UCLA Henry Samueli School of Engineering and Applied Science
Electrical Engineering Department

# Robust Discrete Synthesis against Unspecified Disturbances

Rupak Majumdar, Elaine Render, Paulo Tabuada

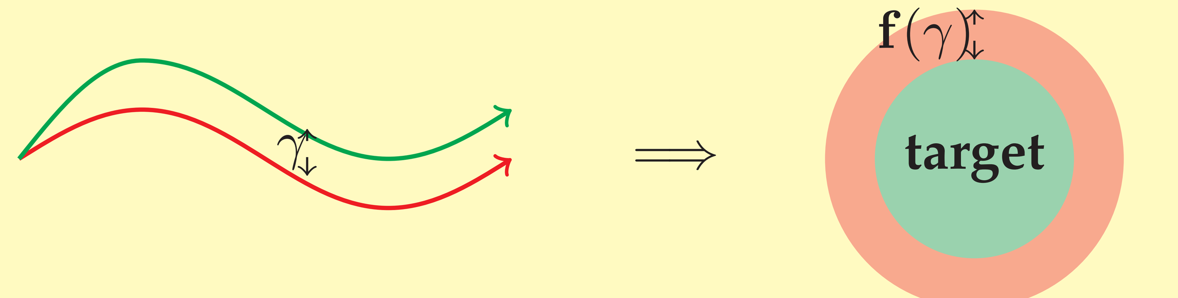
CyPhyLab

## Motivation

Reactive systems working in uncertain environments should possess a **robustness** property with respect to errors and disturbances resulting from

- A mismatch between physical system and computational model
- Sensor errors
- Changing requirements

