Aalto University
School of Engineering

Project Course in Computation and Modelling in Engineering

Rak: C3005

Vishal Singh, Assistant Professor
Department of Civil Engineering, Aalto University

Course: Learning outcomes

LO1: Gain hands-on experience in transforming a problem stated in plain language into computational form

LO2: Able to select and apply one or more of the computational modelling approaches in engineering that suits the given problem

LO3: Understand the practical challenges in implementing and applying the chosen method

LO4: Understand the strengths and weaknesses of the applied methodology

Approach and expectations

- Problem based learning
- Mentors and supervisors
- Tangible solutions/ Presentable to exhibit
- Focus on method and exploration

Timetable of activities

Date	Sessions	Activities	Milestones
25.2.	Class	Overview of the course + preliminary discussion	
4.3.	Class	Project discussions- Brainstorming preliminary ideas	Preliminary ideas
11.3.	Class	Problem formulation and selection for projects	**Common reading review- documented**
18.3.	No Class	Reading break-	Focused literature selection and reading
25.3.	Class	Problem discussion and refinement- Discussion on alternatives for each project	
1.4.	Class	Optional class- discussion on alternatives	**Focused literature review- documented**
8.4.	**Class**	**Mid-course presentations**	Report alternatives
15.4.	Meetings With supervisor	Detailing one technique- (*No class)*- arrange meeting with supervisors. Time and place to be agreed between students and supervisors	
22.4.			
29.4.			
6.5.			
13.5.	**Class**	**Final presentations**	**Demonstration**
20.5.	**Submission**	**Report+ Poster Submission**	**Final reports**

Teachers

Vishal Singh

BIM/ Generative design/
Agent based modeling

Toni Kotnik

Design of structures/
Parametric design

Jarkko Niiranen

Design of structures/
FEM/ Numerical methods

Heikki Remes

Marine structures

Assessment method

- Project work (55%) including how it followed the timeline
- Literature review report (10%)
- Mid-presentation (5%)
- Final presentation (10%)
- Project report (20%) submitted at the end of the project

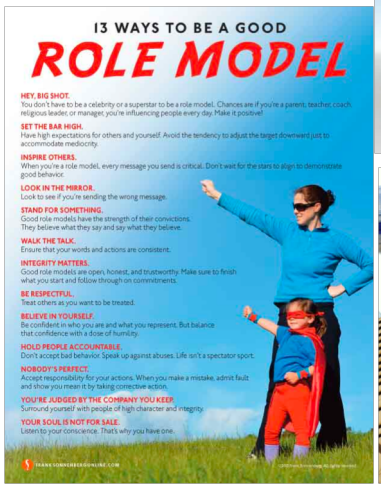
Project report

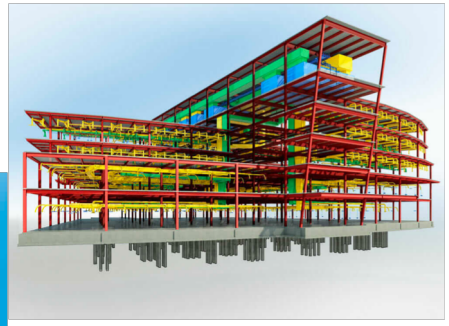
- A template structure will be provided
- Approx. 15 pages (including references, excluding appendix)
- Append copies of digital mock-ups, presentations, analysis

Computation and Modelling

Model

- Fashion model
- Role model
- Theoretical model
- Conceptual model
- Business model
- Physical model
- Virtual model





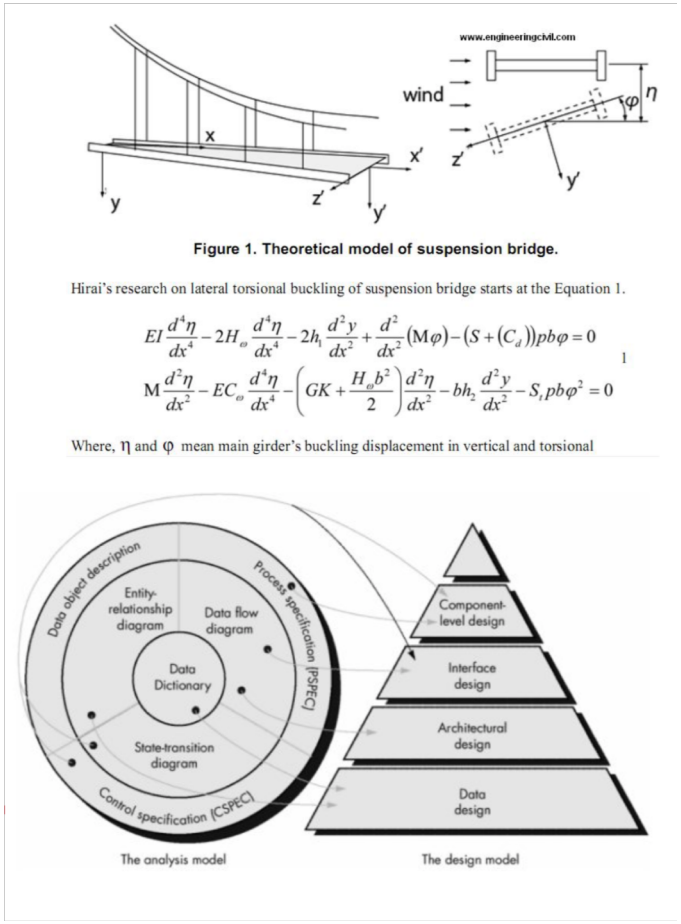


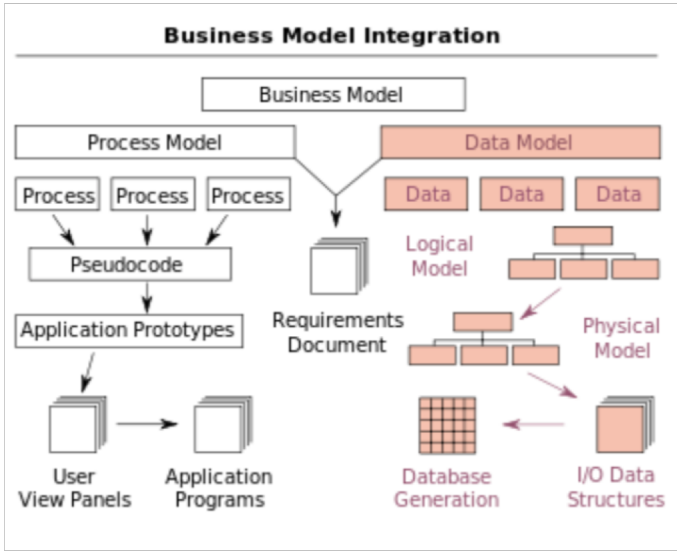
Figure 1. Theoretical model of suspension bridge.

