
Nikos A. Salingaros
University of Texas at San Antonio
Lecture 7

A. Biologically-inspired computation.
B. Genetic algorithms.
C. Computation versus memory retrieval.
D. Evolutionary regression.
A. Biologically-inspired computation.

- An algorithm that follows or mimics biological processes
- Computation guided by biological constraints towards adaptivity
- Algorithm uses biological sensors to endow the configuration with “life”
Process of optimization

- Biological systems optimize themselves via selection
- What is being optimized is either physical form, or behavior, or both
- Evolution is therefore a computation with bio-geometrical constraints on the underlying structure
Example: laying out the plan of a new building

• Suppose we are going to build on a green site, or on an existing lot
• Walk the lot and identify the centers
• Totally emotional, not mathematical
• Use sensory feedback to fit building components into the site

• Influenced by every detail already there
Best initial fit

- Match typological elements “entry”, “main rooms”, “windows”, etc. with where emotion and intuition tells us they belong
- Compute the project’s morphology using cognitive resonance — using our own body’s exquisite biological sensors
- Mark the site with sticks, flags on posts, cardboard panels, chalk on ground, etc.
Participatory design

- These steps are more accurate when taken with the participation of a group of people — an on-site charrette
- Architects working together with eventual users make decisions ON THE SITE
- Reach a consensus about shapes, paths, placements, and configuration
Radical propositions

1. *Consider everything existing on the site* — don’t just wipe everything clean
2. *Make value judgments* about which existing elements are life-enhancing
3. Save those and discard the rest
4. Decide to keep a tree, a large rock, but remove some earth and older structures
Preparing the plan

• Tens of millions bits of information influence the plan subconsciously!
• Less information available in the office
• The entire building, or set of buildings, is conceived on the site
• When on-site design is finished, measure the physical markers on the site to accurately draw the first plan
Method as formulated by Nili Portugali (Tel-Aviv, Israel)

- Adaptive design method used by Christopher Alexander and his students
- Building *grows out of* the site, and is not *imposed* upon the site
- Plan is first encoded on the site itself
- — using low-tech markers and props!
“The Act of Creation and the Spirit of a Place”
Step 1

• Identify the main centers of the site:
  • — focal points of geometry
  • — focal points of activity
• These centers reinforce each other
• Imagine performing each activity in the building already erected — where does it feel to be the best spot for it?
Step 2

• Identify the main and secondary entrances to the site
• Establish the most natural connection to the existing surroundings: roads, buildings, etc.
• Use Alexandrine patterns to define approach and entry so that it is enticing
Step 3

• Internal paths arise from linking the localized activity centers
• *Sequence*: centers first, paths second
• Define the paths while actually walking on the site
• Approach from the outside involves one or more paths
Step 4

- Decide on the rough boundaries of each center of activity
- Those boundaries are now beginning to fix the geometry of the plan
- Choose the entry point for each center
- This determines the path structure more accurately
Step 5

• Decide where to place structures
• Structures are there to accommodate the activity and geometrical centers, not the other way around!
• The most intrusive structures go up in the least attractive parts of the site: Alexander’s **Pattern 104, “Site Repair”**
Step 6

- Stake out the useful urban space
- Decide outdoor activities: walking on a path; sitting outside — and reinforce them
- Coherent urban space is defined by walls, and is not just external left-over space
- Chapter 2 of “Principles of Urban Structure” — 20C ruined urban space!
Step 7

- Lay out the future indoor spaces
- Use all the appropriate patterns from Alexander’s “A Pattern Language”
- Finally, only now decide on the walls!
- Once this preliminary work has been done, proceed to develop the building
Unexpected form

• What are the indications of success towards an adaptive design?
• Examined back in the office, the plan feels RIGHT, and it also looks rather UNEXPECTED — a positive quality
• This means that it is an evolved design, but not an imposed design
• Could not have been made up
“Unexpected” but not “absurd”

• I distinguish between our unexpected designs and the absurd forms of contemporary architecture

• Evolved design in perfectly adapted, not made up arbitrarily!

