

Análisis del consumo energético de los Algoritmos Evolutivos: hacia una computación evolutiva energéticamente eficiente.

F. Fernández de Vega
Universidad de Extremadura, Spain.



Cada vez que me toca viajar... (evostar 2023)

Mejor 28 h 1 día 23 días 6 días 3 h y 10

Sevilla

Aeropuerto de Viena-Schwechat

Sevilla—Viena, Austria

Directos (2 a la semana) 3 h 10 min

Conexión 5 h 10 min o más

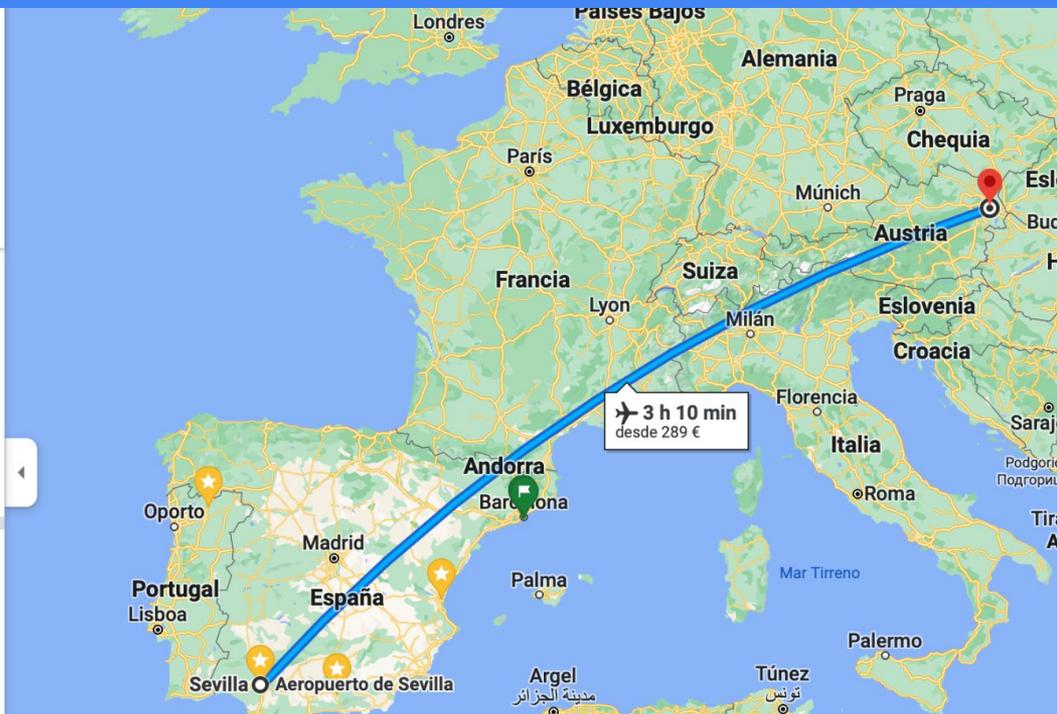
Precio de ida y vuelta, 19–23 abr desde 288 €

Vueling, Lufthansa, Iberia...

[Ver resultados en Google Flights](#)

Explora Aeropuerto de Viena-Schwechat

🍴 🛏️ 🍷 🍹 ...



Cada vez que me toca viajar ... al trabajo

☰

📍 Almendralejo, 06200, Badajoz, España

📍 Mérida, 06800, Badajoz, España

+ Añadir destino

Salir ahora

OPCIONES

📱 Enviar indicaciones al teléfono

🚗 por A-66 **20 min**
La ruta más rápida, el tráfico habitual 29,6 km

DETALLES



Tengo que tomar decisiones



El vehículo que
utilizo

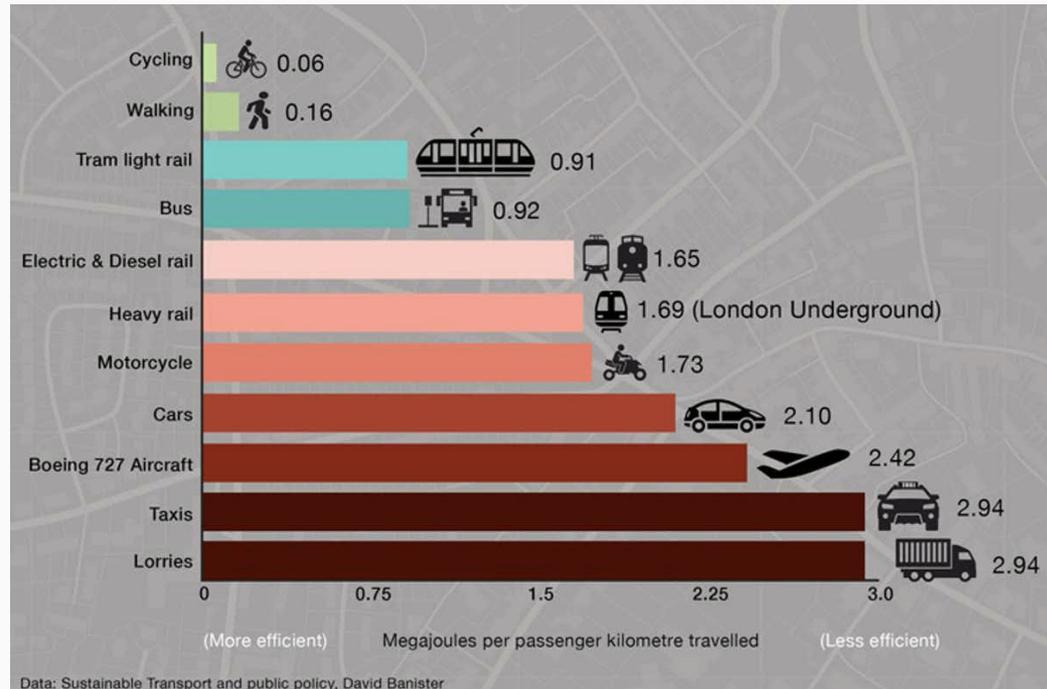
23 minutos de
conducción por
autovía

Decisiones que no puedo tomar :-)

Llegar más rápido en un Bugatti...



Eficiencia energética del transporte



Contenidos

Introducción.

Antecedentes.

Objetivos.

Metodología y Resultados.

Conclusiones

Contenidos

Introducción.

Antecedentes.

Objetivos.

Metodología y Resultados.

Conclusiones.

Introducción

- Programación en el siglo XXI:
 - La Era del “Big Data”.
 - Necesaria la potencia de cálculo para el procesamiento y aprendizaje.
 - Mucha potencia de cálculo disponible desaprovechada.
 - Es necesario popularizar/facilitar el uso de los modelos paralelos/distribuidos.
 - Es necesario buscar la eficiencia (tiempo, energía...).
- Relevante en cualquier área, también en la IA., en los Algoritmos Evolutivos y la Programación Genética.



Introducción - Eficiencia

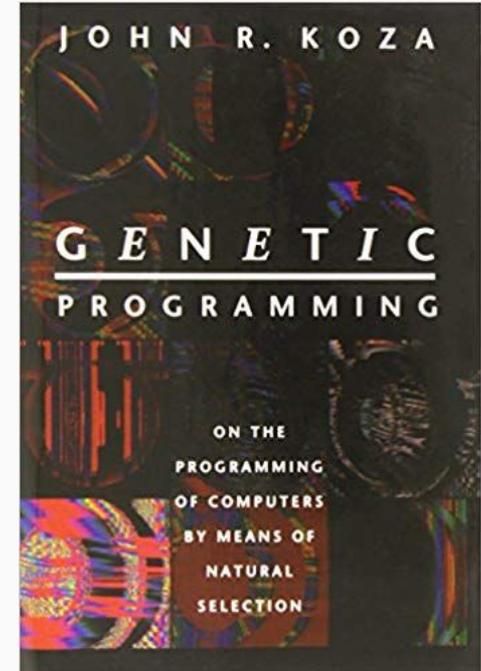
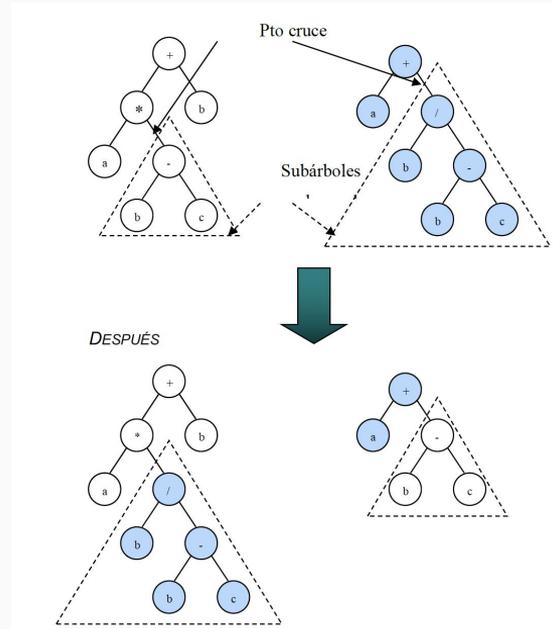
Definición: Dicho de “cosa”: eficaz para lograr el efecto que se desea.

