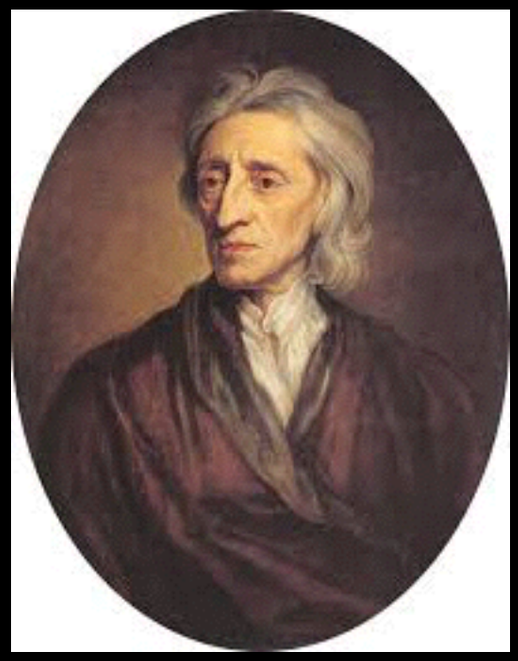
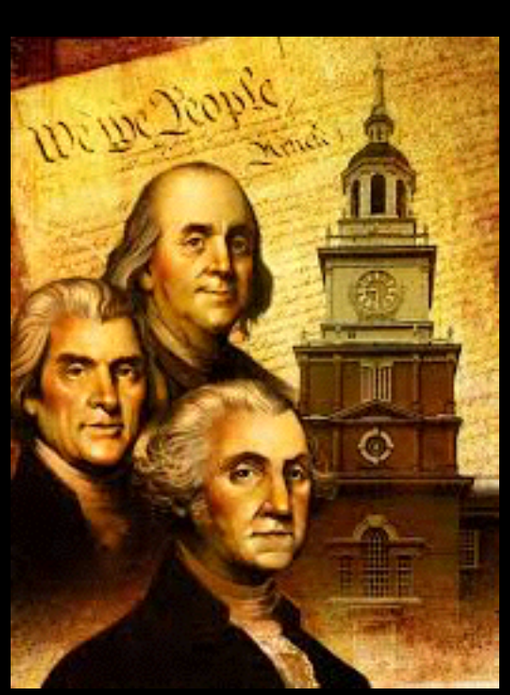
Regulating Highly Automated Robot Ecologies:
Insights from Three User Studies

Wen Shen (UC, Irvine)
Alanoud Al Khemeiri (Masdar Institute)
Abdulla Almehrzi (Masdar Institute)
Wael Al Enezi (Masdar Institute)
Iyad Rahwan (MIT)
Jacob W. Crandall (BYU)

Human Societies

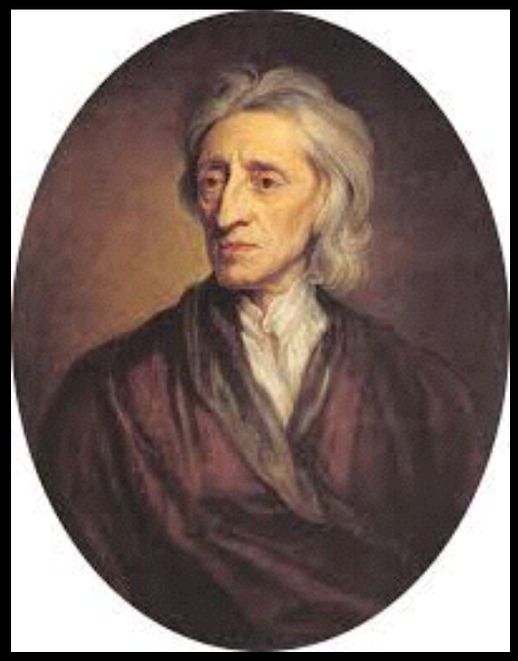
- How do we achieve good human societies?

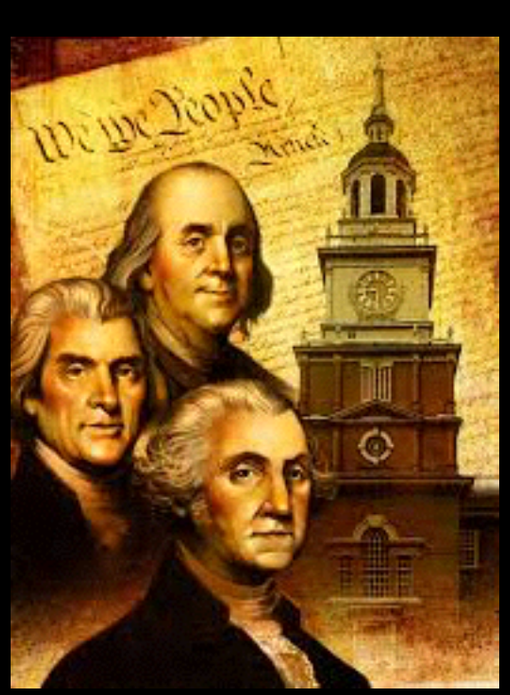




Human Societies

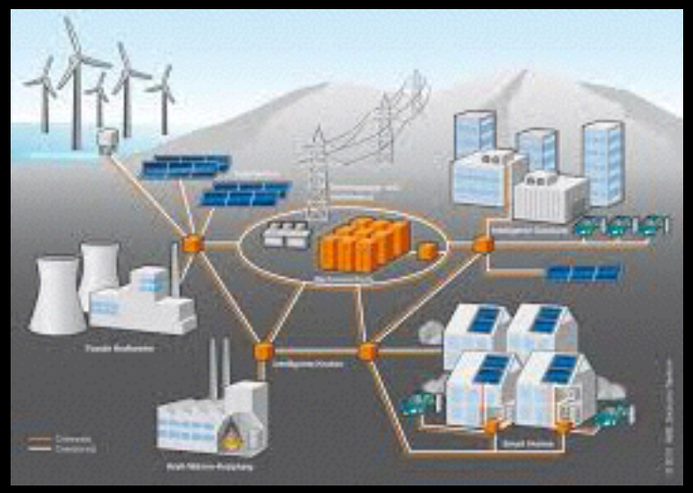
- How do we achieve good human societies?





Strong central authority vs. strong individual rights

Societies of Robots?

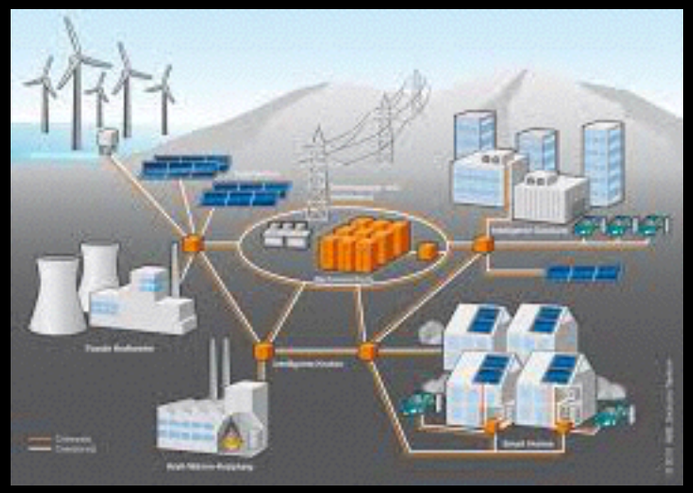


Robotic buildings
connected via a smart grid

Self-driving cars

Financial Markets

Societies of Robots?



Robotic buildings
connected via a smart grid

Self-driving cars

Financial Markets

Strong central authority vs. strong individual rights

Highly **A**utomated **R**obot **E**cologies

- Society of robots or systems
  - Robots are **independent** — owned by different stakeholders
  - Robots are **autonomous** (from the perspective of the regulator)

Highly **A**utomated **R**obot **E**cologies

- Society of robots or systems
  - Robots are **independent** — owned by different stakeholders
- Robots are **autonomous** (from the perspective of the regulator)

How can such systems be “designed” to produce good societal outcomes?

HARE are like what?

HARE are like what?

Supervisory control systems

HARE are like what?

Supervisory control systems

HARE are like what?

~~Supervisory control systems~~

Human Society

HARE are like what?

~~Supervisory control systems~~

~~Human Society~~

HARE are like what?

~~Supervisory control systems~~

~~Human Society~~

Mechanism design problem

HARE are like what?

~~Supervisory control systems~~

~~Human Society~~

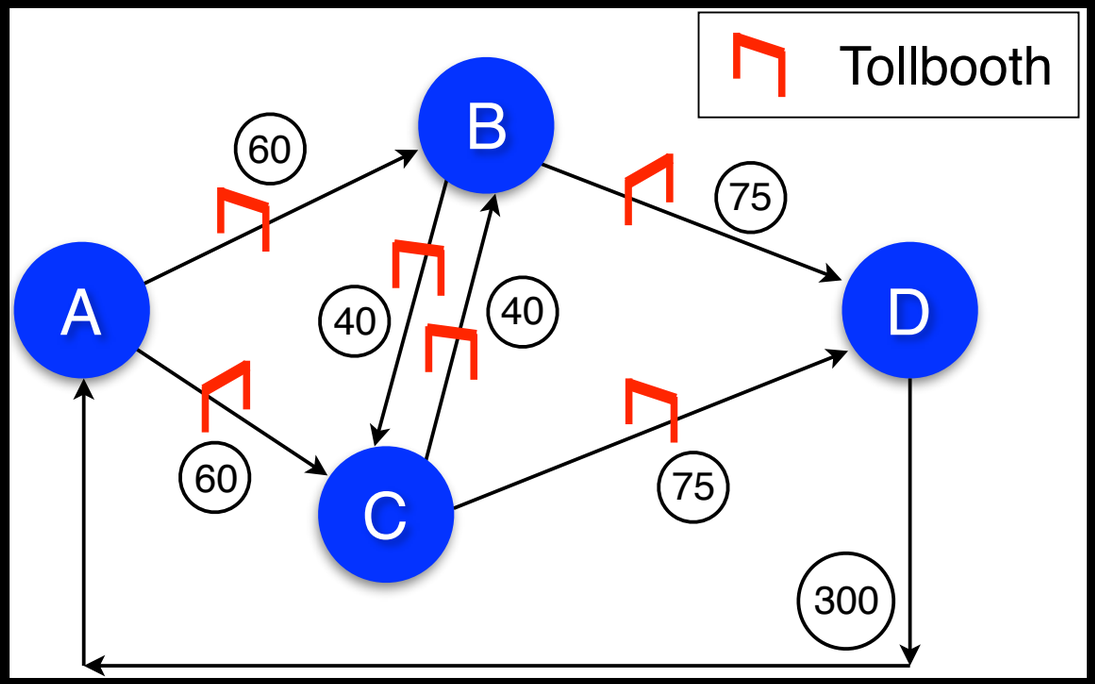
~~Mechanism design problem~~

Challenge: Design efficient HARE

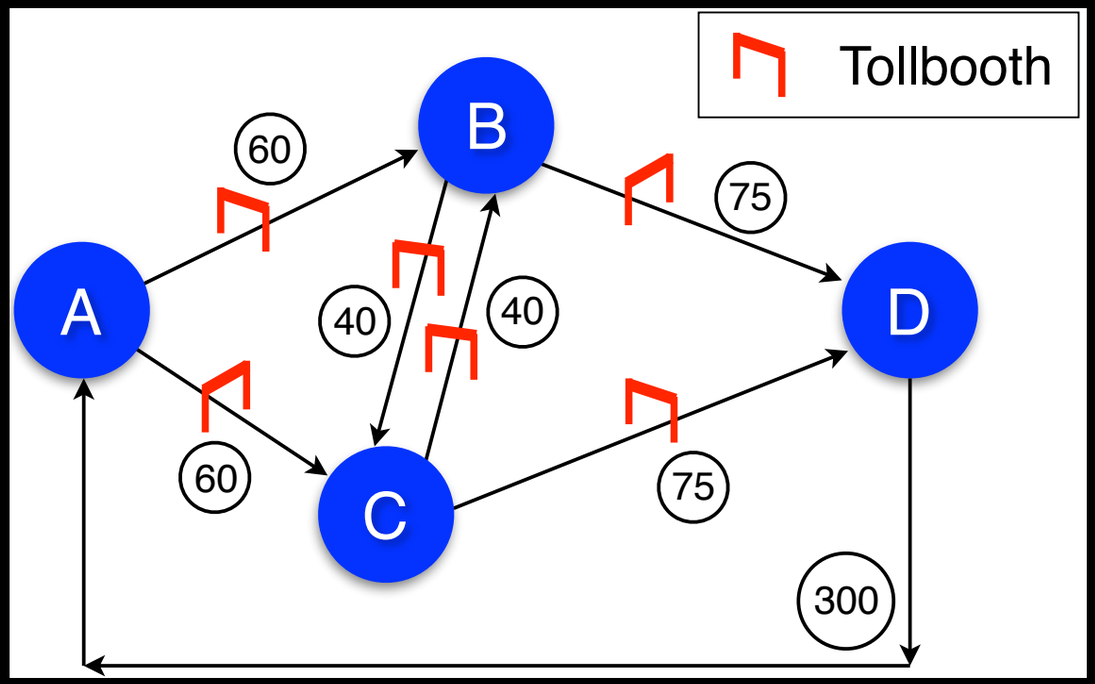
2 “design parameters”

