Artificial Subjects in Psychological Experiments based on the "Socially Augmented Microworld (SAM)"

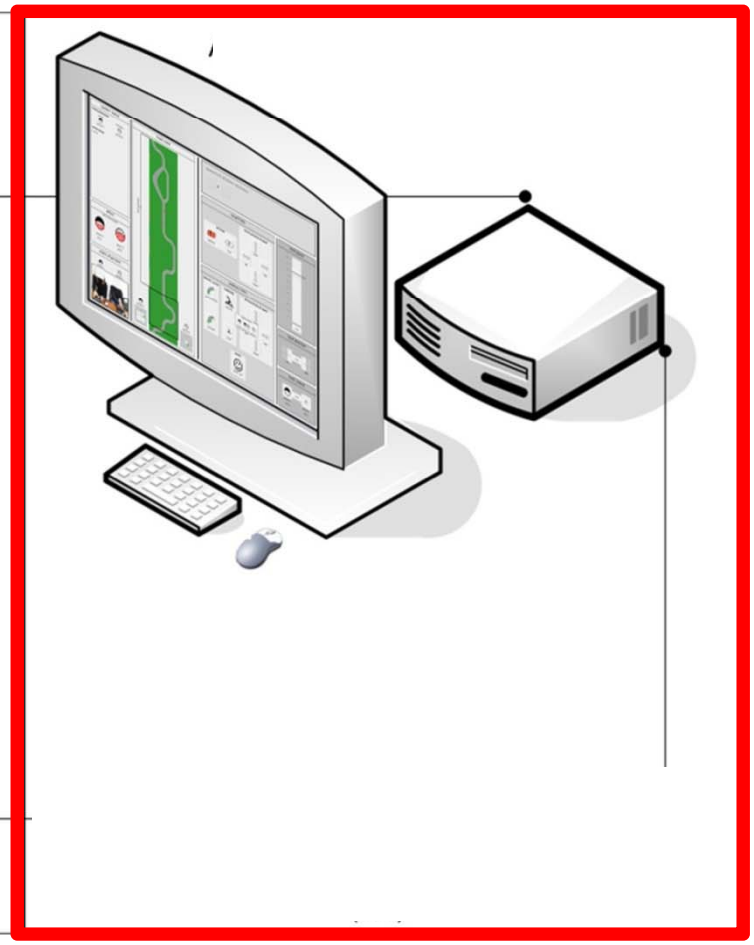
H.D. Burkhard[2)], L. Jahn[2)], C. Meyer[1)], J. Muetterlein[1)],
J. Nachtwei[1)], N. Niestroj[1)], S. Rougk[1)], M.C. Schneider[2)]

Humboldt-University Berlin
[1)]Institute of Psychology [2)]Institute of Informatics

