New Topic: Relaxation-based methods

Recall: There are two classes of techniques for determining satisfiability over integers
1. Elimination-based techniques: Omega Test, Cooper’s method
2. Relaxation-based techniques: Branch-and-bound, Gomory cuts, Cuts-from-Proofs

Today’s topic: Relaxation-based methods

Relaxation-Based Techniques Overview
Main idea behind relaxation-based techniques for solving $A\vec{x} \leq \vec{b}$
1. First, ignore integrality requirement
2. Use LP algorithm (e.g., Simplex) to obtain rational solution
3. No rational solution $\Rightarrow$ unsatisfiable
4. Rational solution also integer $\Rightarrow$ satisfiable
5. Add additional constraint to guide search for integer solution
6. Repeat until (3) or (4) holds

Branch and Bound
- B&B is simplest relaxation-based technique
- Suppose Simplex yields fractional solution
- Means there exists some variable $x_i$ assigned to non-integer $f_i$
- B&B constructs two subproblems:
  $A\vec{x} \leq \vec{b} \land x_i \leq \lfloor f_i \rfloor$
  $A\vec{x} \leq \vec{b} \land x_i \geq \lceil f_i \rceil$
- Original problem has a solution iff either subproblem has integer solution. Why?
- Because there is no integer point in range $(\lfloor f_i \rfloor, \lceil f_i \rceil)$, so aren’t missing any integer points

Popular Relaxation-Based Techniques
Existing relaxation-based techniques:
- Branch-and-bound (B&B): simplest relaxation-based technique
- Gomory cuts: uses reasoning performed by Simplex to infer valid inequalities
- Branch-and-cut: combines B&B and Gomory cuts
- Cuts-from-proofs: new technique (Dillig, Dillig, Aiken 2009)

We’ll only talk about B&B and Cuts-from-Proofs

An Example
- Consider the system:
  
  \[
  \begin{align*}
  2x + y & \leq 5 \\
  -x & \leq -1 \\
  -y & \leq 0 
  \end{align*}
  \]
- Suppose Simplex yields $(\frac{5}{2}, 0)$
- B&B constructs subproblem with $x \geq 3$
- This problem has no rational solution
- B&B constructs subproblem with $x \leq 2$
- All vertices integers, thus Simplex yields integer solution $\Rightarrow$ satisfiable
Problem with Branch and Bound

- B&B very simple, but in many cases, it does not terminate!
- Consider the following set of inequalities:

\[-3x + 3y + z \leq -1
3x - 3y + z \leq 2
z \geq 0\]

- No integer points, but infinitely many rational solutions

Limitation of Branch and Bound

- As example illustrates, B&B does not have termination guarantees
- It always terminates if feasible region is bounded
- If feasible region is unbounded and does not contain integer points, B&B does not terminate
- In some settings, such as operations research, finite feasible region assumption may be sensible
- But if we want to use B&B to determine satisfiability, this assumption is unrealistic

Limitation of Branch and Bound, cont

- **Good news**: If there is integer solution to system \(Ax \leq b\), it must lie within a pre-computable finite bound
- **Bad news**: This bound is finite but very large
- If \(A\) is \(m \times n\) with max coefficient \(c\), solution must satisfy:

\[x_i \leq ((m + n) \cdot n \cdot c)^n\]

- For \(5 \times 5\) matrix with max coefficient 5, this bound is 976, 562, 500, 000
- This only gives a termination guarantee “in theory”

Cuts-from-Proofs

- Look at new algorithm called Cuts-from-Proofs
- This algorithm can be viewed as a generalization of B&B
- **Context**: As a side project, started implementing an SMT solver in 2008
- But observed that existing techniques do not work well on integer constraints generated by program verification tool

Cuts-from-Proofs: Motivating Example

- Cuts-from-Proofs generalizes B&B
- Instead of branching around coordinate planes, it computes a custom plane with no integer points

\[-3x + 3y + z = -1\]

- Then, we create two subproblems by branching around this plane
- Neither subproblem has integer solution, so cuts-from-proof immediately terminates

Some Polyhedral Terminology

Before talking about cuts-from-proofs, some new terminology

- An inequality \(\pi \bar{x} \leq c\) called valid inequality if it is satisfied by all points in polytope
- A face \(P\) is the set of points in the polytope satisfying \(\pi \bar{x} = c\) for some valid inequality \(\pi \bar{x} \leq c\)
- A vertex of polytope is a face of dimension 0
- A facet of polytope \(P\) is a face of dimension \(\dim(P) - 1\)
Overview of Cuts-from-Proofs

