

# Metaheurísticas

**Seminario 1. Ejemplos de resolución de problemas con metaheurísticas: problemas clásicos y reales. Software de metaheurísticas**

---

1. Introducción: Optimización
2. Problemas de Optimización Combinatoria Clásicos
3. Algunos Ejemplos de Problemas de Optimización Reales
4. Software de Metaheurísticas

# 4. Software de Metaheurísticas

---

## Metaheuristic optimization frameworks: a survey and benchmarking

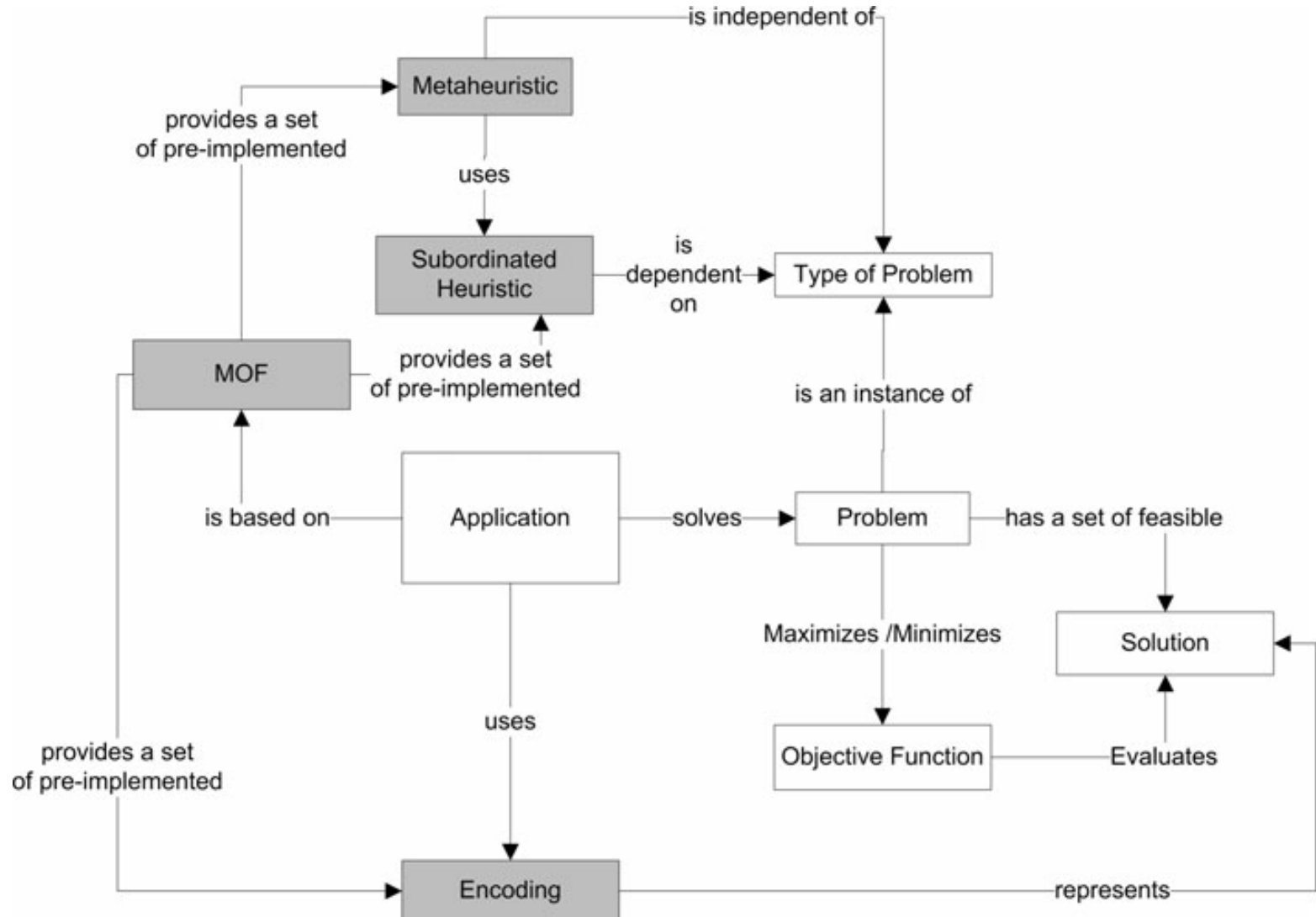
José Antonio Parejo • Antonio Ruiz-Cortés •  
Sebastián Lozano • Pablo Fernandez

Soft Comput (2012) 16:527–561  
DOI 10.1007/s00500-011-0754-8

**Abstract** This paper performs an unprecedented comparative study of Metaheuristic optimization frameworks. As criteria for comparison a set of 271 features grouped in 30 characteristics and 6 areas has been selected. These features include the different metaheuristic techniques covered, mechanisms for solution encoding, constraint handling, neighborhood specification, hybridization, parallel and distributed computation, software engineering best practices, documentation and user interface, etc. A metric has been defined for each feature so that the scores obtained by a framework are averaged within each group of features, leading to a final average score for each frame-

work. Out of 33 frameworks ten have been selected from the literature using well-defined filtering criteria, and the results of the comparison are analyzed with the aim of identifying improvement areas and gaps in specific frameworks and the whole set. Generally speaking, a significant lack of support has been found for hyper-heuristics, and parallel and distributed computing capabilities. It is also desirable to have a wider implementation of some Software Engineering best practices. Finally, a wider support for some metaheuristics and hybridization capabilities is needed.

# 4. Software de Metaheurísticas



# 4. Software de Metaheurísticas

**Table 2** Selected MOFs

Name	Web
EasyLocal (Di Gaspero and Schaerf 2003)	<a href="http://satt.diegm.uniud.it/EasyLocal++/">http://satt.diegm.uniud.it/EasyLocal++/</a>
ECJ (Luke et al. 2009)	<a href="http://cs.gmu.edu/~eclab/projects/ecj/">http://cs.gmu.edu/~eclab/projects/ecj/</a>
EO/ ParadisEO/ MOEO/ PEO (Cahon et al. 2004)	<a href="http://paradiseo.gforge.inria.fr">http://paradiseo.gforge.inria.fr</a> <a href="http://eodev.sourceforge.net/">http://eodev.sourceforge.net/</a>
EvA2 (Kronfeld et al. 2010)	<a href="http://www.ra.cs.uni-tuebingen.de/software/EvA2/">http://www.ra.cs.uni-tuebingen.de/software/EvA2/</a>
FOM (Parejo et al. 2003)	<a href="http://www.isa.us.es/fom">http://www.isa.us.es/fom</a>
HeuristicLab (Wagner 2009)	<a href="http://dev.heuristiclab.com">http://dev.heuristiclab.com</a>
JCLEC (and KEEL) (Ventura et al. 2008)	<a href="http://JCLEC.sourceforge.net">http://JCLEC.sourceforge.net</a> <a href="http://sci2s.ugr.es/keel/">http://sci2s.ugr.es/keel/</a>
MALLBA (Alba et al. 2007)	<a href="http://neo.lcc.uma.es/mallba/easy-mallba/index.html">http://neo.lcc.uma.es/mallba/easy-mallba/index.html</a>
Optimization Algorithm Toolkit (Brownlee 2007)	<a href="http://optalgtoolkit.sourceforge.net">http://optalgtoolkit.sourceforge.net</a>
Opt4j (Martin Lukasiewicz and Helwig 2009)	<a href="http://opt4j.sourceforge.net">http://opt4j.sourceforge.net</a>

- Todos están implementados en Java o C++, con código documentado, permiten la generación de código ejecutable fuera del framework e incluyen al menos dos tipos de metaheurísticas distintas