Background ATEO and SAM

Complex **system** controlled by human **operator** via **interface**

e.g.
- Production system
- Traffic system
- Social system



Background ATEO and SAM

Complex **system** controlled by human **operator** via **interface**

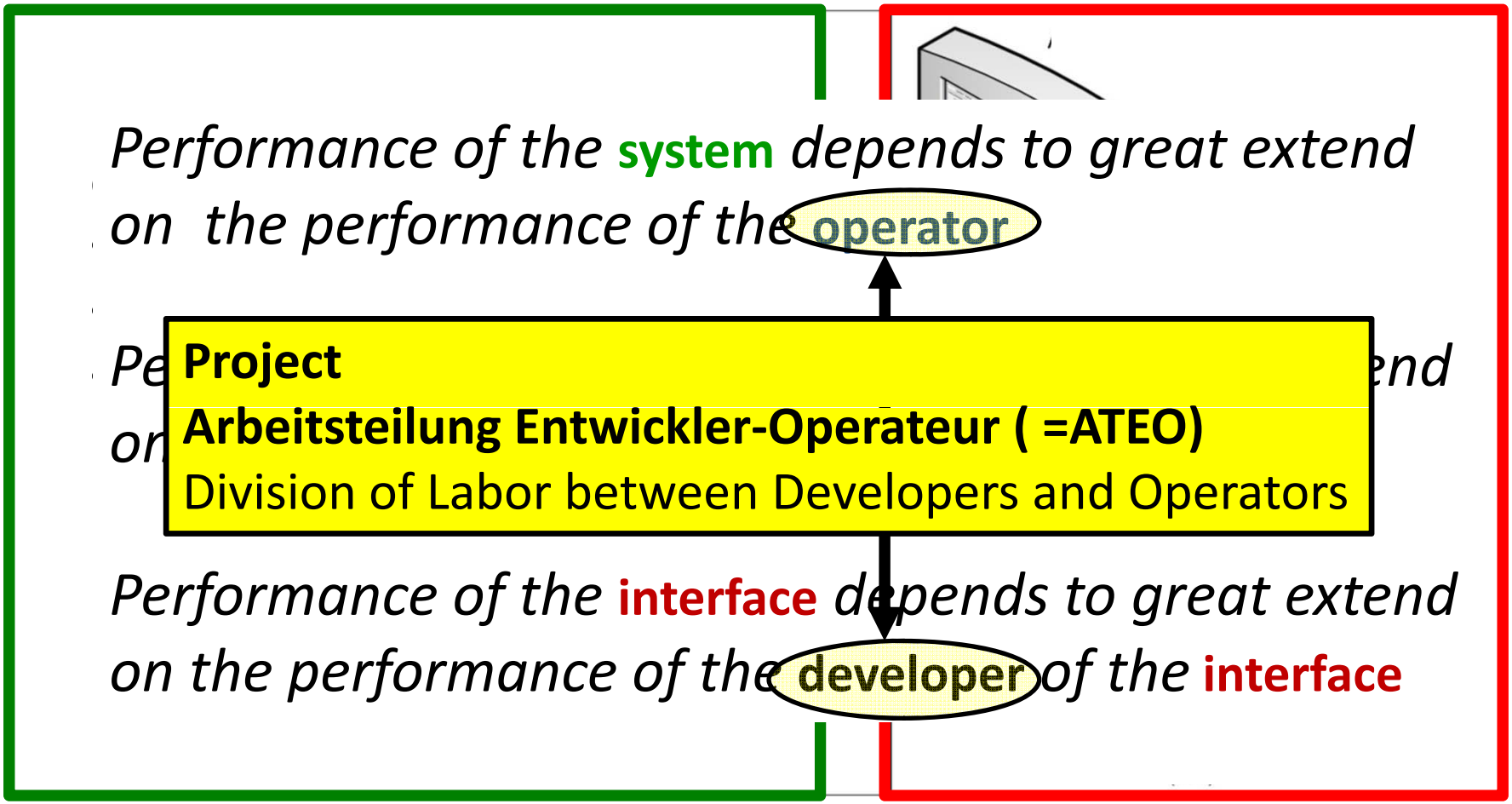
Performance of the **system** *depends to great extend on the performance of the* **operator**

Performance of the **operator** *depends to great extend on the performance of the* **interface**

Performance of the **interface** *depends to great extend on the performance of the* **developer** *of the* **interface**