• Absurd forms do not adapt to anything, not even to the program brief!
Proceed into design

• Make a list of the project’s relevant socio-geometric patterns from Alexander’s “A Pattern Language”
• Derive new patterns needed by the project, if those are not yet developed
• Look to precedent, tradition, successful solutions under similar circumstances
Then combine the patterns

• Combinatoric method from Chapter 8 of “Principles of Urban Structure”
• Combine patterns acting on smaller scales together hierarchically into higher-level patterns
• Link all scales together through the patterns acting on different scales
• Do not privilege the largest scale!
Proceed into design (cont.)

- Decide on your form language
- Tectonic and ornamental vocabulary
- Adopt a pre-existing form language suited to the locality (memory), or create your own form language
- Make sure to use a rich form language — otherwise design cannot be adaptive
Calibrate the form language

• Chapter 11 of “A Theory of Architecture” shows that a rich form language will never contradict the pattern language! (Otherwise, discard)

• Express all tectonic elements in the chosen form language

• Pay special attention to smallest scales
Now design the building

- Follow recursive rules for creating centers (*lecture 5*)
- Previous sequence of seven steps for laying out the ground plan reflects the general approach on all scales
- Adaptive emotion-based computation continues all the way down to the smallest scales of ornamentation
B. Genetic algorithms

- An algorithm is a list of instructions
- An algorithm can evolve using a Darwinian processes that selects for success
- Start with an algorithm that works
- Introduce random variations in the code
- Millions of new variants won’t work
- One variant may work, and could be better than the original algorithm
Monsters from genetic algorithms

- Darwinian process of selection can indeed generate monsters — Chapter 10 of “A Theory of Architecture”
- Occurs when you select forms for “cuteness” or “strangeness”, not for adaptivity to human needs
- Is the system evolving towards a higher intelligence, or into a monster?
The key here is selection

- In contemporary architecture, selection is usually based on generating forms that induce anxiety!
- Anxiety is misinterpreted as the thrill of exciting new forms, but it actually affects our body negatively
- Adaptive design solutions don’t give the same thrill that triggers anxiety
Selection in design

• My friends and I apply selection criteria based on human needs and sensibilities — result generates a feeling of wellbeing

• Our selection criteria are the following adaptive CONSTRAINTS (lecture 5):

• — *pattern languages; universal scaling; universal distribution; centers; fifteen fundamental properties, etc.*
Incomprehensible complexity

- Computer scientist W. Daniel (Danny) Hillis has bred sorting algorithms
- These genetic algorithms are faster than any written by human programmers
- HILLIS DOES NOT UNDERSTAND HOW THEY WORK!
- Their complexity is as long as themselves
Collective intelligence

• Traditional design methods have evolved through Darwinian processes
• Now stored in built memory as part of the traditional environment
• Their complexity equals their code
• No shortcut to understanding how they work — Chapter 10 of “Principles of Urban Structure”
Correctness of traditional architectural and urban typologies

- Evolved solutions — they work even if we don’t understand exactly why
- Known to solve all problems of a particular type, and to always give an adaptive solution
- These proven results of evolved algorithms must be preserved for use!
Algorithmic ambiguity

• Let’s invent a new design algorithm…
• But just because an algorithm gives output, that does not mean its output represents any valid results
• Results may be nonsensical, or toxic
• Validation criteria must come from OUTSIDE the algorithm itself
Rule of thumb

- Design computations must be validated by checking against EVOLVED SOLUTIONS
- Traditional design gives results known independently to be correct
- Use these as a check before proceeding to more innovative design problems for the contemporary world
C. Computation versus memory retrieval

• MY CLAIM: *Most design relies upon a stored memory bank*

• Even when designers think they are being most innovative, subconsciously, they are still drawing upon memory

• It is therefore crucial to have a bank of evolved solutions to draw from!
Example: some famous architects

- Ludwig Mies van der Rohe, Daniel Libeskind, and Frank Gehry all have their own very narrow design style
- Each re-cycles the same image-based design elements in every building
- Their claim to “innovation” occurred in the very beginning — after that, they rely upon their stored vocabulary
Process of memory retrieval

