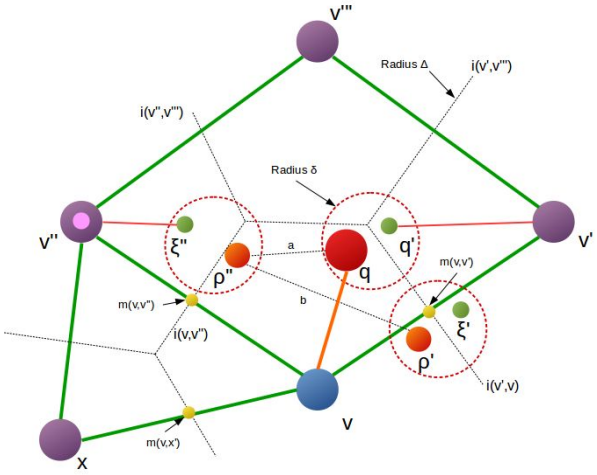
PICKNIK

BOLT

Motion Planning That Learns From Experience

What is Bolt?

Bolt is an experienced-based motion planner that learns over time. It is an approach to generate large graphs of motions optimized for highly-constrained problems.



What is an experienced based motion planning?

It is a paradigm that reduces query resolution time, improves the quality of paths, and results in more predictable motions than typical probabilistic methods. Most previous approaches to motion planning have discarded past solution results and planning from scratch new solutions for every problem. A robot that is in operation for years will never get any better at its routine tasks.

What problem does it solve?

Bolt can efficiently recall previous motions the robot has performed and generalize them to arbitrary new solutions even in the midst of changing obstacle environments. Several key difficulties present themselves in the reuse of previous experiences: efficient storage given memory constraints, quick recall for new queries, verification given changing environments, and adaptation/repair.

How does it solve the problem?

Bolt uses sparse roadmaps that provide theoretical guarantees for asymptotic-near optimality, and lazy collision checking which allows iterative search through a large roadmap of motions.

Why should I care?

This approach has been experimentally shown to offer two orders of magnitude speedups for solving difficult motion planning problems, while at the same time outputting predictable paths comfortable to humans nearby. This means dual arm robots operating in confined spaces will find solutions despite their many restrictions in motion.

What about machine learning approaches?

Applying neural networks to the control of robotic arms is still in its early infancy and has only been shown to be useful for vary narrowly defined problems in academic settings.

Why is determinism important?

Humans do not feel comfortable working around robots that will change their behavior in unexpected and sometimes wild ways. Past approaches to motion planning have relied heavily on probabilistic methods that produce un-human motions that are undesirable in collaborative settings.

Why not use optimization-based approaches?

Optimization-based approaches have no theoretical guarantees to completeness and will be unable to find global solutions in difficult, constrained problems. Optimization approaches can get stuck in local minimum.

Why not use search-based approaches?

Search-based planning approaches are limited by the discretization granularity chosen and are unable to adapt to certain problems such as small narrow passageways.

What are some examples of constrained problems?

Reaching into deep shelf cubby holes, confined work cells, dual arms, robots that must quasi-statically balance themselves on two legs.