Background ATEO and SAM

Complex **system** controlled by human **operator** via **interface**

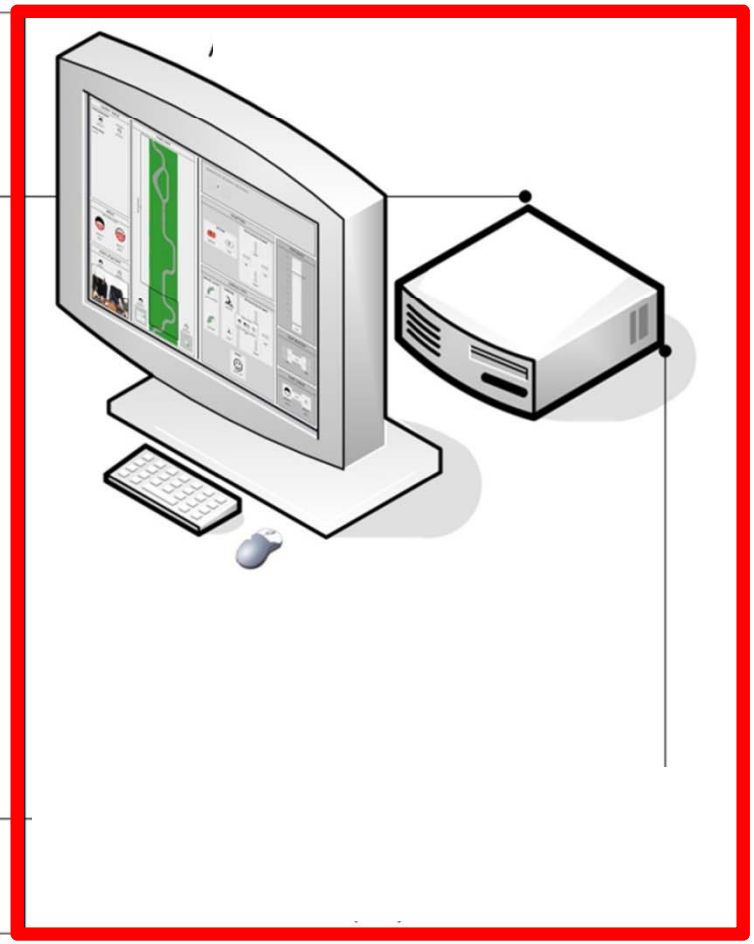


Background ATEO and SAM

Complex **system** controlled by human **operator** via **interface**

The
Socially Augmented Microworld (SAM)

SAM was designed as a testbed for experiments concerning a comparison of performance of **developers** and **operators**



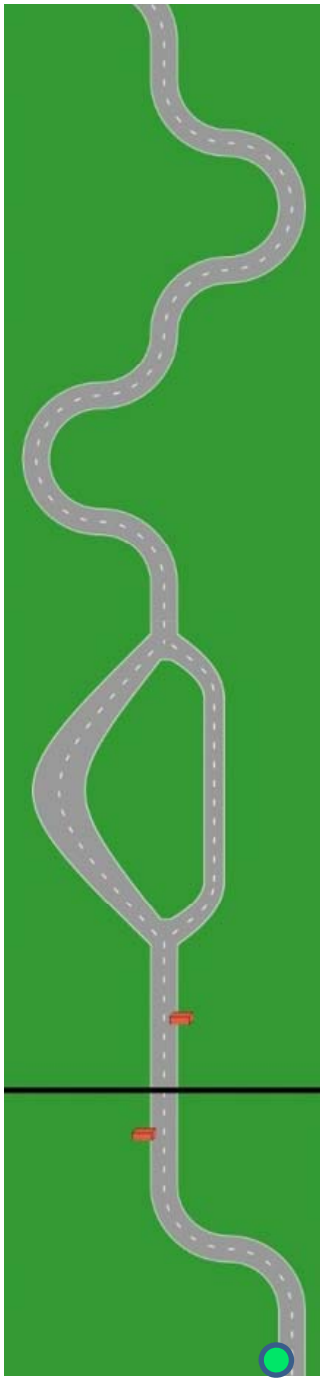
Tracking task

- Road with branches and obstacles
- Car ("**object**") controlled by **navigator** with joystick
  Conflicting goals: *fast* and *accurate*

- Visualization by bitmap with 8 colors
  - about 40000 lines with 800 pixels
  - screen with about 800 lines in front of object

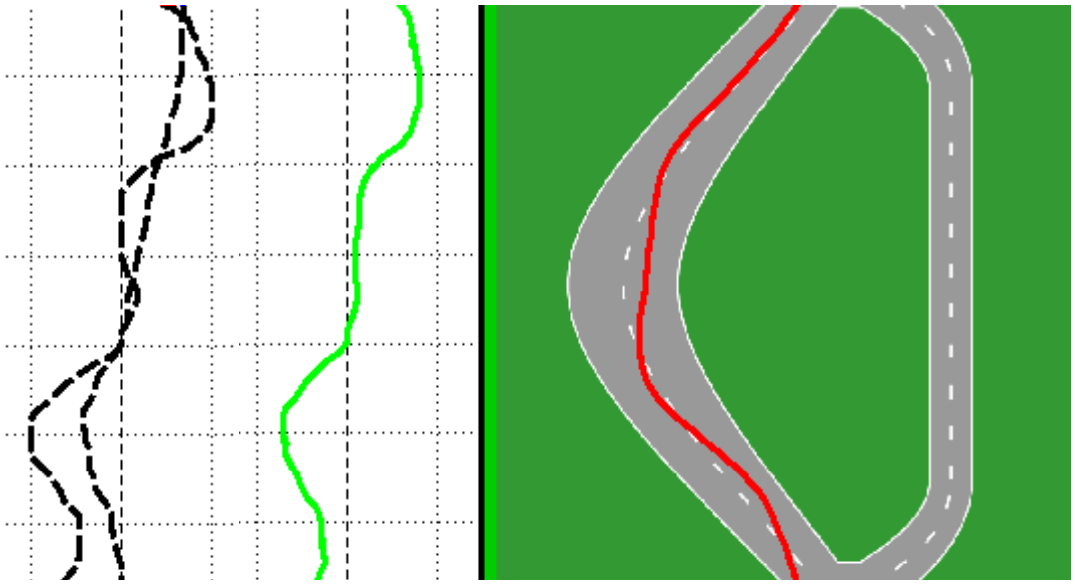
- Actualization cycle: 30 msec
  - SAM reads joystick inputs
  - SAM calculates new position of object
  - SAM presents new image on the screen

