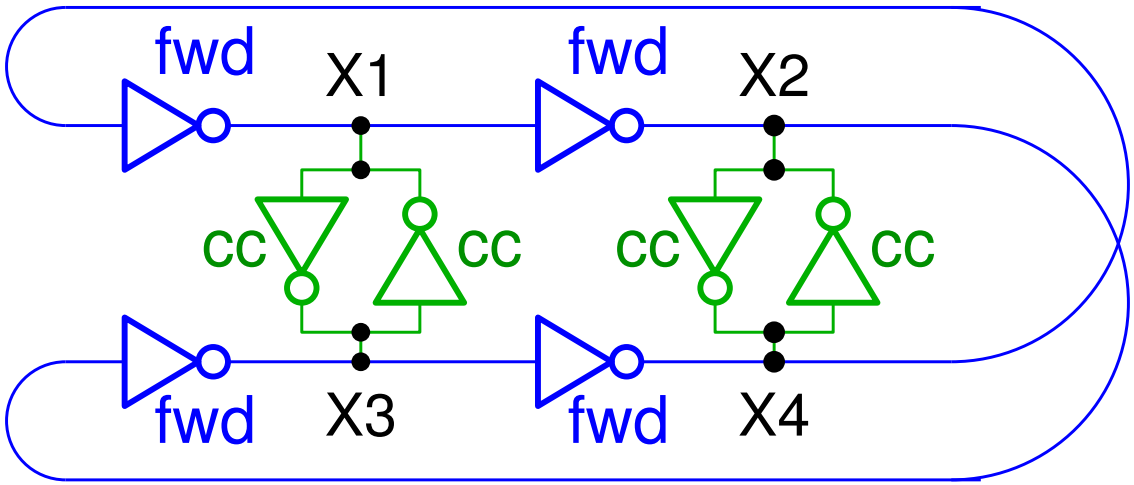
Using Model-Predictive Control and Linear Programming to Bridge Simulation and Formal Verification

Shabab Hossain and Mark Greenstreet

Google and the University of British Columbia

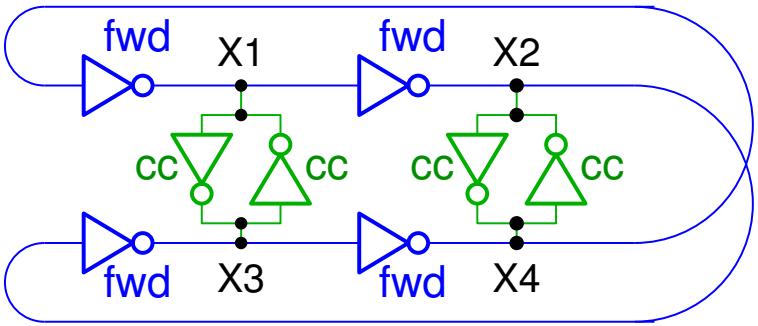
FAC – Feb. 14, 2013

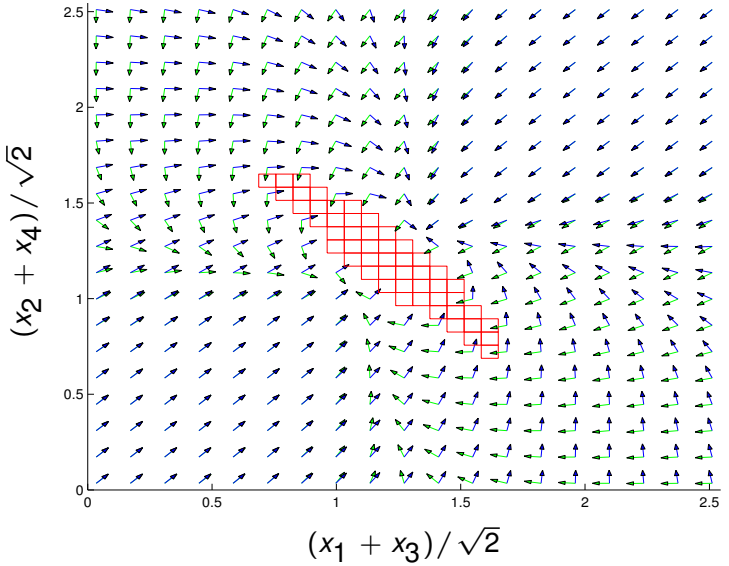
Verifying a Differential Ring Oscillator



- **Proposed by Jones *et al.*, DCC'2008**
- **For what sizes of forward and cross-coupled inverters, does the oscillator start from all initial conditions?**
- Solution (Yan and Greenstreet, FMCAD 2012)
  - Establish differential operation.
  - Show escape from neighborhood of unstable equilibrium.
  - Show unique, periodic attractor.

