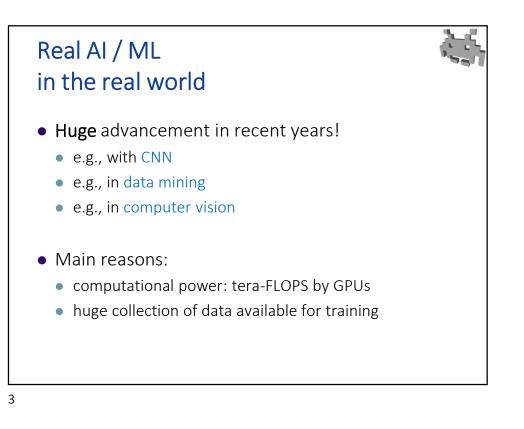
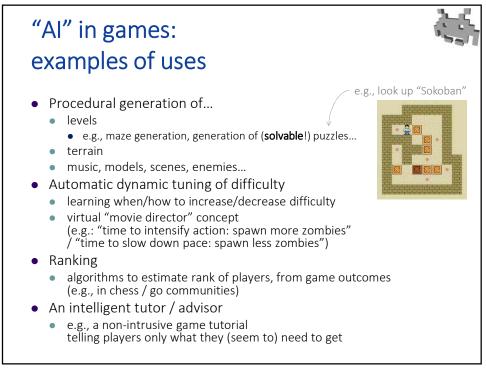
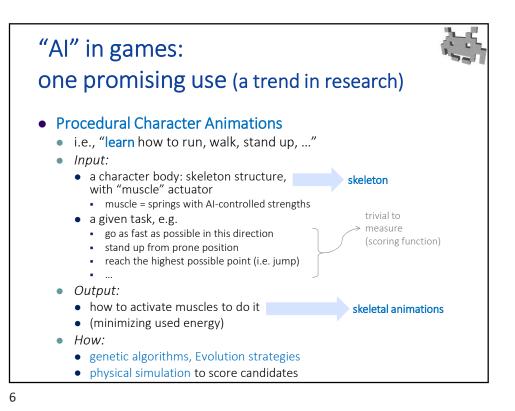


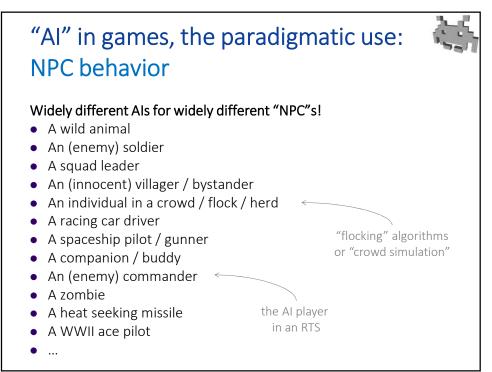
Marco Tarini Università degli Studi di Milano

