

Alessandro Bollini  
bollini@vision.unipv.it

Dipartimento di Informatica e Sistemistica  
Università di Pavia  
Via Ferrata, 1  
27100 Pavia

## Evoluzione Grammaticale



## Programmazione genetica

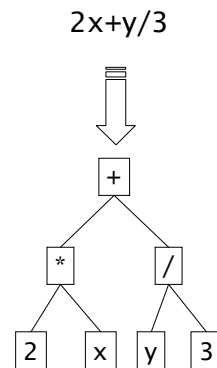
- Algoritmo di base mutuato dagli algoritmi genetici
  - Inizializzazione casuale della popolazione
  - Valutazione della *fitness*
  - Selezione
  - Riproduzione
- Gli individui sono programmi rappresentati come *S-Expression*.

Alessandro Bollini  
Evoluzione Grammaticale



## S-Expression

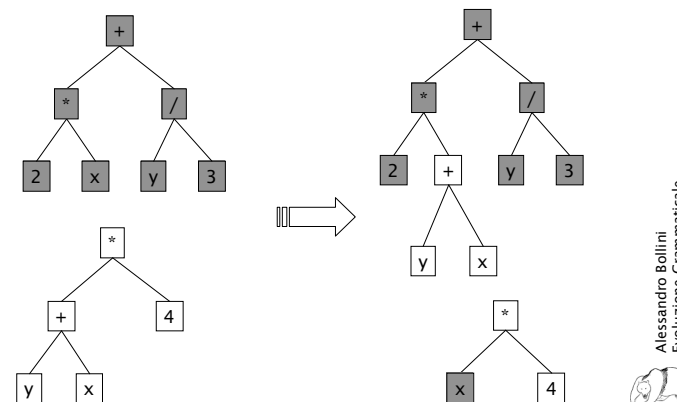
- Funzioni
  - Valori
  - Variabili
- Binding
  - Valori di contesto
  - Effetti collaterali
- Valore
- Operatori genetici
  - Ricombinazione
  - Mutazione
  - ...



Alessandro Bollini  
Evoluzione Grammaticale



## Ricombinazione



Alessandro Bollini  
Evoluzione Grammaticale



## Integrità strutturale

- Gli operatori possono essere applicati solo ad alcuni tipi di dati.
- Possono verificarsi eccezioni nella valutazione degli operatori.
- Soluzioni
  - Chiusura funzionale.
  - Operazioni protette.
  - Gestione delle eccezioni.
  - Programmazione *strongly-typed* tramite operatori estesi.

Alessandro Bollini  
Evoluzione Grammaticale



## Conoscenza di dominio

- Non tutte le soluzioni possibili nello spazio di ricerca sono egualmente ragionevoli
- E' possibile guidare il processo evolutivo integrando la conoscenza di dominio disponibile
- Soluzioni
  - Operatori genetici euristici.
  - Operatori grammaticali.


Alessandro Bollini  
Evoluzione Grammaticale



## Espressività

- Le espressioni rappresentano un sottoinsieme piuttosto ristretto della programmazione
  - Costanti.
  - Programmazione strutturata.
  - Funzioni e procedure.
  - Memoria.
- Soluzioni
  - ERC (*Ephemeral Random Constants*), ER Data.
  - ADF, ADL, ADS, ... (*Automatically Defined Function, Loop, Storage, ...*) tramite operatori estesi

Alessandro Bollini  
Evoluzione Grammaticale



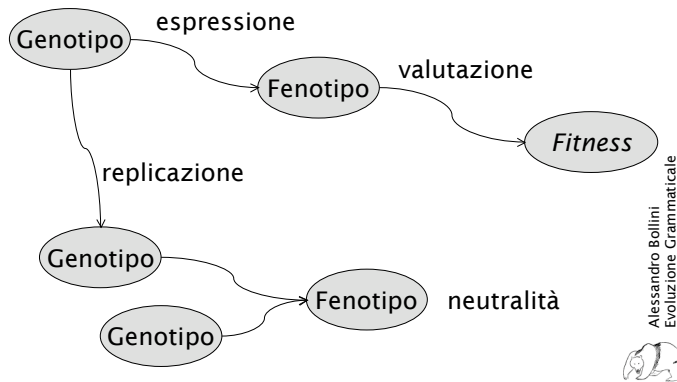
## Evolvibilità

- Le espressioni hanno due ruoli distinti
  - Eseguite come programmi (fenotipo)
  - Modificate tramite operatori genetici (genotipo)
- Non è detto che un buon programma sia anche una buona base di lavoro per gli operatori genetici
- Soluzioni
  - Disaccoppiare la rappresentazione del fenotipo da quella del genotipo

