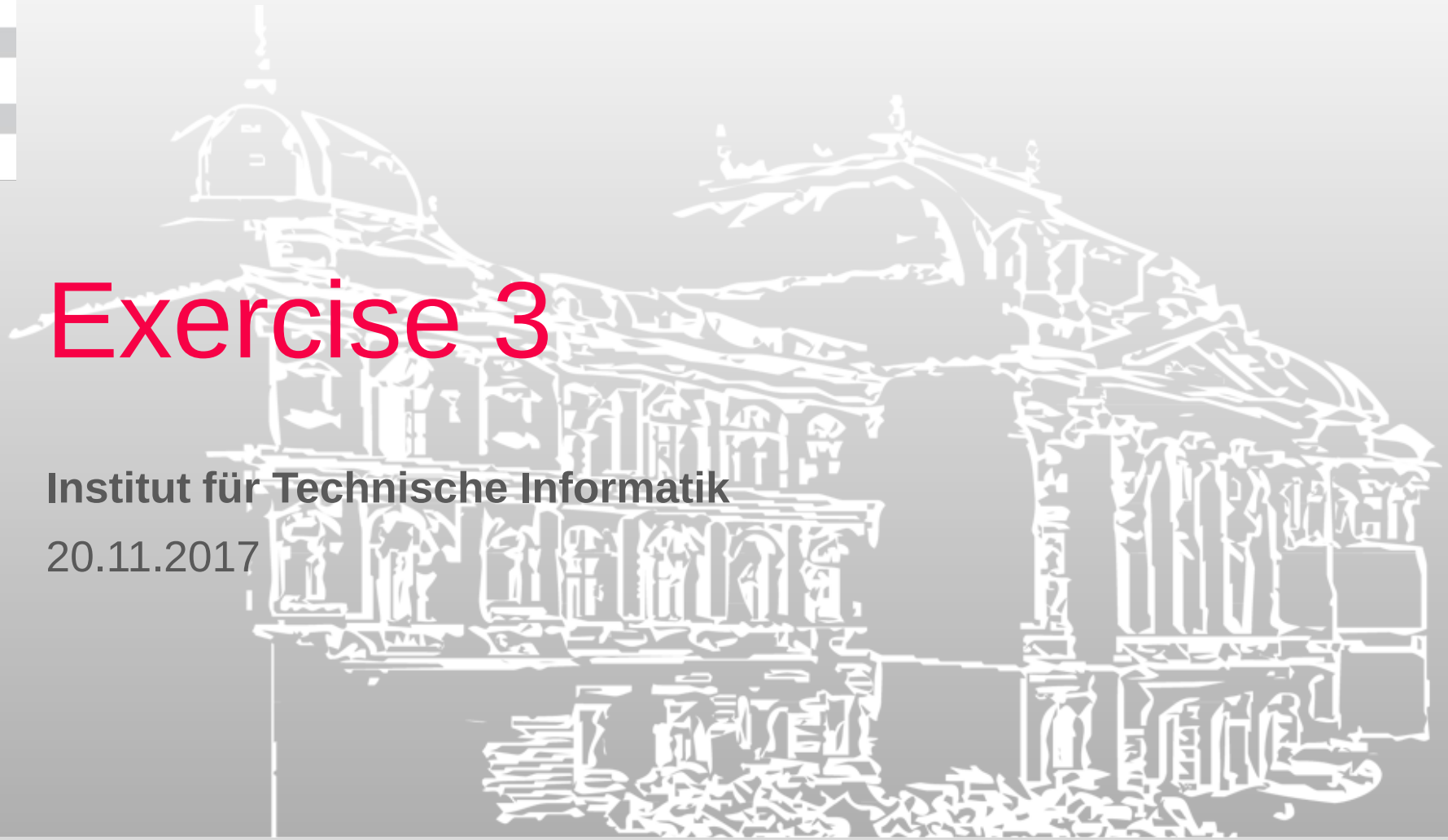


Exercise 3

Institut für Technische Informatik

20.11.2017

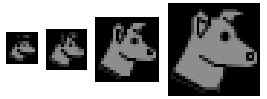
Prey vs. Predator

- Goats & Grass
  - Goats look for grass to eat
  - Goats are “blind”, i.e., they just move randomly without a specific target
  - Grass is static
- Cats & Dogs
  - Dogs look for cats in their range of sight
  - If the dog notices one or more cats, it follows one of them
  - Cats escape from dogs! If they see a dog, they run in the opposite direction
  - Let’s create a NetLogo model…

Prey vs. Predator

- Cats & Dogs
  - First step: create two different breeds

```
breed [dogs dog]
breed [cats cat]
```



  - Second step: create *k* dogs

```
to setup-dogs
   ask n-of k patches [
    sprout-dogs 1 [
      set shape "dog"
      set color blue
      set size 1
    ]
   ]
end
```

(same step applies to cats)

sprout *number* [*commands*]
sprout-<*breeds*> *number* [*commands*]

- Creates *number* new turtles on the current patch.
- The new turtles have random integer headings and color, but can immediately run *commands*

NetLogo Positioning Commands

- heading
  - Built-in turtle variable indicating the direction the turtle is facing (can be a number ≥ 0 and ≤ 359)
  - Heading 0 → turtle facing “north”, heading 90 → turtle facing “east”, ...
- towards
  - Reports the heading from the input agent to the given agent
  - If the wrapping path is shorter, it will use the wrapped path
  - The variant *towardsxy x y* will report the heading from the turtle or patch towards the point (x,y)
  - **Beware!** Asking for the heading from an agent to itself or an agent on the same location will cause a runtime error!

