name		Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein
Ja	3	6	4	5	3	4	3	
Wahrscheinlich	3	1	3	2	4	3	4	
Nein	1	0	0	0	0	0	0	

Ich kann nicht Speichern

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

Doodle Funktionen Preise Konto erstellen Einloggen

Gemeinsam einen Termin finden
Geben Sie Ihren Namen im Eingabefeld unten ein, und wählen Sie die Termine, an denen Sie Zeit haben.

CERG-Outing

Umfrage von Sven Seuken | 7 | 1 | vor weniger als einer Minute

Dinner (<http://www.oepfelchammer.ch/>) + Movie (James Bond: Spectre)

Tabellen-Ansicht Kalender-Ansicht

	November 2015						
	Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
Sven Seuken	✓	✓	✓	✓	✓	✓	✓
Steffen Schuldenzuel	✓	✓	✓	✓	✓	✓	✓
Dmitry Moor	✓	✓	✓	✓	✓	✓	✓
Timo Mennle	✗	✓	✓	✓	✓	✓	✓
Ludwig Dierks	✓	✓	✓	✓	✓	✓	✓
Gianluca Brero	✓	✓	✓	✓	✓	✓	✓
Mike Shann	✓	✓	✓	✓	✓	✓	✓
Ihr Name	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein
Ja	3	6	4	5	3	4	3
Wahrscheinlich	3	1	3	2	4	3	4
Nein	1	0	0	0	0	0	0

Ich kann nicht Speichern

Motivating Example

- **Problem:** find date for CERG social evening
- **Setting:** 7 agents, 8 alternatives
- **Preferences:** agents classify alternatives into *good*, *acceptable*, or *unacceptable*
- **Desideratum (“Doodle”):**
 1. maximum participation
 2. *good* for many agents

The screenshot shows the Doodle interface for a 'CERG-Outing' event. The event is organized by Sven Seuken and has 7 participants. The calendar view shows 8 alternative dates from November 5th to 19th, 2015, each from 18:00 to 23:00. Each participant has indicated their preference for each date using a color-coded system: green for 'good', yellow for 'acceptable', and red for 'unacceptable'. A summary table at the bottom shows the number of 'Ja' (Yes) and 'Nein' (No) responses for each date, along with the total number of 'Ja' and 'Nein' responses across all dates. The 'Speichern' (Save) button is highlighted in blue.

Doodle Funktionen Preise Konto erstellen Einloggen

Gemeinsam einen Termin finden
Geben Sie Ihren Namen im Eingabefeld unten ein, und wählen Sie die Termine, an denen Sie Zeit haben.

CERG-Outing

Umfrage von Sven Seuken | 7 | 1 | vor weniger als einer Minute

Dinner (<http://www.oepfelchammer.ch/>) + Movie (James Bond: Spectre)

Tabellen-Ansicht Kalender-Ansicht

	Do 5	Mo 9	Di 10	Mi 11	Do 12	Mi 18	Do 19
7 Teilnehmer	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00	18:00 – 23:00
Sven Seuken	✓	✓	✓	✓	✓	✓	✓
Steffen Schuldenzuel	✓	✓	✓	✓	✓	✓	✓
Dmitry Moor	✓	✓	✓	✓	✓	✓	✓
Timo Mennle	✗	✓	✓	✓	✓	✓	✓
Ludwig Dierks	✓	✓	✓	✓	✓	✓	✓
Gianluca Brero	✓	✓	✓	✓	✓	✓	✓
Mike Shann	✓	✓	✓	✓	✓	✓	✓
Ihr Name	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein	Ja (Ja) Nein
Ja	3	6	4	5	3	4	3
Wahrscheinlich	3	1	3	2	4	3	4
Nein	1	0	0	0	0	0	0

Ich kann nicht Speichern

- **Observation:** achieve desideratum \rightarrow not strategyproof

Research Question

- **Observation:** achieve desideratum \rightarrow not strategyproof
- **Strategyproof:** Serial Dictatorship \rightarrow fail desideratum

Research Question

- **Observation:** achieve desideratum → not strategyproof
- **Strategyproof:** Serial Dictatorship → fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)

Research Question

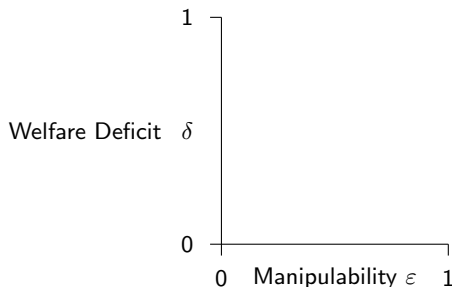
- **Observation:** achieve desideratum → not strategyproof
- **Strategyproof:** Serial Dictatorship → fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)
- **Intermediate:** randomize → intermediate mechanism?

Research Question

- **Observation:** achieve desideratum → not strategyproof
- **Strategyproof:** Serial Dictatorship → fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)
- **Intermediate:** randomize → intermediate mechanism?
- **Question:** *how to make optimal trade-offs when designing ordinal mechanisms?*

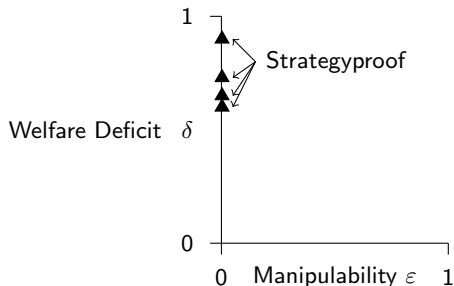
Research Question

- **Observation:** achieve desideratum \rightarrow not strategyproof
- **Strategyproof:** Serial Dictatorship \rightarrow fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)
- **Intermediate:** randomize \rightarrow intermediate mechanism?
- **Question:** *how to make optimal trade-offs when designing ordinal mechanisms?*



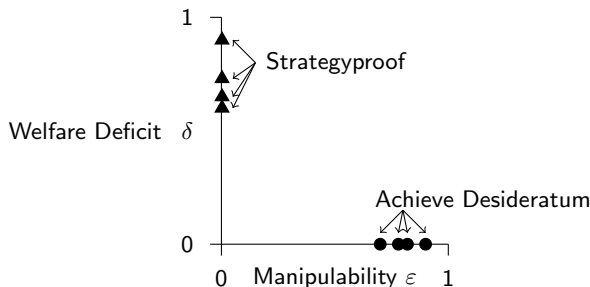
Research Question

- **Observation:** achieve desideratum \rightarrow not strategyproof
- **Strategyproof:** Serial Dictatorship \rightarrow fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)
- **Intermediate:** randomize \rightarrow intermediate mechanism?
- **Question:** *how to make optimal trade-offs when designing ordinal mechanisms?*



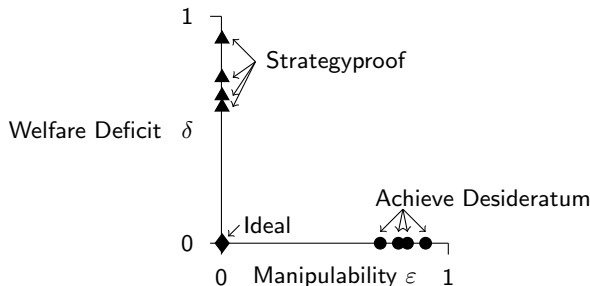
Research Question

- **Observation:** achieve desideratum \rightarrow not strategyproof
- **Strategyproof:** Serial Dictatorship \rightarrow fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)
- **Intermediate:** randomize \rightarrow intermediate mechanism?
- **Question:** *how to make optimal trade-offs when designing ordinal mechanisms?*



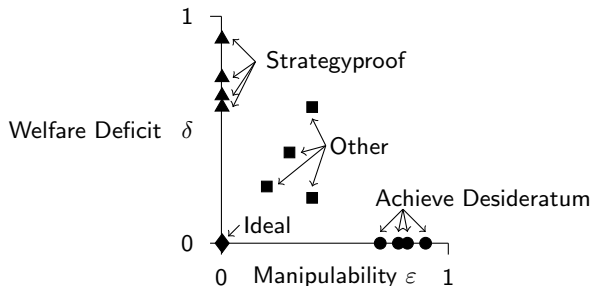
Research Question

- **Observation:** achieve desideratum \rightarrow not strategyproof
- **Strategyproof:** Serial Dictatorship \rightarrow fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)
- **Intermediate:** randomize \rightarrow intermediate mechanism?
- **Question:** *how to make optimal trade-offs when designing ordinal mechanisms?*



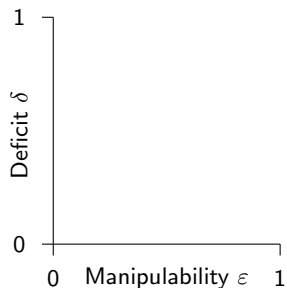
Research Question

- **Observation:** achieve desideratum \rightarrow not strategyproof
- **Strategyproof:** Serial Dictatorship \rightarrow fail desideratum
- **Problem:** strategyproofness in conflict with other desiderata (Gibbard, 1973, 1977; Satterthwaite, 1975)
- **Intermediate:** randomize \rightarrow intermediate mechanism?
- **Question:** *how to make optimal trade-offs when designing ordinal mechanisms?*



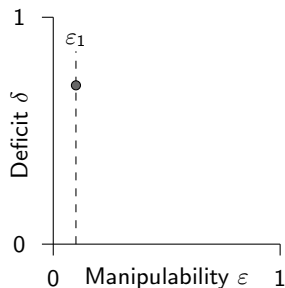
Main Result

- Formalize (*welfare*) *deficit* and *manipulability*



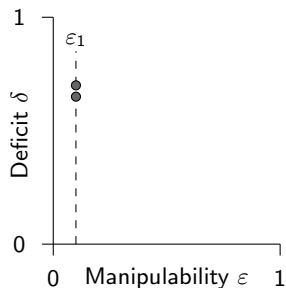
Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.