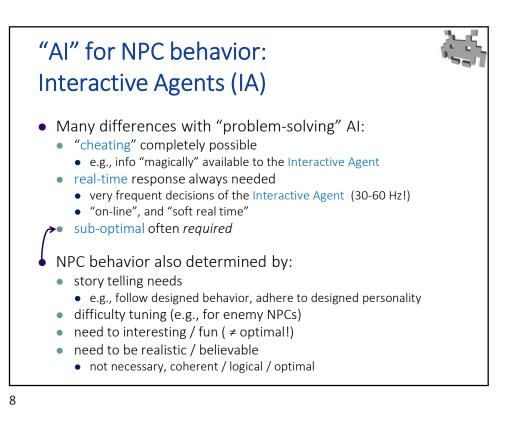




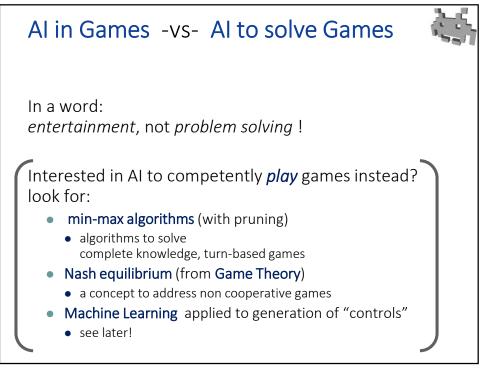


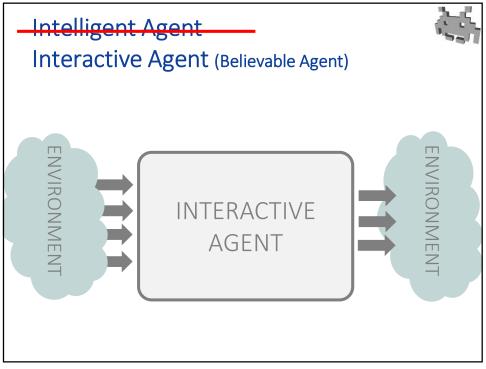


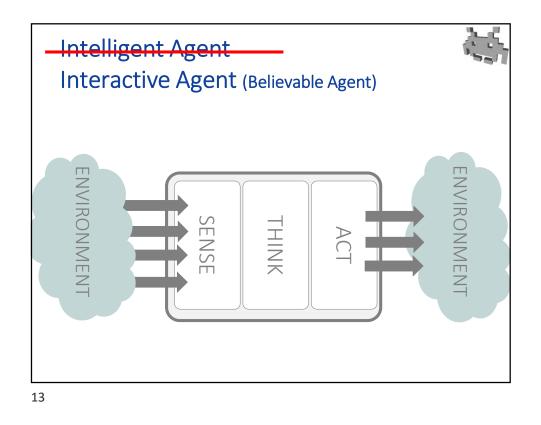


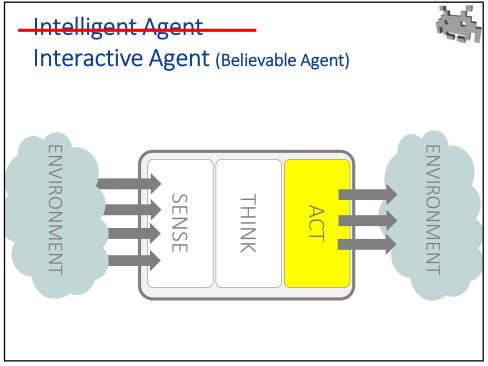


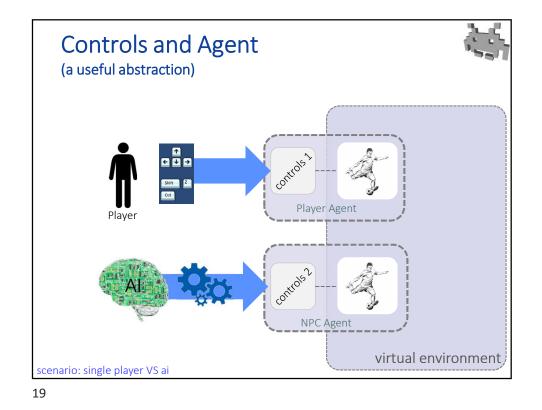
NPC behavior: a common example designer perspective of a very simple, but predicable / exploitable NPC behavior is not necessarily NPC behavior "intelligent" / proficient ("") nor even complex! good at what it does Rather, we may want NPC behavior to be (examples): intuitable / predictable think learnable (by players) < boss battles understandable story-driven exploitable (interesting to exploit) realistic / believable « for immersion; similar to human-players' behavior « compare: a realistic solider (can panic), Allowing game-designers to (examples): an optimal one (cannot) tune difficulty (by controlling NPC proficiency) to fake players in MMO games elicit interesting strategies by the players (e.g. disconnected ones) make a given strategy rewarding give the illusions of competing against humans in MMO games ...

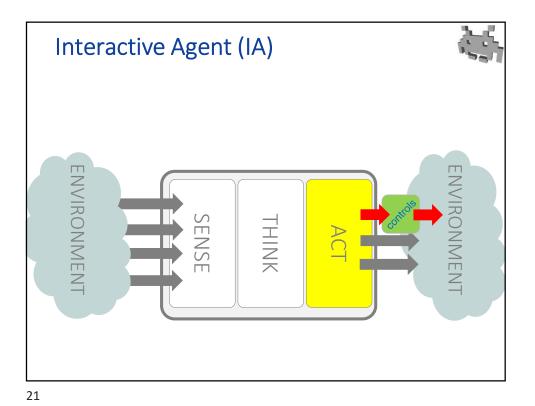




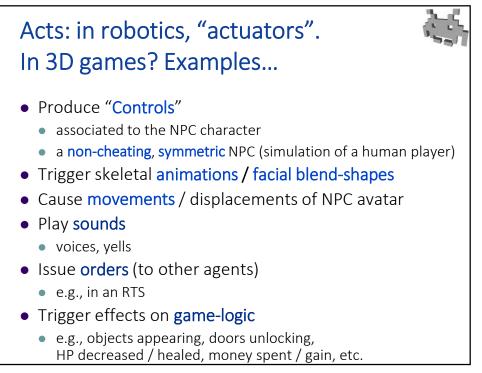


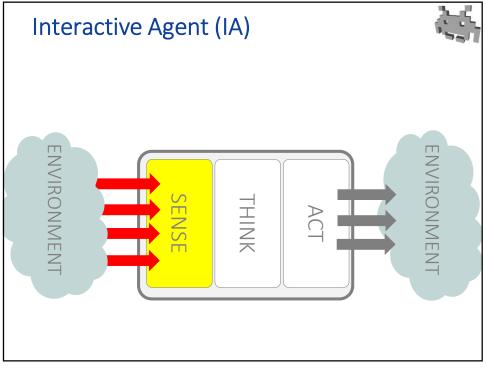


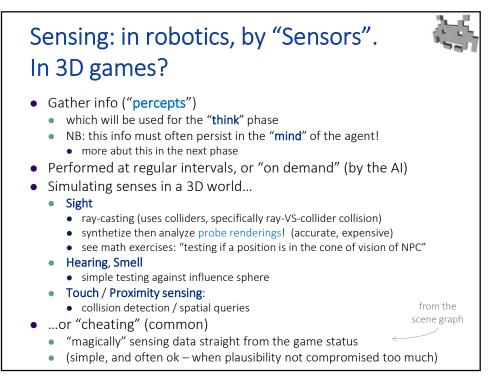


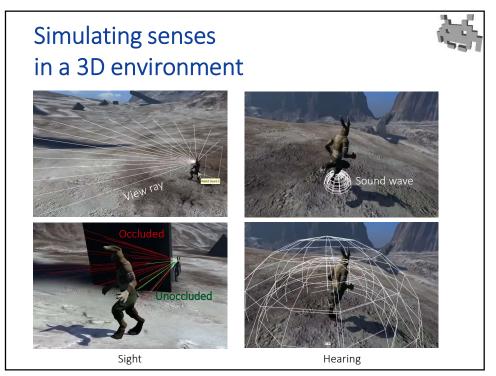


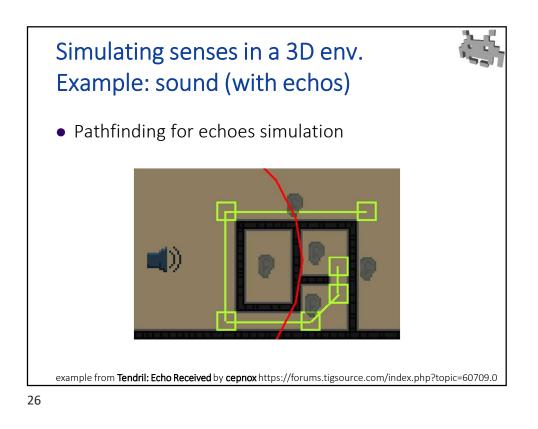
Marco Tarini Università degli Studi di Milano

