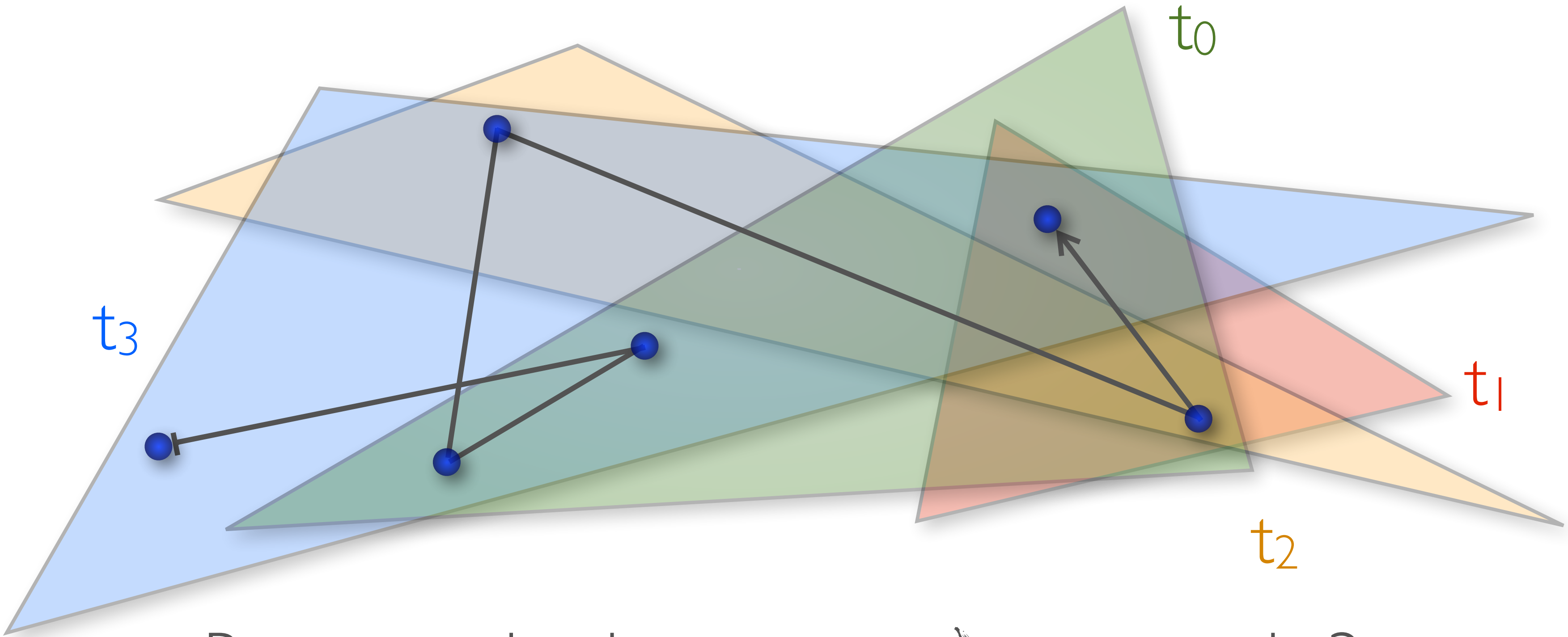
POTW#7: HIKING MAPS

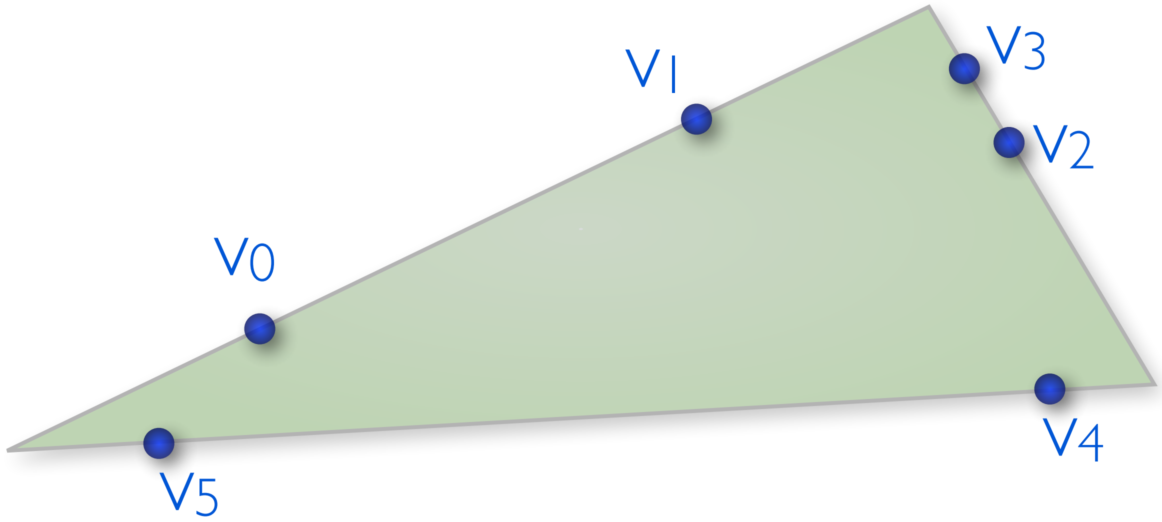
Problem: Given a polygonal path $p_0,\ldots,p_{m-1}$ and triangles $t_0,\ldots,t_{n-1}$, what is the minimum length of an interval $[b,e) \subseteq [0,n)$ s.t. each leg $p_i p_{i+1}$ of the path is contained in at least one of $t_b,\ldots,t_{e-1}$?



Best covering is t_1,t_2,t_3 → answer is 3.