Eficaz: Que produce el efecto propio o deseado.

No obstante, la RAE hace énfasis en los “**recursos**” necesarios para lograr un efecto.

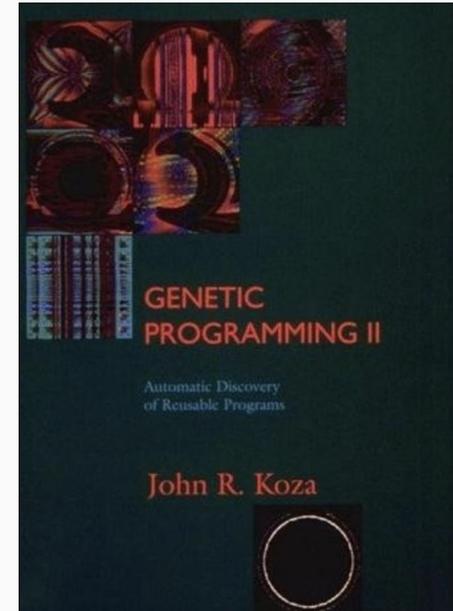
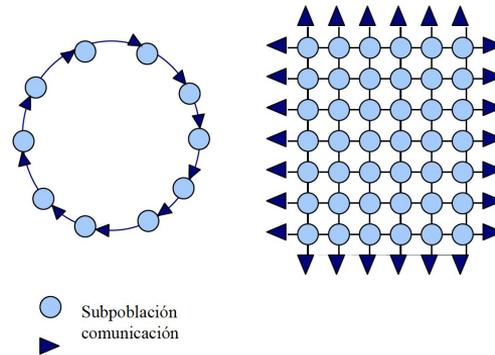
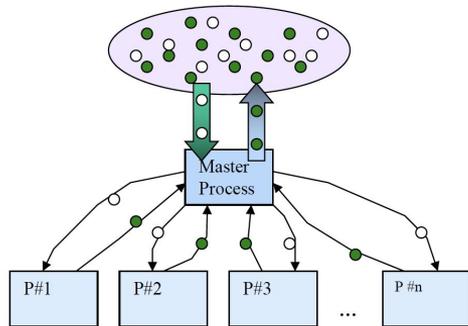
Introducción - PG Paralela Eficiente

Programación Genética (PG): Técnica para que los computadores aprendan a resolver problemas sin ser explícitamente programados. **“Program Induction, as described by J. Koza in Genetic Programming”**.



Introducción - PG Paralela Eficiente

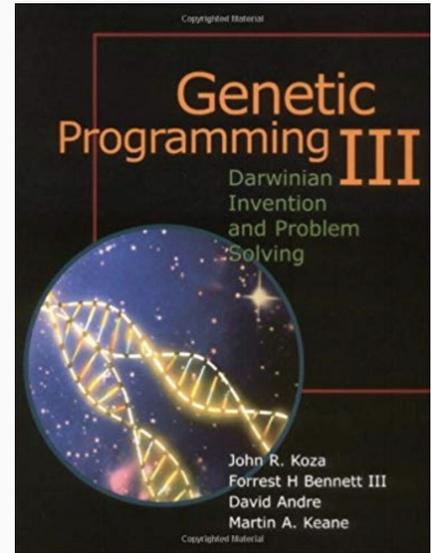
PG Paralela: Utiliza el paralelismo y los sistemas distribuidos para realizar la tarea más rápido.



Introducción - PG Paralela Eficiente

¿Es eficiente la PG? ¿Y la PG Paralela/Distribuida?

- ¿Consiguen resolver problemas sin ser explícitamente programados de forma eficiente?
- ¿Confían los investigadores en PG y (AES) paralelos y distribuidos?
- ¿Confían los “programadores” en los modelos paralelos y distribuidos?

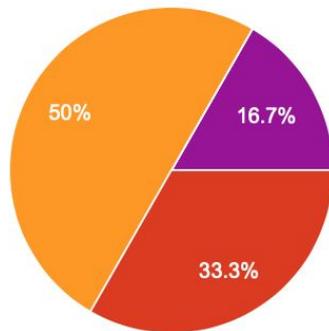
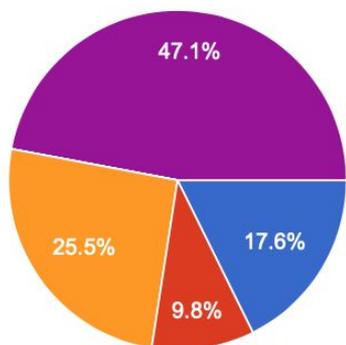


Introducción: P(G) Paralela Eficiente

¿Confían los programadores en la programación paralela y distribuida?

Introducción: P(G) Paralela Eficiente

Cuántos programadores utilizan la programación paralela/distribuida



- < 1 week
- < 1 month
- < 1 year
- Always use it.
- Never use it.

(Software factory in Extremadura - 4th year Computer Science students).

Introducción: P(G) Paralela Eficiente

¿Podemos mejorar estas cifras?

The screenshot shows the Taylor & Francis Online interface. At the top, there is a navigation bar with the Taylor & Francis logo and the text 'Taylor & Francis Online'. To the right of the logo are links for 'Log in' and 'Reg'. Below the navigation bar is a breadcrumb trail: 'Home > All Journals > Journal of Computer Information Systems > List of Issues > Latest Articles > Teaching Programming in the 21st Century'. The main content area features a search bar with the placeholder text 'Enter keywords, authors, DOI, ORCID etc'. Below the search bar, there is a section for the 'Journal of Computer Information Systems' with a 'Latest Articles' heading. Two buttons are visible: 'Submit an article' (green) and 'Journal homepage' (dark blue). The article 'Teaching Programming in the 21st Century' is highlighted, showing it is a 'Research Article' by 'Francisco Fernández de Vega' (with an email icon), published online on '18 Aug 2022'. On the left side of the article, there are statistics: '44 Views', '0 CrossRef citations to date', and '0' (likely downloads). At the bottom of the article, there are links for 'Download citation', the DOI 'https://doi.org/10.1080/08874417.2022.2108934', and a 'Check for updates' button.

Taylor & Francis Online

Log in | Reg

Home > All Journals > Journal of Computer Information Systems > List of Issues > Latest Articles > Teaching Programming in the 21st Century

Journal of Computer Information Systems > Latest Articles

Submit an article | Journal homepage

Enter keywords, authors, DOI, ORCID etc

44 Views

0 CrossRef citations to date

0

Research Article

Teaching Programming in the 21st Century

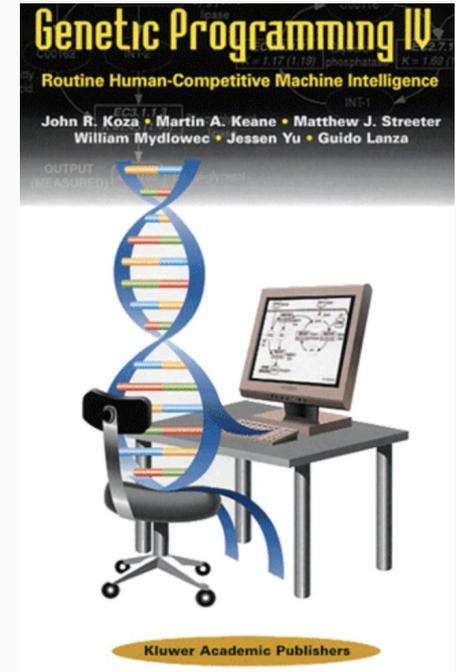
Francisco Fernández de Vega ✉

Published online: 18 Aug 2022

Download citation | <https://doi.org/10.1080/08874417.2022.2108934> | Check for updates

Introducción - PG Paralela Eficiente

- ¿Es eficiente en cuanto a tiempo de cómputo?
 - Problema (i): BLOAT.
 - Problema (ii): Aprovechamiento de recursos.
 - Problema (iii): De modo general se usa PG secuencial.
- ¿Es eficiente en cuanto a consumo energético?
 - Problema (i): Relación entre tiempo/energía no se ha estudiado previamente en profundidad.
 - Consumo energético importante en dispositivos operados con batería.
 - Hardware eficiente para PG.



Introducción: Sist. Paralelos y Distrib.

Interés en varios tipos de modelos/infraestructura, disponibles en el grupo GEA:

- Blade System (200 cores): Librerías MPI, modelo MAP/REDUCE (Hadoop, Spark)... Interés en reducción de tiempos y facilidad de uso de herramientas.
- GPUs: TeslaA100, Tesla K20, s1070; GeForce 1080 Ti.
- Sistemas de cómputo efímero: Modelos P2P/POOL Interés en eficiencia energética.
- Sistemas de Computación voluntaria.