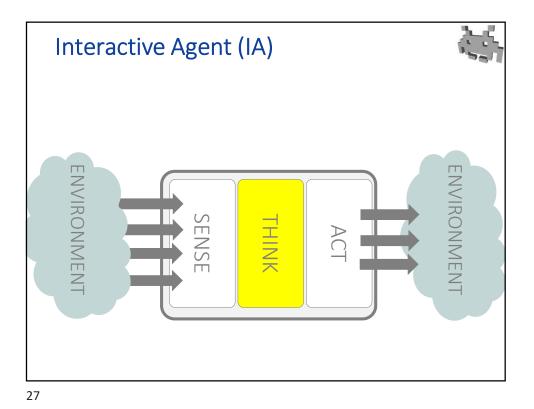


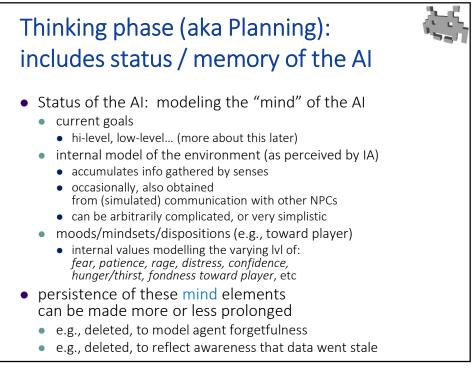


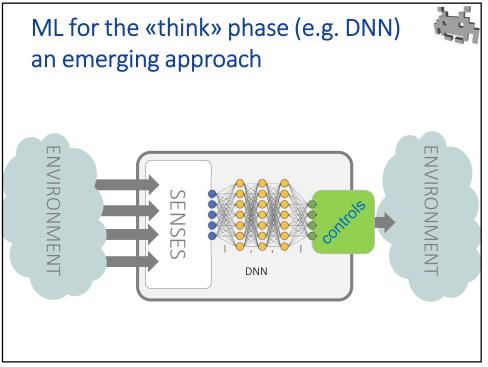


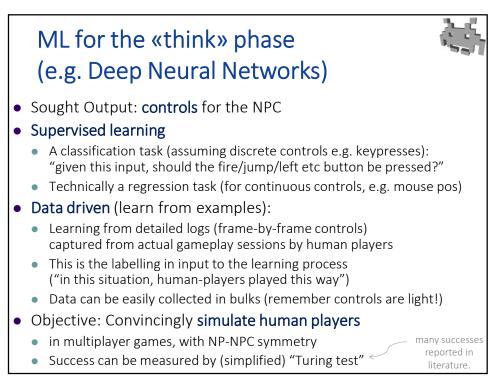


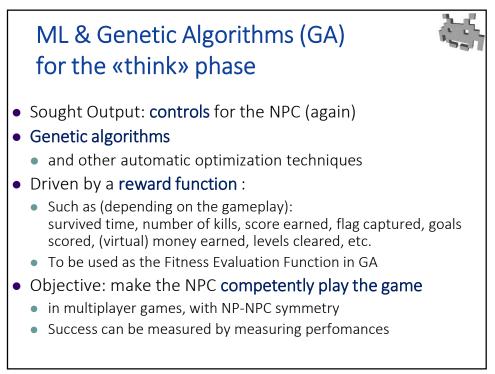


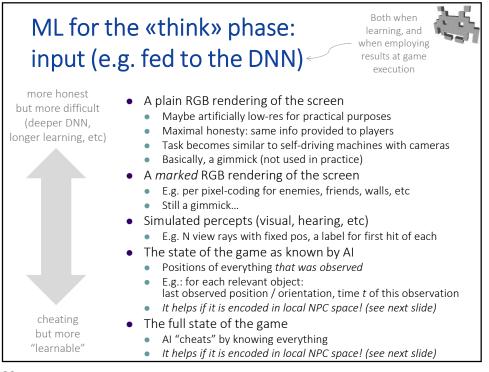


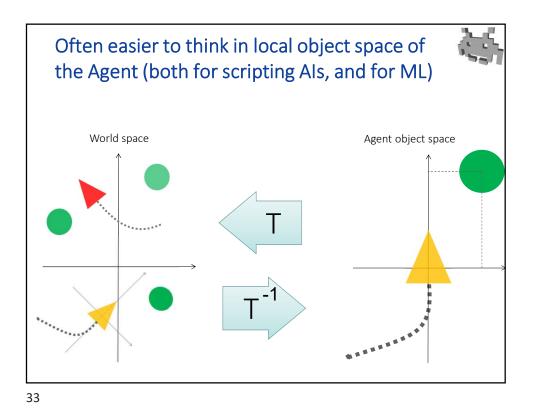


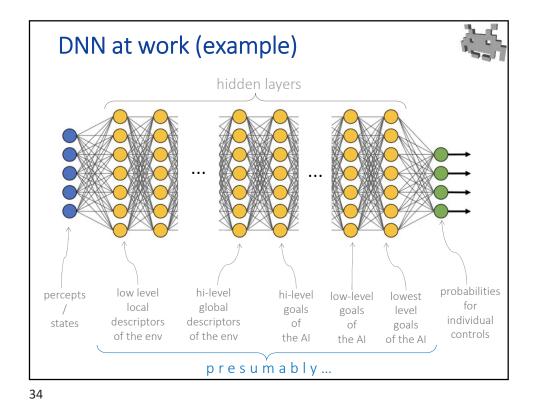


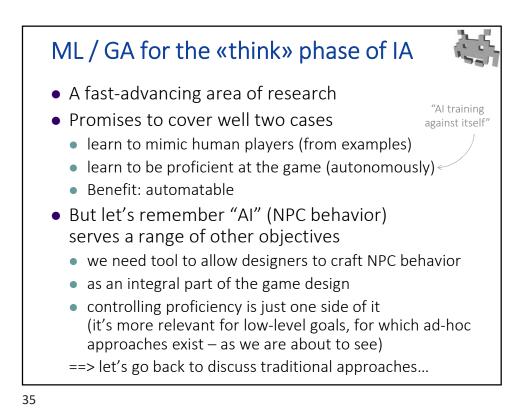