1. Run Simplex, obtain a point $v$ on polytope
2. Compute defining constraints of this point
   - Intersection of defining constraints form face $F$ of polytope
   - $F$ does not have to be a point; can be line, plane, hyperplane...
3. Determine if this face contains integer solutions
   - can be done in polynomial time using Hermite normal forms
4. If face contains integer points, perform regular B&B
   - since face contains integers, there might be an integer that lies on this face and inside polytope

Defining Constraints of Vertex

▶ Suppose we run Simplex on $A\vec{x} \leq \vec{b}$ and obtain point $v$
▶ $A_i\vec{x} = b_i$ is defining constraint of $v$ if:
   1. $A_i\vec{x} \leq b_i$ is a row of $A\vec{x} \leq \vec{b}$
   2. $v$ satisfies $A_i\vec{x} = b_i$
▶ Observe: Intersection of defining constraints is a face of polytope defined by $A\vec{x} \leq \vec{b}$
▶ Simplex always returns a point, but intersection of defining constraints does not have to be a point
▶ It can be line, plane, or $k$-dimensional hyperplane

Checking Integer Solutions on Face

Next step in algorithm: Decide if intersection of defining constraints contains integer points
▶ To do this, we need matrix representation called Hermite Normal Form (HNF)
▶ An $m \times m$ matrix $H$ is in HNF if:
   1. $H$ is lower-triangular
   2. Diagonal entries are positive ($h_{ii} > 0$)
   3. $h_{ij} \leq 0$ and $|h_{ij}| < h_{ii}$ for $i > j$

Example

▶ Consider again the system:
   
   \[
   -3x + 3y + z \leq -1 \\
   3x - 3y + z \leq 2 \\
   z = 0
   \]
▶ Suppose Simplex yields $(\frac{1}{3}, 0, 0)$
▶ What are the defining constraints?
   $-3x + 3y + z = -1$ and $z = 0$
▶ Intersection of defining constraints is the face shown in red line

Unimodular Matrices

▶ We can convert every matrix $A$ to its Hermite Normal Form representation
▶ For this, we need to define unimodular matrix
▶ A matrix $U$ is unimodular if it has only integer entries and $|\det(U)| = 1$
▶ Example:

\[
\begin{bmatrix}
1 & 3 & -7 \\
0 & -1 & 2 \\
-1 & 0 & 2
\end{bmatrix}
\]
HNF of Matrix

- **Theorem:** For every $m \times n$ matrix $A$ with $\text{rank}(A) = m$, there exists $n \times n$ unimodular matrix $U$ such that:
  \[ AU = [H | 0] \]

- Here, $H$ is in HNF and is unique
- $H$ is called the Hermite Normal Form of $A$
- Furthermore, the HNF of any matrix $A$ can be computed in polynomial time!
- Won’t talk about HNF conversion algorithm in class

Why is HNF Useful?

- **Theorem:** The linear equality system $A\vec{x} = \vec{b}$ has solutions if and only if $H^{-1}\vec{b}$ has integer entries (where $H$ is HNF($A$))
- Use theorem to determine if intersection of defining constraints has integer solution
- Let $A'\vec{x} = \vec{b}'$ be defining constraints
- Compute the Hermite Normal Form $H$ of $A'$ and check if $H^{-1}\vec{b}'$ has only integers

Example

- For the point $(\frac{1}{3}, 0, 0)$, defining constraints were $z = 0$ and $-3x + 3y + z = -1$
- Written as a matrix $A'\vec{x} = \vec{b}'$:
  \[
  \begin{bmatrix}
  0 & 0 & 1 \\
  -3 & 3 & 1
  \end{bmatrix}
  \begin{bmatrix}
  x \\
  y
  \end{bmatrix}
  =
  \begin{bmatrix}
  0 \\
  -1
  \end{bmatrix}
  \]
- Hermite normal form $H$ of $A'$:
  \[
  \begin{bmatrix}
  1 & 0 & 0 \\
  -2 & 3 & 0
  \end{bmatrix}
  \]
- $H^{-1}\vec{b}'$ is \[
  \begin{bmatrix}
  0 \\
  \frac{1}{3}
  \end{bmatrix}
  \]
- Does intersection of defining constraints have integer solution? No