Contenidos

Introducción.

Antecedentes.

Objetivos.

Metodología y Resultados.

Publicaciones.

Aplicaciones.

Contenidos

Introducción.

Antecedentes.

Objetivos.

Metodología y Resultados.

Publicaciones.

Aplicaciones.

¿Deberíamos estudiar la EE en los AEs?

El **Consumo Energético** es cada vez más importante

Crítico en computación móvil.

A veces el coste en consumo energético es superior al coste del hardware.

- Albers, S., Energy-efficient algorithms. Communications of the ACM, 53(5):8696, 2010.

¿Cual debería ser el **objetivo** en AEs?

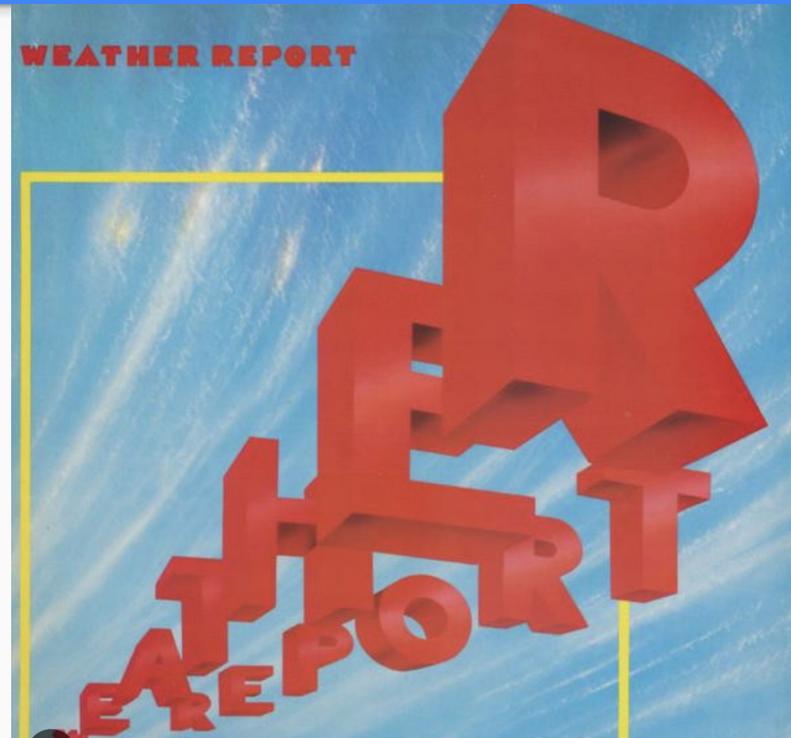
Calidad del Fitness y **Número de Generaciones** (o evaluaciones necesarias, o tiempo de cómputo) son el objetivo estándar..

Pocos intentos para estudiar el consumo energético.

¿**Tiempo** y consumo están relacionados?

Hablemos del tiempo

¿Podemos/conviene estudiar el tiempo desde algún otro punto de vista?



Hablemos del tiempo

Lo que me interesa:

- Programación Genética Paralela/Distribuida ¿Eficiente?:
 - **Prevención del bloat** en modelos paralelos/distribuidos de PG.
 - **Mejoras de consumo energético** de PG y AEs.
- ¿Algún “nuevo” punto de vista sobre “el tiempo”?

Hablemos del tiempo...y del consumo.

¿Es la Programación Genética (y los AEs en general) eficiente?

¿Por qué no se ha estudiado esta temática en AEs?

Quizá **TIEMPO** y **CONSUMO** se ven como dos caras de una misma moneda:

- Si el consumo instantáneo no cambia, tiempo y consumo son proporcionales.
- Si el consumo instantáneo no depende de las operaciones que se ejecutan, la configuración de los algoritmos no influye en el consumo.

Hablemos del tiempo...y del consumo.

En otras áreas, los algoritmos SÍ han sido estudiados desde el punto de vista del consumo energético:

- Albers, S., Energy-efficient algorithms. Communications of the ACM, 53(5):8696, 2010.
- Heinzelman, W. R., Chandrakasan, A., & Balakrishnan, H. (2000, January). Energy-efficient communication protocol for wireless microsensor networks. In System sciences, 2000. Proceedings of the 33rd annual Hawaii international conference on (pp. 10-pp). IEEE.

¿Cual es el hardware que mejora la eficiencia energética?

¿Podemos configurar el algoritmo para reducir el consumo energético?

Hablemos del tiempo...y del consumo.

Bloat: crecimiento del tamaño de los árboles NO ASOCIADO a incremento en calidad de las soluciones.

Interés en desarrollar una perspectiva novedosa del problema.



Metodología: PG Eficiente en paralelo/distrib.

Si tiempo es equivalente a tamaño, ¿Podríamos utilizar el **tiempo** como medida del **bloat**?

- **Idea clave:** La propia ejecución de los Programas implícitamente mide el tiempo de ejecución.
- En arquitecturas paralelas en las que cada núcleo evalúa un individuo, el orden de llegada de los individuos está relacionado con su tiempo de ejecución.

Contenidos

Introducción.

Antecedentes.

Objetivos.

Metodología y Resultados.

Publicaciones.

Aplicaciones.

Contenidos

Introducción.

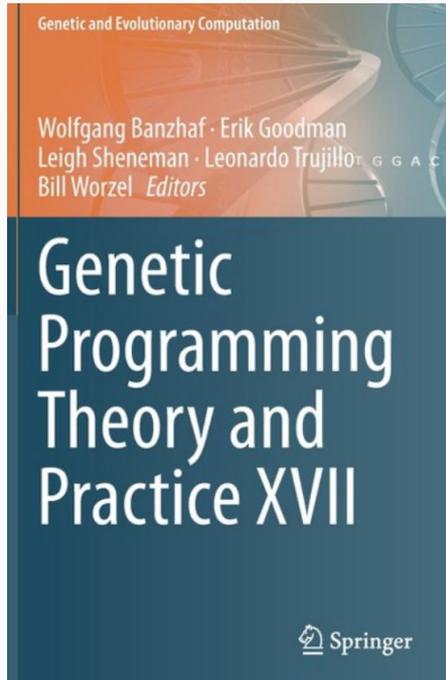
Antecedentes.

Objetivos.

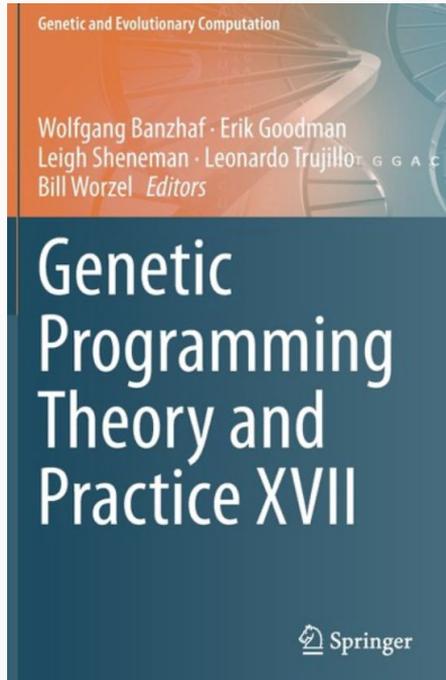
Metodología y Resultados.

Conclusiones.

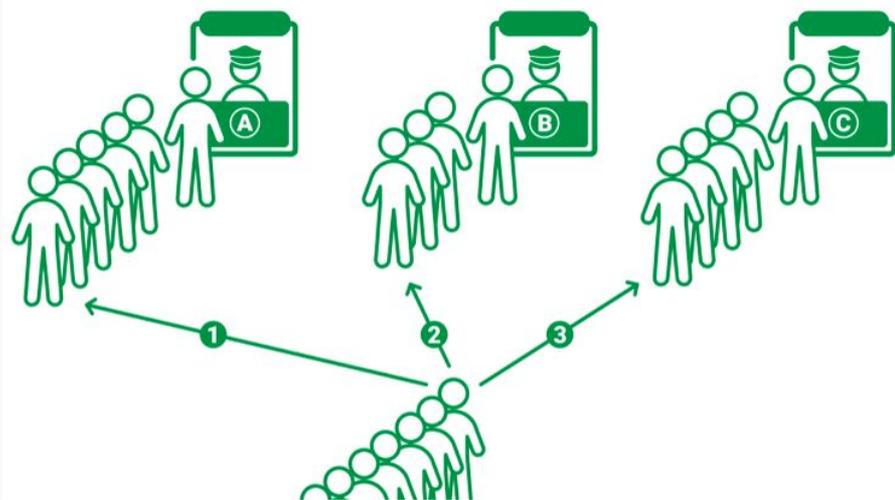
Metodología: PG Eficiente en paralelo/distrib.