Hirai's research on lateral torsional buckling of suspension bridge starts at the Equation 1.

$$EI\frac{d^4\eta}{dx^4} - 2H_\omega\frac{d^4\eta}{dx^4} - 2h_1\frac{d^2y}{dx^2} + \frac{d^2}{dx^2}(M\varphi) - (S + (C_d))pb\varphi = 0$$

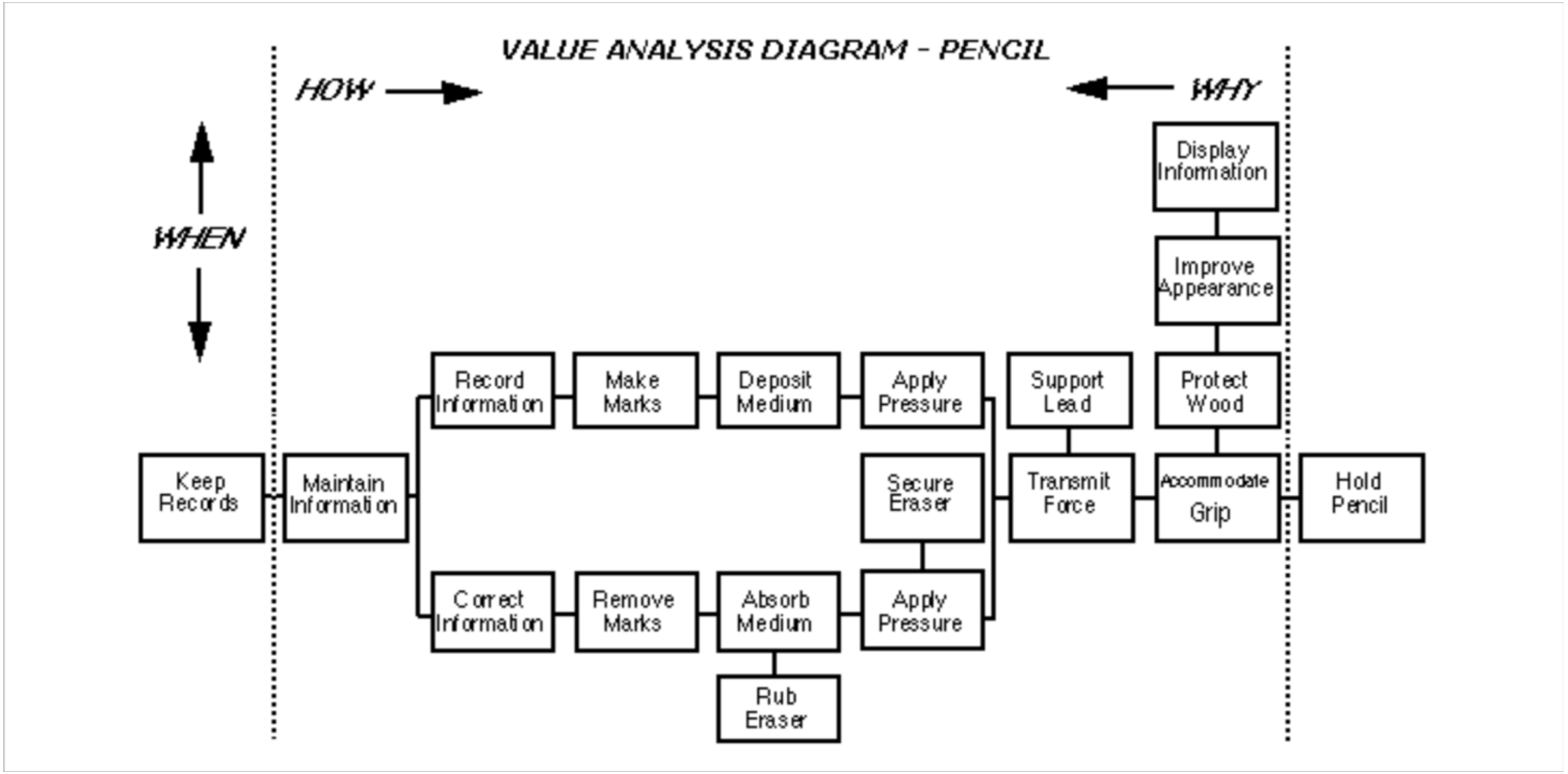
$$M\frac{d^2\eta}{dx^2} - EC_\omega\frac{d^4\eta}{dx^4} - \left(GK + \frac{H_\omega b^2}{2}\right)\frac{d^2\eta}{dx^2} - bh_2\frac{d^2y}{dx^2} - S_t pb\varphi^2 = 0 \quad (1)$$

Where, η and φ mean main girder's buckling displacement in vertical and torsional



COMPUTATIONAL THINKING

Example: Approach to decompose a problem



Computation and Modelling

- **Computation:** Calculation or information processing

- **Modeling:** Representation, structure, relationships…

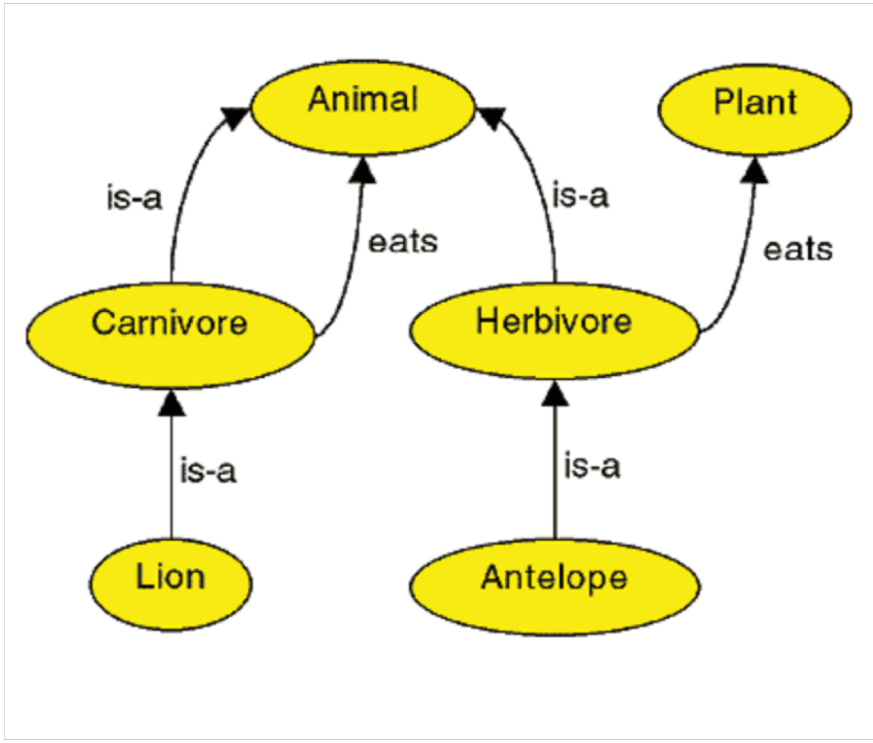
Computation and Modelling- Representation/ structure/ relationships-?