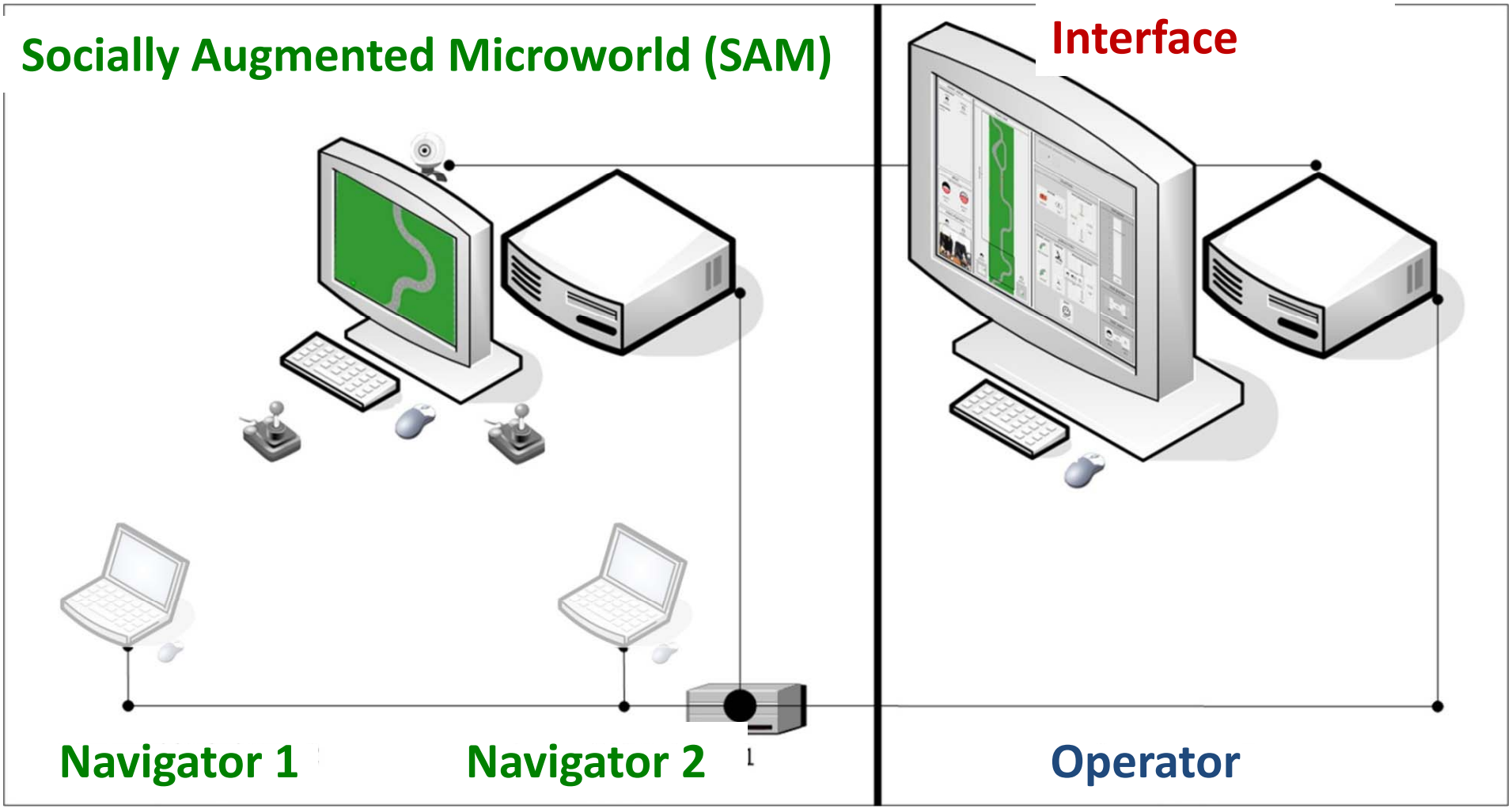
- Asynchronously: Advices by **operator** for **navigator**



