Satisfiability Checking

16 Branch and bound

Prof. Dr. Erika Ábrahám

RWTH Aachen University
Informatik 2
LuFG Theory of Hybrid Systems

WS 22/23

Definition

An integer linear system S is a linear system in general form $Ax = 0$ with $x = x_1, \ldots, x_{n+m}$, $\bigwedge_{i=1}^{m} l_i \leq x_i \leq u_i$ for $i = n+1, \ldots, m$ and with the additional integrality requirement that all variables are of type integer.

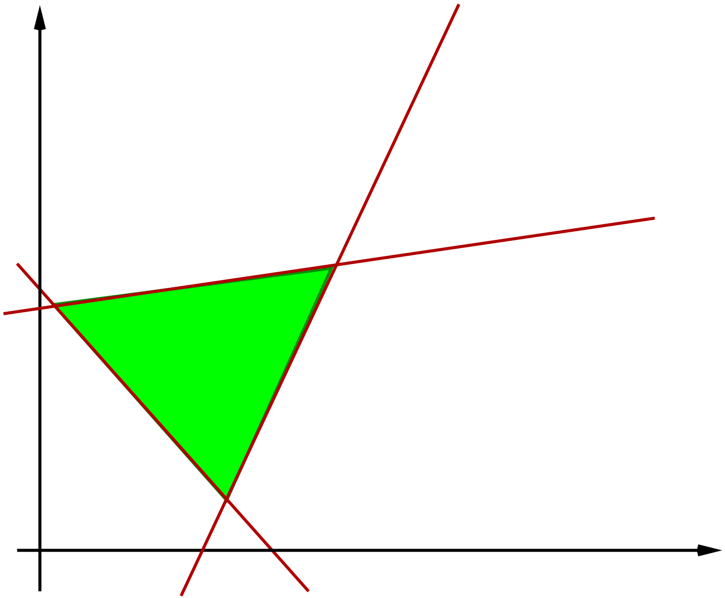
Definition (relaxed system)

Given an integer linear system S, its relaxation relaxed(S) is S without the integrality requirement.

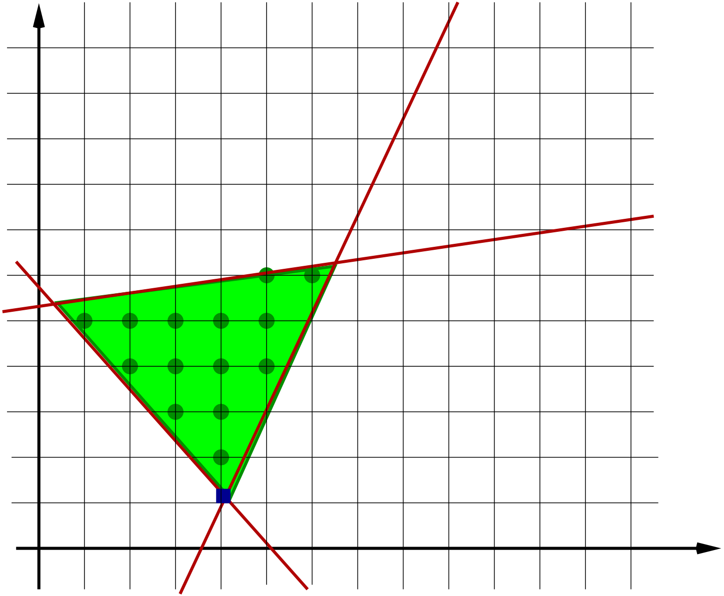
S satisfiable $\Rightarrow$ relax(S) satisfiable

