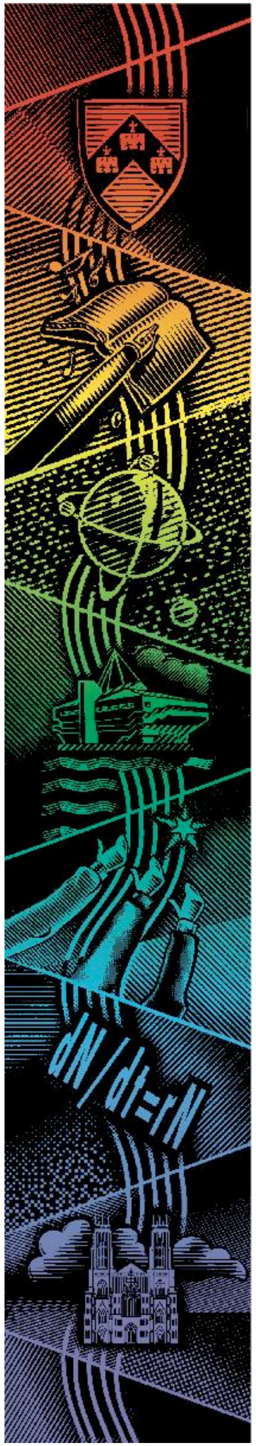


Evolvability - The Missing Link?

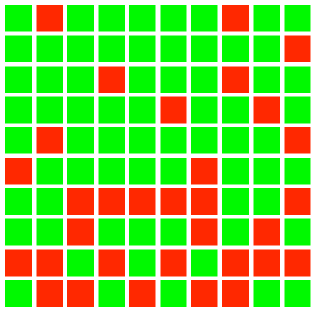
Michael Lones

Intelligent Systems Research Group

Department of Electronics

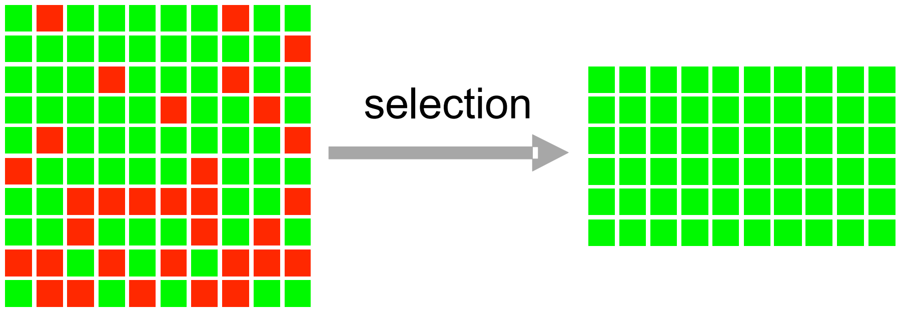
THE UNIVERSITY of York

Evolutionary Computation



– Population of random solutions

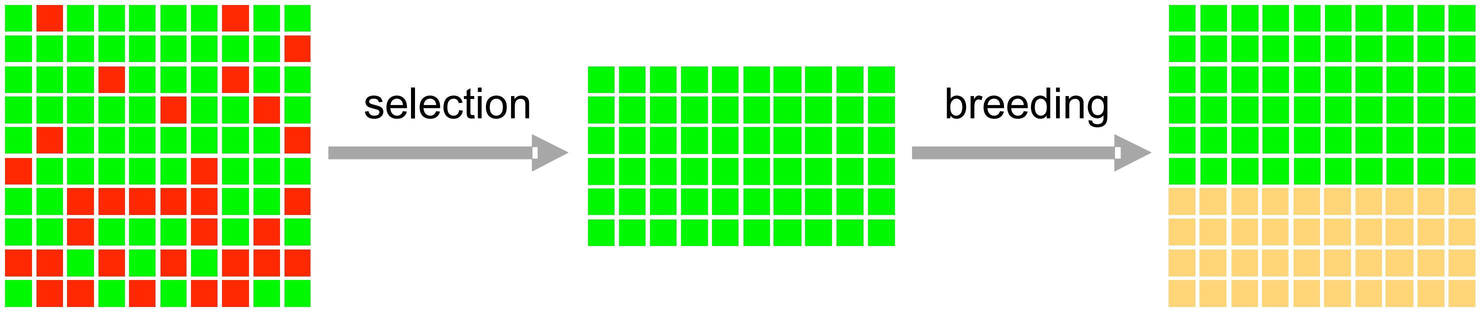
Evolutionary Computation



– Population of random solutions

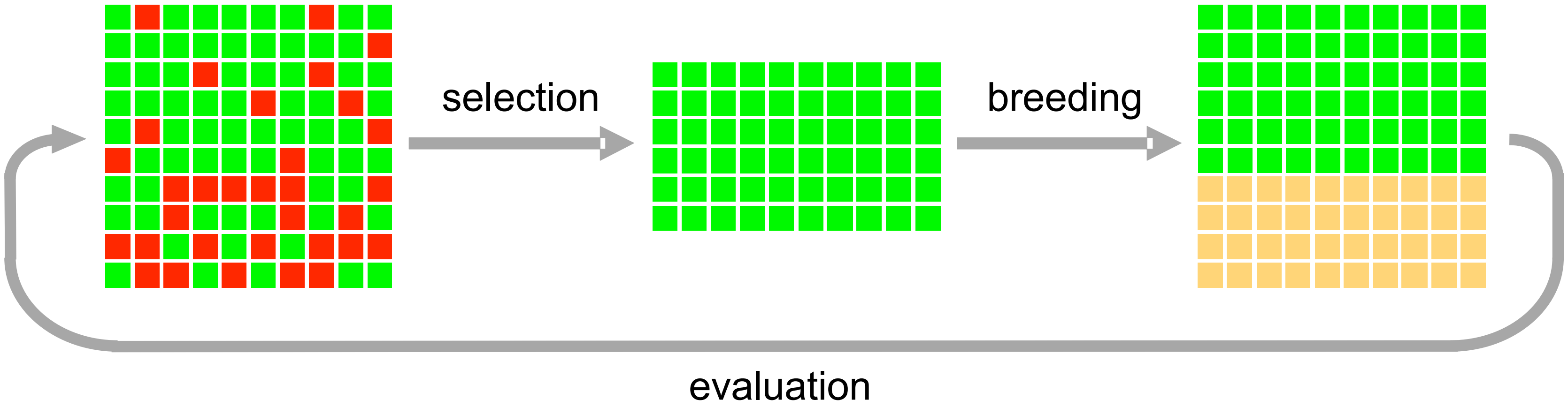
– Selection removes relatively poor solutions

Evolutionary Computation



- Population of random solutions
- Selection removes relatively poor solutions
- Reproduction generates the next generation

Evolutionary Computation



– Population of random solutions

– Selection removes relatively poor solutions

– Reproduction generates the next generation

– Repeat until solved

Evolutionary Computation

Has been used to evolve many things…

- Neural networks
- Robotic controllers
- Bio-sequence motifs
- Electrical circuits
- Quantum circuits
- Aircraft wings
- Beer brewing process

Evolutionary Computation

Is a **heuristic** search algorithm

- assumes that better solutions can be found by making random changes to or recombining existing solutions

