Randomness in search
Rapidly-exploring Random Trees

- Break up C-space into Voronoi regions around random landmarks
- Invariant: landmarks always form a tree
  - known path to root
- Subject to this requirement, placed in a way that tends to split large Voronoi regions
  - coarse-to-fine search
- Goal: feasibility not optimality (*)
RRT: required subroutines

- **RANDOM_CONFIG**
  - samples from C-space

- **EXTEND**(q, q’)
  - local controller, heads toward q’ from q
  - stops before hitting obstacle (and perhaps also after bound on time or distance)

- **FIND_NEAREST**(q, Q)
  - searches current tree Q for point near q
Path Planning with RRTs

RRT = Rapidly-Exploring Random Tree

BUILT_RRT(q_{init}) {
   T = q_{init}
   for k = 1 to K {
      q_{rand} = RANDOM_CONFIG()
      q_{new} = EXTEND(q_{near}, q_{rand})
      T = T + (q_{near}, q_{new})
   }
}

EXTEND(T, q) {
   q_{near} = FIND_NEAREST(q, T)
   q_{new} = EXTEND(q_{near}, q)
   T = T + (q_{near}, q_{new})
}
Path Planning with RRTs

RRT = Rapidly-Exploring Random Tree

BUILT_RRT(q_{init}) {
    T = q_{init}
    for k = 1 to K {
        q_{rand} = RANDOM_CONFIG()
        q_{near} = FIND_NEAREST(q_{rand}, T)
        q_{new} = EXTEND(q_{near}, q_{rand})
        T = T + (q_{near}, q_{new})
    }
}

EXTEND(T, q) {
    q_{near} = FIND_NEAREST(q, T)
    q_{new} = EXTEND(q_{near}, q)
    T = T + (q_{near}, q_{new})
}

[ Kuffner & LaValle, ICRA’00]
Path Planning with RRTs

RRT = Rapidly-Exploring Random Tree

BUILT_RRT(q_init) {
    T = q_init
    for k = 1 to K {
        q_rand = RANDOM_CONFIG()
        q_near = FIND_NEAREST(q, T)
        EXTEND(T, q_rand);
        q_new = EXTEND(q_near, q)
        T = T + (q_near, q_new)
    }
}

EXTEND(T, q) {
    q_near = FIND_NEAREST(q, T)
    q_new = EXTEND(q_near, q)
    T = T + (q_near, q_new)
}
Path Planning with RRTs

RRT = Rapidly-Exploring Random Tree

\[
\text{BUILT}_\text{RRT}(q_{\text{init}}) \{ \\
\quad T = q_{\text{init}} \\
\quad \text{for } k = 1 \text{ to } K \{ \\
\quad \quad q_{\text{rand}} = \text{RANDOM\_CONFIG()} \\
\quad \quad \text{EXTEND}(T, q_{\text{rand}}); \\
\quad \}\}
\]

\[
\text{EXTEND}(T, q) \{ \\
\quad q_{\text{near}} = \text{FIND\_NEAREST}(q, T) \\
\quad q_{\text{new}} = \text{EXTEND}(q_{\text{near}}, q) \\
\quad T = T + (q_{\text{near}}, q_{\text{new}}) \\
\}\}
\]

[Kuffner & LaValle, ICRA'00]
Path Planning with RRTs

RRT = Rapidly-Exploring Random Tree

\[ q_{rand} \]

\[ q_{near} \]

\[ q_{new} \]

\[ q_{init} \]

BUILT_RRT(q_{init}) {
    \[ T = q_{init} \]
    for \( k = 1 \) to \( K \) {
        \[ q_{rand} = \text{RANDOM\_CONFIG()} \]
        \[ q_{near} = \text{FIND\_NEAREST}(q, T) \]
        \[ q_{new} = \text{EXTEND}(q_{near}, q) \]
        \[ T = T + (q_{near}, q_{new}) \]
    }
}

EXTEND(T, q) {
    \[ q_{near} = \text{FIND\_NEAREST}(q, T) \]
    \[ q_{new} = \text{EXTEND}(q_{near}, q) \]
    \[ T = T + (q_{near}, q_{new}) \]
}

[Kuffner & LaValle, ICRA’00]
RRT example

Planar holonomic robot
RRTs explore coarse to fine

- Tend to break up large Voronoi regions
  - higher probability of $q_{\text{rand}}$ being in them
- Limiting distribution of vertices given by $\text{RANDOM\_CONFIG}$
  - as RRT grows, probability that $q_{\text{rand}}$ is reachable with local controller (and so immediately becomes a new vertex) approaches 1
RRT example
RRT for a car (3 dof)
Planning with RRTs

- Build RRT from start until we add a node that can reach goal using local controller
- (Unique) path: root → last node → goal
- Optional: “rewire” tree during growth by testing connectivity to more than just closest node
- Optional: grow forward and backward
Probability
Probability