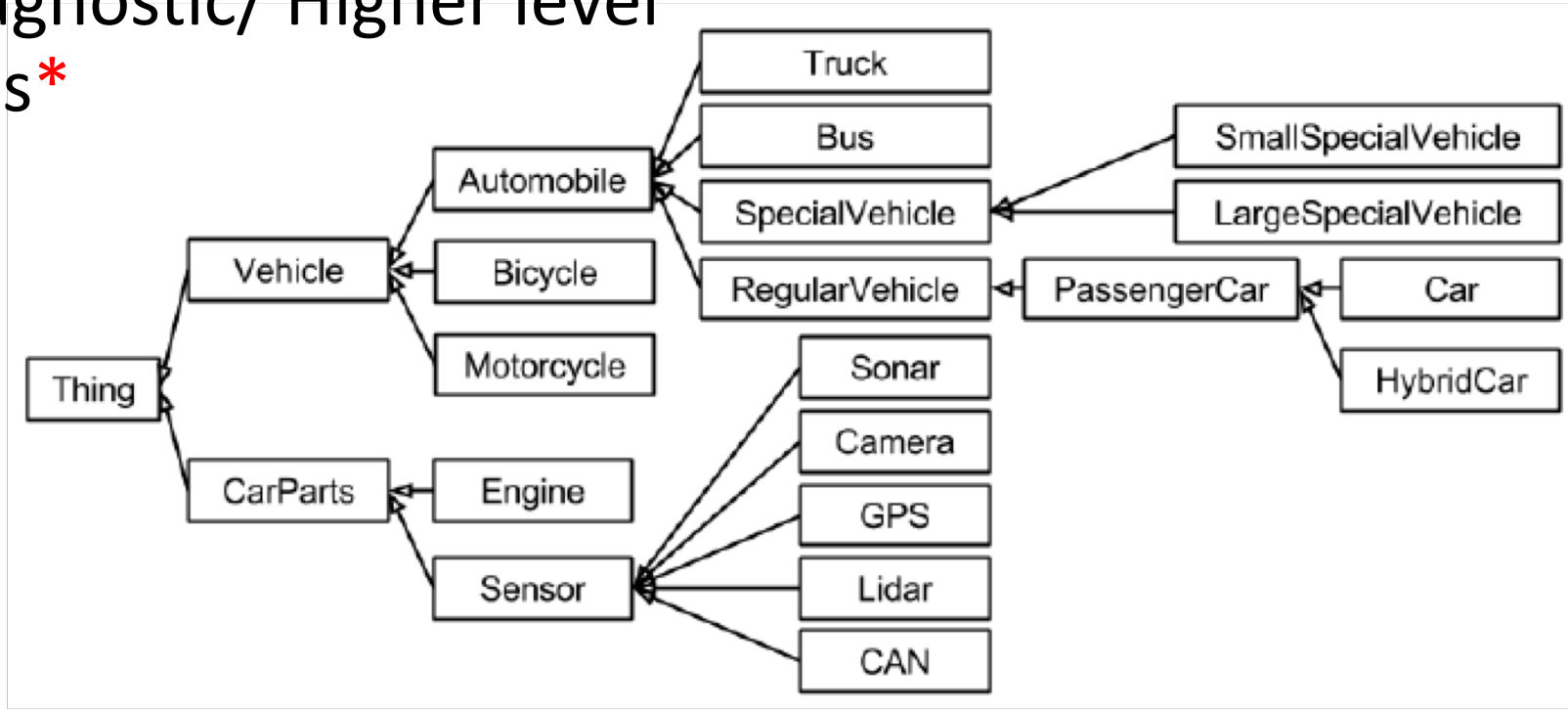
Conceptual to computational

- Ontologies
- Parameters, variables, constraints, objective functions
- Flowcharts
- UML (Unified Modelling Language) diagrams
- Pseudocodes and algorithms
- Data Structures…

Ontology

- Nature of being
- Types, properties, interrelationships between entities of a given domain
- Domain-agnostic/ Higher level ontologies*





*

Example: FBS Ontology in design- Gero

Formulation (1)

Synthesis (2)

Analysis (3)

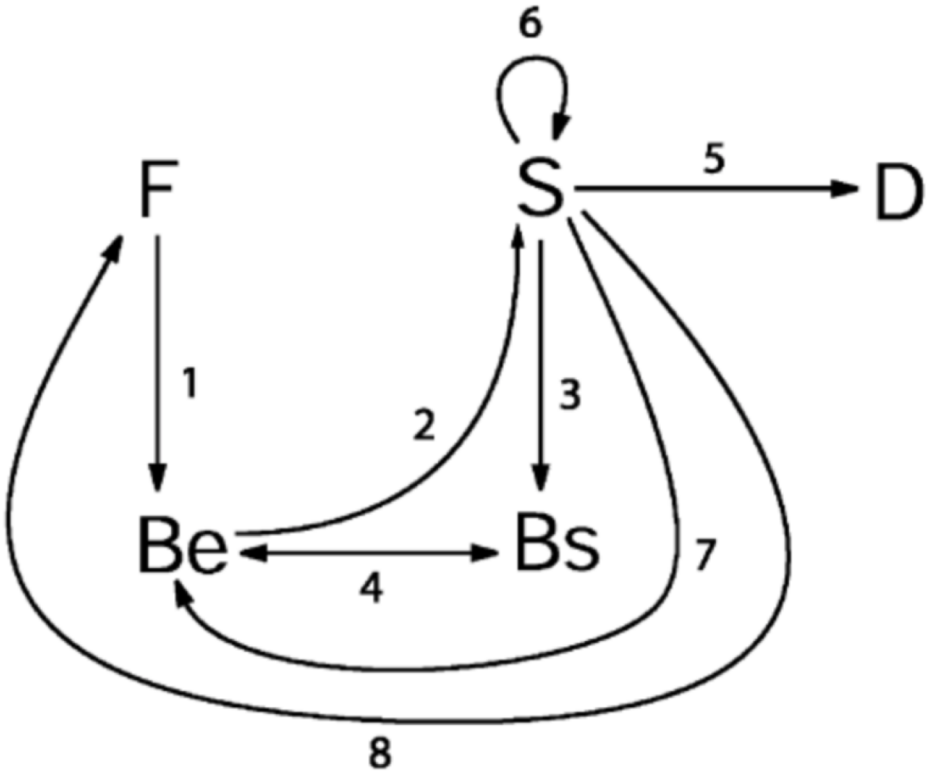
Evaluation (4)

Documentation (5)

Reformulation type 1 (6)

Reformulation type 2 (7)