POTW#7: HIKING MAPS

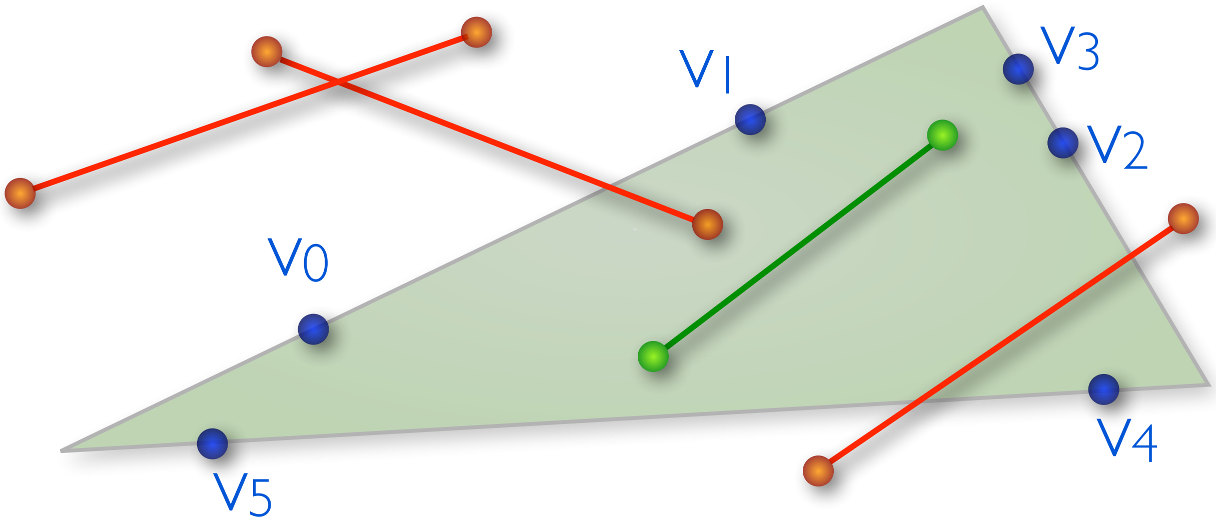
Twist: Each triangle t is given as $(v_0,v_1,v_2,v_3,v_4,v_5)$, where every pair (v_0,v_1), (v_2,v_3), and (v_4,v_5) lies in the relative interior of a different side/edge of t.



Q: How to test whether a triangle contains a leg?

POTW#7: HIKING MAPS

Q: How to test whether a triangle contains a leg?