- Regulatory power
- Robot autonomy (adaptability)

Example: Routing Game

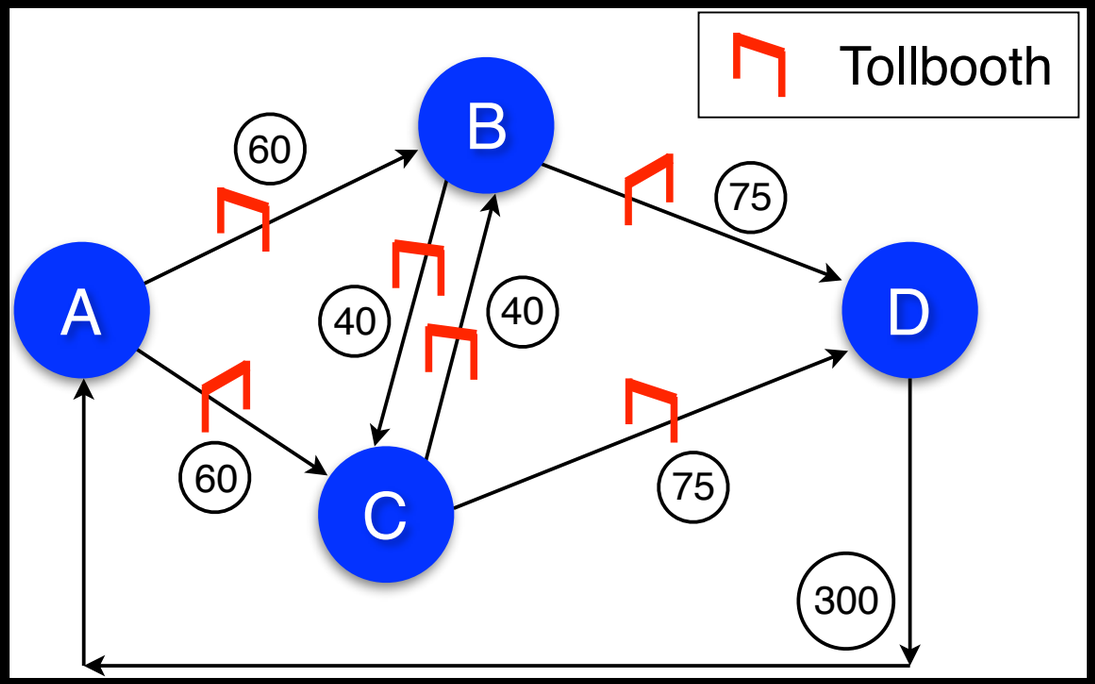


Example: Routing Game



$$V_{ij} = f\,(N_{ij},\ C_{ij})$$

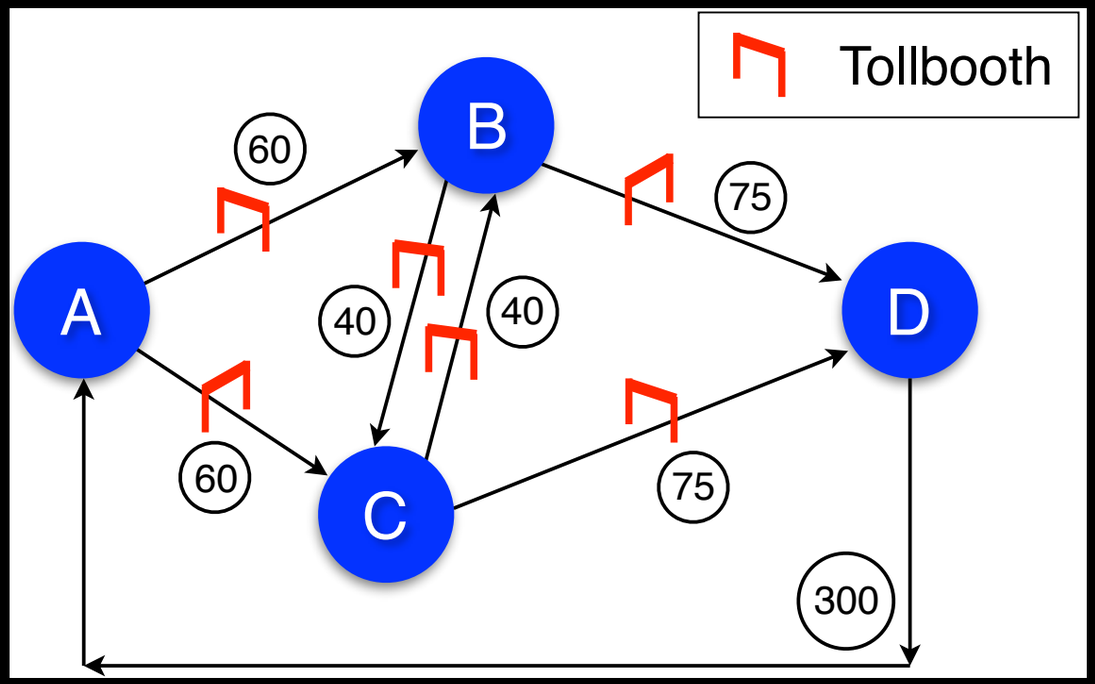
Example: Routing Game



$$V_{ij} = f\,(N_{ij},\ C_{ij})$$

# of vehicles on link i-j

Example: Routing Game

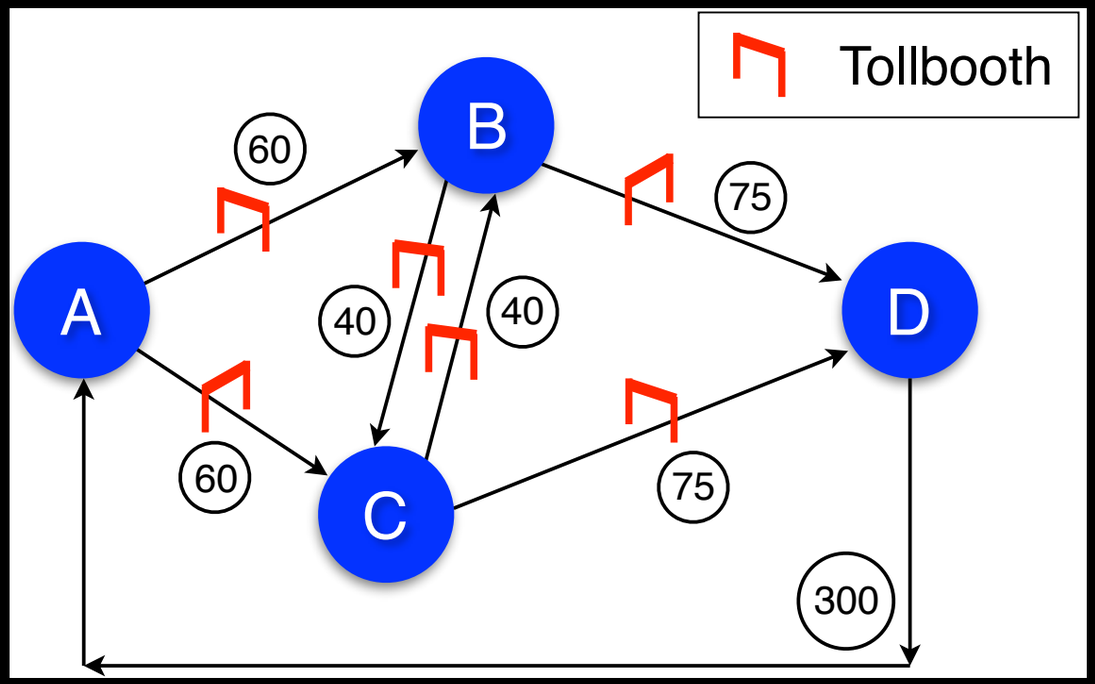


$$V_{ij} = f\,(N_{ij},\ C_{ij})$$

# of vehicles on link i-j

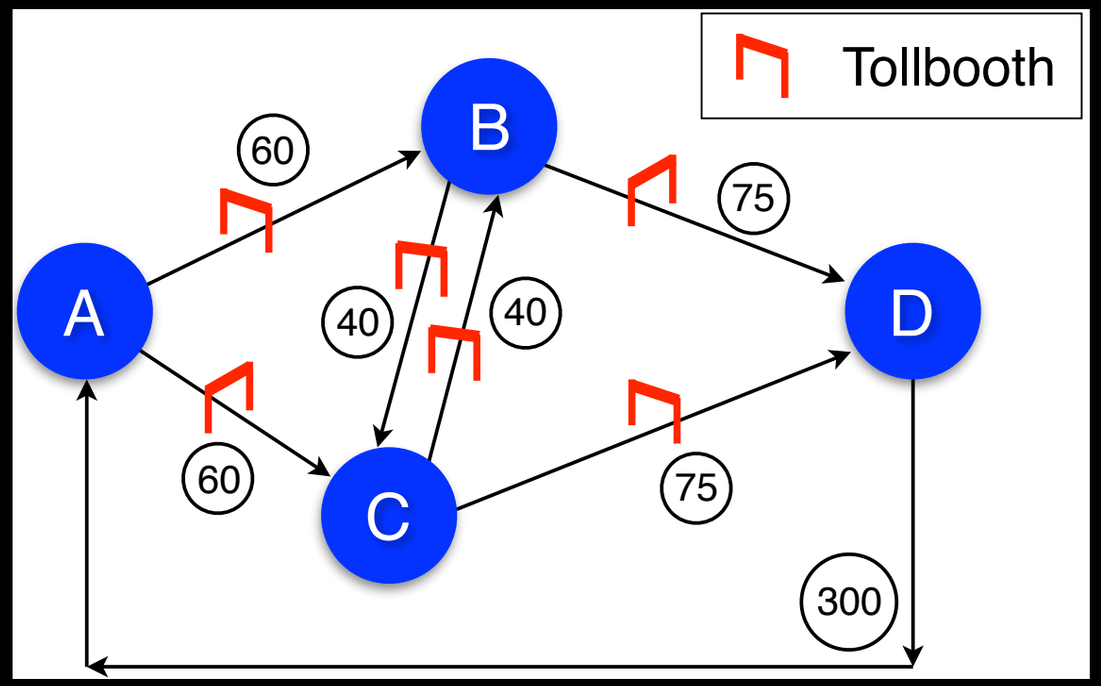
capacity of link i-j

Example: Routing Game



$$V_{ij} \propto \frac{1}{1 + e^{0.25(N_{ij} - C_{ij})}} + 0.1$$

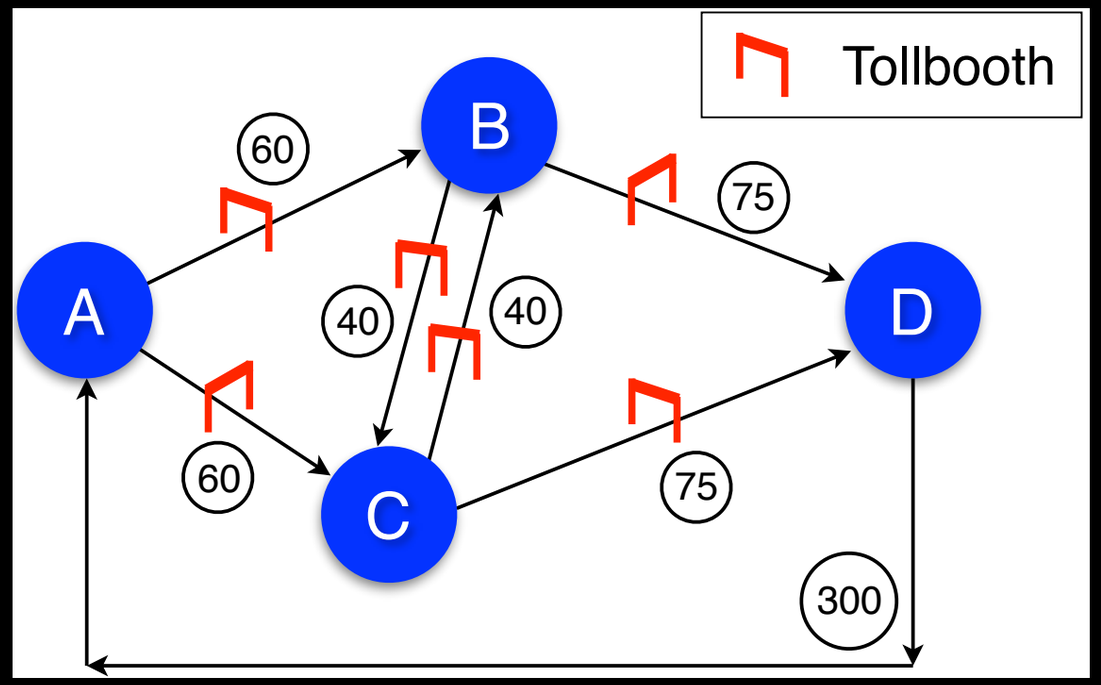
Example: Routing Game



Regulator's Goal:
Maximize throughput
through node D

$$V_{ij} \propto \frac{1}{1 + e^{0.25(N_{ij} - C_{ij})}} + 0.1$$

Example: Routing Game



Regulator's Goal:
Maximize throughput through node D

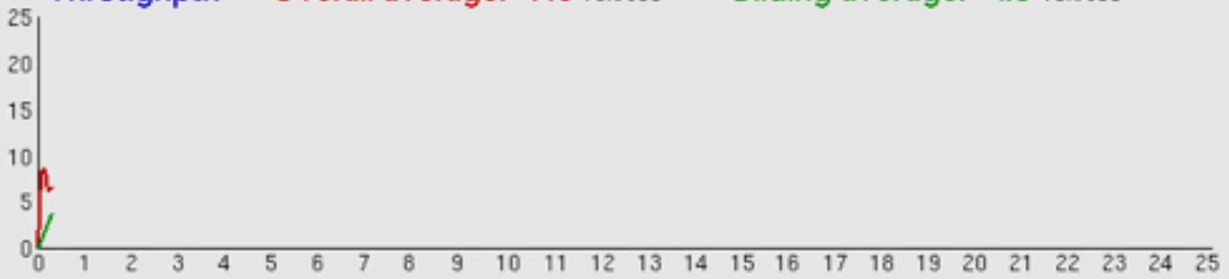
Needs to remove traffic congestion

$$V_{ij} \propto \frac{1}{1 + e^{0.25(N_{ij} - C_{ij})}} + 0.1$$

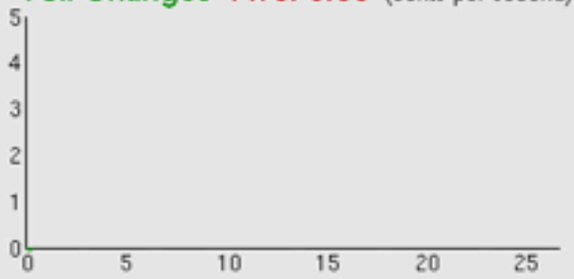
Jiao Tong

00:21

Throughput Overall average: 7.0 veh/sec Sliding average: 4.9 veh/sec



Toll Changes Ave: 0.00 (cents per second)



High Scores

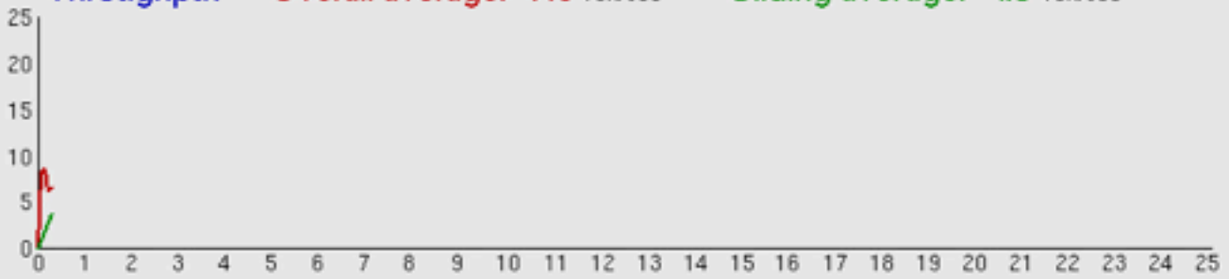
1.	005	14.93
2.	Bill	14.71
3.	07	13.49
4.	08	13.37
5.	017	13.33

Your Score: $ 6.95 / sec

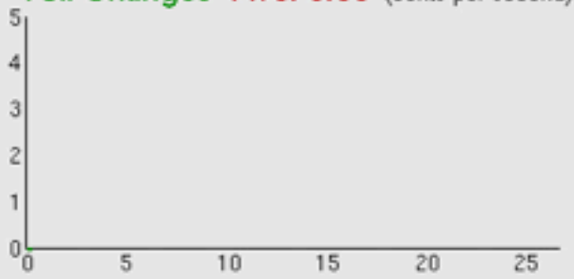
Jiao Tong

00:21

Throughput Overall average: 7.0 veh/sec Sliding average: 4.9 veh/sec



Toll Changes Ave: 0.00 (cents per second)



High Scores

1.	005	14.93
2.	Bill	14.71
3.	07	13.49
4.	08	13.37
5.	017	13.33

Your Score: $ 6.95 / sec

Robot Behaviors

Robot Behaviors

