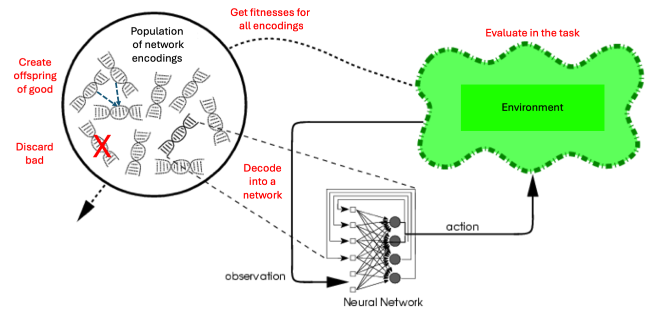
Indirect Encodings: Developmental Approaches

Risto Miikkulainen

October 21, 2024

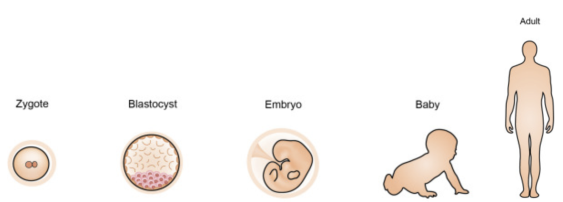
Direct vs Indirect Encodings in Neuroevolution

- Direct encodings map genotype directly to phenotype (one-to-one).
- Indirect encodings compress genotype.
- A simple genotype can then be mapped into a complex phenotype.



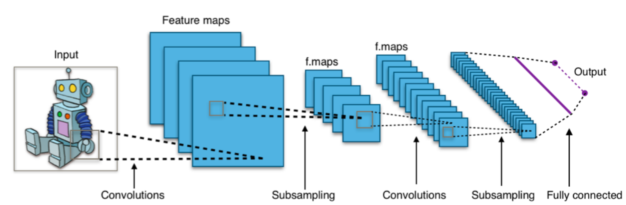
Why Indirect Encodings?

- Biological organisms develop from a single starting point.
- Morphogenesis leads to the creation of complex structures.
- Reuse of structural motifs creates regularity.
- Structures specialize based on context.



The Role of Regularities in Encoding

- Regularity enables compression and compactness of neural networks.
- E.g. convolutional neural networks reuse feature detectors across multiple locations.
- Neuroevolution could discover similar patterns without human intervention.



Is NEAT Indirect Encoding?

- NEAT is a direct encoding algorithm, mapping each network parameter directly to a node or connection.
- Works well for small networks, but is difficult to discover e.g. convolutional networks.

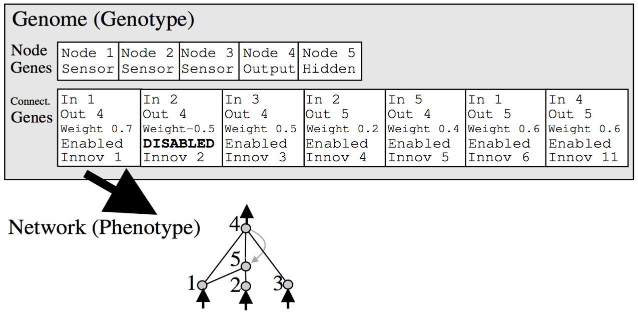
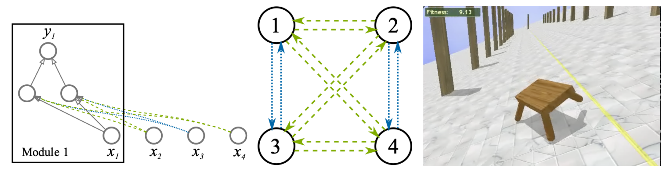


Figure: Example of the NEAT algorithm's direct encoding.

Problem Example: Evolving a Quadruped Robot Controller

- Control for one leg helps control the others (symmetry and pattern reuse).
- Encouraging modularity through manual decomposition can assist neuroevolution.
- Ideally, indirect encodings would capture these patterns automatically.



Developmental Processes in Neuroevolution

- The first major approach to indirect encoding: Imitate biology.
  - There are about 24K genes and 1000 trillion synapses.
  - Most of those are specified through development.
- Development can be mimicked in neuroevolution by
  - Cell chemistry (low-level mechanisms).
  - Grammatical encodings (high-level abstraction).
  - Learning mechanism (engaging the environment).

