Introduction to SPM 9555

Dr. ir. Igor Nikolic, L. Andrew Bollinger Msc

26-11-12

Lecture goals

- Have an overview of the course narrative
- Understand the goals and the rules of the course
- Choose a project to work on: See Spm9555ProjectTopics
- Set up groups

Course goals

- **Knowledge** - Know the different steps involved in creating and using an ABM and understand the way these steps influence each other
- **Insight** - Able to gain insight into a system by decomposing it in a structured manner
- **Application** - Able to independently create an ABM based on the system decomposition
- **Analysis** - Able to make an informed choice of adequate analysis methods and analyze the created ABM using them
- **Synthesis** - Be able to create a coherent narrative about a problem, based on the model development process, model and its analysis

Course materials

- Course materials: Wiki
- book “Agent-Based Modelling of Socio-Technical Systems”
  - Editors: Koen H. van Dam, Igor Nikolic, Zofia Lukszo
  - ISBN : 978-94-007-4933-7 (ebook)
  - http://link.springer.com/book/10.1007/978-94-007-4933-7/page/1

Model development steps

- Step 1: Problem formulation and actor identification
- Step 2: System identification and decomposition
- Step 3: Concept formalisation
- Step 4: Model formalisation
- Step 5: Software implementation
- Step 6: Model verification
- Step 7: Experimentation
- Step 8: Data analysis
- Step 9: Model validation
- Step 10: Model use

Model development 1

[Problem formulation and Actor identification] – *What is the problem?*

- What is the problem, whose problem is it, and who is involved ?

- The main lack of insight addressed is the poor understanding of the factors affecting technology diffusion between greenhouses. The observed emergent pattern is the unequal distribution of technologies among a group of greenhouse farmers, with many farmers owning sub-optimal technologies, despite the availability of better technologies and awareness of the advantages.

Model development 2

- [System identification and decomposition] – *Who is in involved?*
  - Identify what is the relevant system and the relevant level of aggregation. Specifying a generativist experiment.

- Agents are companies and they:
  - grow stuff
  - sell products
  - have a certain amount of money
  - make surface area dependent profit per year
  - own technologies
  - know about some technologies (inlcuding the ones they own)
  - have a certain satisfaction about the technologies they currently own ...

Model development 3

- [Concept formalization]
  - Concepts are translated/formalized into computer readable format. This can be done using a formal ontology, or a translation to computer primitives.

- Companies have:
  - money: integer
  - profit per year: integer
  - technologies: a list
  - satisfaction: integer >= −1 and <= 1
  - technology libraries: list of lists.
  - opinion library: list of lists of integers >= −1 and <= 1.
  - stubbornness: integer >= 0 and <= 1
  - opinion change rate: integer >= 0 and <= 1.

Model development 4

- [Model formalization] – *What is the model?*
  - We create a model narrative (as text or action sequence diagrams). Who does what with whom and when. Use the concepts identified in previous step.
- Companies do:
  - calculate profit
    - profit = revenues - costs
    - revenues = price for their product * surface area * (base production + sum(performance))
      - price for products is dependent on the company type
      - base production is dependent on the company type
      - performance is dependent on the technologies owned and the company type
    - costs = basecost + surface area * (operating cost + ...c

Model development 5

[Software implementation] – *How the model works*

- Build the actual software representation of the formalized model

```
to update-OpinionLibraries
  ask companies [
    set OpinionLibrary2 OpinionLibrary
    let teller 0
    foreach TechLibrary [
      let temp ?
      foreach temp [
        let temp2 ?
        ifelse temp2 = item teller Technologies [
          set OwnOpinion satisfaction] [
          set OwnOpinion item temp2 item teller
              OpinionLibrary]
        let teller2 0
        let temp3 0
        foreach AllMyneighbours [
          ask ? [
            if temp2 = item teller Technologies [
              set teller2 teller2 + 1
              set temp3 temp3 + satisfaction ]]]
        ifelse teller2 > 0 [
          ifelse NoiseChance >= random-float 1 [
            set OtherOpinion random-float 2 - 1][
            set OtherOpinion temp3 / teller2]] [
          set OtherOpinion item temp2 item teller
              OpinionLibrary]
```