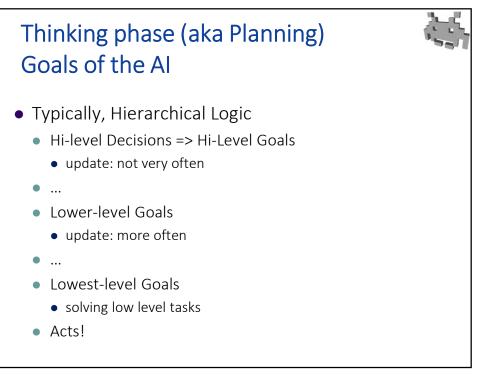


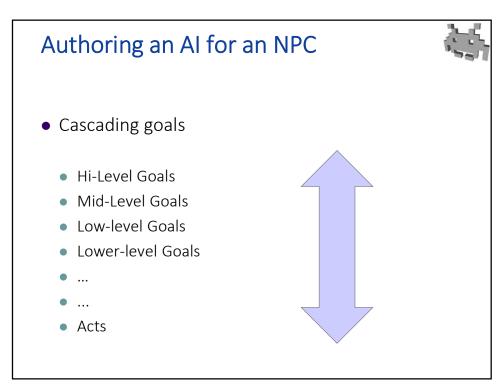


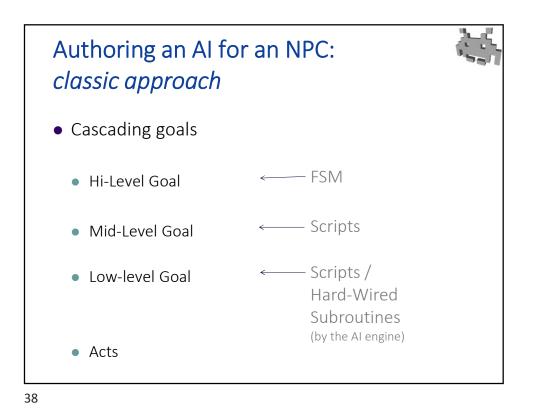


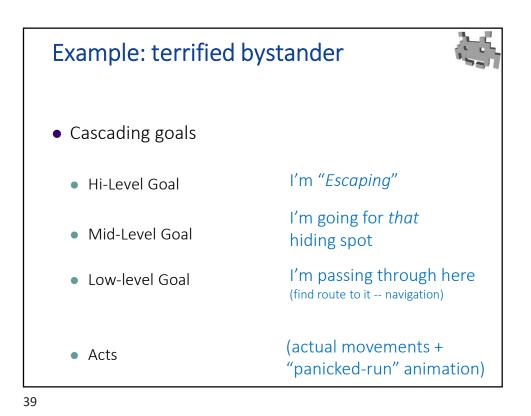


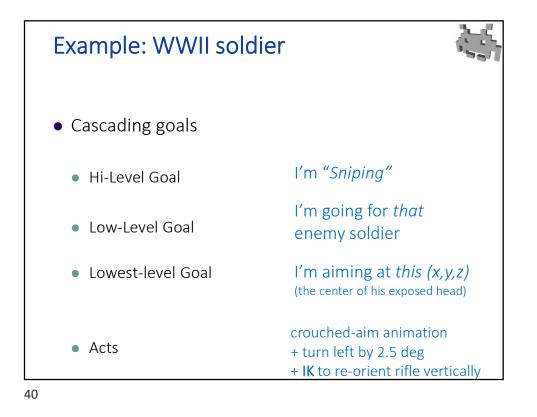


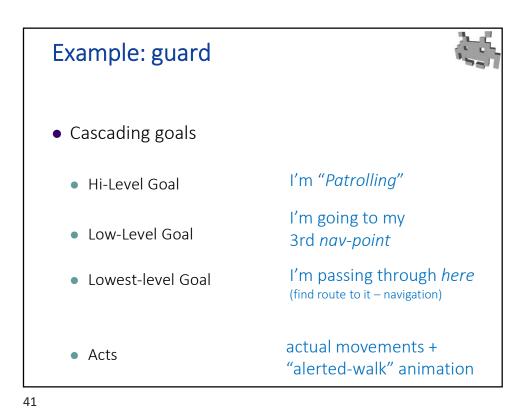


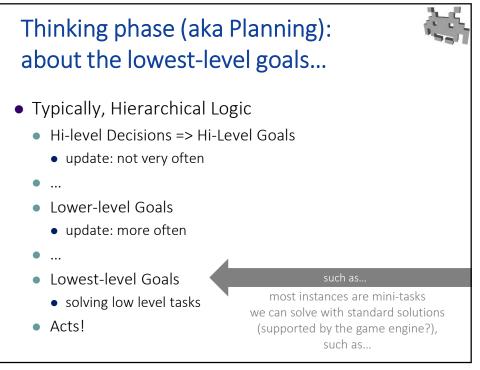


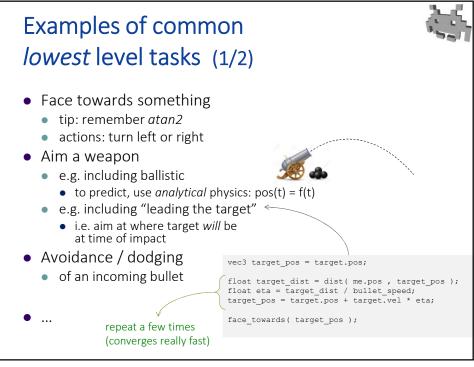


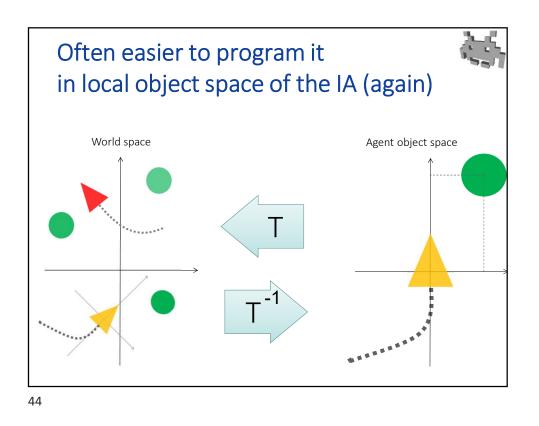


