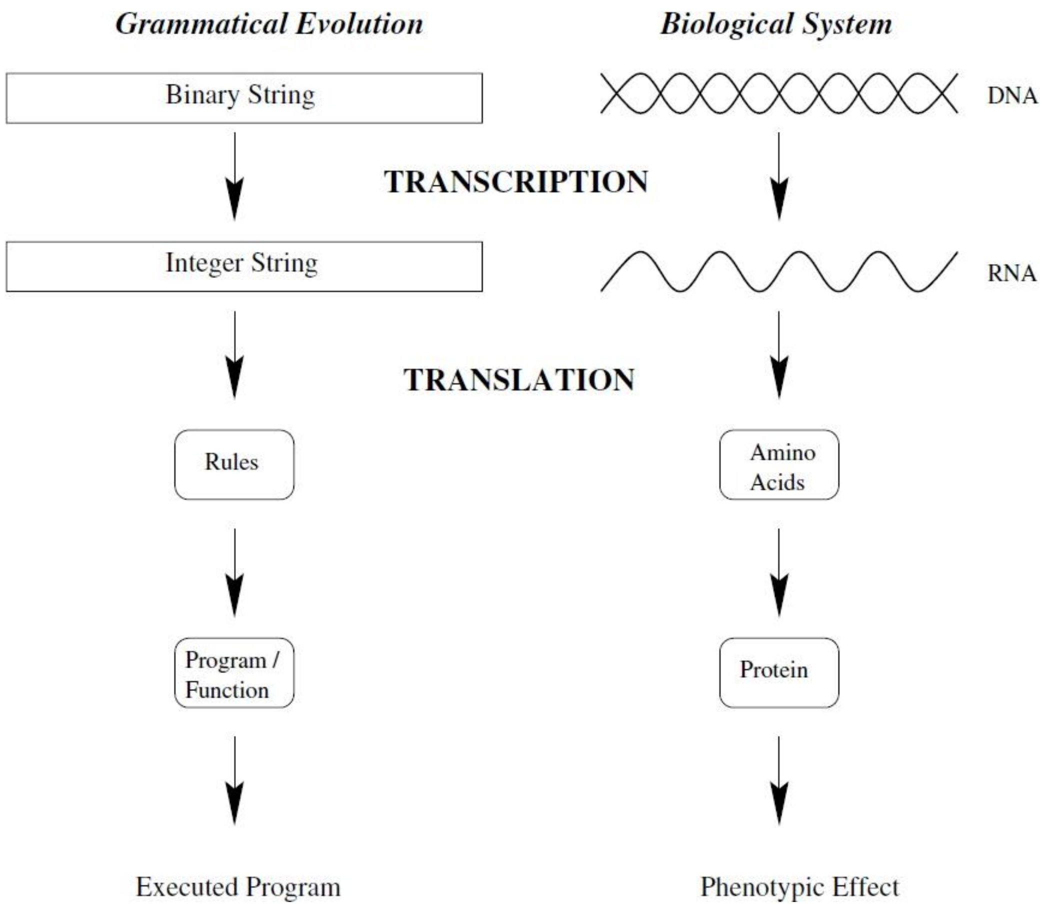
GRAMMATICAL EVOLUTION

Peter Černo

Grammatical Evolution (GE)

- Is an **evolutionary algorithm** that can evolve **programs**.
- Representation: **linear genome** + predefined **grammar**.
- Each individual: variable-length **binary string**.
- **Biological Analogy**:
  - Inspired by the biological process of generating a protein.
  - **DNA** contains the information to produce specific proteins.
  - **DNA** = string of **nucleotides** (A, C, G, T).
  - **Codon** = group of **3 nucleotides**, specifies **amino acids**.
  - **Amino acids** = basic building blocks of **proteins**.
  - In order to generate a protein from the sequence of nucleotides in the DNA, the nucleotide sequence is first **transcribed** into an **RNA**.

Biological Analogy



Mapping Process

- In order to use GE a suitable **grammar in BNF** (Backus-Naur form) must initially be defined.
- **Example**: Grammar for Boolean expressions:

```
(A) <expr> ::= (<expr> <biop> <expr>)     (0)
             | <uop> <expr>                 (1)
             | <bool>                       (2)

(B) <biop> ::= and        (0)
             | or         (1)
             | xor        (2)
             | nand       (3)

(C) <uop> ::= not

(D) <bool> ::= true        (0)
             | false       (1)
```

Mapping Process

- The **genotype** is used to select the **production rules**:

  Rule := C **mod** R
- C is the **codon** integer value,
- R is the **number of choices** for the current non-terminal.
- **Example**:

```
(B) <biop> ::= and      (0)
             | or       (1)
             | xor      (2)
             | nand     (3)
```

- If we assume that the codon C = 6, then 6 mod 4 = 2, i.e. we would select the rule (2).