Model development 6

[Model verification] – *Did we make what we wanted?*

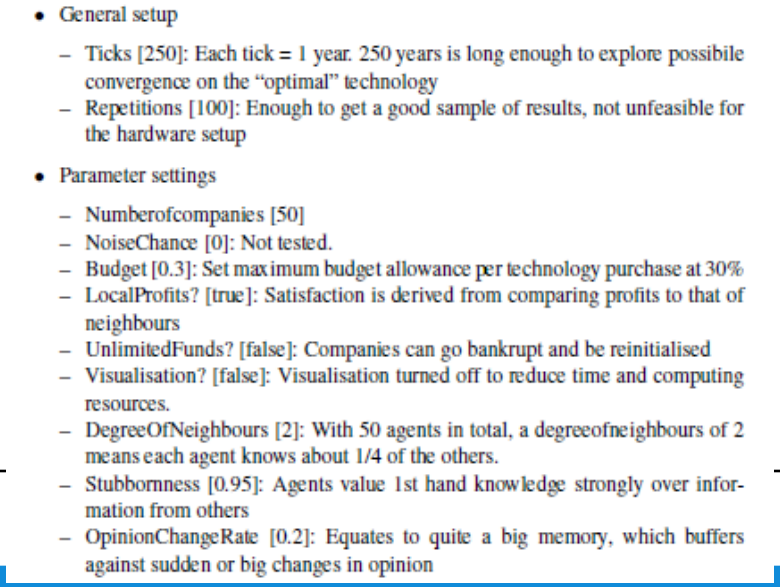
- Verification of the implementation is performed, demonstrating that the encoded agents and their behavior is indeed the behavior that was meant.

Minimal model: Technology and Satisfaction update

- If one neighbour has a given technology but another does not, then after the technology update code, the new technology should appear in the second agent's technology library. **Error found**. Technologies were added to the second agent's technology library that did not correspond to technologies owned by neighbours. **Fixed, revalidated, confirmed.**
- Satisfaction should be normalised to a value between -1 and 1. **Error.** The values were always appearing as either -1 or 1, but nothing in between due to a wrong variable in the code (temp2 instead of temp[14]). **Fixed, revalidated, confirmed.**

Model development 7

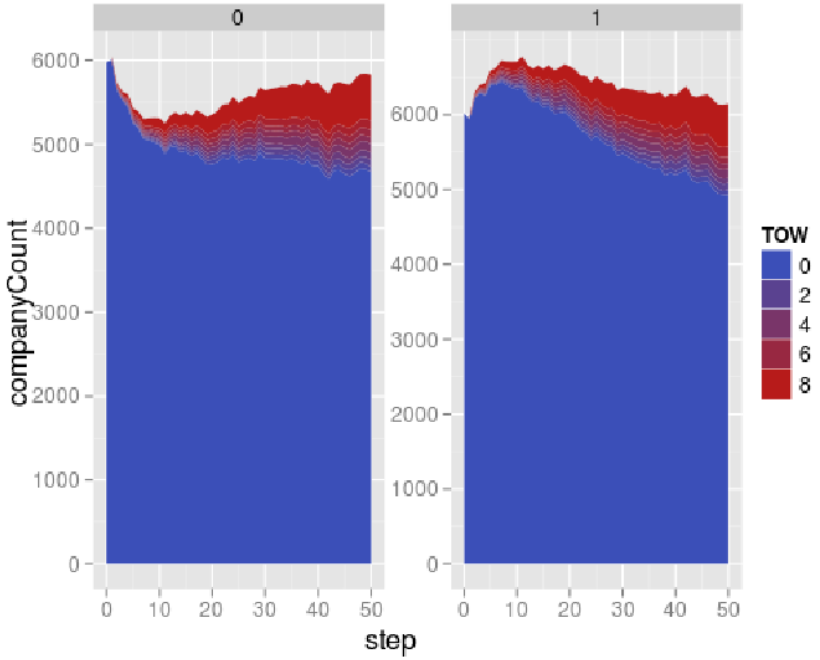
- [Experimentation] – *What does the model do?*
  - During the experimental setup the experiments are designed that will help answer the original questions. This involves identification of relevant (individual and system level) metrics, scenarios, parameter sweeps etc.



