APPLIED aspects of COMPUTATIONAL GEOMETRY

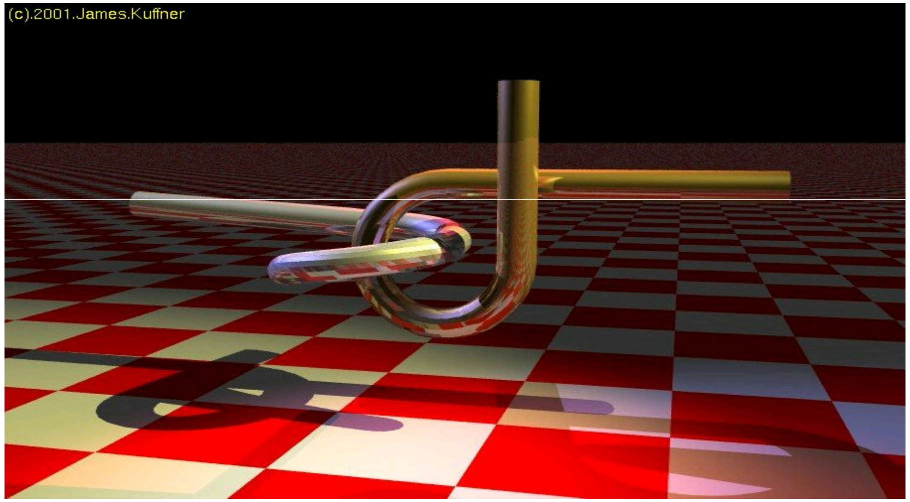
Movable separability and assembly planning

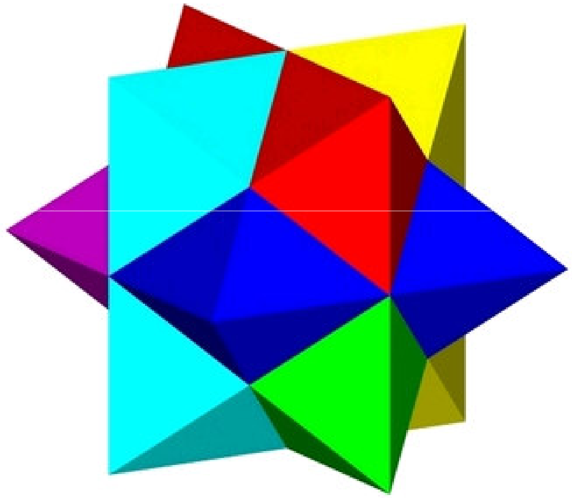
Dan Halperin
School of Computer Science
Tel Aviv University

Overview

- movable separability
- separation sequences for convex objects in 2D
- a general framework for 2-handed assembly planning
- improved algorithm: infinitesimal motions

Movable separability





Definitions

- assembly *A*: a collection of pairwise interior disjoint bodies/parts
- subassembly: a subset of parts of *A* in their relative placement in (the full final) *A*
- assembly operation: a motion that merges *s* subassemblies of *A* into a new subassembly of *A*; *s* is the number of hands
- the reverse is called assembly partitioning
- assembly sequence: a total ordering on assembly operations from separated parts to the full *A*
- monotone operation: only generating subassemblies of the final *A*

we will discuss

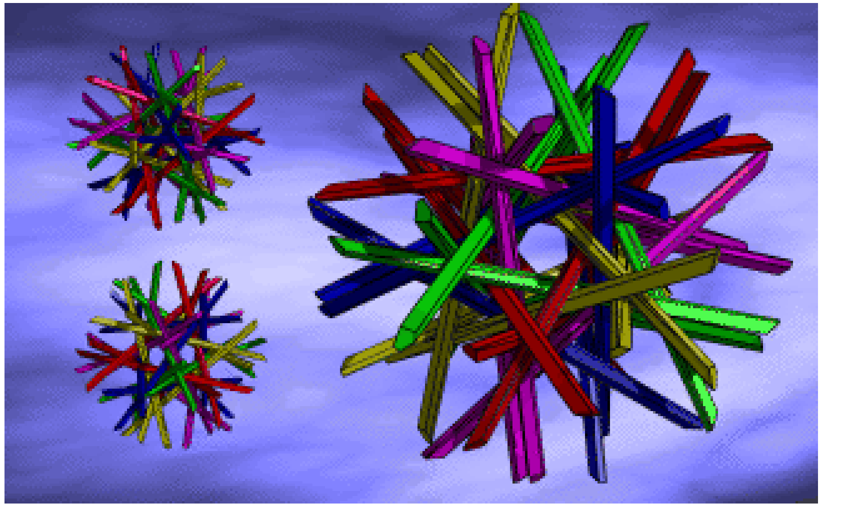
- 2-handed
- monotone
- rigid-parts

assembly planning

hence we can plan assembly by disassembly

Convex objects

- in the plane: admit a disassembly sequence translating one part at a time along a fixed (arbitrary) direction to infinity
- in 3-space?
  - depth order does not always exist
  - moreover, assemblies of convex parts may be interlocked



