Script generated by TTT

Title: Seidl: Programmoptimierung (14.01.2016)

Date: Thu Jan 14 08:37:02 CET 2016

Duration: 87:37 min

Pages: 53

Extension 3

Sometimes, one loop alone does not provide enough opportunities for parallelization.

... but perhaps two successively in a row ...

Example

$$\begin{array}{ll} \text{for } (x = 0; x < n; x{+}{+})\ \{ & \text{for } (x = 0; x < n; x{+}{+})\ \{ \\ \quad R = B[x]; & \quad R = B[x]; \\ \quad S = C[x]; & \quad S = C[x]; \\ \quad T_1 = R + S; & \quad T_2 = R - S; \\ \quad A[x] = T_1; & \quad C[x] = T_2; \\ \} & \} \end{array}$$

Extension 3

Sometimes, one loop alone does not provide enough opportunities for parallelization.

... but perhaps two successively in a row ...

Example

$$\begin{array}{ll} \text{for } (x = 0; x < n; x{+}{+})\ \{ & \text{for } (x = 0; x < n; x{+}{+})\ \{ \\ \quad R = B[x]; & \quad R = B[x]; \\ \quad S = C[x]; & \quad S = C[x]; \\ \quad T_1 = R + S; & \quad T_2 = R - S; \\ \quad A[x] = T_1; & \quad C[x] = T_2; \\ \} & \} \end{array}$$

Extension 3

Sometimes, one loop alone does not provide enough opportunities for parallelization.

... but perhaps two successively in a row ...

Example

$$\begin{array}{ll} \text{for } (x = 0; x < n; x{+}{+})\ \{ & \text{for } (x = 0; x < n; x{+}{+})\ \{ \\ \quad R = B[x]; & \quad R = B[x]; \\ \quad S = C[x]; & \quad S = C[x]; \\ \quad T_1 = R + S; & \quad T_2 = R - S; \\ \quad A[x] = T_1; & \quad C[x] = T_2; \\ \} & \} \end{array}$$

In order to fuse two loops into one, we require that:

- the iteration schemes coincide;
- the two loops access different data.

In case of individual variables, this can easily be verified.

This is more difficult in presence of arrays.

Taking the source program into account, accesses to distinct statically allocated arrays can be identified.

An analysis of accesses to the same array is significantly more difficult ...

Assume that the blocks A, B, C are distinct.

Then we can combine the two loops into:

for $(x = 0; x < n; x{+}{+})$ $\{$

$R = B[x];$	$R = B[x];$
$S = C[x];$	$S = C[x];$
$T_1 = R + S;$	$T_2 = R - S;$
$A[x] = T_1;$	$C[x] = T_2;$
	$\}$

The first loop may in iteration x not read data which the second loop writes to in iterations $< x$.

The second loop may in iteration x not read data which the first loop writes to in iterations $> x$.

If the index expressions of jointly accessed arrays are linear, the given constraints can be verified through integer linear programming ...

$$\begin{array}{rcl} i & \geq & 0 \\ i & \leq & x-1 \end{array} \qquad \begin{array}{rcl} x_{\text{write}} & = & i \\ x_{\text{read}} & = & x \\ x_{\text{read}} & = & x_{\text{write}} \end{array}$$

// x_{read} read access to C by 1st loop
// x_{write} write access to C by 2nd loop

... obviously has no solution.

Assume that the blocks A, B, C are distinct.

Then we can combine the two loops into:

for $(x = 0; x < n; x{+}{+})$ $\{$

$R = B[x];$	$R = B[x];$
$S = C[x];$	$S = C[x];$
$T_1 = R + S;$	$T_2 = R - S;$
$A[x] = T_1;$	$C[x] = T_2;$
	$\}$

The first loop may in iteration x not read data which the second loop writes to in iterations $< x$.

The second loop may in iteration x not read data which the first loop writes to in iterations $> x$.

If the index expressions of jointly accessed arrays are linear, the given constraints can be verified through integer linear programming ...

