ISMS 2013 Keynote – 29 Jan 2013

A Formal Approach to Identify Emergent Properties in Simulation Models

Yong Meng TEO*
Department of Computer Science
National University of Singapore (NUS)
email: teoym@comp.nus.edu.sg
url: www.comp.nus.edu.sg/~teoym

Acknowledgement:
Luong Ba Linh (NUS), Claudia Szabo
(University of Adelaide)

* Visiting Professor, Shanghai Advanced Research Institute,
Chinese Academy of Sciences, China

What is Emergence?

"The whole is greater than the sum of its parts"

- Aristotle

Complex Systems

- Complexity of systems is increasing
  - Large number of components
  - No global control, simple local rules
  - Non-linear connectivity among components

- Some system properties (**emergent properties**), do **not result** from the properties of its interconnected components

→ Component-based system with emergent properties

Component-based Systems

- Composed of multiple interacting components
- Component
  - "… a stand-alone functional element that is defined by its input and output behavior" [Hinton, 1997]
- Behavior
  - "A sequence of state changes it (component) undergoes during **a period of time"** [Chen, 2009]
- Property
  - Characteristic that can be detected at **a point in time**

Outline

- What is Emergence?
- Motivation
- Emergence Perspectives
- Formalizing Emergent Properties
- Propose Grammar-based Approach
- Example: Bird Flocking (extended Boids Model)
- Conclusion and Future Work

Examples/Applications of Emergence

- Game of life [Gardner, 1970]
- Flock of birds [Reynolds, 1987]
- Traffic network [Nagel, 1992]
- World Wide Web [Adamic, 2000]
- Stock market [Chen, 2002]
- More examples [Mogul, 2006]

Game of Life [Gardner, 1970]

- Micro level
  - **Local knowledge**
    - Each cell knows states of its eight neighbors
  - **Simple behavior rules**
    - A live cell with fewer than two live neighbors or more than three live neighbors becomes dead
    - A dead cell with exactly three live neighbors becomes alive

- Macro level
  - Some patterns of cells have the ability to move (glider) or reproduce (oscillators)

- Related studies of emergence in the Game of Life
  - [Chan, 2011], [Sanders, 2009], [Kubik, 2003], [Wuensche, 1999]

Traffic Network [Nagel, 1992]

- Micro level
  - **Local knowledge**
    - Each entity (vehicle, pedestrian, traffic light, etc.) has limited knowledge of its neighborhood.
  - **Simple behaviors rules**
    - Each entity obeys a set of simple rules (e.g. a vehicle obeys movement rules to avoid collision and maximize its speed)
- Macro level
  - Traffic congestion, traffic jam
- Related studies of emergence in traffic network
  - [Han, 2012], [Manley, 2010]

World Wide Web [Adamic, 2000]

- Micro level
  - **Local knowledge**
    - Each page links to some other pages
    - No central organization rationing the number of links
  - **Simple behavior rules**
    - No specific rules, i.e. a page can have a hyperlink pointing to any other page

- Macro level
  - The distribution of links follows a power law in which a few pages are linked to many times and most pages are seldom linked to

Stock Market

- Micro level
  - **Local knowledge**
    - Each investor has knowledge of a limited number of companies
    - No entity that controls the entire market
  - **Simple behavior rules**
    - Investors follow the regulatory rules of the market
    - Each investor tries to obtain the maximum possible profit

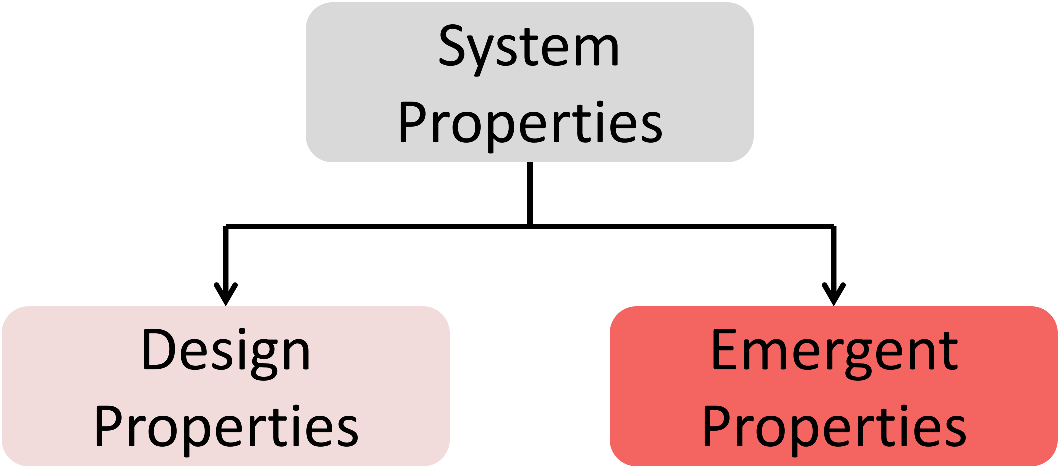
- Macro level
  - Relative security prices of companies across the world are regulated

- Related studies of emergence in stock market
  - [Chen, 2002]

Impacts of Emergence

- **Beneficial**
  - Positive additions to designed properties
    - Users adapt products to support tasks that designers never intended -> more competitive products
  - Reduce the complexity of a system [Chen, 2009]
    - Exploits self-organization
- **Harmful**
  - Leads to unforeseeable failures
  - Make a system harder to design, analyze, and control

Motivation



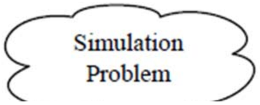
Syntactic Verification
Semantic Validation

Identify, Validation and Reason

2 proposed approaches:

- **objective-based**
- **grammar-based (this talk)**

Modeling & Simulation Lifecycle
- Design Properties

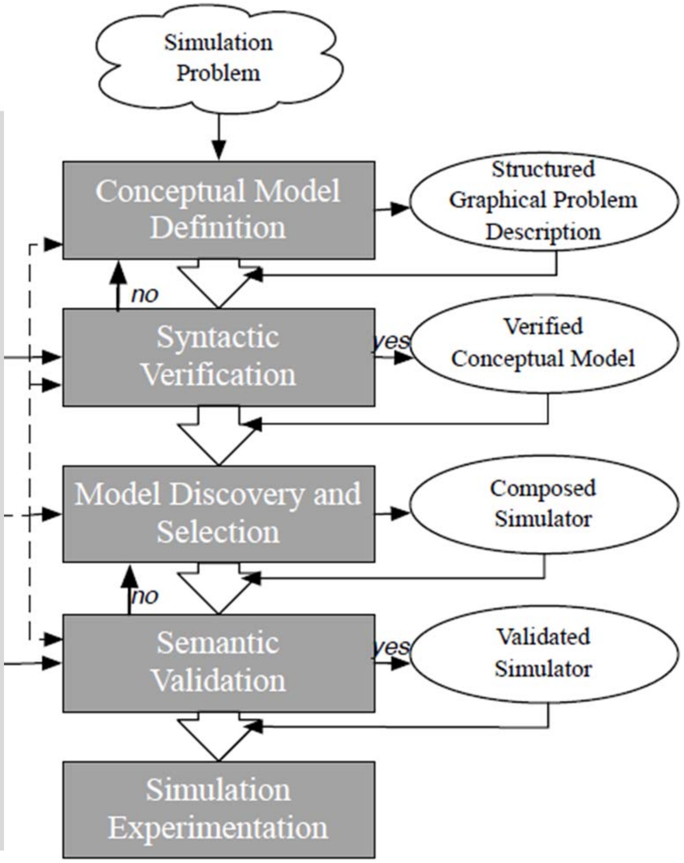


Our Work

- **Component-based modeling**
  - Component abstractions: meta-components, domain knowledge representation [ANSS 2008]
  - Model discovery and selection: partial matches [ANSS 2008, WinterSim2011]
- **Syntactic composability verification**
  - EBNF composition grammar [AMS 2007, PADS2010]
- **Semantic composability validation:**
  - Approach [PADS 2009, PADS 2011]
  - General model properties [PADS2009, PADS2010]
  - Formal time-based formalism of model execution [Wintersim 2009*]
  - Costs of validation [PADS2011, JoS2012]

* ACM SIGSIM Best PhD Student Paper Award

Traditional Modeling & Simulation Lifecycle (adapted from Banks et al., 2005)



Proposed Component-based Modeling & Simulation Life-cycle [Teo2008]

Design Properties – Composability and Semantic Validation

1. C. Szabo and Y.M. Teo, **An Approach to Semantic-based Model Discovery and Selection**, Proc of Winter Simulation Conference, pp. 3054--3066, IEEE Computer Society Press, Phoenix, Arizona, USA, Dec 11-14, 2011.

2. C. Szabo and Y.M. Teo, **An Analysis of the Cost of Validating Semantic Composability**, Proc of 25th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation, pp. 62-69, IEEE Computer Society Press, Nice, France, June 14-17, 2011. [also in Journal of Simulation, 2012]

3. Claudia Szabo, **COMPOSABLE SIMULATION MODELS AND THEIR FORMAL VALIDATION**, PhD Thesis, Department of Computer Science, National University of Singapore, 2010.