Cell-Chemistry Approaches

- Chemical substances (morphogens) diffuse and interact to form patterns.
- E.g. Alan Turing's reaction-diffusion model:
- At each location, concentration C depends on reactions F and diffusion D:

$$\partial C/\partial t = F(C) + \mathbf{D}\nabla^2 C$$

- Results in complex patterns such as seashells, fur, feathers, scales:

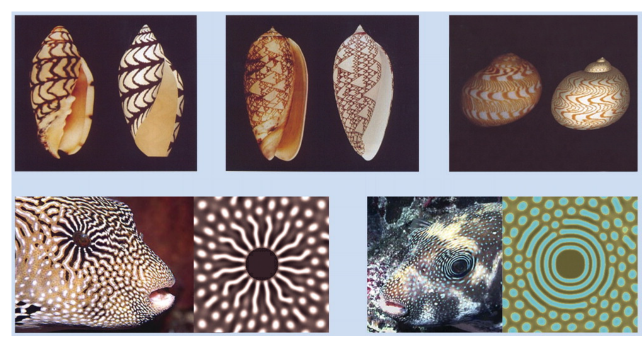
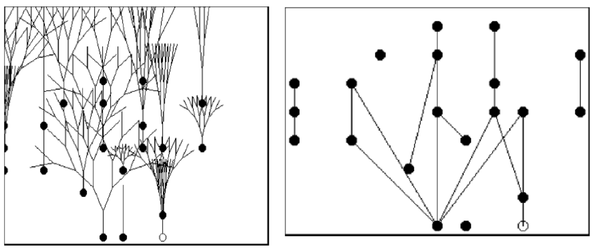


Figure: Natural vs. reaction-diffusion patterns.

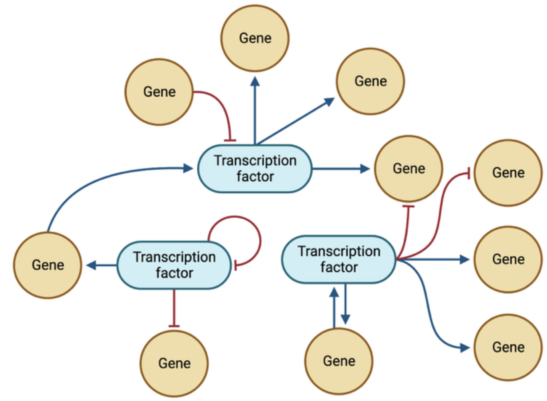
Applying Reaction-Diffusion to Neural Networks

- Diffusion models axonal growth, and chemical reactions model interactions between axons and cells.
- Neuron definitions include location of cell bodies and axon branching rules.
- Growth is pruned to remove non-useful connections, allowing for indirect encoding.



Genetic Regulatory Networks (GRNs)

- Gene expression creates proteins.
- Gene expression is followed by complex networks of gene interactions .
  - Promoting and inhibiting expression
  - Mediated by transcription factors (proteins).
- Much of the complexity in the phenotype is due to GRNs.

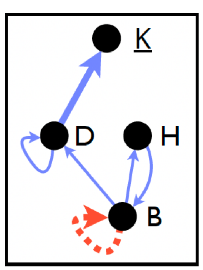


Applying GRNs to Neural Networks

- Indirect encoding through GRNs can evolve complex neural structures.
- Proteins represent neurons; connections form when the proteins match.
- For example,

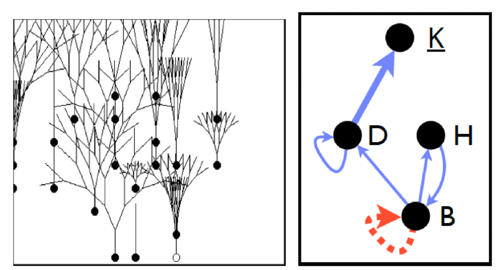
$$\begin{aligned} G \;\&\; \neg B &\implies B \\ C &\implies D, H \\ D &\implies D, K \end{aligned} \tag{1}$$

  - If protein G is present and B is absent, then B is produced.
  - B is similar to C, so D and H are produced.
  - H is similar to G, which enhances B until there's enough B.



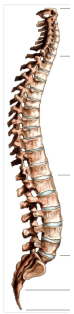
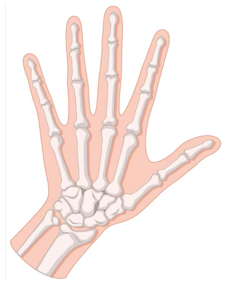
Challenges and Potential of Cell-Chemistry Approaches

- Cell-chemistry approaches are powerful but complex to implement.
- GRNs and reaction-diffusion models offer rich representations but are difficult to scale.
- Simplifications such as Boolean GRNs or string-based genomes help balance complexity and effectiveness.
- Complexity of GRNs may lead to open-ended evolution.