# 4. Software de Metaheurísticas

Table 8 MOFs Programming languages, platforms and licenses

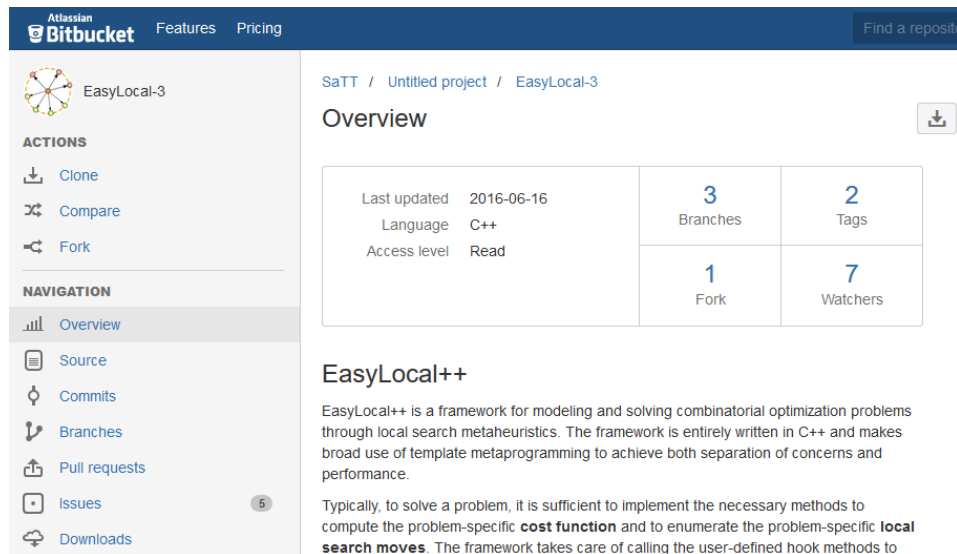
MOF	Prog. Lang.	Platforms	License
EasyLocal	C++	Unix	GPL
ECJ	Java	All	Open Source (Academic free license)
ParadisEO	C++	All (Except for windows if using PEO)	CECILL (ParadisEO) and LGPL (EO)
EvA2	Java	All	LGPL
FOM	Java	All	GPL
HeuristicLab	C#	Windows	GPL
JCLEC (and KEEL)	Java	All	LGPL
MALLBA	C++	Unix	Open source
Optimization Algorithm Toolkit	Java	All	LGPL
Opt4j Martin Lukasiewicz and Helwig (2009)	Java	All	LGPL

Characteristic	Feature	Weight	ECJ	ParadisEO	EvA2	FOM	JCLEC	OAT	Opt4j	EasyLocal	HeuristicLab	MALLBA	Sum
SD/HC	Basic Implementation	0.5	✓	✓	✓	✓		✓		✓	✓	✓	8
	Multi-Start	0.5	✓	✓	✓	✓		✓		✓	✓	✓	7.5
SA	Basic Impl.	0.5		✓	✓	✓			✓	✓	✓	✓	7
	Lineal Annealing	0.1		✓		✓				✓	✓		3
	Exponential/Geometric Annealing	0.1		✓		✓				✓	✓	✓	5
	Logaritmnic Annealing	0.1				✓							1
	Metropolic Acceptance	0.1		✓	✓	✓			✓	✓	✓	✓	7
	Logistic Acceptance	0.1											0
		0.1											0
TS	Basic Impl.	0.3		✓		✓				✓	✓		4
	Recent Features/Moves Based Tabu Memory	0.2		✓		✓				✓	✓		3.5
	Frequency Based Tabu Memory	0.3				✓				✓			2
	Basic Aspiration Criteria	0.2		✓		✓				✓	✓		4
GRASP		1				✓							1
VNS	Basic VNS (VNS)	0.2		✓		✓							2
	Variable Neighborhood Descent (VND)	0.2								✓			1
	Reduced VNS (RVNS)	0.2											0
	VNS with Decomposition	0.2											0
	Skewed VNS (SVNS)	0.2											0
EA	Basic EA Implementation of GA	0.2	✓	✓	✓	✓	✓	✓	✓		✓	✓	9
	Basic EA Implementation of ES	0.2	✓	✓	✓		✓	✓	✓		✓	✓	8
	Basic EA Implementation of GP	0.2	✓	✓	✓		✓	✓	✓		✓		7
	GAVaPS	0.05											0
	Diploid Individuals support	0.05				✓							1
	Coevolution support	0.1	✓										1
	Differential evolution	0.1	✓		✓			✓	✓				4
	Niching Methods	0.1	✓		✓		✓				✓		3.5
PSO	Basic Implementation	0.3	✓	✓	✓				✓		✓	✓	6
	Discrete Variable Support	0.2		✓									1
	Customizable Dynamic Equations	0.2		✓	✓				✓				3
	Topologies	0.2		✓	✓						✓		3
	Lifetime support	0.1											0
AIS	CLONAG	0.25						✓					1
	optIA	0.25											0
	Immune Networks	0.25											0
	Detritic Cell Algorithms	0.25											0
ACS	AS	0.1				✓		✓				✓	3
	ACS	0.2				✓		✓				✓	3
	MMAS	0.4				✓		✓					2
	ASrank	0.2						✓					1
	API	0.1											0
Scatter Search	Basic. Impl.	1			✓								1

# 4. Software de Metaheurísticas

✓ **LOCAL++** (en C++) compuesta por una jerarquía de clases conteniendo *templates* de metaheurísticas de búsqueda, y que permiten especialización para abordar problemas específicos, combinaciones entre ellas o también la creación de nuevas estrategias

✓ **EasyLocal++** (<https://bitbucket.org/satt/easylocal-3>) es una herramienta orientada a objetos (sucesora de LOCAL++) para desarrollar metaheurísticas de búsqueda local compuesta de clases, que implementa partes invariantes de las estrategias que se especializan mediante clases concretas con la parte dependiente del problema específico



Atlassian Bitbucket Features Pricing Find a repository

EasyLocal-3

ACTIONS

- Clone
- Compare
- Fork

NAVIGATION

- Overview
- Source
- Commits
- Branches
- Pull requests
- Issues
- Downloads

SaTT / Untitled project / EasyLocal-3

Overview

Last updated	2016-06-16	3	2
Language	C++	Branches	Tags
Access level	Read	1	7
		Fork	Watchers

EasyLocal++

EasyLocal++ is a framework for modeling and solving combinatorial optimization problems through local search metaheuristics. The framework is entirely written in C++ and makes broad use of template metaprogramming to achieve both separation of concerns and performance.

Typically, to solve a problem, it is sufficient to implement the necessary methods to compute the problem-specific **cost function** and to enumerate the problem-specific **local search moves**. The framework takes care of calling the user-defined hook methods to

✓ C++ y Unix

✓ **P1:** Búsqueda local, Tabú search.

# 4. ECJ

✓ **ECJ** (<http://cs.gmu.edu/~eclab/projects/ecj/>) es un framework implementado en Java, diseñado para ser muy flexible y eficiente, con casi todas las clases (y todos sus ajustes) determinados dinámicamente en tiempo de ejecución por los parámetros proporcionados por el usuario. Todas las estructuras en el sistema están dispuestos para ser fácilmente modificable.

✓ Java

✓ **P2:** Algoritmos evolutivos

*Útil para consultar y obtener código*

## ECJ 24 and 25

### A Java-based Evolutionary Computation Research System

By Sean Luke, Liviu Panait, Gabriel Balan, Sean Paus, Zbigniew Skolicki, Rafal Kicinger, Elena Popovici, Keith Sullivan, Joseph Harrison, Jeff Bassett, Robert Hubley, Ankur Desai, Alexander Chircop, Jack Compton, William Haddon, Stephen Donnelly, Beenish Jamil, Joseph Zelibor, Eric Kangas, Faisal Abidi, Houston Mooers, James O'Beirne, Khaled Ahsan Talukder, Sam McKay, James McDermott, Jason Zou, Anson Rutherford, David Freelan, and Ermo Wel.

ECJ is a research EC system written in Java. It was designed to be highly flexible, with nearly all classes (and all of their settings) dynamically determined at runtime by a user-provided parameter file. All structures in the system are arranged to be easily modifiable. Even so, the system was designed with an eye toward efficiency.

ECJ is developed at George Mason University's [ECLab](#) Evolutionary Computation Laboratory. The software has nothing to do with its initials' [namesake](#), *Evolutionary Computation Journal*. ECJ's sister project is [MASON](#), a multi-agent simulation system which dovetails with ECJ nicely.