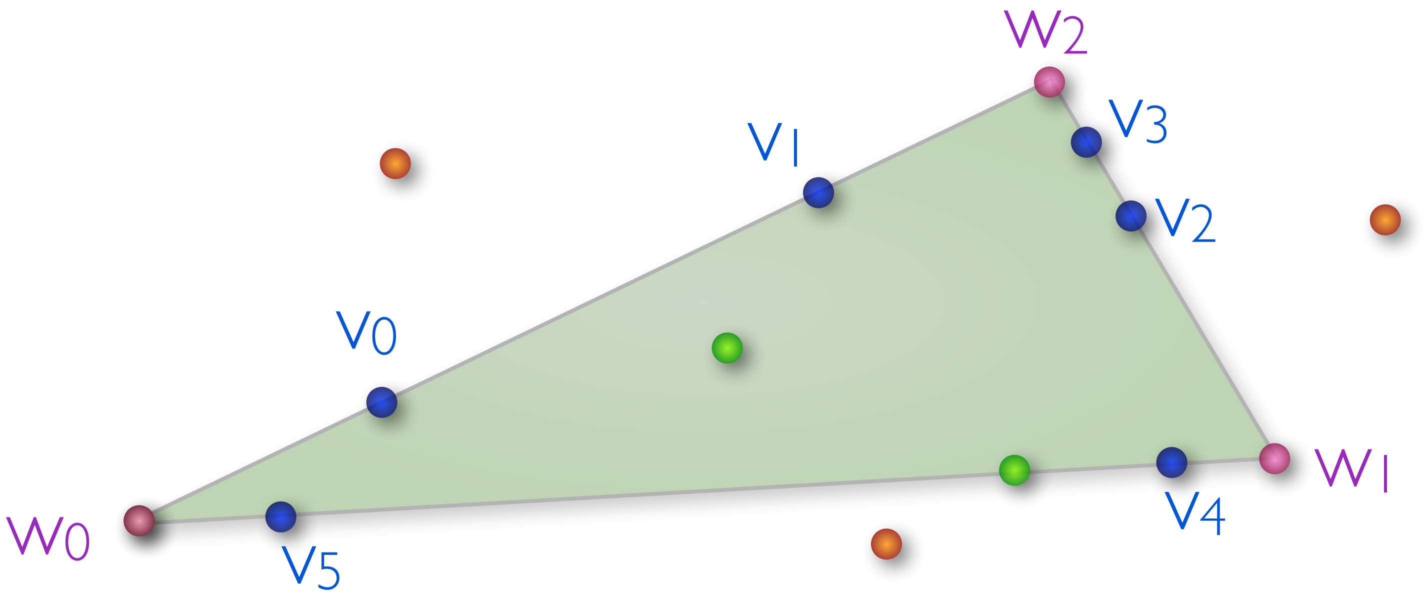


⇔ the triangle contains both endpoints (by convexity)

Q: How to test whether a triangle contains a point?