Grammatical Encodings

- Biological organisms exhibit repetition with variation.
- Examples: Vertebral columns, fingers, bilateral symmetry.
- Grammatical encodings aim to capture such structure.
- They are a high-level abstractions of development.


Lindenmayer Systems (L-systems)

- L-Systems (Lindenmayer systems) represent a formal grammar-based method.
- Each rewrite rule step represents a stage in development.

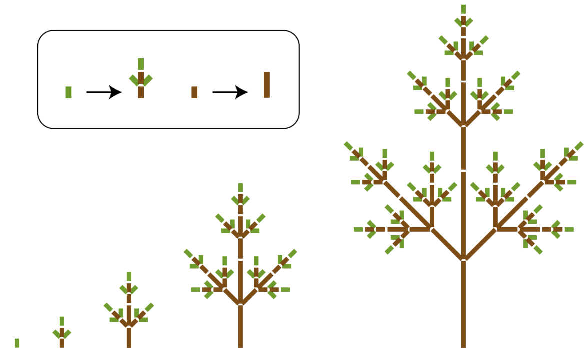
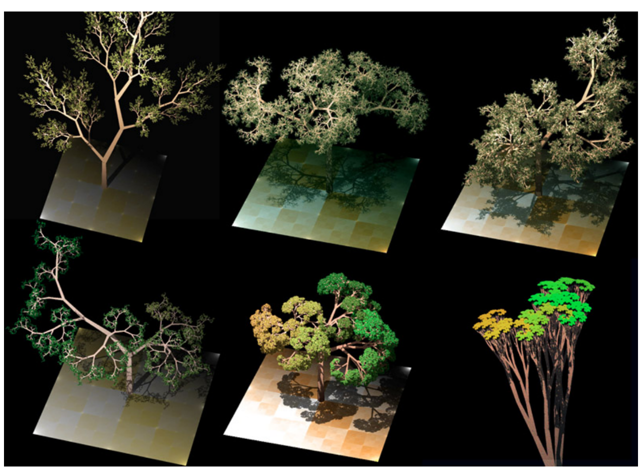


Figure: L-System rewriting steps to generate plant-like structures.

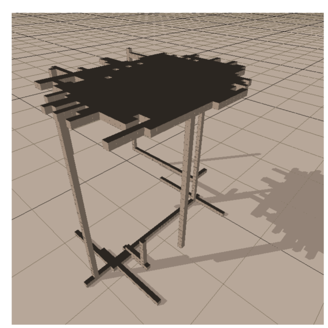
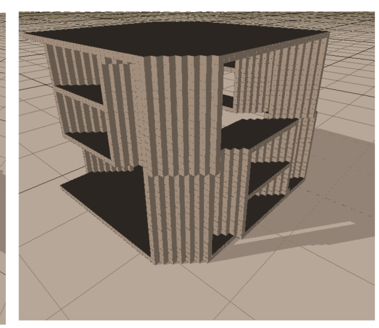
L-Systems in Practice

- L-Systems can produce realistic plant structures and virtual foliage.
- Used in visual effects for movies like Iron Man 3 and Avatar.
- Iterative rewriting generates increasingly complex structures.



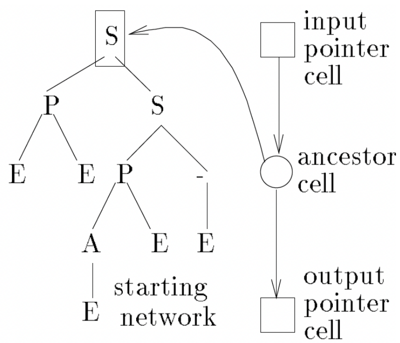
Evolving Designs with L-Systems

- L-Systems can be optimized using evolutionary search methods.
- E.g. evolving table designs
  - Direct encoding lacks reuse of components, producing more irregular structures.
  - Indirect encoding produces more symmetrical, modular, and regular structures.
  - More natural and aesthetically pleasing.

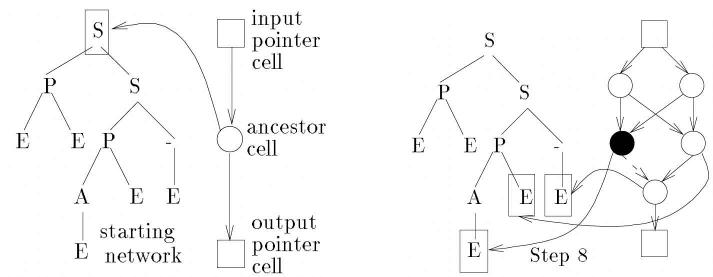
Grammatical Encoding of Neural Networks

- Cellular encoding (CE) uses grammars to describe neural network construction step by step.
- A grammar tree encodes instructions for network modifications.
- Each node of the grammar tree specifies a network transformation.