### New Paper!

The new directions for ECJ (starting with Version 25) were presented at GECCO 2017 in the paper [ECJ Then and Now](#).

### Two Versions of ECJ!

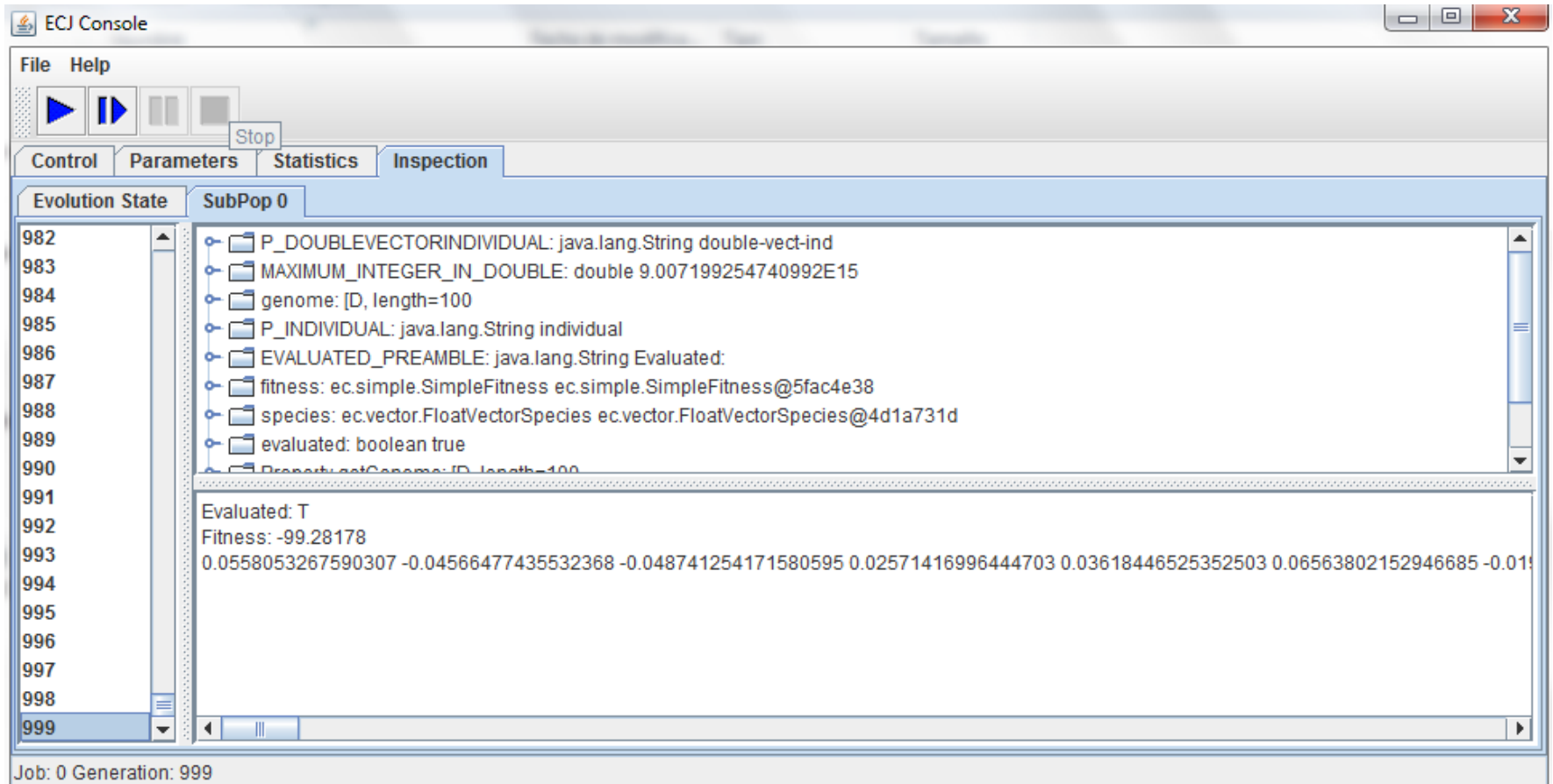
The ECJ team has received a [three-year NSF grant](#) to improve and enhance ECJ. As the first stage of this process, we are releasing two different versions. Version **24** is the final backward-compatible version of ECJ prior to enhancements. It's a legacy version. Version **25** is the new version of ECJ going forward. It has many non-backward-compatible features, but if you're new to ECJ, it is the one you should be using.

### Features

✓ ***Incluye mailing lists***



# 4. ECJ



✓ Ejemplo de ejecución con windows del ejecutable **ecj.bat**  
Algoritmo genetico estacionario usando el fichero de parametros:  
*ecj\ec\app\ecsuite\steady.params* (problema Rastrigin)

# 4. PARADISEO



A Software Framework for Metaheuristics

✓ **ParadisEO** (<http://paradiseo.gforge.inria.fr/>) es otro framework orientado a objetos (C++) que genera código portable a Windows, Linux, Unix y MacOSX. Incorpora técnicas muy diversas como búsquedas locales, algoritmos evolutivos, particle swarm optimization, metaheurísticas paralelas, etc. ***Separa claramente los métodos de los problemas***, lo que proporciona una gran potencia de reutilización de código y diseño

✓ **P1:** Búsqueda local, Tabu search y **P2:** Algoritmos evolutivos

✓ **INSTALACIÓN**

- Descarga: <http://paradiseo.gforge.inria.fr/index.php?n=Download.Download>

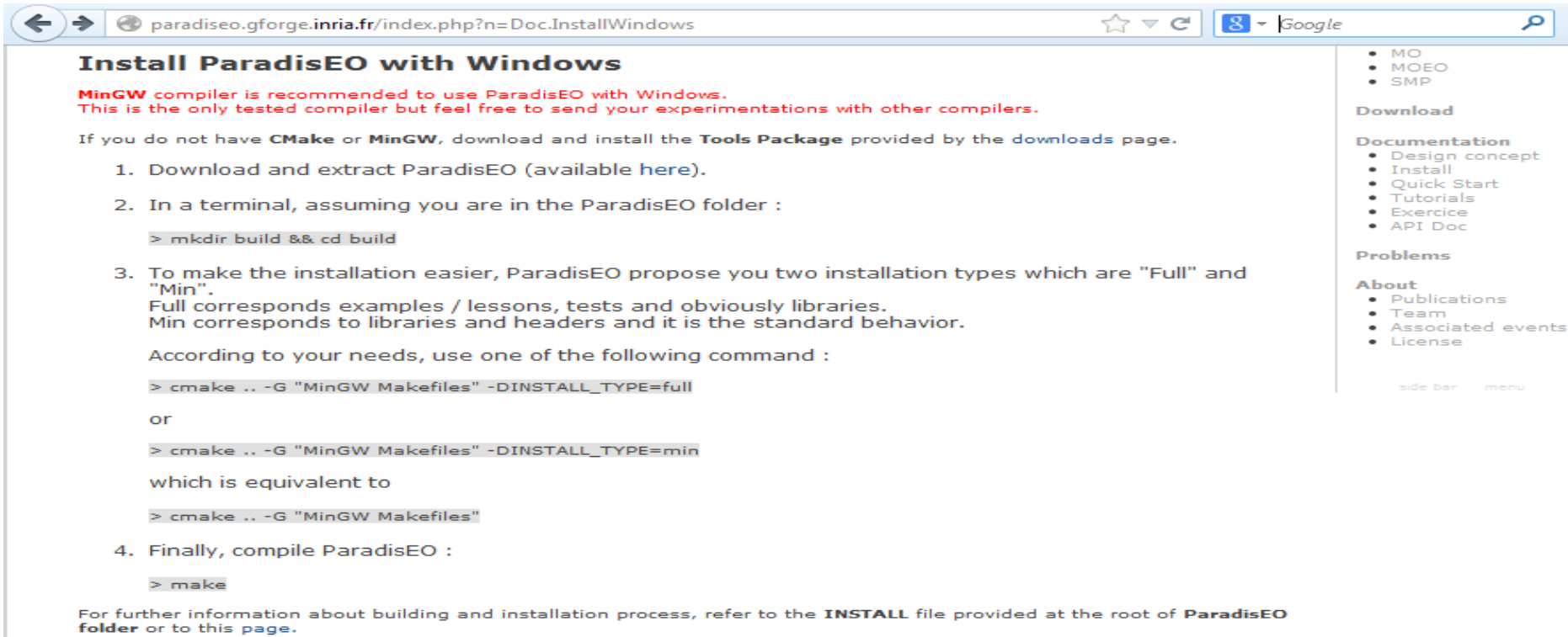
Fuentes para Windows (Cuidado: Error en instalador de windows):  
<https://gforge.inria.fr/frs/download.php/31733/ParadisEO-2.0.1.zip>