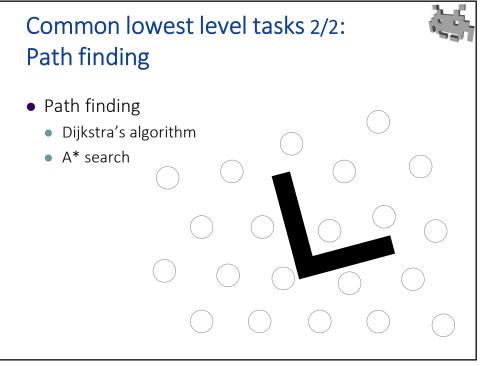


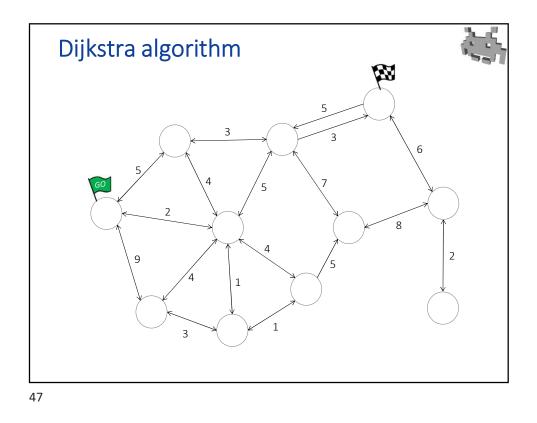


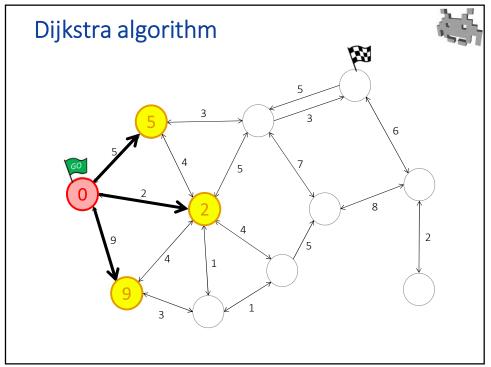


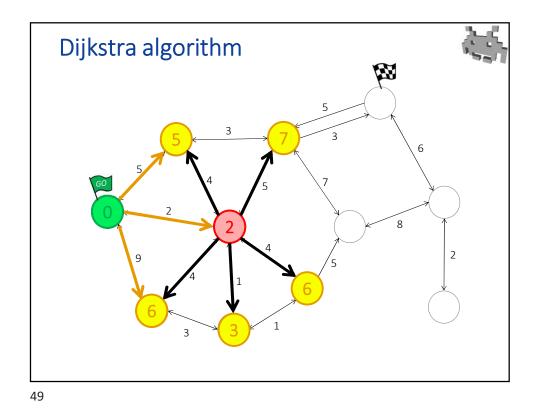


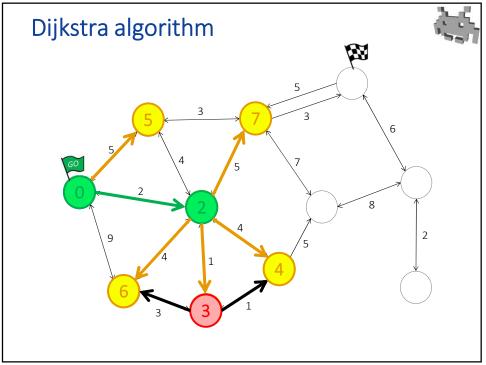


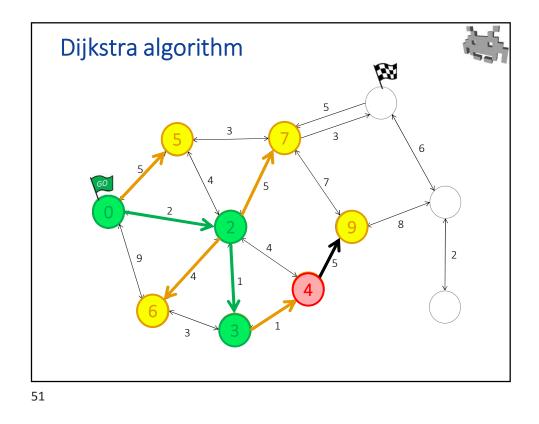


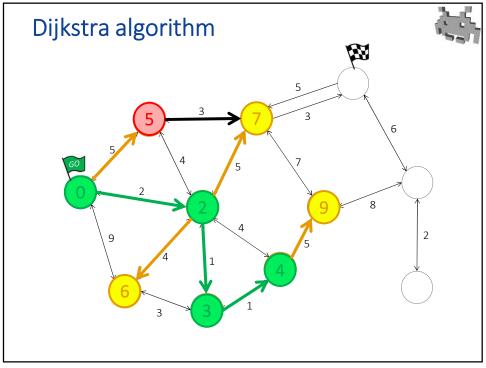


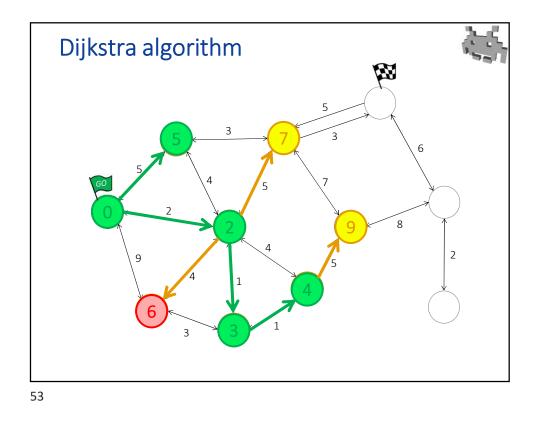


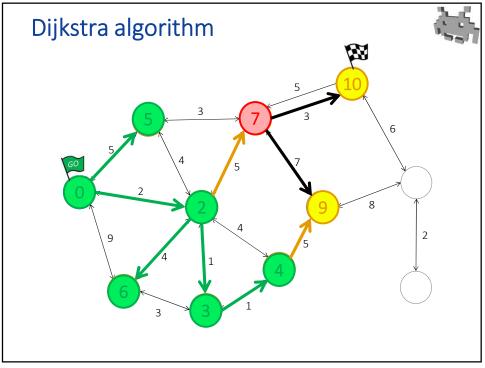