Michael Conrad

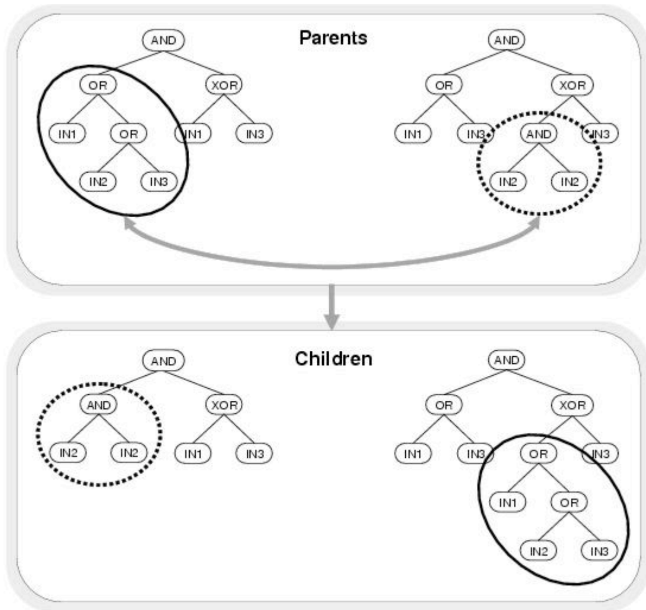
- Programs can't be evolved because mutation would lead to malfunction (…or something to this effect)

Genetic programming…!?

Genetic Programming

has a few problems…

- Scalability
- Program size bloat
- Sub-tree crossover

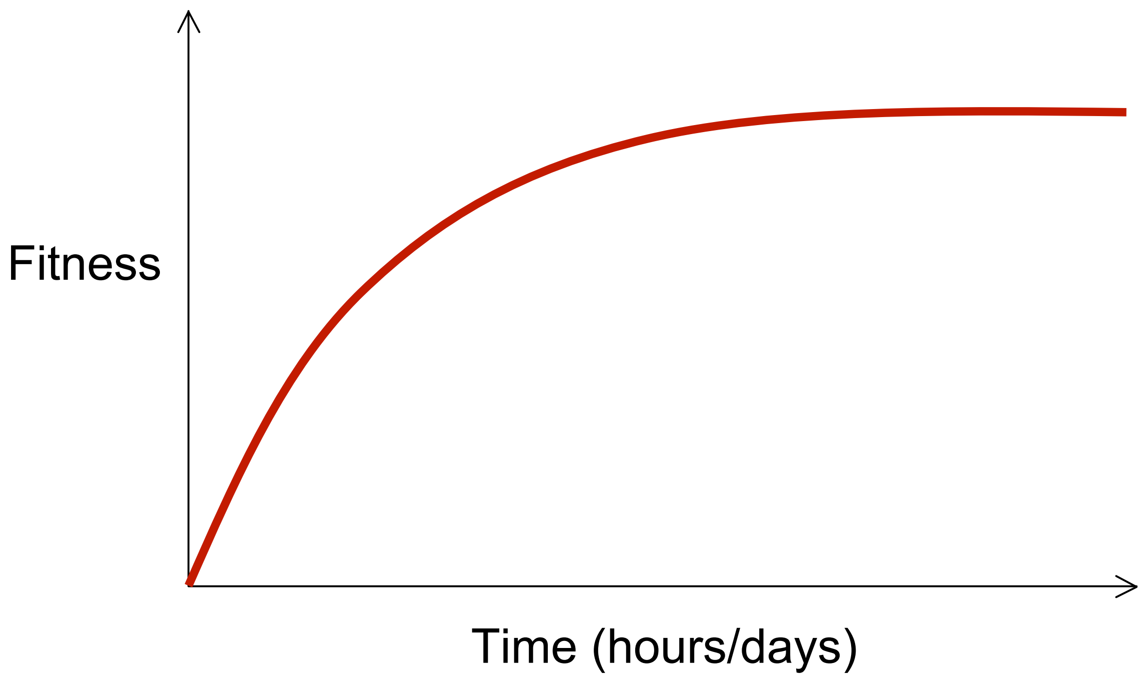


…an operator problem?

…or a representation problem?

Evolvability

A typical evolutionary computation run:



Evolvability

Altenberg, 1994

- "the ability of a population to produce variants fitter than any yet existing"

Kirschner and Gerhart, 1998

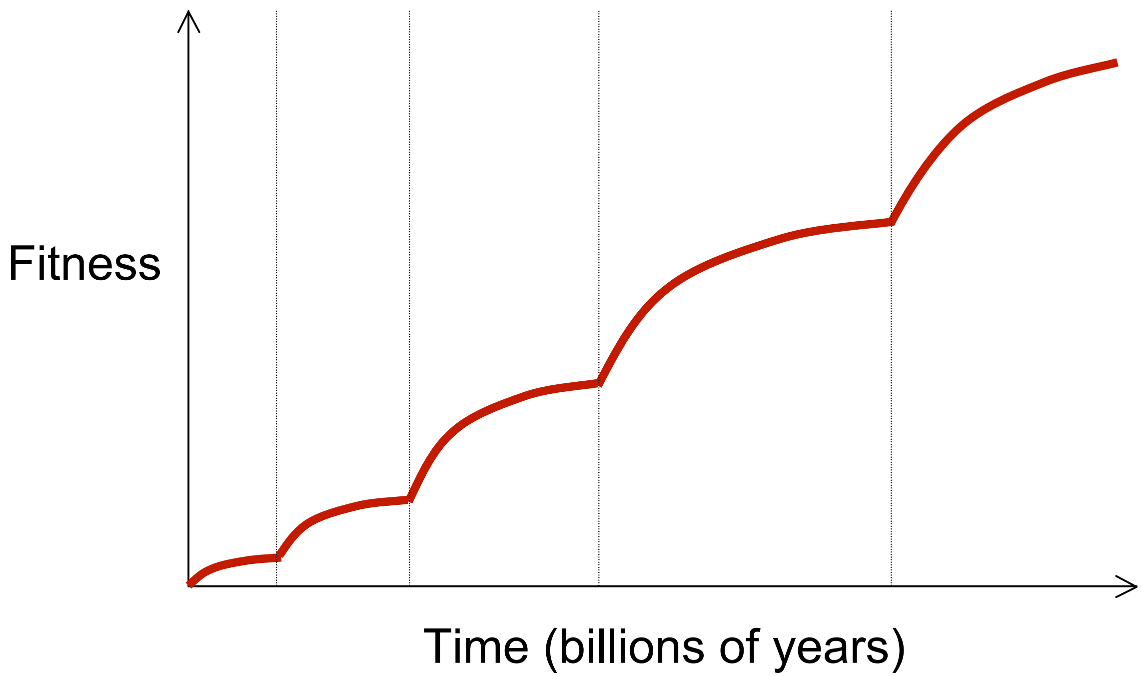
- "the capacity to generate heritable selectable phenotypic variation"

An evolvable system…

- is organised in such a way that change is more likely to lead to adaptation than if it were organised otherwise

Evolution of Evolvability

In biology?



