Reachability for affine functions on the integers

Daniel Fremont
dfremont@mit.edu

Massachusetts Institute of Technology

September 19, 2012

Outline

- Reachability problems for dynamical systems
- Affine functions on $\mathbb{Z}$ $\quad (z \mapsto az + b)$
  - Several simple cases
  - Handling $z \mapsto z + k$
  - General algorithm
- Extensions and future work

Reachability problems for dynamical systems

- Simple rules can lead to unpredictable behavior
- Multidimensional systems often have undecidable reachability problems
  - States in $\mathbb{Q}^2$ evolving under a finite set of affine functions [Bell & Potapov 2006]
- Transition to decidability at lower dimensions often hard to find
- Affine functions on $\mathbb{Q}$?

Affine functions on $\mathbb{Z}$

- Given $x, y \in \mathbb{Z}$ and a finite set F of affine functions $f_i(z) = a_i z + b_i$ with $a_i, b_i \in \mathbb{Z}$, can you get from x to y by applying functions in F?
- Convenient notation:
  - A function $G = s_K \circ \cdots \circ s_1$ for some $K \in \mathbb{N}$ and with each $s_i \in S$ is called an *S-composition*.
  - $x \xrightarrow{S} y$ *via* G means that G is an S-composition and that $G(x) = y$.
  - $x \xrightarrow{S} y$ means there is some such G.
- After conjugation, our question is whether $x \xrightarrow{F} 0$.

Various cases

- Four different classes of functions
  - Constant: $a_i = 0$ $\quad (z \mapsto k)$
  - Expanding: $|a_i| > 1$ $\quad (z \mapsto 2z + k$, for example$)$
  - Involutory: $a_i = -1$ $\quad (z \mapsto -z + k)$
  - Translating: $a_i = 1$ $\quad (z \mapsto z + k)$
- These have substantially different effects in compositions, so we will treat them separately

Expanding case

- If $|a_i| > 1$, then f_i strictly increases absolute value outside of some finite interval

Algorithm

1. *Take $I = [-R, R]$ large enough such that every f_i strictly increases absolute value on $\mathbb{Z} \setminus I$.*
2. *Then any F-composition maps $\mathbb{Z} \setminus I$ onto itself, so a finite computation can find all orbits of x which remain in I.*
3. *Check if 0 occurs in any of these.*

- Can handle a single $g(z) = -z + k$ by putting $I = [\min(-R, -R + k), \max(R, R + k)]$

Handling $g(z) = z + k$

- Functions like g can be moved through F-compositions:

$$(f_i \circ g)(z) = a_i(z + k) + b_i = (a_i z + b_i) + a_i k = (g^{a_i} \circ f_i)(z)$$

- So if G is any F-composition,

$$G(z) = (g^{\alpha} \circ H)(z) = H(z) + \alpha k$$

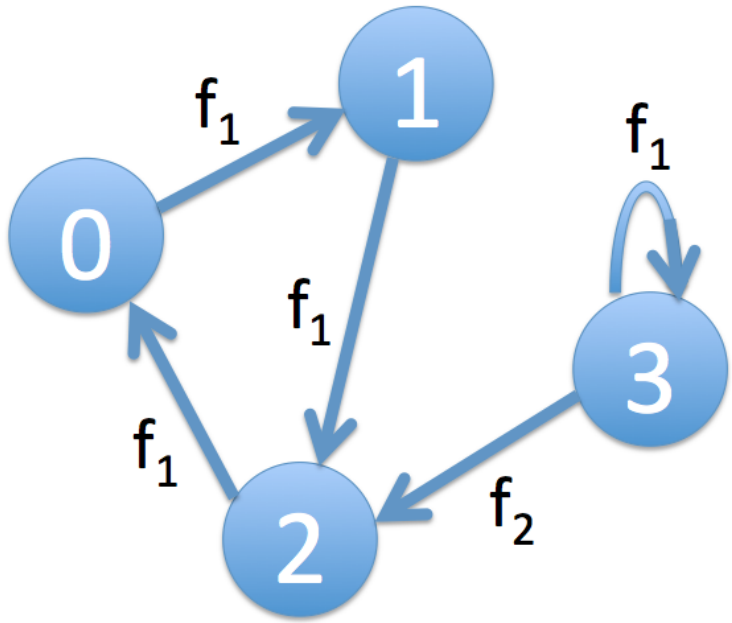
with α a product of the coefficients a_i and where H is G with all instances of g removed

Handling $g(z) = z + k$

- $G(z) = (g^{\alpha} \circ H)(z) = H(z) + \alpha k$
- Putting $S = F \setminus \{g\}$, H is an S-composition.
- Now consider which H satisfy $x \xrightarrow{S} 0 \pmod{k}$ via H.
  1. No H: then $x \xrightarrow{F} 0$ is false.
  2. One where $H(x)$ and k have opposite signs: then $x \xrightarrow{F} 0$.
  3. One where $H = f_{e_0} \circ \cdots \circ f_{e_K}$ with $a_{e_j} < 0$ for some j: we can insert g into H so that $\alpha < 0$, and make $H(x)$ and k have opposite signs; then $x \xrightarrow{F} 0$.
  4. Otherwise, $0 = G(x) = H(x) + \alpha k$ is impossible, and $x \xrightarrow{F} 0$ is false.