$$u(i,g) = v(g) - c_t(i,g) - c_\$(i,g)$$

Robot Behaviors

$$u(i,g) = v(g) - c_t(i,g) - c_\$(i,g)$$

Value of getting to node *g*

Robot Behaviors

$$u(i,g) = v(g) - c_t(i,g) - c_\$(i,g)$$

Value of getting to node *g*

Travel Cost

Robot Behaviors

$$u(i,g) = v(g) - c_t(i,g) - c_\$(i,g)$$

Value of getting to node *g*

Travel Cost

Toll Cost

Robot Behaviors

$$u(i,g) = v(g) - c_t(i,g) - c_\$(i,g)$$

Value of getting to node *g*

Travel Cost

Toll Cost

Robot Autonomy (2 levels)

Robot Behaviors

$$u(i,g) = v(g) - c_t(i,g) - c_\$(i,g)$$

Value of getting to node *g*

Travel Cost

Toll Cost

Robot Autonomy (2 levels)

- **Simple** — Estimate $c_t(i,g)$ assuming no congestion

Robot Behaviors

$$u(i,g) = v(g) - c_t(i,g) - c_\$(i,g)$$

Value of getting to node *g*

Travel Cost

Toll Cost

Robot Autonomy (2 levels)

- **Simple** — Estimate $c_t(i,g)$ assuming no congestion
- **Adaptive** — Estimate $c_t(i,g)$ using reinforcement learning

Regulatory Power

Regulator's ability to change tolls

Regulatory Power

Regulator's ability to change tolls

3 levels

Regulatory Power

Regulator's ability to change tolls

3 levels

- **None** — Regulator can do nothing

Regulatory Power

Regulator's ability to change tolls

3 levels

- **None** — Regulator can do nothing
- **Limited** — Regulator can make limited toll changes

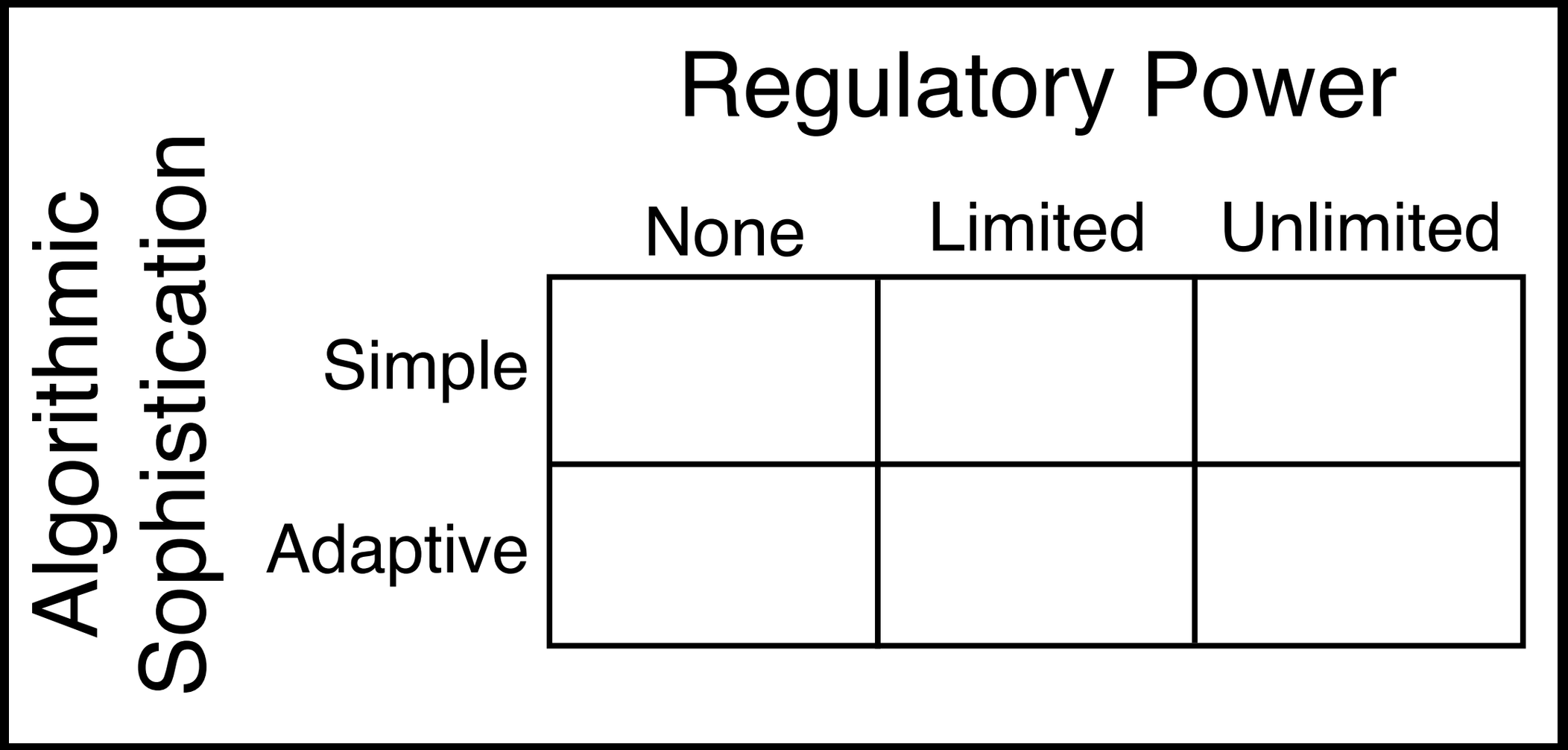
Regulatory Power

Regulator's ability to change tolls

3 levels

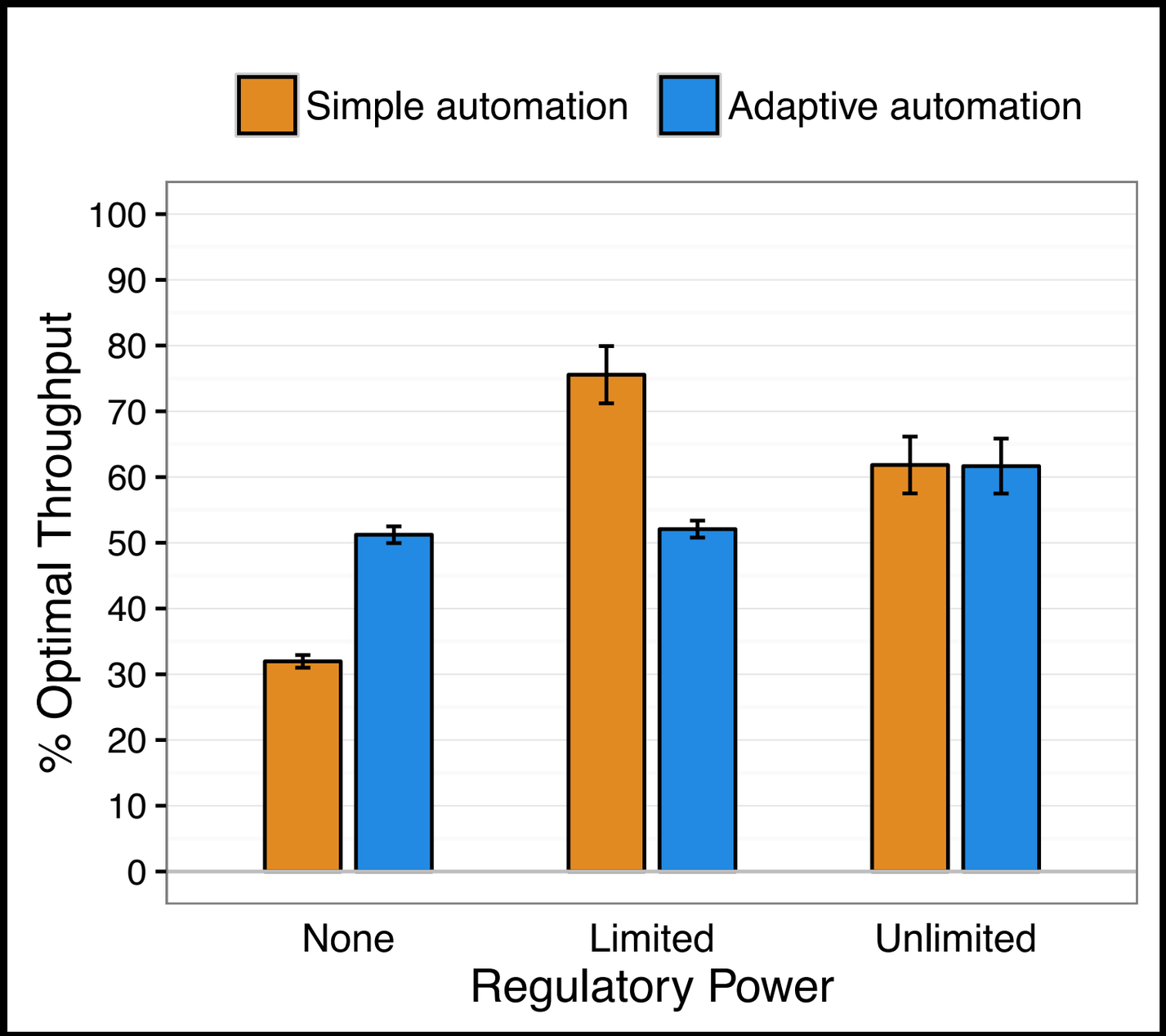
- **None** — Regulator can do nothing
- **Limited** — Regulator can make limited toll changes
- **Unlimited** — Regulator can make unlimited toll changes

