

Современные открытые библиотеки для ЭВОЛЮЦИОННЫХ ВЫЧИСЛЕНИЙ

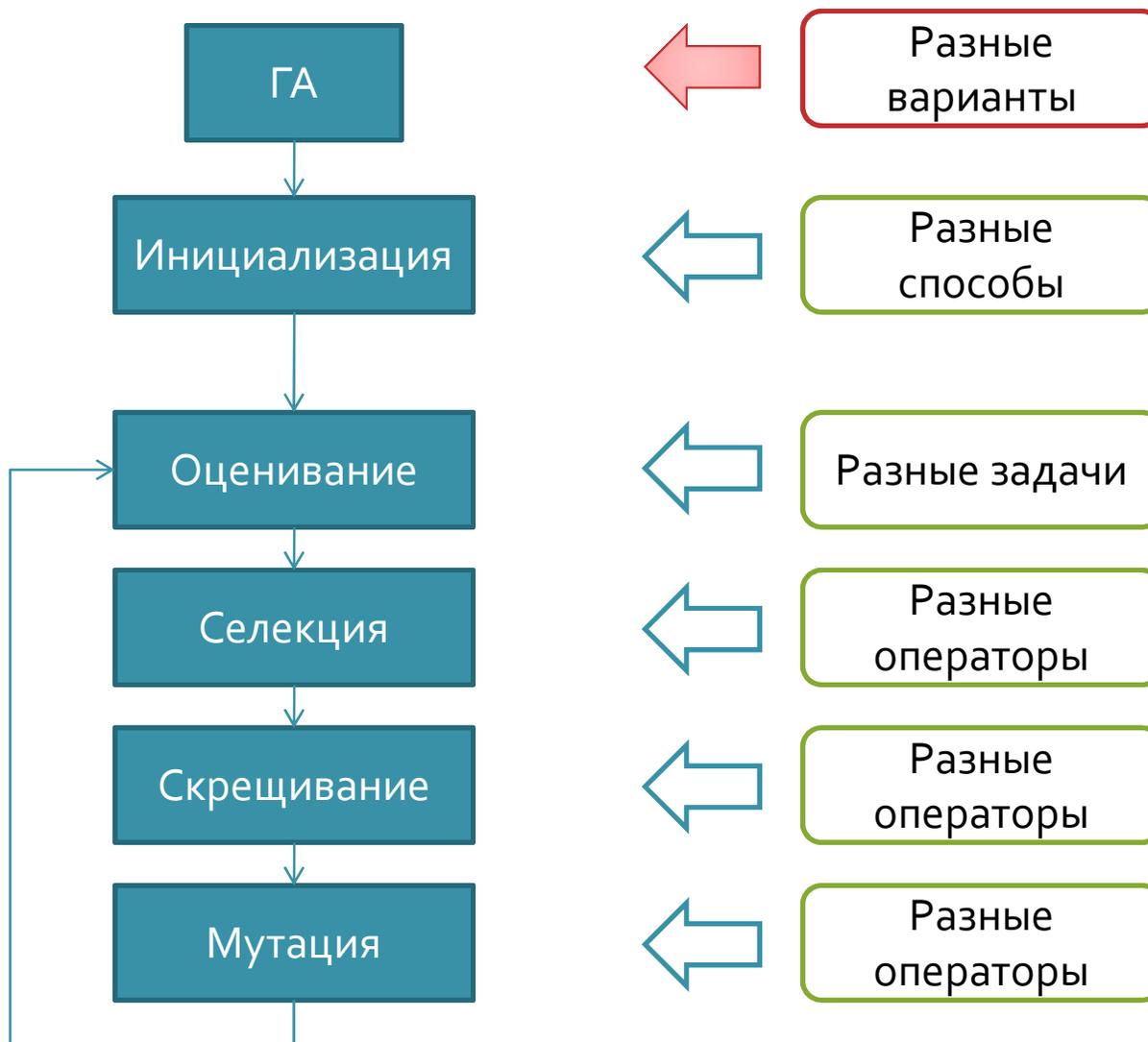
Семинар по эволюционным вычислениям

Докладчик: Цой Ю.Р.

г. Томск, 19 декабря 2009 г.