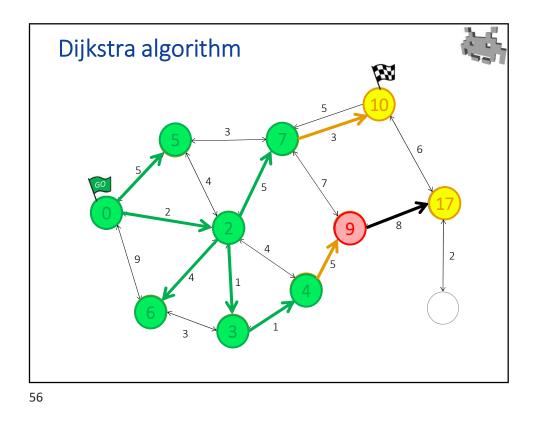


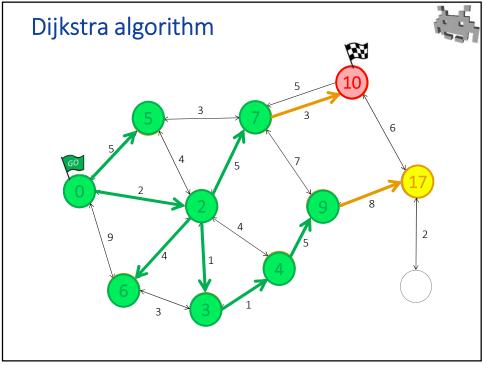


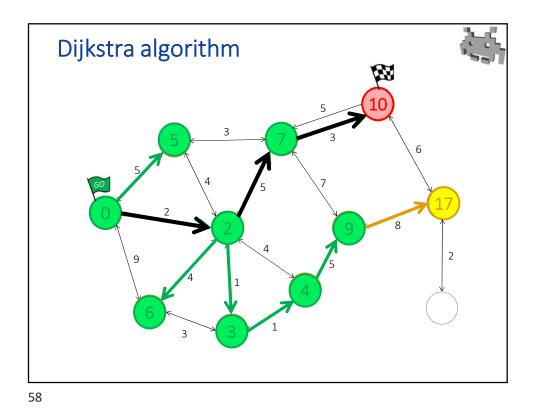








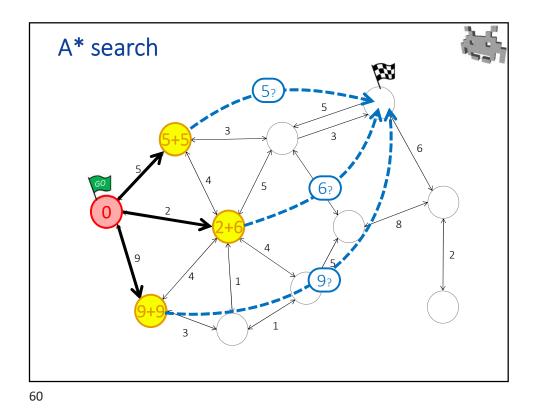


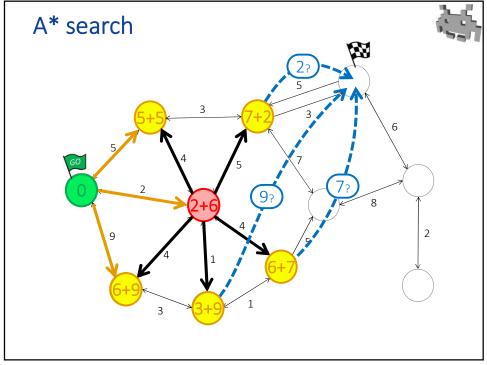


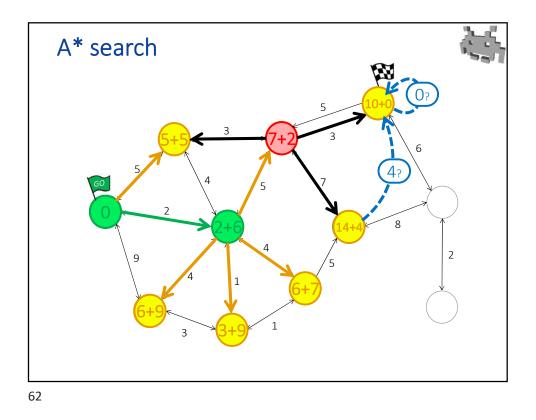
A* search

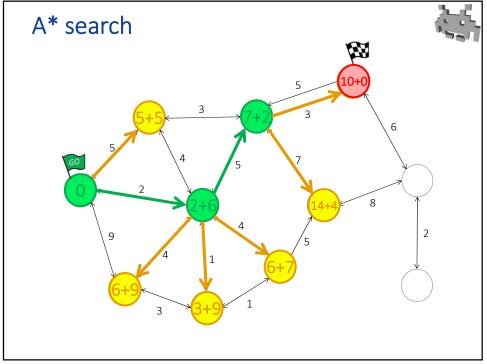
59

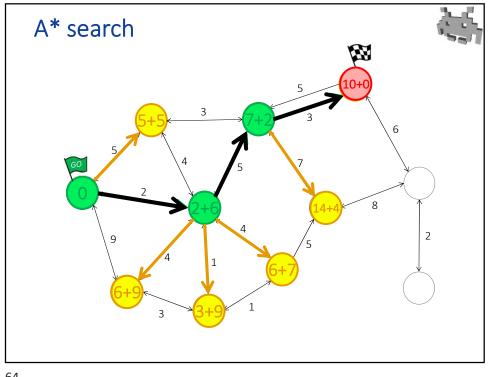
Marco Tarini Università degli Studi di Milano