Pasos windows: <https://gforge.inria.fr/frs/download.php/31733/ParadisEO-2.0.1.zip>

✓ La web tiene **ejercicios** y incluye **mailing lists**

***Útil para implementar y  
consultar-obtener código***

# 4. PARADISEO



The screenshot shows a web browser window with the address bar displaying `paradiseo.gforge.inria.fr/index.php?n=Doc.InstallWindows`. The page title is "Install ParadiseO with Windows". The main content area contains the following text:

**MinGW** compiler is recommended to use ParadiseO with Windows.  
This is the only tested compiler but feel free to send your experimentations with other compilers.

If you do not have **CMake** or **MinGW**, download and install the **Tools Package** provided by the [downloads](#) page.

1. Download and extract ParadiseO (available [here](#)).
2. In a terminal, assuming you are in the ParadiseO folder :  

```
> mkdir build && cd build
```
3. To make the installation easier, ParadiseO propose you two installation types which are "Full" and "Min".  
Full corresponds examples / lessons, tests and obviously libraries.  
Min corresponds to libraries and headers and it is the standard behavior.  
According to your needs, use one of the following command :  

```
> cmake .. -G "MinGW Makefiles" -DINSTALL_TYPE=full
```

  
or  

```
> cmake .. -G "MinGW Makefiles" -DINSTALL_TYPE=min
```

  
which is equivalent to  

```
> cmake .. -G "MinGW Makefiles"
```
4. Finally, compile ParadiseO :  

```
> make
```

For further information about building and installation process, refer to the **INSTALL** file provided at the root of **ParadiseO** folder or to this [page](#).

The right sidebar contains a navigation menu with the following links:

- MO
- MOEO
- SMP
- Download
- Documentation
  - Design concept
  - Install
  - Quick Start
  - Tutorials
  - Exercice
  - API Doc
- Problems
- About
  - Publications
  - Team
  - Associated events
  - License

At the bottom of the sidebar, there are links for "side bar" and "menu".

*build\mo\tutorial\Lesson4>testSimpleTS.exe*

*Tabu Search 1:*

*-----*

*initial: 5 8 0 7 6 1 3 2 4 5*

*STOP in moTimeContinuator: Reached maximum time [2/2]*

*final: 0 8 1 5 0 6 3 7 2 4*

*Tabu Search 2:*

*-----*

*initial: 3 8 1 7 4 3 0 2 6 5*

*STOP in moTimeContinuator: Reached maximum time [3/3]*

*final: 0 8 3 1 7 4 6 0 2 5*

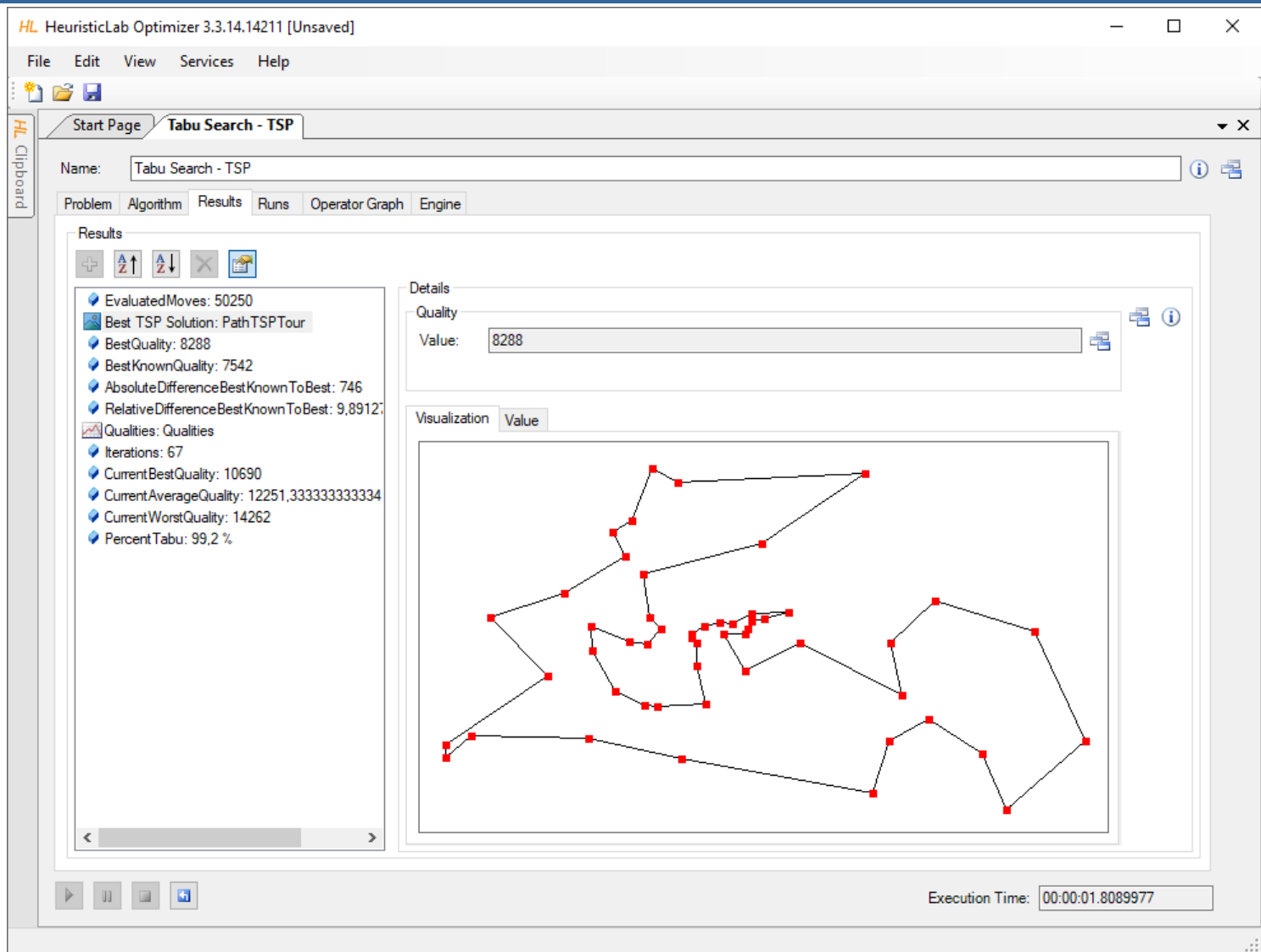
**Ejemplo TABUSEARCH**

<http://paradiseo.gforge.inria.fr/index.php?n=Doc.Tutorials>

- ✓ **HeuristicLab** (<http://dev.heuristiclab.com>) es una herramienta de código abierto para el desarrollo de metaheurísticas realizada en Microsoft .NET y C#. Presenta un interfaz gráfico que permite ajustar y extender los algoritmos para un problema completo sin necesidad de escribir código. Incorpora las búsquedas por trayectorias más conocidas así como particle swarm optimization y distintos algoritmos evolutivos
- ✓ **P1:** Tabu search y **P2:** Algoritmos evolutivos
- ✓ **Descarga:** <http://dev.heuristiclab.com/trac/hl/core/wiki/Download>
- ✓ La web tiene **manuales** y **videos de ayuda**

***Útil para implementar y  
consultar-obtener código***

# 4. HEURISTICLAB



*Ejemplo TabuSearch con problema TSP*

# 4. FOM



✓ **FOM** (Framework for Optimization using Metaheuristics, <http://www.isa.us.es/fom>) es un entorno dirigido a objetos (Java) desarrollado por los autores del artículo comparativo. Incorpora los métodos habituales de trayectorias simples, GRASP y VNS, algoritmos evolutivos y de optimización mediante colonias de hormigas.

