

Cartesian Genetic Programming

Jiří Kubalík
Department of Cybernetics, CTU Prague

Substantial part of this material is based on slides for tutorial 'Cartesian Genetic Programming'
presented at GECCO 2013 by J.F. Miller,
see <http://dl.acm.org/citation.cfm?id=2464578>
and the paper J. A. Walker and J. F. Miller: The Automatic Acquisition, Evolution and Reuse of Modules in CGP



<http://cw.felk.cvut.cz/doku.php/courses/a0m33eoa/start>

ECGP: Module List

All active modules are stored in a **module list**.

The module list

- is shared by all individuals in the population,
- is dynamic and has no restrictions on its maximum size,
- is updated in each generation to include only those modules present in the fittest individual.

Any node in a genotype can represent any primitive function or module.

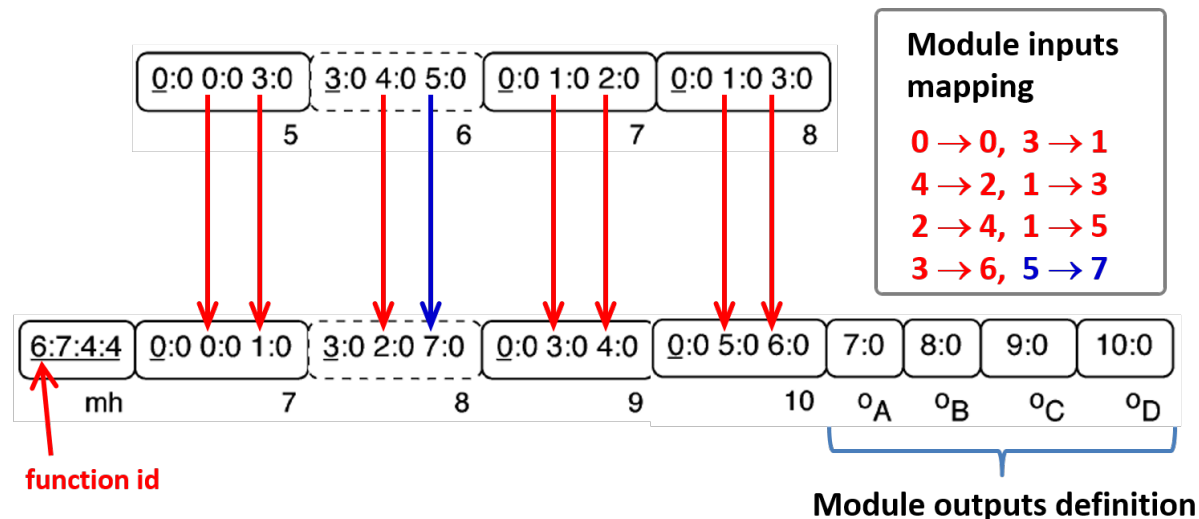
ECGP: Compress Operator - Step 2

The number of module inputs that a module is initialized with is determined by the number of connections leading to the inputs of the nodes being encapsulated into a module.

If there are repeated connections to the output of a previous node, each connection is assigned its own module input.

There are in total 7 inputs and 4 outputs of the module in the example below.

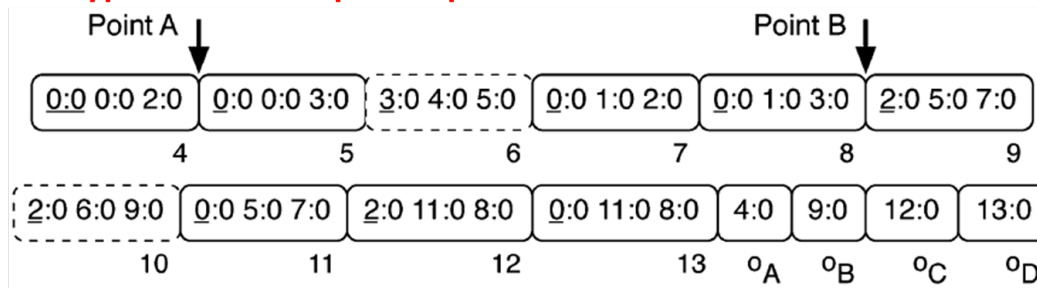
The number of module outputs is determined by the number of connections between the inputs of the later nodes in the genotype (nodes behind the section of the module) and the outputs of the nodes that are going to be encapsulated in the module.



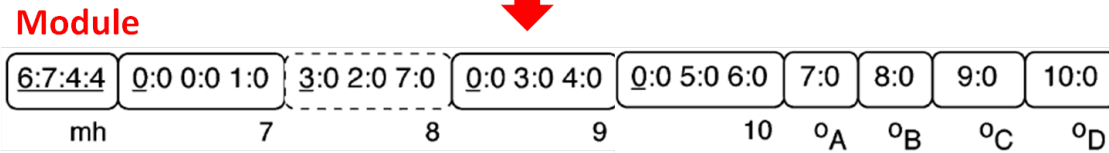
ECGP: Compress Operator - Step 3

A node of type 1 representing the new module is added to the genotype in place of the removed section of nodes.

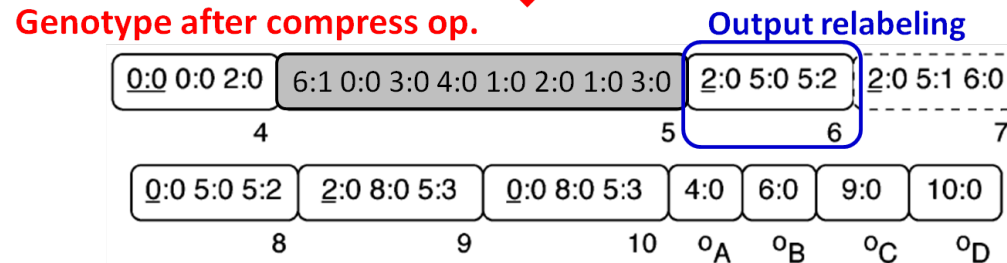
Genotype before compress op.



Module acquisition



Using module in the genotype



ECGP: Expand Operator

An **expand operator** is the reverse of the compress operator, it **deconstructs modules**. It can be **applied only to type 1 nodes**.

The expand operator randomly selects a type 1 node in the genotype and replaces it with the nodes contained in the module represented by the type 1 node.

If this was the only use of the module, the module is removed from the module list.

ECGP: Remarks on Compress and Expand Operators

The inputs of all of the later nodes in the genotype are updated in the final stage of the compress and expand operators, so all of the connections remain intact.

The operators **only make a structural changes to the genotype**, they have no affect on genotype fitness.

CGP: Sources

- Miller, J.F.: GECCO 2013 Tutorial: Cartesian Genetic Programming
<http://portal.acm.org/citation.cfm?id=1389075>
- Home site: <http://www.cartesiangp.co.uk>
- Julian Miller: <http://www.elec.york.ac.uk/staff/jfm7.html>
- Simon Harding: <http://www.cs.mun.ca/~simonh/>
- Lukas Sekanina: <http://www.fit.vutbr.cz/~sekanina/>

Sekanina L., Vašíček Z., Růžička R., Bidlo M., Jaroš J., Švenda P.: Evoluční hardware: Od automatického generování patentovatelných invencí k sebemodifikujícím se strojům. Academia Praha 2009