[Snoeyink-Stolfi 93]

The partitioning problem, hardness

- arbitrary motions: assembly planning for polyhedral objects of constnat maximum complexity each is PSPACE-hard

- 2-handed assembly partitioning for polygonal parts with translational motions only and into connected subassemblies is NP-complete

General framework for assembly planning, the basic ingredients

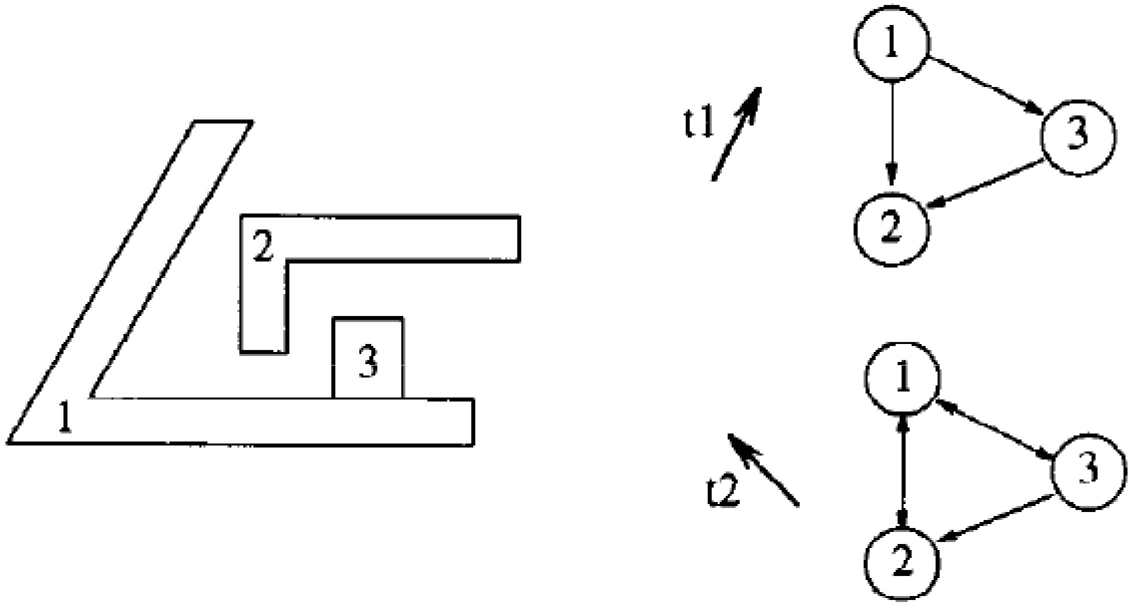
- key factor: the type of motions allowed
- reminder: strong connectivity
- DBG: the directional blocking graph
- motion-space regions (open sets) = blocking regions, M_{ij}
- NDBG: the arrangement of the M_{ij} 's together with their BGS's

Strong connectivity

- a strongly connected component (or strong component) of a directed graph is a maximal subset of nodes such that for any pair of nodes u,v, in this subset a path connects u to v
- a graph is strongly connected if it consists of one strongly connected component

<back>

DBG



<back>

The partitioning procedure

procedure `partition`(S);
    for every cell c in the NDBG of S **do**:
        if the DBG associated with c is not strongly connected
            then return c and a feasible partition of S;
    return failure;

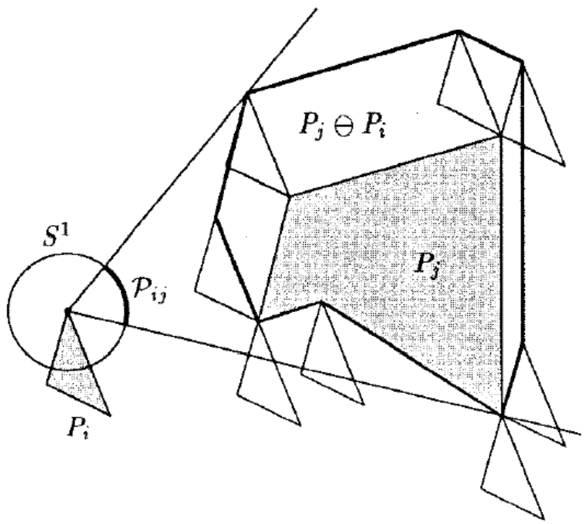
2-handed assembly by disassembly

- input: an assmebly A, the allowable motions
- output: assembly sequence

- algorithm:
  - partition A, and then the two subassemblies recursively, or stop and announce failure
  - reverse the disassembly motions into an assembly sequence