When Face Doesn’t Contain Integers

- If we determine intersection of defining constraints has no integers, want to exclude this entire face from search space.
- Safe because there are no integer points on this face, we won’t be missing any integer points by excluding this face
- To exclude this face from search space, we compute a proof of unsatisfiability for defining constraints

Proof of Unsatisfiability

- A proof of unsatisfiability of $A'\vec{x} = \vec{b}'$ is a single linear equality $E$ such that:
  1. $E$ is implied by $A'\vec{x} = \vec{b}'$
  2. $E$ does not have integer solutions
- In what way does $E$ prove $A'\vec{x} = \vec{b}'$ is unsatisfiable?
- From condition (1), we have $A'\vec{x} = \vec{b}' \Rightarrow E$
- Contrapositive of this is: $\neg E \Rightarrow A'\vec{x} \neq \vec{b}'$
- Since $E$ is unsat, $\neg E$ is true; thus, $A'\vec{x} \neq \vec{b}'$ always true
- Hence, $E$ establishes unsatisfiability of $A'\vec{x} = \vec{b}'$

Computing Proof of Unsatisfiability

- We can compute proofs of unsatisfiability using HNF
- Let $A'\vec{x} = \vec{b}'$ be a system with no integer solutions.
- Compute Hermite Normal Form $H$ of $A'$ and $H^{-1}$
- Now, consider $H^{-1}A'\vec{x} = H^{-1}\vec{b}'$:
  \[
  \begin{bmatrix}
  a_1 & \ldots & a_n \\
  \vdots & \ddots & \vdots \\
  a_{m-1} & \ldots & a_{m+n-1}
  \end{bmatrix}
  \begin{bmatrix}
  \vec{z}
  \end{bmatrix}
  =
  \begin{bmatrix}
  c_1 \\
  \vdots \\
  c_{m+n}
  \end{bmatrix}
  \]
- Proof of unsatisfiability of $A'\vec{x} = \vec{b}'$:
  \[
  a_1 dx_1 + a_2 dx_2 + \ldots + a_{n+m-1} dx_{n+m-1} = n_i
  \]
Example

- Defining constraints for \( \left( \frac{1}{3}, 0, 0 \right) \):
  \[
  \begin{bmatrix}
    0 & 0 & 1 \\
    -3 & 3 & 1
  \end{bmatrix}
  \begin{bmatrix}
    x \\
    y \\
    3x-3y=1
  \end{bmatrix} = \begin{bmatrix}
    0 \\
    -1
  \end{bmatrix}
  \]

- Multiply both sides by \( H^{-1} \):
  \[
  \begin{bmatrix}
    1 & 3 & 0 \\
    2 & 1 & 3 \\
    -1 & 3 & 1
  \end{bmatrix}
  \begin{bmatrix}
    x \\
    y \\
    3x-3y=1
  \end{bmatrix} = \begin{bmatrix}
    1 & 3 & 0 \\
    2 & 1 & 3 \\
    -1 & 3 & 1
  \end{bmatrix}
  \begin{bmatrix}
    0 \\
    0 \\
    -1
  \end{bmatrix}
  \]

Branching around the Proof

- Consider proof of unsatisfiability of the system \( A'x = b' \)
  \[
  a_1x_1 + a_2x_2 + \ldots + a_nx_n = c
  \]

- To "branch around" this plane, need to compute the closest parallel planes \( \parallel \) to the proof plane of unsatisfiability containing integer points

- Let \( g \) be the GCD of \( a_1, a_2, \ldots, a_n \)

- The closest parallel planes containing integers given by:
  \[
  \frac{a_1}{g} x_1 + \frac{a_2}{g} x_2 + \ldots + \frac{a_n}{g} x_n = \left\lfloor \frac{c}{g} \right\rfloor
  \]
  \[
  \frac{a_1}{g} x_1 + \frac{a_2}{g} x_2 + \ldots + \frac{a_n}{g} x_n = \left\lceil \frac{c}{g} \right\rceil
  \]

Example

- Defining constraints for \( \left( \frac{1}{3}, 0, 0 \right) \):
  \[
  \begin{bmatrix}
    0 & 0 & 1 \\
    -3 & 3 & 1
  \end{bmatrix}
  \begin{bmatrix}
    x \\
    y \\
    3x-3y=1
  \end{bmatrix} = \begin{bmatrix}
    0 \\
    -1
  \end{bmatrix}
  \]