✓ **P1:** Tabu Search y GRASP, **P2:** Algoritmos evolutivos

✓ Descargar (**error** en el fichero .jar y no se incluyen códigos):

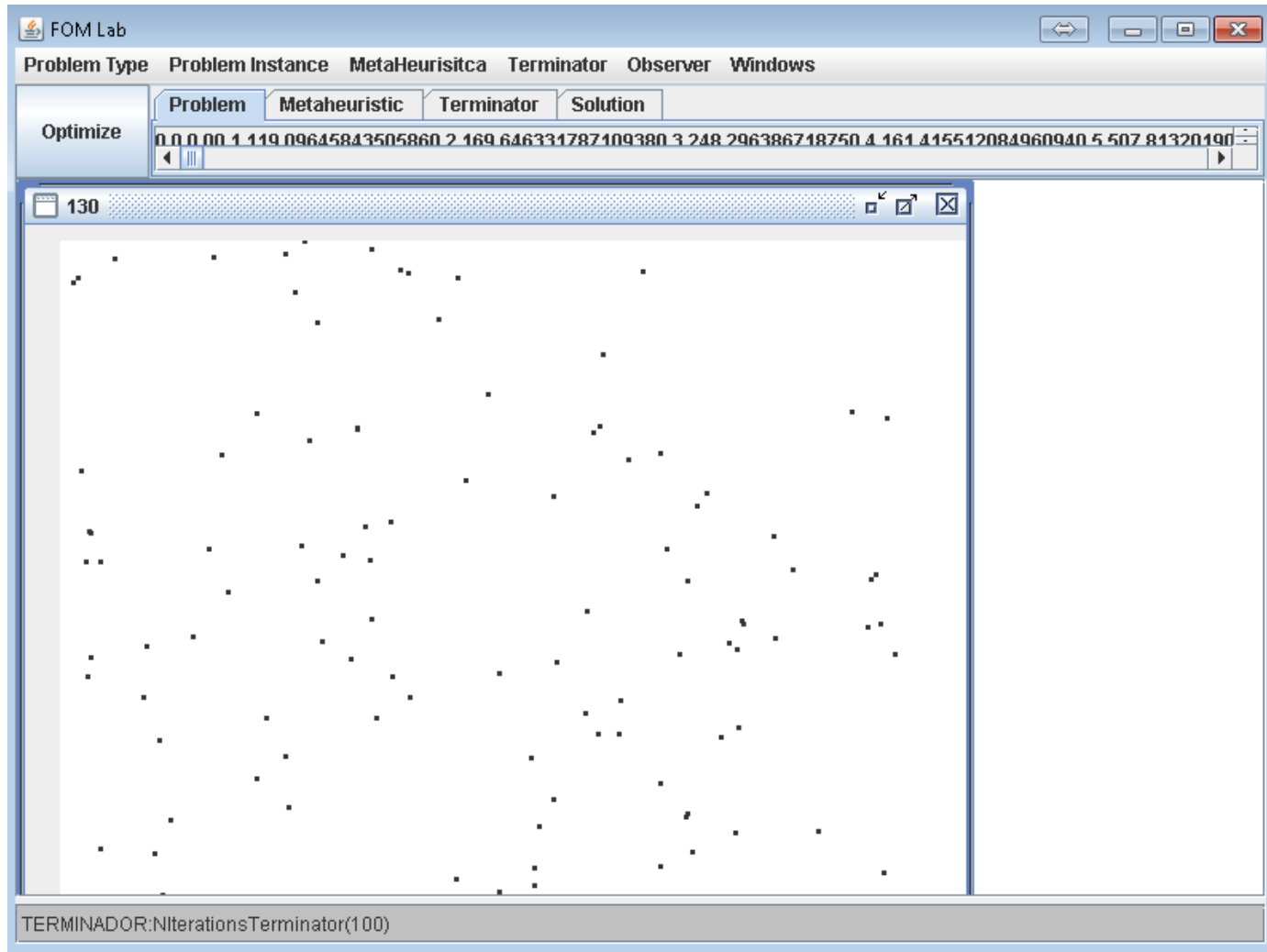
<http://moses.us.es/tools/fom/0.5/fom-0.5.zip>

✓ Descargar (incluye códigos)

<http://moses.us.es/tools/fomlab/0.5/FOMLab-0.5-singlezip-src.tar.gz>

***Útil consultar-obtener código***

# 4. FOM



***Ejemplo con FOM no es posible probarlo debido a los errores***

Manual desactualizado, porque no coincide lo que explica con el código descargado 15

# 4. JCLE

---

✓ **JCLE** (<http://JCLEC.sourceforge.net>) es un sistema de software para investigación en Evolutionary Computation, desarrollado en el lenguaje de programación Java. Proporciona un marco de software de alto nivel para hacer cualquier tipo de algoritmo evolutivo (EA), el apoyo a los algoritmos genéticos (codificación binaria, entero y real), la programación genética y la programación evolutiva.

✓ **P2:** Algoritmos evolutivos

✓ Descargar

```
git clone git://git.code.sf.net/p/jclec/git jclec
```

✓ Descargar: jclec4-base, jclec4-classification y jclec4-tutorial

***Útil consultar-obtener código***



# 4. JCLEC

← [jclec.sourceforge.net/index.php?option=com\\_content&view=article&id=17&Itemid=9](http://jclec.sourceforge.net/index.php?option=com_content&view=article&id=17&Itemid=9)



## MAIN MENU

- Home
- Features
- Documentation
- Examples
- Download
- VisualJCLEC
- EC Software Units

## LATEST NEWS

The alpha version of the VisualJCLEC is available

[VisualJCLEC](#)

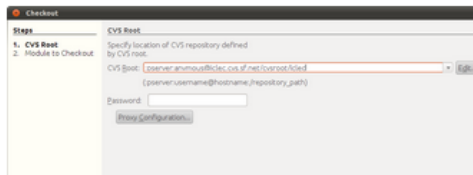
## Import as project in NetBeans

You can download JCLEC and its modules using CVS and import them as NetBeans projects.

1. Team -> CVS -> Checkout

Fill the form using the following data:

CVS Root: `server: anonymous@jclec.sourceforge.net:/cvsroot/jclec`



## Ejemplo GA con el problema TSP

`java -jar jclec4-tutorial.jar examples\TSP.cfg`

```
CA: Administrador: Símbolo del sistema

D:\Trabajo\2013-2014\Metaheurísticas\software\jclec\jclec4-tutorial>java -
jar jclec4-tutorial.jar examples\TSP.cfg ! more
Initializing job
Algorithm started
Generation 0 Report
Best individual: OrderArrayIndividual <genotype = <40 39 5 26 7 45 16 9 41 17 33
27 12 32 13 46 19 15 35 14 43 22 47 34 3 48 6 20 30 31 0 29 50 10 37 2 18 1 36
21 49 8 23 11 38 44 28 4 42 25 24 51>, fitness = net.sf.jclec.fitness.SimpleValu
eFitness@277663d6 [value=24315.32802657221]
Worst individual: OrderArrayIndividual <genotype = <24 28 33 37 42 4 3 8 25 30 1
3 15 16 48 17 50 31 43 14 22 27 1 5 11 19 2 32 45 49 12 38 39 36 0 34 29 9 46 20
47 35 40 7 23 6 10 18 41 44 26 21 51>, fitness = net.sf.jclec.fitness.SimpleVal
ueFitness@159177b9 [value=30862.4232967202551]
Median individual: OrderArrayIndividual <genotype = <5 42 31 13 4 29 19 17 12 50
21 7 20 44 48 1 38 0 34 39 37 23 28 45 24 40 18 27 9 26 25 43 32 2 30 22 49 41
3 35 15 10 33 8 6 36 47 16 14 11 46 51>, fitness = net.sf.jclec.fitness.SimpleVa
lueFitness@5b65afa5 [value=26177.4299583850881]
Average fitness = 27691.720564097326
Fitness variance = 2192924.377113104

Generation 50 Report
Best individual: OrderArrayIndividual <genotype = <37 34 22 20 17 40 8 32 35 15
19 6 41 28 49 30 31 0 29 14 39 36 11 4 47 45 38 46 12 13 25 9 7 48 21 1 16 2 23
43 33 51 27 26 10 50 42 18 44 24 3 5>, fitness = net.sf.jclec.fitness.SimpleValu
eFitness@752dbbf [value=15265.9109860753761]
Worst individual: OrderArrayIndividual <genotype = <5 34 22 20 17 40 8 32 35 15
-- Más --
```