Wrapping

- During the genotype-to-phenotype mapping process, it is possible for individuals to **run out of codons**.
- In this case the **wrap** operator is applied which results in returning the codon reading head **back to first codon**.
- This technique of wrapping the individual draws inspiration from the gene-overlapping phenomenon that has been observed in many organisms.

Simplified Trading System

```
<S> ::= <tradingrule>

<tradingrule>  ::= if(<signal>) {<trade>;} else {<trade>;}

<signal> ::= <value> <relop> <var>
           | (<signal>) AND (<signal>)
           | (<signal>) OR (<signal>)

<value> ::= <int-const> | <real-const>

<relop> ::= <= | >=

<trade> ::= buy
          | sell
          | do-nothing

<int-const> ::= <int-const><int-const>
              | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

<real-const> ::= 0.<int-const>

<var> ::= var0 | var1 | var2 | var3 | var4
        | var5 | var6 | var7 | var8 | var9
```

Mapping Example

- Consider the following genome:

```
42 22 6 104 70 31 13 4 25 9 3 86 44 48 3 27 4 111 56 2
```

- The first codon C = 42.
- The initial non-terminal is <*tradingrule*>.
- As there is only one production rule (42 mod 1 = 0):

```
<tradingrule>  ::= if(<signal>) {<trade>;} else {<trade>;}
```

- It is automatically replaced with the right-hand side:

```
if(<signal>) {<trade>;} else {<trade>;}
```

- Taking the left-most non-terminal <*signal*> there are three possible replacements.

Mapping Example (Continued)

- The codon reading head moves one codon to the right:

  42 **22** 6 104 70 31 13 4 25 9 3 86 44 48 3 27 4 111 56 2

- The second production rule is 22 mod 3 = 1, thus we get:

```
if((<signal>) AND (<signal>)) {<trade>;} else {<trade>;}
```

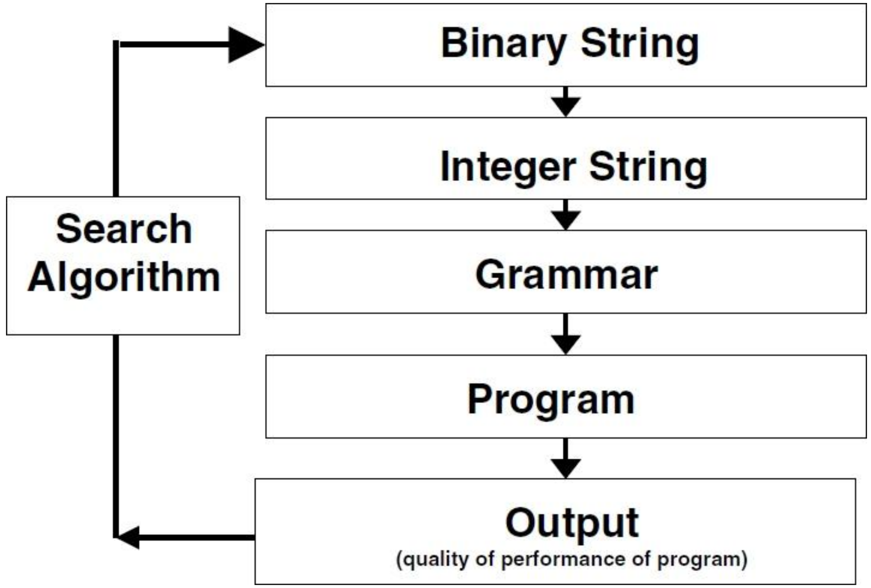
- After reading genome we get the **final trading system**:

```
if((13 <= var5) AND (0.64 <= var3)) {buy();}
else {sell();}
```