Evolvability

Michael Conrad, 1990

– the sources of biological evolvability:

Compartmentalization

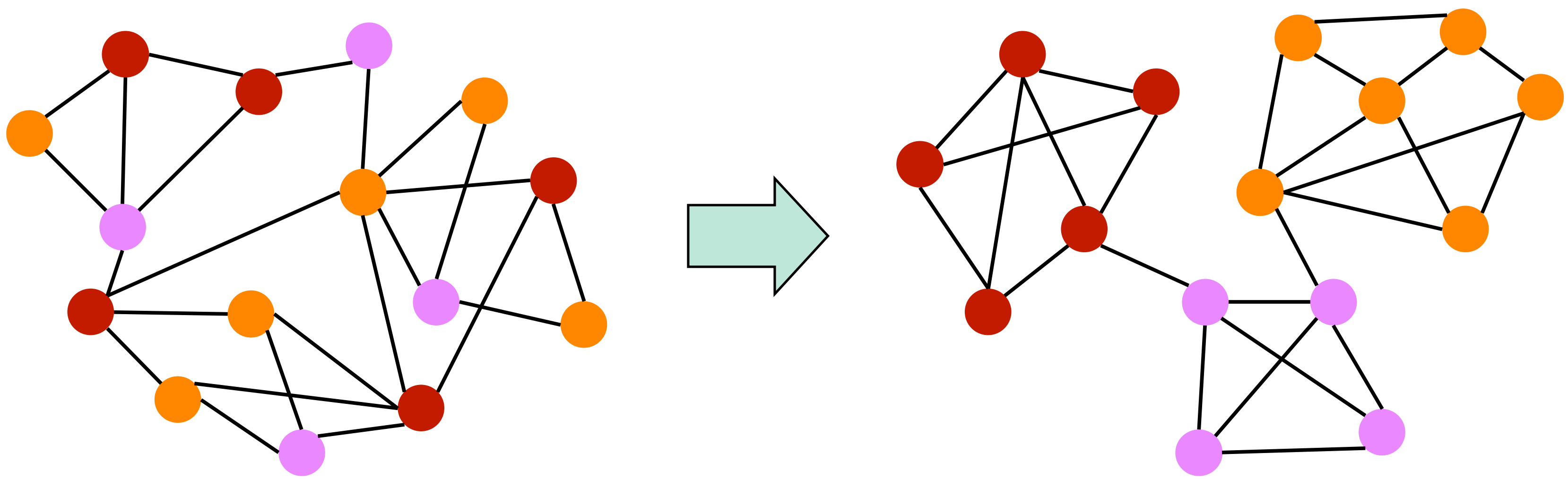
Redundancy

Multiple weak interactions

Compartmentalisation

Split a system into compartments

- Many interactions within a compartment
- Few interactions between compartments



Compartmentalisation

Examples:

- Genetically independent pathways
  - Metabolic pathways can evolve separately
- Embryonic fields
  - Developmental pathways can evolve separately
- Epistatic clustering
  - Improved horizontal gene transfer and crossover

Pleiotropy

Genes are sometimes expressed in more than one biological context

– Can lead to co-adaptation

– Increased variational potential out-weighing interference between sub-systems?

– A trade-off!

Redundancy

Functional redundancy

– Redundant copies of functional components

Structural redundancy

– Redundant structure within components

Weak linkage

– Redundant connections between components

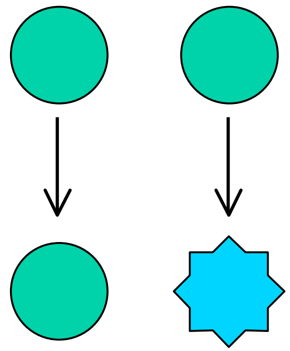
Functional Redundancy

A simple recipe:

– Make two copies of a component

– Evolve one of them

– Keep the other as a backup



Evolution by gene duplication, Ohno 1970

– Gene duplication and divergence is a major component of molecular evolution

Functional Redundancy

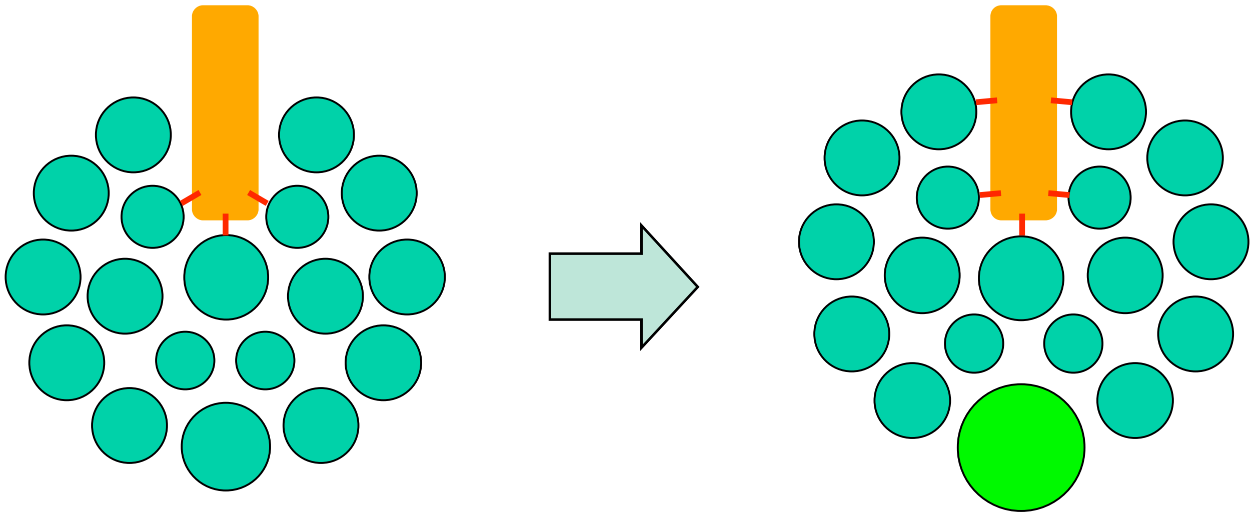
Examples:

- Gene families and pseudo-genes
  - Redundant copies of genes
- Polyploidy
  - Redundant copies of chromosomes
- Allozymes
  - Functionally equivalent enzymes

Structural Redundancy

Functionally-unnecessary structure

– e.g. Redundant amino acids in proteins



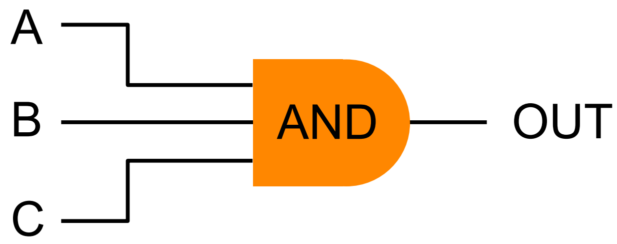
Structural Redundancy

E.g. Non-coding DNA

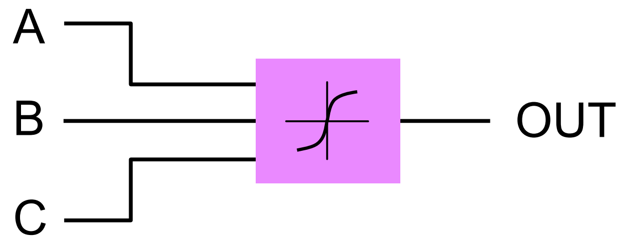
- Segregates genes during crossover
- Allows exon shuffling, Gilbert 1978
- Space for functional redundancy
- Supports mobile elements

Weak Linkage

Strong linkage:



Weak linkage:



– Strongly linked systems are fragile to change

– Weak linkage allows gradual change

Weak Linkage

Examples:

– Transcription regulation

– Neural networks

– Signalling pathways

– Protein folding

– Binding/active sites

Protein Evolution

Three axis of evolution

- Duplication and divergence
- Shape change via substitutions
- Changes in folding

Mutation buffering

- Proteins can undergo many mutations with little or no functional change
- Introduces new evolutionary paths

Neutral Evolution

Exploration of neutral variants

– Populations drift within neutral networks until they find an access point to a higher fitness network

Examples:

– RNA Folding

– Protein folding

Biological Evolvability

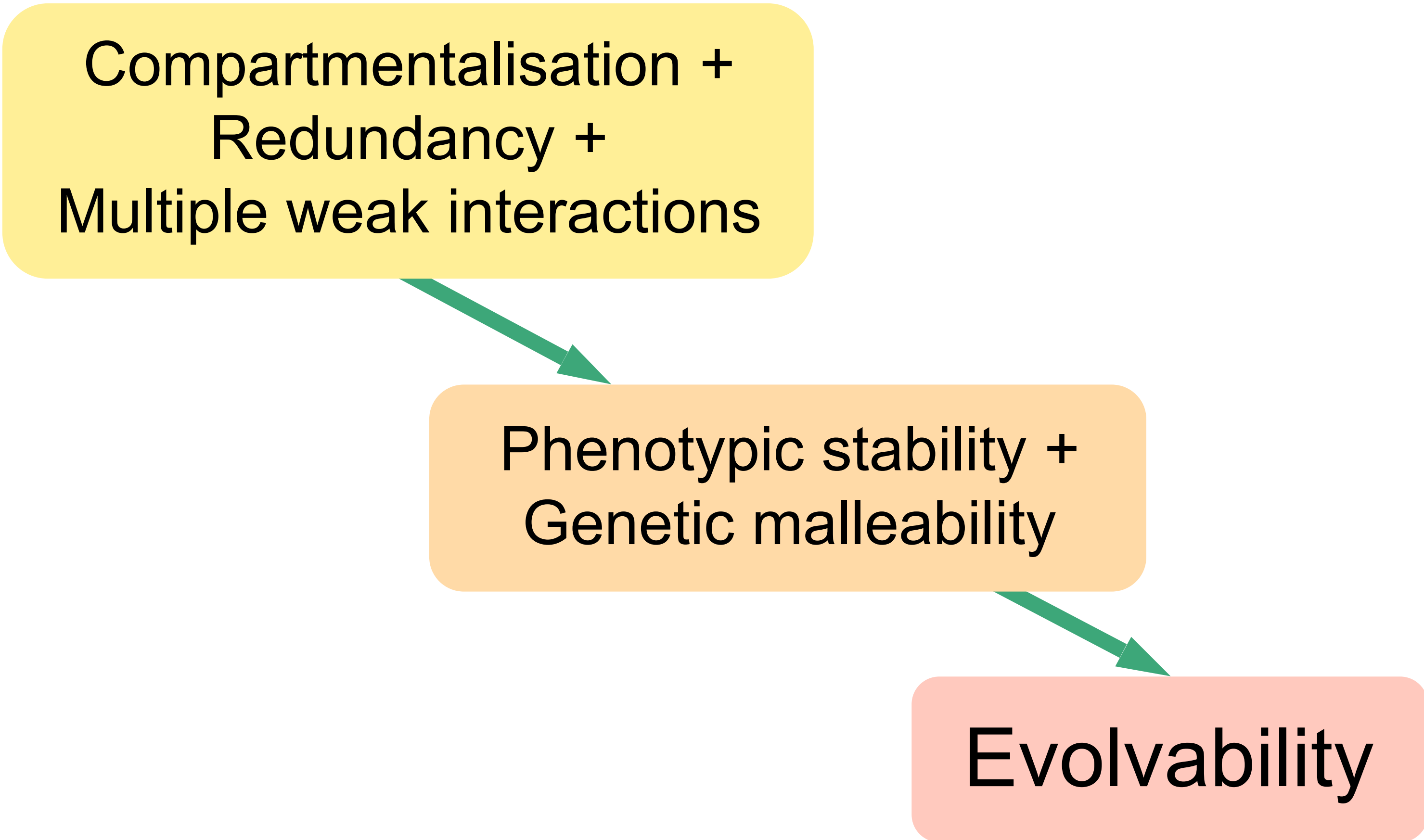
De-constraint

– Impact of change is contained
– Allows parts of a system to evolve separately
– Components can evolve gradually

Adaptability

– Exploration without commitment
– High-level changes are possible

Biological Evolvability



Evolutionary Computation

But what about me..?

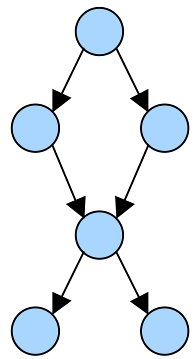
Pleiotropy

Modular decomposition in GP

- Identify modules and allow reuse
- MA [Angeline '94], ARL [Rosca & Ballard '96]