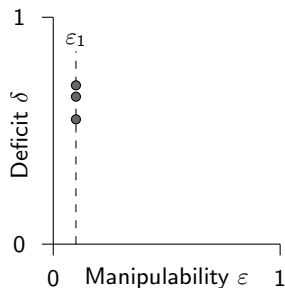
Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.



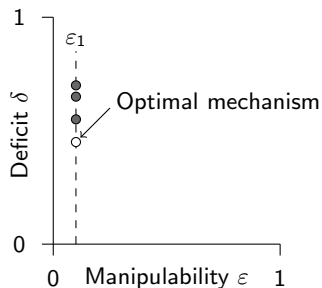
Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.



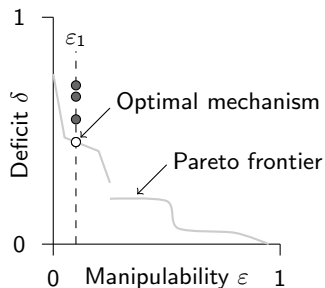
Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.



Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.
- *Pareto frontier*: set of optimal mechanisms



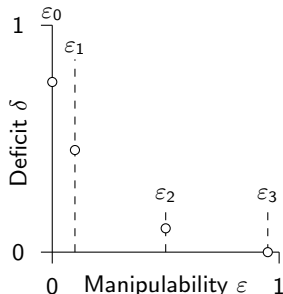
Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.
- *Pareto frontier*: set of optimal mechanisms

Structural characterization:

#1 Solve LP at finite number of *supporting manipulability bounds*

$$\varepsilon_0 < \dots < \varepsilon_K$$

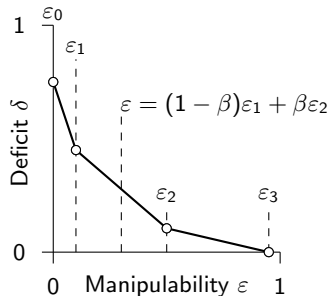


Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.
- *Pareto frontier*: set of optimal mechanisms

Structural characterization:

- #1 Solve LP at finite number of *supporting manipulability bounds* $\varepsilon_0 < \dots < \varepsilon_K$
- #2 Construct *hybrids* at intermediate ε

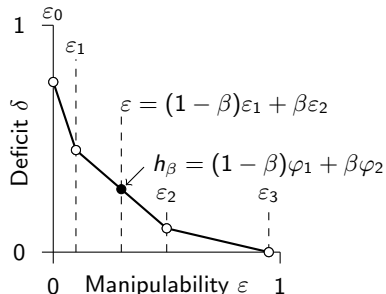


Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.
- *Pareto frontier*: set of optimal mechanisms

Structural characterization:

- #1 Solve LP at finite number of *supporting manipulability bounds* $\varepsilon_0 < \dots < \varepsilon_K$
- #2 Construct *hybrids* at intermediate ε



Main Result

- Formalize (*welfare*) *deficit* and *manipulability*
- *Optimal*: lowest deficit s.t. manipulability $\leq \varepsilon_1$.
- *Pareto frontier*: set of optimal mechanisms

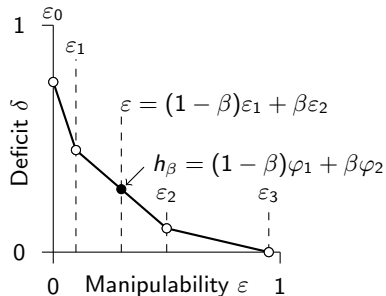
Structural characterization:

#1 Solve LP at finite number of *supporting manipulability bounds*

$$\varepsilon_0 < \dots < \varepsilon_K$$

#2 Construct *hybrids* at intermediate ε

→ Pareto frontier computable



Agenda

- 1 Preliminaries
- 2 #1 Optimal Mechanisms
- 3 #2 Hybrid Mechanisms
- 4 Pareto Frontier
- 5 Generality and Conclusion

Agenda

- 1 Preliminaries
- 2 #1 Optimal Mechanisms
- 3 #2 Hybrid Mechanisms
- 4 Pareto Frontier
- 5 Generality and Conclusion

Domain restrictions:

- Single-peaked preferences (Moulin, 1980)
- Random assignment (Hylland and Zeckhauser, 1979; Abdulkadiroğlu and Sönmez, 1998; Bogomolnaia and Moulin, 2001)

This talk: full & restricted domains

Domain restrictions:

- Single-peaked preferences (Moulin, 1980)
- Random assignment (Hylland and Zeckhauser, 1979; Abdulkadiroğlu and Sönmez, 1998; Bogomolnaia and Moulin, 2001)

This talk: full & restricted domains

Relaxing strategyproofness:

- Strategyproofness in the Large (Azevedo and Budish, 2012)
- Approximate strategyproofness (Carroll, 2013)

This talk: ε -approximate strategyproofness

Domain restrictions:

- Single-peaked preferences (Moulin, 1980)
- Random assignment (Hylland and Zeckhauser, 1979; Abdulkadiroğlu and Sönmez, 1998; Bogomolnaia and Moulin, 2001)

This talk: full & restricted domains

Relaxing strategyproofness:

- Strategyproofness in the Large (Azevedo and Budish, 2012)
- Approximate strategyproofness (Carroll, 2013)

This talk: ϵ -approximate strategyproofness

Trade-offs:

- Approximation of efficient mechanisms in large markets (Procaccia, 2010; Birrell and Pass, 2011)
- SD-efficiency versus (weak) SD-strategyproofness of random mechanisms (Aziz, Brandl and Brandt, 2014)

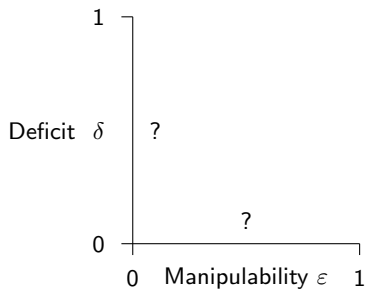
This talk: finite setting & exact trade-offs

- *Agents* $N = \{1, \dots, n\}$, denoted i
- *Alternatives* $M = \{1, \dots, m\}$, denoted a, b, j

- Agents $N = \{1, \dots, n\}$, denoted i
- Alternatives $M = \{1, \dots, m\}$, denoted a, b, j
- Preference order P_i over M :
 - $P_i : a \succeq b \Leftrightarrow i$ weakly prefers a to b
 - $P_i : a \sim b \Leftrightarrow P_i : a \succeq b \ \& \ P_i : b \succeq a$ (indifference)
 - $P_i : a \succ b \Leftrightarrow P_i : a \succeq b \ \& \ P_i : b \not\succeq a$ (strict preference)
 - \mathcal{P} space of preference orders

- Agents $N = \{1, \dots, n\}$, denoted i
- Alternatives $M = \{1, \dots, m\}$, denoted a, b, j
- Preference order P_i over M :
 - $P_i : a \succeq b \Leftrightarrow i$ weakly prefers a to b
 - $P_i : a \sim b \Leftrightarrow P_i : a \succeq b \ \& \ P_i : b \succeq a$ (indifference)
 - $P_i : a \succ b \Leftrightarrow P_i : a \succeq b \ \& \ P_i : b \not\succeq a$ (strict preference)
 - \mathcal{P} space of preference orders
- Preference profile $\mathbf{P} = (P_1, \dots, P_n) = (P_i, P_{-i}) \in \mathcal{P}^n$

- Agents $N = \{1, \dots, n\}$, denoted i
- Alternatives $M = \{1, \dots, m\}$, denoted a, b, j
- Preference order P_i over M :
 - $P_i : a \succeq b \Leftrightarrow i$ weakly prefers a to b
 - $P_i : a \sim b \Leftrightarrow P_i : a \succeq b \ \& \ P_i : b \succeq a$ (indifference)
 - $P_i : a \succ b \Leftrightarrow P_i : a \succeq b \ \& \ P_i : b \not\succeq a$ (strict preference)
 - \mathcal{P} space of preference orders
- Preference profile $\mathbf{P} = (P_1, \dots, P_n) = (P_i, P_{-i}) \in \mathcal{P}^n$
- Random mechanism $\varphi : \mathcal{P}^n \rightarrow \Delta(M)$
 $\varphi(\mathbf{P}) = (x_1, \dots, x_m)$, where $\varphi_j(\mathbf{P}) = x_j$ probability for j