4. C. Szabo and Y.M. Teo, **On Validation of Semantic Composability in Data-driven Simulation**, Proc of 24thACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation, pp. 73-80, IEEE Computer Society Press, Atlanta, USA, May 17-19, 2010.

5. C. Szabo, Y.M. Teo and S. See, **A Time-based Formalism for the Validation of Semantic Composability**, Proc of Winter Simulation Conference, pp. 1411-1422, IEEE Computer Society Press, Austin, Texas, USA, December 13-16, 2009, (ACM SIGSIM Best PhD Student Paper Award).

6. C. Szabo and Y.M. Teo, **An Approach for Validation of Semantic Composability in Simulation Models**, Proc of 23rdACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation, pp. 3-10, IEEE Computer Society Press, New York, USA, Jun 22-25, 2009.

7. Y.M. Teo and C. Szabo, **CODES: An Integrated Approach to Composable Modeling and Simulation**, Proc of 41stAnnual Simulation Symposium, pp. 103-110, IEEE Computer Society Press, Ottawa, Canada, Apr 13-16, 2008.

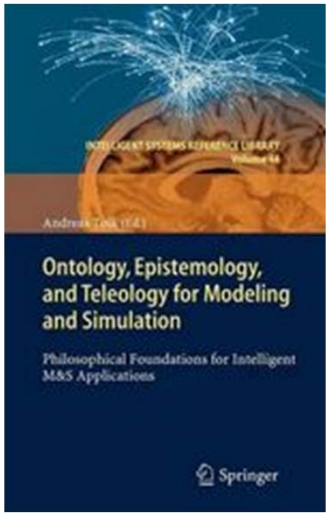
8. C. Szabo and Y.M. Teo, **On Syntactic Composability and Model Reuse**, Proc of the International Conference on Modeling and Simulation, pp. 230-236, IEEE Computer Society Press, Phuket, Thailand, March 2007, (invited paper).

Emergent Properties Detection & Validation

1. C. Szabo and Y.M. Teo, **An Integrated Approach for the Validation of Emergence in Component-based Simulation Models,** Proc of Winter Simulation Conference, IEEE Computer Society Press, Berlin, Germany, Dec 9-12, 2012.

2. C. Szabo and Y.M. Teo, **An Objective-based Approach for Semantic Validation of Emergence in Component-based Simulation Models**, Proc of 26th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation, pp. 155-162, ZhangJiaJie, China, Jul 15-19, 2012.

3. Y.M. Teo and C. Szabo, **Semantic Validation of Emergent Properties in Component-based Simulation Models**, Book Chapter in Ontology, Epistemology, and Teleology of Modeling and Simulation – Philosophical Foundations for Intelligent M&S Applications, edited by Andreas Tolk, pp 319-333, Springer-Verlag, 2013.



Emergence Perspectives

- *Philosophy*
  - Limitations of our knowledge
  - “The **whole** is greater than the **sum** of its parts.” [Aristotle]

- *Natural sciences, social sciences*
  - Explain emergence via theories in physics, biology, chemistry, and human behavior
  - Emergence is the formation of order from disorder based on **self-organization**. [Fromm, 2007], [Trianni, 2011]

  - A property is emergent if it is unexplainable from the properties of components at the **lower levels**. [Baas, 1997], [Johnson, 2008]

Emergence Perspectives

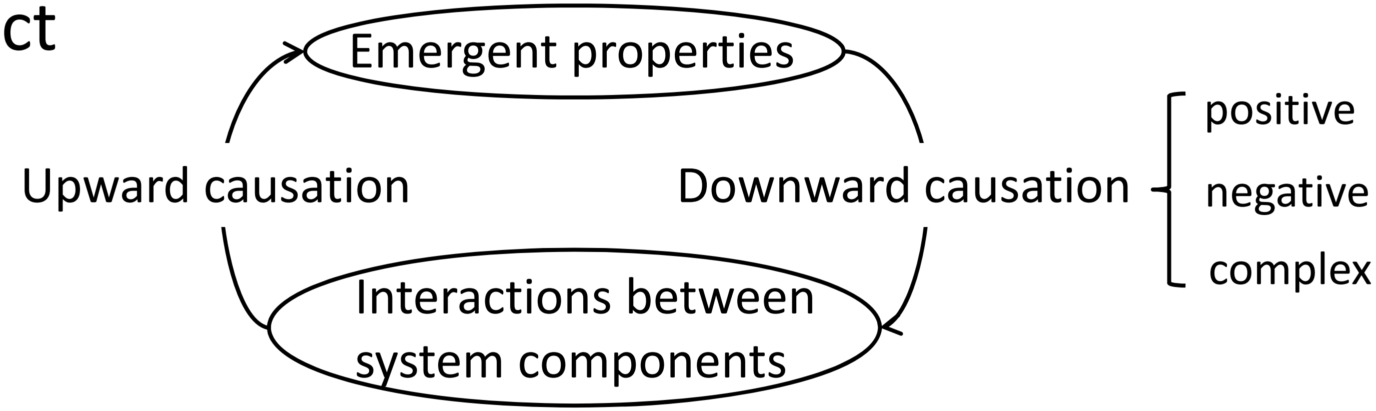
Perspective	Main Problem	Types of Emergence Studied	Concepts of Emergence	Examples	Issues
Philosophy	How intelligence emerges from unintelligent matter?	Strong emergence	More than the sum Limitations in our knowledge	Consciousness	Observer-dependent
Natural Sciences	*Biology*: How life emerges from inanimate matter? *Chemistry*: How a new substance emerges from the other ones?	Simple emergence	Self-organization, Hierarchy	Bird flocking, Ant colonies	Mainly based on self-organization
Social Sciences	How human behaviors emerges from the interactions between them?	Simple emergence		Crowd behavior	-
Computer Science	Detect and validate emergent properties Reason about emergence	Weak emergence Multiple emergence	Upward & downward causation Shifts in complexity	Link distribution in WWW, Priority inversion in O/S	Lack of complete formalism of emergence Emergence validation

Computer Science (Our) Perspective

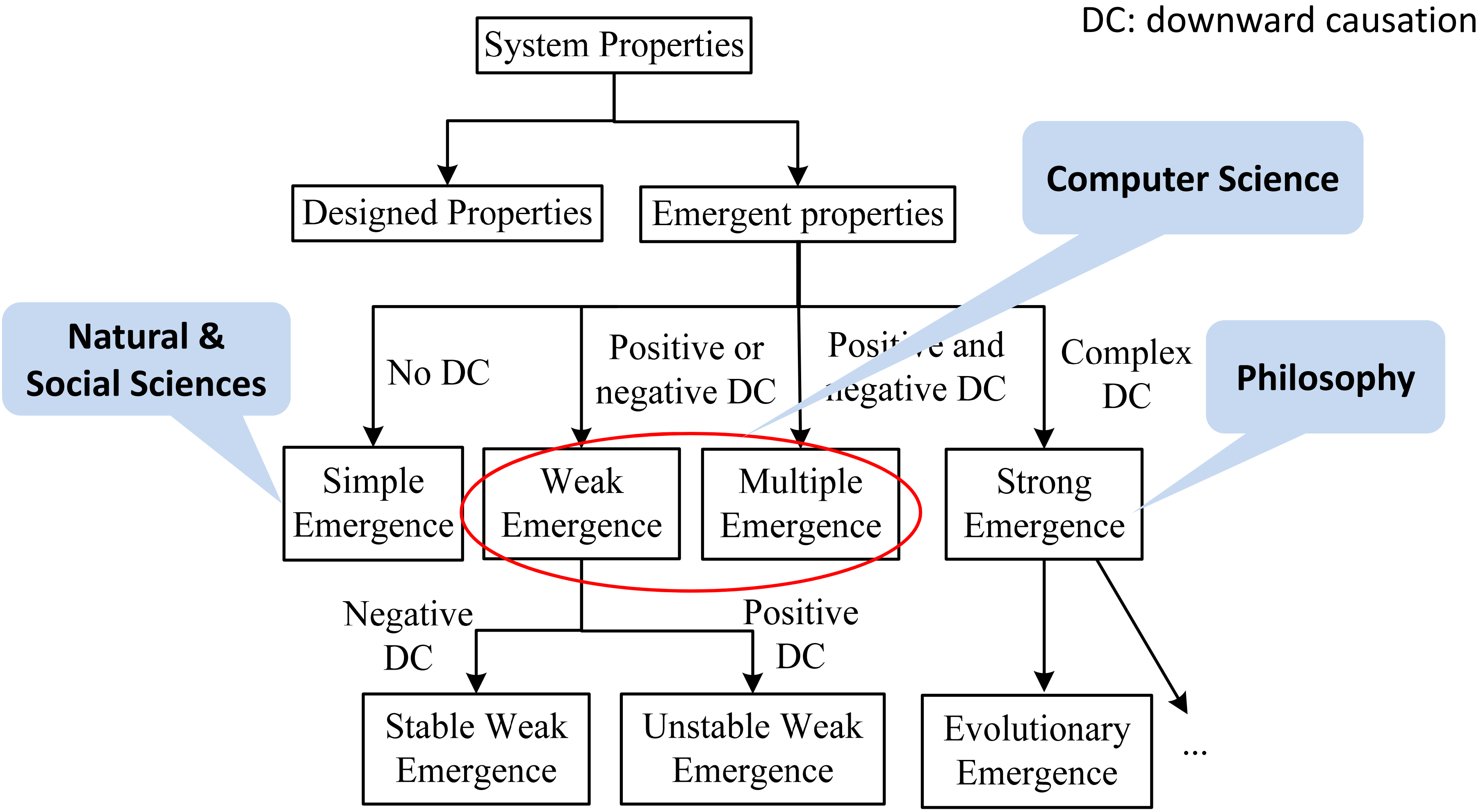
- Emergence is intrinsic to the system, observer - independent
- Emergence can be reasoned