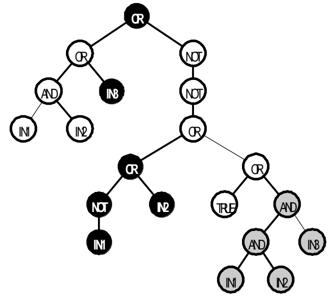
Implicit reuse

- Graph-based representations in GP
- Trade-off can be evolved



Redundancy

Structural redundancy

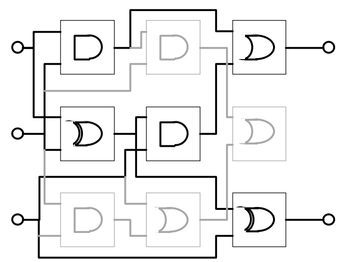


– ‘Introns’ in GA and GP solutions

Coding redundancy

– Many-to-one mappings

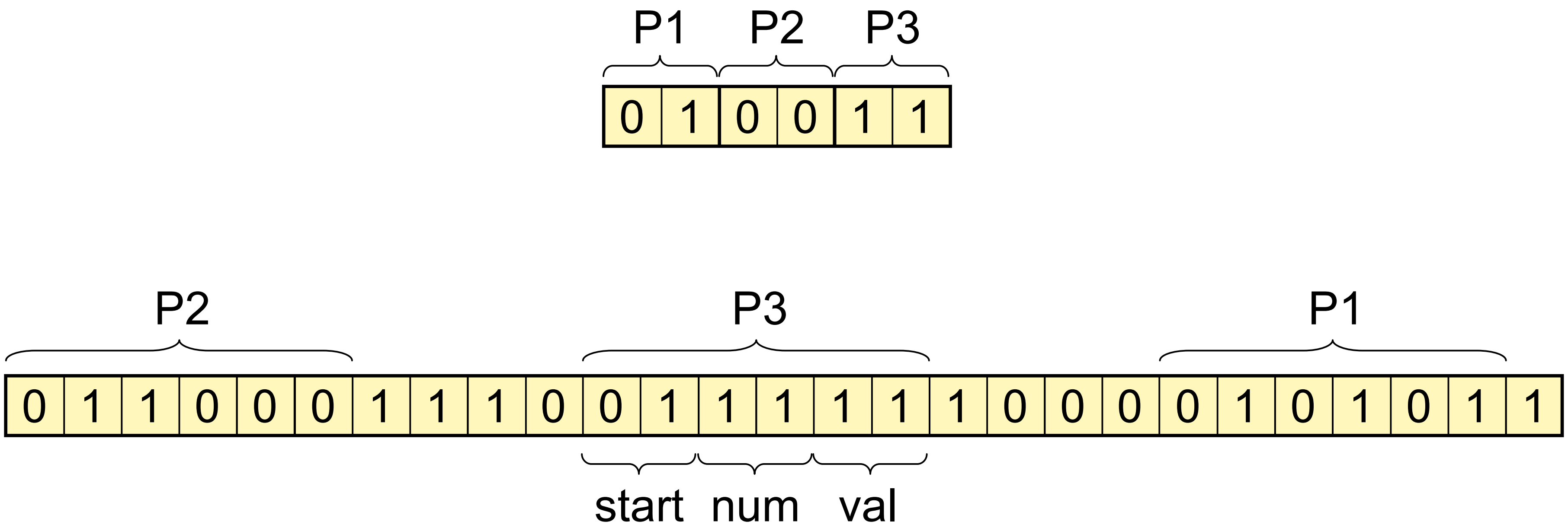
Functional redundancy



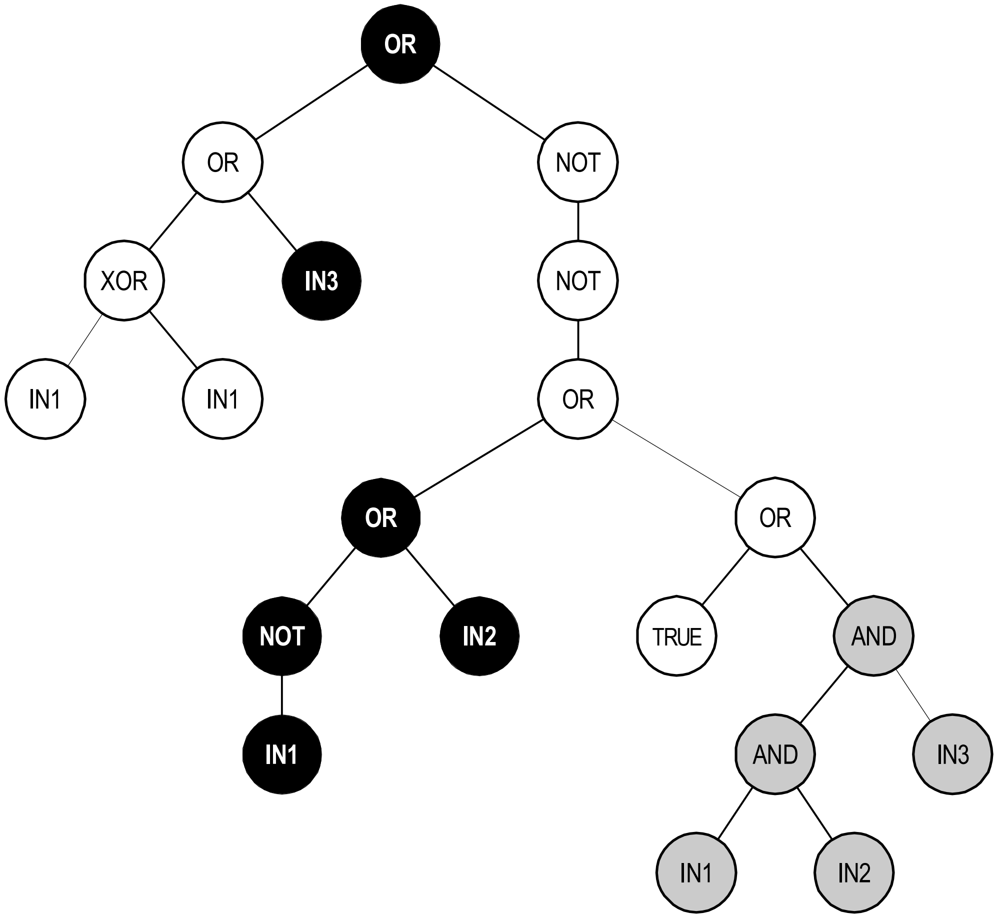
– Redundant solution components

GA Introns

E.g. Wu and Lindsay, 1996:



GP Introns

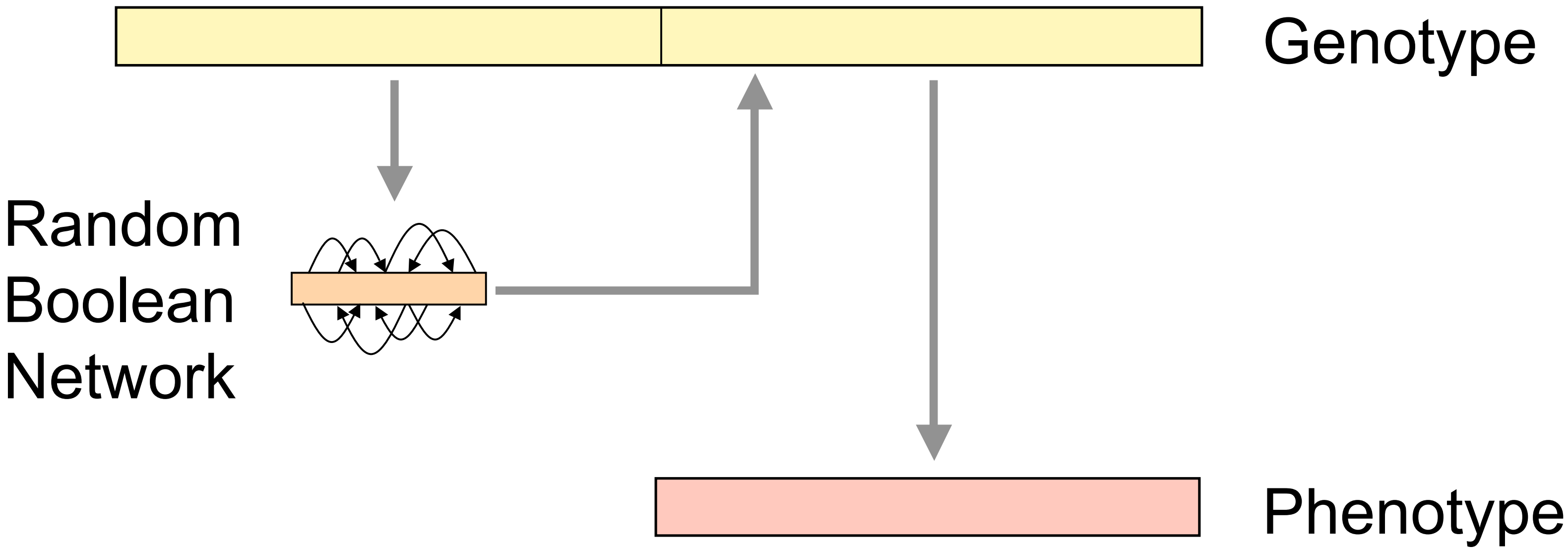


- Effective code
- Semantic intron
- Syntactic intron

Coding Redundancy

Ebner et al., 2001

– Highly neutral mappings are better



THE UNIVERSITY of York

Coding Redundancy

Knowles and Watson, 2002

- RBN mappings impair performance
- Time is wasted exploring neutral networks

Rothlauf and Goldberg, 2002

- Complex mappings have poor ‘locality’
- Offspring are unlikely to resemble parents

