# A “Piano Movers” Problem Reformulated

James Davenport (Bath)

Thanks to Russell Bradford, Matthew England and David Wilson (Bath)

23 September 2013

# Background

- Thirty years ago [SS83b, and others] showed that many problems of robot motion planning can be reduced to Cylindrical Algebraic Decomposition.
- 28 years ago, I tried to do this [Dav86]

!! and failed spectacularly

- There has been lots of progress since then, Moore's Law etc. on the hardware front,
- and at least as much on the software front

So What happens if we try again?

# The specimen problem

(they don't come much simpler than this!)
Can we get the ladder from 1 to 2?

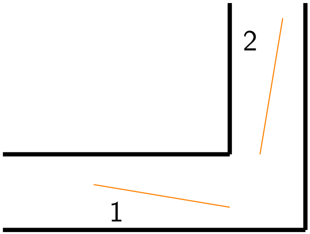



Figure: The piano movers problem considered in [Dav86]

# What space are we in?

Real: In this case $\mathbf{R}^2 = \{(x, y)\}$ a point in space

Configuration: In this case $\mathbf{R}^4 = \{(x, y, w, z)\}$ positions of the two ends of the ladder.

'' Manifold: $\mathbf{R}^4/\langle (x-w)^2 + (y-z)^2 = 9 \rangle$ as above allowing for the length of the ladder

Possibly: $\mathbf{R}^2 \times \mathbf{S}^1 = \{(x, y, \theta)\}$ an end-point in space and an orientation

Isomorphic: $w = x + 3\cos\theta$; $z = y + 3\sin\theta$.

Originally due to Collins [Col75]

| | |
|---|---|
| Input | Polynomials $\mathcal{P}_n = \{p_1, \ldots, p_k\} \subset \mathbf{R}[x_1, \ldots, x_n]$ |
| Output | Decompose $\mathbf{R}^n$ into disjoint connected cells $D_i$ such that |
| Useful | Each cell has an explicit *sample point* |
| Relevant | Each $p_i$ has a constant sign on each cell |
| Cyl. $\forall m < n$ | $\pi_m(D_i)$, $\pi_m(D_j)$ are either equal or disjoint, where $\pi_m$ is projection onto the *first* $m$ coordinates |

# Initial Method (outline)

| | |
|---|---|
| $\mathcal{P}_n \to \mathcal{P}_{n-1}$ | Project on $n-1$ variables |
| $\vdots$ | and so on |
| $\mathcal{P}_2 \to \mathcal{P}_1$ | Project to univariate polynomials |
| Isolate | All the roots of $\mathcal{P}_1$, which is a c.a.d. of $\mathbf{R}^1$ |
| Lift | To a c.a.d. of $\mathbf{R}^2$ sign-invariant for $\mathcal{P}_2$ |
| $\vdots$ | and so on |
| Lift | To a c.a.d. of $\mathbf{R}^n$ sign-invariant for $\mathcal{P}_n$ |

In practice, lifting is by far the most expensive part

# Davenport's formulation [Dav86]

$(x, y)$ and $(w, z)$ are the two ends of the ladder
(Configuration space)

$$\begin{aligned}
&\left[(x-w)^2+(y-z)^2-9=0\right] \\
\wedge &\left[[yz \geq 0] \vee [x(y-z)^2+y(w-x)(y-z) \geq 0]\right] \\
\wedge &\left[[(y-1)(z-1) \geq 0]\right. \\
&\left.\vee [(x+1)(y-z)^2+(y-1)(w-x)(y-z) \geq 0]\right] \\
\wedge &\left[[xw \geq 0] \vee \ [y(x-w)^2+x(z-y)(x-w) \geq 0]\right] \\
\wedge &\left[[(x+1)(w+1) \geq 0]\right. \\
&\left.\vee [(y-1)(x-w)^2+(x+1)(z-y)(x-w) \geq 0]\right].
\end{aligned}$$

In 1985 (12MB memory) he could do the projection (250 polynomials, of degree $\leq 26$) but not the isolation (375 real roots).

# Improvements since 1975

- finer projection operators [McC88];
- Partial CAD to make use of the quantified structure of a formula when lifting [CH91];
- the use of equational constraints [McC99];
- truth-table-invariant CADs to apply equational constraint techniques more widely [BDE+13];
- and an alternative approach to projection and lifting where the problem is solved in complex space and then refined to a CAD of real space [CMMXY09].