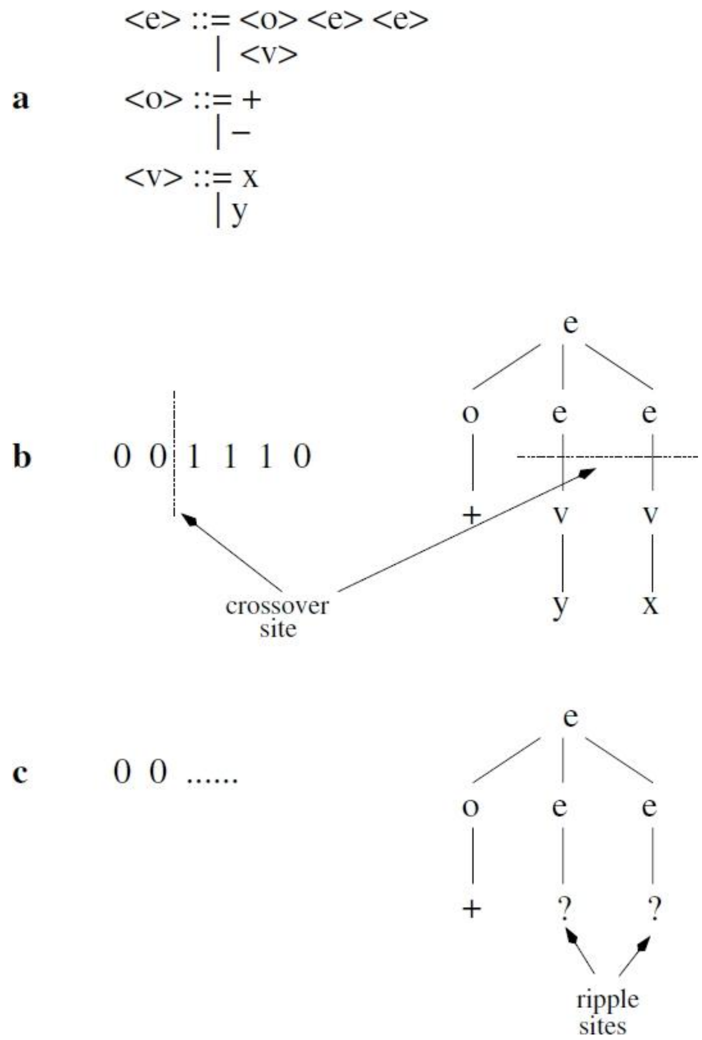
- The variables (*var0* to *var9*) could represent a selection of elements of information drawn from fundamental analysis.
- For instance, *var5* could be a *P/E ratio* and *var3* could represent a *sales growth* over the past 3 years.

Mutation and Crossover

- **Standard genetic operators** such as crossover, mutation and duplication **can be applied** on genotype.
- Therefore an unconstrained **evolutionary search works**!



Ripple Crossover



Recent Developments in GE

- **Alternative Search Engines**:
  - Particle Swarm + Differential Evolution Algorithms.
- **Meta-Grammars (GE^2)**:
  - Grammar that describes the construction of another grammar.
- **πGE**:
  - Replaces the translation process to allow evolution to specify the order in which production rules are mapped.
  - Each codon corresponds to the pair *(nont, rule)*.
  - πGE has shown significant performance gains over the standard GE algorithm on a number of benchmark problem instances.

Case Study

- **Corporate Failure Prediction** Using GE
  - How to uncover **useful rules** which can assist in the prediction of **corporate failure**.
  - Management decisions are not directly observable, but their consequent effect on the financial health of the firm can be observed through the firm’s **financial ratios**.
- **Corporate Failure**:
  - No unique definition exists.
  - Possible definitions range from failure to earn an economic rate of return on invested capital, to legal bankruptcy followed by liquidation of the firm’s assets.
  - Two firms may show a similar financial trajectory towards failure, but one firm may be acquired whilst the other may fail.

Methodology

- **A sample of 178** (89 failed and 89 non-failed) publicly quoted **US firms** was drawn from the period 1991 to 2000 in order to train and test the model.
- **Explanatory Variables**:
  - A subset of 22 of the most commonly used financial ratios:

i. EBIT/Sales
ii. EBITDA/Sales
iii. EBIT/Total Assets
iv. Gross Profit/Sales
v. Net Income/Total Assets
vi. Net Income/Sales
vii. Return on Assets
viii. Return on Equity
ix. Return on Investment
x. Cash/Sales
xi. Sales/Total Assets
xii. Inventory/Cost of Goods Sold
xiii. Inventory/Working Capital
xiv. Fixed Assets/Total Assets
xv. Retained Earnings/Total Assets
xvi. Cash from Operators/Sales
xvii. Cash from Operations/Total Liabilities
xviii. Working Capital/Total Assets
xix. Quick Assets/Total Assets
xx. Total Liabilities/Total Assets
xxi. Leverage
xxii. EBIT/Interest