Handling $g(z) = z + k$

- Which H satisfy $x \xrightarrow{S} 0 \pmod{k}$ via H?
- Whether there are *any* H is an easy modular problem

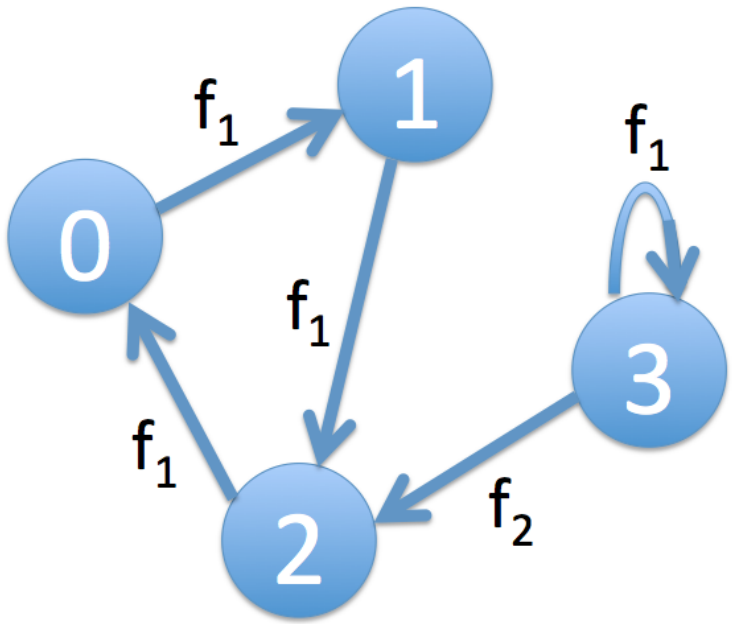


- But what about sign information?

Handling $g(z) = z + k$

Finding the largest $y \equiv 0 \pmod{k}$ such that $x \xrightarrow{S} y$

- Need a good way to describe all S-compositions H such that $H(x) \equiv 0 \pmod{k}$
- Graph $\rightarrow$ finite automaton $\rightarrow$ regular expression
  - An example, starting from $3 \pmod{k}$:



$$\Longrightarrow \quad 1^*21(111)^*$$

Handling $g(z) = z + k$

Finding the largest $y \equiv 0 \pmod{k}$ such that $x \xrightarrow{S} y$

- Let R be the regular expression obtained in this way
- Write $R = S_1 \cup \cdots \cup S_M$ where each S_i has no union operations
- For any sequence of literals ℓ, write P_ℓ for the corresponding S-composition
- We can assume all $a_i > 0$, and thus that any P_ℓ has a positive linear coefficient

Handling $g(z) = z + k$

Finding the largest $y \equiv 0 \pmod{k}$ such that $x \xrightarrow{s} y$

- For some $z \in \mathbb{Z}$ and a subexpression E of R, define $I(z, E)$ to mean that $\exists s \in L(E) : (P_s(z) > z)$
- Consider $E = \ell\alpha^*\beta$ where ℓ is a sequence of literals; then

$$I(z, E) \iff I(P_\ell(z), \alpha) \vee I(z, \ell\beta)$$

  (under our assumption that $a_i > 0$).

- This gives us a recursive algorithm to compute I on any expression

Handling $g(z) = z + k$

Finding the largest $y \equiv 0 \pmod{k}$ such that $x \xrightarrow{S} y$

Algorithm

1. *For each S_i, write $S_i = T_1 \ldots T_K$ where each T_j is a literal or a starred subexpression*
2. *Initialize $s_i = x$. Consider each T_j in order:*
   - *If it is a literal, update s_i to $P_{T_j}(s_i)$*
   - *If it is a starred subexpression, compute $I(s_i, T_j)$. If this is true, return ∞. Otherwise, leave s_i unchanged.*
3. *If we have considered every T_j without returning ∞, then s_i is the largest possible value reachable by S-compositions matching S_i. Return the largest s_i.*

General algorithm

There are several cases.

1. For some j, $a_j = 0$: Recursively determine if $x \xrightarrow{F \setminus \{f_j\}} 0$ or $b_j \xrightarrow{F \setminus \{f_j\}} 0$ and return true iff at least one works.
2. For some j, $a_j = 1$: Use the algorithm just described.
3. For more than one j, $a_j = -1$: Compose two such f_j to get a function of the form $g(z) = z + k$; use the previous case.
4. Otherwise, use the algorithm for the expanding case, with the modification to handle one $a_j = -1$ if necessary.

Extensions and future work

- Affine reachability on $\mathbb{N}$ (decidable)
- Partial affine reachability on $\mathbb{Z}$ or $\mathbb{N}$ (undecidable)
- Affine reachability on $\mathbb{Q}$ (?)
- Polynomial reachability on $\mathbb{Z}$ (?)

`web.mit.edu/~dfremont/www/`