Microworld with 2 navigators

Extended tracking task:
2 independent **navigators**
with limited communication (e.g. gestures, observation)




Socially Augmented Microworld (SAM)
Interface
Navigator 1
Navigator 2
Operator

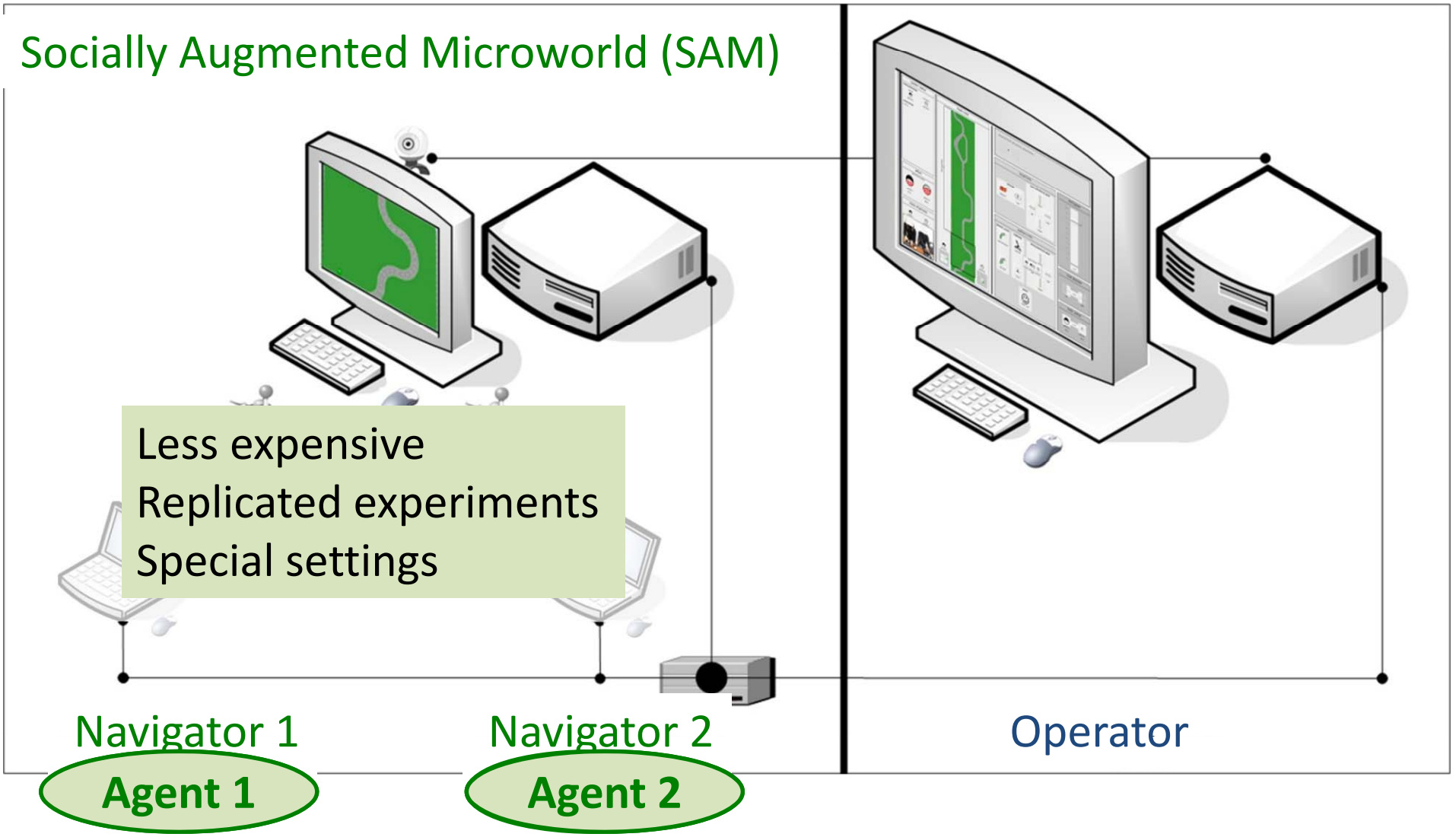
Control of microworld by operator

- Advices to the navigators, e.g.
  - driving style (fast, accurate, …)
  - road information (branch in front, obstacle, …)

- Change of the microworld, e.g.
  - vary the impact of navigators (standard 50% : 50%)
  - hide certain commands, e.g. no left driving