- A property is emergent if it is both generated (**upward causation**) and autonomous (**downward causation**) from the underlying components. [Bedau, 2003]

- **Causation** is the relationship between a cause and an effect

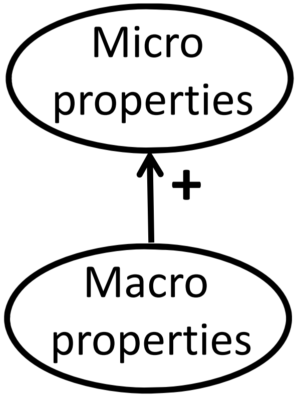


Our Downward Causation-based Emergence Classification



Simple Emergence

- Micro level
  - State of an entity is independent of
    - States of other entities
    - State of the system
    - Environment
- Macro level
  - No downward causation

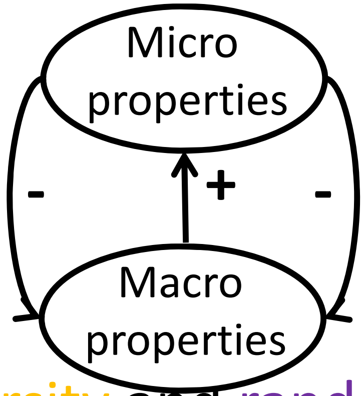


Weak Emergence

- Micro level
  - Direct interaction
    - component interactions lead to the formation of groups and clusters
    - Groups and clusters influence in turn the behavior of the components
  - Indirect interaction through environment
    - Components change the state of the system and the environment through their behaviors
    - Changes in the environment influence in turn the behavior of components

- Macro level
  - can be derived from the micro level properties but only by simulation
  - Positive **or** negative downward causation

Stable Weak Emergence



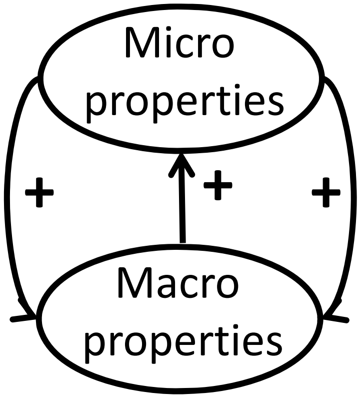
- **Negative** downward causation
- There is a balance between
  - On the one hand: exploration, diversity and randomness through "creative" **upward causation**
    - Diversity and exploration due to autonomy and unique contexts of components
  - On the other hand: exploitation, unity and order through "constraining" **downward causation**
    - Unity and exploitation impose a constraint on agents

Examples: Stable Weak Emergence

- World Wide Web
  - Mirror the huge diversity of the world population, but all web pages follow the unifying standards and constraints of the W3C and other consortiums
- Wikipedia
  - Rely on the diversity of its participants and contributors, but every participant uses the same simple editor and obeys the same easy rules
- Flocks of birds
  - Birds in a flock fly in different directions to avoid collisions, but at the same time they try to match the neighbors’ velocity and steer to the perceived centre of the flock

Unstable Weak Emergence

- A form of undesirable, negative emergence
- Positive downward causation
- Examples



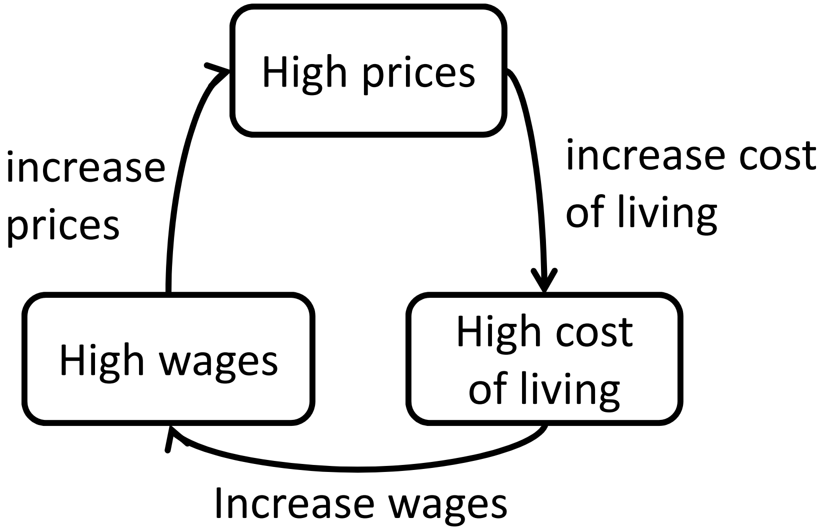
Micro properties

Macro properties

+

+

+



High prices

increase cost of living

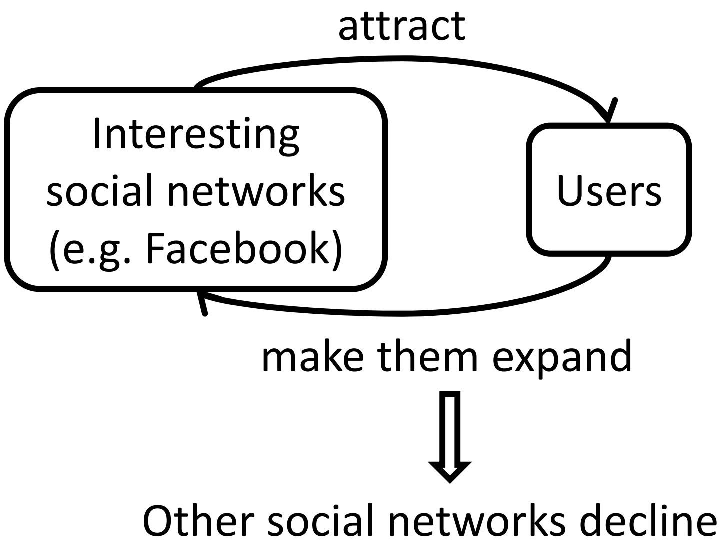
High cost of living

Increase wages

High wages

increase prices

Inflation in Economics



attract

Interesting social networks (e.g. Facebook)

Users

make them expand

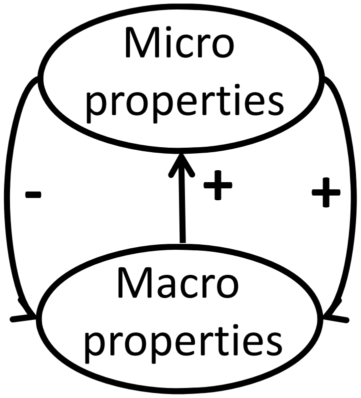
⇩

Other social networks decline

Domination of Some Social Networks

Multiple Emergence

- Both positive and negative downward causation



- Example: stock market
  - Short-range positive downward causation
    - Stocks rise → encourage investors to buy → price rises even more
  - Long-range negative downward causation
    - Buyers know that there must be finally a peak and prices will drop → deterring buyers

Strong Emergence

- **not deducible**, even in principle, from the properties of the underlying components

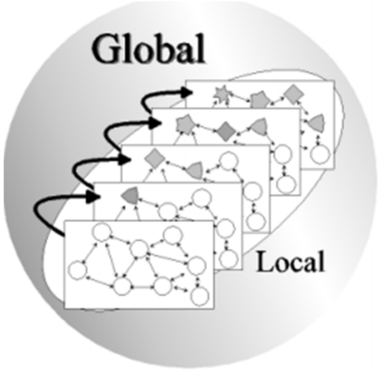
- Similar to the notion of strong emergence of Bedau [Bedau, 1997] and Chalmers [Chalmers, 2006]

- Example: **life** is a strong emergent property of genes, genetic code and nucleic/amino acids

Evolutionary Emergence

- Highly complex adaptive properties

- Entity may add to or delete from its governing rule set in response to either local or global influences



- E.g. biological entities

Research Objective

To develop a *formal approach* for *automatic identification* and *validation* of emergent properties in component-based systems

FORMALIZING EMERGENT PROPERTIES

- Emergence Formalisms
- Basic Emergence [Kubik, 2003]
- Issues in Kubik's Approach

Emergence Formalisms

- Variable-based [Fisch, 2010]
  - One or more variables describe emergent properties
  - The system run is recorded and relationships are deduced
  - Assumes prior knowledge about emergence

- Event-based [Chen, 2009]
  - Behavior as series of simple and complex events
  - Emergence defined as sequence of complex events
  - Assumes prior knowledge about emergence

- Grammar-based [Kubik, 2003]
  - L_{WHOLE}: set of properties of the system as a whole
  - L_{SUM} : set of properties obtained from the union of the parts
  - Basic Emergence (L_{ξ}) = $L_{whole} - L_{sum}$
  - How to obtain L_{sum}, L_{whole} ?

