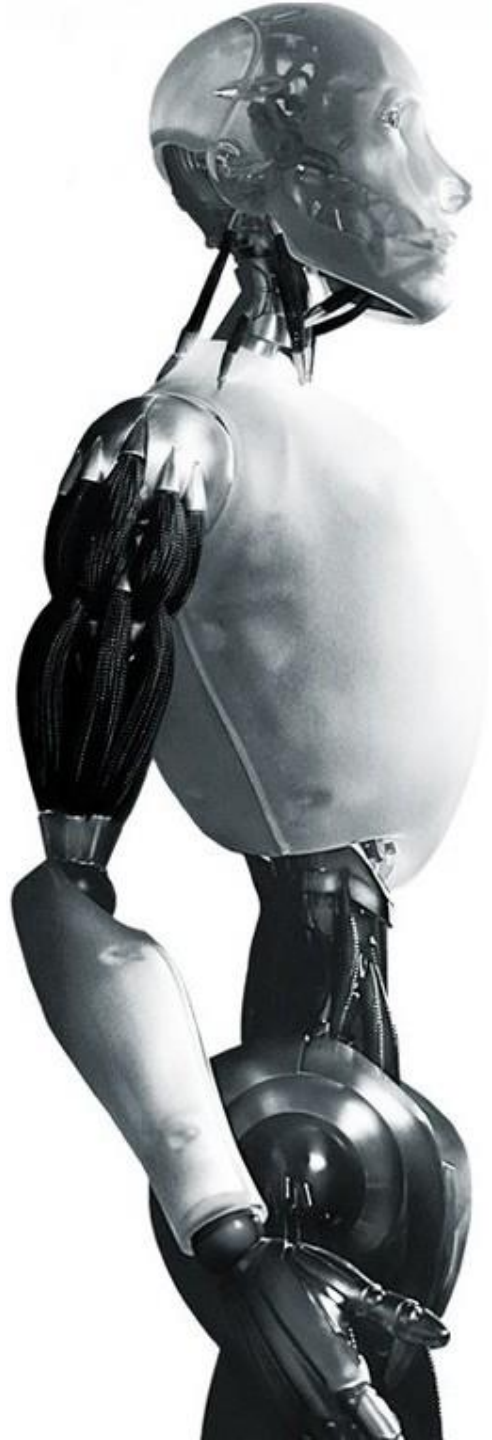
Disclaimer

These slides are intended as presentation aids for the lecture. They contain information that would otherwise be to difficult or time-consuming to reproduce on the board. But they are incomplete, not self-explanatory, and are not always used in the order they appear in this presentation. As a result, these slides should not be used as a script for this course. I recommend you take notes during class, maybe on the slides themselves. It has been shown that taking notes improves learning success.

Reading for this set of slides

- Planning Algorithms (Steve LaValle)
  - 4 The Configuration Space (4.1 – 4.3)
  - 5 Sampling-based Motion Planning (5.1, 5.5, 5.6, also skim the remaining sections)

Please note that this set of slides is intended as support for the lecture, not as a stand-alone script. If you want to study for this course, please use these slides in conjunction with the indicated chapters in the text books. The textbooks are available online or in the TUB library (many copies that can be checked out for the entire semester. There are also some aspects of the lectures that will not be covered in the text books but can still be part of the homework or exam. For those It is important that you attend class or ask somebody about what was covered in class.