Alessandro Bollini  
Evoluzione Grammaticale



## Genotipo-Fenotipo



## Evoluzione Grammaticale

- Le soluzioni basate su operatori estesi complicano notevolmente la struttura dell'algoritmo di base e sono poco maneggevoli
- L'algoritmo di *evoluzione grammaticale* offre una soluzione alternativa basata su:
  - Descrizione grammaticale dello spazio delle soluzioni
  - Mappatura neutrale da genotipo a fenotipo

Alessandro Bollini  
Evoluzione Grammaticale

## Genotipo

- Il genotipo è rappresentato da una stringa di *bit* di lunghezza generalmente fissa.
- Gli operatori genetici sono ricombinazione (*cross-over*) e mutazione a punto.

1 0 1 0 1 0 0 1 1 1 1 1 0 0 1 0 1 0 0 0  
 1 0 1 0 1 0 0 1 1 1 0 1 1 0 0 1 0 1 0 0 0

Alessandro Bollini  
Evoluzione Grammaticale

## Fenotipo

- Il fenotipo è un programma in un linguaggio arbitrario
- descritto tramite una grammatica *context-free*
- espressa nell'usuale notazione BNF
- e.g. espressioni aritmetiche
  - $S \rightarrow E$
  - $E \rightarrow V$
  - $E \rightarrow (EOE)$
  - $V \rightarrow x$
  - $V \rightarrow y$
  - $0 \rightarrow +$
  - $0 \rightarrow -$
  - $0 \rightarrow *$
  - $0 \rightarrow /$

Alessandro Bollini  
Evoluzione Grammaticale

## Mappatura genotipo-fenotipo

- Genera il programma corrispondente al fenotipo
  - sulla base della grammatica che definisce lo spazio delle possibili soluzioni
  - utilizzando l'informazione contenuta nel genotipo per definire l'albero di derivazione

## Trascrizione

Genotipo

1 0 1 0 1 0 0 1 1 1 1 1 1 0 0 1 0 1 0 0 0

5 2 3 7 4 5 0

Codoni

## Traduzione

Codoni	Fenotipo	E -> x	(0)
5 1	( E )	E -> (EOE)	(1)
2 0	( x O E )	O -> +	(0)
3 3	( x / E )	O -> -	(1)
7 1	( x / ( E O E ) )	O -> *	(2)
4 0	( x / ( x O E ) )	O -> /	(3)
5 1	( x / ( x - E ) )		
0 0	( x / ( x - x ) )		

## Strutture dati e algoritmi

- A livello di genotipo strutture dati e operatori genetici sono condivisi con gli algoritmi genetici
- Possono essere utilizzati gli stessi strumenti di analisi e *tuning*
- Possono essere applicati algoritmi GA avanzati (e.g. messyGA)

## Spazio di ricerca

- Il linguaggio in cui è espresso il fenotipo è arbitrario
- Può contenere tutti i costrutti necessari ad esprimere le soluzioni desiderate
- L'algoritmo di mappatura garantisce che ad ogni genotipo corrisponda un fenotipo sintatticamente corretto

Alessandro Bollini  
Evoluzione Grammaticale



## Flessibilità

- La descrizione grammaticale dello spazio delle soluzioni permette di catturare con un unico strumento i vincoli relativi all'integrità strutturale e alla conoscenza di dominio.
- Senza richiedere interventi sugli operatori genetici
- In modo facilmente comprensibile e intuitivo

Alessandro Bollini  
Evoluzione Grammaticale



## Neutralità

- La neutralità introdotta dall'algoritmo di mappatura permette di adattare indipendentemente genotipo e fenotipo
  - Il genotipo all'azione degli operatori genetici
  - Il fenotipo alla funzione di valutazione della *fitness*
- L'evolubilità può essere controllata modificando l'algoritmo di trascrizione/traduzione
- Indipendentemente dall'algoritmo genetico di base

Alessandro Bollini  
Evoluzione Grammaticale



## Riferimenti

- [www.grammatical-evolution.org](http://www.grammatical-evolution.org)
- M. O'Neill, C. Ryan. *Under the Hood of Grammatical Evolution*. In W. Banzhaf et al. (eds.). GECCO-99: Proceedings of the Genetic and Evolutionary Computation Conference, 1999.
- Symbolic Regression Tool

Alessandro Bollini  
Evoluzione Grammaticale