Emergence Formalisms - Comparison

Approach	Emergence Definition	Reason about Emergence
Variable-based [Fisch,2010],[Seth,2008], [Mnif,2006]	a priori	difficult
Event-based [Chen,2009]	a priori	possible
Grammar-based [Kubik,2003]	not required	possible

Formal Definition of Basic Emergence [Kubik, 2003]

- Basic emergence – “behavior reducible to agent-to-agent interactions without any evolutionary processes involved”

- Defines basic emergence using grammars

- Grammars:
  - alphabet (**V**) = {symbols}
    - Power of alphabet ($\mathbf{V}^k$) = set of strings of length **k**
  - word(**w**) = a reachable string of symbols
  - rewriting rules (**R**) translate a word to another word

Model of a System

- A multi-agent system consists of **agents** (**A**) interacting in an **environment** (**E**) that is subdivided into cells (**e**)

- **Behavior (L)** of an agent is characterized by rule-based behavioral descriptions (behavior rules **R**)

- States:
  - System state (**S**)
  - State of the environment ($\mathbf{S_E}$) and cell e ($\mathbf{s_e}$)
  - State of all agents ($\mathbf{S_A}$) and agent instance $\mathbf{A_j}$ ($\mathbf{s_j}$)

Grammar-based System Model

- A system of **n** agents, A_1, A_2, …, A_n, interacting in an environment (E) consisting of **c** cells is defined as

$$\mathbf{System = (V_A, V_E, A_1, A_2, \ldots, A_n, S(0))}$$

where

- **V_A**: set of possible agent states
- **V_E**: set of possible cell states
- **V** = $V_A \cup V_E$
- **A_j**: agent instance j ($1 \leq j \leq n$)
- **S(t)** $\in V^{c+n}$: system state at time t, and S(0) denotes the initial system state at time 0

Environment

- Cell (e)
  - $\mathbf{V_e}$: set of possible states of cell e
  - $\mathbf{s_e(t)} \in V_e$: state at time t, and $\mathbf{s_e(0)}$ denotes the initial state at time 0

- Environment (E)
  - $\mathbf{V_E} = \bigcup_{e=1}^{c} V_e$
  - $\mathbf{S_E(t)} \in V_E{}^c$: state at time t, and $\mathbf{S_E(0)}$ denotes the initial state at time 0

Agents

- Agent A_j is defined as

$$\mathbf{A_j = (V_j, R_j, s_j(0))}$$

where

- $\mathbf{V_j} \subseteq V_A$: set of possible states of A_j
- $\mathbf{R_j}$: set of behavior rules for A_j
- $\mathbf{s_j(t)} \in V_j$: state of A_j at time t, and $\mathbf{s_j(0)}$ denotes the initial state of A_j at time 0

- Behavior rules of A_j (R_j):

$$\mathbf{R_j}: s_j(t) \rightarrow s_j(t+1) \text{ or } \mathbf{R_j}: V_j \rightarrow V_j$$

Issues in Kubik's Approach

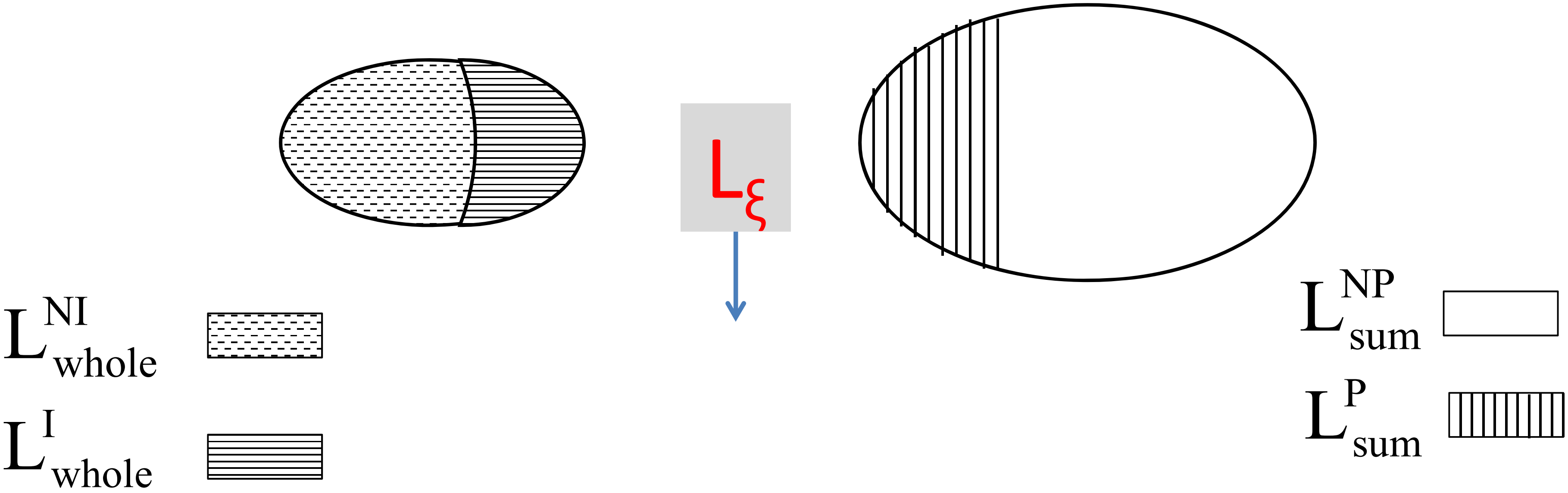
- Broad definition of emergence
- Emergent formalism:

1. agent types
   [introduce agent type, A_{ij} type i $(1 \leq i \leq m)$]
2. mobile agents
   [define mobility as attributes of agents:
   $P_i = P_{i\_mobile} \cup P_{i\_others}$]
3. Open system
   Game of life example, agents exist throughout the observation period, and number of agents (n) = number of cells (c)
   [agents can enter and leave system – model open system, n varies over time]

PROPOSED GRAMMAR-BASED APPROACH

- Objectives
- Overview of Approach
- System Model
- Proposed Grammar
- Example: Bird Flocking (extended Boids Model)

Proposed Emergence Definition



Kubik: $L_\xi = L_{whole} - L_{sum}$

Proposed: $L_{whole} = L^{NI}_{whole} + L^{I}_{whole}$

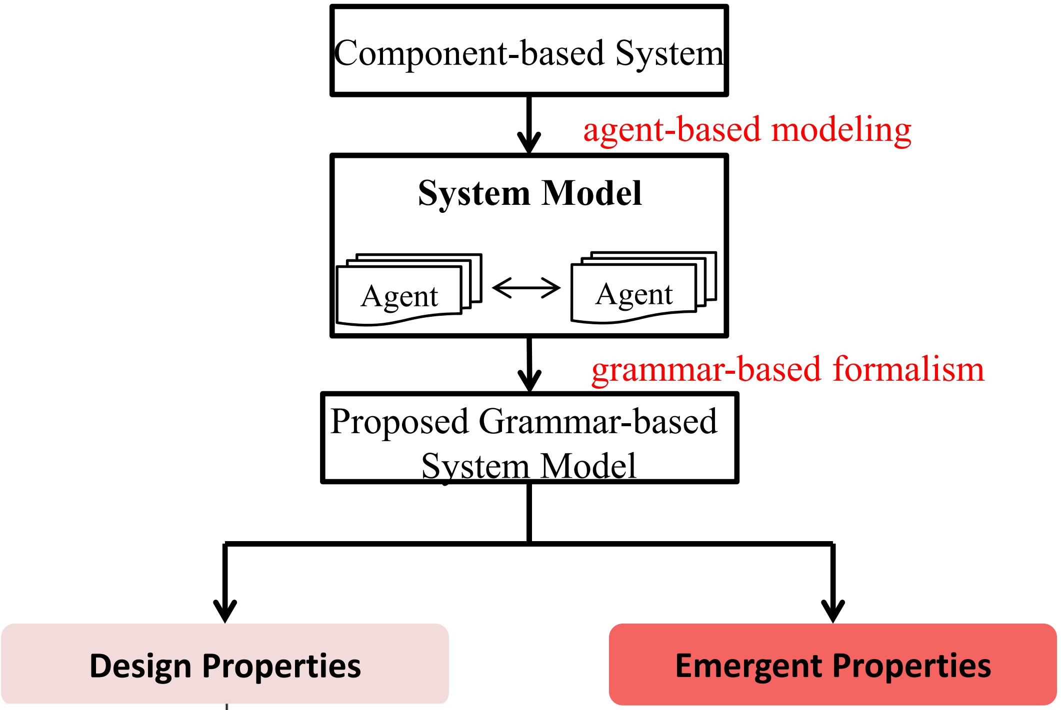
$L_{sum} = L^{NP}_{sum} + L^{P}_{sum}$

$L_\xi = L^{I}_{whole} - L^{P}_{sum}$

Design Considerations

- Computer science perspective
- Basic (weak) emergence
- Component-based systems - modeled as a multi-agent system
- Extended grammar-based formalism: agent types, mobile agents, open systems
- Automatic detection, validation and reasoning of emergence