# 4. EVA2

- ✓ **EVA2** (<http://www.ra.cs.uni-tuebingen.de/software/EvA2/>) es un framework escrito en Java para los Algoritmos Evolutivos.
- ✓ **P2:** Algoritmos evolutivos
- ✓ Descargar <https://gitlab.cs.uni-tuebingen.de/eva2/eva2>
- ✓ [http://www.ra.cs.uni-tuebingen.de/software/EvA2/down\\_direct.htm](http://www.ra.cs.uni-tuebingen.de/software/EvA2/down_direct.htm)

EBERHARD KARLS  
UNIVERSITÄT  
TÜBINGEN

MATHEMATISCH-  
NATURWISSENSCHAFTLICHE FAKULTÄT  
FACHBEREICH INFORMATIK

Suche →

Uni A-Z →



CS-HOME CHAIR STAFF RESEARCH PROJECTS SOFTWARE PUBLICATIONS TEACHING THESES VACANCIES ADDRESS

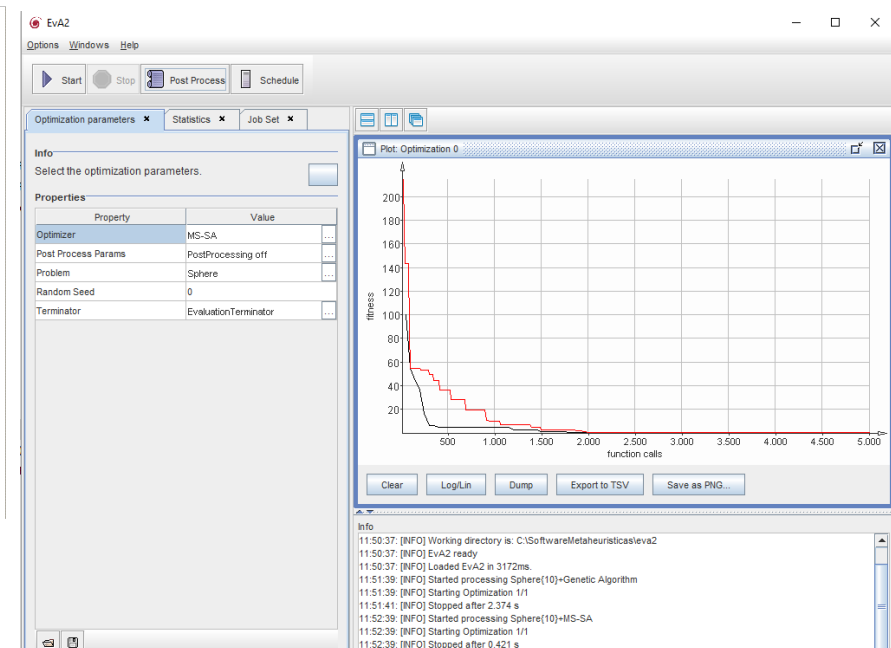
Home > Fakultäten > Mathematisch-Naturwissenschaftliche Fakultät > Fachbereich Informatik > Cognitive Systems

**COGNITIVE SYSTEMS**

About the Chair  
Staff  
Research Projects  
Software  
ToxDBScan  
MARCAR DataTools  
RPPApipe  
GWAS Pathway Identifier



A Java™ based framework for Evolutionary Algorithms -- formerly known as JavaEvA/EvA



# 4. MALLBA

✓ **MALLBA** (<http://neo.lcc.uma.es/mallba/easy-mallba/index.html>) es una librería implementada en C++ (para UNIX) para problemas de optimización combinatoria (que incluye métodos, heurísticas e híbridos) que pueden trabajar con el **paralelismo** de una manera fácil y, al mismo tiempo, eficiente.

MALLBA LIBRARY v2.0

✓ **P2:** Algoritmos evolutivos

✓ Descargar:

<http://neo.lcc.uma.es/mallba/easy-mallba/source/mallba.tar.gz>

Table of Contents	
•	Mallba Library <ul style="list-style-type: none"><li>◦ Introduction</li><li>◦ Mallba Architecture</li><li>◦ Implementation</li><li>◦ How to Install Mallba</li></ul>
•	Algorithms <ul style="list-style-type: none"><li>◦ <a href="#">Genetic Algorithm (GA)</a></li><li>◦ <a href="#">Simulated Annealing (SA)</a></li><li>◦ <a href="#">CHC Method</a></li><li>◦ <a href="#">Evolution Strategy (ES)</a></li><li>◦ <a href="#">Ant Colony Optimization (ACO)</a></li><li>◦ <a href="#">GA+SA Hybrid Algorithm</a></li><li>◦ <a href="#">Cooperative Local Search (CLS)</a></li><li>◦ <a href="#">Particle Swarm Optimization (PSO)</a></li></ul>
•	Examples <ul style="list-style-type: none"><li>◦ MAXSAT</li><li>◦ ONEMAX</li><li>◦ Sphere Function</li><li>◦ Rastrigin Function</li></ul>
•	Appendix: EA parameters <ul style="list-style-type: none"><li>◦ Selection Methods</li><li>◦ Intra-population Operators</li><li>◦ Inter-population Operators</li></ul>
•	Download Mallba v2.0 <a href="#">[Download]</a>
•	Mallba Groups Sites: <a href="#">MÁlaga</a> , <a href="#">La Laguna</a> , <a href="#">BArcelona</a>

→ Last Updated: 7/6/06

✓ La última actualización fue hace años. Para poder utilizar los códigos es necesario arreglar algunos fallos. Por ejemplo: incluye los archivos de encabezado de la iostream antigua (iostream.h) en lugar de utilizar los encabezado de la nueva iostream estándar de C++ (<iostream>)

# 4. OAT

✓ **Optimization Algorithm Toolkit (OAT)** (<http://optalgtoolkit.sourceforge.net>) es un conjunto herramientas para desarrollar, evaluar, experimentar y jugar con algoritmos de optimización en problemas estándar. El software implementado en Java incluye implementaciones de referencia de algoritmos, gráficos, visualizaciones, y mucho más.

✓ **P2:** Algoritmos evolutivos

✓ Descargar (ejecutable y fuentes):

<http://downloads.sourceforge.net/optalgtoolkit>

✓ Entorno gráfico amigable

✓ Difícil la adaptación a otros problemas

## Optimization Algorithm Toolkit

[Home](#) | [Screenshots](#) | [About](#) | [Support](#)

**Explore** the behaviours of classical and state-of-the-art optimization algorithms.

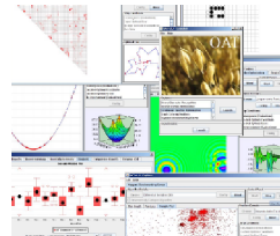
**Experiment** with algorithms against benchmark problems using powerful statistical tools.