Почему?



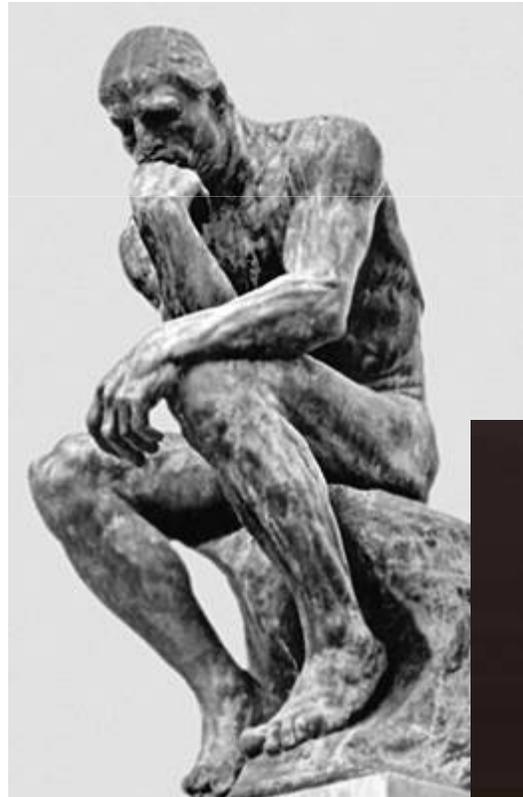
Представьте себе, представьте себе...

Мы хотим сделать исследование ГА или любого другого эволюционного алгоритма.

- Объем копипастинга и закомментаривания/раскомментаривания.
- Нет уверенности, что ничего не забыли
- В результате: опухшая голова
- Нет гарантии, что получили именно те результаты, которые хотели (вдруг что-то забыли, а голова-то уже опухла, и законы Мерфи неумолимы ;)).

Кто виноват?

Надо было раньше думать...



Что делать?

- Высокая степень переиспользуемости (*reusability*) кода в эволюционных алгоритмах (селекция, сбор статистики, общая последовательность применения операторов, формирование отчетов и т.д.).
- Неудобства «обычного» подхода с записыванием в программу всего подряд и «закомментированием» ненужного кода.
- Как следствие, страдают расширяемость и возможности отладки.
- И вообще, копи-пастинг – это для новичков. А мы то – люди грамотные и с опытом 😊

Общие требования

- Единообразная (обобщенная) работа с различными способами кодирования.
- Возможность работы с задачами минимизации и максимизации, а также нетрадиционными способами определения приспособленности (многокритериальные задачи, категориальные оценки).
- Обобщенные операторы.
- Обобщенная эволюционная модель.
- Управление входными параметрами.
- Настраиваемый отчет.

Gagné C., Parizeau M. Genericity in Evolutionary Computation Software Tools: Principles and Case-study // International Journal on Artificial Intelligence Tools. 2006. Vol. 15. No. 2. P. 173-194.

(<http://vision.gel.ulaval.ca/~parizeau/Publications/IJAIT06.pdf>)

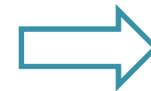
Существующие библиотеки

- Open BEAGLE, C++ (<http://beagle.gel.ulaval.ca/>)
- ECJ, Java (<http://cs.gmu.edu/~eclab/projects/ecj/>)
- Evolving Objects, EO, C++ (<http://eodev.sourceforge.net/>)
- GAlib, C++ (<http://lancet.mit.edu/ga/>)
- PyBrain, Python (<http://pybrain.org/>)
- The Watchmaker Framework, Java (<http://watchmaker.uncommons.org/>)
- EvA2, Java (<http://www.ra.cs.uni-tuebingen.de/software/EvA2/>)
- ECW, C++ (<http://code.google.com/p/ecw/>)
- и многие другие...