Experimental Setup

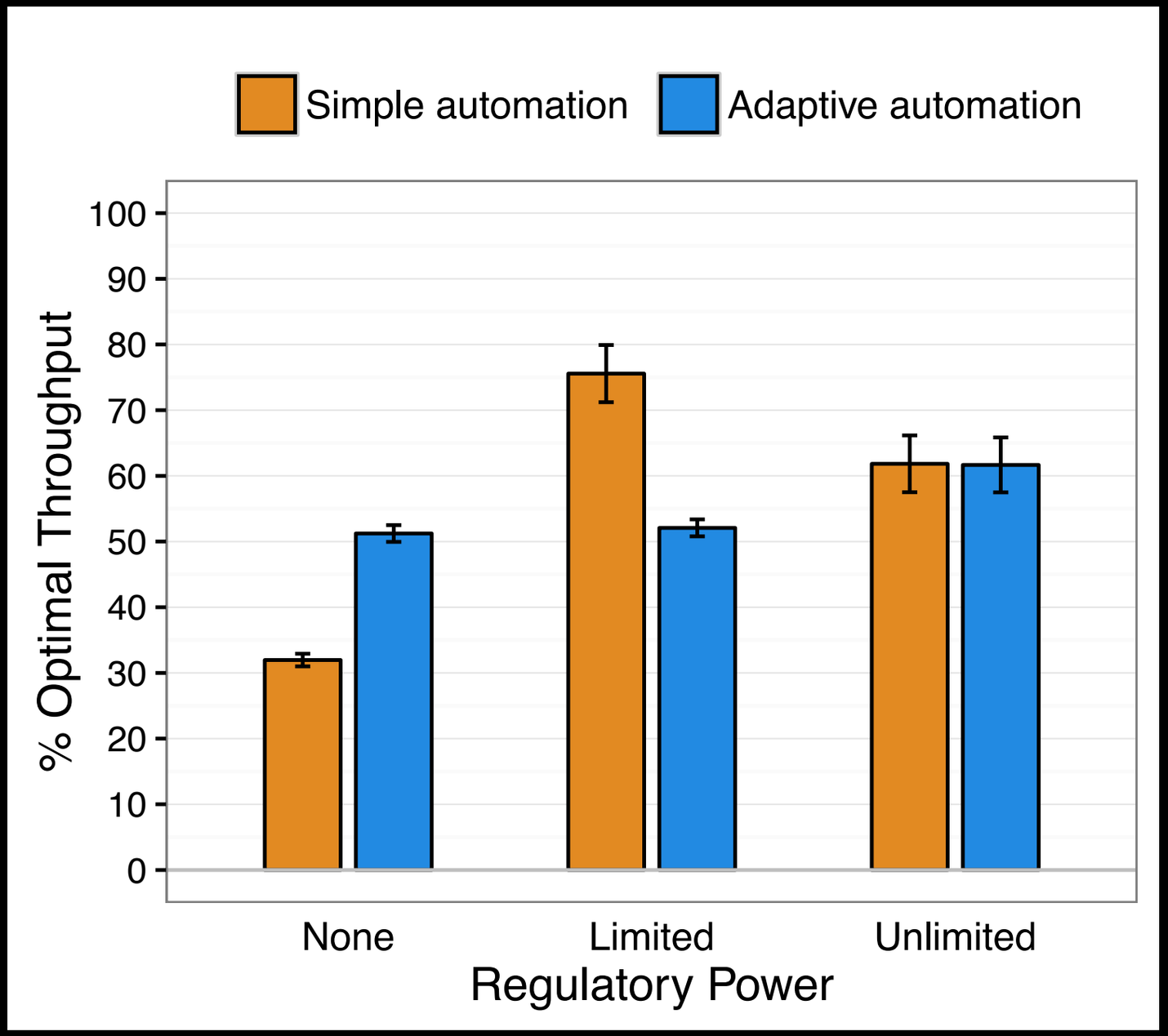


Which one will be best?

Outcome


Simple automation
Adaptive automation
% Optimal Throughput
100
90
80
70
60
50
40
30
20
10
0
None
Limited
Unlimited
Regulatory Power

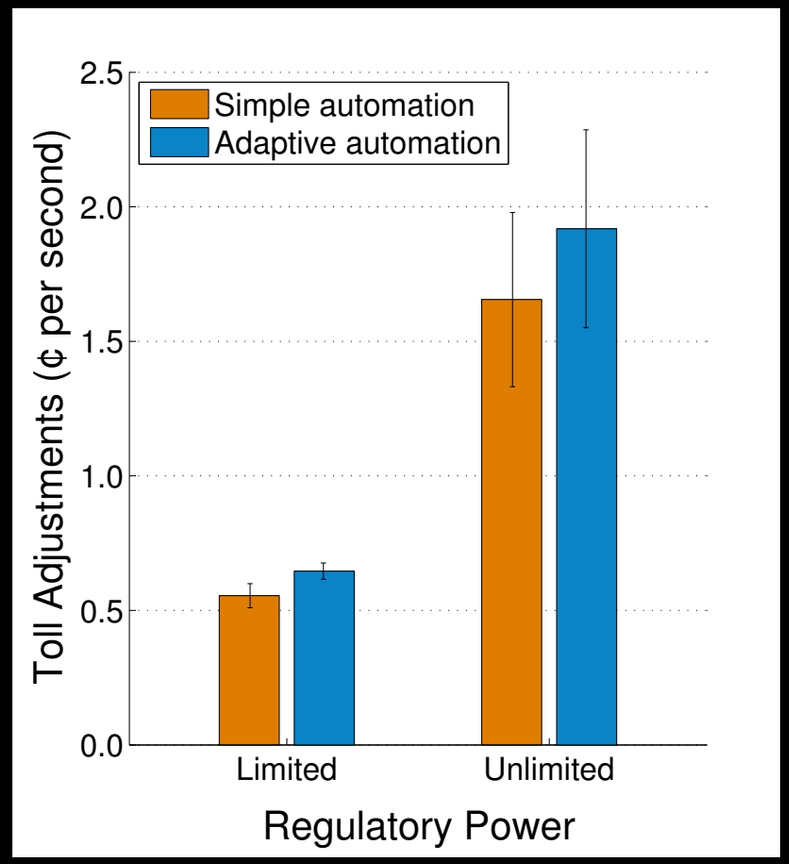
Outcome

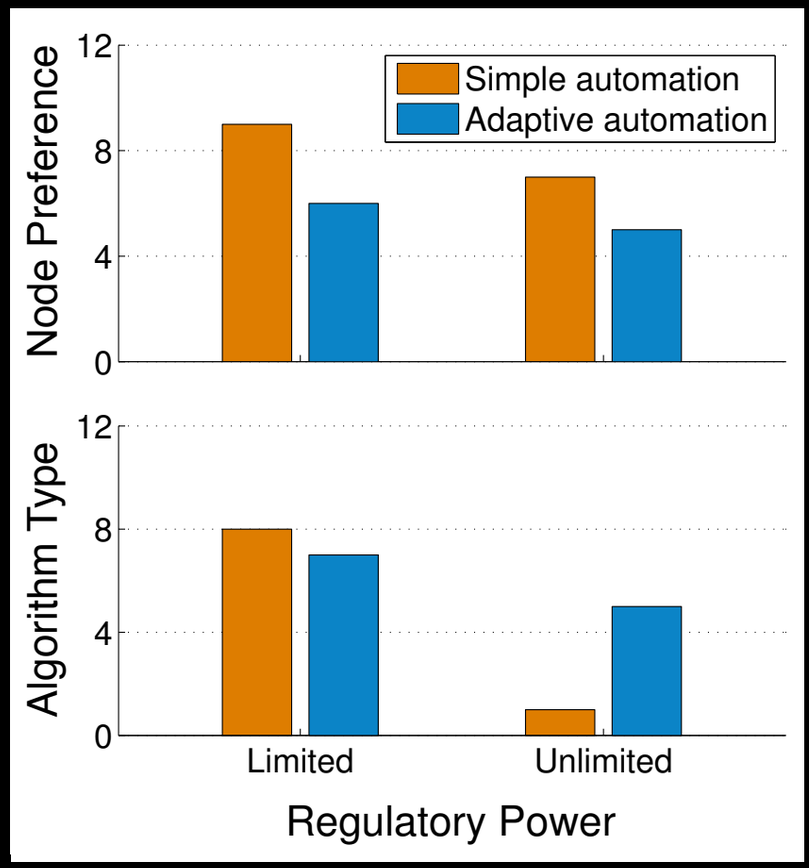


Algorithmic Sophistication

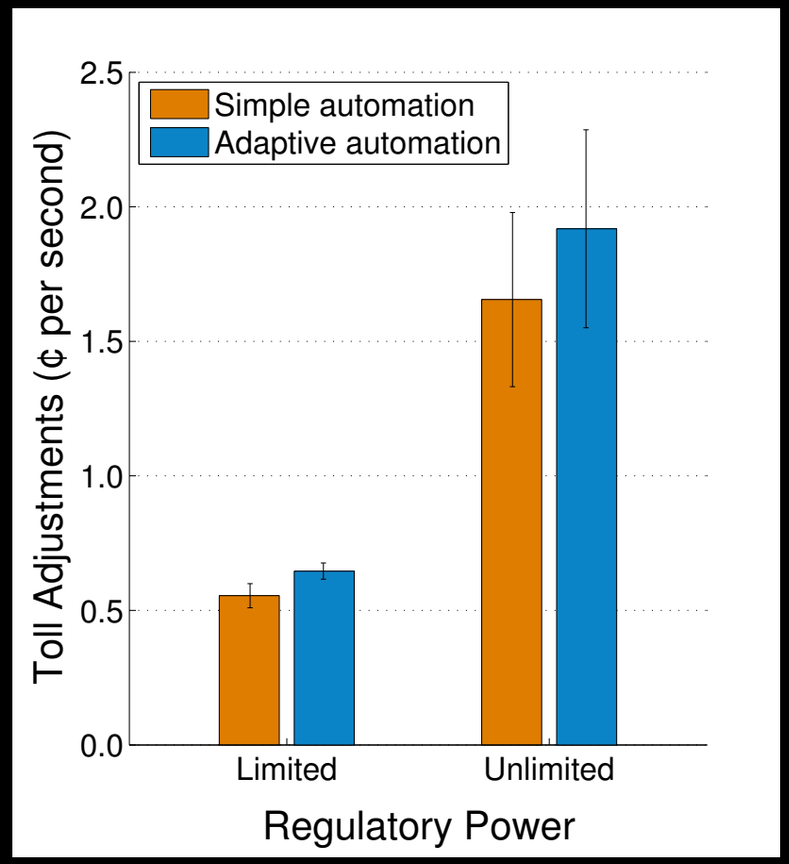
	Regulatory Power: None	Limited	Unlimited
Simple	6	1	T2
Adaptive	T4	T4	T2

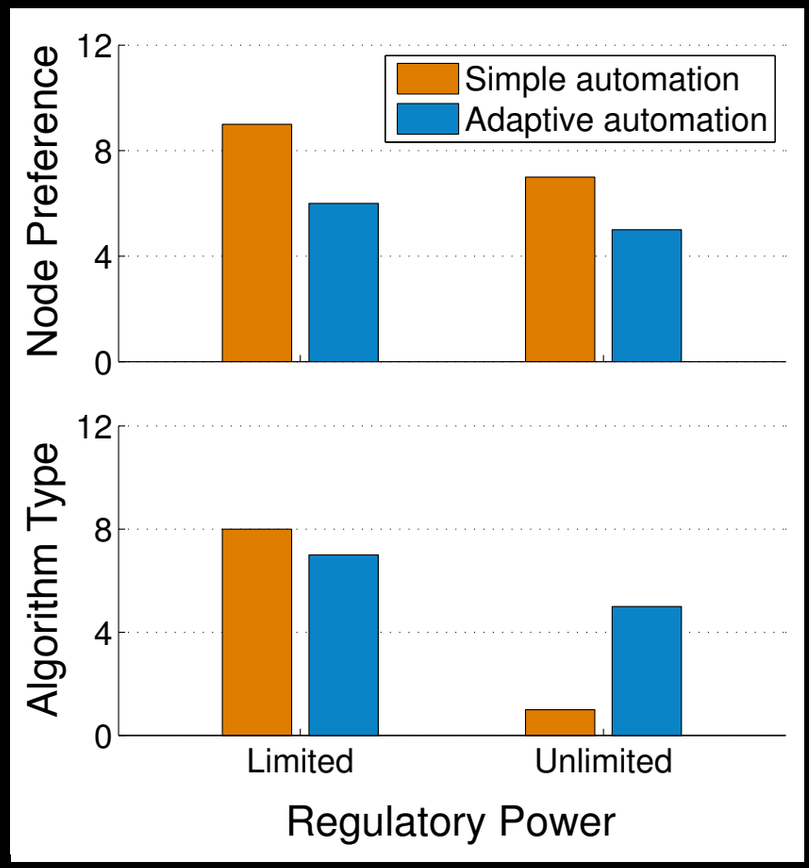
Why Simple-Unlimited?





