Rewriting in Practice

Ashish Tiwari
SRI International
Menlo Park, CA 94025
tiwari@csl.sri.com

Collaborators: (Systems Biology) Carolyn Talcott, Steven Eker, Peter Karp, Markus Krummenacker, Alexander Shearer, Ingrid Keseler, Merrill Knapp, Patrick Lincoln, Keith Laderoute
(Program analysis) Sumit Gulwani, Guillem Godoy, Manfred Schmidt-Schauß, Adria Gascon

Systems Biology

Enormous amounts of data being generated

- DNA sequencing: Fully sequencing genomes is rapid and easy
- DNA microarray: Which genes are being transcribed
- Proteomics: Which proteins are present
- Flow cytometry: Concentration in individual cells

And how to use it to predict clinical observations and phenotypes?

Systems Biology

Model-based development

Also, a common feature in embedded system design

Goal: Models can help

- perform *in-silico* experiments
- guide *wet lab* experiments
- suggest novel drug targets

Nutrient Sets

Goal: Starting from the genome, find nutrient sets on which that organism will grow

- Sequence genome of the organism
- Extract genes
- Predict metabolic network
- Predict growth on nutrient sets

Metabolic Network: Rewriting-based Modeling

Rewriting is used as a language for writing Petrinets

Petrinets: Ground AC rewrite systems with 1 AC symbol

Example:

$$\begin{aligned} a_1 : \quad & A + B \quad \rightarrow \quad C + D \\ a_2 : \quad & C + A \quad \rightarrow \quad E \end{aligned}$$

The numeric parameters a_1, a_2 capture relative affinity/preference/ likelihood

Typical metabolic networks have 1000's of reactions and metabolites

Rewrite Rules as Models

Rewrite rules used to model

- metabolic networks
- cell signaling
- gene regulatory networks

Terms can have complex structure: compartments, binding sites

Three different semantics of these rules

- stochastic
- deterministic
- nondeterministic

Stochastic Firing: Chemical Master Equation

Strategy for firing rewrite rules: stochastic

Physics-based models of biochemical reaction networks: stochastic Petrinets

Semantics is given using the CME

X: set of metabolites, $|X| = n$; e.g. $X = \{A, B, C, D, E\}$

R: set of reactions

r: a reaction, element of $\mathbb{N}^n$; e.g. $A + C \to E \mapsto [-1, 0, -1, 0, 1]$

P: map from $N^{+^n} \times \mathbb{R}^+ \mapsto [0, 1]$

$$\frac{dP(X,t)}{dt} = \sum_{r \in R} a(P(X - r, t), r)$$

Stochastic Firing: Example

$$a_1: \quad A + B \;\rightarrow\; C + D \qquad\qquad a_2: \quad C + A \;\rightarrow\; E$$

Evolving probability distribution:

	A=2,B=1,C=D=E=0	A=1,B=0,C=1,D=1,E=0	A=0,B=0,C=0,D=1,E=1
1	1	0	0
2	1/2	1/2	0
3	1/4	1/2	1/4
4	1/8	3/8	1/2
5	...	...	...
6	0	0	1

Difficulty: Not enough data to know how to compute a

High-dimensional Markov Chain: Does not scale

Deterministic Firing: Mass Action Dynamics

Approximation of CME using ordinary differential equations

$$a_1 : \quad A + B \;\rightarrow\; C + D \qquad\qquad a_2 : \quad C + A \;\rightarrow\; E$$

ODE model using mass action dynamics:

$$\begin{aligned}
\frac{dA(t)}{dt} &= -a_1 * A(t) * B(t) - a_2 * A(t) * C(t) \\
\frac{dB(t)}{dt} &= -a_1 * A(t) * B(t) \\
\frac{dC(t)}{dt} &= -a_2 * A(t) * C(t) + a_1 * A(t) * B(t) \\
\frac{dD(t)}{dt} &= a_1 * A(t) * B(t) \\
\frac{dE(t)}{dt} &= a_2 * A(t) * C(t)
\end{aligned}$$

