

Behavior trees in robotics

Masterproject "Intelligent Robotics" SoSe2020

Mirko Hartung

Technical Aspects of Multimodal Systems

2019-06-25

1 Motivation

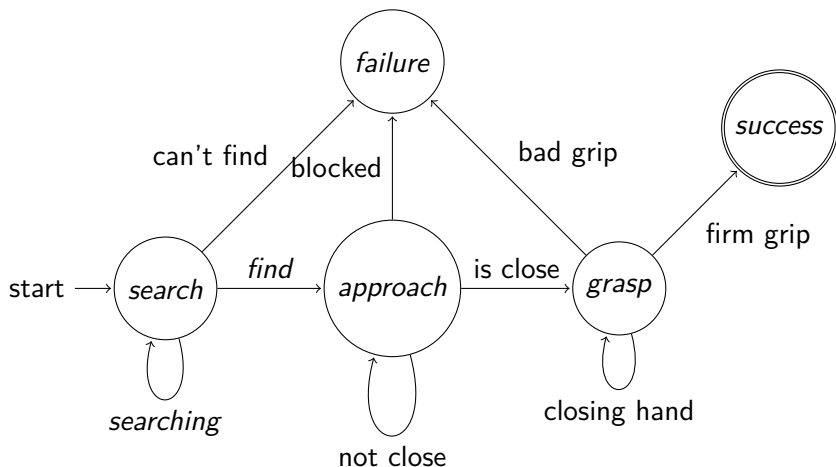
2 Introduction

3 Behavior trees in action

Autonomous robots

- Specified goal
- Execution
 - Sense
 - Plan
 - Act
 - Repeat
- Traditional approach: STRIPS

Robot grasping ball using state machine



Problems?

- Handling failure → Separate failure states needed
- Adding complexity → Exponential growth of states

Why behavior trees

Advantages

- handling failure
- modularity / scalability
- graphical representation

Applications

- Videogames [**bitr**]
- Robotics

Behavior trees are not the only solution!

What is a Behavior Tree?

- Directed acyclic graph
- Types of nodes specify behavior
- Control propagates from root node using “ticks”
- Global state (blackboard)

Nodes of behavior trees

Node Types

Root

Selector

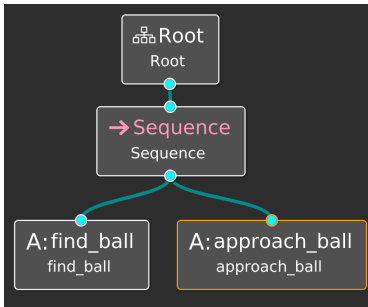
Sequence

Parallel

Decorator

Action

Condition



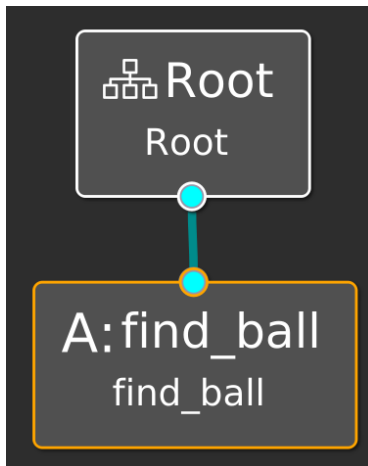
Root

- Entrypoint for execution
- Returns value of only child



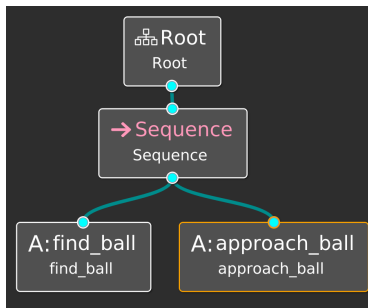
Action

- Represents the execution of tasks
- Returns success, failure or running determine control action



Sequence

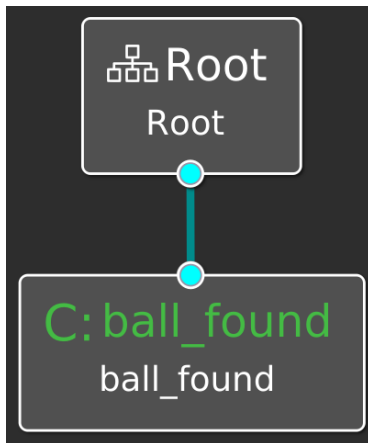
- Tick children in sequence
- Stop when running or failure is returned
- Return value of last ticked child



approach_ball will only be executed, if the action *find_ball* returns success!

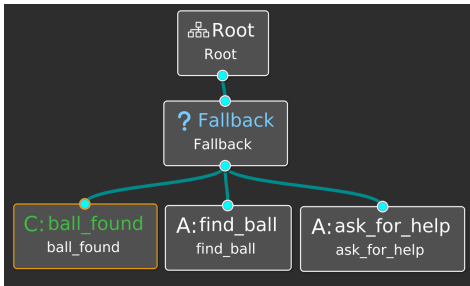
Condition

- State-space function $X_n(t)$...
 - $\dots \in S \rightarrow \textit{success}$
 - $\dots \in F \rightarrow \textit{failure}$
 - No changes to global state!