Coding Redundancy

Barreau, 2002

Symbol	Non-redundant code	Redundant code	
A	000	0 000	1 110
B	001	0 001	1 011
C	010	0 010	1 010
D	011	0 011	1 000
E	100	0 100	1 100
F	101	0 101	1 111
G	110	0 110	1 001
H	111	0 111	1 101

Coding Redundancy

Barreau, 2002

Symbol	Non-redundant code	Redundant code	
A	000	0 000	1 110
B	001	0 001	1 011
C	010	0 010	1 010
D	011	0 011	1 000
E	100	0 100	1 100
F	101	0 101	1 111
G	110	0 110	1 001
H	111	0 111	1 101

Coding Redundancy

Barreau, 2002

Symbol	Non-redundant code	Redundant code
A	000	0 000 1 110
B	001	0 001 1 011
C	010	0 010 1 010
D	011	0 011 1 100
E	100	0 100 1 100
F	101	0 101 1 111
G	110	0 110 1 001
H	111	0 111 1 101

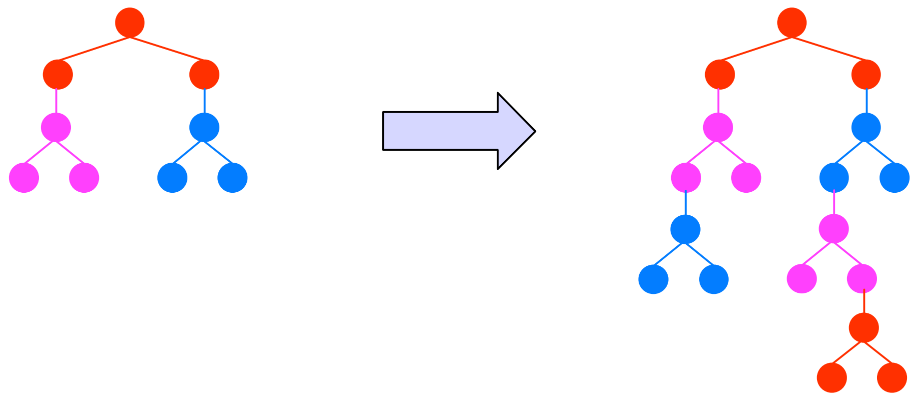
– High correlation between removal of local optima and performance gain

– Too much neutrality reduces performance

Functional Redundancy

Haynes, 1996

- Duplicated coding sub-trees
- More duplication, faster evolution
- Replace syntatic introns with coding regions?



Polyploidy

Multi-chromosome GA, e.g. Goldberg 1989

- Effective for dynamic fitness functions
- Provides temporal memory - better than mutation
- Benefits for up to 9 chromosomes [Collingwood 1996]

Structured GA, Dasgupta 1992

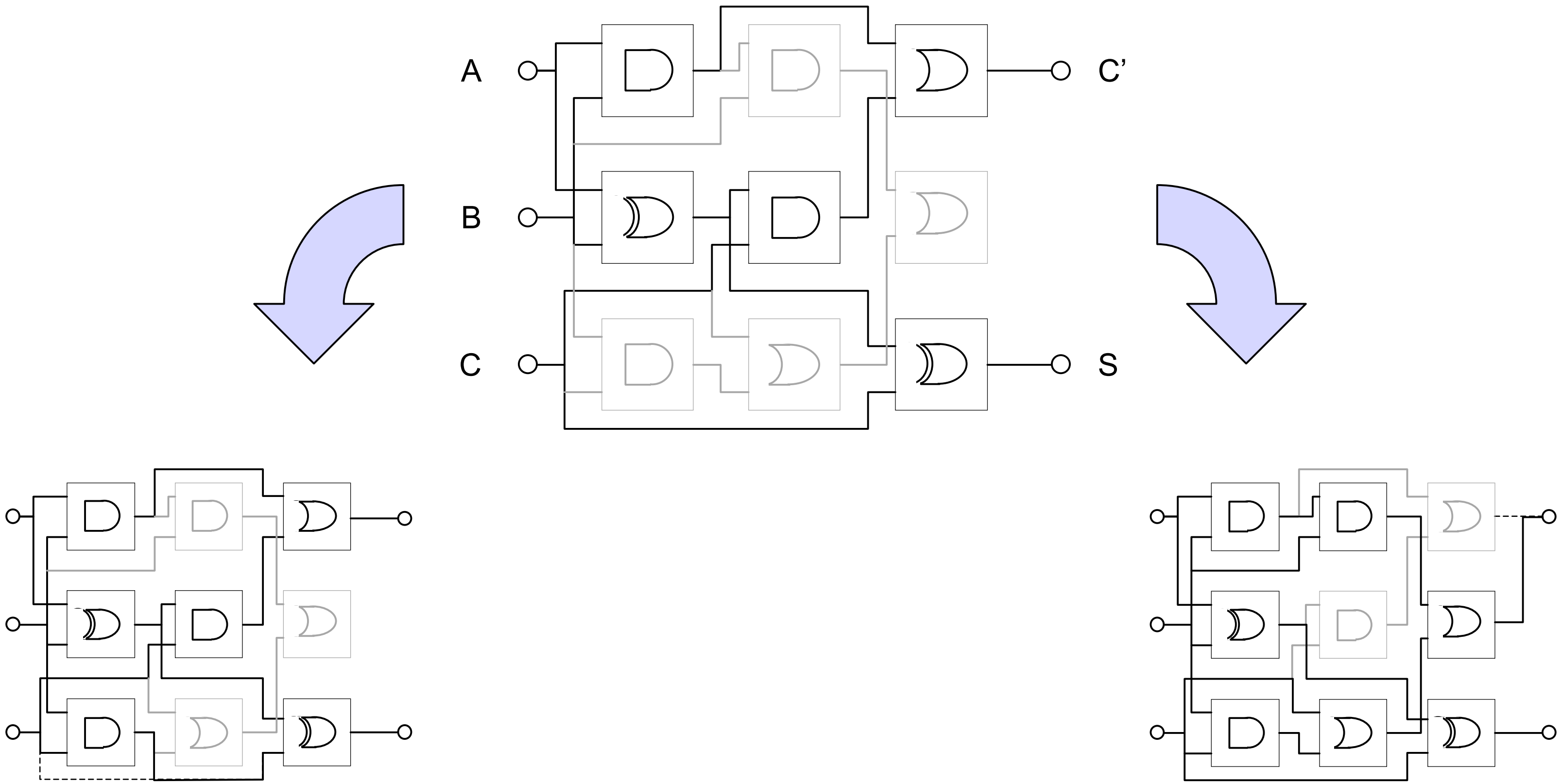
- Complex evolved regulation hierarchies
- Effective on dynamic and stationary functions