- Earlier, we computed proof of unsatisfiability as:
  \[
  -3x + 3y + 3z = -1
  \]

- What are the closest parallel planes containing integers?
  \[
  -x + y + z = -1
  \]
  \[
  -x + y + z = 0
  \]

Example, cont

- Result of multiplication:
  \[
  \begin{bmatrix}
    0 & 0 & 1 \\
    -1 & 1 & 1
  \end{bmatrix}
  \begin{bmatrix}
    x \\
    y \\
    3x-3y=1
  \end{bmatrix} = \begin{bmatrix}
    -0 \\
    -1
  \end{bmatrix}
  \]

- What is proof of unsatisfiability?
  \[
  -3x + 3y + 3z = -1
  \]

- This equality has no integer solutions because the GCD of coefficients does not evenly divide constant on RHS.

- Furthermore, it is implied by the defining constraints \(-3x + 3y + z = -1\) and \(z = 0\)

Branching Around the Proof, cont.

- Thus, if solution is to the "left" of proof plane, must satisfy:
  \[
  \frac{a_1}{g} x_1 + \frac{a_2}{g} x_2 + \ldots + \frac{a_n}{g} x_n \leq \left\lfloor \frac{c}{g} \right\rfloor \quad (1)
  \]

- And if it is to the "right", it must satisfy:
  \[
  \frac{a_1}{g} x_1 + \frac{a_2}{g} x_2 + \ldots + \frac{a_n}{g} x_n \geq \left\lceil \frac{c}{g} \right\rceil \quad (2)
  \]

- Thus, we construct two subproblems one where we conjoin \( A'x \leq b' \) with (1), and another where we conjoin it with (2)

Example, cont

- Thus, we solve two subproblems, one with additional constraint:
  \[
  -x + y + z \leq -1
  \]

- The other subproblem has additional constraint:
  \[
  -x + y + z \geq 0
  \]

- Geometrically, this corresponds to solving these two problems:
  
  - Neither subproblem has solution
  
  - Algorithm terminates after one step
What Have We Achieved?

- **Guarantee:** By branching around proof of unsatisfiability, Simplex will never yield point on this face again!
- **Proof:** Suppose defining constraints in current iteration were \( A' \bar{v} = \bar{b}' \) with proof of unsatisfiability \( E' \)
- Suppose we get a vertex \( v \) with same defining constraints next
- **Observation 1:** By definition, \( v \) must satisfy \( A' \bar{v} = \bar{b}' \)
- **Observation 2:** \( v \) cannot satisfy \( E \) because it would violate the new inequalities we added
  - But since \( A' \bar{v} = \bar{b}' \Rightarrow E \), this implies \( v \) can’t satisfy \( A' \bar{v} = \bar{b}' \)
  - Thus, we will never get a solution with same defining constraints again

What About an Unsatisfiable Subset?

- Ok, we excluded intersection of defining constraints from search space
- But what if there is a subset of these defining constraints whose intersection has no integers?
- For instance, suppose we have 3 defining constraints, but the intersection of 2 of them has no integers
- If we are in 3D space, intersection of 3 constraints a point, but intersection of 2 constraints is a whole line
- Thus, we want to exclude the higher-dimensional face with no integers

Another Guarantee

- If we add new inequalities in a certain order, we are also guaranteed to eventually find proof of unsatisfiability for this higher dimensional face!
- Suppose again we have 3 defining constraints, but the intersection of 2 of them has no integers
- If Simplex keeps yields another point on this line, we will obtain a proof of unsatisfiability of this line in the next step

Experiments

- We compared performance of this new algorithm against four other tools solving qff linear inequalities
  1. CVC3: uses Omega Test
  2. Yices: uses branch-and-cut (B&B + Gomory cuts)
  3. Z3: uses branch-and-cut (B&B + Gomory cuts)
  4. MathSAT: uses Omega Test + branch-and-cut

Experiments

Number of variables vs. average running time.

Number of variables vs. percent of successful runs with a 1200 second time-out.
Experiments

Maximum coefficient vs. average running time for 10 × 20 system

Cuts-from-Proofs Summary

- New relaxation-based technique for solving qff linear integer inequalities
- Computes custom planes to branch around for each problem
- These custom planes are generated by computing proof of unsatisfiability for defining constraints
- At least for some problems, this algorithm does significantly better than existing ones