Manipulability

- vNM utilities $u_i : M \rightarrow [0, 1]$ (i.e., bounded),
 $u_i(a) \geq u_i(b) \Leftrightarrow P_i : a \succeq b$, denoted $u_i \sim P_i$

Manipulability

- vNM utilities $u_i : M \rightarrow [0, 1]$ (i.e., bounded),
 $u_i(a) \geq u_i(b) \Leftrightarrow P_i : a \succeq b$, denoted $u_i \sim P_i$
- φ strategyproof if for all profiles $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$,
utilities $u_i \sim P_i$, misreports $P'_i \in \mathcal{P}$:

$$\mathbb{E}_{\varphi(P'_i, P_{-i})} [u_i] - \mathbb{E}_{\varphi(P_i, P_{-i})} [u_i] \leq 0$$

Manipulability

- vNM utilities $u_i : M \rightarrow [0, 1]$ (i.e., bounded),
 $u_i(a) \geq u_i(b) \Leftrightarrow P_i : a \succeq b$, denoted $u_i \sim P_i$
- φ strategyproof if for all profiles $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$,
utilities $u_i \sim P_i$, misreports $P'_i \in \mathcal{P}$:

$$\mathbb{E}_{\varphi(P'_i, P_{-i})} [u_i] - \mathbb{E}_{\varphi(P_i, P_{-i})} [u_i] \leq 0$$

Idea: limit gain from manipulation

Manipulability

- vNM utilities $u_i : M \rightarrow [0, 1]$ (i.e., bounded),
 $u_i(a) \geq u_i(b) \Leftrightarrow P_i : a \succeq b$, denoted $u_i \sim P_i$
- φ *strategyproof* if for all profiles $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$,
utilities $u_i \sim P_i$, misreports $P'_i \in \mathcal{P}$:

$$\mathbb{E}_{\varphi(P'_i, P_{-i})} [u_i] - \mathbb{E}_{\varphi(P_i, P_{-i})} [u_i] \leq 0$$

Idea: limit gain from manipulation

- φ ε -*approximately strategyproof* (ε -ASP) if for all profiles
 $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$, utilities $u_i \sim P_i$, misreports $P'_i \in \mathcal{P}$:

$$\mathbb{E}_{\varphi(P'_i, P_{-i})} [u_i] - \mathbb{E}_{\varphi(P_i, P_{-i})} [u_i] \leq \varepsilon$$

- vNM utilities $u_i : M \rightarrow [0, 1]$ (i.e., bounded),
 $u_i(a) \geq u_i(b) \Leftrightarrow P_i : a \succeq b$, denoted $u_i \sim P_i$
- φ *strategyproof* if for all profiles $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$,
utilities $u_i \sim P_i$, misreports $P'_i \in \mathcal{P}$:

$$\mathbb{E}_{\varphi(P'_i, P_{-i})} [u_i] - \mathbb{E}_{\varphi(P_i, P_{-i})} [u_i] \leq 0$$

Idea: limit gain from manipulation

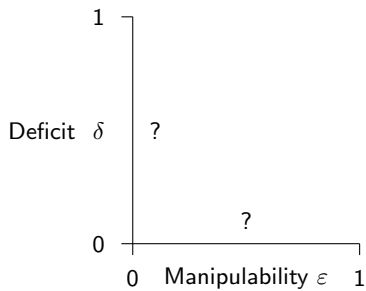
- φ ε -*approximately strategyproof* (ε -ASP) if for all profiles
 $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$, utilities $u_i \sim P_i$, misreports $P'_i \in \mathcal{P}$:

$$\mathbb{E}_{\varphi(P'_i, P_{-i})} [u_i] - \mathbb{E}_{\varphi(P_i, P_{-i})} [u_i] \leq \varepsilon$$

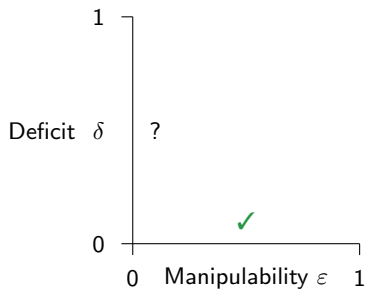
- *Manipulability* of mechanism φ

$$\varepsilon(\varphi) = \min\{\varepsilon' \in [0, 1] : \varphi \text{ is } \varepsilon'\text{-approximately strategyproof}\}$$

Manipulability



Manipulability



Encoding desiderata

- *Welfare function* $w : M \times \mathcal{P}^n \rightarrow [0, 1]$,
 $w(j, \mathbf{P})$ welfare from j at \mathbf{P}

Encoding desiderata

- *Welfare function* $w : M \times \mathcal{P}^n \rightarrow [0, 1]$,
 $w(j, \mathbf{P})$ welfare from j at \mathbf{P} , e.g.,
 - *Positional scoring*: $w(j, \mathbf{P}) \sim$ score of j at \mathbf{P}

Encoding desiderata

- *Welfare function* $w : M \times \mathcal{P}^n \rightarrow [0, 1]$,
 $w(j, \mathbf{P})$ welfare from j at \mathbf{P} , e.g.,
 - *Positional scoring*: $w(j, \mathbf{P}) \sim$ score of j at \mathbf{P}
 - *Pareto optimality*: $w(j, \mathbf{P}) = \mathbb{1}_{\{j \text{ not Pareto dominated at } \mathbf{P}\}}$

Encoding desiderata

- *Welfare function* $w : M \times \mathcal{P}^n \rightarrow [0, 1]$,
 $w(j, \mathbf{P})$ welfare from j at \mathbf{P} , e.g.,
 - *Positional scoring*: $w(j, \mathbf{P}) \sim$ score of j at \mathbf{P}
 - *Pareto optimality*: $w(j, \mathbf{P}) = \mathbb{1}_{\{j \text{ not Pareto dominated at } \mathbf{P}\}}$
... or doodle:

$$w(j, \mathbf{P}) = \begin{cases} \#\{i : j \text{ good for } i\}/n, & \text{if participation maximal at } j, \\ 0, & \text{else} \end{cases}$$

Encoding desiderata

- Welfare function $w : M \times \mathcal{P}^n \rightarrow [0, 1]$,
 $w(j, \mathbf{P})$ welfare from j at \mathbf{P} , e.g.,
 - Positional scoring: $w(j, \mathbf{P}) \sim$ score of j at \mathbf{P}
 - Pareto optimality: $w(j, \mathbf{P}) = \mathbb{1}_{\{j \text{ not Pareto dominated at } \mathbf{P}\}}$
... or doodle:

$$w(j, \mathbf{P}) = \begin{cases} \#\{i : j \text{ good for } i\} / n, & \text{if participation maximal at } j, \\ 0, & \text{else} \end{cases}$$

- Welfare of lottery \mathbf{x} : $w(\mathbf{x}, \mathbf{P}) = \sum_{j \in M} w(j, \mathbf{P}) \cdot x_j$
Interpretation: *expected score, probability of optimality*

Encoding desiderata

- Welfare function $w : M \times \mathcal{P}^n \rightarrow [0, 1]$,
 $w(j, \mathbf{P})$ welfare from j at \mathbf{P} , e.g.,
 - Positional scoring: $w(j, \mathbf{P}) \sim$ score of j at \mathbf{P}
 - Pareto optimality: $w(j, \mathbf{P}) = \mathbb{1}_{\{j \text{ not Pareto dominated at } \mathbf{P}\}}$
... or doodle:

$$w(j, \mathbf{P}) = \begin{cases} \#\{i : j \text{ good for } i\}/n, & \text{if participation maximal at } j, \\ 0, & \text{else} \end{cases}$$

- Welfare of lottery \mathbf{x} : $w(\mathbf{x}, \mathbf{P}) = \sum_{j \in M} w(j, \mathbf{P}) \cdot x_j$
Interpretation: *expected score, probability of optimality*

Measuring performance

- Welfare deficit of lottery \mathbf{x} at \mathbf{P}

$$\delta(\mathbf{x}, \mathbf{P}) = \max_{j \in M} (w(j, \mathbf{P})) - w(\mathbf{x}, \mathbf{P})$$

Encoding desiderata

- *Welfare function* $w : M \times \mathcal{P}^n \rightarrow [0, 1]$,
 $w(j, \mathbf{P})$ welfare from j at \mathbf{P} , e.g.,
 - *Positional scoring*: $w(j, \mathbf{P}) \sim$ score of j at \mathbf{P}
 - *Pareto optimality*: $w(j, \mathbf{P}) = \mathbb{1}_{\{j \text{ not Pareto dominated at } \mathbf{P}\}}$
... or doodle:

$$w(j, \mathbf{P}) = \begin{cases} \#\{i : j \text{ good for } i\}/n, & \text{if participation maximal at } j, \\ 0, & \text{else} \end{cases}$$

- *Welfare of lottery* \mathbf{x} : $w(\mathbf{x}, \mathbf{P}) = \sum_{j \in M} w(j, \mathbf{P}) \cdot x_j$
Interpretation: *expected score, probability of optimality*