$i := x$

$$\begin{array}{rcl} i & \geq & 0 \\ i & \leq & x-1 \end{array} \qquad \begin{array}{rcl} x_{\text{write}} & = & i \\ x_{\text{read}} & = & x \\ x_{\text{read}} & = & x_{\text{write}} \end{array}$$

// x_{read} read access to C by 1st loop
// x_{write} write access to C by 2nd loop

... obviously has no solution.

General Form:

$$\begin{array}{rcl} s & \geq & t_1 \\ t_2 & \geq & s \\ y_1 & = & s_1 \\ y_2 & = & s_2 \\ y_1 & = & y_2 \end{array}$$

for linear expressions s, t_1, t_2, s_1, s_2 over i and the iteration variables.

This can be simplified to:

$$\boxed{0 \leq s - t_1 \qquad 0 \leq t_2 - s \qquad 0 = s_1 - s_2}$$

What should we do with it ???

Simple Case:

The two inequations have no solution over $\mathbb{Q}$.

Then they also have no solution over $\mathbb{Z}$.

... in Our Example:

$$\begin{array}{rcll} x & = & i & \\ 0 & \leq & i & = x \\ 0 & \leq & x - 1 - i & = -1 \end{array}$$

The second inequation has no solution.

One Variable:

The inequations where x occurs positive, provide lower bounds.

The inequations where x occurs negative, provide upper bounds.

If G, L are the greatest lower and the least upper bound, respectively, then all (integer) solution are in the interval $[G, L]$.

Example

$$\begin{array}{rcl} 0 & \leq & 13 - 7 \cdot x \\ 0 & \leq & -1 + 5 \cdot x \end{array} \iff \begin{array}{rcl} x & \leq & \frac{13}{7} \\ x & \geq & \frac{1}{5} \end{array}$$

$\left[\frac{1}{5}, \frac{13}{7}\right]$

The only integer solution of the system is $x = 1$.

Discussion

- Solutions only matter within the bounds to the iteration variables.
- Every integer solution there provides a conflict.
- Fusion of loops is possible if no conflicts occur.
- The given special case suffices to solve the case one variable over $\mathbb{Z}$.
- The number of variables in the inequations corresponds to the nesting-depth of for-loops $\Longrightarrow$ in general, is quite small.

Discussion

ISAT

- Integer Linear Programming (ILP) can decide satisfiability of a finite set of equations/inequations over $\mathbb{Z}$ of the form:

$$\sum_{i=1}^{n} a_i \cdot x_i = b \quad \text{bzw.} \quad \sum_{i=1}^{n} a_i \cdot x_i \geq b\,, \qquad a_i \in \mathbb{Z}$$

- Moreover, a (linear) cost function can be optimized.
- Warning: The decision problem is in general, already NP-hard !!!
- Notwithstanding that, surprisingly efficient implementations exist.
- Not just loop fusion, but also other re-organizations of loops yield ILP problems ...

$$\sum_{i=1}^{n} a_i x_i \leq b$$

$$x_i \geq 0$$

$$x_i \leq 1$$

Discussion

- Integer Linear Programming (ILP) can decide satisfiability of a finite set of equations/inequations over $\mathbb{Z}$ of the form:

$$\sum_{i=1}^{n} a_i \cdot x_i = b \quad \text{bzw.} \quad \sum_{i=1}^{n} a_i \cdot x_i \geq b\,, \qquad a_i \in \mathbb{Z}$$

- Moreover, a (linear) cost function can be optimized.
- Warning: The decision problem is in general, already NP-hard !!!
- Notwithstanding that, surprisingly efficient implementations exist.
- Not just loop fusion, but also other re-organizations of loops yield ILP problems ...

Background 5: Presburger Arithmetic

Many problems in computer science can be formulated without multiplication.

Let us first consider two simple special cases ...

1. Linear Equations

$$\begin{array}{rcrcrcr} 2x & + & 3y & & & = & 24 \\ x & - & y & + & 5z & = & 3 \end{array}$$

Question

- $\boxed{\text{Is there a solution over } \mathbb{Q} \text{ ?}}$
- $\boxed{\text{Is there a solution over } \mathbb{Z} \text{ ?}}$
- Is there a solution over $\mathbb{N}$?