NetLogo Positioning Commands

- face
  - Set the caller agent's heading towards the specified agent
  - If the caller and the agent are at the same position, the caller's heading will not change
  - *facexy x y* sets the caller's heading towards the point (x,y)
- in-radius
  - Reports an agentset that includes only those agents from the original agentset whose distance from the center of the patch in which the caller is located is ≤ than the input number (it can include the calling agent itself)

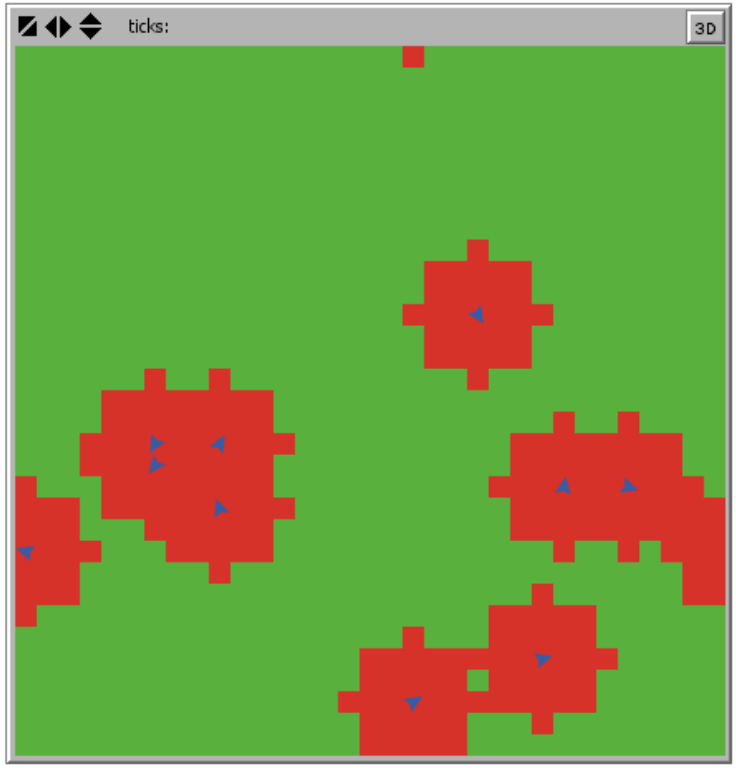
```
ask turtles  [
    ask patches in-radius 3  [
        set pcolor red
    ]
]
```

Each turtle makes a red splotch around itself

NetLogo Positioning Commands

- in-radius
  - Example

```
to example

  ask patches [
    set pcolor green
  ]

  ask n-of 10 patches [

    sprout 1 [
      set color blue
      set size 1

      ask patches in-radius 3  [
        set pcolor red
      ]

    ]

  ]
end
```



NetLogo Positioning Commands

- distance
  - Reports the distance from the current agent to the given turtle or patch
  - The distance to or from a patch is measured from the center of the patch and wrapped distances are used
  - *distancexy x y* reports the distance from the calling agent to the specified the point (x,y)

Prey vs. Predator

- Cats & Dogs
  - Third step: move dogs

```
to go
  ask cats [ move-cats ]
  ask dogs [ move-dogs ]
end
```

  - Head towards a cat

```
to move-dogs
    set heading towards target-dogs sight-dogs
    ;; move dog
    ;; eat cats
end
```

Prey vs. Predator

- Cats & Dogs
  - Find cats in the neighborhood

```
to-report target-dogs [radius]
    [   ;; Check if there is any cat in the neighborhood
        report one-of cats in-radius radius
    ]
    [
        ;; Do something else
    ]
end
```

  - Move-cats is built similarly, but cats move in the opposite direction w.r.t. dogs (rt 180)!

Prey vs. Predator

- Cats & Dogs
  - Let's imagine the following parameters:
    total dogs: 1
    total cats: 10
    sight cats: 5
    sight dogs: 10
    movement cats: 1 unit / tick
    movement dogs: 1 unit / tick
  - What will happen if we simulate this?