We want to be able to build systems in which

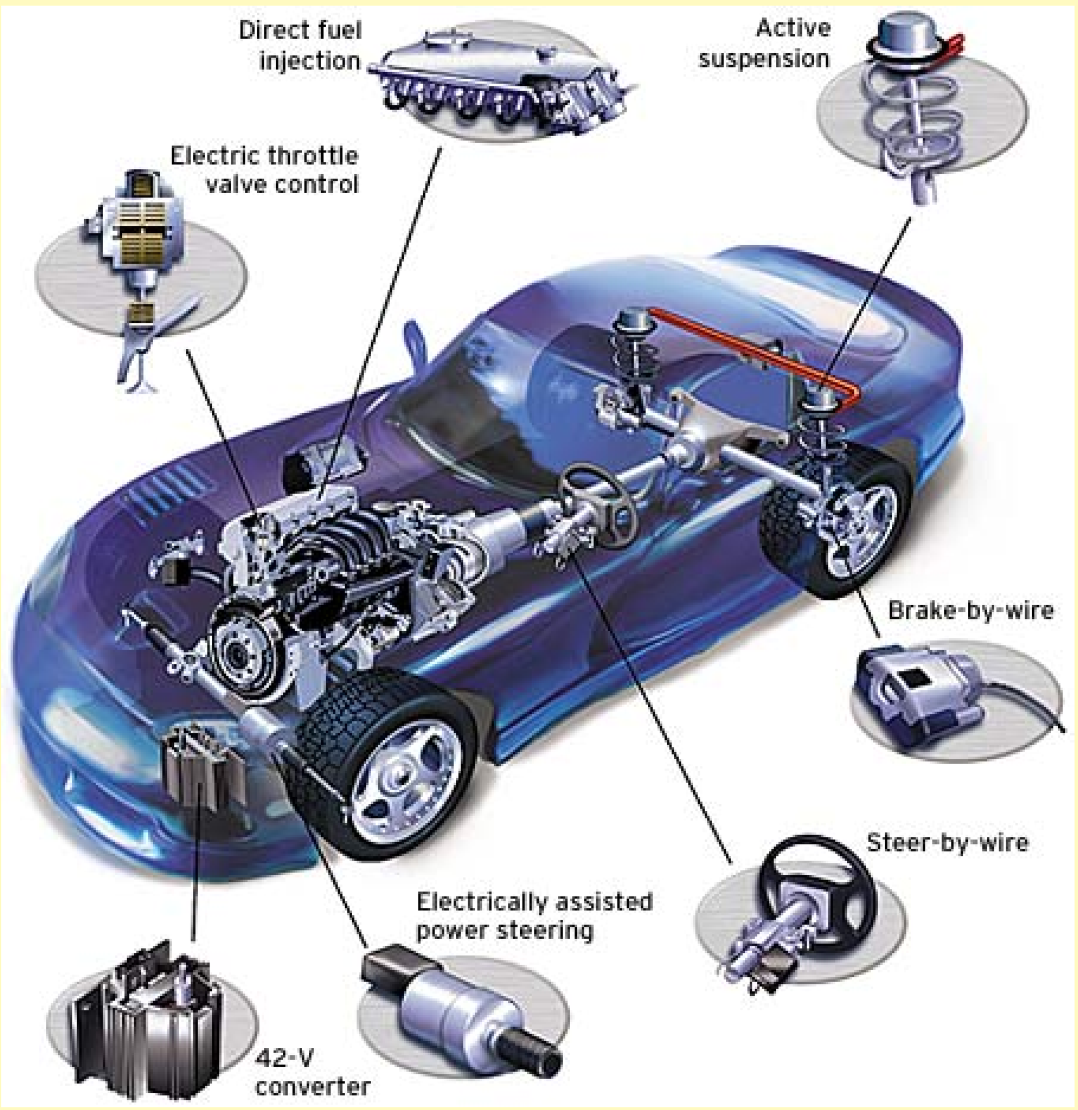



small (unmodeled) disturbances in system ⟹ small difference in outcome

**APPLICATIONS**

**TRANSIENT ERRORS:** For example in electronic circuits such as FPGAs.

**AUTOMOTIVE:**




## Extensions

- Take into account modeled adversarial behavior of the environment in addition to unmodeled disturbances
- Combine with discrete abstractions of continuous systems

## Existing theory

| Continuous systems | Discrete systems (software) |
|---|---|
| Controller synthesis | Supervisory controller synthesis using games on graphs |
| There exists a theory of robust control | **What is the corresponding theory here?** |

The existing boolean approach is too coarse to distinguish "small" errors.

**WANT:** To develop a theory of robustness for discrete systems.

**PROBLEM:** No notion of topology or metric - how do we quantify the effect of an error?

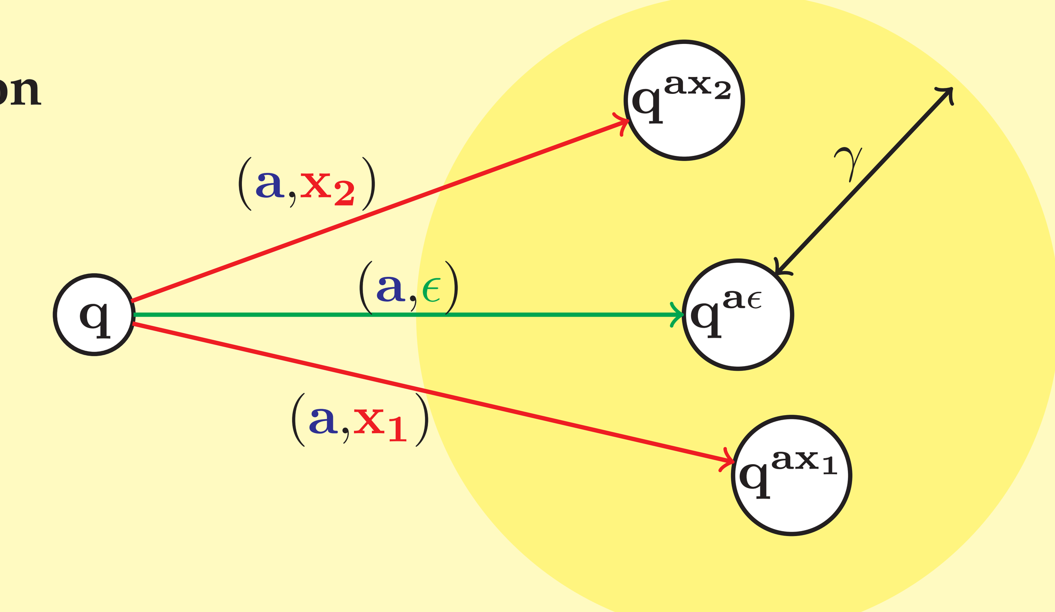
**OUR SOLUTION:** Metric automata.

## Metric Automata

A metric automaton is an automaton augmented with a distance function on $Q$. Both the system and the environment control the actions of the automaton; the effect of the environment's input is bounded by $\gamma$ as shown in the right hand picture. The environment input $\epsilon \in X$ denotes the nominal outcome - the outcome which would result if no disturbances were present.

$$\mathbf{A} = ((\mathbf{Q}, \mathbf{d}), \underbrace{\Sigma \times X}, \mathbf{q_0}, \delta, \mathbf{Acc}, \gamma)$$

**transition function**
**initial state**
**acceptance condition**
**disturbance bound**
**system & environment input alphabets**
**metric space of states**




A (memoryless) *strategy* is a function $S : Q \to \Sigma$ which specifies an input choice at each state in $Q$. A strategy is *winning* if, regardless of which inputs the environment chooses, the acceptance condition of $A$ is still satisfied.