Issue: (i) approximate (ii) Still need a_1, a_2

Nondeterministic Firing: Rewriting

Preferable because we do not need extra parameters

Organism grows if it can produce biomass compounds starting from nutrients

This is a reachability question

Petrinet reachability is decidable, but inefficient

Example: If A, B are nutrients, and E is a biomass compound, then:

$$2A + B \;\rightarrow\; A + C + D \;\rightarrow\; E + D$$

Reachability: Via Constraint Solving

We can perform approximate reachability via constraint solving

Example:

$$A + B \;\rightarrow\; C + D \qquad\qquad C + A \;\rightarrow\; E$$

Constraints: Suppose initial state is $2A + B$, we want to reach $D + E$

$$\begin{aligned}
A: &\quad -r_1 - r_2 + 2 = 0 \\
B: &\quad -r_1 + 1 = 0 \\
C: &\quad r_1 - r_2 = 0 \\
D: &\quad r_1 - 1 = 0 \\
E: &\quad r_2 - 1 = 0
\end{aligned}$$

If $D + E$ is reachable from $2A + B$, then above constraints are satisfiable

This is called Flux Balance Analysis

Nutrient Sets for E.Coli

We have used constraint solving for finding (minimal) nutrient sets for E.Coli

Flux Balance Analysis: an overapproximation of the reachability relation

We developed a constraint-based approach that captures reachability more accurately than FBA

Results:
(1) About 75% accuracy with experimental results
(2) Predicted growth of E.Coli on cynate as both Carbon and Nitrogen source, which was experimentally verified
(3) Can compute all minimal nutrient sets for E.Coli

Rewriting in Biology

Apart from metabolic networks, rewrite rules are also commonly used for modeling signalling pathways

Signaling pathway: Biochemical reactions that show how signals are transmitted from the cell surface to the cell cytoplasm to nucleus