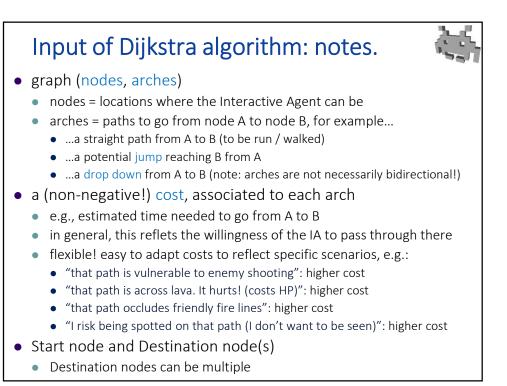


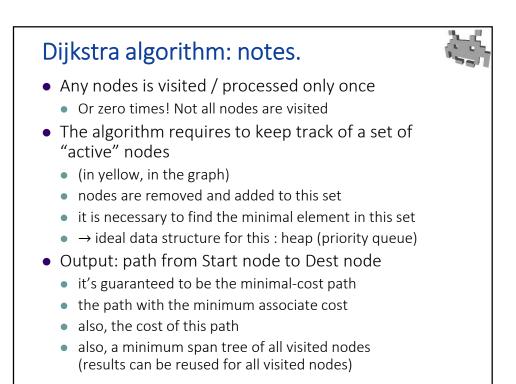


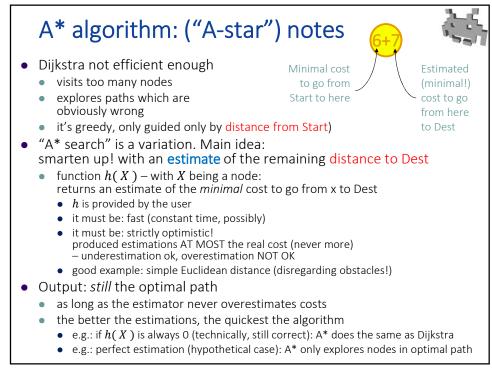




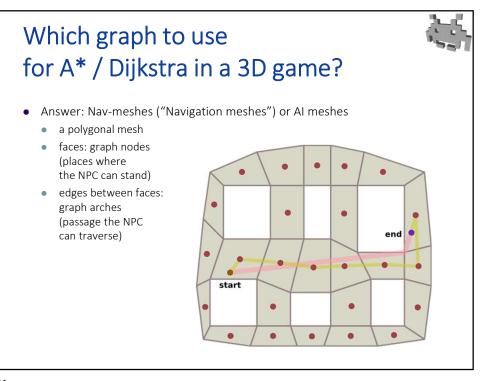


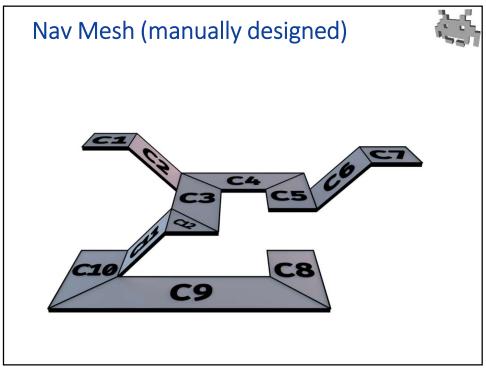


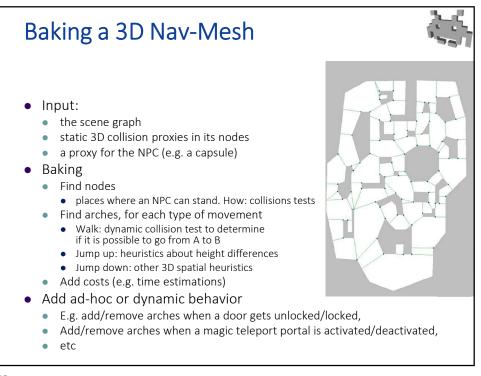


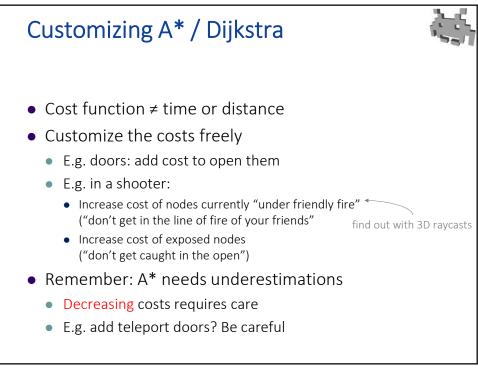


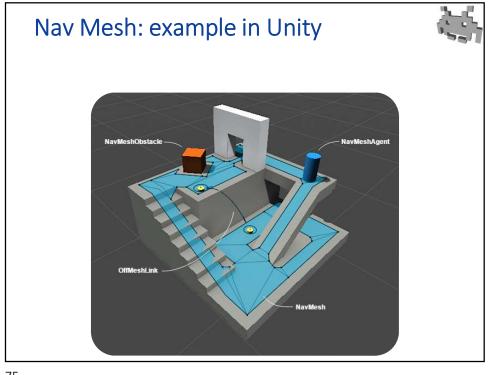




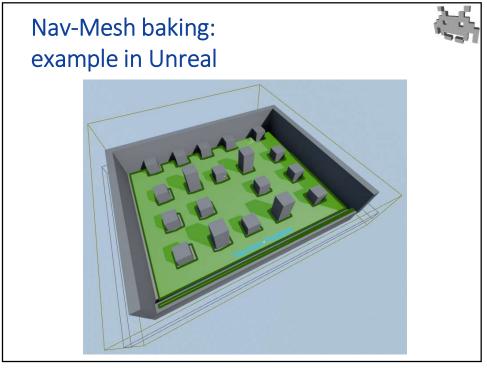


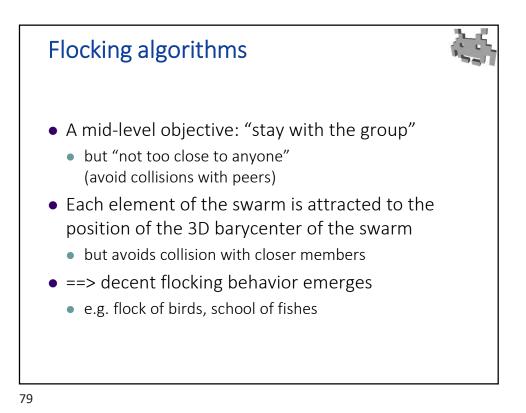


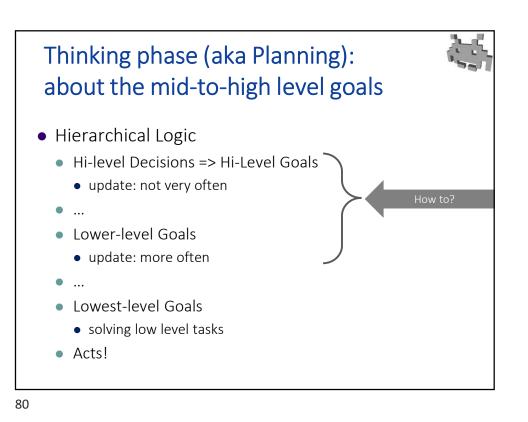


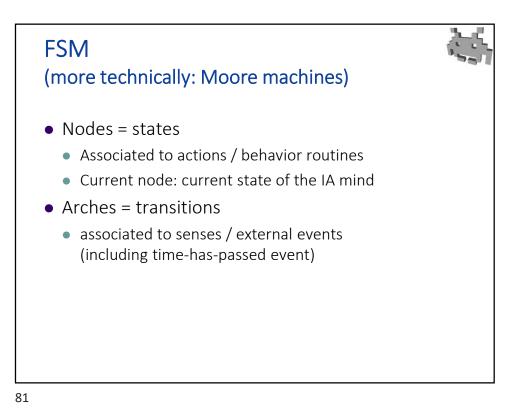


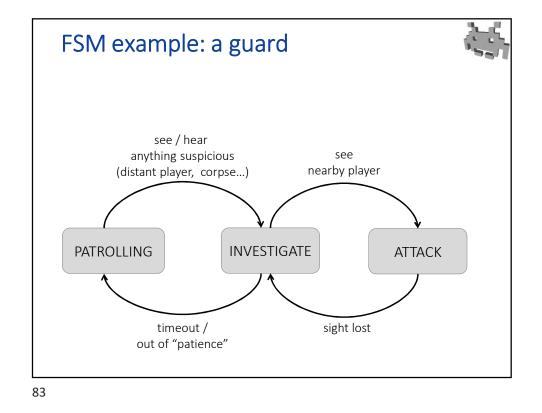






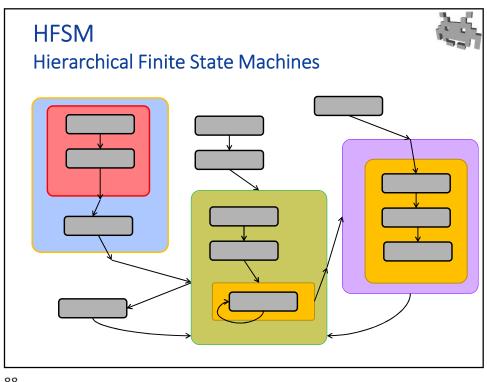


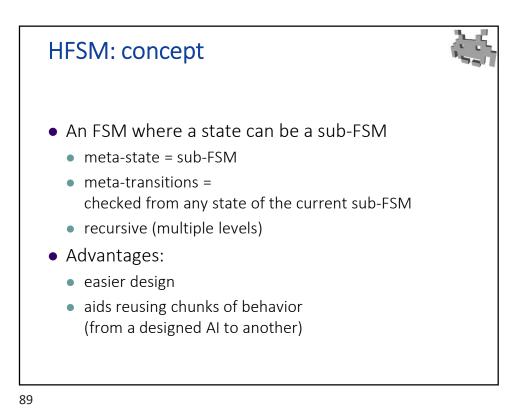


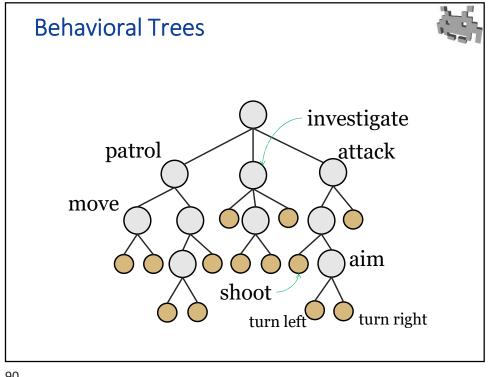


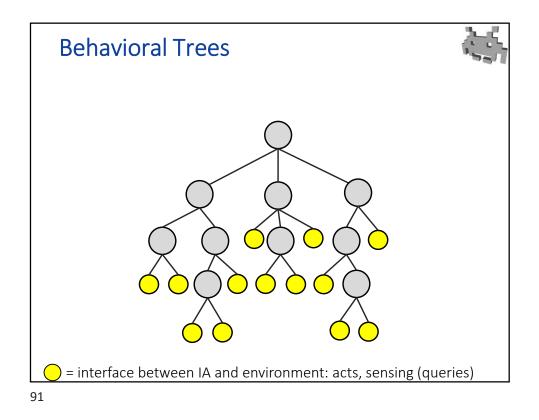
if (status==PATROLING) Implementing FSM then **doPatroling()**; if (status==ATTACK) then doAttack(); procedure doPatroling() { • FSM can