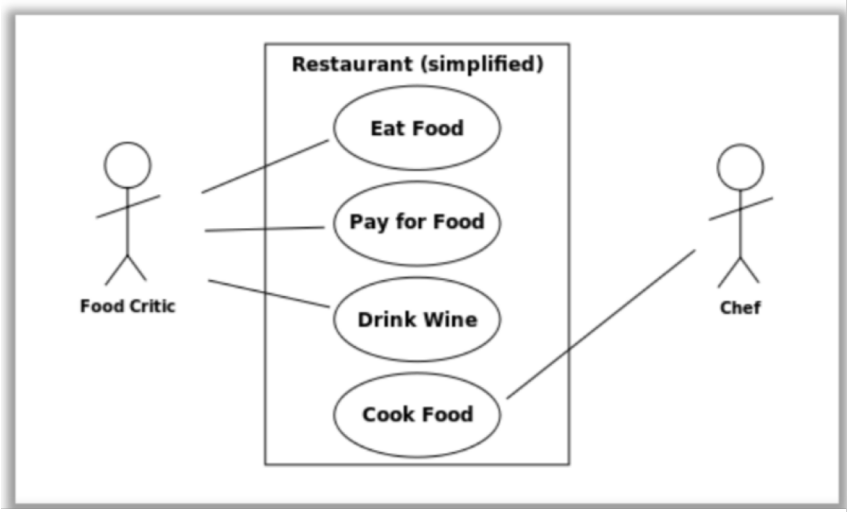
Reformulation type 3 (8)

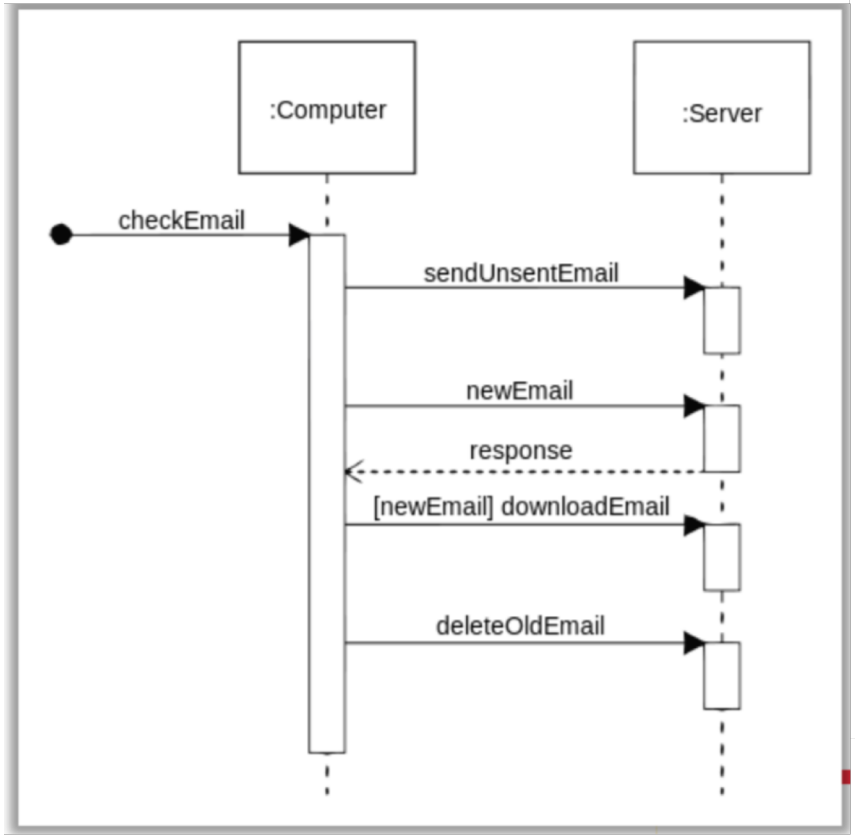


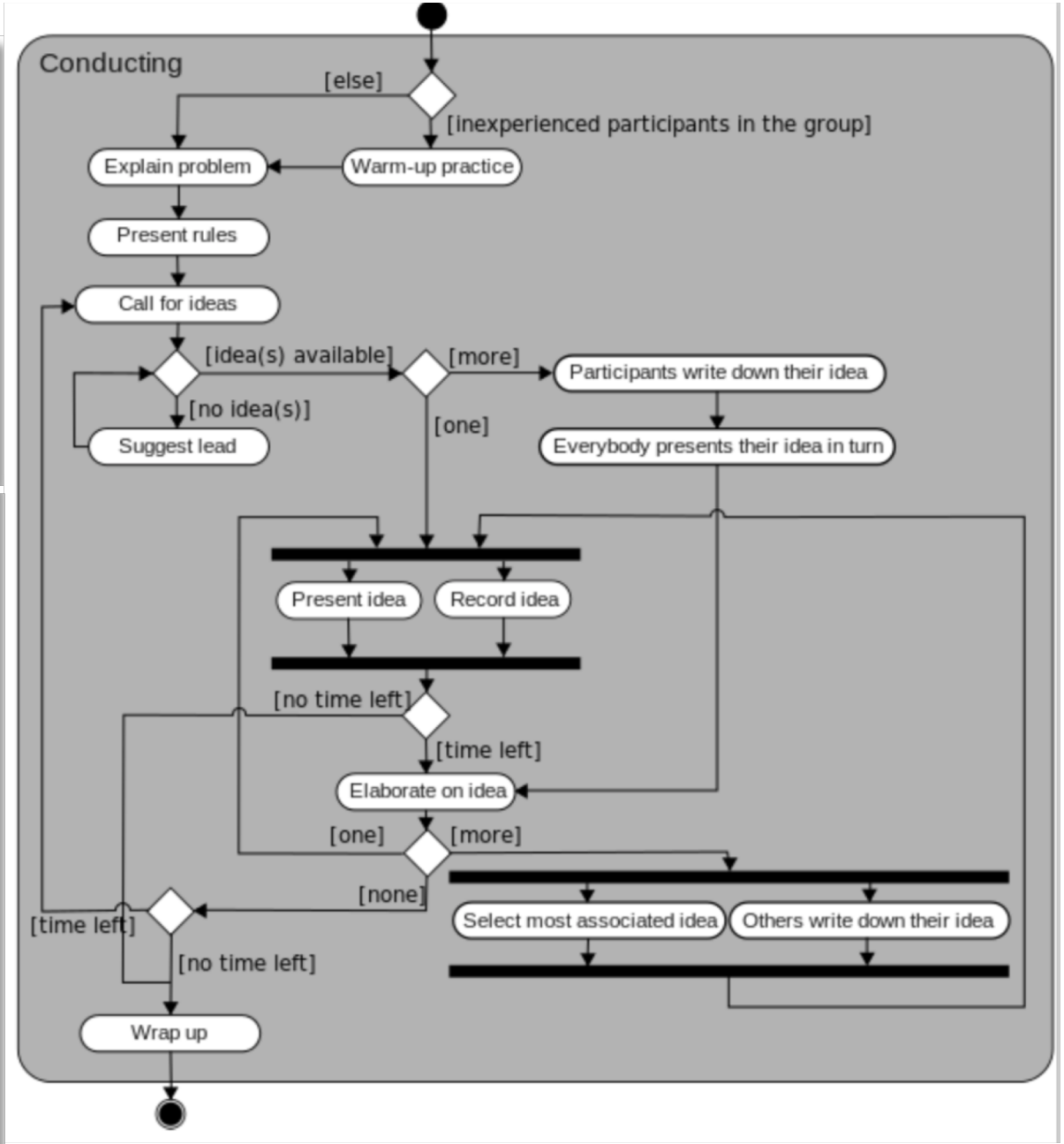
Be = expected behaviour
Bs = behaviour derived from structure
D = design description
F = function
S = structure

