Hill Climbing

- **Simple, general idea:**
  - Start wherever
  - Always choose the best neighbor
  - If no neighbors have better scores than current, quit

- **Why can this be a terrible idea?**
  - Complete?
  - Optimal?
Hill Climbing Diagram

- Random restarts?
- Random sideways steps?
welcome!
Simulated Annealing

- Idea: Escape local maxima by allowing downhill moves
  - But make them rarer as time goes on

function \texttt{SIMULATED-ANNEALING}( \textit{problem}, \textit{schedule} ) \textbf{returns} a solution state

inputs: \textit{problem}, a problem
\hspace{1em} \textit{schedule}, a mapping from time to “temperature”

local variables: \textit{current}, a node
\hspace{1em} \textit{next}, a node
\hspace{1em} \textit{T}, a “temperature” controlling prob. of downward steps

\textit{current} \leftarrow \text{MAKE-NODE(\text{INITIAL-STATE}[\textit{problem}])}

for \textit{t} \leftarrow 1 \text{ to } \infty \text{ do}

\hspace{1em} \textit{T} \leftarrow \text{\textit{schedule}[t]}

\hspace{1em} \textbf{if } \textit{T} = 0 \text{ then return } \textit{current}

\hspace{1em} \textit{next} \leftarrow \text{a randomly selected successor of } \textit{current}

\hspace{1em} \Delta \textit{E} \leftarrow \text{\text{VALUE}[\textit{next}] - \text{VALUE}[\textit{current}]}

\hspace{1em} \textbf{if } \Delta \textit{E} > 0 \text{ then } \textit{current} \leftarrow \textit{next}

\hspace{1em} \textbf{else } \textit{current} \leftarrow \textit{next} \text{ only with probability } e^{\Delta \textit{E}/\textit{T}}
Simulated Annealing

- **Theoretical guarantee:**
  - Stationary distribution: \( p(x) \propto e^{\frac{E(x)}{kT}} \)
  - If \( T \) decreased slowly enough, will converge to optimal state!

- Is this an interesting guarantee?

- Sounds like magic, but reality is reality:
  - The more downhill steps you need to escape, the less likely you are to ever make them all in a row
  - People think hard about *ridge operators* which let you jump around the space in better ways
- IDA*
- Beam Search
- Forward vs. Backward vs. Bidirectional
Switch to “Machine Learning” by Tom Mitchell, chapter 9