Multi-chromosome GP, Cavill 2005

- Redundancy improves performance

Implicit Redundancy

E.g. Cartesian GP, Miller 2000



Implicit Redundancy

Vassilev and Miller, 2000

- Neutral mutations improve evolution

Lones and Tyrrell, 2003

- Non-coding components improve evolution

Oltean and Dumitrescu, 2002

- Expressed the non-expressed bits
- Performed well on standard GP problems

Weak Linkage

Volkert and Conrad, 1998

- Evolved non-uniform CAs
- With and without weak linkage

Weak linkage beneficial to…

- Exploratory scope
- Performance of solutions
- Tolerance to mutation

Evolvability

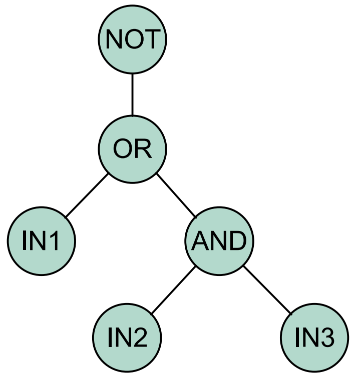
Concepts are not easily applied

- Existing representations are inflexible
- Strong linkage, no redundancy…

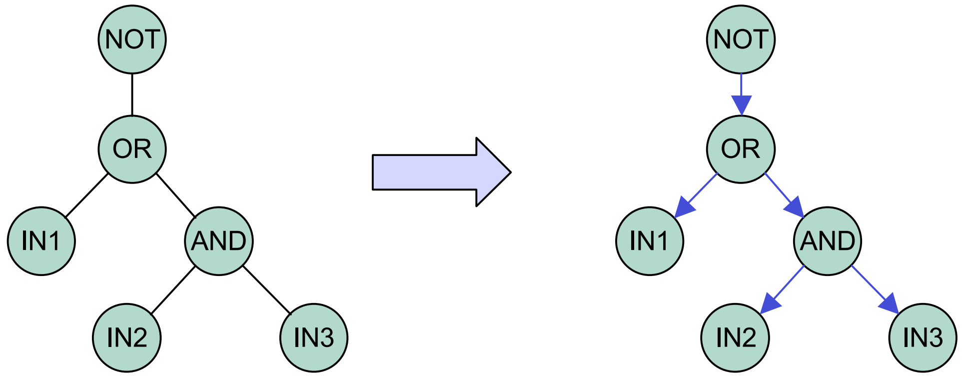
Possible solutions:

- Genotype to phenotype mappings
- Adopt novel representations

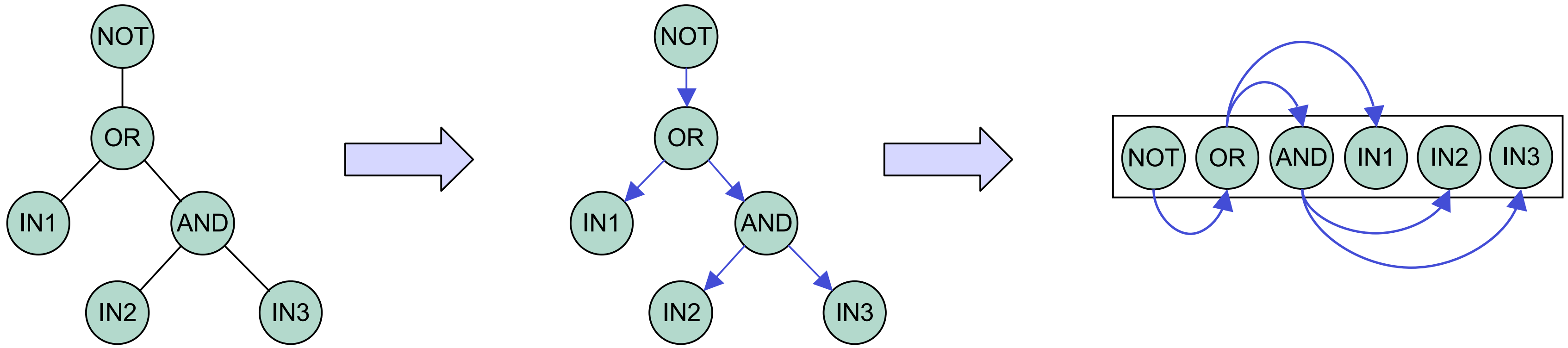
Implicit Context



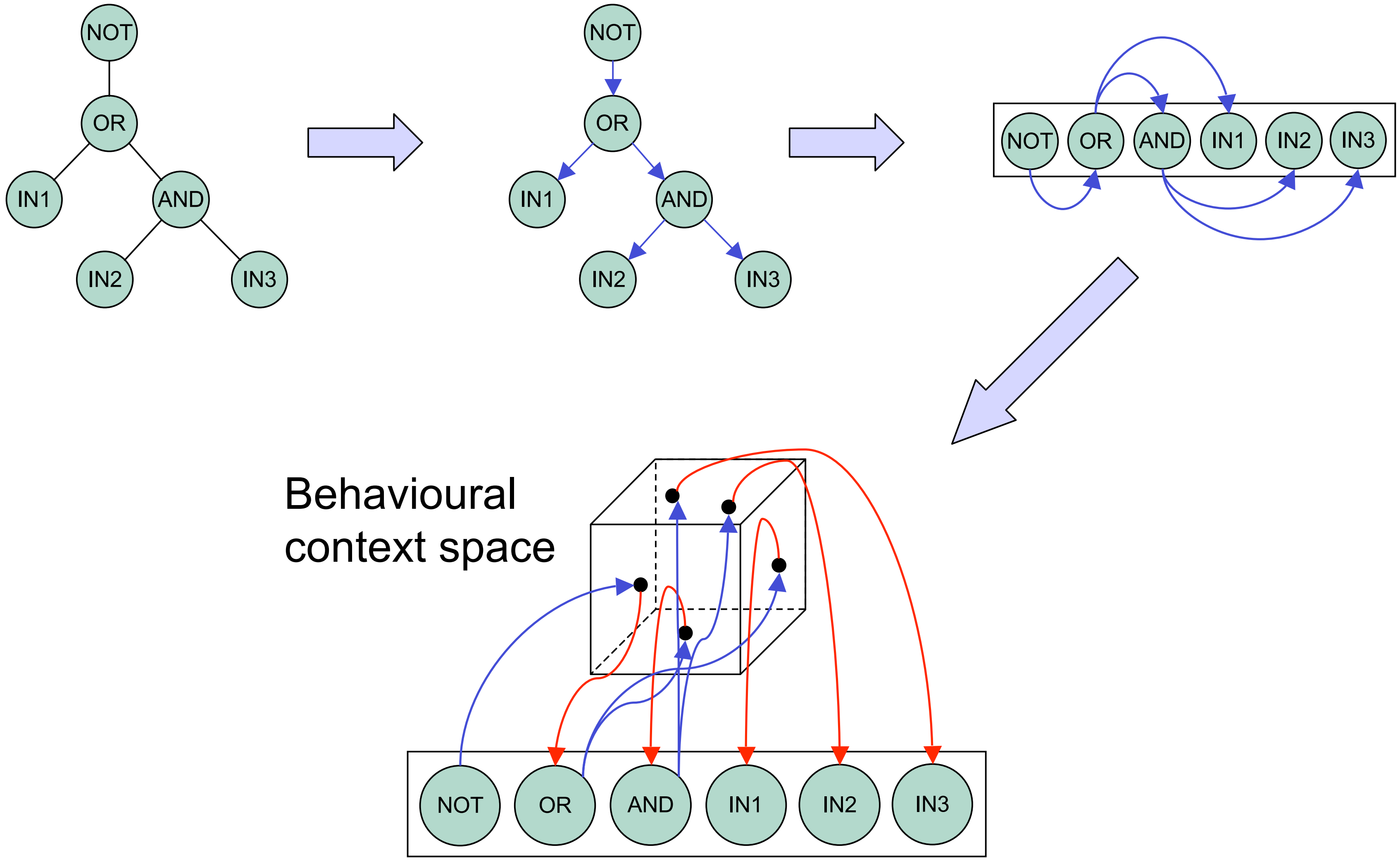
Implicit Context



Implicit Context



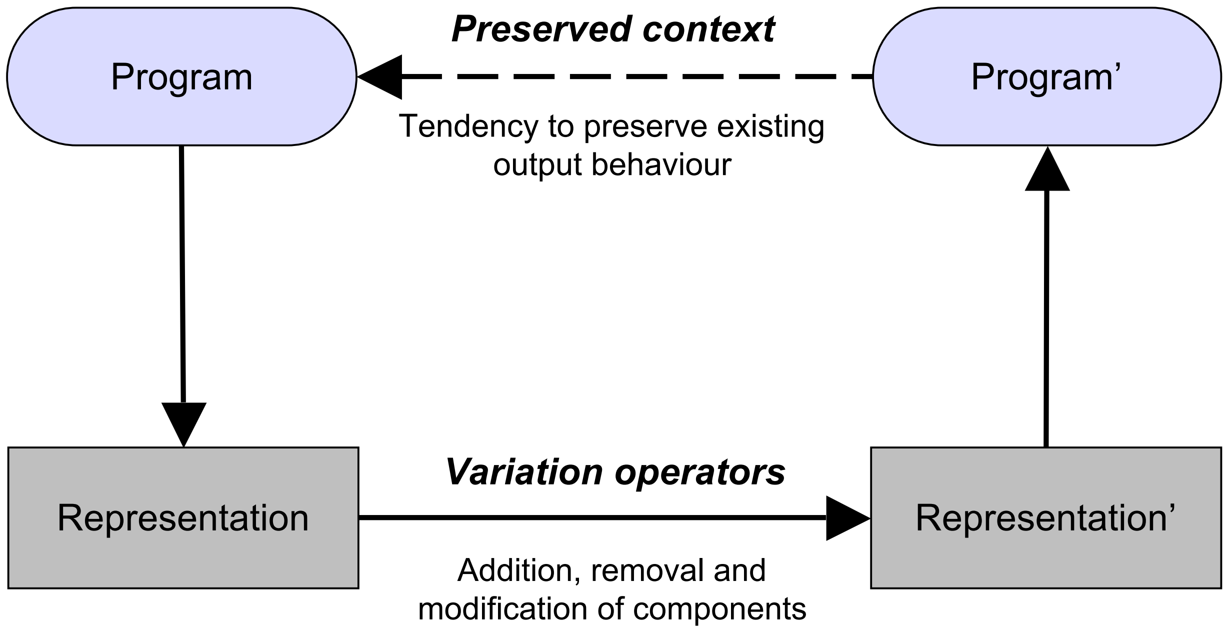
Implicit Context



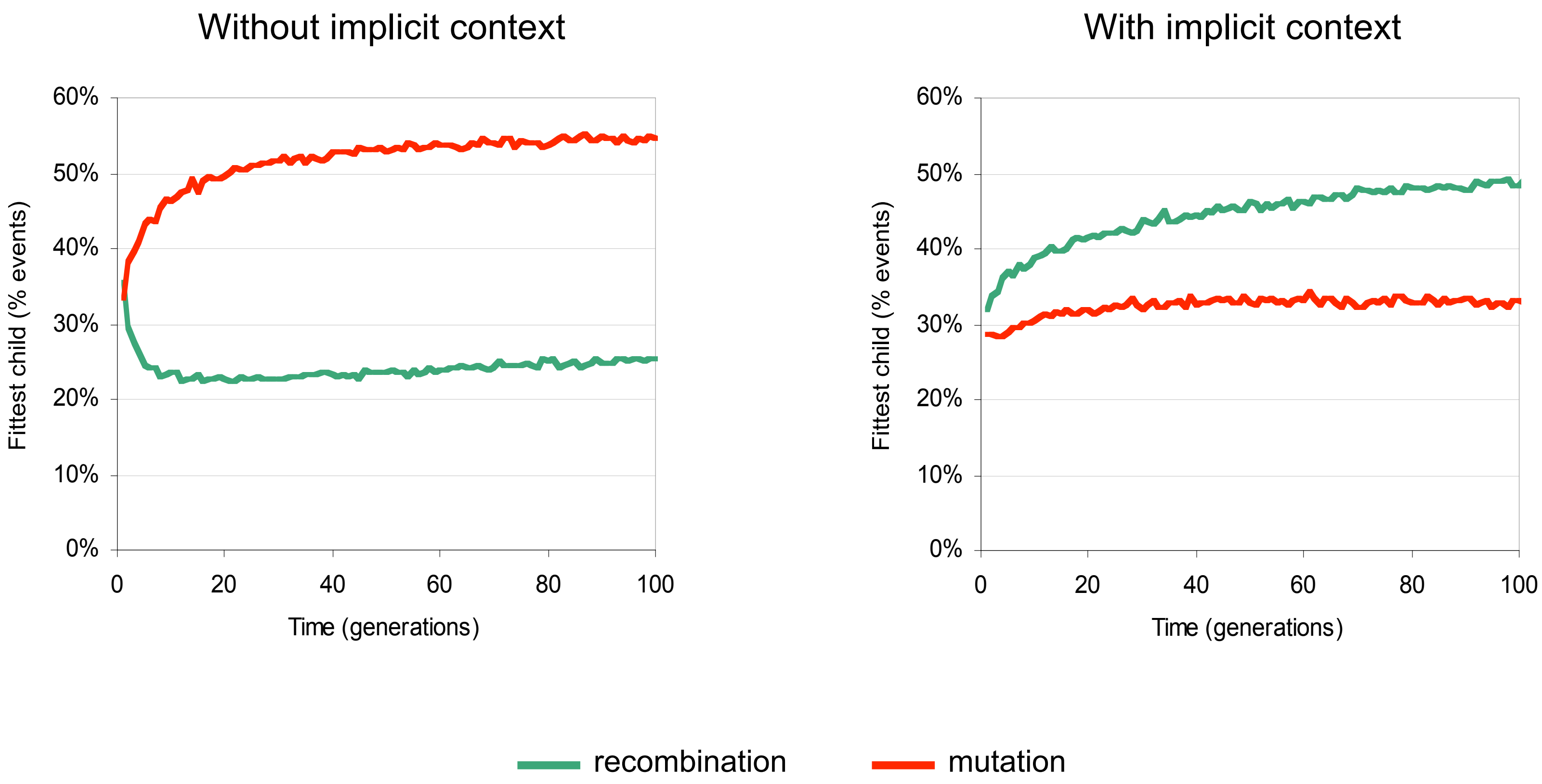
Implicit Context

Variation filtering

– Secondary selection through self-organisation



Implicit Context



Conclusions

Biological systems are evolvable!

– Lots of decoupling

Many other systems are not

– Not designed to evolve, so why should they?

– Can be made evolvable, to an extent…

Evolvability is the missing link!

Conclusions

Is there more to life than the variation-selection paradigm?

– No

– and Yes