serve as a coding guideline • use one "status" variable if next nav point reached . • transitions: manually coded in // state transitions if (target_in_sight)
then status = ATTACK; Or, as a behavior authoring tool • intended for the AI designer • supported by game engine • WYSIWYG editors possible • transitions: conditions (to be checked automatically) • statuses: linked to effects (sound, animation,...) Limitation: scalability (with complexity) only good for simple behavior quickly produces intricated nets Let's see alternatives: blur distinction between HFSM hi-level / low-level goals Behavioral Trees 👡 also blur classic distinction between sensing / thinking / acting

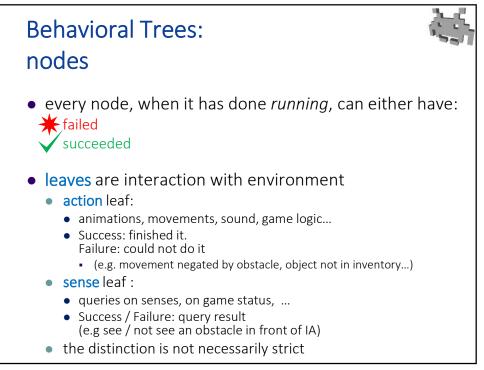


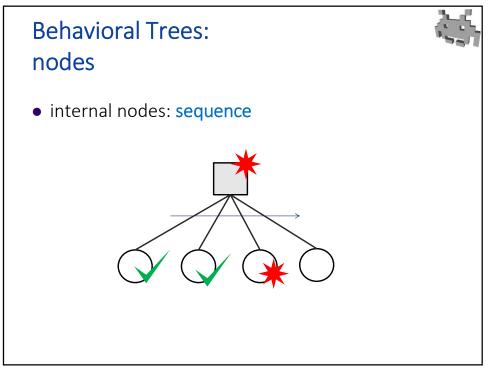


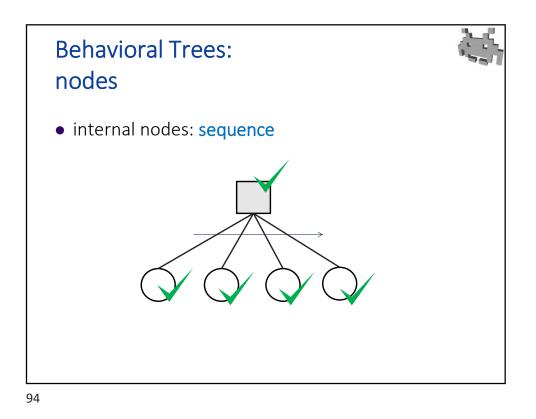












Behavioral Trees: nodes • internal nodes: selector