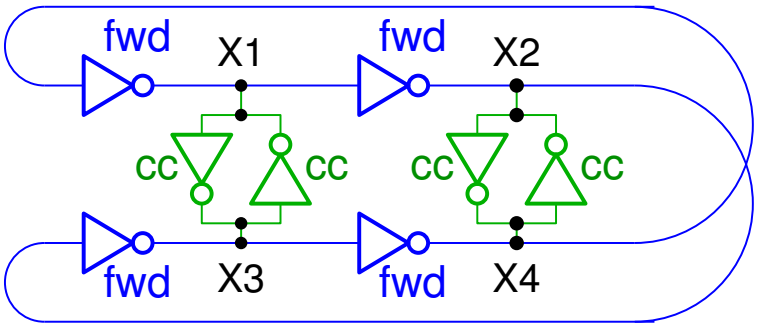
Verifying a Differential Ring Oscillator

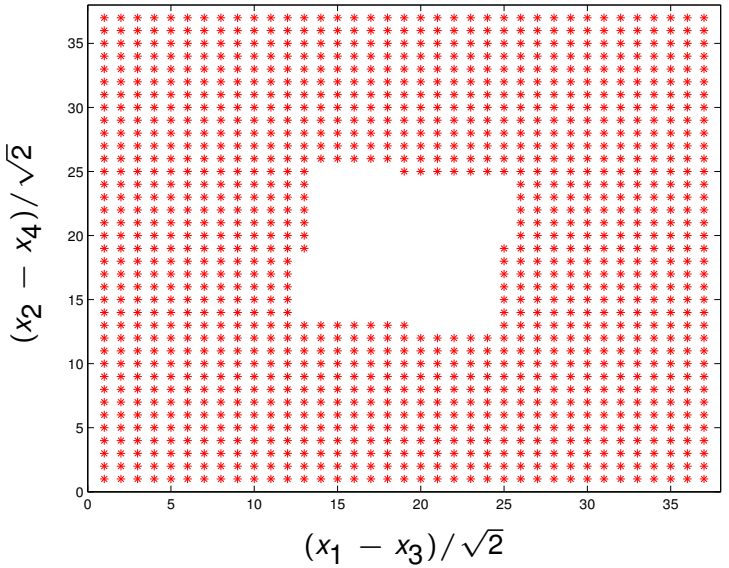




- Proposed by Jones *et al.*, DCC'2008
- For what sizes of forward and cross-coupled inverters, does the oscillator start from all initial conditions?
- Solution (Yan and Greenstreet, FMCAD 2012)
  - **Establish differential operation.**
  - Show escape from neighborhood of unstable equilibrium.
  - Show unique, periodic attractor.

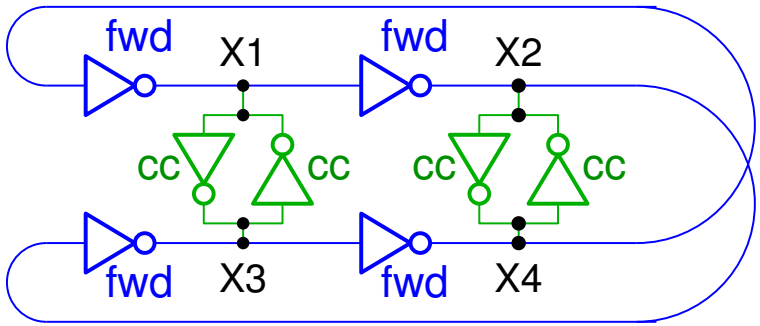
Verifying a Differential Ring Oscillator