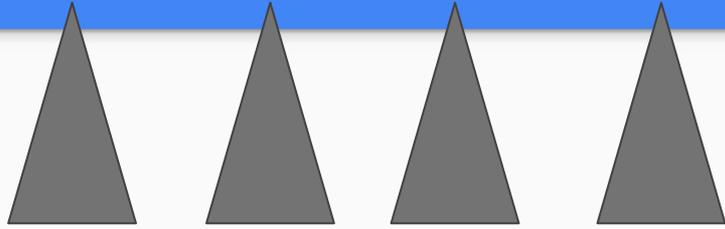


Metodología: PG Eficiente en paralelo/distrib.

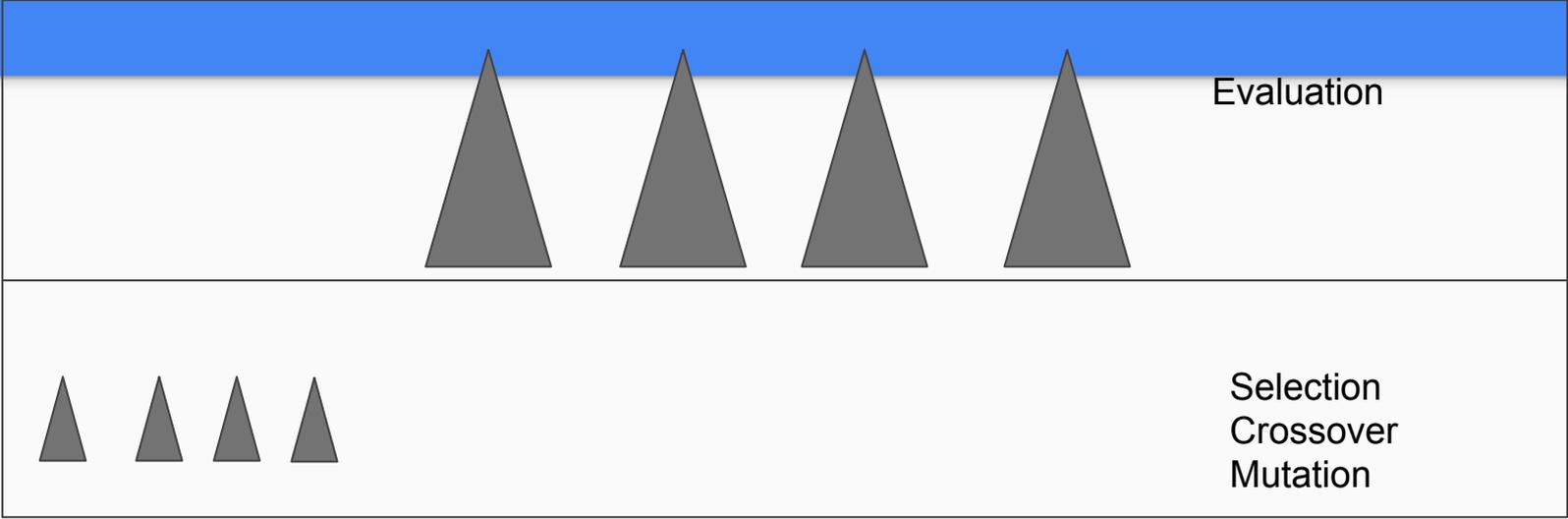


Metodología: PG Eficiente en paralelo/distrib.





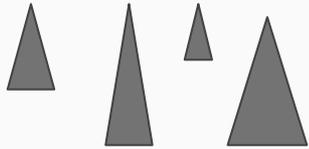
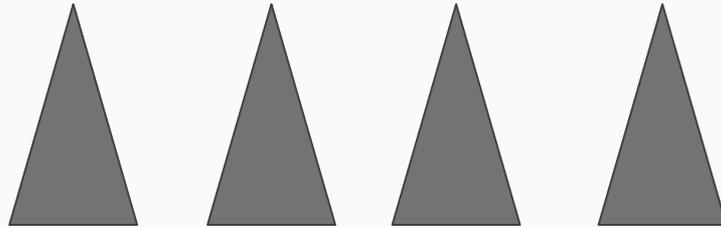
Evaluation



Evaluation

Selection
Crossover
Mutation

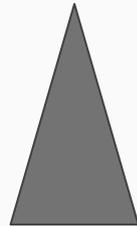
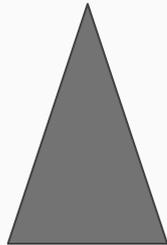
New Individuals



Evaluation

Selection
Crossover
Mutation

New Individuals



Metodología (y resultados): Eficiencia en paralelo/distrib.

Probemos la idea en benchmarks: Ant, Multiplexer, Parity, Lawnmover, Regression.

Herramienta: **ECJ multihilo**, adaptada con mecanismo específico de **load-balancing**:

- Paquetes de individuos con tiempo de ejecución similar.

30 ejecuciones independientes. Test estadísticos aplicados para garantizar las conclusiones alcanzadas.

Número de grupos, nuevo parámetro del modelo.

Podemos incluso aplicar la idea en entornos secuenciales: individuos que sólo se reproducen en “grupos” de tiempo similar de ejecución.

Objetivo: Eficiencia en paralelo/distrib.

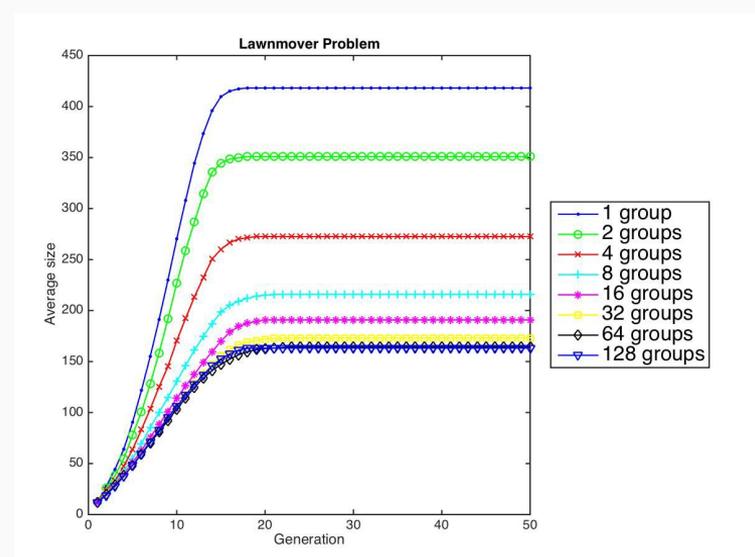
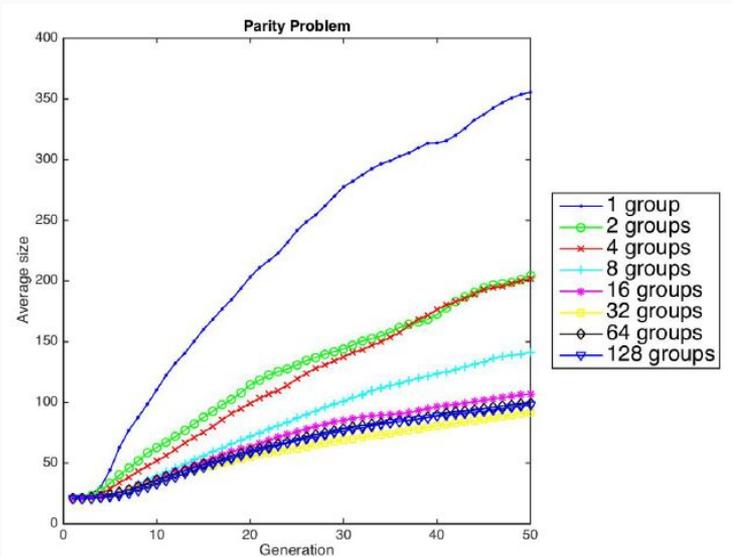
Pruebas: Intel XEON 8 cores, 2.4 GHz.

Paquetes de individuos de tiempo similar enviados al mismo hilo de ejecución.
2, 4, 8, 16, 32, 64, 128 grupos-hilos.