Open BEAGLE

- Tree-based genetic programming
- Bit string, integer-valued vector, and real-valued vector genetic algorithms
- Evolution strategy
- Distributed BEAGLE

Это BEAGLE Puppy – собачка
и библиотека для генетического
программирования



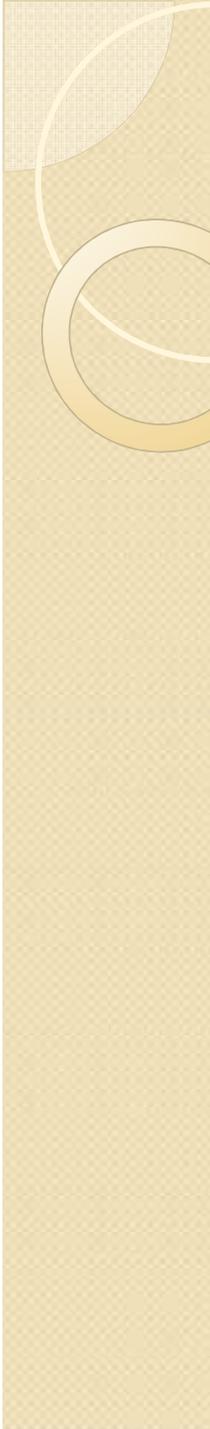


ECJ, Java-based Evolutionary Computation Research System

- Genetic Algorithms
- Genetic Programming
- Evolutionary-Strategies
- Particle Swarm Optimization
- Differential Evolution
- Spatially embedded evolutionary algorithms
- Single- and Multi-population coevolution
- Multiobjective optimization
- Island models over TCP/IP

Evolving Objects, EO

- Paradigm-free (GA, GP, ES, EP)
- Flexible.
- Checkpoints.
- Supports EASEA (EAsea Specification of Evolutionary Algorithms,
<http://sourceforge.net/projects/easea/>)
- Multi-objective EAs (ParadisEO)



GAlib

- Genetic Algorithms (generational, steady-state).
- Island model.
- Speciation (niching).
- Supports various encoding schemes.

PyBrain



- Particle Swarm Optimization (PSO)
- Evolution Strategies (ES)
- Covariance Matrix Adaption ES (CMA-ES)
- Natural Evolution Strategies (NES)
- Fitness Expectation-Maximization (FEM)
- Genetic Algorithms (GA)
- Competitive Co-Evolution
- (Inner/Inverse) Memetic Search
- Multi-Objective Optimization NSGA-II
- И еще: (Stochastic) Hill-climbing, Machine Learning, Reinforcement Learning, Neural Networks



The Watchmaker Framework

- Multi-Threaded Evolution Engine.
- Pluggable Selection Strategies.
- Flexible Evolution Schemes.
- Re-usable Operators for Common Types
- Island Model Evolution Steady-State Evolution
- Interactive Evolutionary Algorithms.
- Distributed Processing.

EvA2

- Monte-Carlo search,
- Hill-Climber,
- Simulated Annealing,
- Genetic Algorithms,
- Evolution Strategies,
- Genetic Programming,
- Differential Evolution,
- Particle Swarm Optimization,
- TRIBES,
- Cluster-based Niching,
- Population Based Incremental Learning,
- Model-Assisted Evolution Strategies,
- Grammatical Evolution





ECW

- Genetic Algorithm.
- Steady-State GA.
- Plus/Comma Evolution strategies.
- Integer and Real coding schemes and operators
- Neuroevolution algorithm (Self-Adaptive Regulatory Network, SARN)

Genericity criteria	ECJ 13	EO 0.9.3a	GAlib 2.4.6	lil-gp 1.1	GPLAB 2	Open BEAGLE 2.2.0
Generic representation	2	2	2	0	0	2
Generic fitness	2	2	0	0	0	2
Generic operations	2	2	1	2	2	2
Generic evolutionary model	2	2	1	1	1	2
Parameter management	2	2	2	1	2	2
Configurable output	2	1	0	1	0	2

(2 = complete, 1 = partial, 0 = missing)

Gagné C., Parizeau M. Genericity in Evolutionary Computation Software Tools: Principles and Case-study // International Journal on Artificial Intelligence Tools. 2006. Vol. 15. No. 2. P. 173-194.

(<http://vision.gel.ulaval.ca/~parizeau/Publications/IJAIT06.pdf>)



Современные открытые библиотеки для ЭВОЛЮЦИОННЫХ ВЫЧИСЛЕНИЙ

Эта презентация:

http://qai.narod.ru/TomskWorkshop/osea_12_2009.pdf