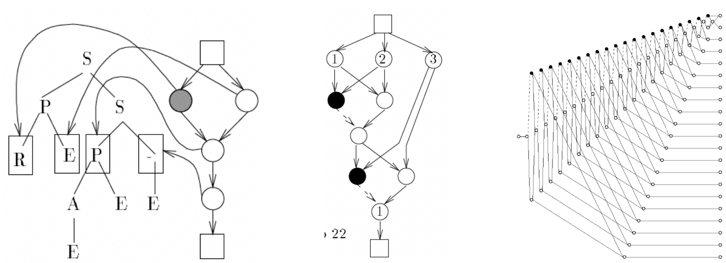
Cellular Encoding Example: XOR Network

- An example of CE: XOR neural network construction.
- Sequential and parallel divisions of the ancestor node.
- Final network implements XOR logic after several steps.



Recurrency in Cellular Encoding

- Recurrency allows repeating entire structures in a grammar.
- By traversing the grammar multiple times, larger networks can be evolved.
- E.g. parity and symmetric networks can be created with recurrency.



Learning as Indirect Encoding

- Biological development also involves learning from interaction with the environment.
- Learning mechanisms allow individuals to adapt their structure and behavior during their lifetime.
- Synergy between evolution and learning can be a powerful computational mechanism.

Lamarckian Evolution

- Lamarckian evolution suggests that acquired traits can be inherited.
- Discredited in biology: Darwinian selection achieves the same result.

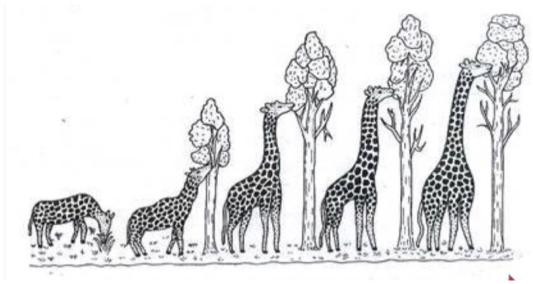
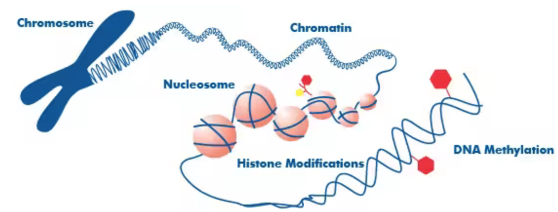


Figure: Inheritance of acquired characteristic: A stretched neck.

Epigenetic Evolution

- However, epigenetic mechanisms have recently been found with a similar effect.
- Parts of DNA covered in methylation during lifetime.
- The methylation can be inherited.
- Methylation affects RNA transcription, and eventually behavior.
- E.g. fearfulness that lasts for two or more generations.



Computational Lamarckian Evolution

- Computationally easy to take advantage of Lamarckian and epigenetic principles.
- Can be applied e.g. to evolving deep learning networks.
- Gradient-based learning is combined with evolutionary exploration of architectures.
- Success e,g, in evolving convolutional architectures for image processing.

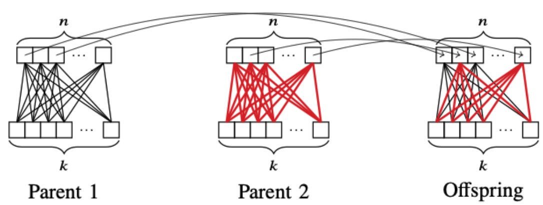
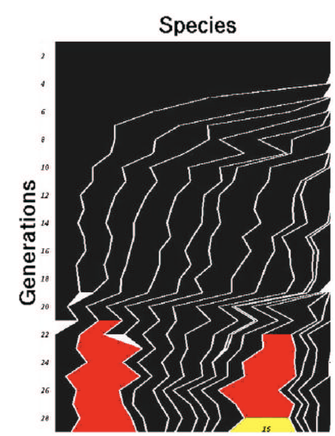


Figure: Crossover of convolutional networks.