→ = transformation
↔ = comparison

UML diagrams

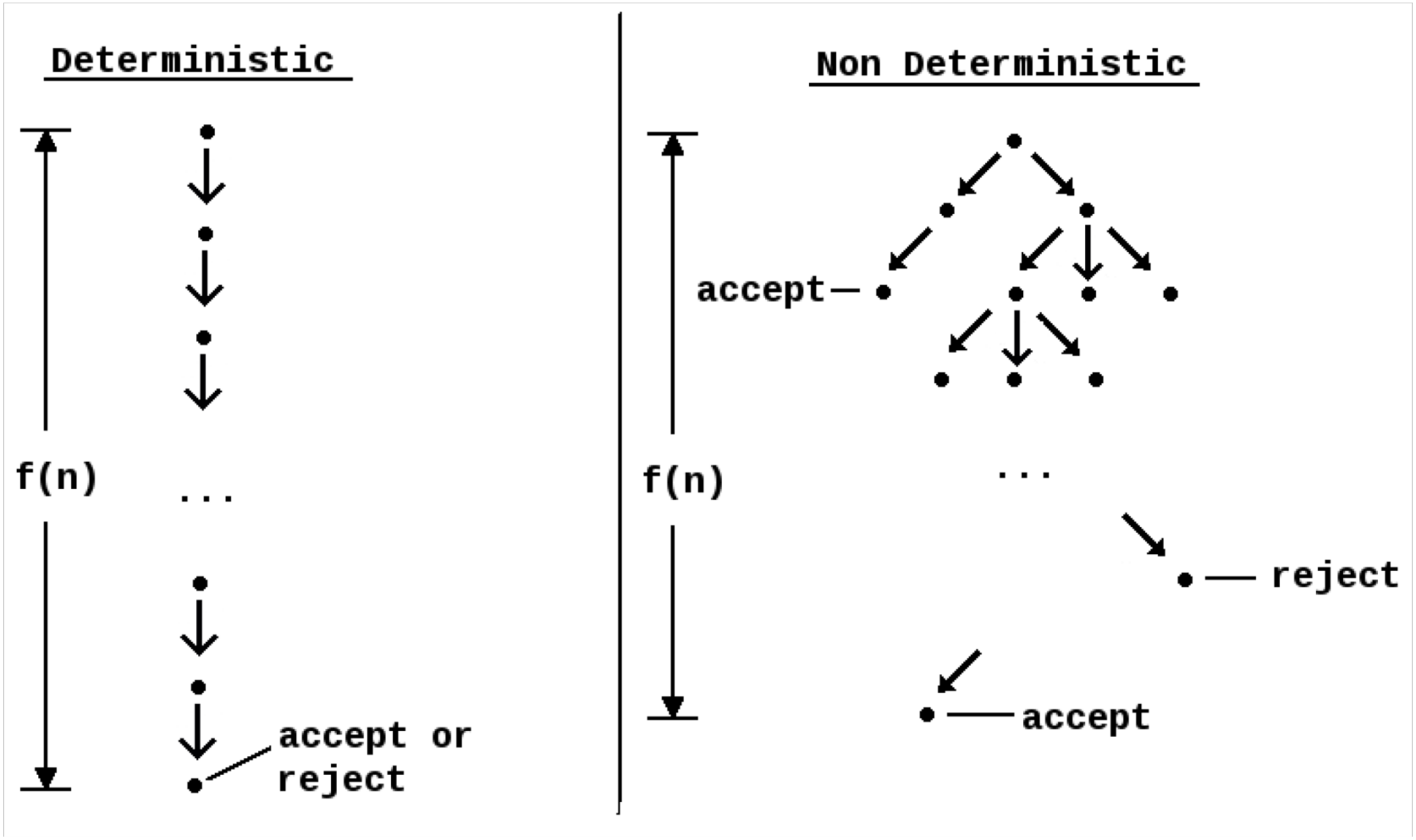






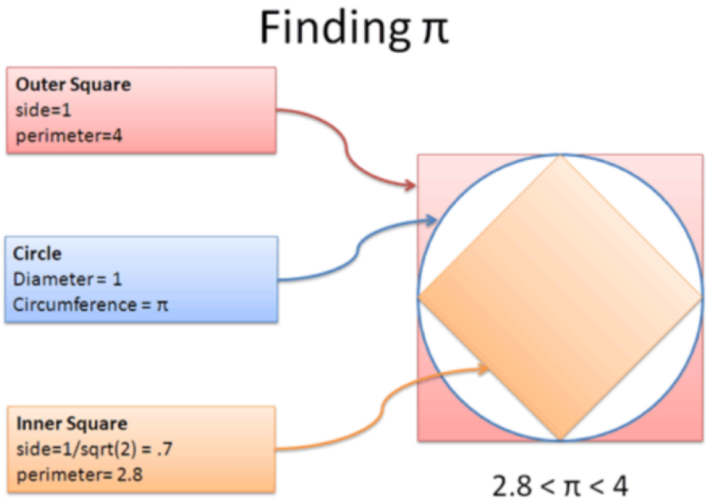
Solutions depend on how you define the problem!

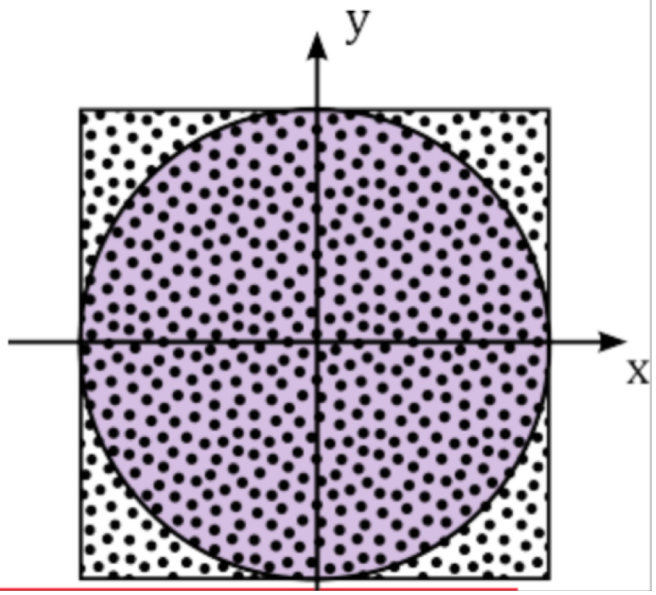
Example: Computing the value of pi?



Example: Computing the value of pi?