Overview of Approach



Component-based System

agent-based modeling

System Model

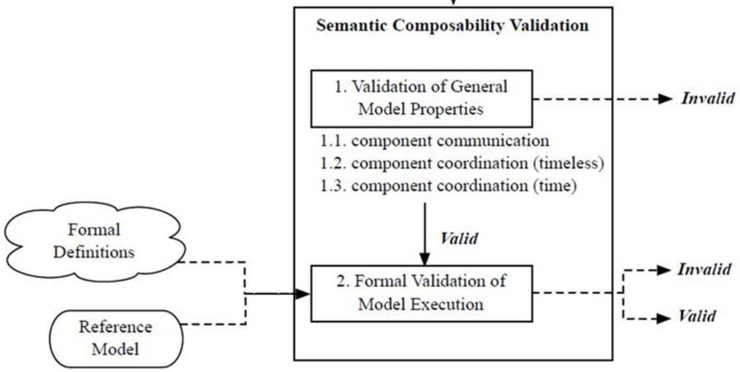
Agent ↔ Agent

grammar-based formalism

Proposed Grammar-based System Model

Design Properties

Emergent Properties



Detection, Validation and Reason

System Model

- A system is modeled as a multi-agent system consisting of different types of **agents** (**A**) interacting in an **environment** (**E**). E is modeled as a 2D grid that is subdivided into units called cells (**e**)

- **Behavior (L)** of an agent is characterized by rule-based behavioral descriptions (behavior rules (**R**))

- States:
  - – System state (**S**)
  - – State of the environment ($\mathbf{S_E}$) and cell e ($\mathbf{s_e}$)
  - – State of all agents ($\mathbf{S_A}$) and agent $\mathbf{A_{ij}}$ ($\mathbf{s_{ij}}$)

Proposed Grammar

A system consisting of *m* agent types and *n* agents A_{1n_1}, …, A_{mn_m} (n = n_1 + … + n_m, n_i agents of type i) interacting in an environment (2D grid) consisting of *c* cells is defined as

$$\mathbf{GBS = (V_A, V_E,} A_{1n_1}, \ldots, \boldsymbol{A_{mn_m}}, \mathbf{S(0))}$$

where

- **A_{ij}**: agent instance j ($1 \leq j \leq n_i$) of type i ($1 \leq i \leq m$)
- $\mathbf{V_A} = \bigcup_{i=1}^{m} V_{A_i}$: set of possible agent states for all agent types, and $\mathbf{V_{A_i}}$ denotes the set of possible states for agents of type i
- **V_E**: set of possible cell states
- **V** = $V_A \cup V_E$
- **S(t)** $\in V^{c+n}$: system state at time t, and **S(0)** denotes the initial system state at time 0

Agents

- Agent A_{ij} $(1 \le i \le m, 1 \le j \le n)$, is defined as

$$\mathbf{A_{ij} = (P_i, R_i, s_{ij}(0))}$$

where

- $\mathbf{P_i}$: set of attributes for agents of type *i*
  $P_i = P_{i\_mobile} \cup P_{i\_others}$
  $\mathbf{P_{i\_mobile}} = \{x | x \text{ is an attribute that models mobility}\}$
  $\mathbf{P_{i\_others}} = P_i \setminus P_{i\_mobile}$

- $\mathbf{V_{A_i}}$: set of possible states for agents of type *i*

- $\mathbf{R_i}$: set of behavior rules for agents of type *i*
  $R_i = R_{i\_mobile} \cup R_{i\_others}$
  $\mathbf{R_{i\_mobile}}$: $V_{A_i} \text{ -> } V_{A_i}$
  $\mathbf{R_{i\_others}} = P_i \setminus P_{i\_mobile}$

- $\mathbf{s_{ij}(t)} \in V_{A_i}$: state of A_{ij} at time *t*, and $\mathbf{s_{ij}(0)}$ denotes the initial state of A_{ij} at time 0

$\mathbf{L}^{\mathbf{I}}_{\mathbf{whole}}$

- Behavior of the system as a whole

$$\mathbf{L}^{\mathbf{I}}_{\mathbf{whole}} = \{\mathbf{w} \in \mathbf{V}^{\mathbf{c+n}} \,|\, \mathbf{S(0)} \Rightarrow^{*}_{\mathbf{GROUP}} \mathbf{w}\}$$

- Includes interactions among agents (GROUP)
- $\mathrm{L}^{\mathrm{I}}_{\mathrm{whole}}$ returns a set of words (**w**) that represents the set of system states reachable from an initial system state, S(0)

$\mathbf{L^{P}_{sum}}$

- Sum of agent behaviors:

$$\mathbf{L^{P}_{sum} = superimpose(L(A_{1n_1}),\ldots, L(A_{mn_m}))}$$

returns a set of words resulting from superimposing words of behaviors of individual agents

- $L(A_{ij})$ - behavior of an agent A_{ij}, is defined as

$$\mathbf{L(A_{ij}) = \{w \in V^{c+1} \mid (s_{ij}(0) \cup S_E(0)) \Rightarrow^{*} w\}}$$

considering only this agent in the system, behavior returns a set of words (**w**) that represents the set of system states reachable from an initial system state, $s_{ij}(0) \cup S_E(0)$

Example: Bird Flocking

- Boids model [Reynolds, 1987] captures the motion of bird flocking and is a seminar example for studying emergence [Szabo, 2012], [Chan, 2011]
- Extension of the boids model with two types of birds, ducks and geese

Agent-based Model

- The system is modeled as a multi-agent system consisting of two types of agents (duck and goose) interacting in an **environment** represented as a **2D grid** (**E**) that is subdivided into cells (**e**) of equal size.

- **Behavior (L)** of an agent is characterized by rule-based behavioral descriptions (behavior rules (**R**))

- States:
  - System state ($\mathbf{S}$)
  - State of the environment ($\mathbf{S_E}$) and cell e ($\mathbf{s_e}$)
  - State of all agents ($\mathbf{S_A}$) and agent $\mathbf{A_{ij}}$ ($\mathbf{s_{ij}}$)

Environment

- 2D grid of **8**x**8** cells
- Cell states
  - **occupied** (by a bird)
  - **free** (not occupied)
- A cell fits one duck or one goose
- Neighborhood distance: 3 cells

Agents – Ducks and Geese

- Attributes
  - Color: duck – red, goose - blue (for visualization)
  - Mobility:
    - Position: position of agent in the environment
    - Velocity:
      - speed: 0, 1, or 2 cells per step
      - 8 directions: north, north-east, east, south-east, south, south-west, west, north-west
      - **represented as a vector (discrete)**

- Behavior Rules [Reynolds 1987]
  - *Separation*: avoid collision with nearby birds
  - *Alignment*: fly as fast as nearby of the same type
  - *Cohesion*: stay close to nearby of the same type

Proposed Grammar-based Formalism

$$\mathbf{GBS = (V_A, V_E, A_{11}, \ldots, A_{15}, A_{21}, \ldots, A_{25}, S(0))}$$

where

- Agent types: 1 denotes duck, 2 denotes goose
- $\mathbf{A_{1j}}$ is duck instance j ($1 \leq j \leq 5$), and A_{2j} is goose instance j ($1 \leq j \leq 5$)
- $\mathbf{V_A} = \mathrm{V}_{\mathrm{A}_1} \cup \mathrm{V}_{\mathrm{A}_2}$: set of possible states for the ducks ($\mathrm{V}_{\mathrm{A}_1}$) and the geese ($\mathrm{V}_{\mathrm{A}_2}$)
- $\mathbf{V_E}$: set of possible cell states
- $\mathbf{V} = V_A \cup V_E$
- **S(t)** $\in V^{64+10}$: system state at time t, and **S(0)** denotes the initial system state at time 0

Environment

- Cell (e)
  - $\mathbf{V_e}$ = {o, f}, where occupied (o), free (f)
  - $\mathbf{s_e(t)} \in V_e$

- Environment (E)
  - $\mathbf{V_E} = \bigcup_{e=1}^{64} V_e = \{o, f\}$
  - $\mathbf{S_E(t)} \in V_E^{64}$

Agents – Ducks

- Duck instance A_{1j} $(1 \le j \le 5)$ is defined as

$$\mathbf{A_{1j} = (P_1, R_1, s_{1j}(0))}$$

where

  - $P_1 = P_{1\_mobile} \cup P_{1\_others}$
    $\mathbf{P_{1\_mobile}} = \{position(g_{1j}), velocity(v_{1j})\}$, $\mathbf{P_{1\_others}} = \{color\}$

  - $\mathbf{V_{A_1}} = \{(x,y) | 1 \le x \le 8; 1 \le y \le 8\} \times \{(\alpha,\beta) | -2 \le m \le 2; -2 \le n \le 2\} \times \{red\}$

  - $R_1 = R_{1\_mobile} \cup R_{1\_others}$
    $\mathbf{R_{1\_mobile}}, \mathbf{R_{1\_others}} = \emptyset$

  - $\mathbf{s_{1j}(t)} \in V_{A_1}$

Similarly for Geese!