Today (3.1GHz processor, 8192MB memory) we still can't compute this c.a.d.

# Alternative Formulations

[SS83a] A non-c.a.d. method for $\mathbf{R}^2$, which does not generalise

[Wan96] uses "simple reasoning" to deduce that the ladder cannot traverse the corridor if and only if it intersects all four walls simultaneously

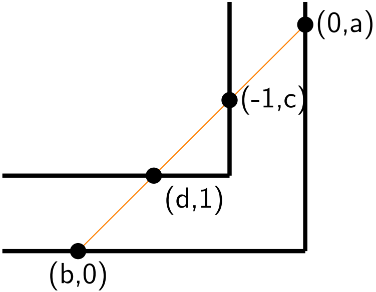



Figure: A configuration of a ladder in which all four walls are intersected.

# Wang's formulation

$$(\exists a)(\exists b)(\exists c)(\exists d)[a^2 + b^2 = r^2 \wedge r > 0$$
$$\wedge\, a \geq 0 \wedge b < 0 \wedge c \geq 1 \wedge d < -1$$
$$\wedge\, c - (1 + b)(c - a) = 0 \wedge d - (1 - a)(d - b) = 0].$$

Due to its simplicity and the small number of free variables (only $r$ is unquantified) QEPCAD can almost instantly deduce that the maximal length of the ladder is $\sqrt{8}$, using a CAD of 19 cells.

Also (1991) We can use 'topological reasoning' to deduce that three intersections imply four

# Alternative Formulations (II)

[SS83a] A non-c.a.d. method for $\mathbf{R}^2$, which does not generalise

[Wan96] uses "simple reasoning" to deduce that the ladder cannot traverse the corridor if and only if it intersects all four walls simultaneously.

Also (1991) We can use 'topological reasoning' to deduce that three intersections imply four

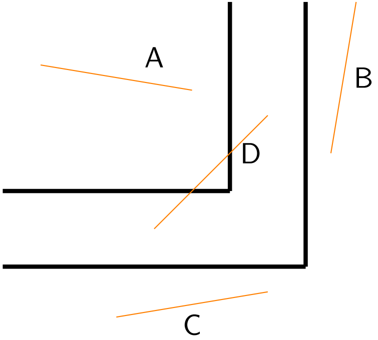
[YZ06] Subtle geometry implies $r$ satisfies

$$(\forall x)\, 4x^8 - 4(r-3)x^6 - 2(3r-6)x^4 - 2(r-3)x^2 + 1 > 0.$$

It takes QEPCAD 1.936 seconds and 5 cells to return $r^2 < 8 \vee r < 0$

These two are in real space, and don't return a route

# New Formulation: consider the illegal positions




In A–C, only one end need be "illegal"

Figure: Four canonical invalid positions of the ladder.

# So **invalid** regions are

A $x < -1 \wedge y > 1$ or $w < -1 \wedge z > 1$: this describes any collision with the 'inside' walls along with the ladder being on the other side of these.

B $x > 0$ or $w > 0$: this includes any collision with the rightmost wall along with the ladder being on the other side.

C $y < 0$ or $z < 0$: this includes any collision with the bottommost wall along with the ladder being on the other side.

D $(\exists t)[0 < t \wedge t < 1 \wedge x + t(w - x) < -1 \wedge y + t(z - y) > 1]$: this is the condition that there is any point of the line that lies in the invalid top-left region.

Valid space is the negation of $(A) \vee (B) \vee (C) \vee (D)$
QEPCAD (2 seconds) eliminates $t$ from $(A) \vee (B) \vee (C) \vee (D)$ to give $(A) \vee (B) \vee (C) \vee (D')$

# $(A) \vee (B) \vee (C) \vee (D')$

$$
\begin{aligned}
&[y < 0] \vee [w > 0] \vee [x > 0] \vee [z < 0] \\
&\vee [x + 1 < 0 \wedge y - 1 > 0] \vee [w + 1 < 0 \wedge z - 1 > 0] \\
&\vee [w + 1 < 0 \wedge yw - w + y + x \geq 0 \\
&\qquad \wedge xz + z - yw + w - y - x > 0] \\
&\vee [yw - w + y + x < 0 \wedge z - 1 > 0 \\
&\qquad \wedge xz + z - yw + w - y - x < 0] \\
&\vee [y - 1 > 0 \wedge yw - w + y + x < 0].
\end{aligned}
$$