Example: infinite translational partitioning in 2D

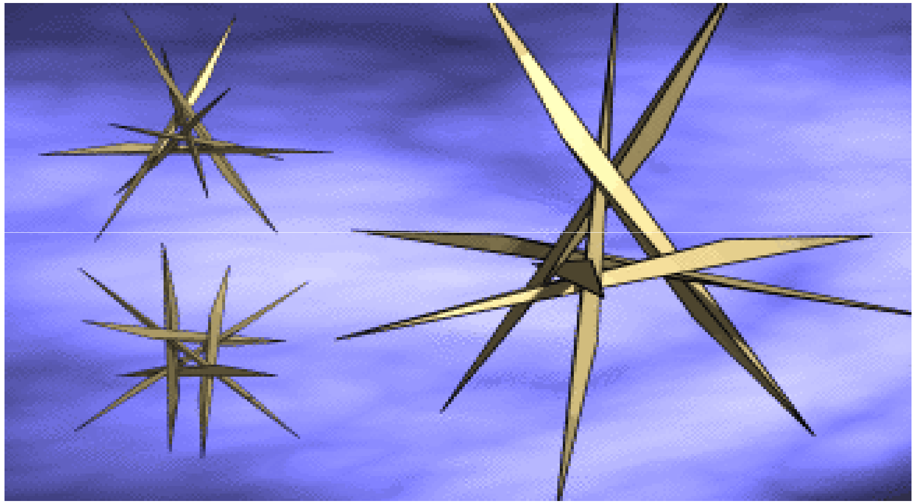
- polygonal parts $P_1, P_2, \ldots P_n$
- motion-space model: the unit circle
- blocking region M_{ij} = the motions where P_j blocks P_i

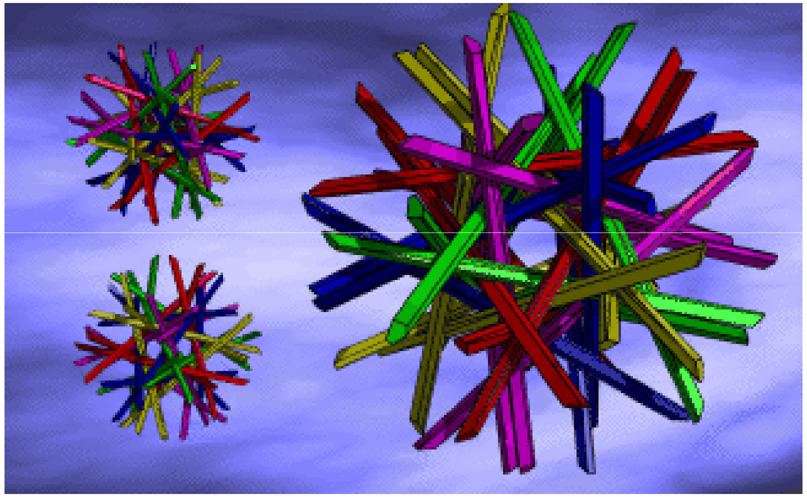


infinite translational partitioning in 2D, complexity

- n parts, at most q vertices in a part
- computing an M_{ij} takes O(q) time
- computing the motion-space arrangement, $O(n^2)$ time
- constructing the NDBG takes $O(n^2(\log n+q^2))$
- checking all DBGs for strong connectivity, at $O(n^2)$ a piece, takes total $O(n^4)$ time
- (dis)assemble takes $O(n^5)$ time