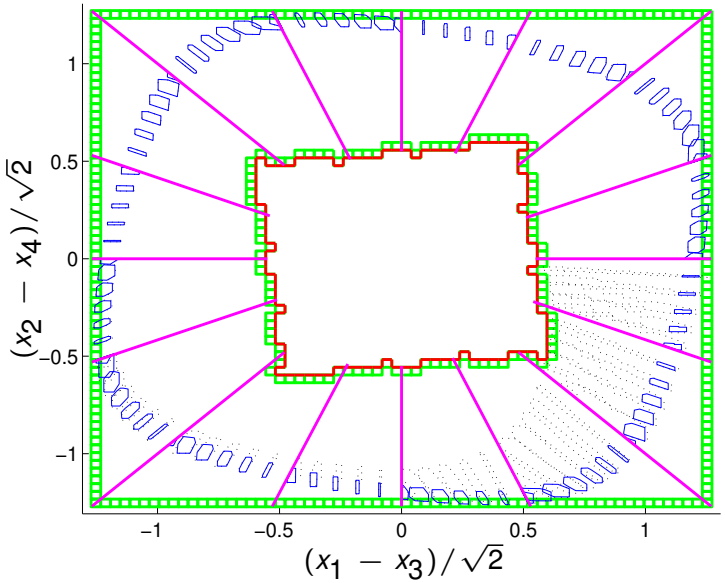




- Proposed by Jones *et al.*, DCC'2008
- For what sizes of forward and cross-coupled inverters, does the oscillator start from all initial conditions?
- Solution (Yan and Greenstreet, FMCAD 2012)
  - Establish differential operation.
  - **Show escape from neighborhood of unstable equilibrium.**
  - Show unique, periodic attractor.

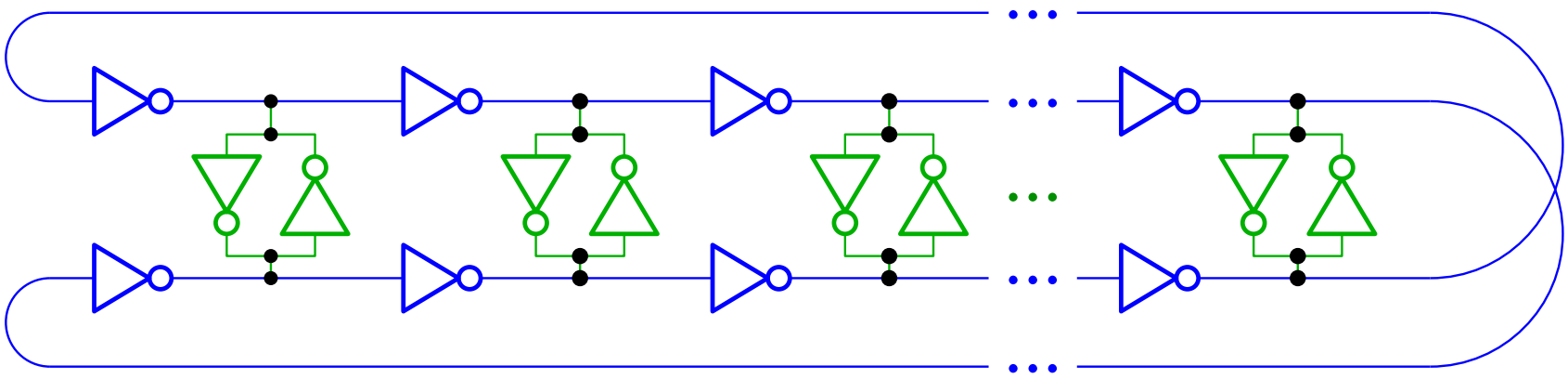
Verifying a Differential Ring Oscillator