Let us reconsider the equations:

$$\begin{array}{rcrcrcr} 2x & + & 3y & & & = & 24 \\ x & - & y & + & 5z & = & 3 \end{array}$$

Answers

- Is there a solution over $\mathbb{Q}$? Yes
- Is there a solution over $\mathbb{Z}$? No
- Is there a solution over $\mathbb{N}$? No

Complexity

- Is there a solution over $\mathbb{Q}$? Polynomial
- Is there a solution over $\mathbb{Z}$? Polynomial
- Is there a solution over $\mathbb{N}$? NP-hard

Solution Method for Integers

Observation 1

$$a_1x_1 + \ldots + a_kx_k = b \qquad (\forall i: \ a_i \neq 0)$$

has a solution iff

$$\mathrm{gcd}\{a_1, \ldots, a_k\} \quad | \quad b$$

Example

$$5y - 10z = 18$$

has no solution over $\mathbb{Z}$.

Example

$$5y - 10z = 18$$

has no solution over $\mathbb{Z}$.

Observation 2

Adding a multiple of one equation to another does not change the set of solutions.

Example

$$\begin{array}{rcrcrcr} 2x & + & 3y & & & = & 24 \\ x & - & y & + & 5z & = & 3 \end{array}$$

Example

$$\begin{array}{rcrcrcr} 2x & + & 3y & & & = & 24 \\ x & - & y & + & 5z & = & 3 \end{array}$$

$$\Longrightarrow$$

$$\begin{array}{rcrcrcr} & & 5y & - & 10z & = & 18 \\ x & - & y & + & 5z & = & 3 \end{array}$$

Observation 3

Adding multiples of columns to another column is an invertible transformation which we keep track of in a separate matrix ...

$$
\begin{array}{ccc|rcrcrcr}
1 & 0 & 0 & & & 5y & - & 10z & = & 18 \\
0 & 1 & 0 & x & - & y & + & 5z & = & 3 \\
0 & 0 & 1 & & & & & & &
\end{array}
$$

$$\Longrightarrow$$

$$
\begin{array}{ccc|rcrcrcr}
1 & 0 & 0 & & & 5y & & & = & 18 \\
0 & 1 & 2 & x & - & y & + & 3z & = & 3 \\
0 & 0 & 1 & & & & & & &
\end{array}
$$

Observation 3

Adding multiples of columns to another column is an invertible transformation which we keep track of in a separate matrix ...

$$
\begin{array}{ccr|rcrcrcr}
1 & 0 & 0 & & & 5y & & & = & 18 \\
0 & 1 & 2 & x & - & y & + & 3z & = & 3 \\
0 & 0 & 1 & & & & & & &
\end{array}
$$

$$\Longrightarrow$$

$$
\begin{array}{ccr|rcrcr}
1 & 0 & -3 & & & 5y & = & 18 \\
0 & 1 & 2 & x & - & y & = & 3 \\
0 & 0 & 1 & & & & &
\end{array}
$$

$\Longrightarrow$ triangular form !!

Observation 4

- A special solution of a triangular system can be directly read off.
- All solutions of a homogeneous triangular system can be directly read off.
- All solutions of the original system can be recovered from the solutions of the triangular system by means of the accumulated transformation matrix.

Observation 4

- A special solution of a triangular system can be directly read off.
- All solutions of a homogeneous triangular system can be directly read off.
- All solutions of the original system can be recovered from the solutions of the triangular system by means of the accumulated transformation matrix.