Behavior Rules for Ducks – R_{1_mobile}

- For duck instance A_{1j} ($1 \leq j \leq 5$) at time t with position $g_{1j}(t)$ and velocity $v_{1j}(t)$:
  - **Update position:**
    $g_{1j}(t+1) = g_{1j}(t) + v_{1j}(t+1)$
  - **Update velocity:**
    $v_{1j}(t+1) = v_{1j}(t) + \text{separation}(A_{1j}) + \text{alignment}(A_{1j}) + \text{cohesion}(A_{1j})$

Similarly for Geese!

Separation Rule

- Goal: avoid collision with nearby birds
- How: if duck **a** is close to another bird **b**, i.e. within ε cells, then **a** flies away from **b**

$$\text{separation(a)} = \sum_{distance(a,b) \leq \varepsilon} a.position - b.position$$

```
separation(duck a)
   vector c = 0;
   for each boid b
        if |a.position - b.position| ≤ ε then
           c = c - (b.position - a.position)
 return c
```

Alignment Rule

- Goal: fly as fast as nearby ducks

- How: change velocity of duck a by λ% towards the average velocity of its neighboring ducks

$$\text{alignment(a)} = \left(\left(\sum_{\substack{duck(b)\\ neighbor(a,b)}}^{k} b.velocity\right)/k \; - a.velocity\right)/\lambda$$

```
alignment(duck a)
   vector c = 0;
   integer k = 0;
   for each neighboring duck b
         k = k + 1;
         c = c + b.velocity;
    endfor
    c = c / k;
return (c - a.velocity) / λ
```

Cohesion Rule

- Goal: stay close to nearby ducks
- How: move duck a by γ% towards the center of its neighboring ducks

$$\text{cohesion(a)} = \left(\left(\sum_{\substack{duck(b)\\ neighbor(a,b)}}^{k} b.position\right)/k - a.position\right)/\gamma$$

```
cohesion(duck a)
    vector c = 0;
    integer k = 0;
    for each neighboring duck b
        k = k + 1;
        c = c+ b.position;
    endfor
    c = c / k;
return (c - a.position) / γ
```

Example of Emergence Detection

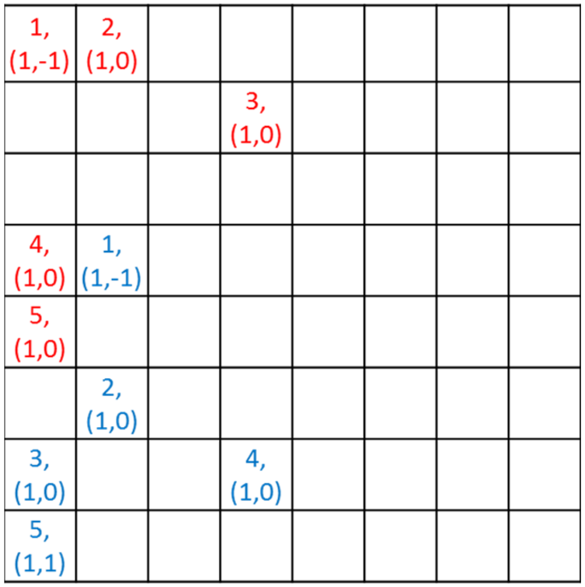
- Objective: show how flocking of birds, a known emergence, is detected

- **Flocking -** at least 4 birds of the same type fly together

- Together - each bird has at least one immediate neighbor of the same type

Example of Emergence Detection

- Given an initial system state:



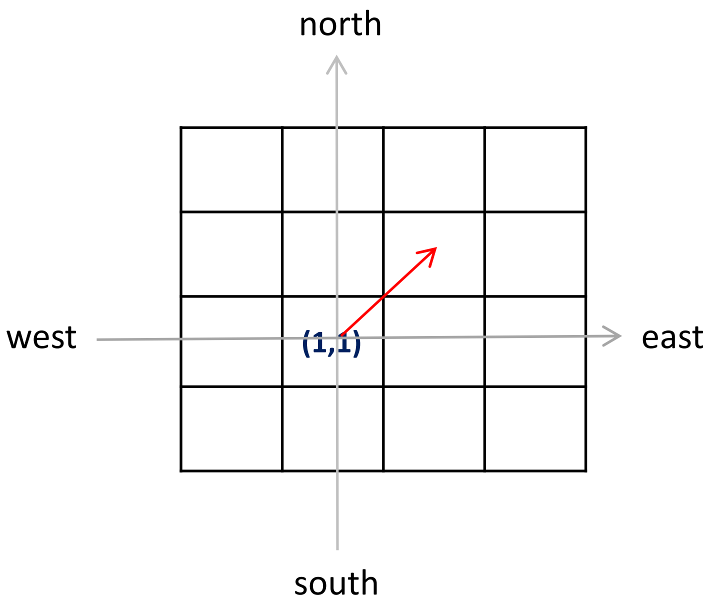
1, (1,-1)	2, (1,0)						
			3, (1,0)				
4, (1,0)	1, (1,-1)						
5, (1,0)							
	2, (1,0)						
3, (1,0)			4, (1,0)				
5, (1,1)							

where red denotes duck and blue denotes goose,
<j, (α,β)> denotes a bird, with velocity (α,β)

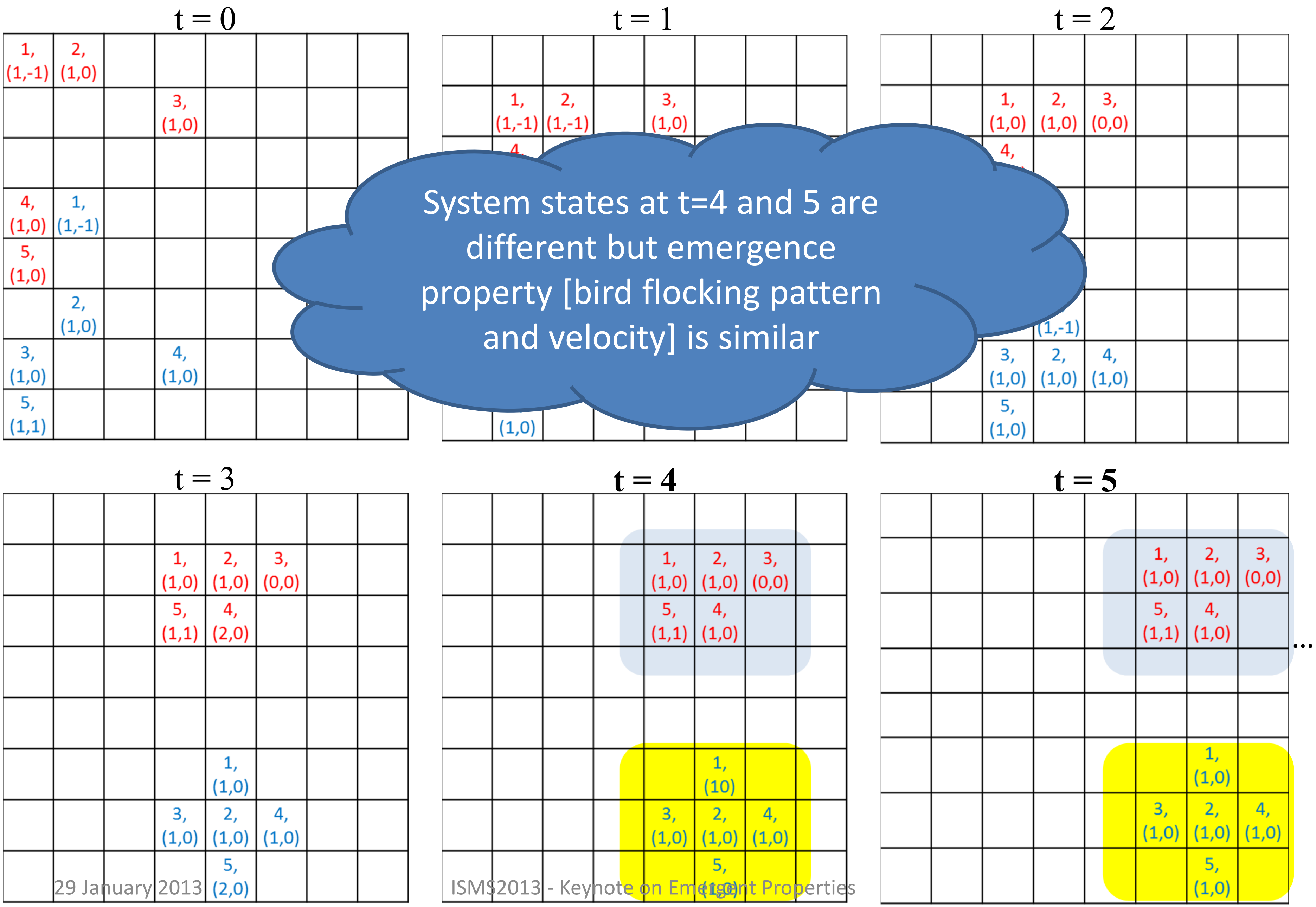
- Determine basic emergence = $L^{I}_{whole} - L_{sum}$