Measuring performance

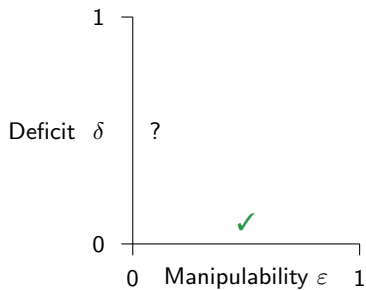
- *Welfare deficit* of lottery \mathbf{x} at \mathbf{P}

$$\delta(\mathbf{x}, \mathbf{P}) = \max_{j \in M} (w(j, \mathbf{P})) - w(\mathbf{x}, \mathbf{P})$$

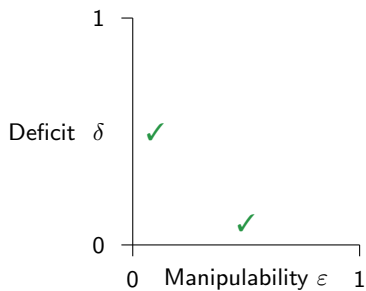
- *Welfare deficit* of mechanism φ

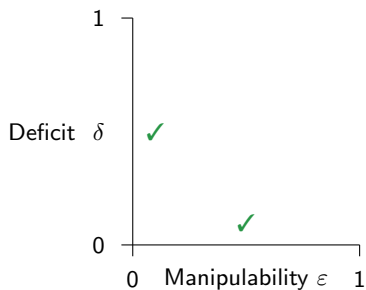
$$\delta(\varphi) = \max_{\mathbf{P} \in \mathcal{P}^n} \delta(\varphi(\mathbf{P}), \mathbf{P})$$

Measures

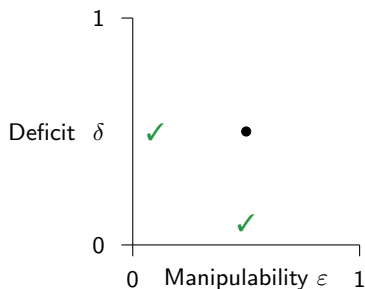


Measures





- Fix *problem* (N, M, δ)

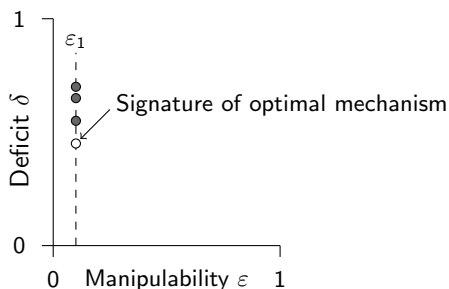


- Fix *problem* (N, M, δ)
- $(\varepsilon(\varphi), \delta(\varphi)) \in [0, 1] \times [0, 1]$ *signature*

Agenda

- 1 Preliminaries
- 2 #1 Optimal Mechanisms**
- 3 #2 Hybrid Mechanisms
- 4 Pareto Frontier
- 5 Generality and Conclusion

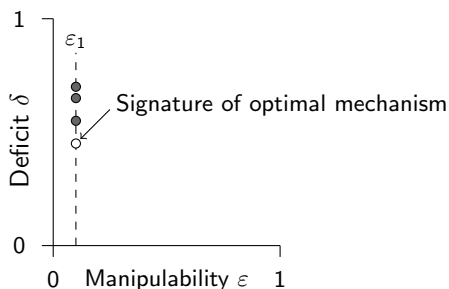
#1 Optimal Mechanisms



Definition (Optimal mechanisms)

$$\text{OPT}(\varepsilon) = \{\varphi : \varphi \text{ is } \varepsilon\text{-ASP}, \delta(\varphi) = \min\{\delta(\varphi') : \varphi' \text{ is } \varepsilon\text{-ASP}\}\}$$

#1 Optimal Mechanisms



Definition (Optimal mechanisms)

$$\text{OPT}(\varepsilon) = \{\varphi : \varphi \text{ is } \varepsilon\text{-ASP}, \delta(\varphi) = \min\{\delta(\varphi') : \varphi' \text{ is } \varepsilon\text{-ASP}\}\}$$

- $\text{OPT}(\varepsilon) \neq \emptyset$

#1 Optimal Mechanisms

OPT(ε) solution to optimization problem:

minimize $\delta(\varphi)$ (minimize deficit)
subject to φ random mechanism
 φ ε -approx. SP

#1 Optimal Mechanisms

OPT(ε) solution to optimization problem:

minimize $\delta(\varphi)$ (minimize deficit)
subject to φ random mechanism
 φ ε -approx. SP \leftarrow Uncountable

#1 Optimal Mechanisms

OPT(ε) solution to optimization problem:

$$\begin{array}{ll} \text{minimize} & \delta(\varphi) & \text{(minimize deficit)} \\ \text{subject to} & \varphi \text{ random mechanism} \\ & \varphi \text{ } \varepsilon\text{-approx. SP} & \leftarrow \text{Uncountable} \end{array}$$

Theorem (Finite Equivalent Set of Linear Constraints)

φ ε -approximately strategyproof if and only if for all profiles $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$, misreports $P'_i \in \mathcal{P}$, indices $k \in \{1, \dots, m\}$:

$$\sum_{j \in M: \text{rank}_{\succeq_i}(j) \leq k} \varphi_j(P'_i, P_{-i}) - \varphi_j(P_i, P_{-i}) \leq \varepsilon$$

#1 Optimal Mechanisms

OPT(ε) solution to optimization problem:

$$\begin{array}{ll} \text{minimize} & \delta(\varphi) & (\text{minimize deficit}) \\ \text{subject to} & \varphi \text{ random mechanism} \\ & \varphi \text{ } \varepsilon\text{-approx. SP} & \leftarrow \begin{array}{l} \text{Uncountable} \\ \text{Finite, linear} \end{array} \end{array}$$

Theorem (Finite Equivalent Set of Linear Constraints)

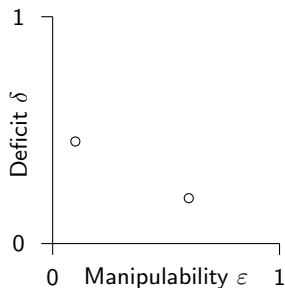
φ ε -approximately strategyproof if and only if for all profiles $\mathbf{P} \in \mathcal{P}^n$, agents $i \in N$, misreports $P'_i \in \mathcal{P}$, indices $k \in \{1, \dots, m\}$:

$$\sum_{j \in M: \text{rank}_{\succeq_i}(j) \leq k} \varphi_j(P'_i, P_{-i}) - \varphi_j(P_i, P_{-i}) \leq \varepsilon$$

Agenda

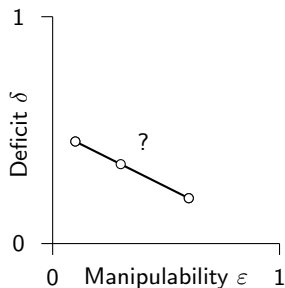
- 1 Preliminaries
- 2 #1 Optimal Mechanisms
- 3 #2 Hybrid Mechanisms**
- 4 Pareto Frontier
- 5 Generality and Conclusion

#2 Hybrid Mechanisms



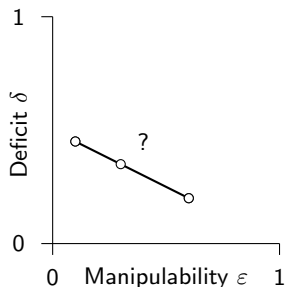
Idea: “mix” mechanisms for intermediate signatures

#2 Hybrid Mechanisms



Idea: “mix” mechanisms for intermediate signatures

#2 Hybrid Mechanisms



Idea: “mix” mechanisms for intermediate signatures

Definition (β -hybrid)

$$h_{\beta} = (1 - \beta)\varphi + \beta\psi, \beta \in [0, 1]$$

#2 Hybrid Mechanisms

Theorem (Guarantees for Hybrids)

For any mechanisms φ, ψ and $\beta \in [0, 1]$,

$$\varepsilon(h_\beta) \leq (1 - \beta)\varepsilon(\varphi) + \beta\varepsilon(\psi),$$

$$\delta(h_\beta) \leq (1 - \beta)\delta(\varphi) + \beta\delta(\psi)$$

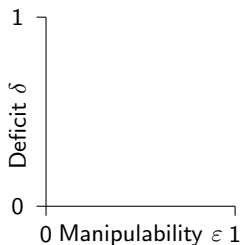
#2 Hybrid Mechanisms

Theorem (Guarantees for Hybrids)

For any mechanisms φ, ψ and $\beta \in [0, 1]$,

$$\varepsilon(h_\beta) \leq (1 - \beta)\varepsilon(\varphi) + \beta\varepsilon(\psi),$$

$$\delta(h_\beta) \leq (1 - \beta)\delta(\varphi) + \beta\delta(\psi)$$



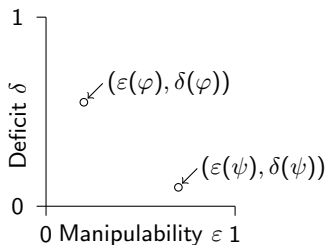
#2 Hybrid Mechanisms

Theorem (Guarantees for Hybrids)