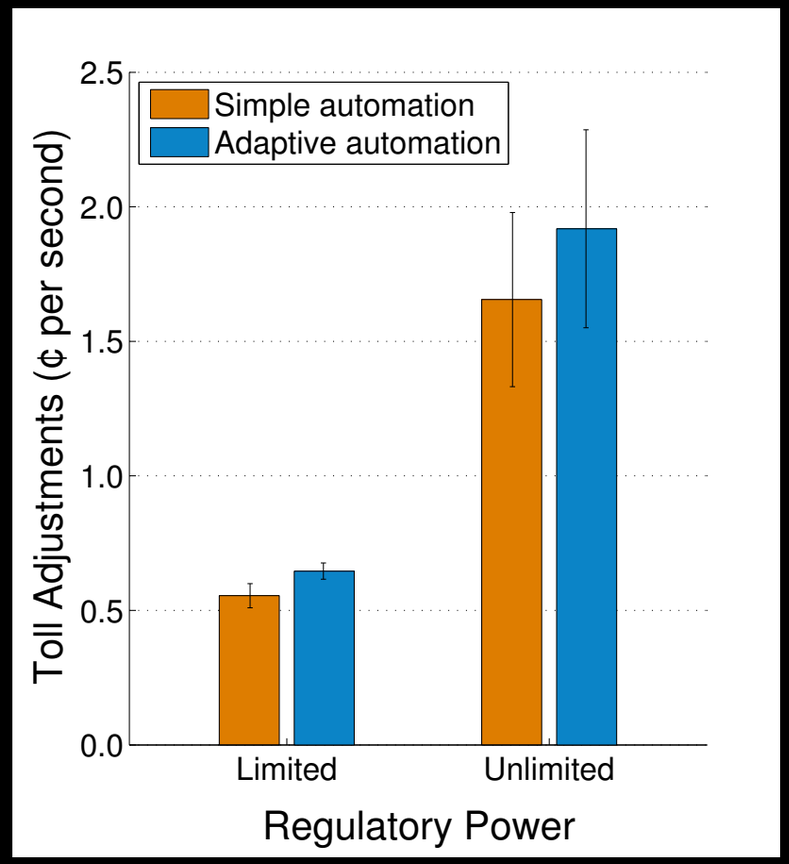
Why Simple-Unlimited?

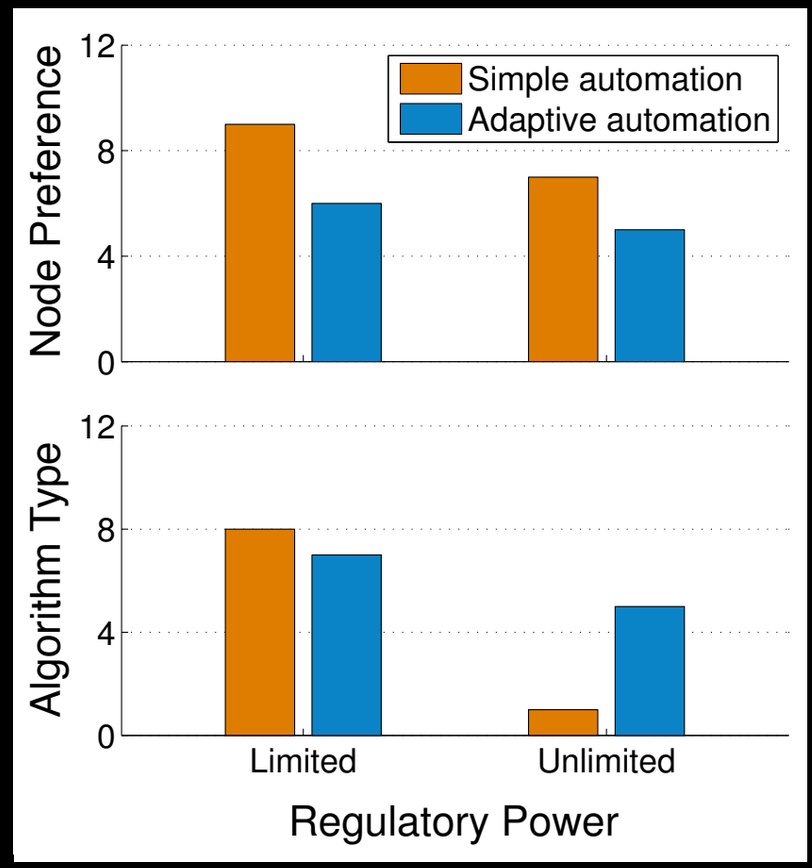




Given Unlimited Power,
Regulators used power
they didn't need

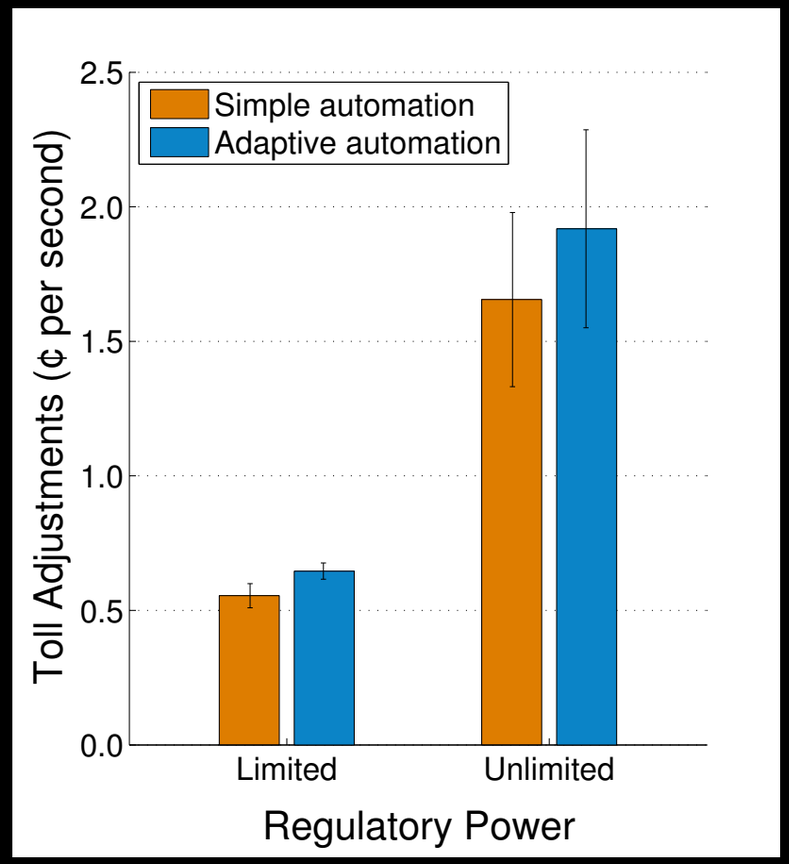
Why Simple-Unlimited?

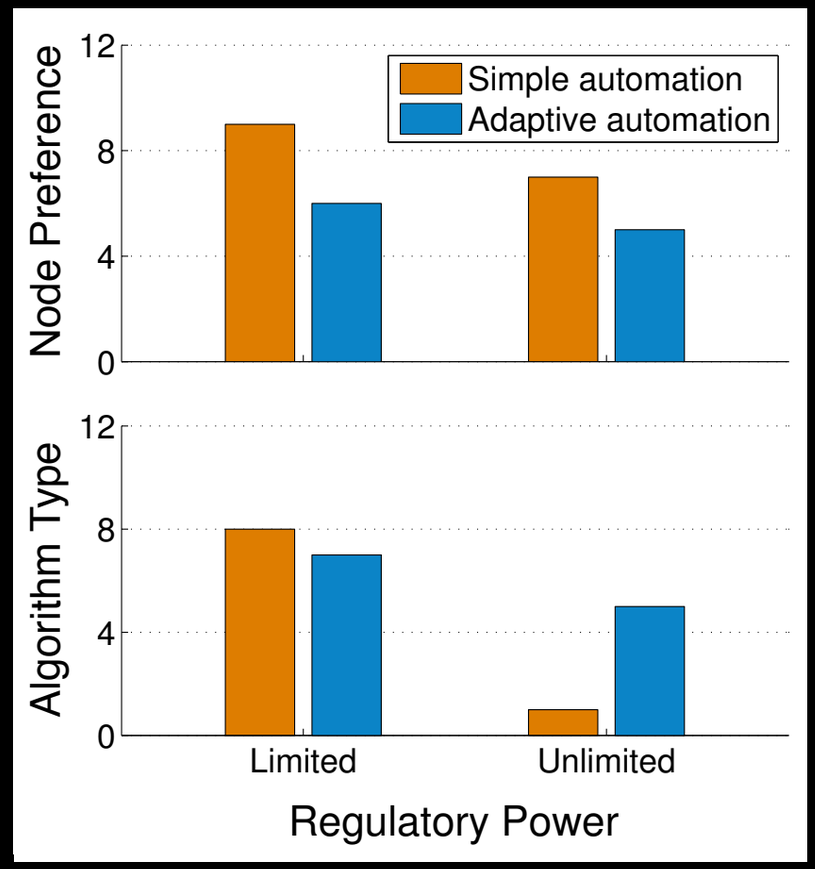




Given Unlimited Power,
Regulators used power
they didn't need

Why Simple-Unlimited?