Vector Representation of Velocity

Direction	**Speed**		
	0	1	2
North	(0,0)	(0,1)	(0,2)
North-east	(0,0)	(1,1)	**(1,2), (2,1), (2,2)**
East	(0,0)	(1,0)	(2,0)
South-east	(0,0)	(1,-1)	(1,-2), (2,-1), (2,-2)
South	(0,0)	(0,-1)	(0,-2)
South-west	(0,0)	(-1,-1)	(-1,-2), (-2,-1), (-2,-2)
West	(0,0)	(-1,0)	(-2,0)
North-west	(0,0)	(-1,1)	(-1,2), (-2,1), (-2,2)



L_{whole}



System states at t=4 and 5 are different but emergence property [bird flocking pattern and velocity] is similar

L^{I}_{whole}

- Observation
  - At t=4 and t=5, the system states are different but the **emergent property** is the same [bird flocking pattern and velocity]
  - S(12) = S(4)

- Hence by definition,

$$L^{I}_{whole} = \{S(0), S(1), S(2), S(3), S(4), S(5), \ldots, S(11)\}$$

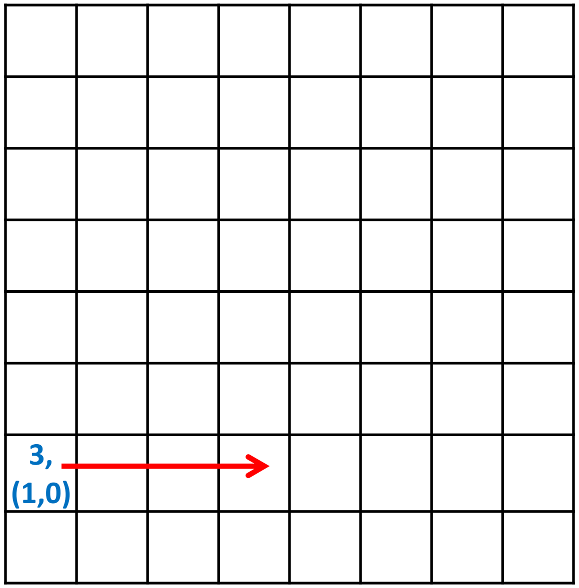
L_{sum}

- L_{sum} = superimpose($L(A_{11})$, …, $L(A_{15})$, $L(A_{21})$, …, $L(A_{25})$)

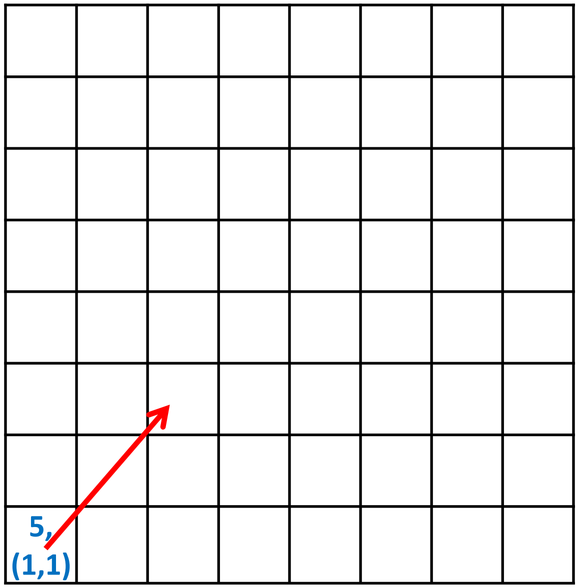
- For illustration, consider two geese:

  L_{sum} = superimpose($L(A_{23})$, $L(A_{25})$)

$s_{23}(0)$

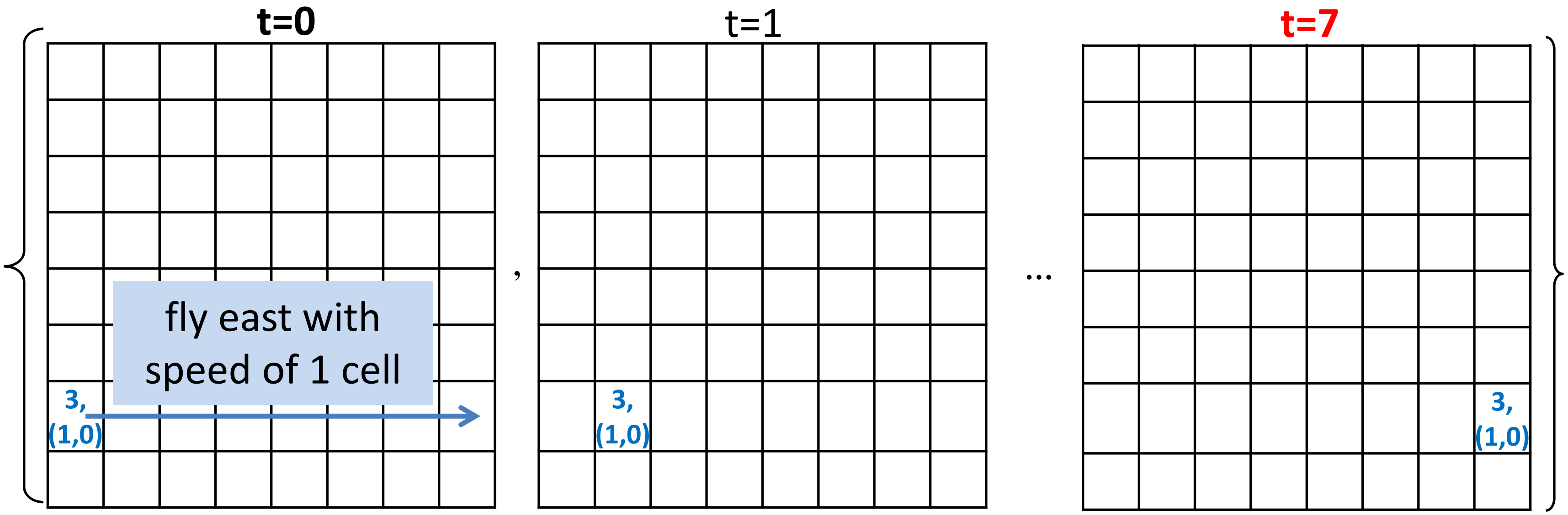


$s_{25}(0)$



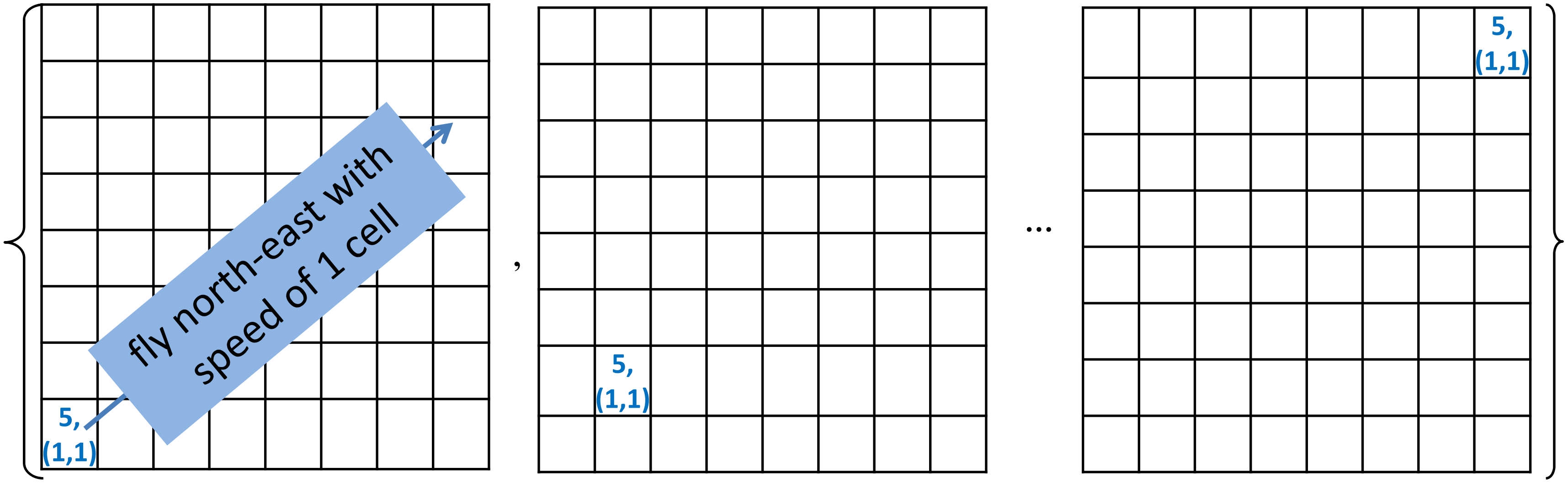
$L(A_{23})$ and $L(A_{25})$

$L(A_{23}) =$



$= \{s_{23}(0), s_{23}(1), \ldots, s_{23}(7)\}$ where $s_{23}(0) = \{f, f, \ldots, (\text{blue}, (1,2), (1,0)), f, f, \ldots, f\}$