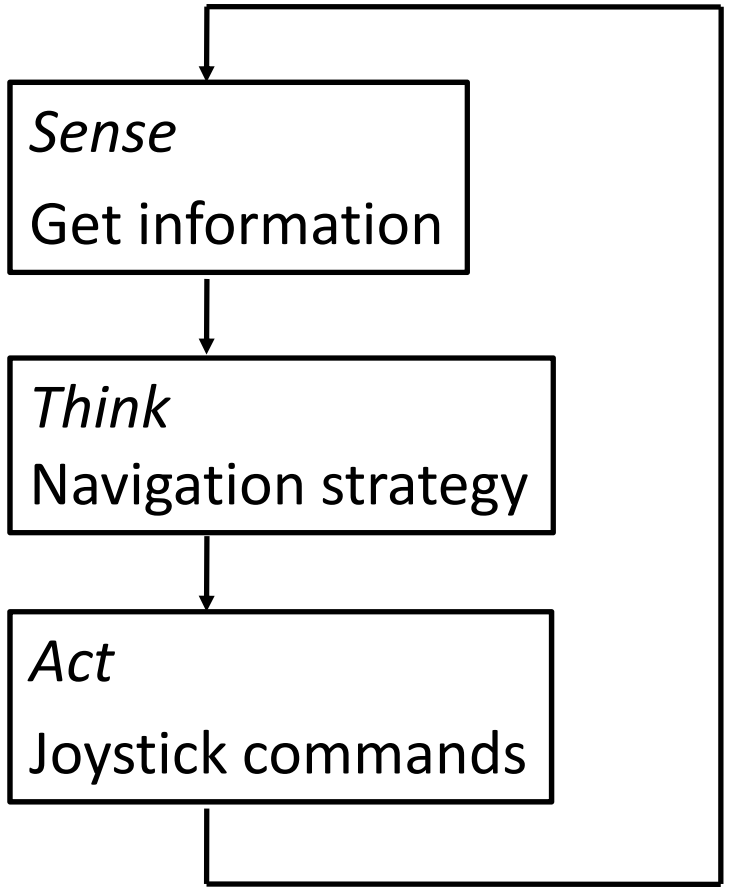
Experiments wit different interfaces.

Substitute human navigators by agents



Agent basics

- Sense-think-act-cycle
- Cycle time: 30 cycles per second



- Communication with SAM via
  - UDP for high frequent signals
  - TCP for low frequent signals

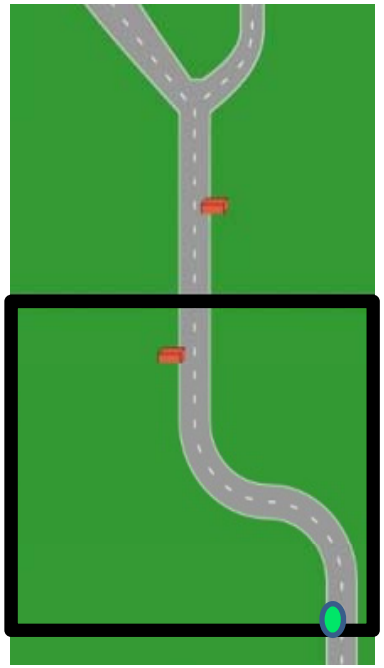
Reactive vs. deliberative behavior

Reactive:
No internal state:
Same action for
same sensory information

Deliberative:
Internal states like
• former information (world model)
• goals, plans
have impact on actions