Questions of interest to biologists vary

visualization

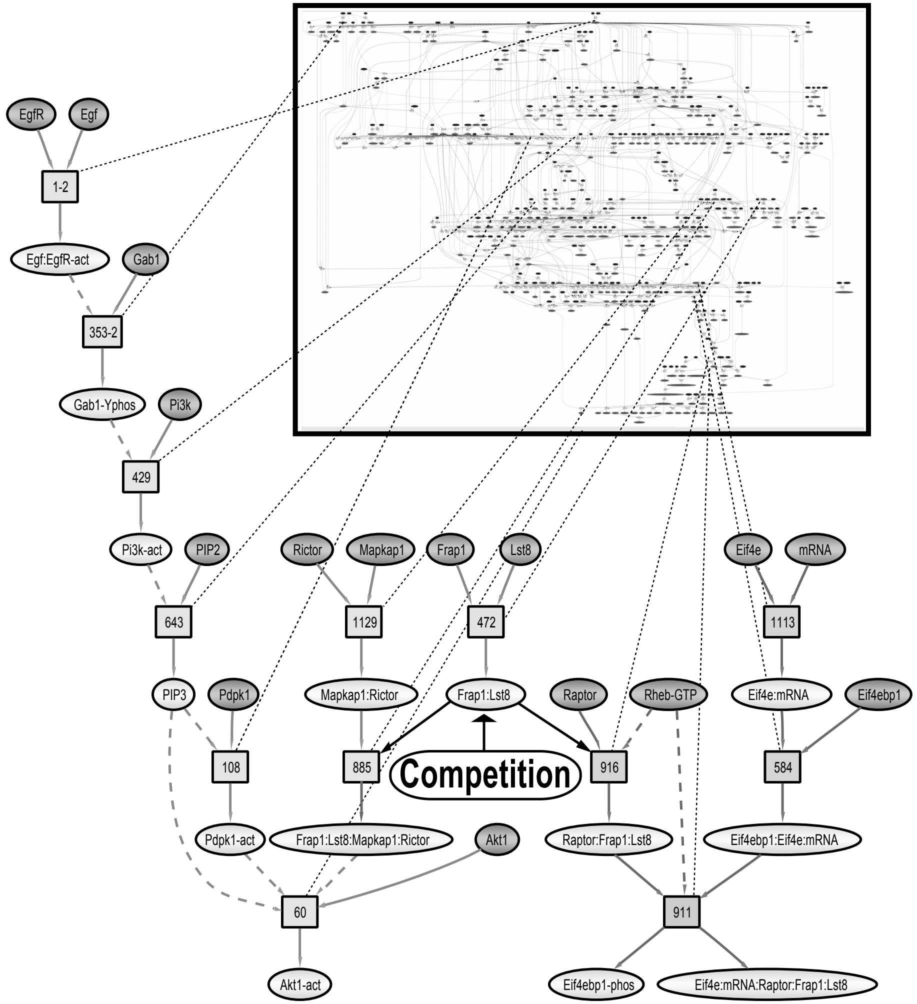
reachability pathways

conflicts: $A \to^* C$ and $B \to^* D$, but $A + B - (A \cap B) \not\to^* C + D$

knockouts: Is it possible $A \to^* C$, but without using B

All analysis techniques should scale

Competing Rules in EGF Stimulation Pathway



Outline

Rewriting in

- Systems Biology
- Algorithm Description and Design
- Theorem Proving

Algorithms

Rewriting is useful in two different ways in the study of algorithms:

- Rewriting-based descriptions for algorithms
- Rewriting as a paradigm for algorithm design

Rewriting-based Descriptions

- Express the algorithmic problem by identifying the term structure of initial and final configuration
- Define an ordering on the space of configurations such that the final configuration is minimal
- Find local transition rules that decrease configuration measure

Rewriting-based Descriptions

Such descriptions are obtained when writing algorithms in rewriting logic (such as, in Maude)

Example: Sorting can be described by

$$X, a, Y, b, Z \quad \rightarrow \quad X, b, Y, a, Z \qquad \text{if } a > b$$

Benefit:

- Separates implementation from the algorithm
- Correctness argument simpler
- Algorithms are nondeterministic

Algorithmic Design Paradigms

Some paradigms taught in a course on algorithms:

- greedy
- divide and conquer
- dynamic programming
- branch and bound

One important paradigm often not taught:

- completion

Completion as Paradigm

for Algorithm Design

- Express the algorithmic problem by identifying configurations as sets of facts
- Define an ordering on the facts and proofs
- Find local transition rules that add or delete facts such that
  - proofs of (provable) facts do not get any bigger
  - some proof gets smaller

In the final configuration, all facts have minimal proofs

Completion-based Procedures: Examples

Shortest-path in a graph:

$$\text{Deduce} \quad \frac{C := \{\ldots, \mathtt{path}(u, v, d_{uv}), \mathtt{path}(v, w, d_{vw}), \ldots\}}{C \cup \{\mathtt{path}(u, w, d_{uv} + d_{vw})\}}$$

$$\text{Delete} \quad \frac{C := \{\ldots, \mathtt{path}(u, v, d), \mathtt{path}(u, v, d'), \ldots\}}{C - \{\mathtt{path}(u, v, d')\}} \mathtt{if} \;\; d < d'$$

Orderings determine what deduction and deletion steps are acceptable

Deleted facts should have smaller proof using remaining facts

Deduced facts should make some proof smaller

Benefits

- Uniform understanding of several algorithms
- Different orderings will yield different algorithms
- Strategy for applying the inference steps can be determined by other factors

Can optimize an algorithm by

- choosing an appropriate ordering
- choosing an appropriate strategy
- choosing an appropriate data structure

Completion-based Algorithms

- Union-find
- Congruence closure
- Rational linear arithmetic (Simplex)
- Fourier-Motzkin
- Gröbner basis
- Ordered resolution

In this talk,

- Linear equalities
- Linear equalities + inequalities
- Nonlinear equalities
- Nonlinear equalities + inequalities