For any mechanisms φ, ψ and $\beta \in [0, 1]$,

$$\varepsilon(h_\beta) \leq (1 - \beta)\varepsilon(\varphi) + \beta\varepsilon(\psi),$$

$$\delta(h_\beta) \leq (1 - \beta)\delta(\varphi) + \beta\delta(\psi)$$



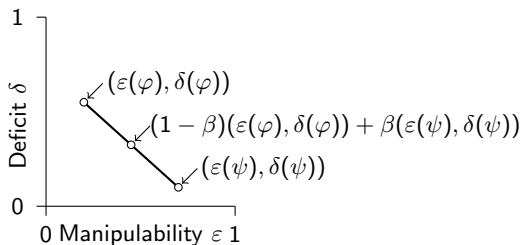
#2 Hybrid Mechanisms

Theorem (Guarantees for Hybrids)

For any mechanisms φ, ψ and $\beta \in [0, 1]$,

$$\varepsilon(h_\beta) \leq (1 - \beta)\varepsilon(\varphi) + \beta\varepsilon(\psi),$$

$$\delta(h_\beta) \leq (1 - \beta)\delta(\varphi) + \beta\delta(\psi)$$



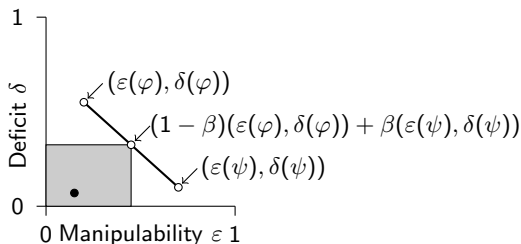
#2 Hybrid Mechanisms

Theorem (Guarantees for Hybrids)

For any mechanisms φ, ψ and $\beta \in [0, 1]$,

$$\varepsilon(h_\beta) \leq (1 - \beta)\varepsilon(\varphi) + \beta\varepsilon(\psi),$$

$$\delta(h_\beta) \leq (1 - \beta)\delta(\varphi) + \beta\delta(\psi)$$



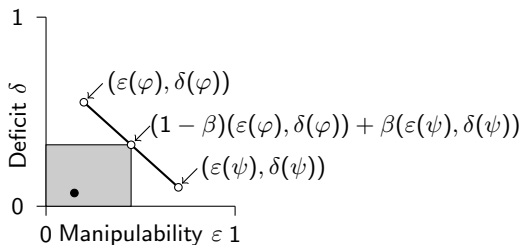
#2 Hybrid Mechanisms

Theorem (Guarantees for Hybrids)

For any mechanisms φ, ψ and $\beta \in [0, 1]$,

$$\varepsilon(h_\beta) \leq (1 - \beta)\varepsilon(\varphi) + \beta\varepsilon(\psi),$$

$$\delta(h_\beta) \leq (1 - \beta)\delta(\varphi) + \beta\delta(\psi)$$



→ Anonymity and neutrality are “free”

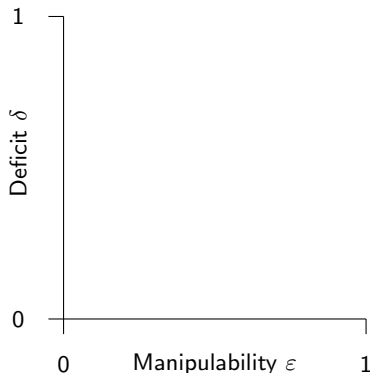
Agenda

- 1 Preliminaries
- 2 #1 Optimal Mechanisms
- 3 #2 Hybrid Mechanisms
- 4 Pareto Frontier**
- 5 Generality and Conclusion

Pareto Frontier - Characterization Result

Result (informal):

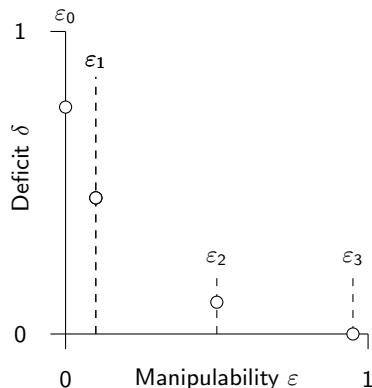
- *Pareto frontier* set of optimal mechanisms



Pareto Frontier - Characterization Result

Result (informal):

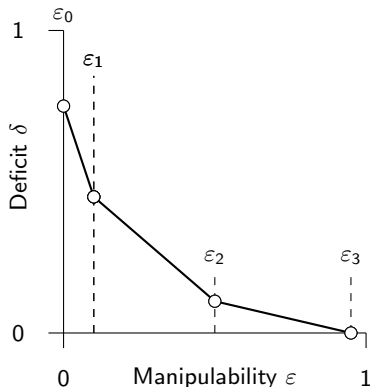
- Pareto frontier set of optimal mechanisms
- Finite set of *supporting manipulability bounds* \rightarrow use LP



Pareto Frontier - Characterization Result

Result (informal):

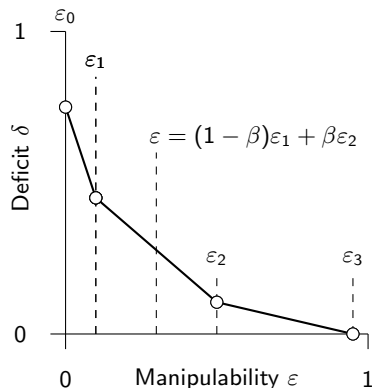
- Pareto frontier set of optimal mechanisms
- Finite set of *supporting manipulability bounds* \rightarrow use LP



Pareto Frontier - Characterization Result

Result (informal):

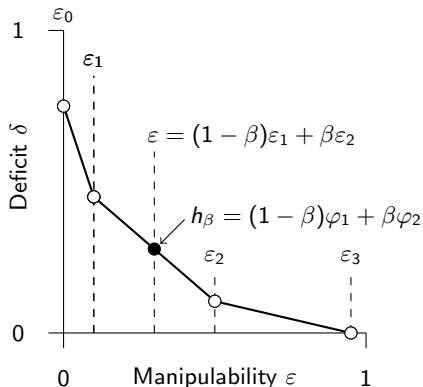
- Pareto frontier set of optimal mechanisms
- Finite set of *supporting manipulability bounds* \rightarrow use LP
- Linear for intermediate manipulability bounds \rightarrow use hybrids



Pareto Frontier - Characterization Result

Result (informal):

- Pareto frontier set of optimal mechanisms
- Finite set of *supporting manipulability bounds* \rightarrow use LP
- Linear for intermediate manipulability bounds \rightarrow use hybrids



Result (formal):

- $\bar{\varepsilon} = \min\{\varepsilon \in [0, 1] \mid \delta(\varepsilon) = 0\}$

Result (formal):

- $\bar{\varepsilon} = \min\{\varepsilon \in [0, 1] \mid \delta(\varepsilon) = 0\}$
- *Pareto frontier* $P_F = \bigcup_{\varepsilon \in [0, \bar{\varepsilon}]} \text{OPT}(\varepsilon)$: no strict reduction of deficit without strict increase of manipulability

Result (formal):

- $\bar{\varepsilon} = \min\{\varepsilon \in [0, 1] \mid \delta(\varepsilon) = 0\}$
- *Pareto frontier* $P_F = \bigcup_{\varepsilon \in [0, \bar{\varepsilon}]} \text{OPT}(\varepsilon)$: no strict reduction of deficit without strict increase of manipulability

Theorem (Characterization of P_F)

Given a problem (N, M, δ) , there exist finitely many supporting manipulability bounds

$$0 = \varepsilon_0 < \dots < \varepsilon_K = \bar{\varepsilon},$$

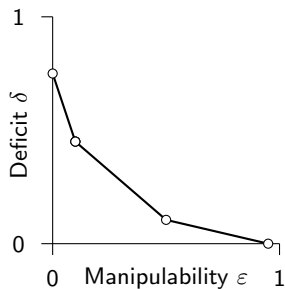
such that for any $[\varepsilon_{k-1}, \varepsilon_k]$ and $\varepsilon = (1 - \beta)\varepsilon_{k-1} + \beta\varepsilon_k$:

$$\begin{aligned}\text{OPT}(\varepsilon) &= (1 - \beta)\text{OPT}(\varepsilon_{k-1}) + \beta\text{OPT}(\varepsilon_k), \\ \delta(\varepsilon) &= (1 - \beta)\delta(\varepsilon_{k-1}) + \beta\delta(\varepsilon_k).\end{aligned}$$

Properties of signature plot $\varepsilon \mapsto \delta(\varepsilon)$

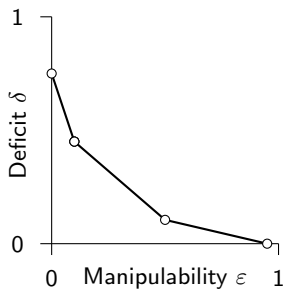
Properties of signature plot $\varepsilon \mapsto \delta(\varepsilon)$