• A trivial mathematical explanation
• Previous results of computations using an algorithm are stored in memory
• Table of products lists entries in a 2-D array
• Look up the result — no computation involved, only recall
• For example, $3 \times 5 = 15$
## Multiplication table

<table>
<thead>
<tr>
<th>x</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>4</td>
<td>6</td>
<td>8</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>3</td>
<td>6</td>
<td>9</td>
<td>12</td>
<td>15</td>
</tr>
<tr>
<td>4</td>
<td>4</td>
<td>8</td>
<td>12</td>
<td>16</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>10</td>
<td>15</td>
<td>20</td>
<td>25</td>
</tr>
</tbody>
</table>
Memory is liable to corruption

- Using two coordinates, locate product in memory array: \((3, 5) = 15\)
- But informational virus can invade the memory bank
- Virus replaces data with copies of itself
- Virus uses memory to propagate — computations that rely only on memory make copies of the virus
Multiplication table corrupted by the data virus “5”

<table>
<thead>
<tr>
<th>x</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
<td>5</td>
</tr>
</tbody>
</table>
False results

• Computation accesses the correct data position (cell) in memory
• No algorithm is involved
• But memory is corrupted by virus
• *Output is wrong*: $3 \times 5 = 5$
Check result by using algorithm

• Problem: \( S = 3 \times 5 \)
• Write product as a sum \( S = 5 + 5 + 5 \)
• Add the first two numbers \( S = 10 + 5 \)
• Repeat process until you have only a single digit
• \( S = 15 \)
• This is the answer
Set up the two memory banks needed for adaptive design

- Memory bank 1 contains typology
- Memory bank 2 contains socio-geometrical patterns in the sense of Christopher Alexander
- Set up architectural memory bank 1 — a three-dimensional reference system containing evolved solutions
• **First axis is urban density**: use the Transect system of Andrés Duany and Elizabeth Plater-Zyberk, labeled T1 to T6

• **Second axis is world location (which includes a form language)**: Algeria, South-Eastern USA, Nigeria, Scotland, etc.

• **Third axis is functional typology**: school, bank, church, apartment building, house, store, factory, etc.
Architectural memory bank 1
Urban density: Transect system

- Classification of different urban density
- T1 = natural (nature preserve)
- T2 = rural (farmhouses)
- T3 = sub-urban (mixed-use, not sprawl!)
- T4 = general urban (small town)
- T5 = urban center (dense urban fabric)
- T6 = urban core (the densest part of cities)
World location: local culture, climate & materials

• Form language — language of building developed alongside spoken language
• Evolved via trial-and-error over many years and fine-tuned by generations of users and builders
• Form expression of typologies shaped by culture, climate, local materials
Architectural memory bank 2

• The Alexandriner pattern data base — separate memory bank solely for patterns
• Universal architectural pattern typologies
• Largely independent of culture, climate, local materials
• Correct solutions depend upon geometry, evolved along with human physiology
Adaptive design that draws from evolved memory banks

- Define your project in terms of a specific coordinate in 3-D memory
  - \((x, y, z) = (\text{Transect of urban density, world location, functional typology})\)
- Look up the unique address in Memory 1
- Recall the relevant patterns for use from Memory 2, then design adaptively
Healthy evolution of architectural memory bank 1

- Solutions can change over time
- One typology can merge into another
- Urban density usually changes in time
- Underlying culture in the same place changes, influenced by others
- Solutions adapt to changing conditions
Corruption of memory banks!

- Viruses invade architectural memory 1 and substitute for the architectural data cells
- Memory bank 1 entries now contain only glass or concrete cubes
- All buildings, in any urban density, all over the world, simply replicate the virus
- *Pattern memory bank 2 is wiped out — erased because it is a viral antibody!*
Architectural cube virus
Non-adaptive result from corrupted memory bank 1

• Go to position \((x, y, z) = (T2\text{-rural, eastern Pakistan, schoolhouse})\)
• Memory cell is glass and concrete box
• Go to position \((x, y, z) = (T5\text{-urban center, coastal Japan, apartment})\)
• Memory cell is glass and concrete box
• But result is not adaptive to either task!
The need for adaptive algorithms

- Architects should apply algorithms that adapt structure to human needs
- Simple algorithms connect pattern languages to form languages
- Process successfully generates adaptive design, and corrects irrelevant forms that have corrupted memory
Use an algorithm, or memory?