Solving Linear Equations

Facts: $a_1 x_1 + \cdots + a_n x_n + b = 0$

Pick an ordering $x_1 \succ x_2 \succ \cdots \succ x_n$

Define measure $m(a_1 x_1 + \cdots + a_n x_n + b = 0) := \{x_i \mid a_i \neq 0\}$, and $m(x_i > 0) := \{x_i\}$

Order facts by $\succ^m$ on their measures

Measure of a proof := measure of all facts used in it

Deduce $\dfrac{C := \{\ldots, ax + Y = 0, bx + Z = 0, \ldots\}}{C \cup \{bY - aZ = 0\}}$

But, here, we need to do this even when x is not maximal

Solving Linear Arithmetic Equations

Get more flexibility in ordering facts

Distinguish: $a_1x_1 + \cdots + a_nx_n + b = 0$ and $a_1x_1 = -a_2x_2 - \cdots - a_nx_n - b$

Pick an ordering $x_1 \succ x_2 \succ \cdots \succ x_n$

Define measure $m(a_1x_1 + \cdots + a_nx_n + b = 0) := \{x_i \mid a_i \neq 0\}$, and $m(a_1x_1 = -a_2x_2 - \cdots - a_nx_n - b) := \{x_1\}$

Order facts by $\succ^m$ on their measures

Measure of a proof := measure of all facts used in it

Deduce $\dfrac{C := \{\ldots, ax = Y, bx = Z, \ldots\}}{C \cup \{bY - aZ = 0\}}$

Now, we only need to overlap on largest x

Procedure for solving equations (triangular form)

Example: Solving Linear Equations

Example: Ordering $x \succ y$

$$\frac{\dfrac{\dfrac{x + 2y = 0,\ x - y = 0}{x \rightarrow -2y,\ x \rightarrow y}}{x \rightarrow -2y,\ -2y = y}}{x \rightarrow -2y,\ -3y \rightarrow 0}$$

This is a solved/triangular form

Linear Arithmetic Simplex

Consider equality and inequality facts, $x_i > 0$

We are interested in whether the facts together are consistent

How can rewriting help?

First, note that:

$$p_1 = 0,\ p_2 = 0,\ x_1 > 0,\ x_2 > 0 \text{ is unsatisfiable} \quad \texttt{iff}\ \exists p :$$

$$(1) \quad p_1 = 0 \wedge p_2 = 0 \ \Rightarrow\ p = 0$$

$$(2) \quad x_1 > 0 \wedge x_2 > 0 \ \Rightarrow\ p > 0$$

How to determine if such a p exists?

Key idea from rewriting: Make this witness smaller.

Example: Linear Arithmetic Simplex

Example:

Ordering: $x \succ y$

$$\frac{\frac{\frac{x + 2y = 0,\ x - y = 0,\ x > 0}{x \to -2y,\ x \to y,\ x > 0}}{x \to -2y,\ -2y = y,\ x > 0}}{x \to -2y,\ -3y \to 0,\ x > 0}$$

No contradiction detected.

Ordering: $y \succ x$

$$\frac{\frac{\frac{\frac{x + 2y = 0,\ x - y = 0,\ x > 0}{2y \to -x,\ y \to x,\ x > 0}}{2y \to -x,\ -x = 2x,\ x > 0}}{2y \to -x,\ 3x = 0,\ x > 0}}{\perp}$$

Contradiction detected.

$3x = 2(x - y) + (x + 2y)$ is the required witness for unsatisfiability.