50 Generaciones.

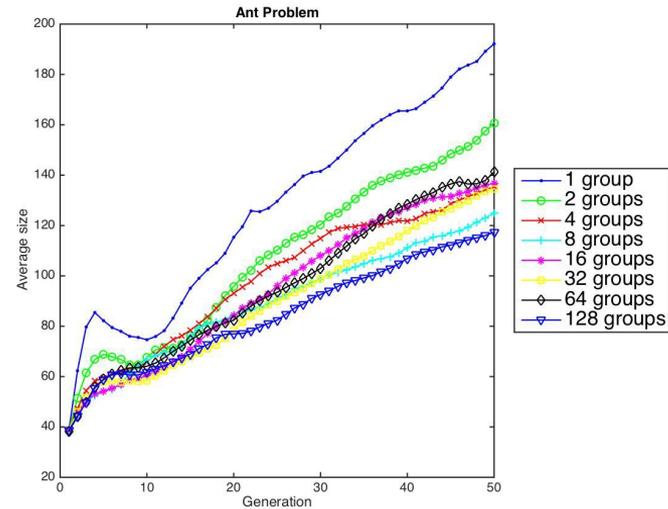
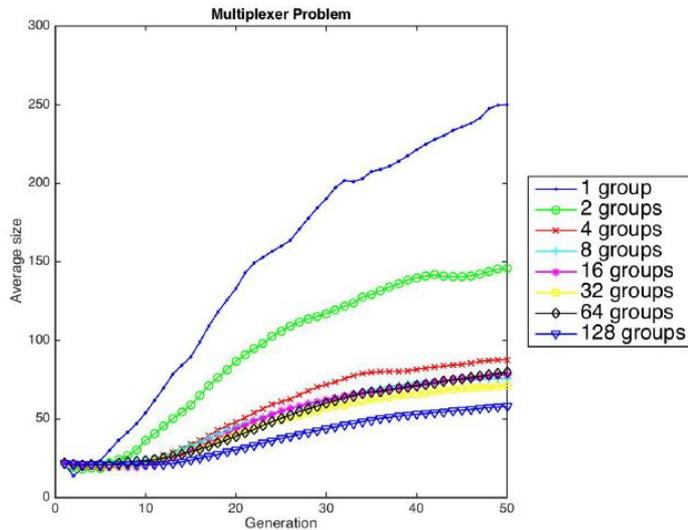
(Metodología y) Resultados: Eficiencia en paralelo/distrib.

Evolución del tamaño de los individuos cuando agrupamos por tiempo de ejecución.



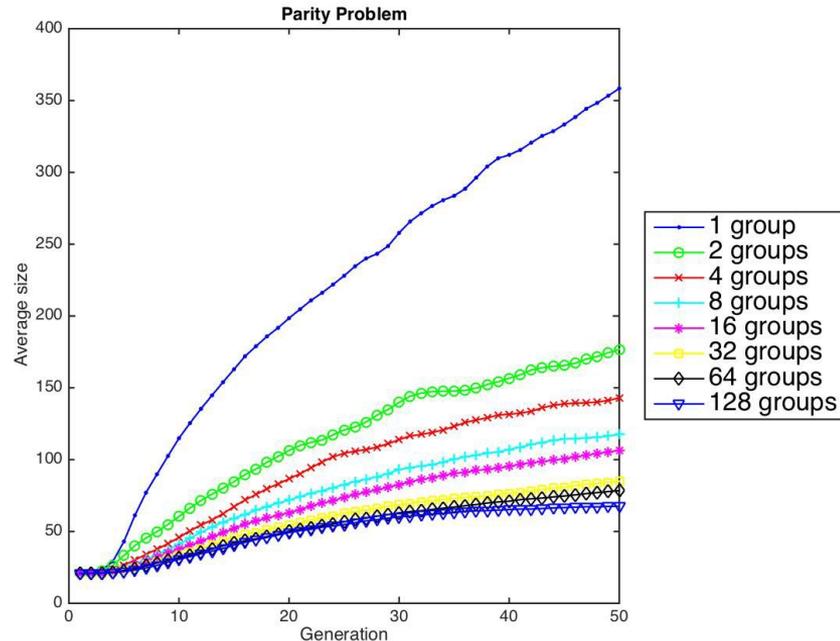
(Metodología y) Resultados: Eficiencia en paralelo/distrib.

Evolución del tamaño de los individuos cuando agrupamos por tiempo de ejecución.



(Metodología y) Resultados: Eficiencia en paralelo/distrib.

En modelo estándar/secuencial también funciona.



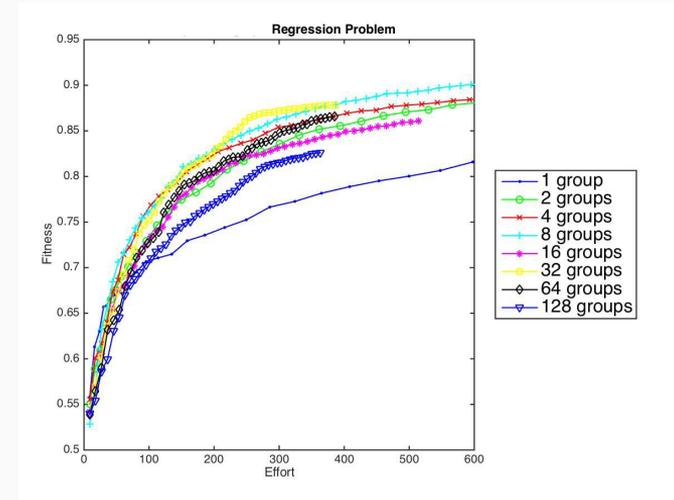
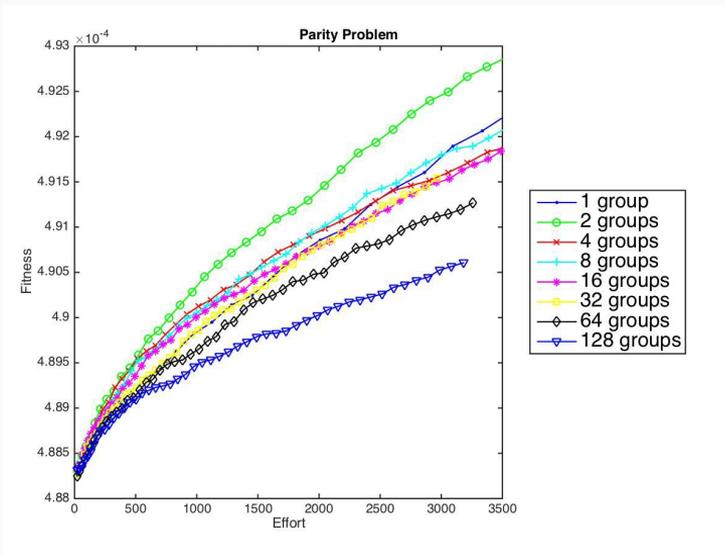
(Metodología y) Resultados: Eficiencia en paralelo/distrib.

¿Y la calidad del fitness?

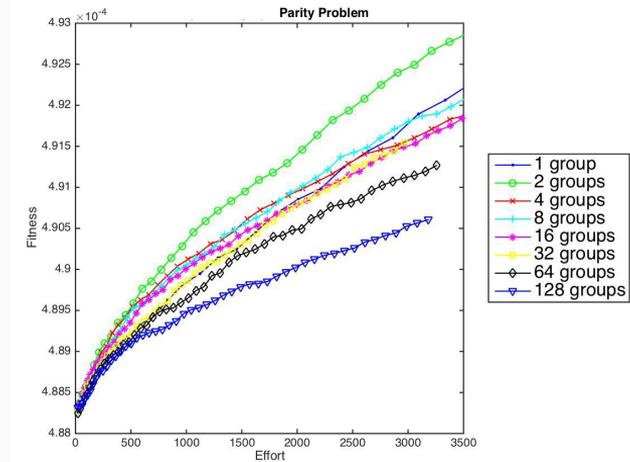
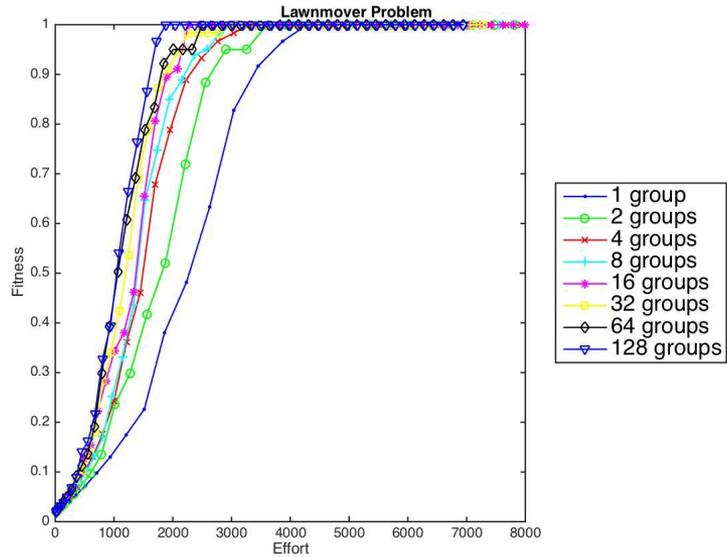
- Estudiemos cómo mejora la calidad a medida que aplicamos esfuerzo de cómputo (número total de nodos evaluados).

(Metodología y) Resultados: Eficiencia en paralelo/distrib.

¿Y la calidad del fitness?



Resultados: Eficiencia en paralelo/distrib.



Resultados: Eficiencia en paralelo/distrib.