Example

$$\begin{array}{ccc|ccccc} 1 & 0 & -3 & & & 5y & = & 15 \\ 0 & 1 & 2 & x & - & y & = & 3 \\ 0 & 0 & 1 & & & & & \end{array}$$

One special solution:

$$[6, 3, 0]^\top$$

All solutions of the homogeneous system are spanned by:

$$[0, 0, 1]^\top$$

Example

$$\begin{array}{ccc|ccccc} 1 & 0 & -3 & & & 5y & = & 15 \\ 0 & 1 & 2 & x & - & y & = & 3 \\ 0 & 0 & 1 & & & & & \end{array}$$

One special solution:

$$[6, 3, 0]^\top$$

All solutions of the homogeneous system are spanned by:

$$[0, 0, 1]^\top$$

695

Solving over $\mathbb{N}$

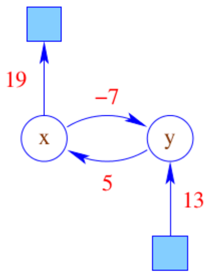
- ... is of major practical importance;
- ... has led to the development of many new techniques;
- ... easily allows to encode NP-hard problems;
- ... remains difficult if just three variables are allowed per equation.

Solving over $\mathbb{N}$

- ... is of major practical importance;
- ... has led to the development of many new techniques;
- ... easily allows to encode NP-hard problems;
- ... remains difficult if just three variables are allowed per equation.

696

Idea: Represent the system by a graph:

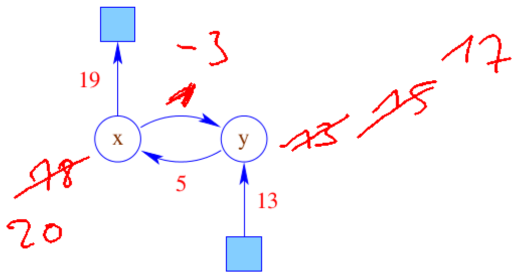


2. One Polynomial Special Case

$$
\begin{array}{rcl}
x & \geq & y+5 \\
19 \geq & x & \\
y & \geq & 13 \\
y & \geq & x-7
\end{array}
$$

- There are at most 2 variables per in-equation;
- no scaling factors.

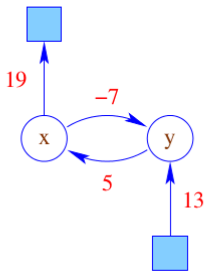
Idea: Represent the system by a graph:

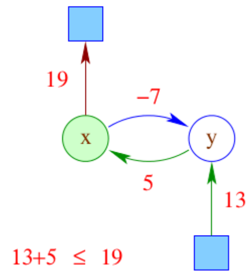


The in-equations are satisfiable iff

- the weight of every cycle are at most 0;
- the weights of paths reaching x are bounded by the weights of edges from x into the sink.

Idea: Represent the system by a graph:





3. A General Solution Method

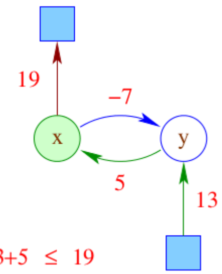
Idea: Fourier-Motzkin Elimination

- Successively remove individual variables x !
- All in-equations with positive occurrences of x yield lower bounds.
- All in-equations with negative occurrences of x yield upper bounds.
- All lower bounds must be at most as big as all upper bounds.

3. A General Solution Method

Idea: Fourier-Motzkin Elimination

- Successively remove individual variables x !
- All in-equations with positive occurrences of x yield lower bounds.
- All in-equations with negative occurrences of x yield upper bounds.
- All lower bounds must be at most as big as all upper bounds.



$13+5 \leq 19$

	0	1	2	3
x	$-\infty$	$-\infty$	18	18
y	$-\infty$	13	13	13

The in-equations are satisfiable iff

- the weight of every cycle are at most 0;
- the weights of paths reaching x are bounded by the weights of edges from x into the sink.

$\Longrightarrow$

Compute the reflexive and transitive closure of the edge weights!

or use Bellman-Ford!