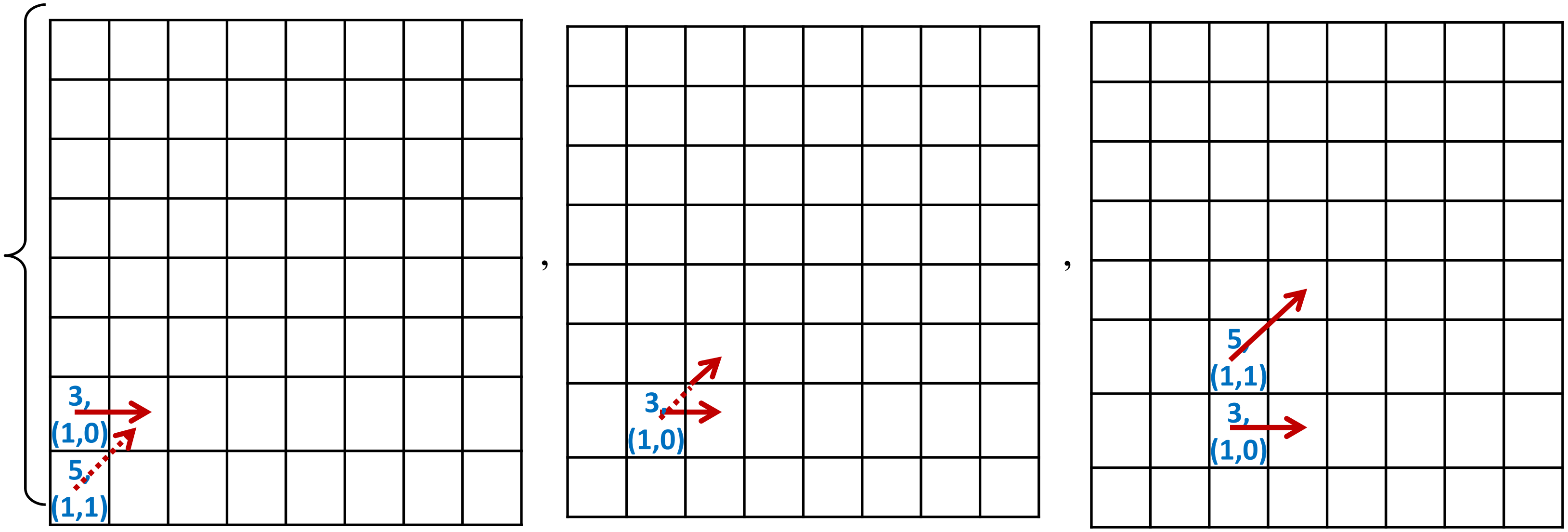
$L(A_{25}) =$



$= \{s_{25}(0), s_{25}(1), \ldots, s_{25}(7)\}$

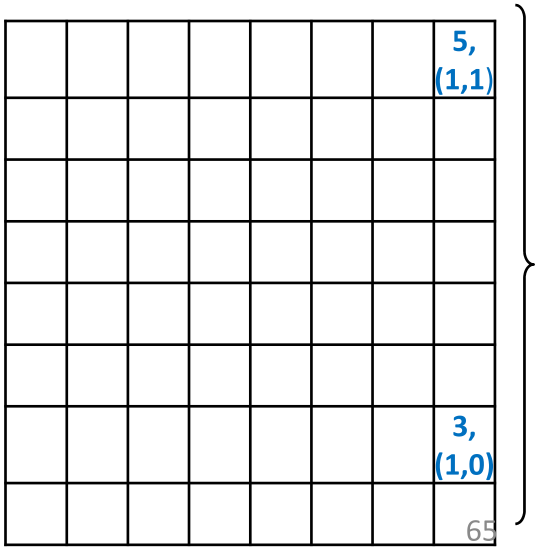
L_{sum} = superimpose(L(A_{23}), L(A_{25}))

= **L(A_{23}) superimpose (L(A_{25}))** U L(A_{25}) superimpose (L(A_{23}))



L(A_{23}) superimpose (L(A_{25}))
= {(f,f,f…(blue, (1,2), (1,0)),…(blue, (1,1), (1,1)),…),
(f,f,f…(blue, (2,2), (1,0)),…),
(f,f,f…(blue, (3,3), (1,1)),…(blue, (3,2), (1,0)),…),
….}

…,



Emergent Property States – Flocks of birds

- L_ξ = {S(1), S(2), S(3), S(4), …, S(11)}
- Based on flocking definition (at least 4 birds of the same type fly together):
  1. known emergent states - S(2), S(3), S(4), …, S(11)
  2. unknown emergent state - S(1)

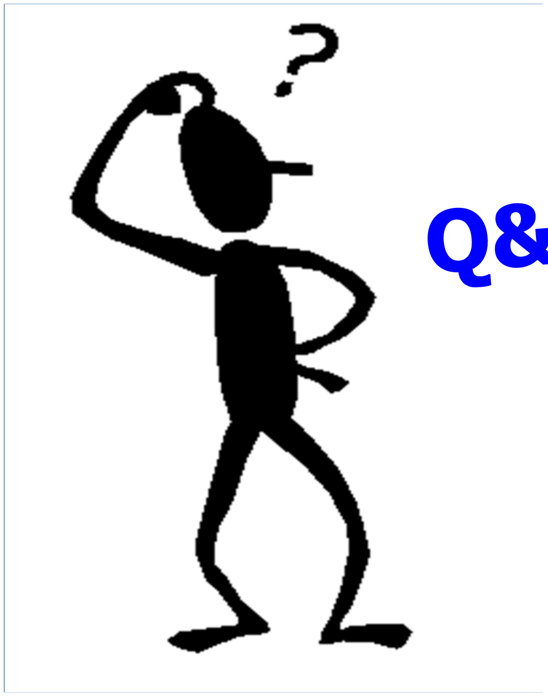
Conclusion

- Grammar-based approach to identify and validate emergent properties
  - Tighter definition of emergence
  - Emergent formalism: agent types, mobile agents, open systems
- Extended boids model
  - Large problem size can lead to state explosion problem
  - Mobile agents have more impacts on complexity

Further Work

- How to determine $\mathrm{L}_{\mathrm{sum}}^{\mathrm{P}}$?
- Defining known emergence and making sense of unknown emergence
- Reasoning of emergence – useful, harmful, …
- Applications – road traffic networks, concurrent program verification, social networks, …

Thank you



Q&A

http://www.comp.nus.edu.sg/~teoym
Email: teoym@comp.nus.edu.sg

References

1. N. A. Baas and C. Emmeche. On emergence and explanation. Intellectica, pages 67–83, 1997.
2. M. A. Bedau. Downward causation and the autonomy of weak emergence. Principia 3, 3:5–50, 2003.
3. C. C. Chen. Complex Event Types for Agent-Based Simulation. PhD thesis, University College London, 2009.
4. D. Fisch, M. Janicke, B. Sick, and C. Muller-Schloer. Quantitative emergence – a refined approach based on divergence measures. In 4th IEEE International Conference on Self-Adaptive and Self-Organizing Systems, pages 94 –103, 2010.
5. J. Fromm. Types and forms of emergence. http://arxiv.org/abs/nlin.AO/0506028, 2007.
6. H. M. Hinton. Under-specification, composition and emergent properties. In Proceedings of the 1997 Workshop on New Security Paradigms, pages 83–93, New York, NY, USA, 1997.

References

7. A. Kubik. Toward a formalization of emergence. Artificial Life IX, 9(1):41–65, 2003.
8. J. Johnson. Multidimensional events in multilevel systems. In The Dynamics of Complex Urban Systems: An Interdisciplinary Approach, pages 311–334. Physica-Verlag, 2008.
9. M. Mnif and C. Muller-Schloer. Quantitative emergence. In IEEE Mountain Workshop on Adaptive and Learning Systems, pages 78–84, Logan, UT, USA, 2006.
10. K. Nagel and M. Schreckenberg. A cellular automaton model for freeway traffic. Journal de Physique I, 2(12):2221–2229, Dec. 1992.
11. V. Trianni and S. Nolfi. Engineering the evolution of self-organizing behaviors in swarm robotics: A case study. Artificial Life, 17(3):183–202, 2011.
12. A. K. Seth. Measuring emergence via nonlinear granger causality. Artificial Life XI, 324(1):545–552, 2008.