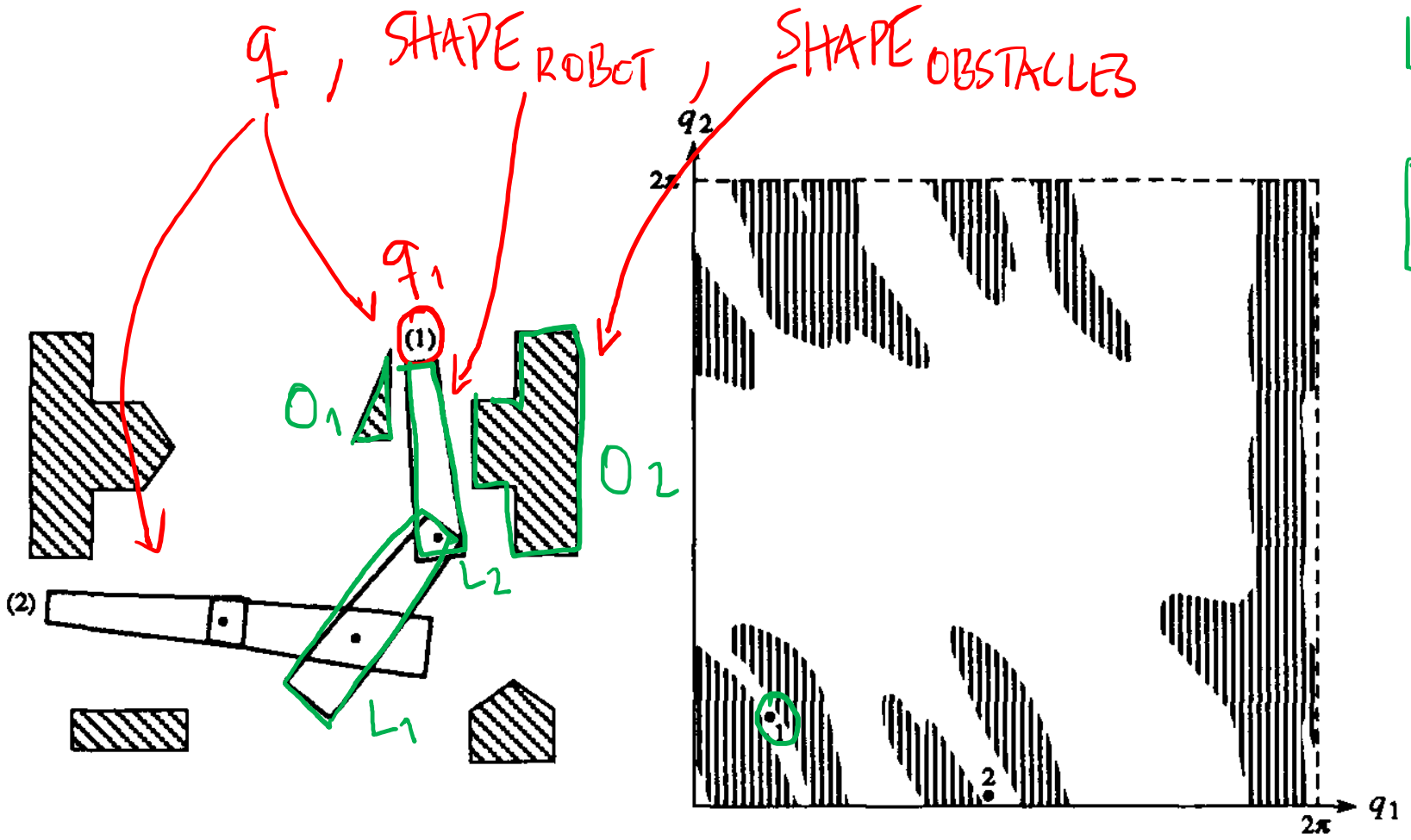
Robotics

Collision Checking

TU Berlin

Oliver Brock

Collision Checking / Distance Computation



L_1, L_2, O_1, O_2

NO NO

$L_1 \cap O_1 \vee L_1 \cap O_2 \vee$

$L_2 \cap O_1 \vee L_2 \cap O_2$

NO NO

⇝ NO!

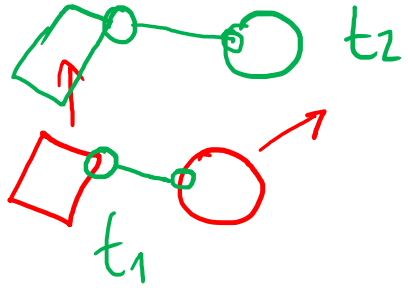
Basic Primitive: Collision Detection

- Computational complexity collision detection
  - n objects have O(n^2) interactions
  - Robot with l links and n obstacles has O($l * n$)
  - Each object has many features – 1000s!!!
  - In practice a **costly** operation

30 x 1000 = 30,000


BLUE
ROBOT
GO

Tricks for Distance Computation



- Exploit spatial coherency
  - Group primitives hierarchically
  - Exploit adjacency (on single object)
- Exploit temporal coherency
  - Exploit former relation between multiple objects
- Heuristics are good
- Problem remains computationally complex