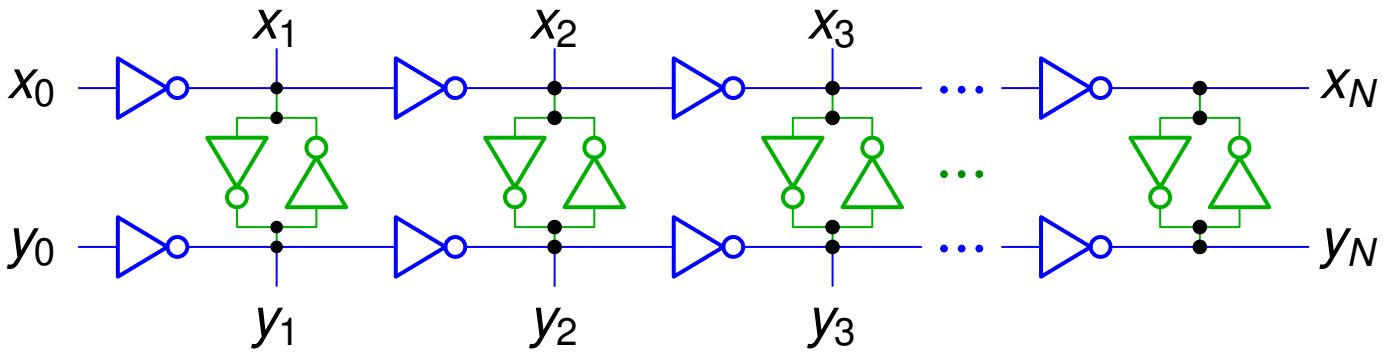
- Proposed by Jones *et al.*, DCC'2008
- For what sizes of forward and cross-coupled inverters, does the oscillator start from all initial conditions?
- Solution (Yan and Greenstreet, FMCAD 2012)
  - Establish differential operation.
  - Show escape from neighborhood of unstable equilibrium.
  - **Show unique, periodic attractor.**

Parameterized Verification

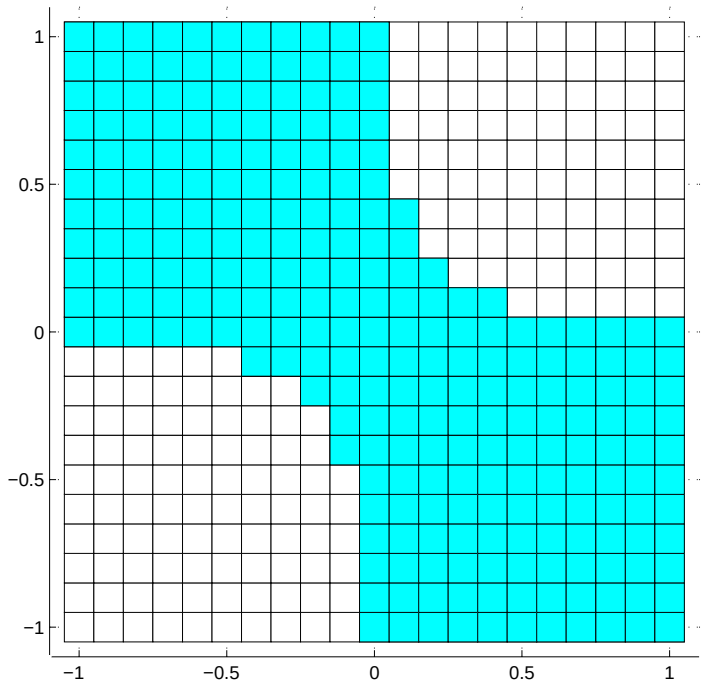


- Consider an N-stage oscillator
  - harmonic modes need to be considered
  - are there other failure modes?
- Regular structures are common in AMS designs:
  - switched capacitor VCOs
  - switchable transistor fingers to adjust drive strength/impedance
  - offset-nulling for sense-amps
  - DACs, ADCs, etc.
- Can we verify for all N?
  - Strategy: use the 2-stage verification as a starting point.
  - Parameterize each proof step for N-stage oscillator.