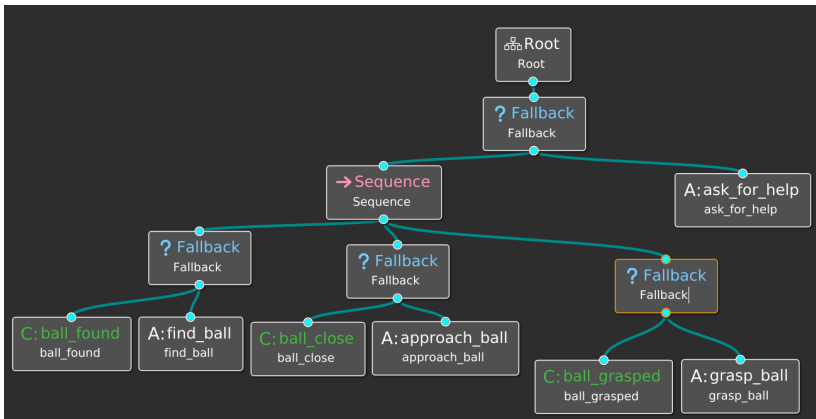


Selector

- Return, if child returns *running* or *success*
- On *failure*, tick next child



Complex example

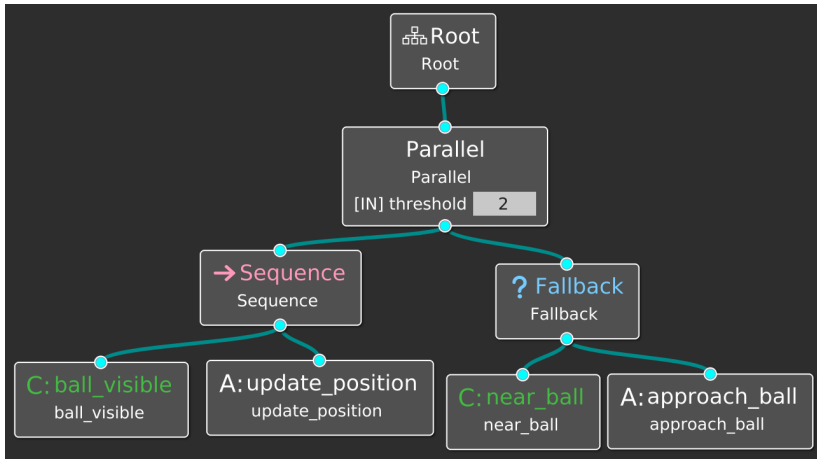


Unique selling point of BTs

Parallel

- Tick all N children
- Success, if $\geq M$ children return success
- Failure, $N - M + 1$ return failure
- Running, else

Example: Why is parallel useful



Behavior trees in action

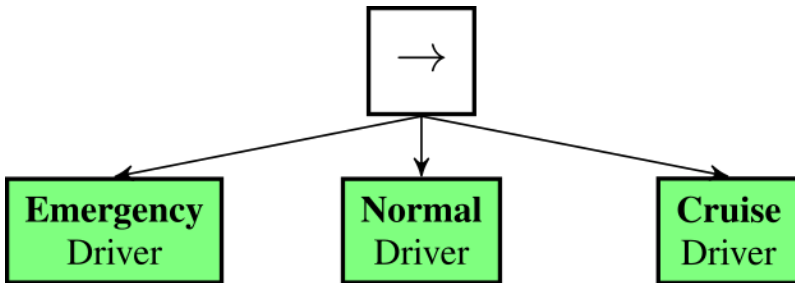


Figure: [ubt-framework]

Root-node omitted

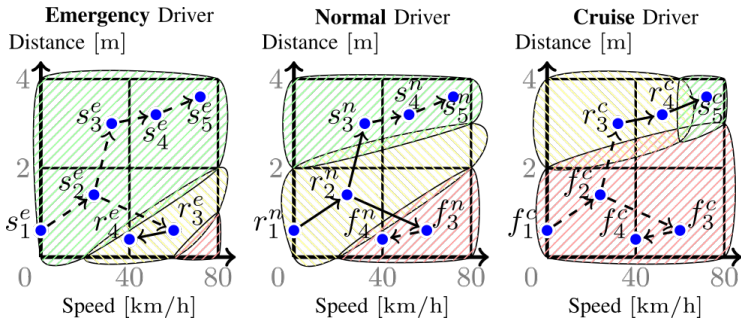


Figure: [ubt-framework]

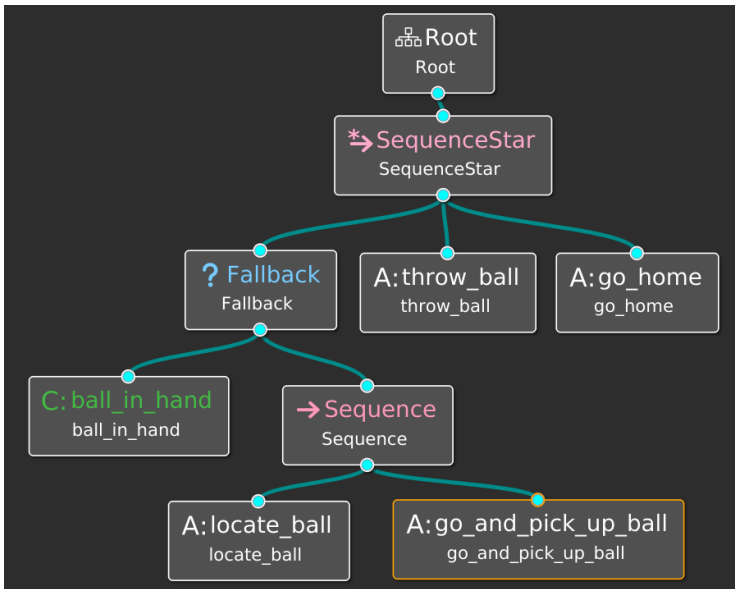
Star-nodes

Problem

- Ticking already completed nodes is ...
 - ... unnecessary
 - ... harmful
 - Walking to wall, then turn right. Condition *infront_of_wall* is invalidated by later actions

Solution

- Remember last node which returned terminal state



Thank you for your attention!