Simplex: Changing ordering (aka pivoting) helps us detect unsatisfiability

Nonlinear Equations

Algorithm for computing Gröbner basis is a completion algorithm

Idea behind completion:

- Starting with a set of facts
- Add new facts (saturation)
  - that do not have a smaller proof using existing facts
- Delete any fact (simplification)
  - that do have a smaller proof using other facts

Gröbner Basis: Example

Ordering: Total degree lex with precedence $x \succ y$

View as completion enables optimizations

$$
\begin{array}{c}
xy^2 - x = 0,\ x^2y - y^2 = 0 \\
\hline
xy^2 \to x,\ x^2y \to y^2 \\
\hline
xy^2 \to x,\ x^2y \to y^2[y],\ x^2 = y^3 \\
\hline
xy^2 \to x,\ x^2y \to y^2[y],\ y^3 \to x^2 \\
\hline
xy^2 \to x[y],\ x^2y \to y^2[y],\ y^3 \to x^2,\ xy = x^3 \\
\hline
xy^2 \to x[y],\ x^2y \to y^2[y],\ y^3 \to x^2,\ x^3 \to xy \\
\hline
xy^2 \to x[y, x^2],\ x^2y \to y^2[y, x],\ y^3 \to x^2,\ x^3 \to xy
\end{array}
$$

Property of Gröbner Basis

If

$$\begin{array}{lcl} p' & \in & \mathit{Ideal}(P) \\ G & : & \text{Gröbner basis for } P \end{array}$$

Then

$$\begin{array}{lcll} p' & \leftrightarrow_P^* & 0 & \text{definition of ideal} \\ p' & \rightarrow_G^* & 0 & \text{definition of GB} \end{array}$$

Claim. If there is no $p'' \prec p'$ s.t. $p'' \in \mathit{Ideal}(P)$, then $p' \in G$.
Proof. If $p' \rightarrow_G p'' \rightarrow_G^* 0$, then $p' \succ p''$ and both $p', p'' \in \mathit{Ideal}(P)$.

Nonlinear Simplex

We can generalize the idea of Simplex for linear constraints to nonlinear constraints

Problem: Given a set of nonlinear equations and inequalities:

$$\begin{aligned} p &= 0, & p &\in P \\ q &> 0, & q &\in Q \\ r &\geq 0, & r &\in R \end{aligned}$$

where $P, Q, R \subset \mathbb{Q}[\vec{x}]$ are sets of polynomials over $\vec{x}$

Is the above set unsatisfiable over the reals?

Nonlinear Simplex: Examples

Examples of satisfiable constraints:

$$\{x^2 = 2\}$$
$$\{x^2 = 2, \;\; x < 0, y \geq x\}$$

Examples of unsatisfiable constraints:

$$\{x^2 = -2, \; y \geq x\}$$
$$\{x^2 = 2, \;\; 2x > 3\}$$

Applications in: control, robotics, solving games, static analysis, hybrid systems, ...

Nonlinear Simplex: Known Results

- The full FO theory of reals is decidable [Tarski48]
  Nonelementary decision procedure, impractical
- Double-exponential time decision procedure [Collins74, MonkSolovay74]
- Exponential space lower bound
- Collin's algorithm based on "cylindrical algebraic decomposition" has been improved over the years and implemented in QEPCAD.
  In practice, could fail on $p > 0 \wedge p < 0$.

Obtaining efficient, sound and complete method unlikely

SMT+/SMT-: Can we obtain efficiency by relaxing completeness?

Nonlinear Simplex-

The approach is reminiscent of Simplex

- Introduce slack variables s.t. all inequality constraints are of the form $v > 0$, or $w \geq 0$

$$\begin{array}{llll} P = 0, & Q > 0, & R \geq 0 & \mapsto \\ \underline{P = 0}, & \underline{Q - \vec{v} = 0}, & \underline{R - \vec{w} = 0}, & \vec{v} > 0,\ \vec{w} \geq 0 \end{array}$$

- Search for a polynomial p s.t.

$$\begin{array}{rcl} P = 0 \wedge Q = \vec{v} \wedge R = \vec{w} & \Rightarrow & p = 0 \\ \vec{v} > 0,\ \vec{w} \geq 0 & \Rightarrow & p > 0 \end{array}$$

- If we find such a p, return "unsatisfiable" else return "maybe satisfiable"

How to search for p?