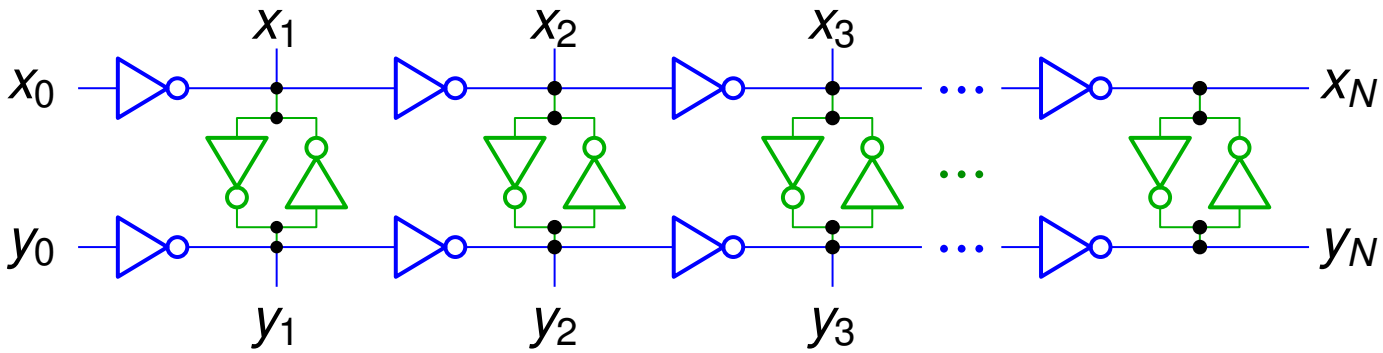
Verifying Differential Operation



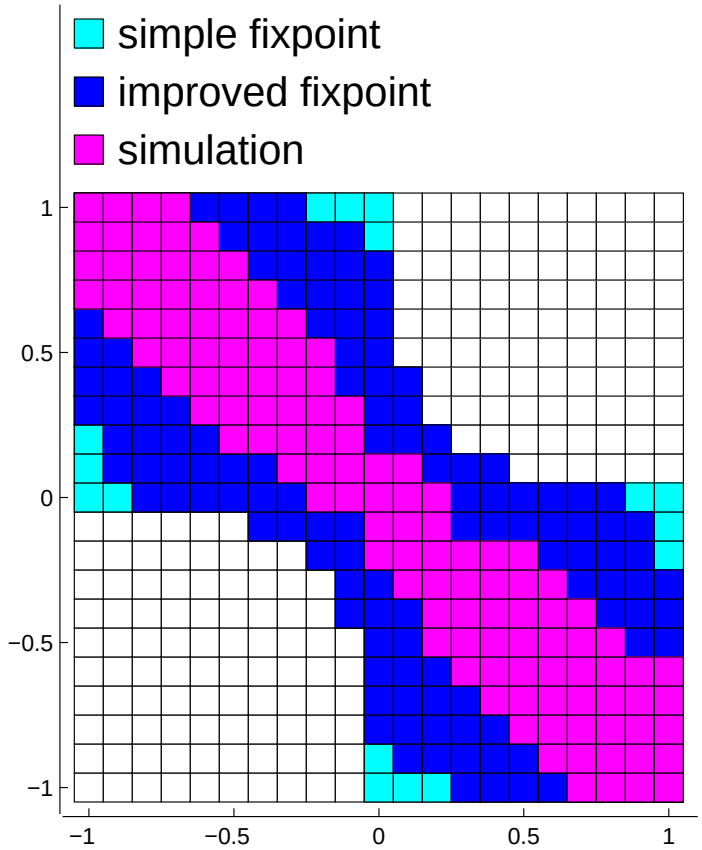
- Allow x_0 and y_0 to be arbitrary waveforms bounded by ± 1.
- For x_{i+1} and y_{i+1} divide $[-1, +1]^2$ into cells, and compute "next-cell" relation.
- Find strongly-connected components of this relation – there's exactly one.
- Discard all other cells.
- Repeat until a fix point is reached.