POTW#7: HIKING MAPS

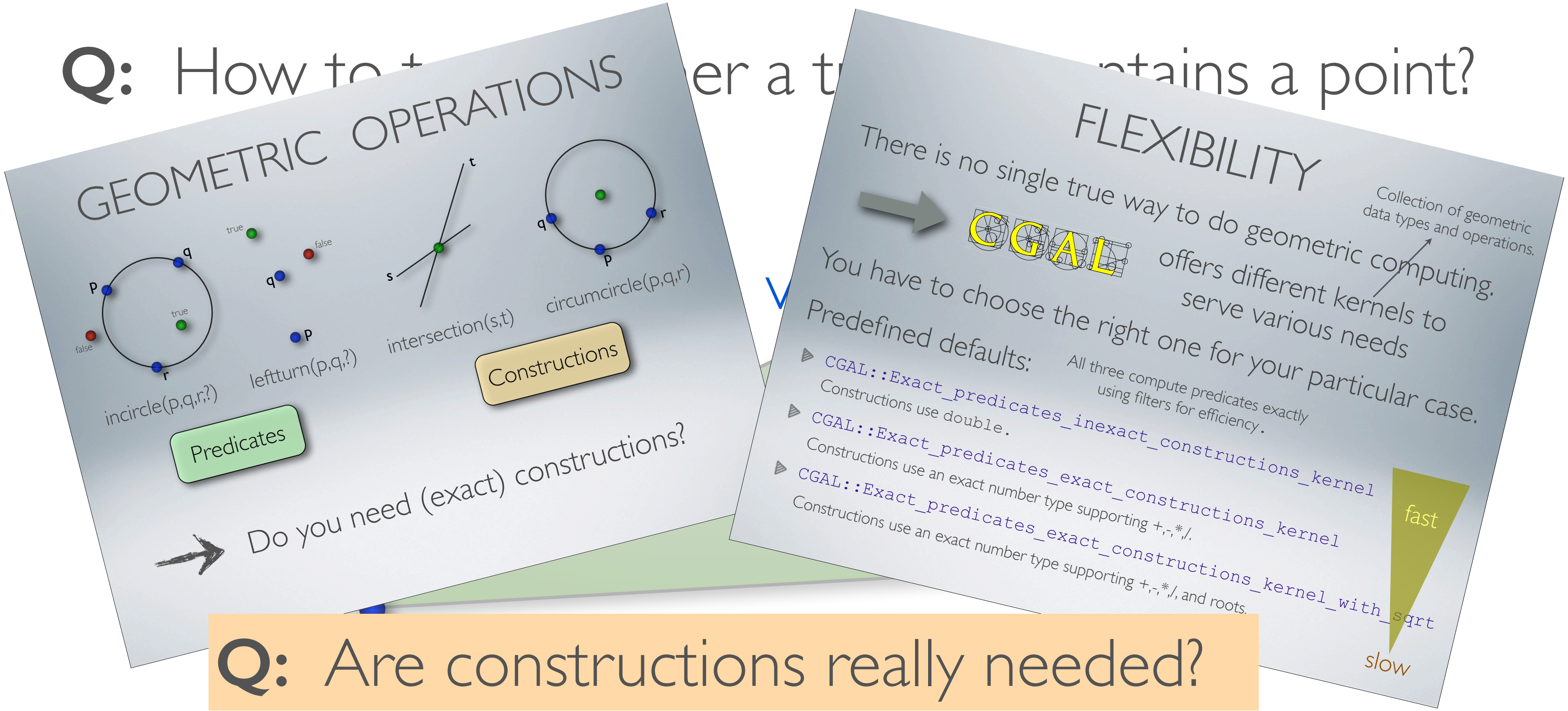
Q: How to test whether a triangle contains a point?



Idea #1: Construct the vertices w_0, w_1, w_2 of t, feed them into a `CGAL::Triangle_2 t`, and use `!t.has_on_unbounded_side(p)`. (or `CGAL::do_intersect(t,p)`)

POTW#7: HIKING MAPS

Q: How to t er a t ntains a point?



Q: Are constructions really needed?

Idea #1: Construct the vertices w_0, w_1, w_2 of t, feed them into a `CGAL::Triangle_2 t`, and use `!t.has_on_unbounded_side(p)`. (or `CGAL::do_intersect(t,p)`)