**Play** with parameters, graph performance, and learn about computational intelligence.

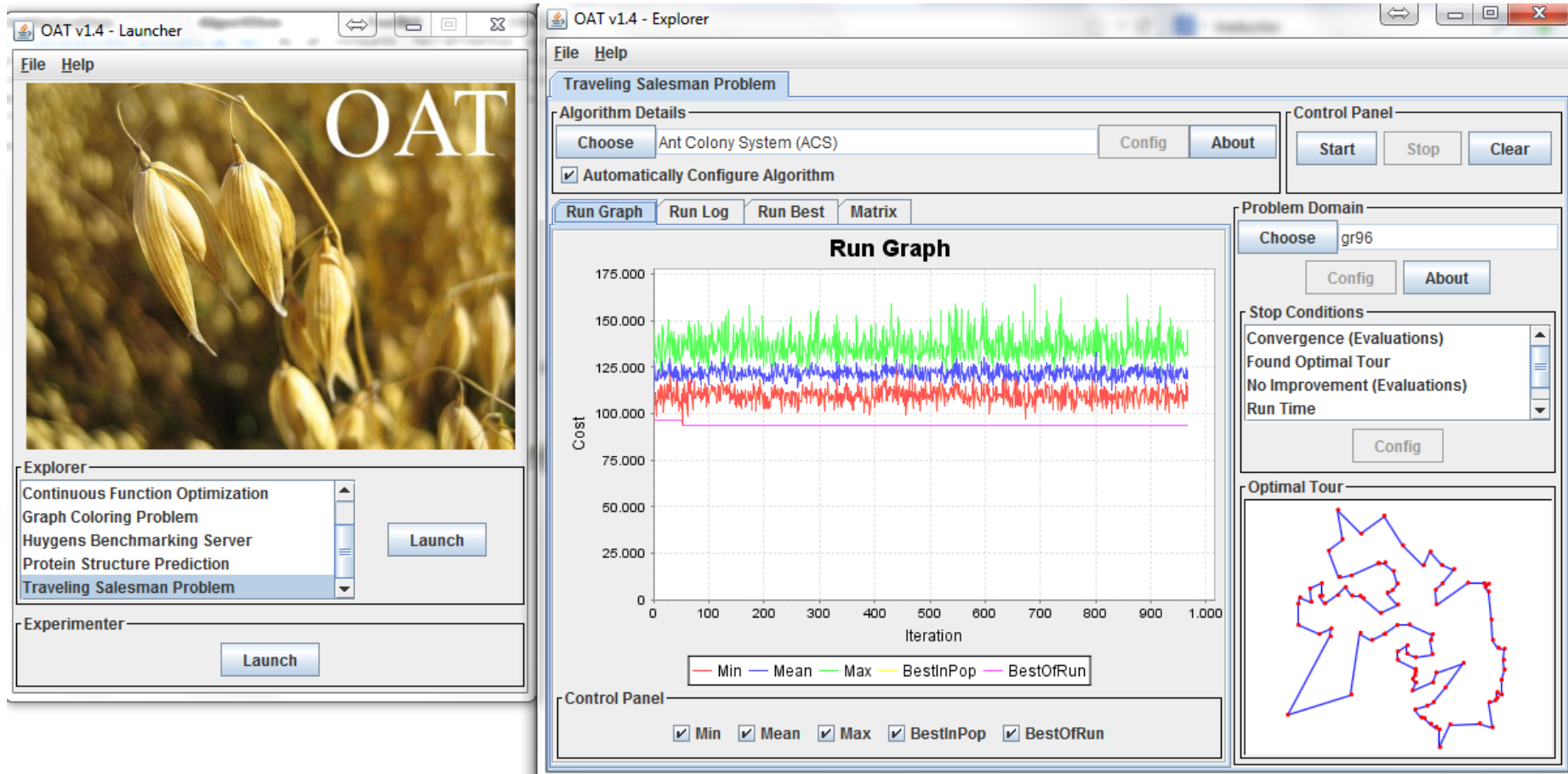
**Download Now!**  
(Requires Java)

SOURCEFORGE.NET®

©2006-2008 Jason Brownlee



# 4. OAT

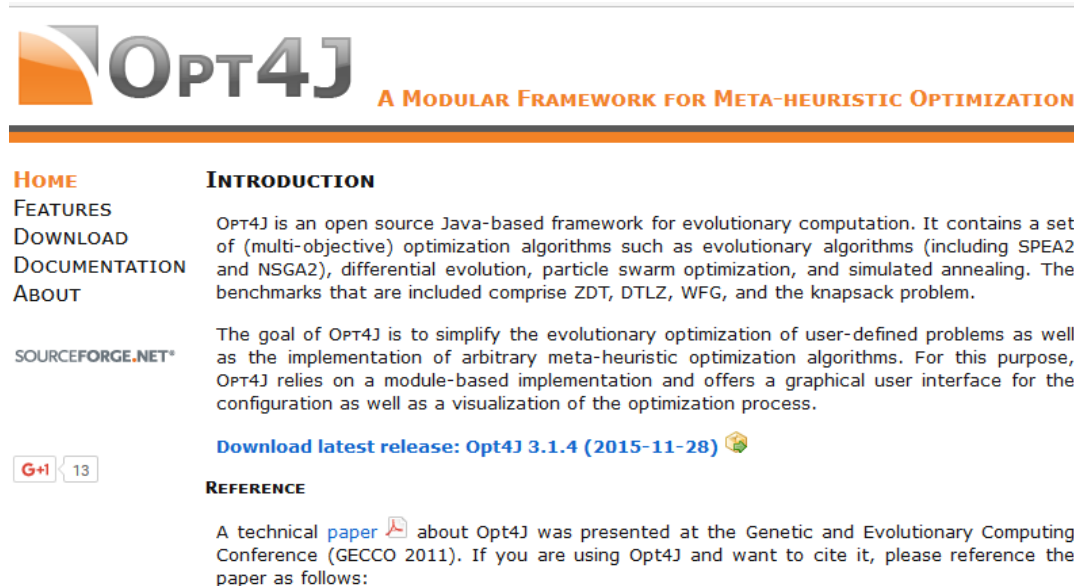


*Ejemplo: Sistema de Colonia de Hormigas con TSP*

# 4. Opt4J

✓ **Opt4J A Modular Framework for Meta-heuristic Optimization** (<http://opt4j.sourceforge.net/>) es framework escrito en Java para computación evolutiva, que contiene un conjunto de algoritmos de optimización (algoritmos evolutivos, evolución diferencial, simulated annealing, etc...). Opt4J tiene una implementación basada en módulos y ofrece una interfaz gráfica de usuario para la configuración, así como una visualización del proceso de optimización.

✓ **P2: Algoritmos evolutivos**




The screenshot shows the Opt4J website homepage. At the top, there is a logo with the text "OPT4J" in a large, bold, sans-serif font, preceded by a stylized orange and grey graphic. Below the logo, the tagline "A MODULAR FRAMEWORK FOR META-HEURISTIC OPTIMIZATION" is written in a smaller, orange, sans-serif font. A horizontal orange line separates the header from the main content area. On the left side, there is a vertical navigation menu with links: "HOME", "FEATURES", "DOWNLOAD", "DOCUMENTATION", and "ABOUT". The "HOME" link is highlighted in orange. To the right of the menu, the "INTRODUCTION" section is visible. It contains two paragraphs of text. The first paragraph describes Opt4J as an open source Java-based framework for evolutionary computation, listing various optimization algorithms it includes. The second paragraph states the goal of Opt4J is to simplify the evolutionary optimization of user-defined problems. Below the introduction, there is a link to "Download latest release: Opt4J 3.1.4 (2015-11-28)" with a download icon. At the bottom left, there is a "G+1" social media button and a "13" count. At the bottom right, there is a "REFERENCE" section with a link to a technical paper about Opt4J presented at the Genetic and Evolutionary Computing Conference (GECCO 2011).

**HOME**  
FEATURES  
DOWNLOAD  
DOCUMENTATION  
ABOUT


**INTRODUCTION**

Opt4J is an open source Java-based framework for evolutionary computation. It contains a set of (multi-objective) optimization algorithms such as evolutionary algorithms (including SPEA2 and NSGA2), differential evolution, particle swarm optimization, and simulated annealing. The benchmarks that are included comprise ZDT, DTLZ, WFG, and the knapsack problem.

The goal of Opt4J is to simplify the evolutionary optimization of user-defined problems as well as the implementation of arbitrary meta-heuristic optimization algorithms. For this purpose, Opt4J relies on a module-based implementation and offers a graphical user interface for the configuration as well as a visualization of the optimization process.