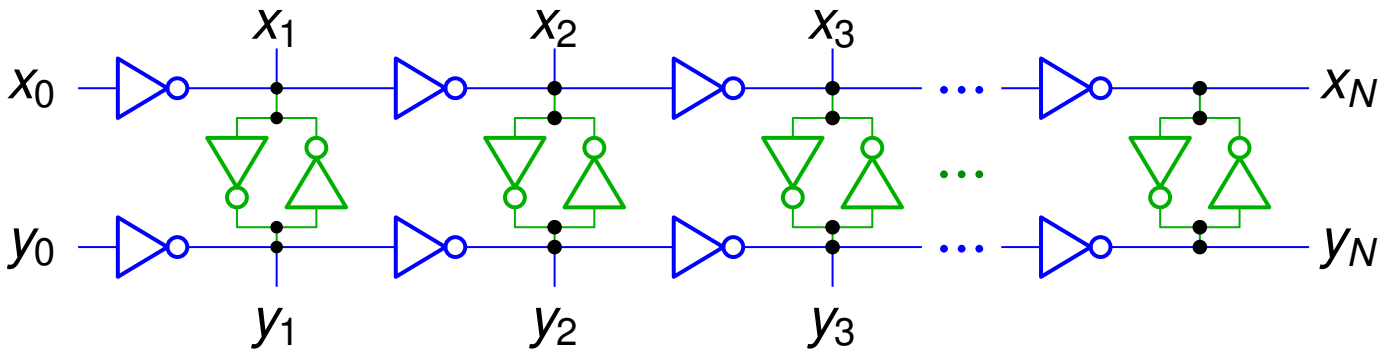
Tightening the Over-Approximation



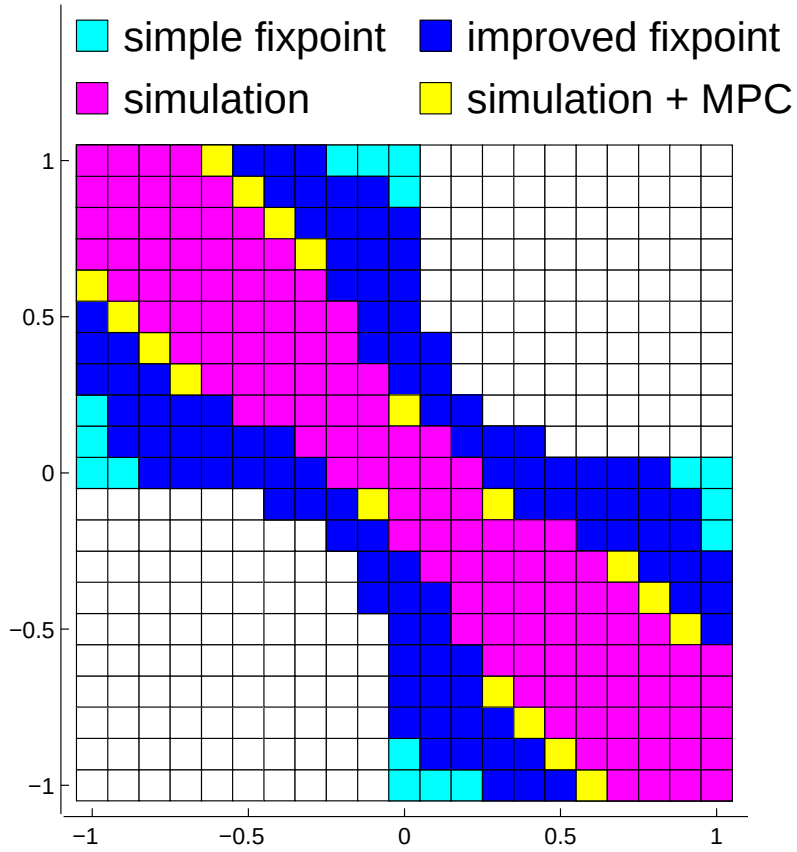
- Fix point region is way bigger than for 2-stage case.
- We tightened the fix point by tracking intervals where each box-face can be crossed.
- The result is still much larger than what was reached by simulation.



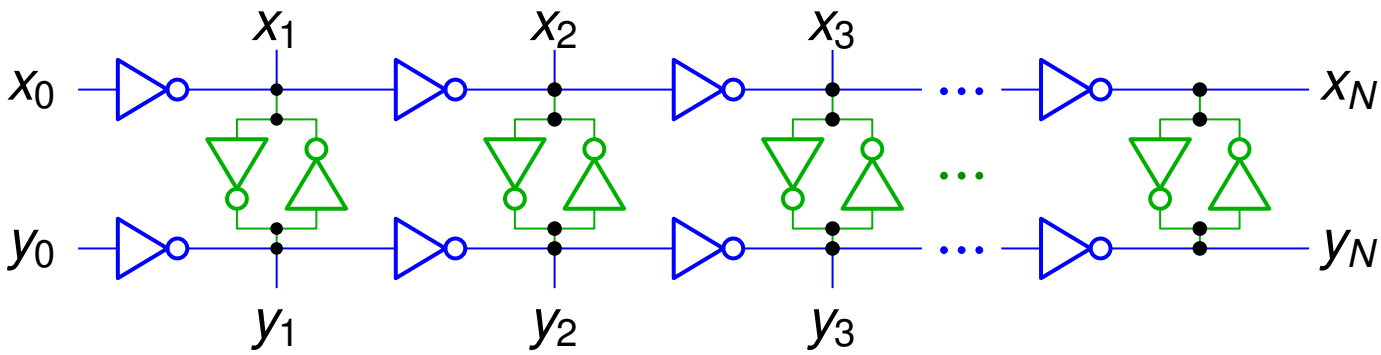
Model-Predictive Control for Better Coverage