Learning the partner's navigation strategy needs internal states

Sense: Visual information



Screen: Bitmap of road segment

about 800 lines with 800 pixels = 640000pixels

After preprocessing maximal 8 items per line:

left margin, middle, right margin of the road

left margin, middle, right margin of parallel road (if any)

position of object (only at bottom line)

position of obstacle (if any)

Maintaining by small updates per cycle:

new bottom line (according to speed of object)

new position of object on bottom line

new lines on the top of the image (according to speed)

Sense: Advices by Operator

Predefined textual input for

- Advices like
  - More/less accurate
  - More/less fast
- Information
  - Strong curve right/left in front
  - Obstacle/branch in front

Experiments wit different interfaces.

Sense: Observation of partner

- Not specified up to now
- Could be raw information about joystick movement
  or emotional reactions

In conflict with other visual information
(not both information at the same time?)

Act: Joystick commands

Position of joystick

- Steering: left/right (-100%,…,+100%)
- Acceleration: forward/backward (-100%,…,+100%)

Could realize more than human possibilities

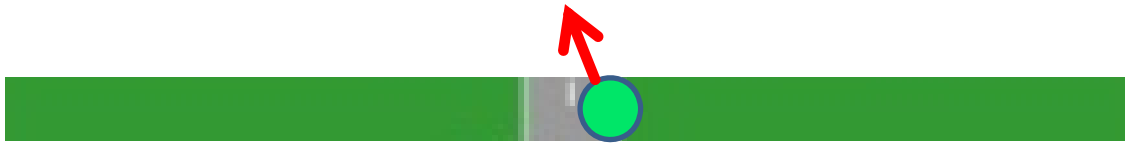
(e.g. fast rotating joystick …)

"Braitenberg Vehicle"

Simple reactive behavior

Keep the car on the middle of the road.

If object left from middle line, then drive right.
If object right from middle line, then drive left.



Deviation from middle line $d \in -800,\ldots,+800$ (number of pixels)

Steering $s \in -100\%,\ldots,+100\%$

Acceleration $a \in -100\%,\ldots,+100\%$

Proportional to deviation: $s = p_s \cdot d \quad a = p_a \cdot d$ *(p-controller)*

with parameters p_s , p_a

Reactive behavior

Parameters p_s and p_a

can be changed according to advices of operator, e.g. faster driving

Parameters p_s and p_a

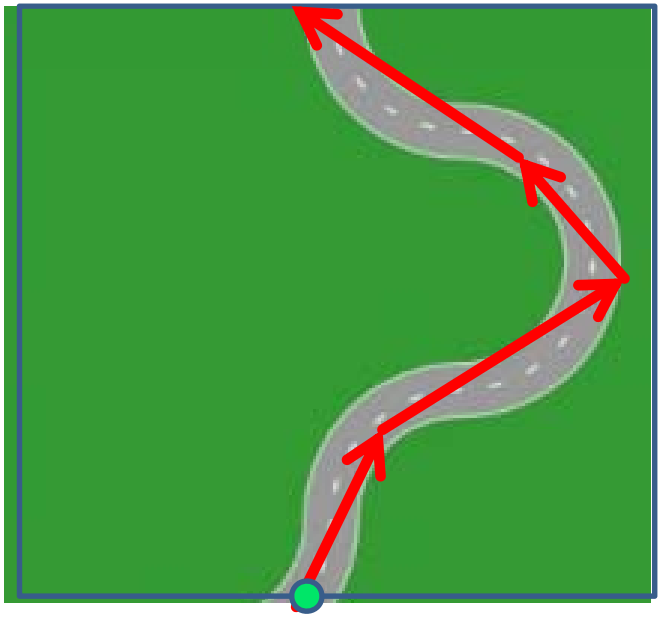
can be tuned by Optimization/Machine Learning

according to the (conflicting) goals speed and accuracy

Moderate performance

With foresight

Drive according to some calculation.



Reactive:

New calculation at each cycle

Deliberative:

Keep calculated options (“plan”)

Further parameters: Amount of foresight, stability of plans, …

2 Navigators

- Interference of different navigation strategies
- Indirectly observable by changed behavior

Modeling of partner needs

- memory of behavior and results
- knowledge about own strategy
- assumptions about the design of partner

Adaptation

Adaption to partner's behavior:

Bad results expected if both agents follow same adaptation style (by results from socionics experiments)

Further parameters for adaptation.
Operator can change the influence or give advices.