Methodology

- **GE System Setup**:
  - The construction of classifier system consists of two **components**: **valuation rule** and **cut-off value** (> *0.5* = *fail*, < *0.5* = *non-fail*).
- **Grammar**:

```
<lc> ::= output = <expr> ;

<expr> :: ( <expr> ) + ( <expr> )
       | <coeff> * <var>

<var> ::= var1[index]  | var2[index]  | var3[index]
        | var4[index]  | var5[index]  | var6[index]
        | var7[index]  | var8[index]  | var9[index]
        | var10[index] | var11[index] | var12[index]
        | var13[index] | var14[index] | var15[index]
        | var16[index] | var17[index] | var18[index]
        | var19[index] | var20[index] | var21[index]
        | var22[index]

<coeff> ::=  ( <coeff> ) <op> ( <coeff> )
          | <float>

<op> ::= + | - | *

<float> ::= 20 | -20 | 10 | -10 | 5 | -5 | 4 | -4
          | 3 | -3 | 2 | -2 | 1 | -1 | .1 | -.1
```

Methodology

- The above grammar generates classifiers of the **form**:

```
output = (<some expression>*varX) + (<some expression>*varY) + ...
```

  - The generated rules have a linear form.
  - The grammar definition could be easily altered to allow the construction of non-linear models.
- **LDA Method**:
  - Results obtained from the GE classifier are benchmarked against rules arising from *Linear Discriminant Analysis* (LDA).
  - LDA derives a linear combination of characteristics (variables) which best discriminates between a series of predefined classes.

Results

- **Three series of models** were constructed:
  - Using explanatory variables drawn from one, two, three years (T1, T2, T3) prior to failure.
- For each set of models:
  - 30 runs, population size of 500, 100 generations, one-point crossover (90% prob.), one bit mutation (1% prob.), along with roulette selection.
- The **classification results** show promise:

Years Prior to Failure	In-Sample Accuracy	Out-Of-Sample Accuracy
1	85.9%	80%
2	82.8%	80%
3	75.8%	70%

Evolved Classifiers

- The **best classifiers** evolved for each period:

One Year Prior to Failure:

```
Output = -3*Financial leverage -5*Return on Assets
+3*Inventory/Working Capital-20*Retained Earnings/Total Assets
+4*Total Liabilities/Total Assets
```

Two Years Prior to Failure:

```
Output = -2*Return on Assets+10*Sales/Total Assets-10*Fixed
Assets/Total Assets-2*varEBIT/Interest
```

Three Years Prior to Failure:

```
Output= -4*Return on Assets+20*Sales/ Total Assets-72.9*Cash from
Operations/Sales-10*EBIT/Interest
```

- It is notable that each model employed only a small subset of 22 potential explanatory variables.

Other uses of GE

- **Index Trading**
  - Design of simple trading systems based on **technical indicators**.
  - GE can simultaneously evolve both a good selection of model inputs and a good model form.
  - Moreover, GE produces **human-readable rules** that have the potential to enhance understanding of the problem domain.
- **Adaptive Trading**
  - Rather than employing a single fixed training period, the trading system continues to retrain as new data becomes available using a variant of the **moving window approach**.
  - This permits the system to adapt to **dynamic market conditions**, while maintaining a memory of good solutions that worked well in past market environments.

Other uses of GE

- **Intra-day Trading**
  - Financial markets generate a huge quantity of tick data each day. An actively-traded share on a major exchange may trade multiple times per minute.
  - Traders can see this data in real time and can use it in making trading decisions.
- **Foreign Exchange Trading Rules**
  - The prediction of foreign exchange rates is a difficult task. Many interconnected political and macroeconomic factors impact on the fundamental value of a currency.
  - GE can be used to uncover a series of useful technical trading rules which can be used to trade spot foreign exchange markets.

Other uses of GE

- **Bond Rating**
  - Many large firms use both share and debt capital to provide long-term finance for their operations.
  - The debt capital may be raised from a bank loan, or may be obtained by selling bonds directly to investors.
  - When a company wants to issue traded debt (bonds), it must obtain a credit rating for the issue at least from one recognized rating agency (Standard & Poor’s, Moody’s, etc.).
  - GE could be used to construct a model which can predict the bond rating of a firm.

References

- *Anthony Brabazon*, *Michael O'Neill*:
- **Biologically Inspired Algorithms for Financial Modelling**