Jean Baptiste Joseph Fourier, 1768–1830

Jean Baptiste Joseph Fourier, 1768–1830

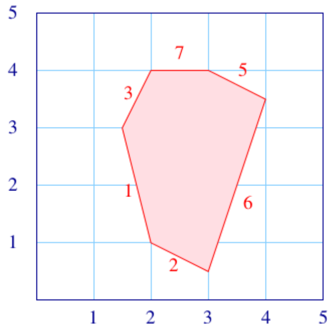
3. A General Solution Method

Idea: Fourier-Motzkin Elimination

- Successively remove individual variables x !
- All in-equations with positive occurrences of x yield lower bounds.
- All in-equations with negative occurrences of x yield upper bounds.
- All lower bounds must be at most as big as all upper bounds.

Example

$$
\begin{array}{rcll}
9 & \leq & 4x_1 + x_2 & (1) \\
4 & \leq & x_1 + 2x_2 & (2) \\
0 & \leq & 2x_1 - x_2 & (3) \\
6 & \leq & x_1 + 6x_2 & (4) \\
-11 & \leq & -x_1 - 2x_2 & (5) \\
-17 & \leq & -6x_1 + 2x_2 & (6) \\
-4 & \leq & -x_2 & (7)
\end{array}
$$



For x_1 we obtain:

$$
\begin{array}{rcll}
9 & \leq & 4x_1 + x_2 & (1) \\
4 & \leq & x_1 + 2x_2 & (2) \\
0 & \leq & 2x_1 - x_2 & (3) \\
6 & \leq & x_1 + 6x_2 & (4) \\
-11 & \leq & -x_1 - 2x_2 & (5) \\
-17 & \leq & -6x_1 + 2x_2 & (6) \\
-4 & \leq & -x_2 & (7)
\end{array}
\qquad
\begin{array}{rcll}
\frac{9}{4} - \frac{1}{4}x_2 & \leq & x_1 & (1) \\
4 - 2x_2 & \leq & x_1 & (2) \\
\frac{1}{2}x_2 & \leq & x_1 & (3) \\
6 - 6x_2 & \leq & x_1 & (4) \\
x_1 & \leq & 11 - 2x_2 & (5) \\
x_1 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (6) \\
-4 & \leq & -x_2 & (7)
\end{array}
$$

If such an x_1 exists, all lower bounds must be bounded by all upper bounds, i.e.,

$$
\begin{array}{rcll}
\frac{9}{4} - \frac{1}{4}x_2 & \leq & 11 - 2x_2 & (1,5) \\
\frac{9}{4} - \frac{1}{4}x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (1,6) \\
4 - 2x_2 & \leq & 11 - 2x_2 & (2,5) \\
4 - 2x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (2,6) \\
\frac{1}{2}x_2 & \leq & 11 - 2x_2 & (3,5) \\
\frac{1}{2}x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (3,6) \\
6 - 6x_2 & \leq & 11 - 2x_2 & (4,5) \\
6 - 6x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (4,6) \\
-4 & \leq & -x_2 & (7)
\end{array}
\quad \text{or} \quad
\begin{array}{rcll}
-35 & \leq & -7x_2 & (1,5) \\
-\frac{7}{12} & \leq & \frac{7}{12}x_2 & (1,6) \\
-7 & \leq & 0 & (2,5) \\
\frac{7}{6} & \leq & \frac{7}{3}x_2 & (2,6) \\
-22 & \leq & -5x_2 & (3,5) \\
-\frac{17}{6} & \leq & -\frac{1}{6}x_2 & (3,6) \\
-5 & \leq & 4x_2 & (4,5) \\
\frac{19}{6} & \leq & \frac{19}{3}x_2 & (4,6) \\
-4 & \leq & -x_2 & (7)
\end{array}
$$