Experiments wit different interfaces.

Imitate human navigators

Identify *navigator types* according to

typical behaviors at different situations like

- Straight road
- Curves
- Branches
- Obstacles

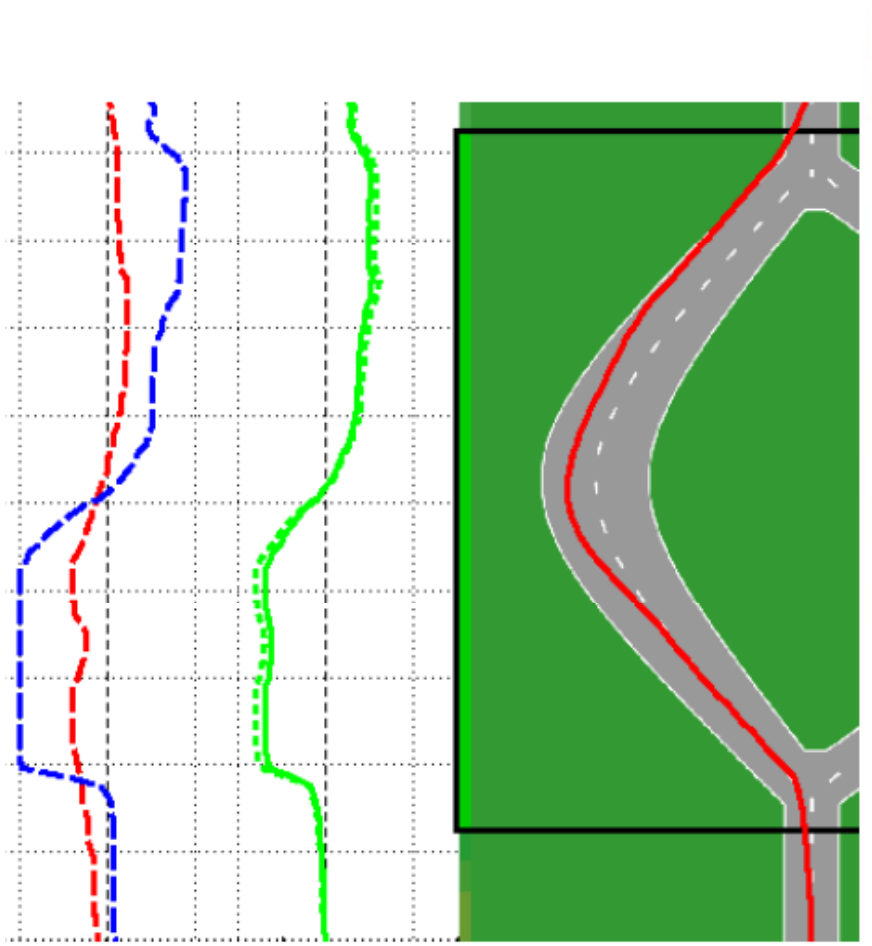
by analyzing logfiles of experiments with human navigators.

Up to now analysis by hand, later by computer.

Example of steering type

“Extrem steering type” (blue):

- maximal deflection at curves
- no weak deflections
- fast changes of direction
- oscillations on straight roads
- often in combination
  with „non steering type“



Analysis of human navigators

Classification of commands by nominal scales (e.g. none, light,…).

Categorize behavior at branches and obstacles.

Identification of *3 Types for acceleration*:

- Adjusting the speed according to the road (“normal”)
- Same speed all the time (“none”).
- Decelerate speed rapidly while not on the road (“decelerator”).

Possible types for steering (under work)

- Extreme steering
- No steering
- “Jittering” (corresponds to our simple reactive implementation)
- Direction oriented

Conclusions/Ongoing work

- Different approaches in interdisciplinary work.
- Implementation issues force more concrete descriptions.
- Usage of rules and fuzzy descriptions for navigator types.
- Implementations of navigator types.
- Machine Learning for imitating human navigators.
- Analysis of Experiments.
- Further applications behind the scope of SAM/ATEO.

Thank you!