- Archimedes' method (deterministic)
- Monte Carlo technique (stochastic)

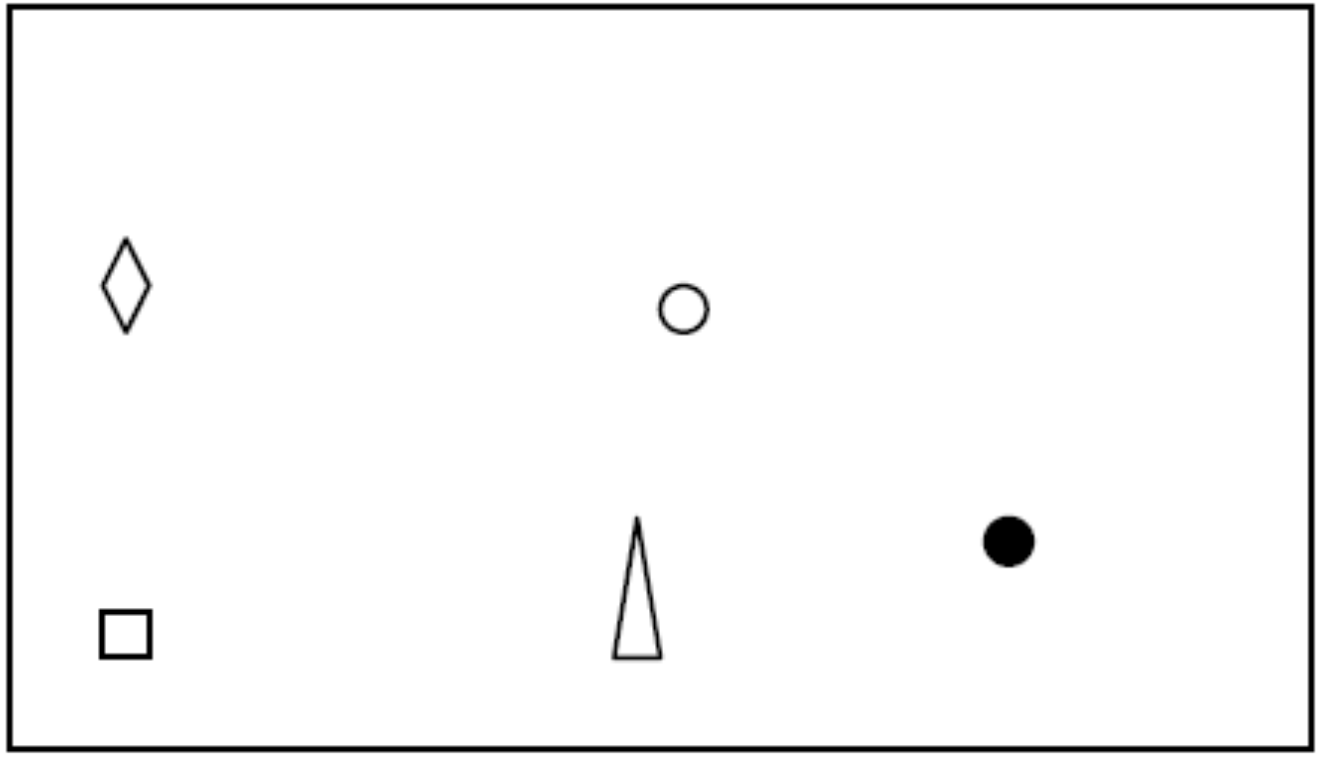




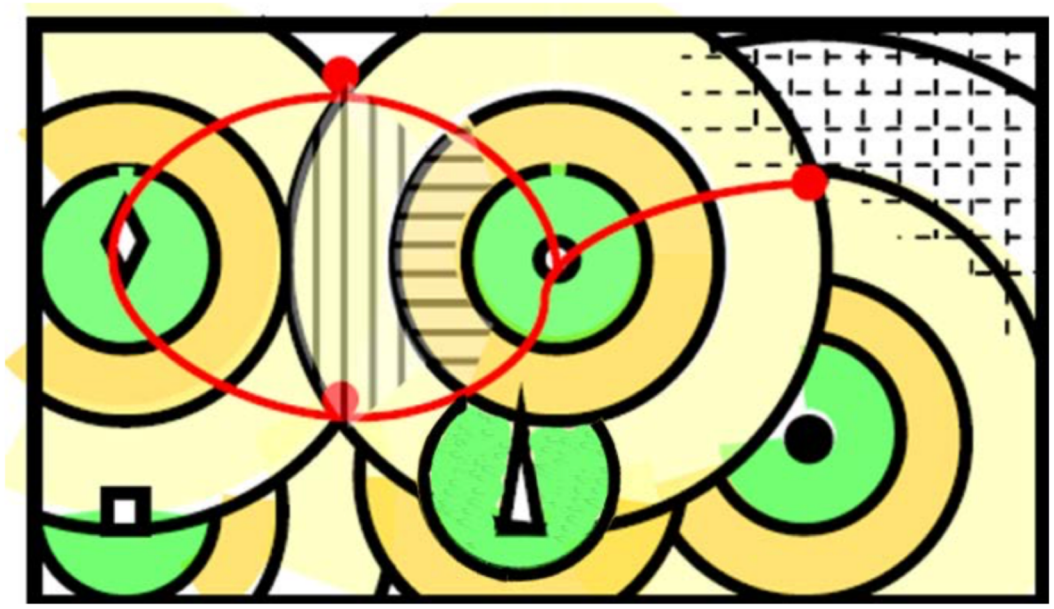
http://www.math.ubc.ca/~cass/courses/m446-03/pi.pdf

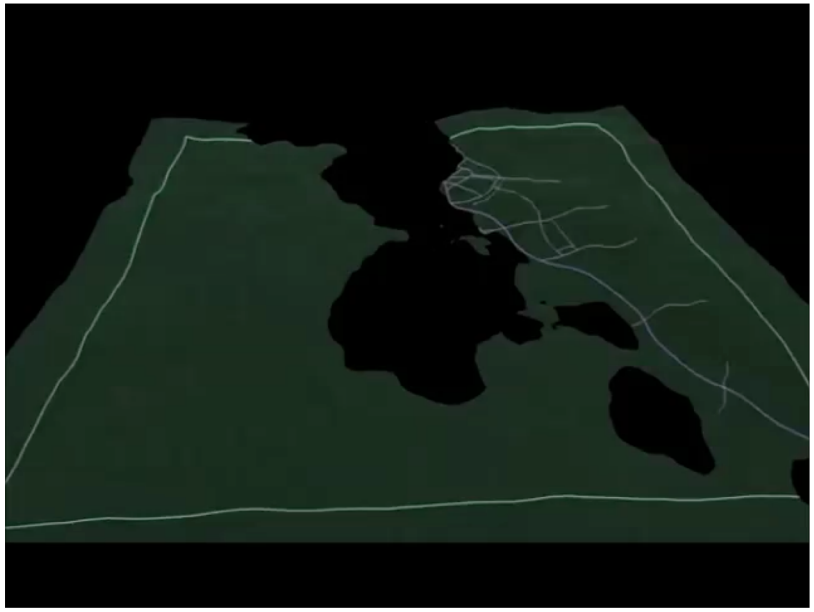
http://mathfaculty.fullerton.edu/mathews/n2003/montecarlopimod.html