## Main Result

**Let $A$ be a metric automaton with either:**

**a (finite) reachability acceptance condition** OR
**an (infinite) $\omega$-regular acceptance condition.**

**There exists an algorithm to construct a winning strategy for an inflated acceptance condition in polynomial time, where the size of the inflation depends linearly on $\gamma$.**

## Key idea: control Lyapunov functions

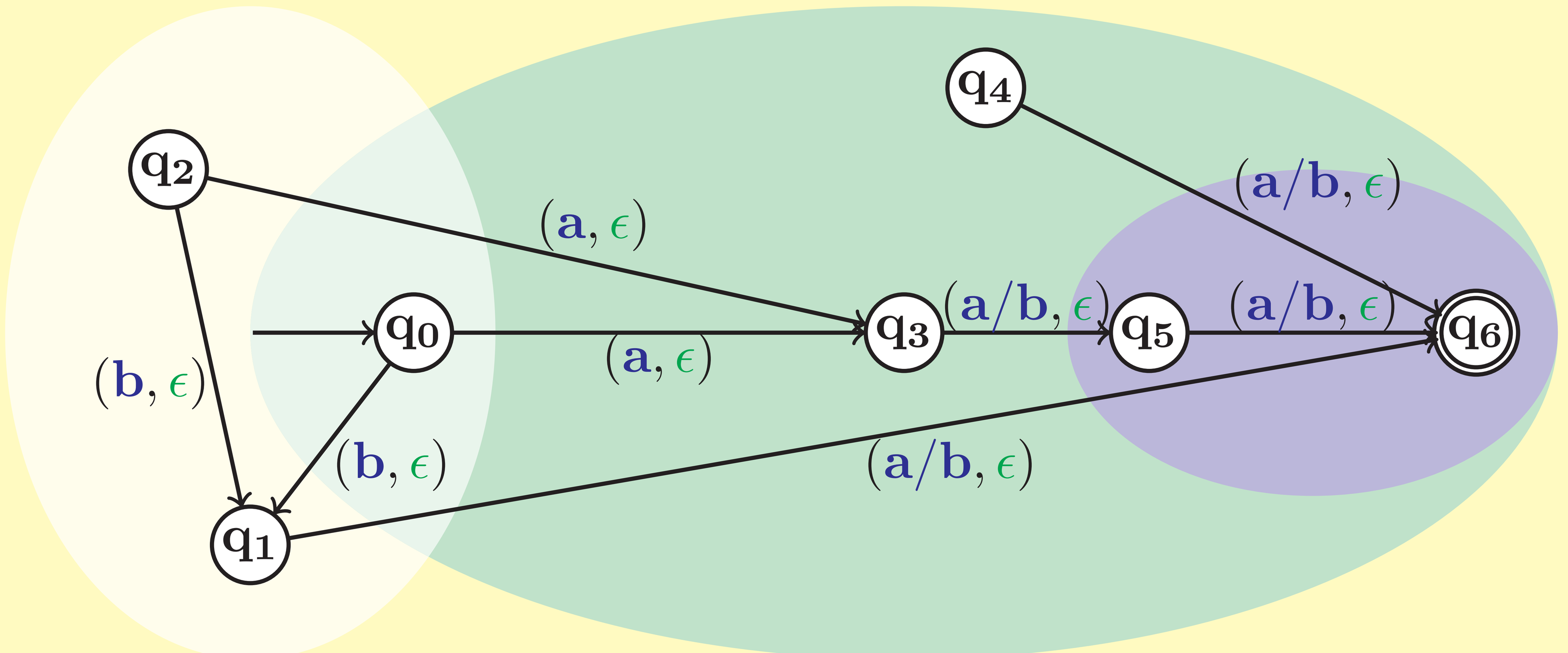
Our approach to strategy synthesis relies on the construction of a special type of *rank function*. The following definition is inspired by methods in continuous control.

A function $V : Q \to \mathbb{R}_0^+$ satisfying suitable conditions is a **control Lyapunov function** if for all $q \in Q$ there exists $a \in \Sigma$ such that

$$\mathbf{V}(\mathbf{q}^{\mathbf{a}\epsilon}) - \mathbf{V}(\mathbf{q}) \leq -\mathbf{f}(\mathbf{d}(\mathbf{q}, \mathbf{Acc}))$$

where $d(q, Acc)$ is the distance between $q$ and $Acc$ and $f$ is a non-linear gain.

## Example




Consider the nominal reachability automaton shown to the left. The disturbance is bounded by $\gamma = 1$. The relative distances of the states in $Q$ are approximated by their layout in the graph. For example, choosing $b$ at state $q_0$ could result in reaching any one of the states in the cream ellipse under the effects of a disturbance.

**SHORTEST PATH STRATEGY** Define $S_1 : Q \to \Sigma$, $q \mapsto b$ for all $q \in Q$. This strategy chooses the shortest path to $q_6$ from every state. Under a disturbance bounded by $\gamma$, the strategy $S_1$ can guarantee that a state in the green ellipse will be reached in finite time.

**OUR STRATEGY** Define $S_2 : Q \to \Sigma$, $q \mapsto a$ for all $q \in Q$. A control Lyapunov function was synthesized for the automaton using our polynomial time algorithm. This is achieved by solving an optimal reachability problem with a suitably defined cost function. The strategy $S_2$ is then induced from the resulting control Lyapunov function. Under a disturbance bounded by $\gamma$, the strategy $S_2$ can guarantee that a state in the blue ellipse will be reached in finite time.