Interlocking and infinitesimal motions



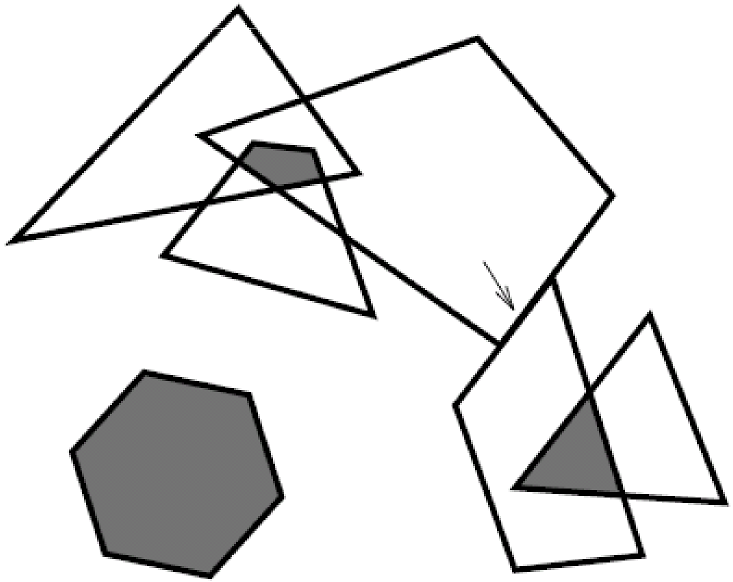


Example: infinitesimal translation in 3D

- The approach works in motion-space of any dimension, especially interesting d=2,5; we exemplify for d=2 (translation in 3-space)
- polyhedral parts $P_1, P_2, \ldots P_n$
- motion-space model: the (upper) hemisphere, or the plane tangent to the North pole
- the non-blocking region Q_{ij} = where P_j does not block P_i
  closed sets
- The motion-space arrangement
  - covering set
  - maximally covered cells

Maximally covered cells

- covered by more sets than any of their immediate neighbors, or
- a cell C is maximally covered if the covering set of any point on the relative boundary of C is the same as the covering set of its interior, and C is a maximal connected region of d-space with this covering set



Maximally covered cells suffice

claim: if there is a solution (S,δ) to the infinitesimal partitioning problem with subassembly S and direction δ, then there is a solution (S,δ’) where δ’ is in a maximally covered cell

Motion-space arrangement in d-space

- K: # of pairs of parts in contact
- N: total # of constraints

thus we have K polytopes in Rd with a total of N facets

- combinatorial bounds
  - arrg of hyperplanes

  $O(N^d)$

  - arrg of K polytopes with N facets

  $O(N^{\lfloor \frac{d}{2} \rfloor} K^{\lceil \frac{d}{2} \rceil})$

The number of maximally covered cells

- K: # of pairs of parts in contact
- N: total # of constraints

- Q: how many maximally covered cells?
- A: $O(K^d)$

The number of DBG's to test depends on the number of pairs of parts in contact and not on the complexity of the parts or the complexity of their contacts

K is never greater than N and typically much smaller

Finding a point in each maximally covered cell (MCC)

- consider the bottommost point in each bounded MCC (lowest in the X_d direction)
- it is a meeting point of at least d facets that are coming from at most d polytopes (more facets may meet there but we consider only d)
- look at the bottommost point in the intersection of any i polytopes in our set for i = 1,2, …, d
- for unbounded cells, add a bounding box B of all the vertices of the arrg, and repeat the above with B as one of the at most d polytopes

Infinitesimal partitioning, algorithm

1. for $i = 1, \ldots, d$
    2. for each subset R of input polytopes such that $|R| = i$
        3. $H \leftarrow \bigcup_{r \in R}$(Halfspaces in r)
        4. $e \leftarrow \; LP(H, X_d \downarrow)$
        5. If $e =$ NULL goto 2, else
        6. If DBG(e) is not strongly connected
            7. Output e and movable subassembly
            8. exit
9. Report "INTERLOCKED"

repeat the above while taking the bounding box B in each subset R

Infinitesimal partitioning, analysis

- solving all the LP's takes total time $O(K^{d-1}N)$
- cannot use adjacency in building DBG's, needs point location structures
- let m_i denote the number of facets of the i-th polytope
- overall construction time

$$\sum_{i=1}^{K} O(m_i^{\lfloor \frac{d}{2} \rfloor + \epsilon}) = O(N^{\lfloor \frac{d}{2} \rfloor + \epsilon})$$

- overall query time

$$O(K^d) \sum_{i=1}^{K} O(\log m_i) = O(K^{d+1} \log N)$$

Infinitesimal trans + rot in 3-space

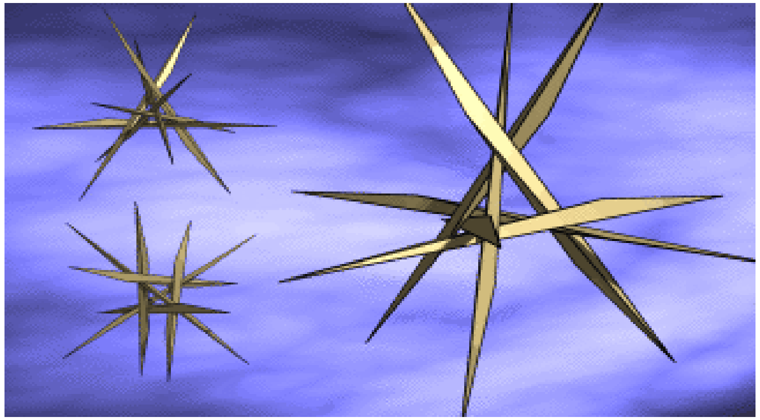
- motion space is S^5
- contact constraints transformed into point/plane contacts
- each contact constraint is a 5-dim hyperplane