[Download latest release: Opt4J 3.1.4 \(2015-11-28\)](#) 

**REFERENCE**

A technical [paper](#)  about Opt4J was presented at the Genetic and Evolutionary Computing Conference (GECCO 2011). If you are using Opt4J and want to cite it, please reference the paper as follows:

# 4. Opt4J

---

✓ **Descargar programa:**

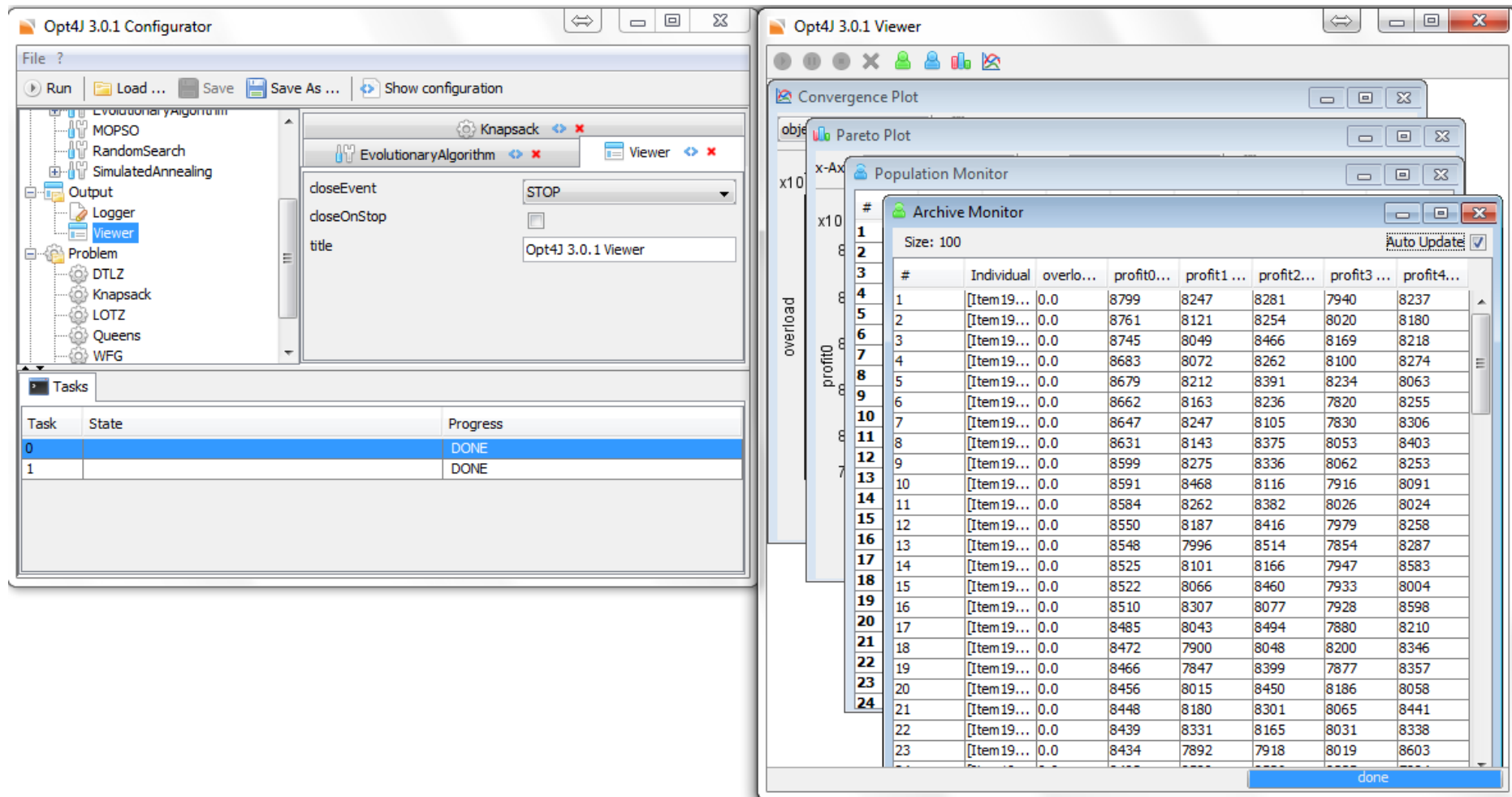
<http://sourceforge.net/projects/opt4j/>

✓ **Descargar fuentes en Java:**

1. Descargar e instalar Mercurial: <http://hginit.com/>
2. Usando Mercurial clonar <http://hg.code.sf.net/p/opt4j/hgroot>

***Útil para consultar y  
obtener código***

# 4. Opt4J

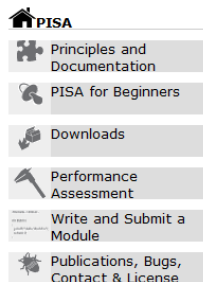
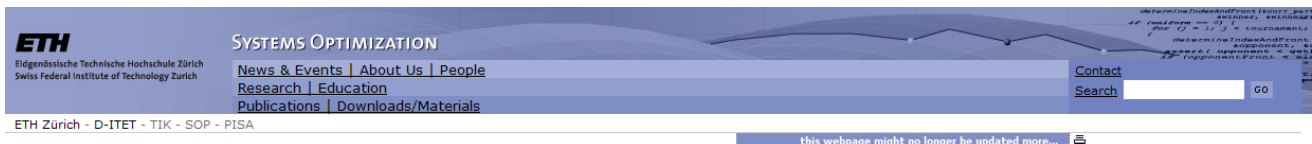


*Ejemplo: Algoritmo Evolutivo con el problema de la mochila*<sup>24</sup>



# 4. PISA

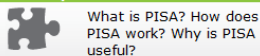
✓ **Pisa** (<http://www.tik.ee.ethz.ch/sop/pisa/>) es una interfaz para algoritmos de búsqueda (dedicada principalmente a la **búsqueda multi-objetivo**). El proceso de optimización se divide en dos módulos. Un módulo contiene todas las piezas específicas para el problema de optimización (por ejemplo, evaluación de soluciones, representación del problema, la variación de las soluciones). El otro módulo contiene las partes que son independientes del problema de optimización (principalmente el proceso de selección). Estos dos módulos se implementan como programas separados que se comunican a través de archivos de texto.



## PISA

### A Platform and Programming Language Independent Interface for Search Algorithms

#### Principles and Documentation



What is PISA? How does PISA work? Why is PISA useful?

#### PISA for Beginners



The first steps in order to get familiar with PISA.

#### Downloads



Download Selectors, Variators and Performance Assessment Tools

#### Performance Assessment



Compare algorithms on benchmark problems using appropriate indicators and statistical methods.

#### Write and Submit a Module



How to write your own PISA module.

#### Miscellaneous



List of Publications, Bugs, Contact Information and License.

#### Crucial Bugfix

Jaroslav Hajek pointed out a severe bug in the **WFG selector**, please redownload the module if your version is older than 2010/02/03.

✓ Fuentes en C, Java, Matlab.

✓ Plataformas: Solaris, Windows, Linux.

# 4. Software de Metaheurísticas

---

## ■ Otros:

✓ **HotFrame** (Heuristic Optimization Framework, <http://www1.uni-hamburg.de/IWI/hotframe/hotframe.html>) proporciona componentes adaptables en C++, incluyendo diversas metaheurísticas y una arquitectura de colaboración entre las distintas componentes y clases específicas de aplicaciones y permitiendo la hibridación y la incorporación de nuevas metaheurísticas. *¡¡Sin versiones nuevas desde 2003!!*

✓ **CPLEX** es una biblioteca comercial estándar de resolución de problemas de optimización desarrollada por IBM ILOG. Ganó el primer concurso internacional de la Sociedad de Investigación Operativa estadounidenses INFORMS en 2004. Incorpora *técnicas clásicas* (Simplex). Incluye interfaces para C++, C#, Java, Python, Excel y Matlab.

✓ **GLPK** (<http://www.gnu.org/software/glpk/>) es un software libre en ANSI C que implementa el método de Simplex revisado

# 4. Software de Metaheurísticas

---

- **Para desarrollar las prácticas de la asignatura se podrá emplear el software que se desee**, bien sea cualquier framework existente o bien código desarrollado por el propio alumno o bajado de Internet
- El profesor de prácticas proporcionará distintos códigos básicos de metaheurísticas desarrollados en C
- El alumno deberá indicar el software considerado en su documentación de prácticas y proporcionar las fuentes y los ejecutables realizados