- monotonic & decreasing \rightarrow trade-offs



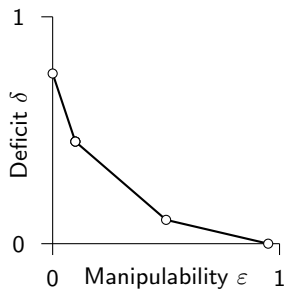
Properties of signature plot $\varepsilon \mapsto \delta(\varepsilon)$

- monotonic & decreasing \rightarrow trade-offs
- convex \rightarrow decreasing marginal reduction of deficit



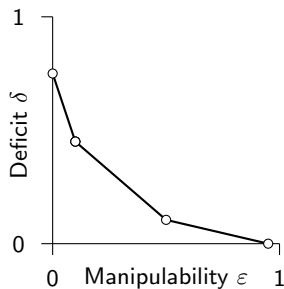
Properties of signature plot $\varepsilon \mapsto \delta(\varepsilon)$

- monotonic & decreasing \rightarrow trade-offs
- convex \rightarrow decreasing marginal reduction of deficit
- continuous \rightarrow smooth change



Properties of signature plot $\varepsilon \mapsto \delta(\varepsilon)$

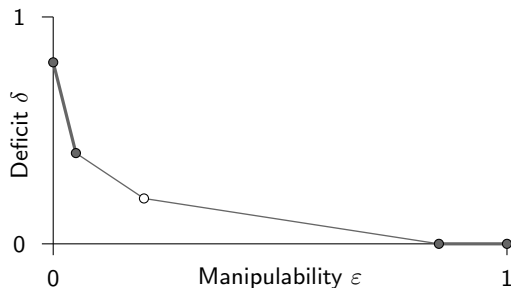
- monotonic & decreasing \rightarrow trade-offs
- convex \rightarrow decreasing marginal reduction of deficit
- continuous \rightarrow smooth change
- piecewise linear \rightarrow PF computable



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

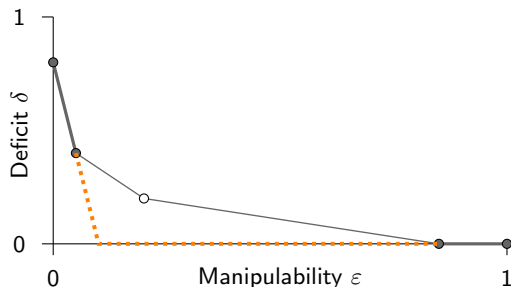
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

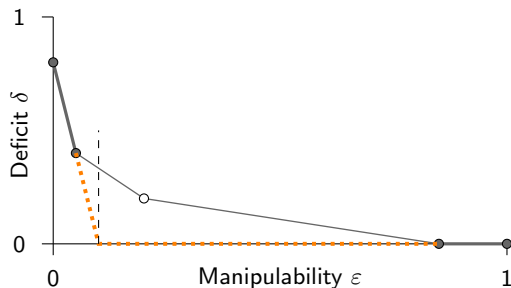
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

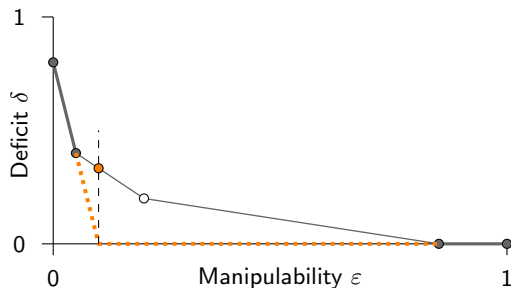
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

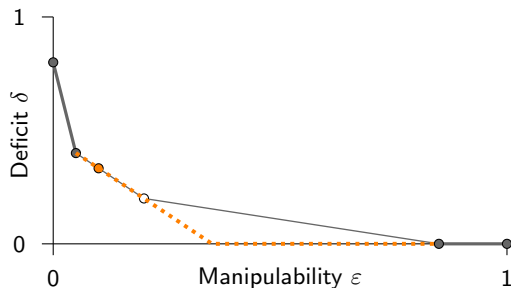
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

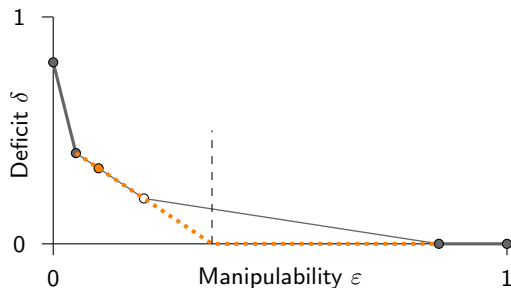
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

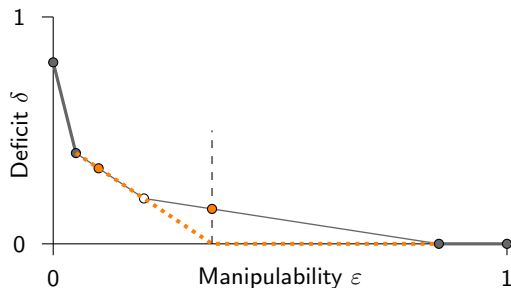
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

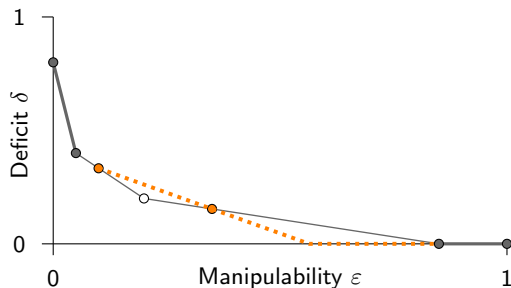
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

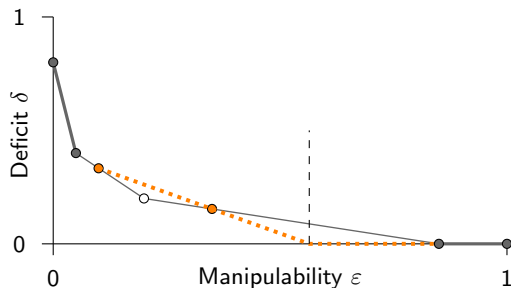
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

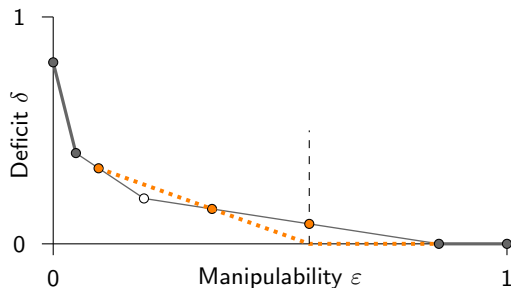
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

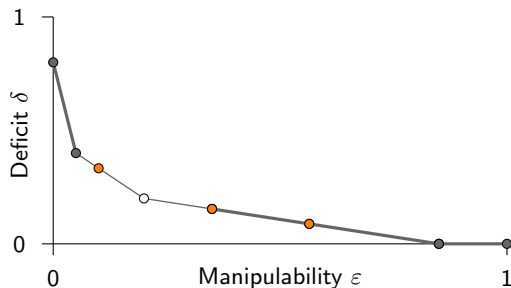
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

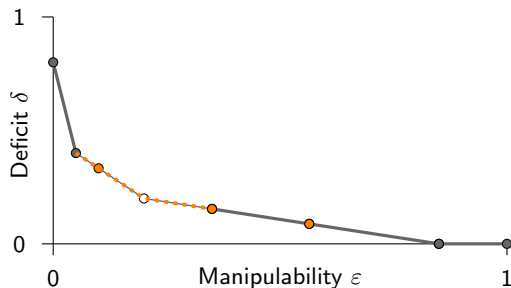
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

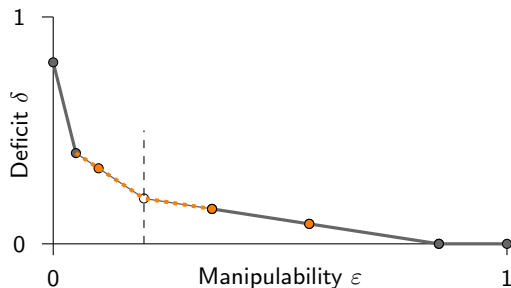
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

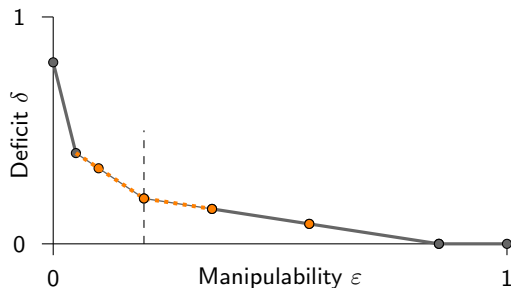
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

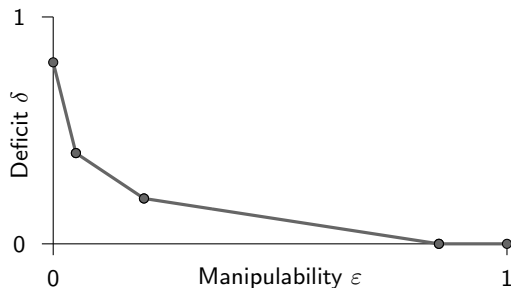
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