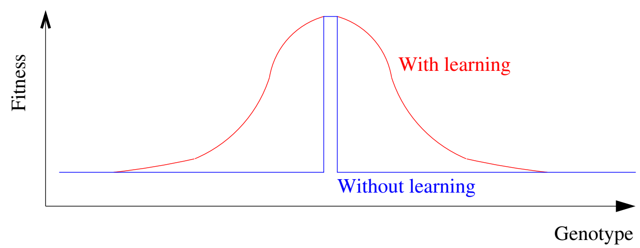
Challenges in Lamarckian Evolution

- Gradient-based learning can lead to a loss of diversity.
- Population diversity can be maintained using ensembling, data batching, speciation.
- Balancing exploration and learning is an ongoing research challenge.



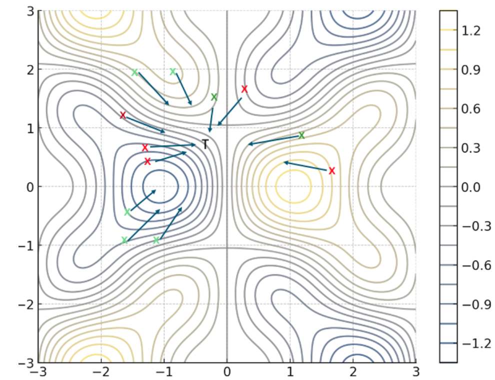
Darwinian Alternative: The Baldwin Effect

- The Baldwin effect suggests that learning guides evolution without encoding changes in the genome.
- Learning helps evolution discover promising individuals by broadening the search space.
- Even just random walk helps solve difficult problems, such as the needle-in-the-haystack.
  - Probability of success depends on initial distance to the goal.



Synergetic Development

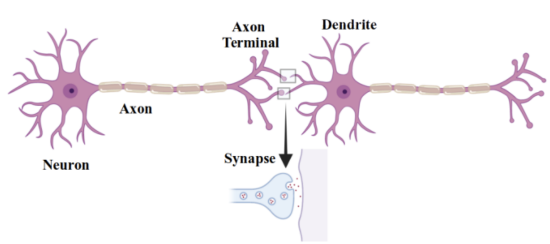
- Surprise: Evolution discovers good starting points for learning rather than near-optimal solutions.
  - Learning will happen, so evolution discovers how to take it into account.
  - A synergy between learning and evolution.
- Solutions are more robust and more effective.



Hebbian Learning as an Online Adaptation Mechanism

- Hebbian learning is a local learning mechanism where neurons that fire together wire together.
- It requires no learning targets and is more biologically plausible than gradient descent.
- Hebbian learning can drive adaptive behavior by strengthening useful connections.
- Not powerful enough to learn complex behavior, but could adapt to changing environments.

$$\Delta w_{ij} = \alpha_{ij} o_i o_j - \beta_{ij} w_{ij} \tag{2}$$



Hebbian Adaptation

- Hebbian learning can evolve networks capable of switching tasks based on experience.
- For example, turning on a light and then moving to a target area.
- It provides flexibility, but does it outperform recurrent activation for adaptation?

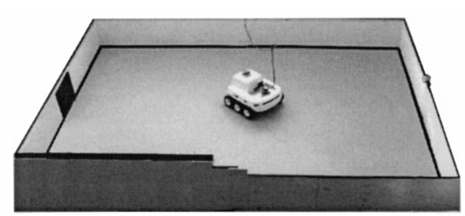


Figure: Adapting to the switch location.

Hebbian Learning vs Recurrent State

- In simple tasks, recurrent networks can adapt by representing a state (e.g. light on or off).
- Recurrent networks without Hebbian learning are simpler and often more efficient.
- However, weight adaptation persists over many trials, and may be an advantage in more complex tasks.

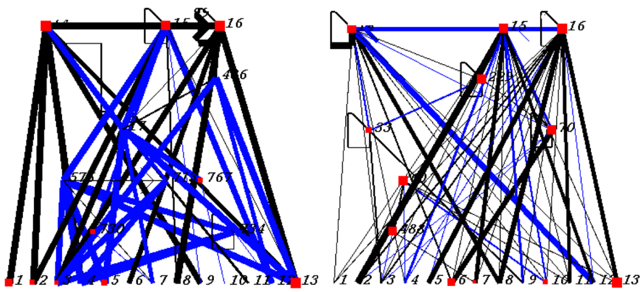


Figure: Network with Hebbian learning vs network with simple recurrence.

Development Conclusions

- Development can be utilized to construct complex neural networks.
  - Based on low-level chemistry, high-level grammars, and learning processes.
  - Indirect encoding of solutions.
- Understanding biology; building artificial systems.
- The interplay between evolution, development, learning, and adaptation remains an intriguing research direction.