- Random variables
- Atomic events
- Sample space
Probability

- Events
- Combining events
Probability

- Measure:
  - disjoint union:
  - e.g.:
  - interpretation:

- Distribution:
  - interpretation:
  - e.g.:
### Example

<table>
<thead>
<tr>
<th>Weather</th>
<th>AAPL price</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun</td>
<td>up</td>
</tr>
<tr>
<td></td>
<td>0.09</td>
</tr>
<tr>
<td>rain</td>
<td>0.21</td>
</tr>
</tbody>
</table>
### Bigger example

<table>
<thead>
<tr>
<th>Weather</th>
<th>AAPL price</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>up</td>
<td>same</td>
<td>dow</td>
<td></td>
</tr>
<tr>
<td><strong>PIT</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sun</td>
<td>0.03</td>
<td>0.05</td>
<td>0.02</td>
<td></td>
</tr>
<tr>
<td>rain</td>
<td>0.07</td>
<td>0.12</td>
<td>0.05</td>
<td></td>
</tr>
<tr>
<td><strong>LAX</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sun</td>
<td>0.14</td>
<td>0.23</td>
<td>0.09</td>
<td></td>
</tr>
<tr>
<td>rain</td>
<td>0.06</td>
<td>0.10</td>
<td>0.04</td>
<td></td>
</tr>
</tbody>
</table>
Notation

- $X=x$: event that r.v. $X$ is realized as value $x$
- $P(X=x)$ means probability of event $X=x$
  - if clear from context, may omit "$X=$"
  - instead of $P(\text{Weather}=\text{rain})$, just $P(\text{rain})$
  - complex events too: e.g., $P(X=x, Y\neq y)$
- $P(X)$ means a function: $x \rightarrow P(X=x)$
Functions of RVs

- Extend definition: any deterministic function of RVs is also an RV
- E.g., “profit”:

<table>
<thead>
<tr>
<th>Weather</th>
<th>AAPL price</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>up</td>
</tr>
<tr>
<td>sun</td>
<td>–11</td>
</tr>
<tr>
<td>rain</td>
<td>–11</td>
</tr>
</tbody>
</table>
Suppose we watch for 100 days and count up our observations.
Law of large numbers

(simple version)

- If we take a sample of size $N$ from distribution $P$, count up frequencies of atomic events, and normalize (divide by $N$) to get a distribution $\tilde{P}$
- Then $\tilde{P} \rightarrow P$ as $N \rightarrow \infty$
Working w/ distributions

- **Marginals** (eliminate an irrelevant RV)
- **Conditionals** (incorporate an observation)
- **Joint** (before marginalizing or conditioning)
## Marginals

<table>
<thead>
<tr>
<th>Weather</th>
<th>AAPL price</th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>up</strong></td>
<td><strong>same</strong></td>
<td><strong>down</strong></td>
<td></td>
</tr>
<tr>
<td><strong>sun</strong></td>
<td>0.09</td>
<td>0.15</td>
<td>0.06</td>
<td></td>
</tr>
<tr>
<td><strong>rain</strong></td>
<td>0.21</td>
<td>0.35</td>
<td>0.14</td>
<td></td>
</tr>
</tbody>
</table>
Law of total probability

also called “sum rule”

- Two RVs, X and Y
- Y has values $y_1, y_2, \ldots, y_k$
- $P(X) = P(X, Y=y_1) + P(X, Y=y_2) + \ldots$
Conditioning on an observation

Two steps:
- enforce consistency
- renormalize

Notation:

<table>
<thead>
<tr>
<th>Weather</th>
<th>Coin</th>
</tr>
</thead>
<tbody>
<tr>
<td>sun</td>
<td>0.15</td>
</tr>
<tr>
<td>rain</td>
<td>0.35</td>
</tr>
</tbody>
</table>
## Conditionals

The table below shows the probability of AAPL price changes under different weather conditions at different locations.

<table>
<thead>
<tr>
<th></th>
<th>up</th>
<th>same</th>
<th>down</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PIT</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sun</td>
<td>0.03</td>
<td>0.05</td>
<td>0.02</td>
</tr>
<tr>
<td>rain</td>
<td>0.07</td>
<td>0.12</td>
<td>0.05</td>
</tr>
<tr>
<td><strong>LAX</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>sun</td>
<td>0.14</td>
<td>0.23</td>
<td>0.09</td>
</tr>
<tr>
<td>rain</td>
<td>0.06</td>
<td>0.10</td>
<td>0.04</td>
</tr>
</tbody>
</table>
Conditionals

- Thought experiment: what happens if we condition on an event of zero probability?
Notation

- $P(X \mid Y)$ is a function: $x, y \rightarrow P(X=x \mid Y=y)$
- So:
  - $P(X \mid Y) P(Y)$ means the function $x, y \rightarrow$
Conditionals in literature

When you have eliminated the impossible, whatever remains, however improbable, must be the truth.

—Sir Arthur Conan Doyle, as Sherlock Holmes
Exercise