Example project-Site layout



Site layout- Using shape grammar



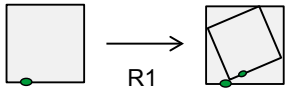


CityEngine®

Design Grammar

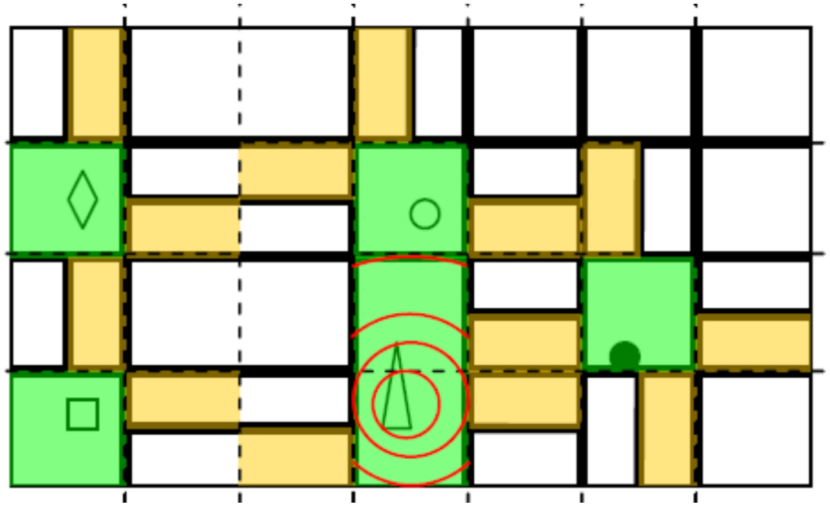
Shape Grammar (SG)

- Set of terminal shapes
- Set of operators
- Set of production rules
- Initial shape



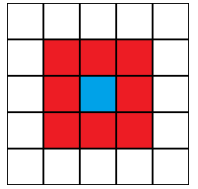
composition 

Site layout-Using Cellular Automata

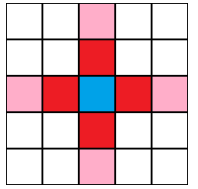


Cellular automata

- Grid/ Matrix
- Cell states
- Neighbourhood relationships
- Set of rules
- Assumption of universality