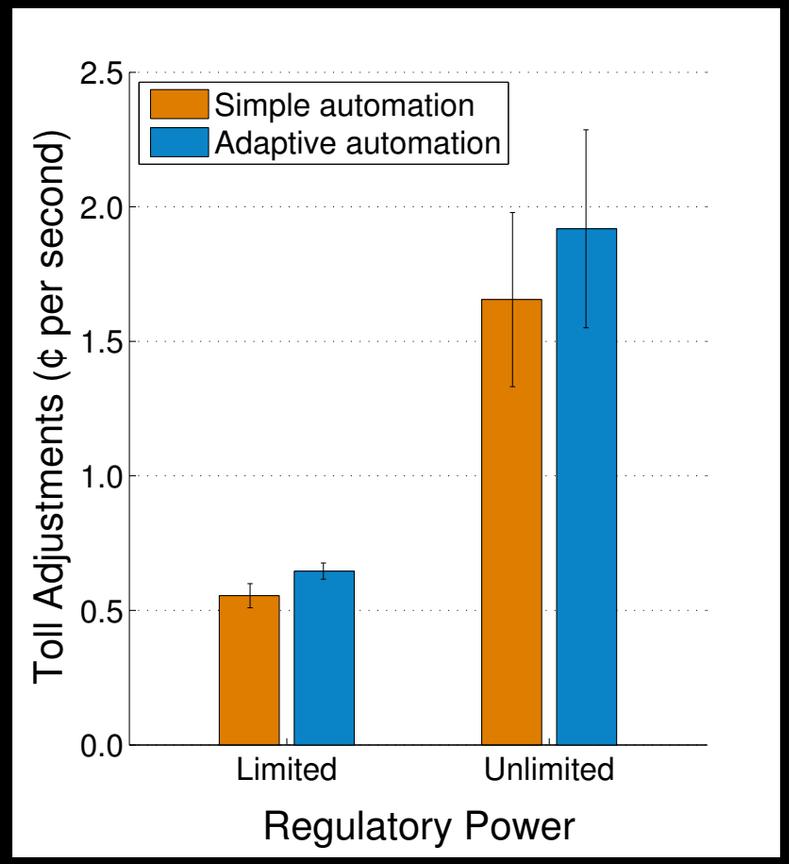


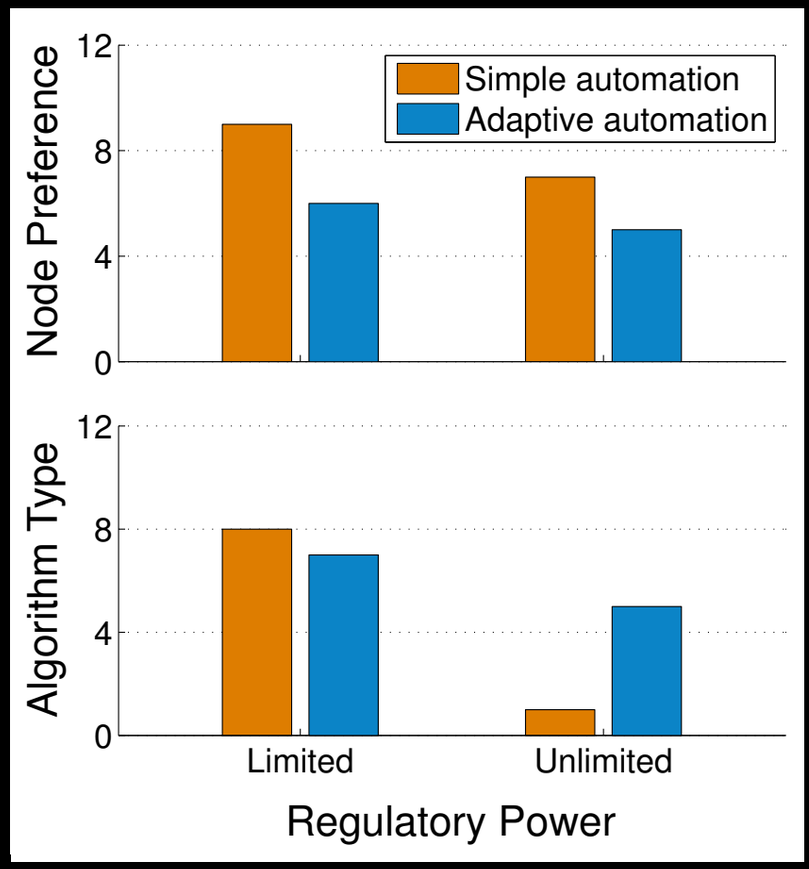


Given Unlimited Power, Regulators used power they didn't need

Regulators had poorer models of robot behavior

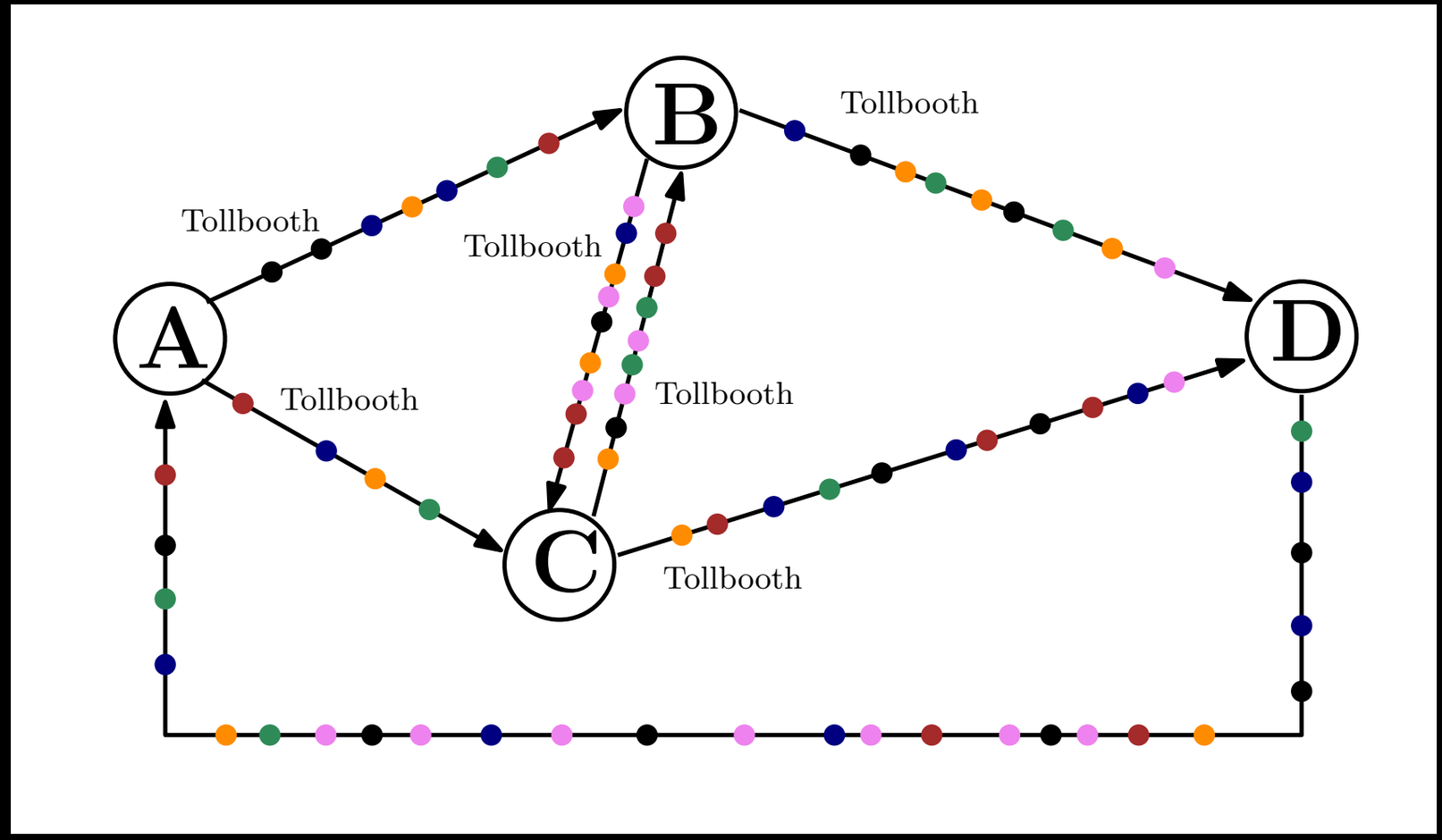
Why Simple-Unlimited?