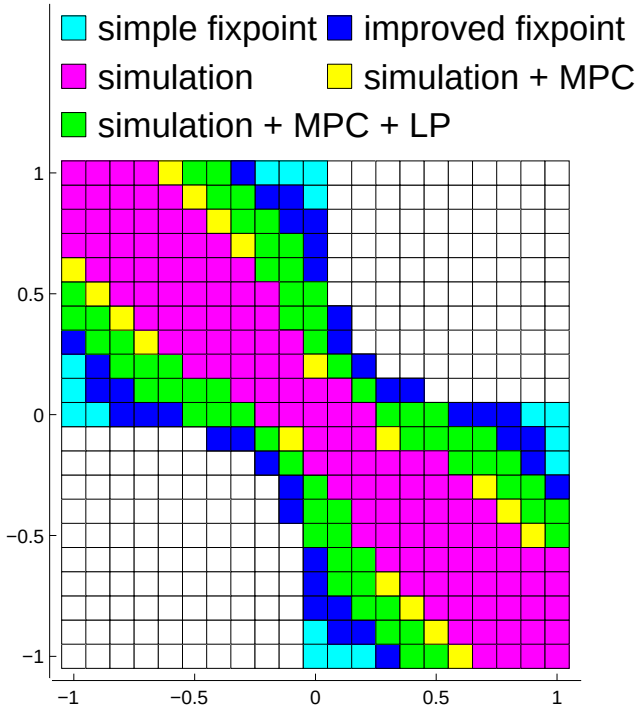
- Use cubic splines to make a smooth input function from a set of time-voltage points.
- Compute partial derivatives of outputs with respect to the voltages at these control points.
- Use greedy search (gradient descent) to find simulation inputs that drive the output to other boxes.



Finding Paths with Linear Programming



- Use “next-box” relation to find a candidate sequence of edge crossings to a target box.
- Each path segment between consecutive edges can be over approximated by a conjunction of linear constraints.
- The solution of the resulting linear program gives an approximate trajectory.
- Use MPC to turn the LP-approximation into a real trajectory.



Observations

- Simulation coverage:
  - We had expected that the over-approximation was too big.
  - We discovered that our simulation cases were too limited.
- The linear program over-approximates the set of possible trajectories
  - Derivative approximated by upper and lower bounds in each box.
  - The over-approximation means that the linear program approach can refute reachability.
  - We discovered that our simulation cases were too limited.
- For this example, MPC readily turned the trajectories generated by the LP solver into real trajectories.

Future work

- Consider better representations of reachable regions, e.g.
  - Ellipsoids
  - Projectagons
  - Zonotopes
- Integrate with SMT solver
  - Exploits LP's ability to refute paths.
  - MPC lets us know when we've found a path.
  - If LP and MPC disagree, need to refine the LP.
- Use MPC to improve simulation coverage

Summary

- Linear-Programming combined with Model-Predictive Control offer a powerful combination for finding and refuting trajectories.
- LP found cases that were missed by human chosen simulations:
  - The computer is more tolerant of the tedium of trying many cases.
  - LP and MPC guide the search.
  - The new trajectories revealed implicit assumptions of the human.
- What's next?
  - Automate the computation of derivatives for MPC.
  - Automatic generation of simulation test cases:

$$\begin{array}{rl} & \text{A specification of allowed inputs} \\ + & \text{A specification of valid outputs} \\ \hline & \text{Automatic generation and checking of simulation cases} \end{array}$$

    If we give designers an incentive to write specifications, they might actually do so. ☺
- Parameterized verification.