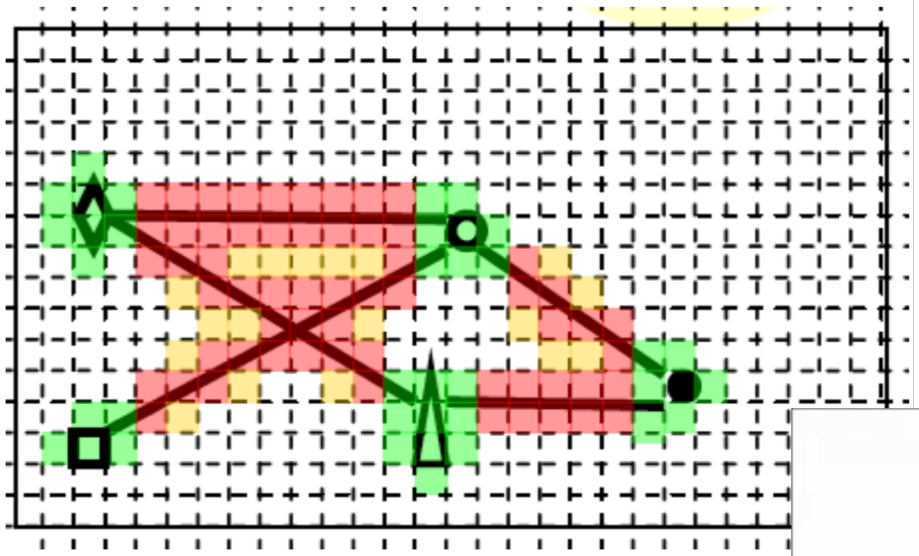


Moore neighbourhood



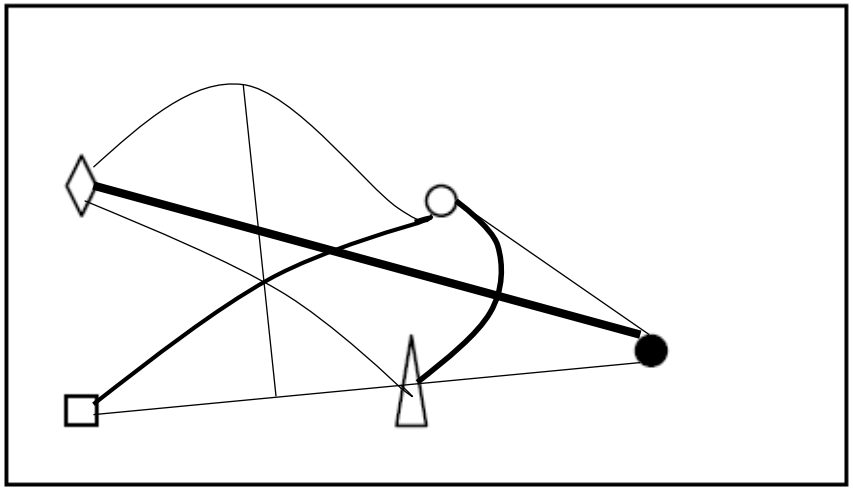
von Neumann neighbourhood

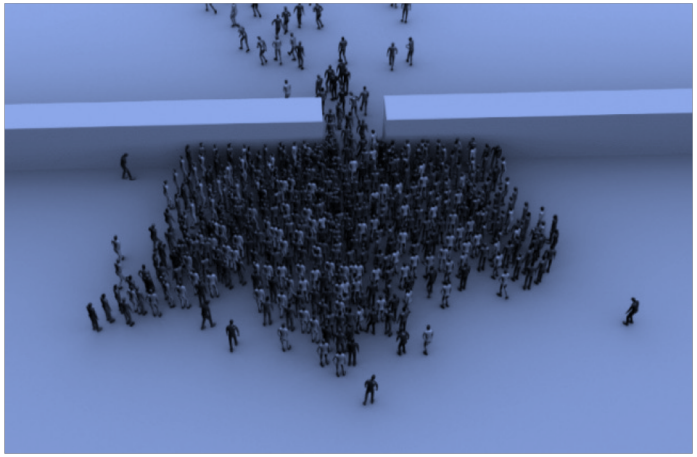
Site layout- Using Cellular Automata



VOXOPOLIS

Site layout: Using Agent Based Model



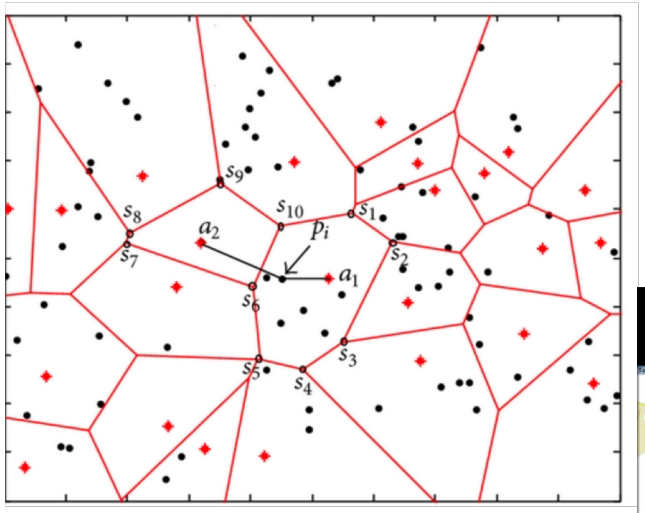


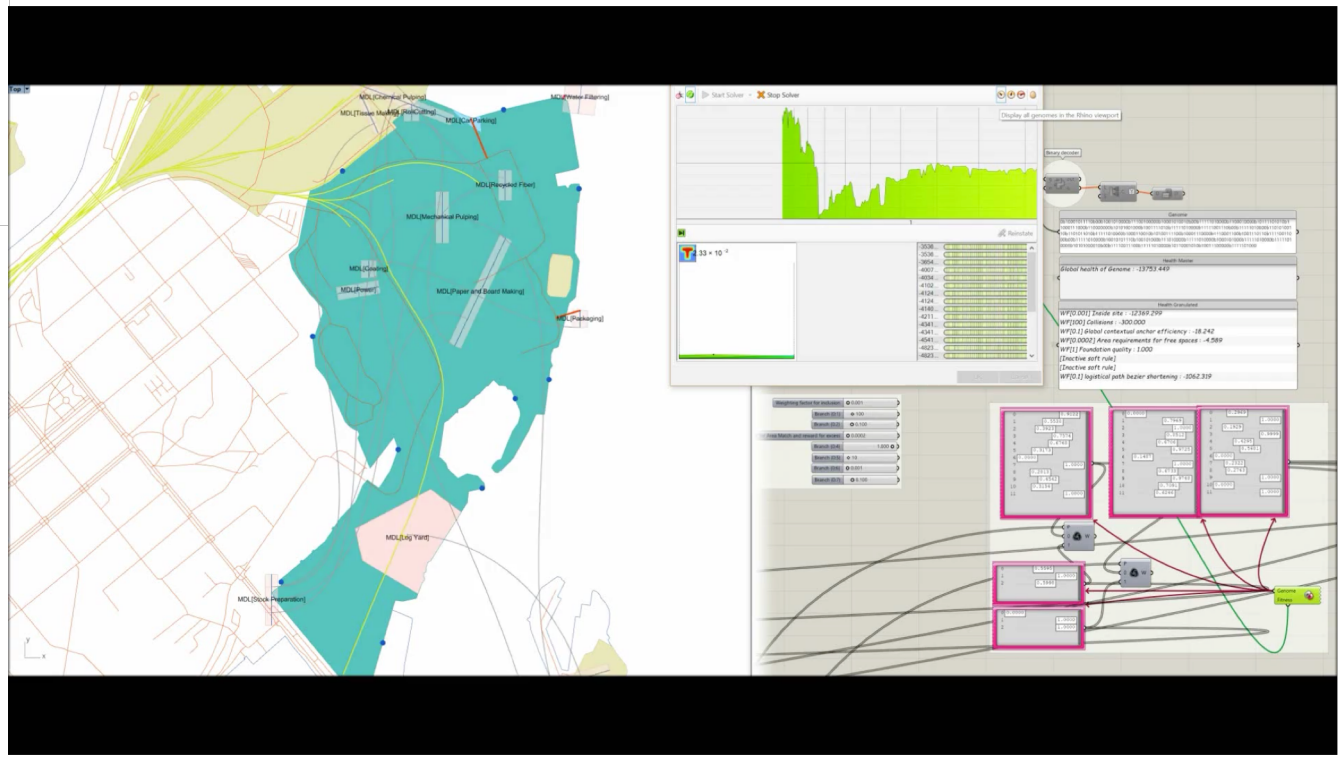
Agent based models/ Multi-agent systems

- Autonomous agents
- Set of actions/ goals/ behaviour/ intentions
- Interaction with each other and the environment
- Influence/ forces/ “social physics”
- Agents: usually people but …

Other methods…

Voronoi diagram, spring mass models, gravitational model …



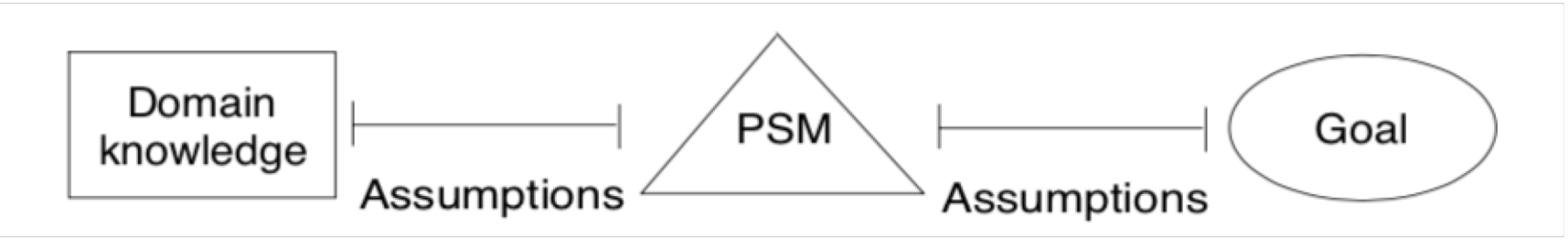


Solutions depend on how you define the problem!

BUT

Defining Problem-solution approach depends on assumptions you make!

Assumptions in problem solving Method (PSM)



- Teleological assumption- (problem definition)- gap between the problem solving method (PSM) and goal
  - These assumptions (weaken) reduce the complexity of the problem

- Ontological assumption- (problem representation)- gap between the PSM and the domain knowledge (approximation)
  - With greater domain knowledge these assumptions strengthen the PSM
  - These also reduce the complexity of part of the problem

Requirements from the course

- Understand problem context
- Define problem in plain language
- Investigate problem requirements
- Define problem conceptually
- Narrow down key parameters, variables, functions
- Note your assumptions. Reflect on the role of the assumption.
- Define/ represent problem computationally
- Explore different alterative representations
- Choose one of the alternatives (justify)
- Implement the chosen alternative
- Report the solution as well as the entire process!

Online references- quick read

https://courses.edx.org/c4x/MITx/6.00.1x/asset/files_ps04_files_WhyPseudocode.pdf

https://www.ibm.com/developerworks/rational/library/769.html

http://www-ksl.stanford.edu/kst/what-is-an-ontology.html

https://protege.stanford.edu/publications/ontology_development/ontology101-noy-mcguinness.html

https://www.w3schools.in/data-structures-tutorial/intro/

https://ampl.com/BOOK/CHAPTERS/11-linprog.pdf

http://www.me.utexas.edu/~jensen/or_site/models/unit/lp_model/lp_terms/lp_terms.html

Questions?

Thank You!