S unsatisfiable $\Leftarrow$ relax(S) unsatisfiable

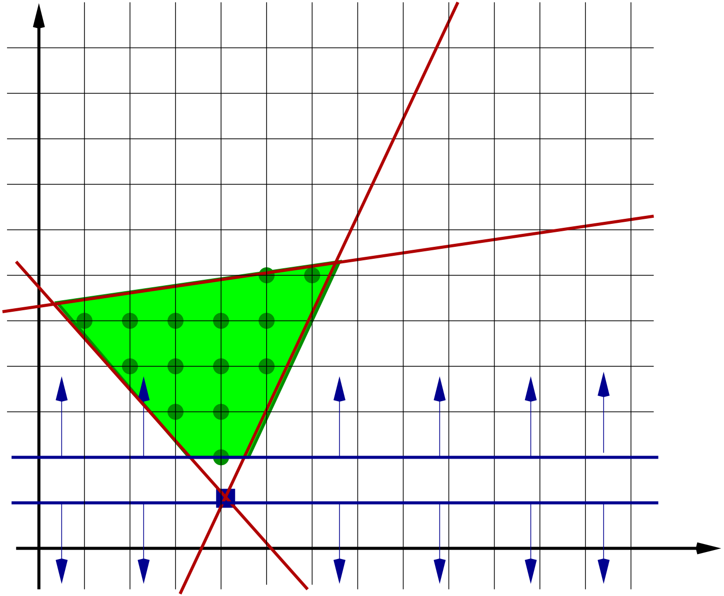
Geometric interpretation



Geometric interpretation

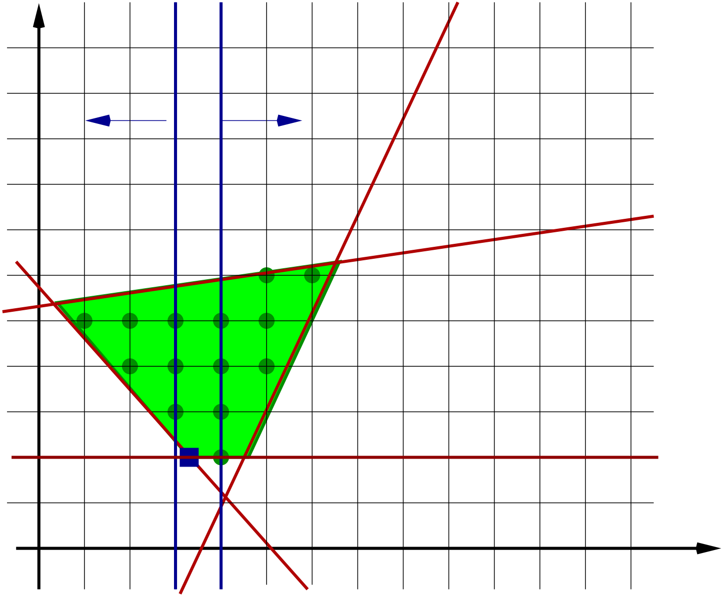


Geometric interpretation

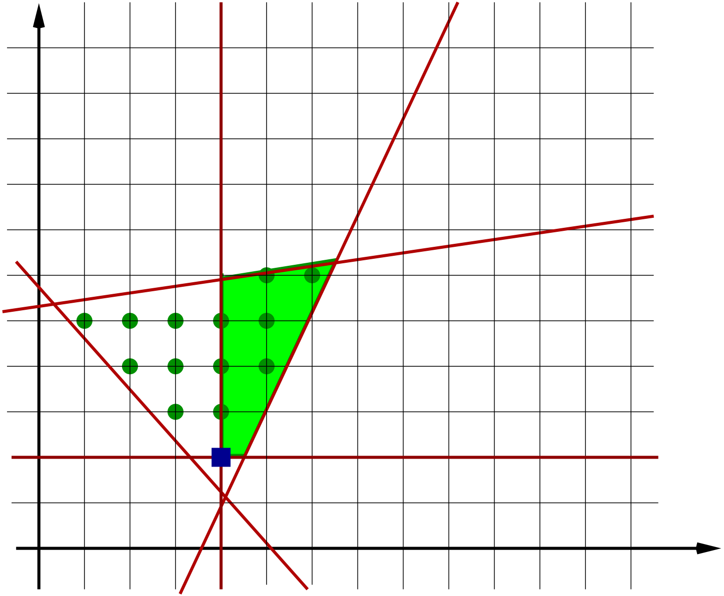


Branch:
分成2部分
⩽ 向下取整
⩾ 向上取整

Geometric interpretation



Geometric interpretation

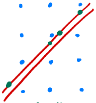


Branch and bound algorithm

Input: An integer linear system S
Output: SAT if S is satisfiable, UNSAT otherwise

procedure Branch-and-Bound(S) {
  linear programming
  res = LP(relaxed(S));
  if (res==UNSAT) return UNSAT;
  else if (res is integral) return SAT;
  else {
    Select a variable v that is assigned a non-integral value r;
    round down
    if (Branch-and-Bound($S \cup \{v \leq \lfloor r \rfloor\}$)==SAT) return SAT;
    else if (Branch-and-Bound($S \cup \{v \geq \lceil r \rceil\}$)==SAT) return SAT;
    round up
    else return UNSAT;
  }
}

Branch and bound



很多点，每个都branch-bound ⇒ 无法结束

- The algorithm is incomplete.
- Example: $1 \leq 3x - 3y \leq 2$ has unbounded real solutions but no integer solutions $\rightarrow$ the algorithm loops forever.
- The algorithm can be made complete for formulae with the small-model property: if there is a solution, then there is also a solution within a (computable) finite bound.
- The algorithm can be extended to mixed integer linear programming, where some of the variables are integer-valued while the others are real-valued.

Branch and bound

- Branch: Split the search space
- Bound: Exclude unsatisfiable sub-spaces

bound: 形成界限limit, 即把不符合要求的区域排除
(整数之间的区域)

- We have seen: Depth-first search
- Also possible: Breadth-first search

Branch and bound

Optimizations:

- Constraints can be removed: $x_1 + x_2 \leq 2$, $x_1 \leq 1$, $x_2 \leq 1$.
  First constraint is redundant.
- Bounds can be tightened: $2x_1 + x_2 \leq 2$, $x_2 \geq 4$, $x_1 \leq 3$
  From the first two constraints we get $x_1 \leq -1$

  General case: Assume a constraint $\sum_i a_i x_i \leq b$ with $l_i \leq x_i \leq u_i$.

  $\rightsquigarrow a_k x_k \leq b - \sum_{i \neq k} a_i x_i$

  如果ai > 0, lower bound　如果ai < 0, upper bound

  If $a_k > 0$, we have $x_k \leq \frac{b - (\sum_{i \neq k, a_i > 0} a_i l_i) - (\sum_{i \neq k, a_i < 0} a_i u_i)}{a_k}$.

  If $a_k < 0$, we have $x_k \geq \frac{b - (\sum_{i \neq k, a_i > 0} a_i u_i) - (\sum_{i \neq k, a_i < 0} a_i l_i)}{a_k}$.

  After the next summary, we will learn interval constraint propagation, which generalizes this idea to polynomial constraints.

Learning target

- How can we extend the simplex method to branch-and-bound in order to find integer solutions?
- Is branch-and-bound complete?
- Which preprocessing steps are possible to simplify the input integer linear system?