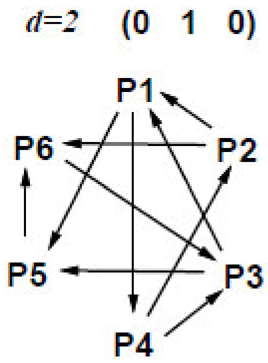
- running time $O(K^4N+K^6 \log N)$ after $O(N^{2+eps})$ expected-time preprocessing

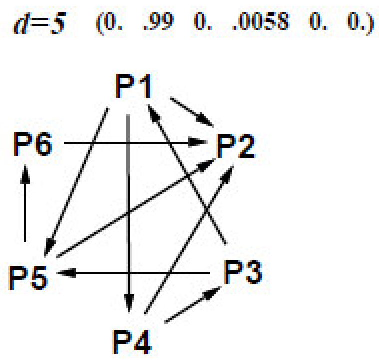
Experiments, K vs N

	Puzzle	Assembly	Snoeyink-Stolfi 6	Snoeyink-Stolfi 6	Snoeyink - Stolfi 30	Engine	Engine
d	2	5	2	5	2	2	5
K	24	6	24	24	300	24	24
N	352	140	96	96	1,200	1,112	1,112
# of Rep. Pts.	3	3	22	190	2,838	4	6
# of S.C. DBGs	2	2	22	118	2,838	0	0
# of Parts	6	3	6	6	30	12	12
Time for the Constraints	104.8	12.3	4.0	4.0	808.0	1881.7	1881.7
Time for the Rep. Pts.	4.7	2.8	2.7	1286.9	509.0	12.6	7880.1
Time for the DBGs	12.7	4.7	27.0	240.5	53,021.3	59.9	120.3

Infinitesimal partitioning, example

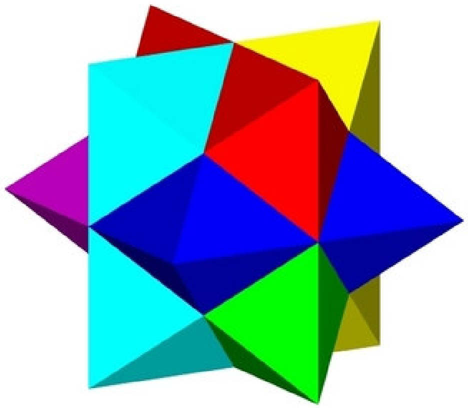


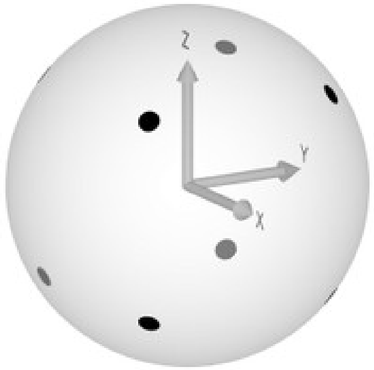




- for d=2, 22 representative points/DBG's generated, all confirmed to be strongly connected
- for d=5, 190 representative DBG's, only 118 of which are strongly connected; above, right: an example of non-strongly connected DBG

Exercise: infinite translation in 3-space





Additional topics, challenges

- tolerancing, sensitivity analysis
- assembly planning with more complex motions
- optimization

References

- [Wilson-Latombe ‘94]

  Geometric reasoning about mechanical assembly, Artifical Intelligence 71(2):371-396
- [Snoeyink-Stolfi ‘94]

  Objects that cannot be taken apart with two hands, DCG 12:367-384
- [Natarajan ’88]

  On planning assemblies, SoCG 299-308
- [Kavraki-Kolountzakis ‘95]

  Partitioning a planar assembly into two-connected parts is NP-complete, IPL 55:159-165
- [Guibas-Halperin-Hirukawa-Latombe-Wilson ‘98]

  Polyhedral assmebly partitioning using maximally covered cells in arrangements of convex polytopes, IJCGA 8:179-200
- [Halperin-Latombe-Wilson ‘00]

  A general framework for assembly planning: The motion space approach, Algorithmica 26:577-601
- [Halperin-Fogel ‘08]

  Polyhedral assembly partitioning with infinite translations or the importance of being exact , WAFR

THE END