Adaptation in Natural Systems

- Direct communication among components of self-organizing system
  - Flock of birds, school of fishe, swarm of bees…
- Basic mechanism
  - Attraction and repulsion rules
    1. Keep minimum distance from other objects
    2. Match speed of neighbors
    3. Move towards the perceived center of mass of fish/birds/bees in the neighborhood
- Adaptation
  - Obstacle avoidance
  - Escaping predators

Adaptation in Natural Systems

- Example: Swarms of bees
  - Each bee monitors the number of bees in the neighborhood
  - Follow the direction of other bees (i.e., follow the average heading of the other bees in the neighborhood, if any)

```
to-report average_bees_heading
  report atan sum [sin heading] of other bees in-radius radius
              sum [cos heading] of other bees in-radius radius
end
```

atan x y reports the arctangent, in degrees (0-360) of x/y.

  - Let's simulate this!

School of Fish

- Fish cluster in groups
  - For hunting food (higher chances to identify food and inform the others)
  - For reproduction
  - For defense
    - A higher number of fish can identify a predator quicker
    - A predator having more targets at the same time worsens its efficiency

→ The school of fish is mainly a protection system that increases the probability of surviving to predators

School of Fish

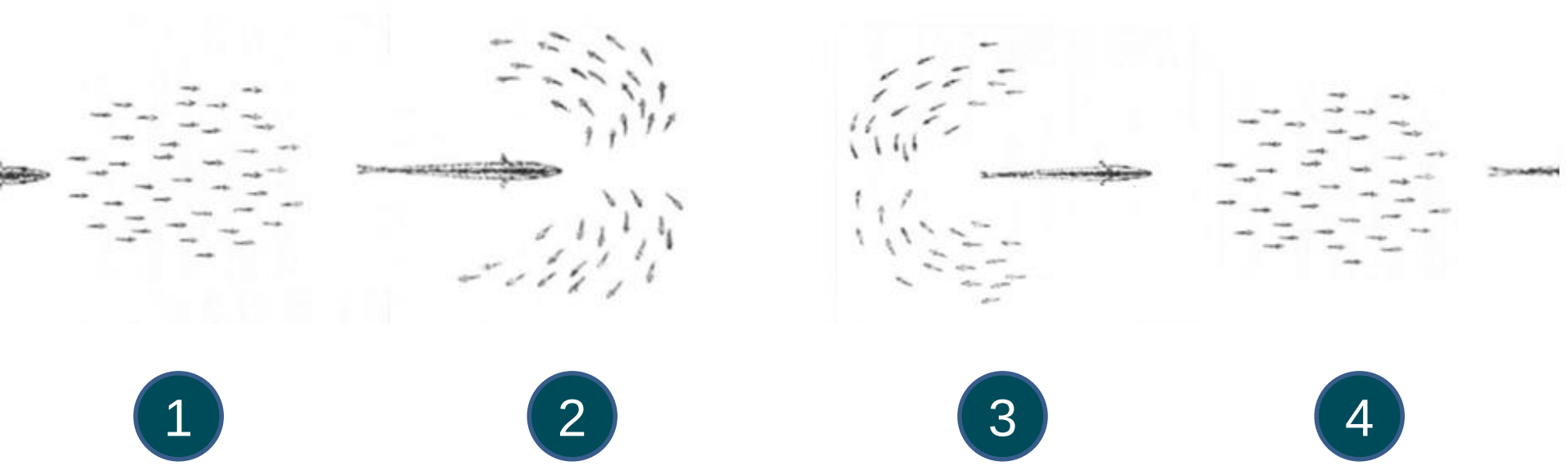
- Fish maneuver to avoid predators
  - Flash expansion

1

2

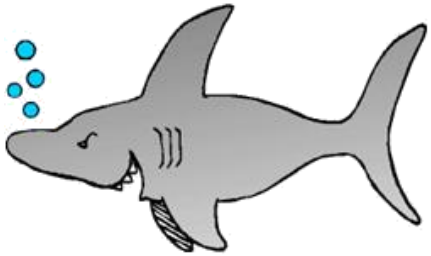
School of Fish

- Fish maneuver to avoid predators
  - Fountain effect



Exercise 3: School of Fish

- Create a NetLogo model of school of fish, to show that being part of a school actually increases the probability of survival to predators
  - Two different breeds (fish, sharks)
  - Sharks look for fish and target them
  - Fish try to escape
  - Sharks eat fish to survive (energy)



- Behavior of fishe
  - Escape in the opposite direction of the shark or follow the average heading of the school
  - Actual behavior is ruled by the variable α
  - Goal: show a school of fish generating a flash expansion

Exercise 3: School of Fish

- Explore the role of α
- α is a variable with range [0, 1]
  - Vector S: heading to escape from the shark
  - Vector F: average heading of the other fishe around
  - The final heading of the fish is H = (αS) + ((1- α)F)
- Extended task: change your model and generate also a fountain-effect behaviour of the school of fish
- Deadline is Sunday, 09.12.2018, at 23:59 CET
  - Follow the instructions carefully!

Questions?