Time and individual duration in genetic programming

FF de Vega, G Olague, D Lanza, W Banzhaf, E Goodman, ...
IEEE Access 8, 38692-38713

It is time for new perspectives on how to fight bloat in GP

FF de Vega, G Olague, F Chávez, D Lanza, W Banzhaf, E Goodman
Genetic Programming Theory and Practice XVII, 25-38

Hablemos de Eficiencia Energética

Diferentes niveles en los que podemos hacer el estudio:

- Modelos/Algoritmos/Parametrización..
- Implementaciones/Lenguajes/Compiladores..
- Hardware.

Hablemos de Eficiencia Energética

Diferentes niveles en los que podemos hacer el estudio:

- Modelos/Algoritmos/**Parametrización**..
- Implementaciones/**Lenguajes**/Compiladores..
- **Hardware**.

Hardware, AEs y eficiencia energética.

En 2016, planteamos por primera vez en el área de los AEs el interés de esta temática:

- ¿Qué plataforma hardware conviene para llegar a la solución con el menor gasto energético posible en GP?

Problema testado: Multiplexer-6 (GP).

Software utilizado: lilgp

30 ejecuciones por experimento (mismas semillas utilizadas en todas las plataformas).

Tablet Android, Macbook, Laptop Linux, raspberri Pi y Blade system.

Power tutor App, Hardware Monitor, Multímetro, VMware hypervisor.

Hardware, AEs y eficiencia energética.

+ Time



- Energy



- time



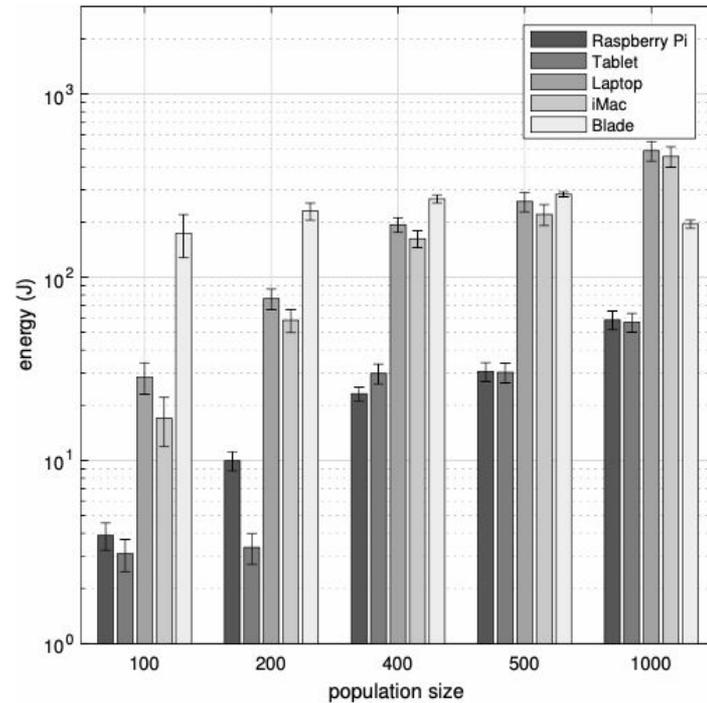
+ energy

Hardware, AEs y eficiencia energética.

System	population size				
	100	200	400	500	1000
raspberry pi	7.77 ± 1.31	19.91 ± 2.41	46.22 ± 4.01	61.10 ± 7.19	116.80 ± 13.55
laptop	1.73 ± 0.31	4.43 ± 0.54	10.60 ± 0.97	13.89 ± 1.68	27.13 ± 3.36
iMac	1.38 ± 0.28	3.69 ± 0.48	8.98 ± 0.84	11.74 ± 1.44	22.95 ± 2.85
tablet	4.43 ± 0.75	4.85 ± 0.78	35.68 ± 4.15	36.17 ± 4.17	68.70 ± 7.89
blade	2.59 ± 0.53	6.88 ± 0.91	16.78 ± 1.59	22.28 ± 2.77	43.53 ± 5.48

Análisis del Tiempo de ejecución (s)

Hardware, AEs y eficiencia energética.



Hardware, AEs y eficiencia energética.

Fernández de Vega, F., Chávez, F., Díaz, J., García, J. A., Castillo, P. A., Merelo, J. J., & Cotta, C. (2016, September). **A cross-platform assessment of energy consumption in evolutionary algorithms.** In International Conference on Parallel Problem Solving from Nature (pp. 548-557). Springer, Cham.

+ Time



- Energy



- time



+ energy

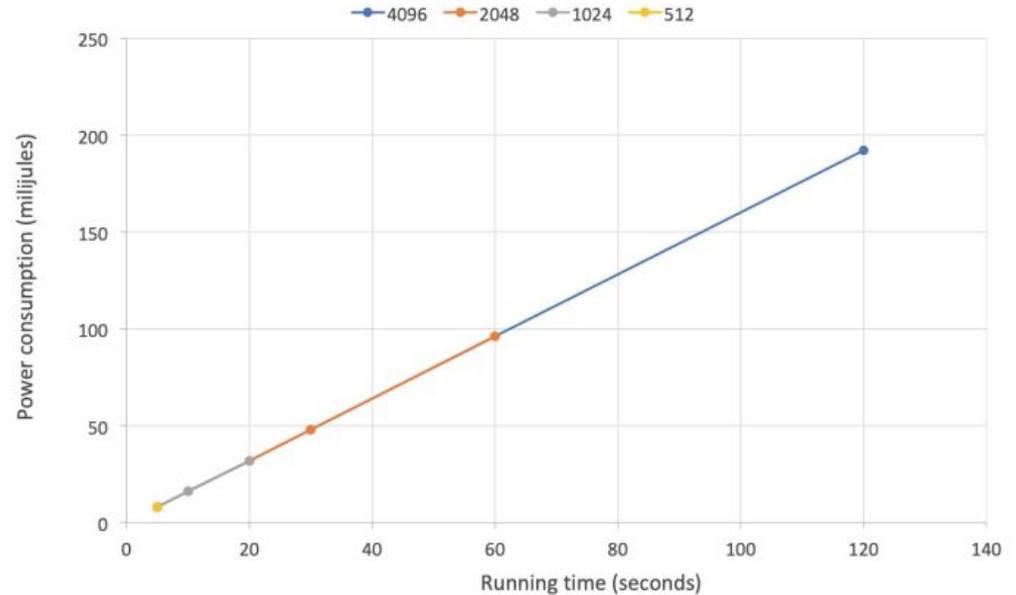
Algoritmos/parametrización y Eficiencia Energética en AEs

- Hasta el año 2016 no había trabajos que hicieran este análisis.
- ¿Cual es la razón?

Algoritmos/parametrización y Eficiencia Energética en AEs

Si tiempo y energía son proporcionales, y los parámetros no influyen en el consumo:

- A mayor tiempo, mayor consumo.



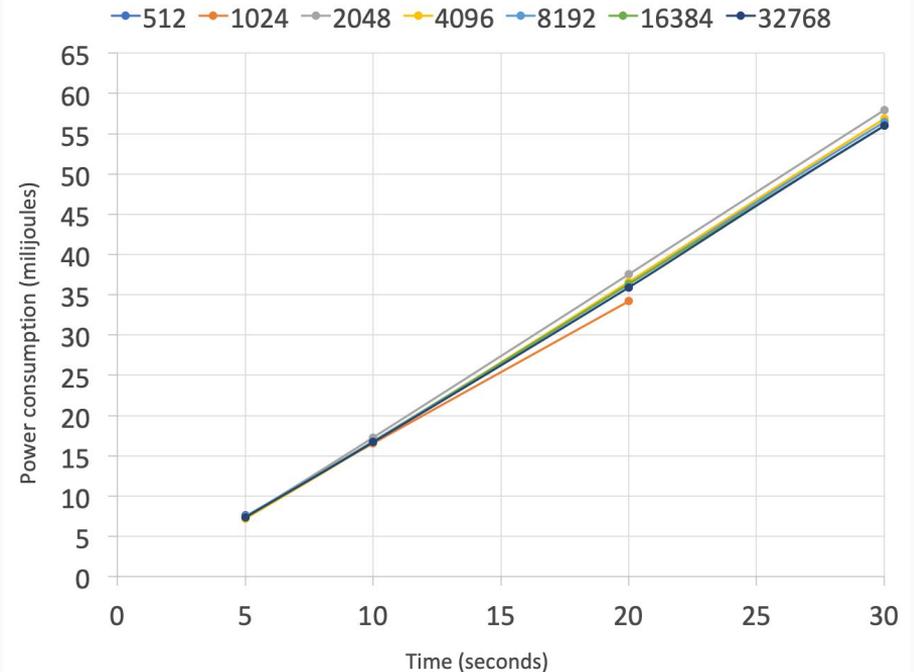
Algoritmos/parametrización y Eficiencia Energética en AEs

- Conviene poner en duda las ideas asumidas.