Hence we need

$$\left[(x - w)^2 + (y - z)^2 = 9\right] \wedge \neg(\text{the above})$$

# And the answer is . . . : QEPCAD 16,933.701 seconds

$$
\begin{aligned}
& x \le 0 \wedge y \ge 0 \wedge w \le 0 \wedge z \ge 0 \wedge (y-z)^2 + (x-w)^2 = 9 \\
& \wedge \Big[ [x+1 \ge 0 \wedge w+1 \ge 0] \vee \big[ y-1 \le 0 \wedge w+1 \ge 0 \\
& \qquad \wedge y^2w^2 - 2yw^2 + x^2w^2 + 2xw^2 + 2w^2 - 2xy^2w \\
& \qquad\quad + 4xyw - 2x^3w - 4x^2w - 4xw + x^2y^2 - 2x^2y \\
& \qquad\quad + x^4 + 2x^3 - 7x^2 - 18x - 9 \ge 0 \big] \\
& \quad \vee \big[ x+1 \ge 0 \wedge yw - w + y + x \ge 0 \wedge w^2 - 2xw + y^2 \\
& \qquad - 2y + x^2 - 8 > 0 \wedge z - 1 \le 0 \big] \\
& \quad \vee \big[ x+1 \ge 0 \wedge yw - w + y + x \ge 0 \wedge y^2w^2 - 2yw^2 \\
& \qquad + x^2w^2 + 2xw^2 + 2w^2 - 2xy^2w + 4xyw - 2x^3w \\
& \qquad\quad - 4x^2w - 4xw + x^2y^2 - 2x^2y + x^4 + 2x^3 - 7x^2 \\
& \qquad\quad - 18x - 9 \le 0 \wedge z - 1 \le 0 \big] \\
& \quad \vee [y-1 \le 0 \wedge z-1 \le 0] \Big]
\end{aligned}
$$

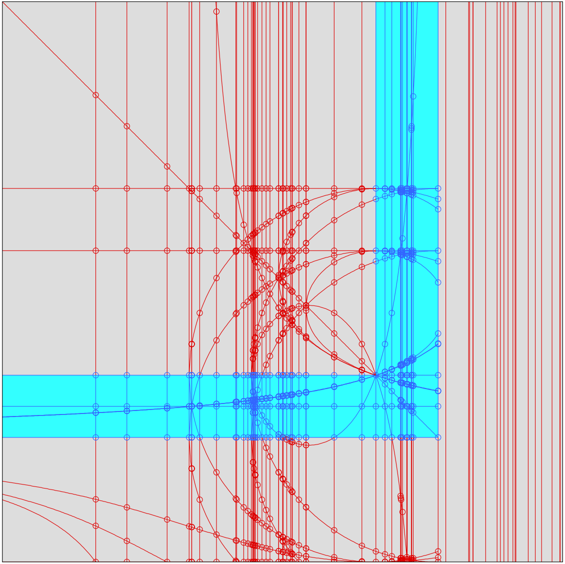
# If that's the answer, what does it mean?

Good question (285,419 cells)! The formula can be seen as

$$\text{general conditions} \wedge \left[\text{bottom corridor} \vee (\text{intermediates})^3 \vee \text{ upper corridor}\right]$$

and the question is whether cells representing these "intermediate" positions connect the two corridors.
Need it really be this complicated?

# A two-dimensional CAD of the $(x, y)$ configuration space



# Adjacency

is actually a non-trivial question itself

[ACM84] describe adjacency in 2D, implemented in QEPCAD

[ACM88] describe adjacency in 3D

[SS83b] describe adjacency of $n$ and $n-1$ dimensional cells

But we have an equational constraint, so need adjacency of $n-1$ and $n-2$ dimensional cells

# Why have we got further than [Dav86]?

- Intuitively, new formulation has lower degree
- Backed up by `sotd` heuristic on the formulations: 100/33
- Not so convincing on the projections: 2006/1693
- Over 100 polynomials in $\mathcal{P}_1$ in both cases
- `ndrr` [BDEW13] gives 367/301

[Wan96] `sotd`=19 (98 for projection), `ndrr`=17

A better heuristic (here!) is "sum of weighted total degree" (`sowtd`) —give $x_i$ the weight of $i$ (or $i/2$ if quantified over).