POTW#7: HIKING MAPS

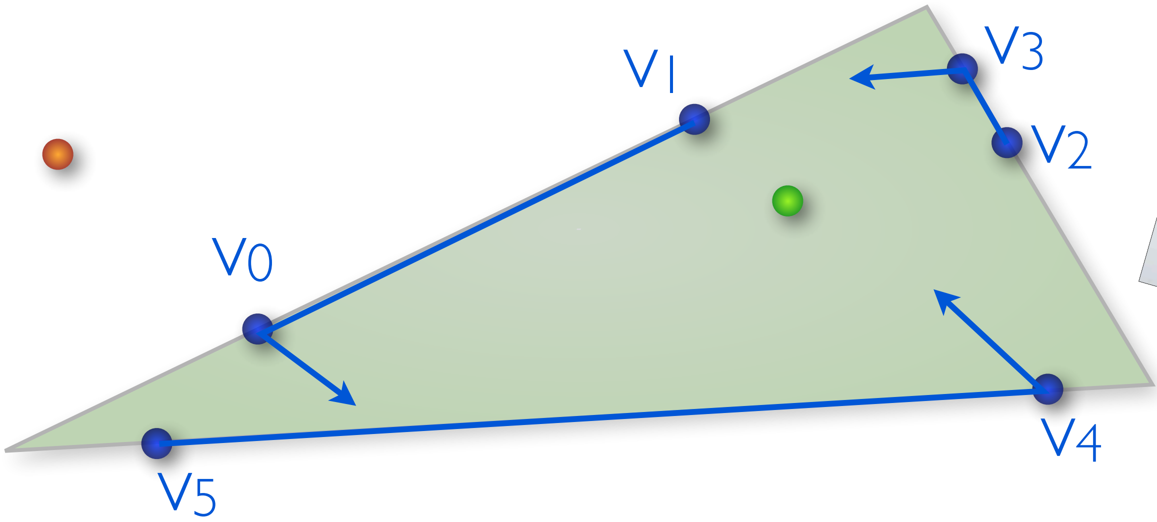
Q: What's the problem with constructions?

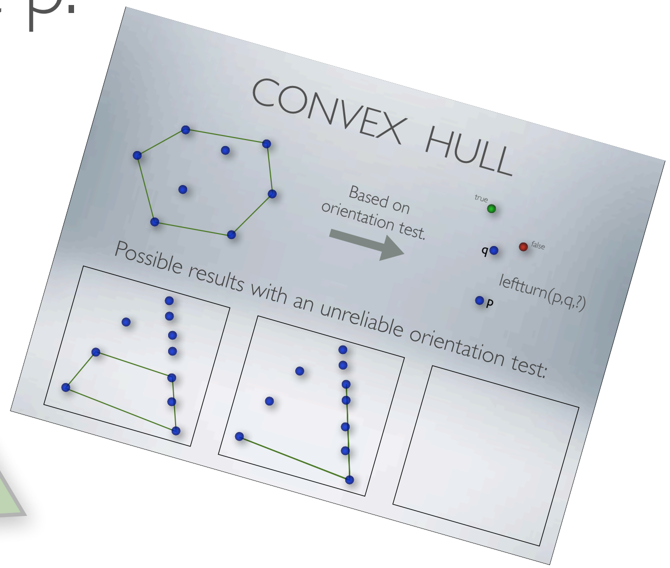
Coordinates of constructed objects may be much larger than those of the input objects.

in terms of the #bits needed to represent them

POTW#7: HIKING MAPS

Idea #2: Use $(v_0,v_1,v_2,v_3,v_4,v_5)$ directly to test whether a triangle t contains a point p.





Obs: $p \in t \Leftrightarrow$ none of (v_1,v_0,p), (v_2,v_3,p), and (v_5,v_4,p) form a right-turn.

This is not only more efficient than Idea#1 but also less work to code.

POTW#7: HIKING MAPS

Problem: Given a polygonal path $p_0,\ldots,p_{m-1}$ and triangles $t_0,\ldots,t_{n-1}$, what is the minimum length of an interval $[b,e) \subseteq [0,n)$ s.t. each leg $p_i p_{i+1}$ of the path is contained in at least one of $t_b,\ldots,t_{e-1}$?

Q: How to find a minimum length interval $[b,e)$?

Idea #1: Try all of them.

There are $\binom{n}{2} = \Theta(n^2)$ possible intervals and testing whether an interval is a cover can be done in $O(mn)$.

→ an $O(mn^3)$ algorithm

That is a lot! Can we do better?

POTW#7: HIKING MAPS

Problem: Given a polygonal path $p_0,\ldots,p_{m-1}$ and triangles $t_0,\ldots,t_{n-1}$, what is the minimum length of an interval $[b,e)\subseteq[0,n)$ s.t. each leg p_ip_{i+1} of the path is contained in at least one of $t_b,\ldots,t_{e-1}$?

Q: How to find a minimum length interval $[b,e)$?

Idea #2: Similar intervals cover a similar set of legs. Update covering information rather than recompute it.

→ cf. Search Snippets…