Algoritmos/parametrización y Eficiencia Energética en AEs

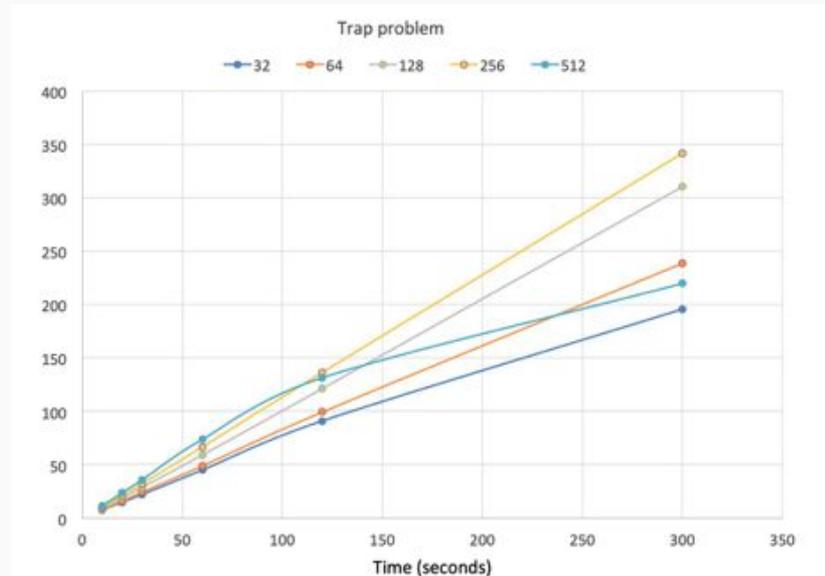
Diferentes tamaños de cromosomas en el problema **maxone**.

Tamaño población = 10

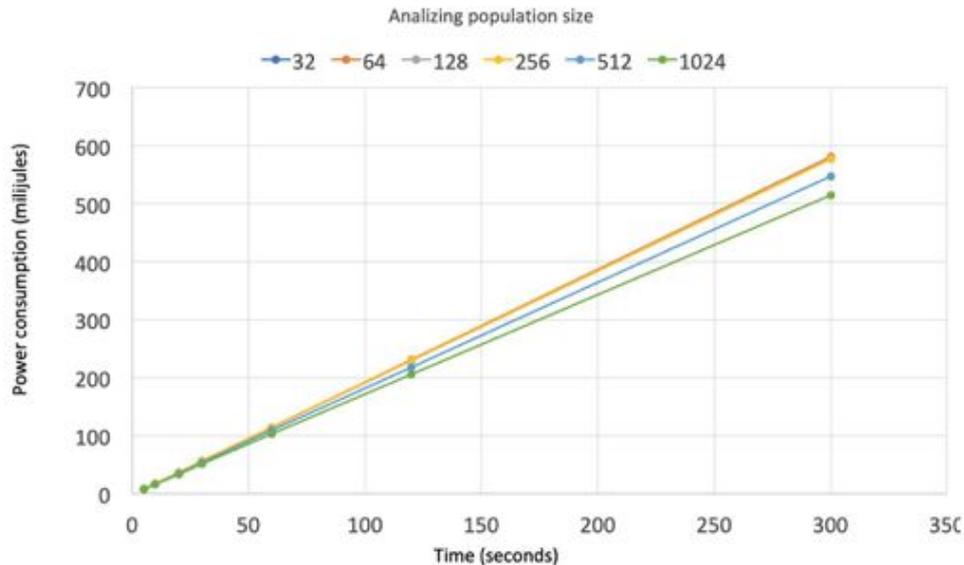


Algoritmos/parametrización y Eficiencia Energética en AEs

Trap problem. Diferentes tamaños de población y cromosoma de tamaño fijo = 200.



Algoritmos/parametrización y Eficiencia Energética en AEs



maxone con diferentes tamaños de población.

Tamaño cromosoma = 16384

¿Y en GP?

 **SAGE** journals

Search this journal ▾

Enter search terms...



[Advanced search](#)

Browse by discipline ▾

Information for ▾

Measurement and Control



Impact Factor: **1.648** / 5-Year Impact Factor: **1.628**



Open access



Research article

First published online April 28, 2022

Population size influence on the energy consumption of genetic programming

[Josefa Díaz-Álvarez](#)  , [Pedro A Castillo](#), [...], and [Jorge Alvarado](#)  [View all authors and affiliations](#)

[Volume 55, Issue 1-2](#) | <https://doi.org/10.1177/00202940211064471>

Unificando medidas

Using a power meter **YOKOGAWA WT310E** (and also APPowermeter-RAPL) to measure power consumption on:

- Raspberry Pi 3 B+. Quad Core 1.2 GHz. Raspbian.
- ASUS laptop. I5 2450-M, 2.5 GHz. Ubuntu.
- Desktop i5 4430-M 3.5 GHz.



Benchmarks and Experiments

GP benchmarks: (extracted from “Genetic Programming Needs Better Benchmarks”, McDermott et al, 14th, Gecco conference, 2012).

- LID.
- ORDER TREE.

Population sizes tested: 256, 512, 1024, 2048, 4096, 8192.

30 independent runs per experiment.

300s max time allowed.

ECJ to run experiments. Total time per generation added to logs.

Experiments

Three different settings:

- Running experiments for 300s.
- Running experiments until solution found.
- Further experiments for in depth analysis.

Average best fitness values computed in both generations before and after 300 seconds.

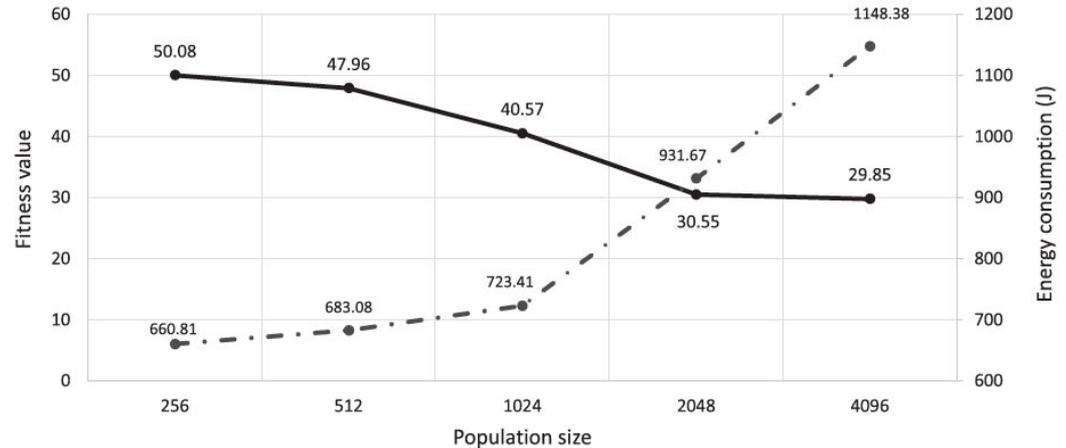
Average energy consumption computed both at exactly 300 seconds and when completing the last generation after 300 seconds.

Results: Last generation after 300 seconds

LID problem

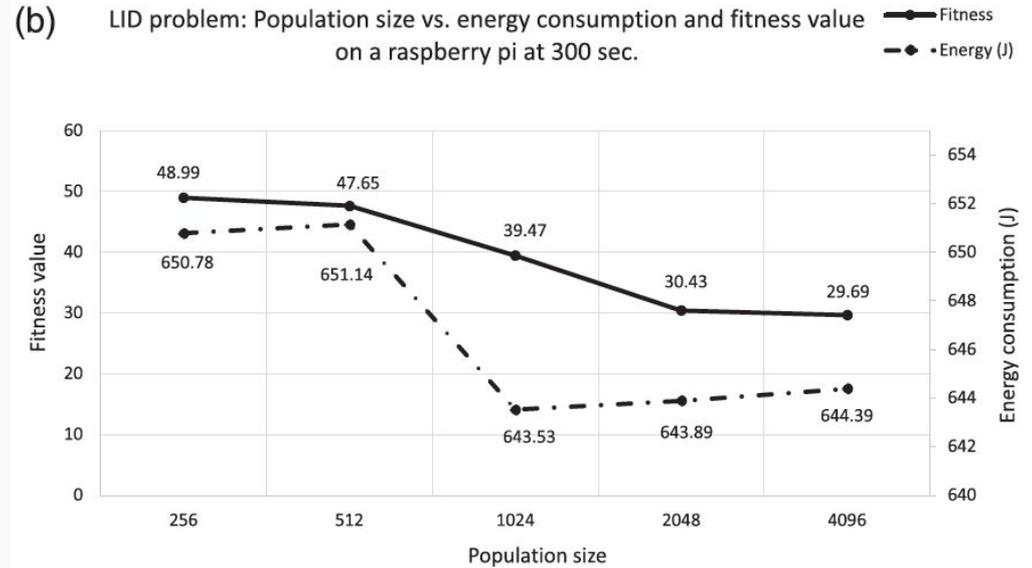


(a) LID problem: Population size vs. energy consumption and fitness value on a raspberry pi until 300 sec. are exceeded