# sowtd

- [Dav86] (unquantified): `sowtd` = 148.
- New formulation (unquantified): `sowtd` = 72.
- [Wan96]'s formulation: `sowtd` = 27.
- [YZ06]'s formulation: `sowtd` = 23.

The `sowtd` measure gives the right order to these formulations, and has plausible-looking differences.

# Extensions

We can consider

- Ladder's of different lengths: 3, 2, $\frac{5}{4}$, $\frac{3}{4}$
- Obtuse angles (No [YZ06] here)
- Acute angles (note that [Wan96] is inadequate, as the ladder can turn round in the corner)

!! Our formulation runs out of time on both of these — we really benefited from the fact that the walls were aligned with the axes.

See paper

# Future work

A c.a.d. of $\mathbf{R}^n$ is overkill

- We only need the manifold $(z-w)^2+(y-z)^2=9$, not $\mathbf{R}^4$
- We only need cells of codimension 0 and 1 *on this manifold*

Since connectivity through a cell of codimension $>1$ is "walking a tightrope"

Hence Layered Manifold c.a.d. [WE13]

# Bibliography


D.S Arnon, G.E. Collins, and S. McCallum.
Cylindrical Algebraic Decomposition II: An Adjacency Algorithm for the Plane.
*SIAM J. Comp.*, 13:878–889, 1984.

D.S. Arnon, G.E. Collins, and S. McCallum.
An Adjacency Algorithm for Cylindrical Algebraic Decompositions of Three-dimensional Space.
*J. Symbolic Comp.*, 5:163–187, 1988.

R.J. Bradford, J.H. Davenport, M. England, S. McCallum, and D.J. Wilson.
Cylindrical Algebraic Decompositions for Boolean Combinations.
In *Proceedings ISSAC 2013*, pages 125–132, 2013.

# Bibliography II


R.J. Bradford, J.H. Davenport, M. England, and D.J. Wilson.
Optimising Problem Formulation for Cylindrical Algebraic Decomposition.
In J. Carette *et al.*, editor, *Proceedings CICM 2013*, pages 19–34, 2013.

G.E. Collins and H. Hong.
Partial Cylindrical Algebraic Decomposition for Quantifier Elimination.
*J. Symbolic Comp.*, 12:299–328, 1991.

# Bibliography III


C. Chen, M. Moreno Maza, B. Xia, and L. Yang.
Computing Cylindrical Algebraic Decomposition via Triangular Decomposition.
In J. May, editor, *Proceedings ISSAC 2009*, pages 95–102, 2009.

G.E. Collins.
Quantifier Elimination for Real Closed Fields by Cylindrical Algebraic Decomposition.
In *Proceedings 2nd. GI Conference Automata Theory & Formal Languages*, pages 134–183, 1975.

J.H. Davenport.
On a "Piano Movers" Problem.
*SIGSAM Bulletin 1/2*, 20:15–17, 1986.

# Bibliography IV


S. McCallum.
An Improved Projection Operation for Cylindrical Algebraic Decomposition of Three-dimensional Space.
*J. Symbolic Comp.*, 5:141–161, 1988.

S. McCallum.
On Projection in CAD-Based Quantifier Elimination with Equational Constraints.
In S. Dooley, editor, *Proceedings ISSAC '99*, pages 145–149, 1999.

J.T. Schwartz and M. Sharir.
On the "Piano-Movers" Problem: I. The Case of a Two-dimensional Rigid Polygonal Body Moving Amidst Polygonal Barriers.
*Comm. Pure Appl. Math.*, 36:345–398, 1983.

# Bibliography

V


J.T. Schwartz and M. Sharir.
On the "Piano-Movers" Problem: II. General Techniques for Computing Topological Properties of Real Algebraic Manifolds.
*Adv. Appl. Math.*, 4:298–351, 1983.

D. Wang.
Geometry machines: From AI to SMC.
In J. Calmet, J.A. Campbell, , and J. Pfalzgraf, editors, *Proceedings Artificial Intelligence and Symbolic Mathematical Computation*, pages 213–239, 1996.

# Bibliography VI


D.J. Wilson and M. England.
Layered cylindrical algebraic decomposition.
Technical Report CSBU-2013-05 Department of Computer Science University of Bath, 2013.

L. Yang and Z. Zeng.
Symbolic solution of a piano movers' problem with four parameters.
*Automated Deduction in Geometry*, pages 59–69, 2006.