Witness for unsatisfiability p satisfies:

$$P = 0 \wedge Q = \vec{v} \wedge R = \vec{w} \quad \Rightarrow \quad p = 0 \tag{1}$$

$$\vec{v} > 0,\ \vec{w} \geq 0 \quad \Rightarrow \quad p > 0 \tag{2}$$

We need efficient sufficient checks

Sufficient check for Condition **??**: $p \in \mathit{Ideal}(P, Q - \vec{v}, R - \vec{w})$

Sufficient check for Condition **??**: p is a positive polynomial over $\vec{v}, \vec{w}$

To search for p, compute the Gröbner basis for P making $\vec{v}, \vec{w}$ smaller in the ordering

Example: Easy Instance

Consider $E = \{x^3 = x,\ x > 2\}$.

$$\begin{array}{ll}
x^3 - x = 0, & x - v - 2 = 0 \\ \hline
(v+2)^3 - (v+2) = 0, & x - v - 2 = 0 \\ \hline
v^3 + 6v^2 + 11v + 6 = 0, & x - v - 2 = 0 \\ \hline
 & \perp
\end{array}$$

Computing GB and projecting it onto the slack variables discovers the witness p for unsatisfiability

May not work always ...

Example: Harder Instance

Let $I = \{v_1 > 0, v_2 > 0, v_3 > 0\}$.

$$\frac{v_1 + v_2 - 1 = 0, \qquad v_1 v_3 + v_2 - v_3 - 2 = 0}{\dfrac{v_1 + v_2 - 1 = 0, \qquad (1 - v_2)v_3 + v_2 - v_3 - 2 = 0}{\dfrac{v_1 + v_2 - 1 = 0, \qquad v_2 v_3 - v_2 + 2 = 0}{}}}$$

This is a Gröbner basis.

There is an unsatisfiability witness p for this example, but we failed to find it.

Define $v_2 v_3 = v_4$ and make $v_2 \succ v_4$:

$$\frac{v_1 + v_2 - 1 = 0, \qquad -v_2 + v_4 + 2 = 0}{\underline{v_1 + (v_4 + 2) - 1 = 0}, \quad -v_2 + v_4 + 2 = 0}$$

Nonlinear Simplex: Summary

- Turn all inequalities into equations by introducing slack variables
- Compute Gröbner basis of the equations
- If a positive polynomial is ever generated, return unsatisfiable
- If not, introduce new definitions to try different orderings and repeat

Invariant Generation for Dynamical Systems

Problem: Given a continuous dynamical system, find its invariants.

Instance: Given CDS $\frac{dx}{dt} = y, \; \frac{dy}{dt} = -x$, find p s.t. $\frac{dp}{dt} = 0$

Solution: Make $d(p)$ terms smaller than others.

$$\frac{\dfrac{\dfrac{d(x) = y, \; d(y) = -x, \; d(x^2) = x * d(x), \; d(y^2) = y * d(y)}{y \to d(x), \; x \to -d(y), \; x * d(x) \to d(x^2), \; y * d(y) \to d(y^2)}}{\text{...completion...}}}{d(x^2) + d(y^2) = 0}$$

The invariant $x^2 + y^2 = c$ is discovered

Other Applications

There are plenty of other applications of rewriting

- Within SMT solvers: Due to incrementality and backtracking requirements, completion-based decision procedures are preferred
- Program analysis/Logical interpretation: Uninterpreted functions is a common abstraction

$$x := x * y \quad \mapsto \quad x := f(x, y)$$
$$x := x \to \mathtt{next} \quad \mapsto \quad x := \mathtt{next}(x)$$

  Equality assertion checking: equational reasoning/unification (STGs)
  Interprocedural analysis: context unification
- Theorem proving: ordered resolution
- Rewriting

Conclusion

From numeric-centric to symbolic-centric:
Rewriting an important symbolic approach

Future directions:

- Stochastic rewrite systems, SSA, Bayesian networks
- Approximate reachability using constraint solving/ abstractions
- Playing with orderings– more algorithms to be discovered?

Other topics:

- Confluence: Basic concepts?
- Learning: personalized therapeutics
- Dynamical systems