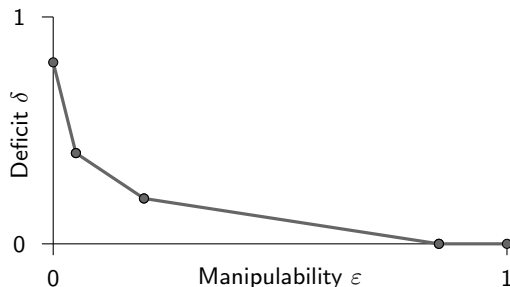
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Pareto Frontier - Computation

Algorithm compute $\text{OPT}(\varepsilon_k)$ for all $k \in \{0, \dots, K\}$:

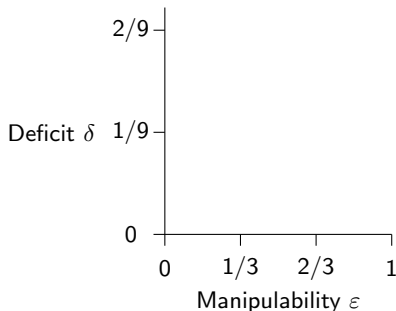
- Interpolate \rightarrow potential supporting manipulability bound ε
- Compute $\delta(\varepsilon) \rightarrow$ verify/discard ε
- Repeat ...



Runtime: at most $4K + \log_2(1/\varepsilon_1) - 1$ executions of LP

#3 Example (1/2)

Problem: 3 agents, 3 alternatives, only strict preferences,
welfare function: positional scoring $v = (1, 0, 0)$ (Plurality)

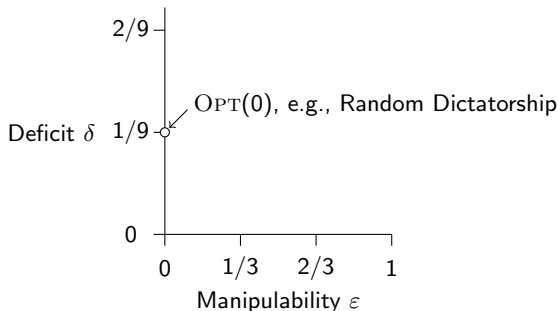


#3 Example (1/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $v = (1, 0, 0)$ (Plurality)

Result:

- *Random Dictatorship* optimal strategyproof mechanism

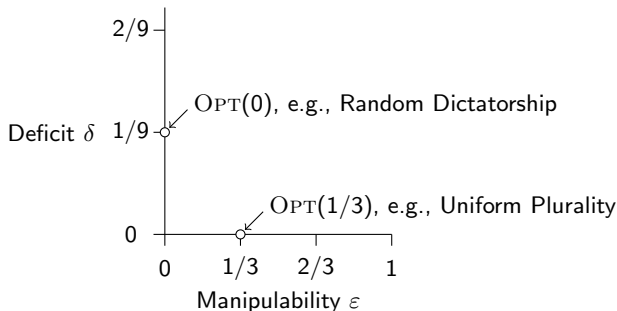


#3 Example (1/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $v = (1, 0, 0)$ (Plurality)

Result:

- *Random Dictatorship* optimal strategyproof mechanism
- *Uniform Plurality* least manipulable efficient mechanism

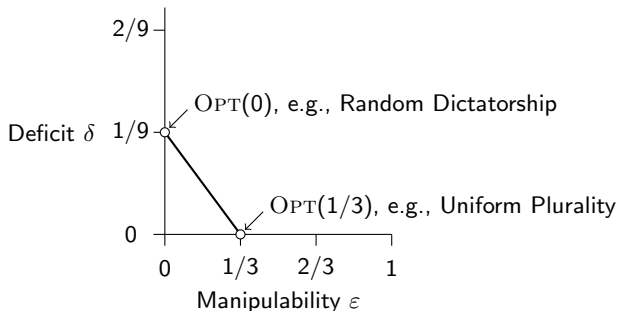


#3 Example (1/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $v = (1, 0, 0)$ (Plurality)

Result:

- *Random Dictatorship* optimal strategyproof mechanism
- *Uniform Plurality* least manipulable efficient mechanism
- No other supporting bounds

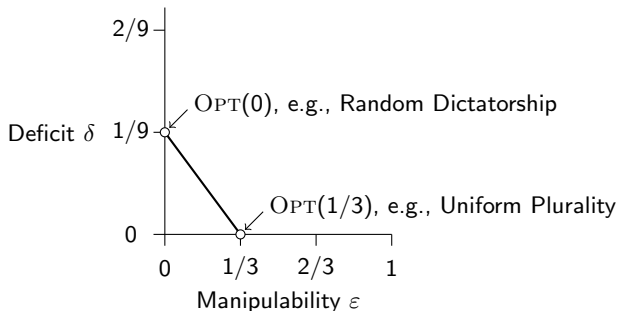


#3 Example (1/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $v = (1, 0, 0)$ (Plurality)

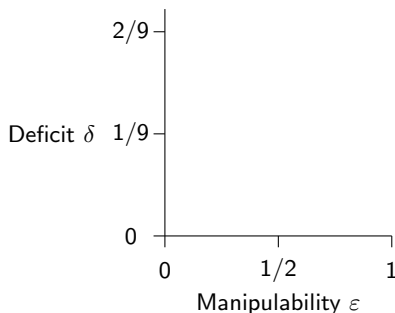
Result:

- *Random Dictatorship* optimal strategyproof mechanism
- *Uniform Plurality* least manipulable efficient mechanism
- No other supporting bounds \rightarrow hybrids optimal



#3 Example (2/2)

Problem: 3 agents, 3 alternatives, only strict preferences,
welfare function: positional scoring $\mathbf{v} = (1, 1, 0)$ (**Veto**)

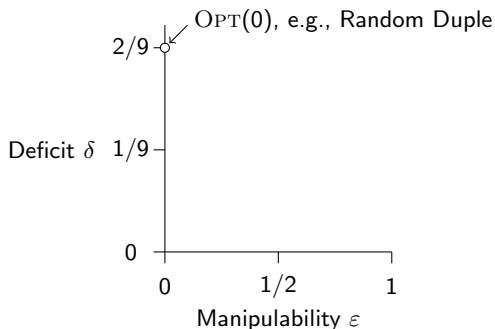


#3 Example (2/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $\mathbf{v} = (1, 1, 0)$ (**Veto**)

Results:

- *Random Duple* optimal strategyproof mechanism

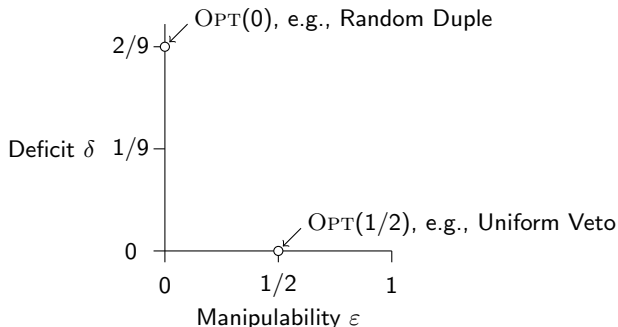


#3 Example (2/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $\mathbf{v} = (1, 1, 0)$ (**Veto**)

Results:

- *Random Duple* optimal strategyproof mechanism
- *Uniform Veto* least manipulable efficient mechanism

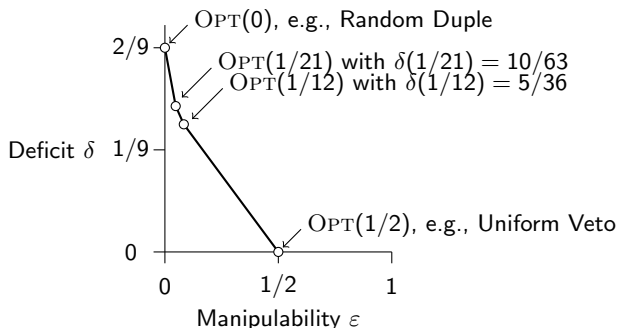


#3 Example (2/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $\mathbf{v} = (1, 1, 0)$ (**Veto**)

Results:

- *Random Duple* optimal strategyproof mechanism
- *Uniform Veto* least manipulable efficient mechanism
- Other supporting bounds $\varepsilon_1 = 1/21, \varepsilon_2 = 1/12$

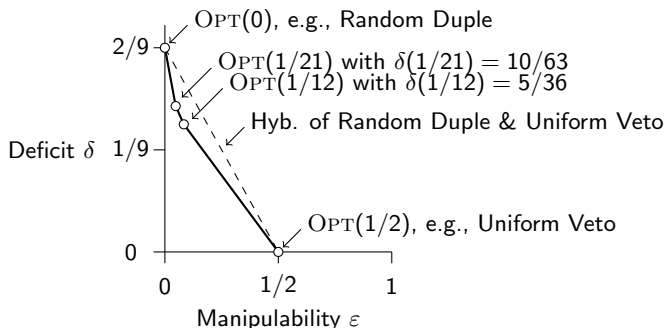


#3 Example (2/2)