- Use a proven memory bank that archives evolved solutions
- Often just as good as computing a new solution
- When architectural memory banks are corrupted, however, we need to re-compute the solutions all over again
- Pattern languages prevent corruption
D. Evolutionary regression

• It is very easy to evolve a system backwards, thus reversing its development

• Simply reverse the selection criteria, and a Darwinian process takes care of the rest — it is still adaptation!
The blind Mexican cavefish *Astyanax*

- Fish originally lived outside caves, and evolved a well-developed eye
- Its outside relatives have good eyes
- *Astyanax* in caves has lost its eyes
- Backwards evolutionary adaptation to light-less cave environment
Jan Michl (Oslo, Norway)

- Most design is in fact redesign
- A sequence of corrections, additions, modifications, improvements, refinements
- Adaptive design of artifacts is quintessentially Darwinian
- Selection is a “process of tinkering”
- Design as innovation is only a myth
Evolution of tools and artifacts

- Designers never begin from a clean slate
- Function depends upon existing form
- The slogan “form follows function” really means “form follows purpose”
- The formalist purpose is simply to impose new criteria for selection
Function never precedes form!

- The *purpose* of Bauhaus designers was to promote a palette of industrial materials and a narrow stylistic vocabulary of sleek, shiny forms.
- Unconcerned with the actual function of artifacts, buildings, or cities.
- Modernist designers validated their *purpose* by referring to fictive demands of the “Spirit of the Age.”
More by Jan Michl

• “The modernist artistic visions were inflicted on the captive audience of the socially weak sections of the population” — no selection there!
• Governments and the media embraced the allure of the avant-garde
• We adopted the non-adapted modernist style because it was IMPOSED on us!
Forward evolution of books

- Biological evolution created mechanism of the eye-brain system
- Typography evolved over several centuries to optimize information transfer
  - serif fonts (Times); black ink; matte soft-white paper; 12 point font size, etc.
- Minimizes the brain-information interface
- Maximizes information transfer rate
Backwards evolution of architecture books

• Use sans-serif fonts (Helvetica)

• Use light gray instead of black ink

• Use too small a font size (9-10 point)

• Use highly glossy paper — too reflective to focus on

• Use unusual page formatting to show off fashionable typographic “style”
Ease of reading is degraded

• No paragraph breaks — confusion of textual and logical subdivision!
• No paragraph indents — ugly!
• Photos are intentionally blurred!
• Yet, these negative characteristics have proliferated — they define a highly successful antipattern
Selection over a few decades

- The Bauhaus style introduced sans-serif fonts with the “machine aesthetic”
- Selection criteria imposed by architects, authors, publishers, and a public fascinated with the “new look”
- Style takes precedence over legibility
- Obvious in recent architecture books
One example among many

- One recent book on architectural theory by a world-famous architect, co-authored with a world-famous philosopher, has holes punched in it!
- Book is intentionally illegible
- Highly praised, and recommended as course textbook in our elite universities
Architectural images have also evolved backwards

- Architectural renderings as vague translucent screen shots — no detail
- One cannot grasp the overall forms
- “Competition project style” combines reflective with transparent surfaces
- But their optical properties prevent the eye from focusing — cannot see them!
Another example among many

- Lectures by a world-famous architect
- Illustrated with blurry monochrome brown photos of the famous architect, intentionally made grainy
- Book would probably not sell if the photos were clear and focused!
- Only the fashionable “style” sells
Information obtained only through pain

- Contemporary architecture books intentionally or unintentionally strain the eyes of the reader
- Yet students have to study them for their courses
- Is this regression an oversight, or is it causing pain to reinforce power?
Conclusions

• Algorithmic design that is adaptive relies upon emotion — uses the human computer
• Architecture schools teach rationalization for each design decision — but that’s not computational!
• It is really the rationalization of a *style*
• True computation results in unusual and unexpected (not “absurd”) configurations