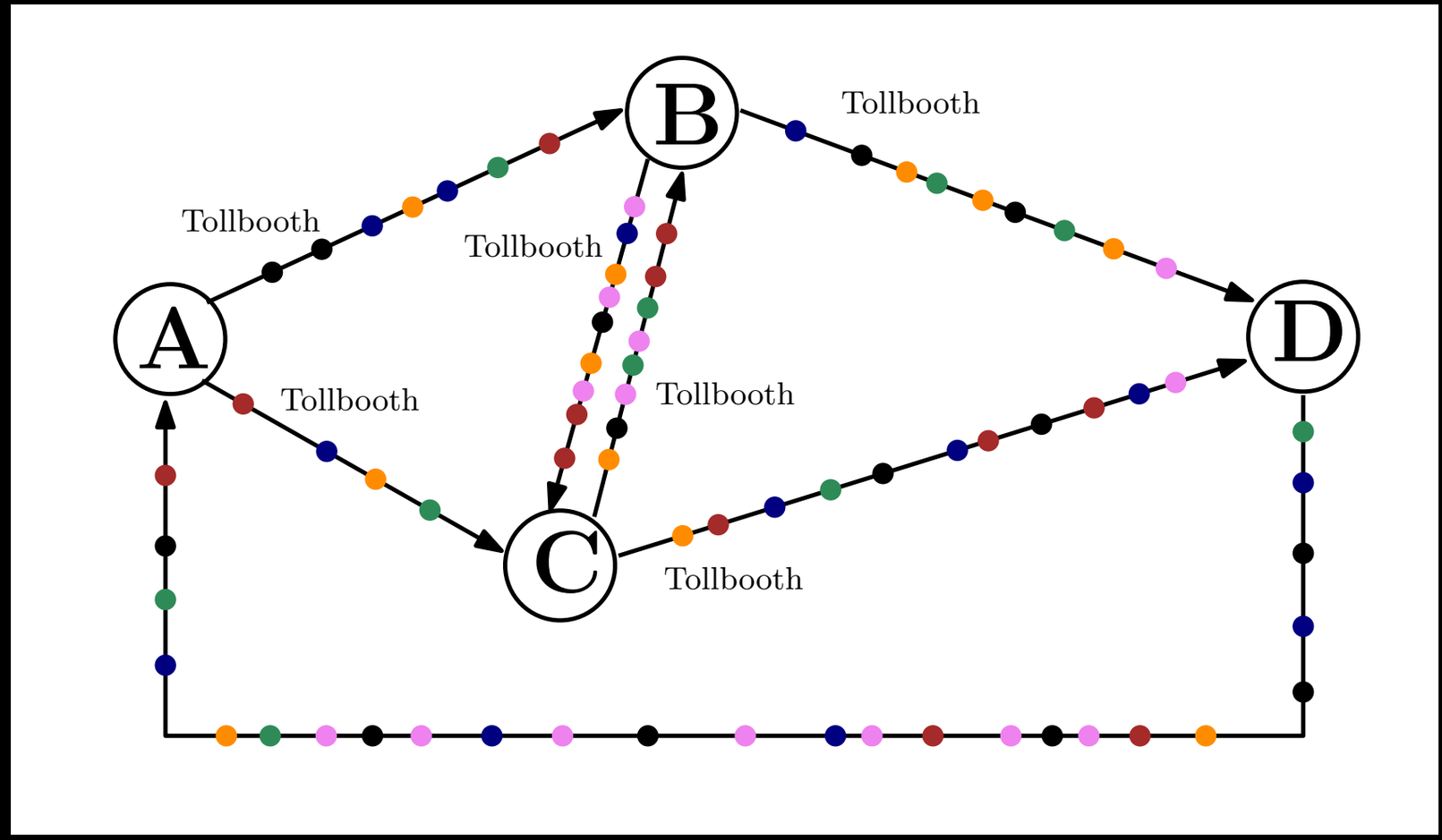
Simple automation was easier to model

Automated Help



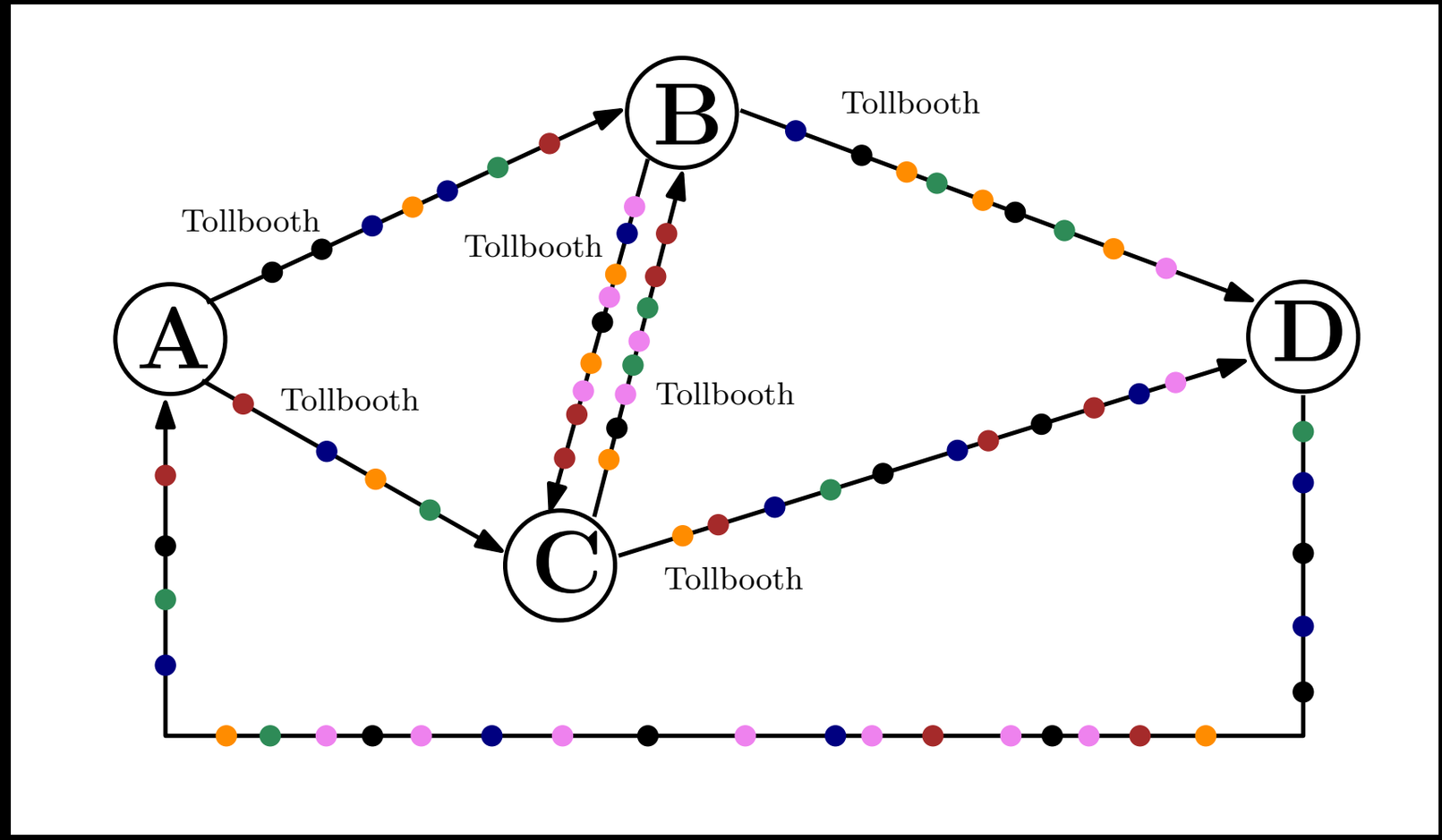
Automated Help

- Predict when the congestion will occur



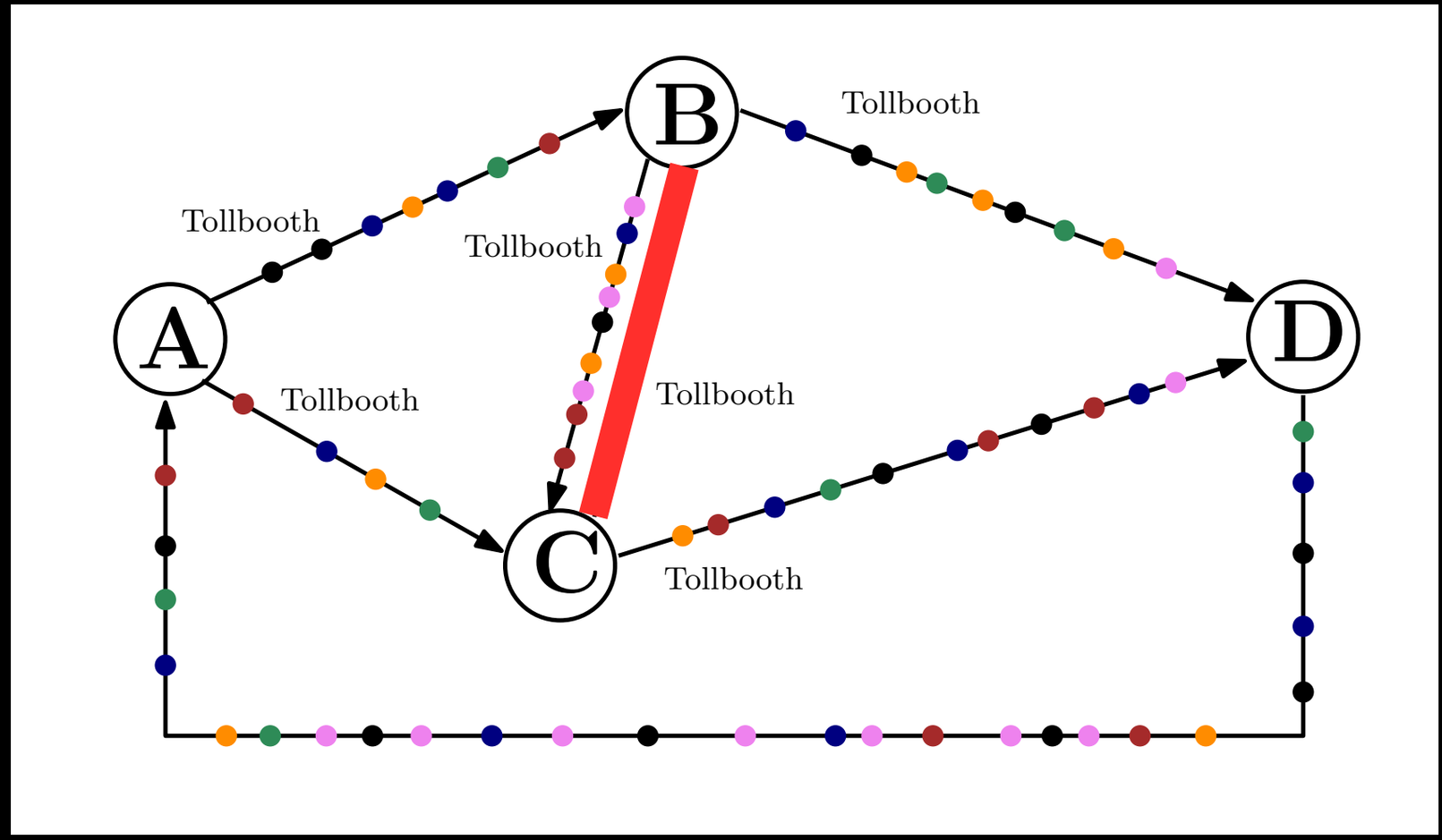
Automated Help

- Predict when the congestion will occur
- Alert the regulator of predicted congestion

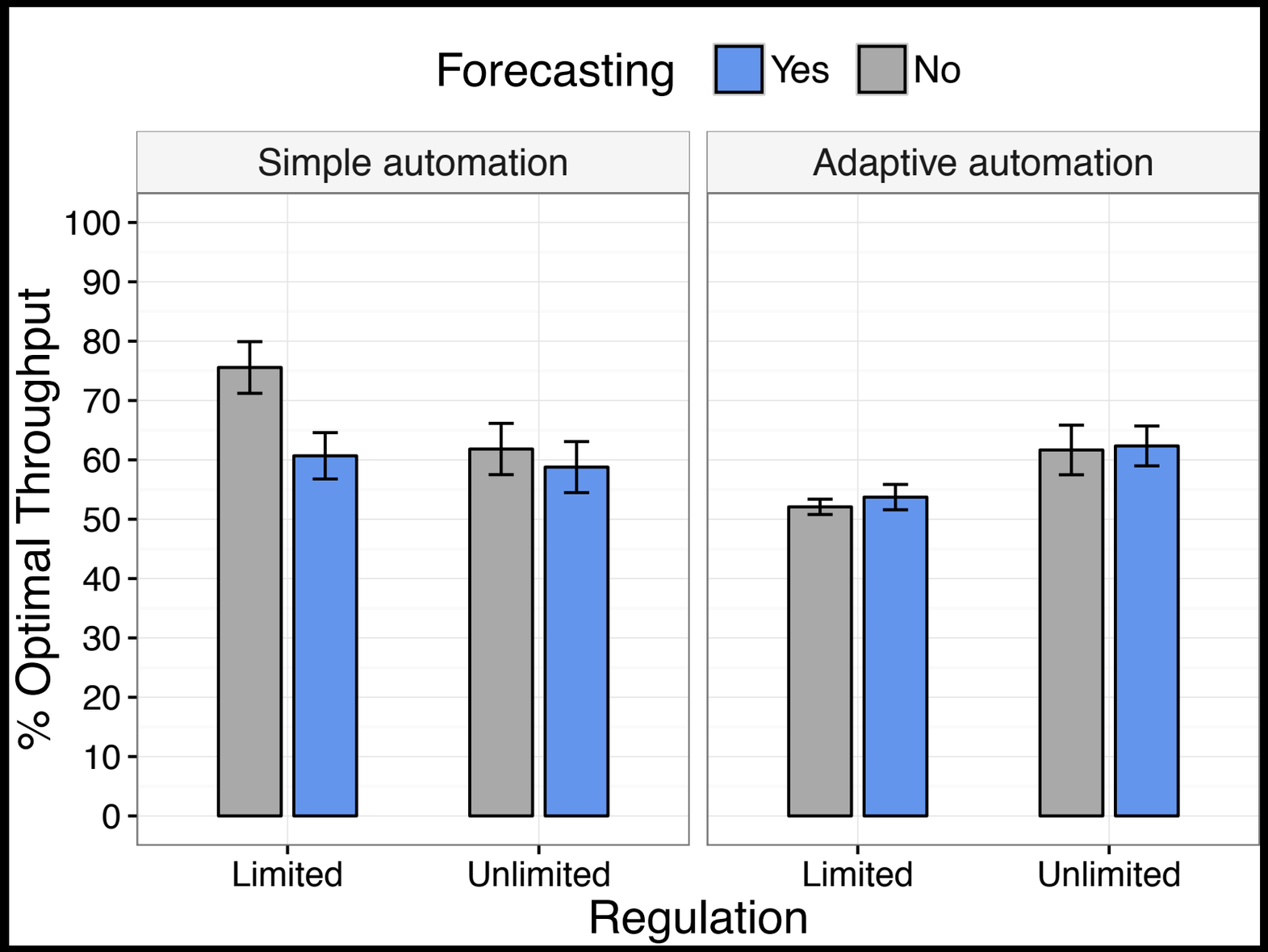


Automated Help

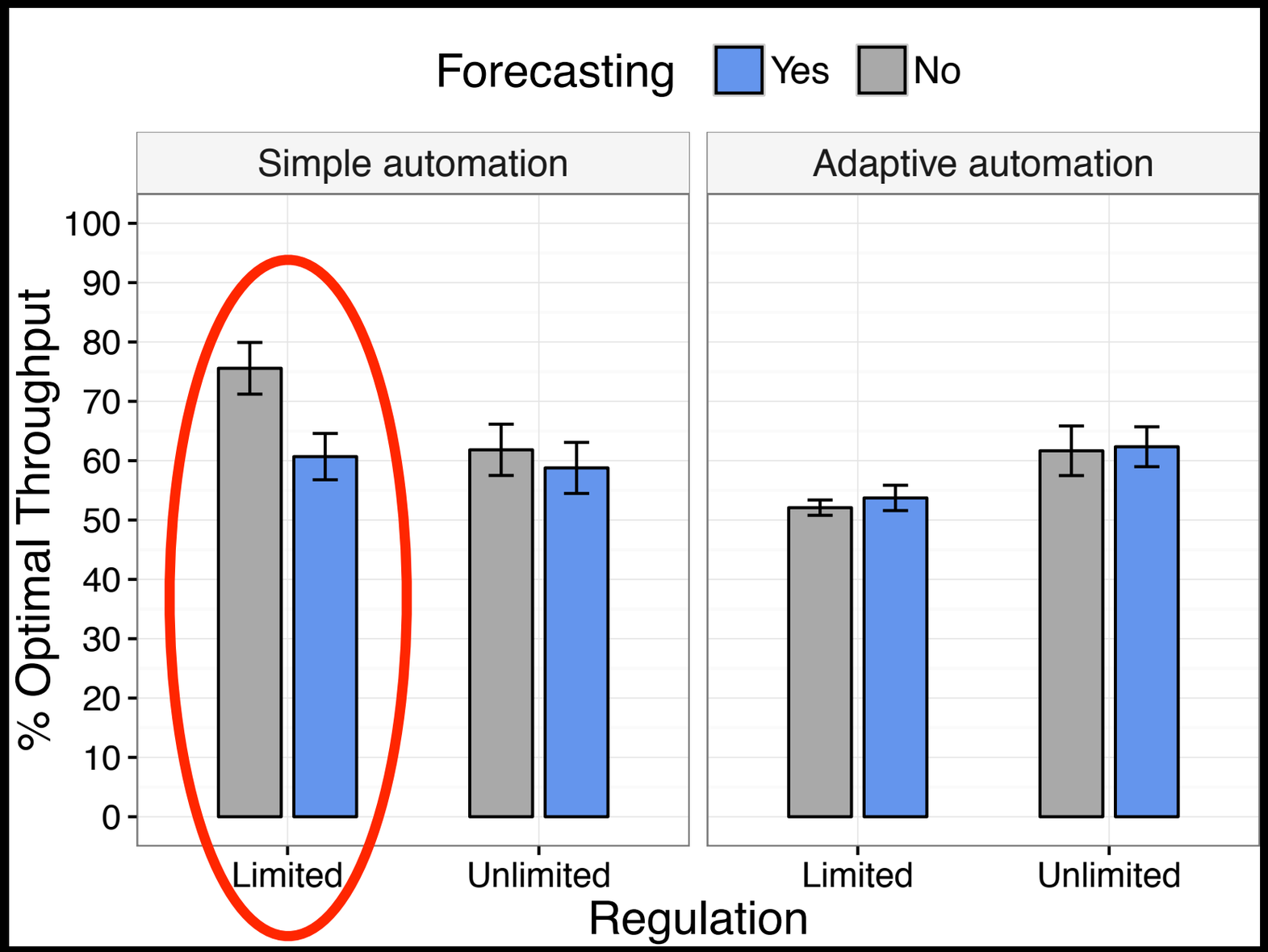
- Predict when the congestion will occur
- Alert the regulator of predicted congestion