Problem: 3 agents, 3 alternatives, only strict preferences, welfare function: positional scoring $\mathbf{v} = (1, 1, 0)$ (**Veto**)

Results:

- *Random Duple* optimal strategyproof mechanism
- *Uniform Veto* least manipulable efficient mechanism
- Other supporting bounds $\varepsilon_1 = 1/21, \varepsilon_2 = 1/12$



Agenda

- 1 Preliminaries
- 2 #1 Optimal Mechanisms
- 3 #2 Hybrid Mechanisms
- 4 Pareto Frontier
- 5 Generality and Conclusion**

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching
- Desiderata and welfare functions:
 - **Assumption:** linearity of welfare for lotteries

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching
- Desiderata and welfare functions:
 - **Assumption:** linearity of welfare for lotteries
 - *Binary* properties, e.g., Pareto optimality, Condorcet consistency, unanimity, v -rank efficiency (assignment), stability (matching)

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching
- Desiderata and welfare functions:
 - **Assumption:** linearity of welfare for lotteries
 - *Binary* properties, e.g., Pareto optimality, Condorcet consistency, unanimity, v -rank efficiency (assignment), stability (matching)
 - *Quantified* properties, e.g., positional scoring, v -rank value (assignment), min. number of blocking pairs (matching)

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching
- Desiderata and welfare functions:
 - **Assumption:** linearity of welfare for lotteries
 - *Binary* properties, e.g., Pareto optimality, Condorcet consistency, unanimity, v -rank efficiency (assignment), stability (matching)
 - *Quantified* properties, e.g., positional scoring, v -rank value (assignment), min. number of blocking pairs (matching)
 - *Target* mechanisms and correspondences

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching
- Desiderata and welfare functions:
 - **Assumption:** linearity of welfare for lotteries
 - *Binary* properties, e.g., Pareto optimality, Condorcet consistency, unanimity, v -rank efficiency (assignment), stability (matching)
 - *Quantified* properties, e.g., positional scoring, v -rank value (assignment), min. number of blocking pairs (matching)
 - *Target* mechanisms and correspondences
 - *Logical combinations:* min \rightarrow “and,” max \rightarrow “or”

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching
- Desiderata and welfare functions:
 - **Assumption:** linearity of welfare for lotteries
 - *Binary* properties, e.g., Pareto optimality, Condorcet consistency, unanimity, v -rank efficiency (assignment), stability (matching)
 - *Quantified* properties, e.g., positional scoring, v -rank value (assignment), min. number of blocking pairs (matching)
 - *Target* mechanisms and correspondences
 - *Logical combinations:* min \rightarrow “and,” max \rightarrow “or”
- Deficit notions:
 - Used *absolute* and *worst-case*

- *Ordinal domain restrictions*
 - **Assumption:** full utility
 - Examples: strict preferences, assignment, matching
- Desiderata and welfare functions:
 - **Assumption:** linearity of welfare for lotteries
 - *Binary* properties, e.g., Pareto optimality, Condorcet consistency, unanimity, v -rank efficiency (assignment), stability (matching)
 - *Quantified* properties, e.g., positional scoring, v -rank value (assignment), min. number of blocking pairs (matching)
 - *Target* mechanisms and correspondences
 - *Logical combinations:* min \rightarrow “and,” max \rightarrow “or”
- Deficit notions:
 - Used *absolute* and *worst-case*
 - *Relative* (i.e., normalized)
 - *Ex-ante* (i.e., expectation wrt. prior over preferences)

Conclusion

Problem: trade off *manipulability* $\varepsilon(\varphi)$ and *deficit* $\delta(\varphi)$

Conclusion

Problem: trade off *manipulability* $\varepsilon(\varphi)$ and *deficit* $\delta(\varphi)$

Results: characterization of Pareto frontier:

there exist finitely many supporting manipulability bounds

$$0 = \varepsilon_0 < \dots < \varepsilon_K = \bar{\varepsilon},$$

Conclusion

Problem: trade off *manipulability* $\varepsilon(\varphi)$ and *deficit* $\delta(\varphi)$

Results: characterization of Pareto frontier:

there exist finitely many supporting manipulability bounds

$$0 = \varepsilon_0 < \dots < \varepsilon_K = \bar{\varepsilon},$$

such that

- At ε_k : optimal mechanisms given by linear program

Conclusion

Problem: trade off *manipulability* $\varepsilon(\varphi)$ and *deficit* $\delta(\varphi)$

Results: characterization of Pareto frontier:

there exist finitely many supporting manipulability bounds

$$0 = \varepsilon_0 < \dots < \varepsilon_K = \bar{\varepsilon},$$

such that

- At ε_k : optimal mechanisms given by linear program
- At $\varepsilon \in (\varepsilon_{k-1}, \varepsilon_k)$: optimal hybrids of φ_{k-1} and φ_k

Conclusion

Problem: trade off *manipulability* $\varepsilon(\varphi)$ and *deficit* $\delta(\varphi)$

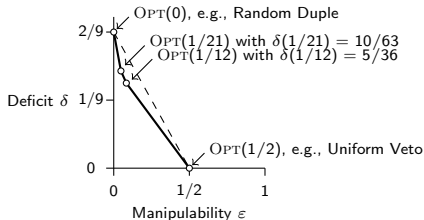
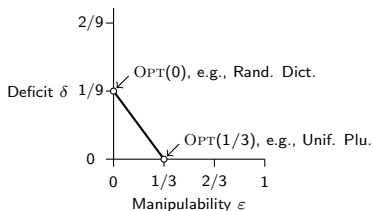
Results: characterization of Pareto frontier:

there exist finitely many supporting manipulability bounds

$$0 = \varepsilon_0 < \dots < \varepsilon_K = \bar{\varepsilon},$$

such that

- At ε_k : optimal mechanisms given by linear program
- At $\varepsilon \in (\varepsilon_{k-1}, \varepsilon_k)$: optimal hybrids of φ_{k-1} and φ_k



Conclusion

Problem: trade off *manipulability* $\varepsilon(\varphi)$ and *deficit* $\delta(\varphi)$

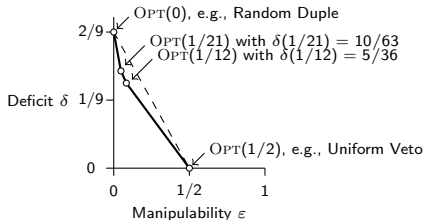
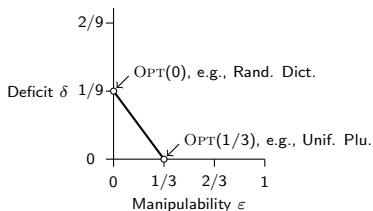
Results: characterization of Pareto frontier:

there exist finitely many supporting manipulability bounds

$$0 = \varepsilon_0 < \dots < \varepsilon_K = \bar{\varepsilon},$$

such that

- At ε_k : optimal mechanisms given by linear program
- At $\varepsilon \in (\varepsilon_{k-1}, \varepsilon_k)$: optimal hybrids of φ_{k-1} and φ_k



Thank you!

- Abdulkadiroğlu, Atila, and Tayfun Sönmez.** 1998. "Random Serial Dictatorship and the Core from Random Endowments in House Allocation Problems." *Econometrica*, 66(3): 689–702.
- Azevedo, Eduardo M., and Eric Budish.** 2012. "Strategyproofness in the Large as a Desideratum for Market Design." Extended Abstract.
- Aziz, Haris, Florian Brandl, and Felix Brandt.** 2014. "On the Incompatibility of Efficiency and Strategyproofness in Randomized Social Choice." 67–72.
- Birrell, Eleanor, and Rafael Pass.** 2011. "Approximately Strategy-Proof Voting." 67–72.
- Bogomolnaia, Anna, and Hervé Moulin.** 2001. "A New Solution to the Random Assignment Problem." *Journal of Economic Theory*, 100(2): 295–328.
- Carroll, Garbiel.** 2013. "A Quantitative Approach to Incentives: Application to Voting Rules." Working Paper.
- Gibbard, Allan.** 1973. "Manipulation of Voting Schemes: a General Result." *Econometrica*, 41(4): 587–601.
- Gibbard, Allan.** 1977. "Manipulation of Schemes That Mix Voting with Chance." *Econometrica*, 45(3): 665–81.
- Hylland, Aanund, and Richard Zeckhauser.** 1979. "The Efficient Allocation of Individuals to Positions." *The Journal of Political Economy*, 87(2): 293–314.
- Moulin, Hervé.** 1980. "On Strategy-Proofness and Single Peakedness." *Public Choice*, 35(4): 437–455.
- Procaccia, Ariel.** 2010. "Can Approximation Circumvent Gibbard-Satterthwaite?" 836–841.
- Satterthwaite, Mark.** 1975. "Strategy-proofness and Arrow's Conditions: Existence and Correspondence Theorems for Voting Procedures and Social Welfare Functions." *Journal of Economic Theory*, 10(2): 187–217.