$$
\begin{array}{rclr}
\frac{9}{4} - \frac{1}{4}x_2 & \leq & 11 - 2x_2 & (1,5) \\
\frac{9}{4} - \frac{1}{4}x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (1,6) \\
4 - 2x_2 & \leq & 11 - 2x_2 & (2,5) \\
4 - 2x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (2,6) \\
\frac{1}{2}x_2 & \leq & 11 - 2x_2 & (3,5) \\
\frac{1}{2}x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (3,6) \\
6 - 6x_2 & \leq & 11 - 2x_2 & (4,5) \\
6 - 6x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (4,6) \\
-4 & \leq & -x_2 & (7)
\end{array}
\quad \text{or} \quad
\begin{array}{rclr}
-35 & \leq & -7x_2 & (1,5) \\
-\frac{7}{12} & \leq & \frac{7}{12}x_2 & (1,6) \\
-7 & \leq & 0 & (2,5) \\
\frac{7}{6} & \leq & \frac{7}{3}x_2 & (2,6) \\
-22 & \leq & -5x_2 & (3,5) \\
-\frac{17}{6} & \leq & -\frac{1}{6}x_2 & (3,6) \\
-5 & \leq & 4x_2 & (4,5) \\
\frac{19}{6} & \leq & \frac{19}{3}x_2 & (4,6) \\
-4 & \leq & -x_2 & (7)
\end{array}
$$

$$
\begin{array}{rclr}
\frac{9}{4} - \frac{1}{4}x_2 & \leq & 11 - 2x_2 & (1,5) \\
\frac{9}{4} - \frac{1}{4}x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (1,6) \\
4 - 2x_2 & \leq & 11 - 2x_2 & (2,5) \\
4 - 2x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (2,6) \\
\frac{1}{2}x_2 & \leq & 11 - 2x_2 & (3,5) \\
\frac{1}{2}x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (3,6) \\
6 - 6x_2 & \leq & 11 - 2x_2 & (4,5) \\
6 - 6x_2 & \leq & \frac{17}{6} + \frac{1}{3}x_2 & (4,6) \\
-4 & \leq & -x_2 & (7)
\end{array}
\quad \text{or} \quad
\begin{array}{rclr}
-5 & \leq & -x_2 & (1,5) \\
-1 & \leq & x_2 & (1,6) \\
-7 & \leq & 0 & (2,5) \\
\frac{1}{2} & \leq & x_2 & (2,6) \\
-\frac{22}{5} & \leq & -x_2 & (3,5) \\
-17 & \leq & -x_2 & (3,6) \\
-\frac{5}{4} & \leq & x_2 & (4,5) \\
\frac{1}{2} & \leq & x_2 & (4,6) \\
-4 & \leq & -x_2 & (7)
\end{array}
$$

This is the one-variable case which we can solve exactly:

$$
\max\left\{-1, \boxed{\tfrac{1}{2}}, -\tfrac{5}{4}, \tfrac{1}{2}\right\} \;\leq\; x_2 \;\leq\; \min\left\{5, \tfrac{22}{5}, 17, \boxed{4}\right\}
$$

From which we conclude: $x_2 \in [\frac{1}{2}, 4]$.

In General:

- The original system has a solution over $\mathbb{Q}$ iff the system after elimination of one variable has a solution over $\mathbb{Q}$.
- Every elimination step may square the number of in-equations $\Longrightarrow$ exponential run-time.
- It can be modified such that it also decides satisfiability over $\mathbb{Z}$ $\Longrightarrow$ Omega Test

William Worthington Pugh, Jr.
University of Maryland, College Park

Idea

- We successively remove variables. Thereby we omit division ...
- If x only occurs with coefficient ± 1, we apply Fourier-Motzkin elimination.
- Otherwise, we provide a bound for a positive multiple of x ...

Consider, e.g., (1) and (6) :

$$\begin{aligned} 6 \cdot x_1 &\leq 17 + 2x_2 \\ 9 - x_2 &\leq 4 \cdot x_1 \end{aligned}$$

W.l.o.g., we only consider strict in-equations:

$$\begin{aligned} 6 \cdot x_1 &< 18 + 2x_2 \\ 8 - x_2 &< 4 \cdot x_1 \end{aligned}$$

... where we always divide by gcds:

$$\begin{aligned} 3 \cdot x_1 &< 9 + x_2 \\ 8 - x_2 &< 4 \cdot x_1 \end{aligned}$$

This implies:

$$3 \cdot (8 - x_2) \quad < \quad 4 \cdot (9 + x_2)$$