- General setup
  - Ticks [250]: Each tick = 1 year. 250 years is long enough to explore possibile convergence on the "optimal" technology
  - Repetitions [100]: Enough to get a good sample of results, not unfeasible for the hardware setup
- Parameter settings
  - Numberofcompanies [50]
  - NoiseChance [0]: Not tested.
  - Budget [0.3]: Set maximum budget allowance per technology purchase at 30%
  - LocalProfits? [true]: Satisfaction is derived from comparing profits to that of neighbours
  - UnlimitedFunds? [false]: Companies can go bankrupt and be reinitialised
  - Visualisation? [false]: Visualisation turned off to reduce time and computing resources.
  - DegreeOfNeighbours [2]: With 50 agents in total, a degreeofneighbours of 2 means each agent knows about 1/4 of the others.
  - Stubbornness [0.95]: Agents value 1st hand knowledge strongly over information from others
  - OpinionChangeRate [0.2]: Equates to quite a big memory, which buffers against sudden or big changes in opinion

Model development 8

- 8 - [Data analysis] – *What are the results?*
  - The data generated during (multiple) model runs will be explored. Based on the type of data and model, different data analysis and visualization techniques will be employed.



Model development 9

- 9 - [Model validation] – *How realistic is the model?*
  - Explore the validity of the created model and its outcomes.

Model development 10

- 10 - [Model use] – *What can we do with the mode?*
  - How can the model and its outcomes be used in order to answer the original research question.

Projects

- During the course, you will create a model in groups of three.

- Will use NetLogo as the modeling environment
- Choice of systems to be modeled available from the Spm9555ProjectTopics list.
- The topics are taken from current research, and are coached by researchers.
  - (if we have more than 27 student, students can suggest their own projects, that must be approved by the teachers)

- Advanced Tool Project (2x3 students)
  - Students with VERY advanced computer skills may apply for this stream, teachers will select the students involved.
  - Expected skill set: solid Java programming skills, effortless working with EclipseIDE / svn, high level of computer literacy and ability to work with large and diverse datasets.
  - Will use Java/SPARQL/OWL ontologies/ simulation engine developed at the E&I group
  - Same examination rules apply as for the regular stream.

Course lectures and homework

- Each week there are two contact moments :
  - Monday afternoon seventh and eight hour - 15:45 - 17:30
    - Classroom D2
  - Wednesday morning, third and fourth hour - 10:45 - 12:30
    - Classroom C
- Lectures will consists of theory presentation and discussion of problems/progress so far.
- During the scheduled WorkOnModel moments there are no formal lectures, but you can interact with the teachers.

- You are expected to perform the modeling steps for your project between the lectures.

Use of wiki

- Entire course in the wiki
  - http://wiki.tudelft.nl/Education/SPM955xABMofCAS
  - Reports of the projects are stored on the StudentPages subweb
  - In the wiki anyone can see *who* did *what when*

Examination

- The grade will be determined by the project report, uploaded to / written on the wiki.
- Project presentations during the exam slot
  - 1 February 2013, 9:00 – 12:00 Lecture room A

Wrap up

- This is going to be a quite challenging course, with a fair amount of work involved.
- The depth and extent of how far we can go greatly depends on your involvement and enthusiasm.

- We are going to do our best to make it as fun as I can.
- Lets go and discuss projects !