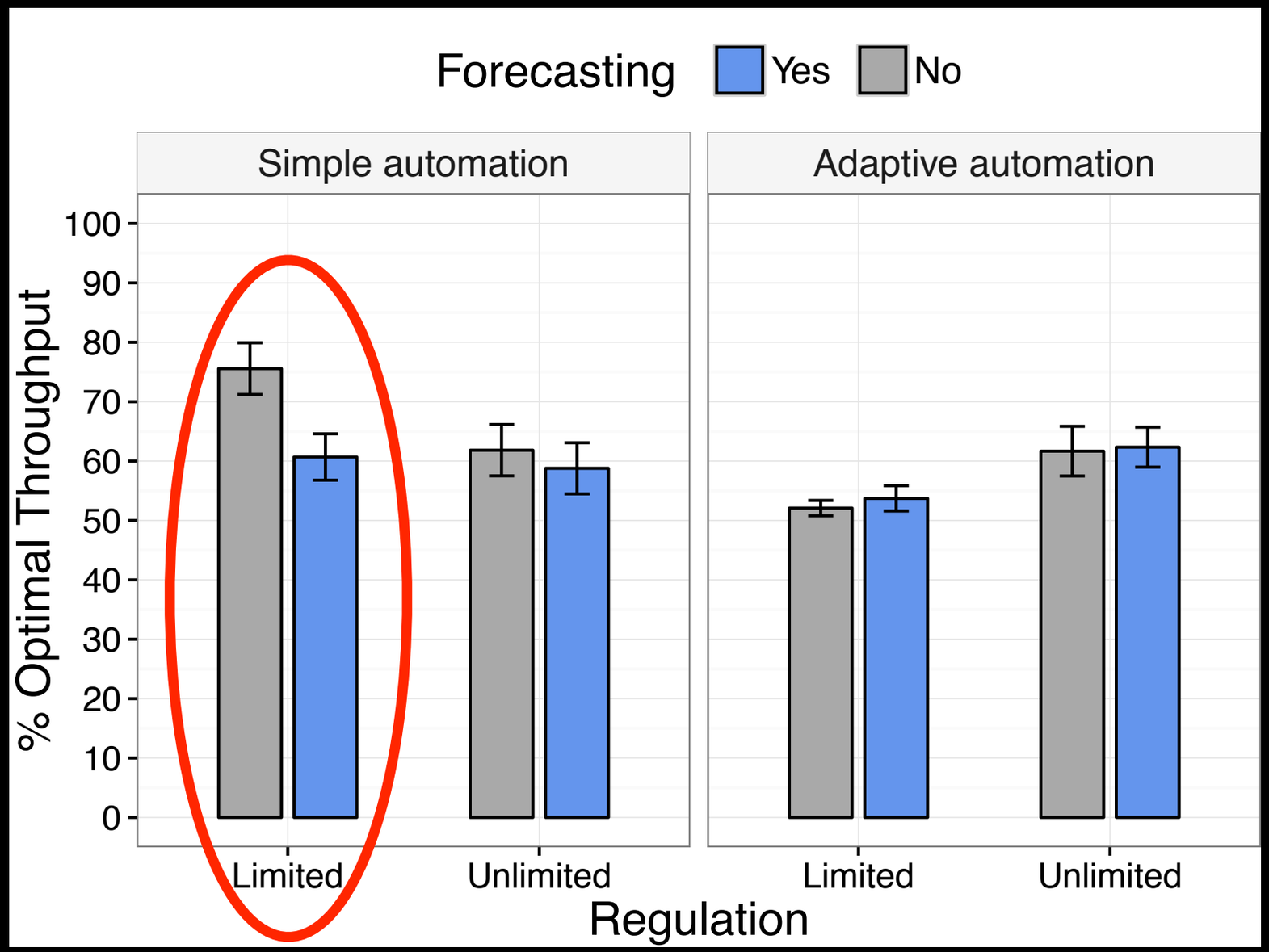
Outcome


Forecasting
Yes
No
Simple automation
Adaptive automation
% Optimal Throughput
100
90
80
70
60
50
40
30
20
10
0
Limited
Unlimited
Limited
Unlimited
Regulation

Outcome

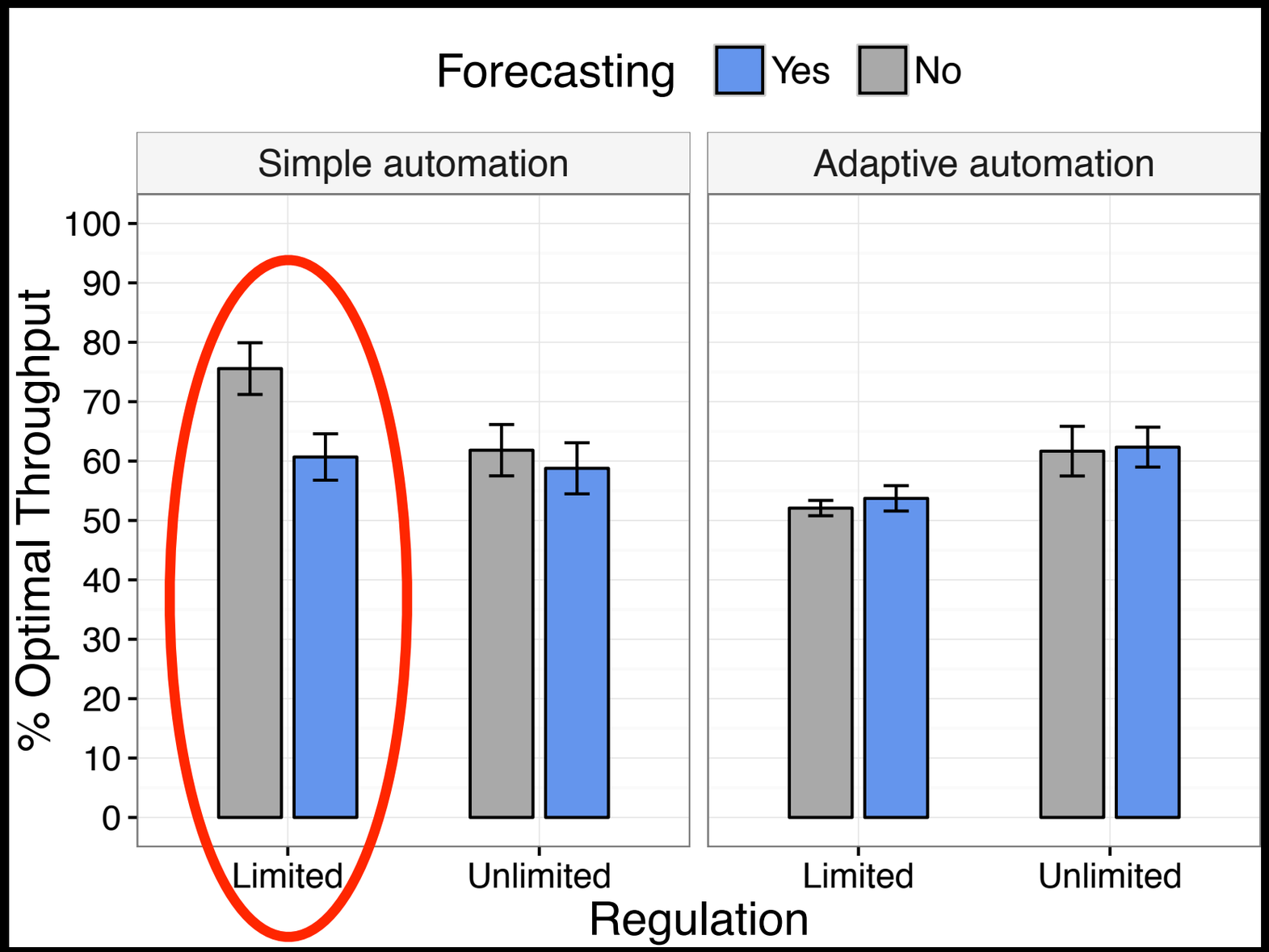

Forecasting
Yes
No
Simple automation
Adaptive automation
% Optimal Throughput
100
90
80
70
60
50
40
30
20
10
0
Limited
Unlimited
Limited
Unlimited
Regulation

Outcome



Decision support made Simple-Limited worse!

Outcome



Decision support made Simple-Limited worse!

Why? Regulators had a poorer model of the cars.

Toward a General Theory

3 “Forces”:

- Adaptive robots -> Regulator must spend more time modeling
- Adaptive robots -> Regulators need more regulatory power
- More regulator power -> Decreased time modeling robots

Conclusions and Future Work

Conclusions and Future Work

- Data points that suggest less is more

Conclusions and Future Work

- Data points that suggest less is more
  - Limited regulator power with simple robots produced the best results

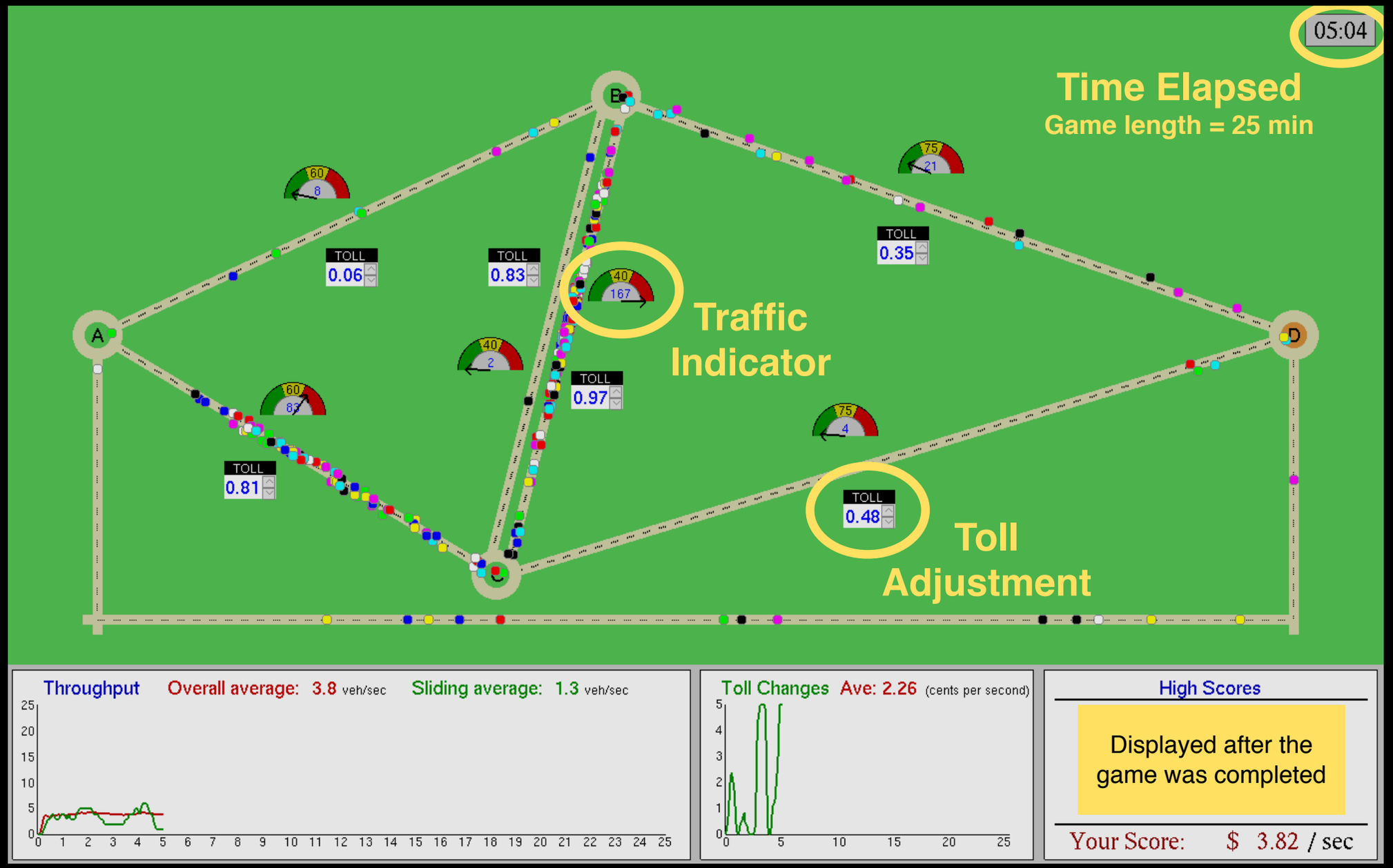
Conclusions and Future Work

- Data points that suggest less is more
  - Limited regulator power with simple robots produced the best results
- Just outliers? Or part of a general trend?

Conclusions and Future Work

- Data points that suggest less is more
  - Limited regulator power with simple robots produced the best results
- Just outliers? Or part of a general trend?
- Can we find a way to do more with more?

Extras



High Scores

Displayed after the game was completed

Your Score: $ 3.82 / sec