Results: At exactly 300 seconds

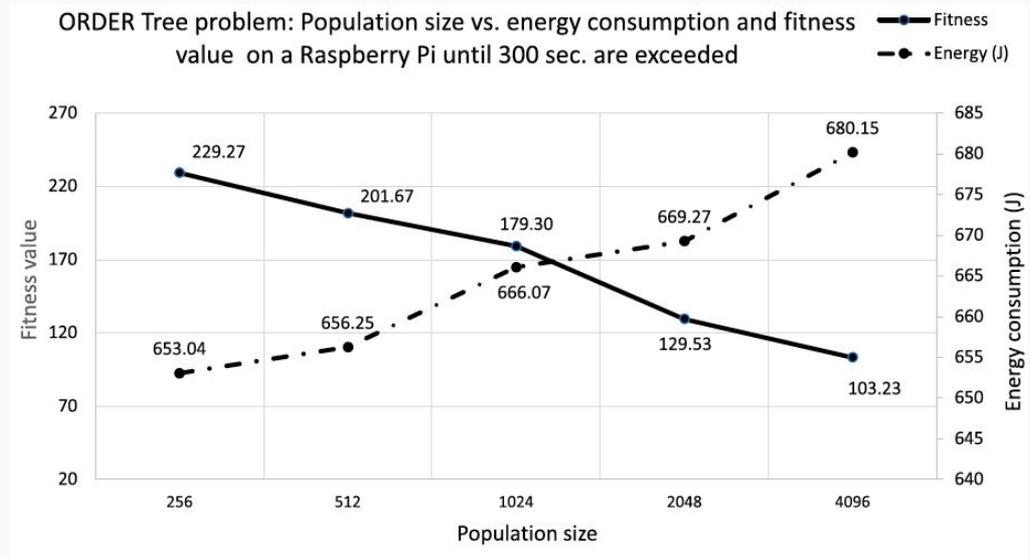
LID problem



Results: Last generation after 300 seconds

ORDER TREE problem

Raspberry Pi



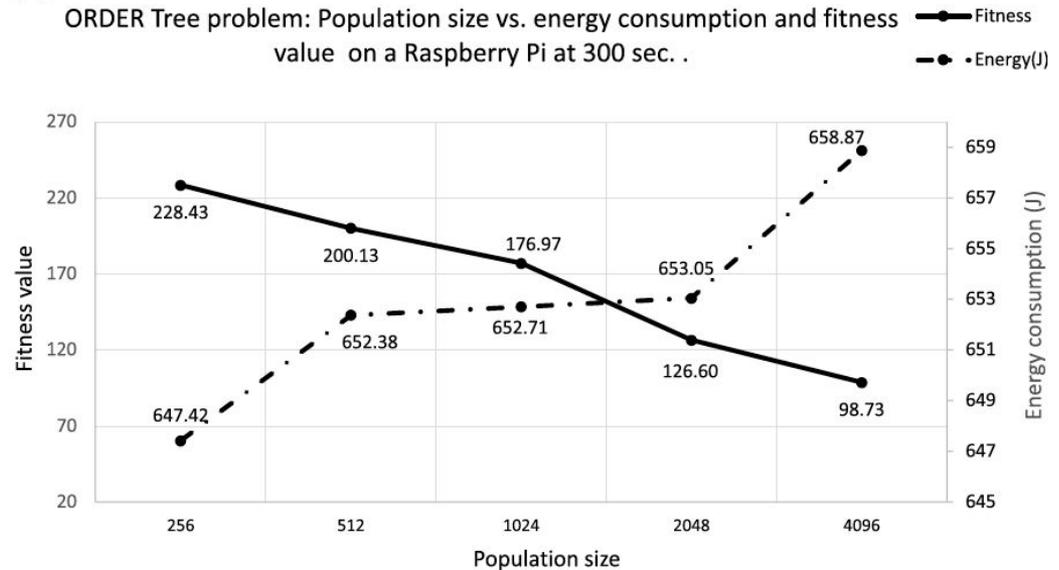
Results: At exactly 300 seconds

ORDER TREE problem



(b)

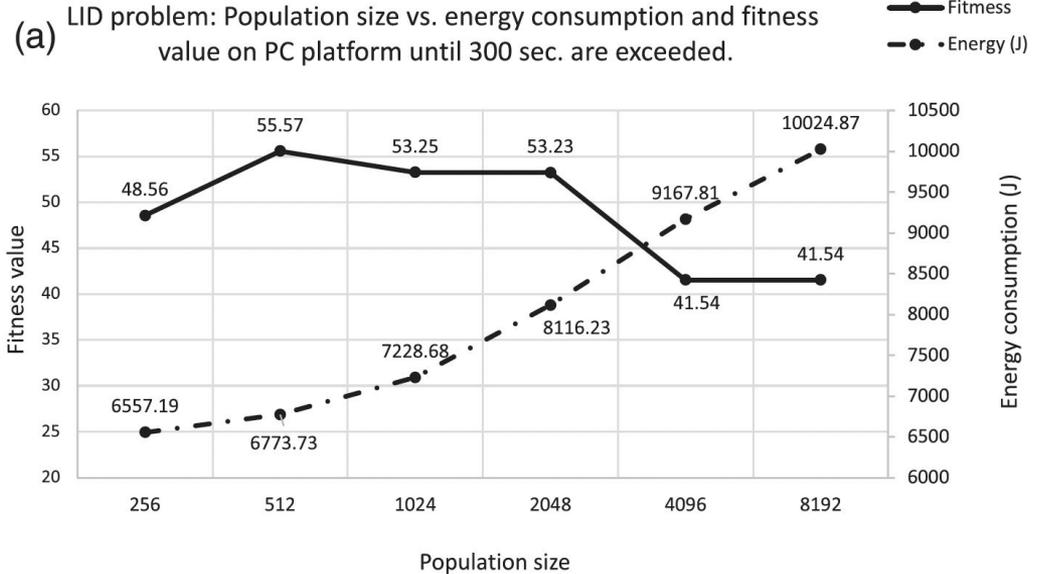
ORDER Tree problem: Population size vs. energy consumption and fitness value on a Raspberry Pi at 300 sec. .



Results: Last generation after 300 seconds

LID problem

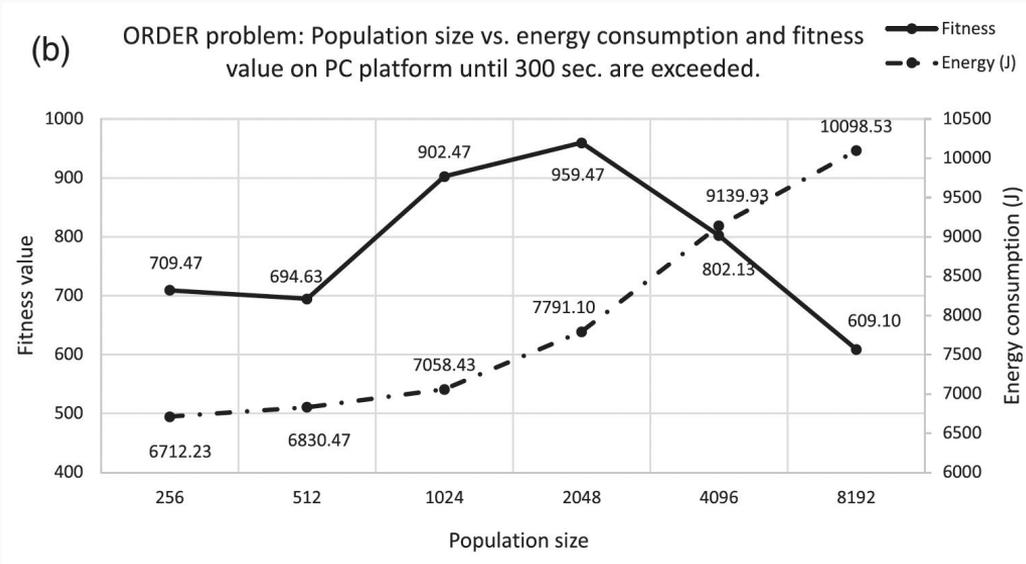
Desktop computer



Results: Last generation after 300 seconds

ORDER TREE problem

Desktop computer



Comparing hardware platforms

Evolution in fitness value for experiments with time limit 300 seconds.

	Raspberry Pi				Laptop				PC			
Pop. Size	Lid		ORDER		Lid		ORDER		Lid		ORDER	
	Fv	Ge	Fv	Ge	Fv	Ge	Fv	Ge	Fv	Ge	Fv	Ge
256	50.08	52.67	229.27	124.50	58.33	8708.53	579.07	4883.93	48.56	7499.70	709.47	4953.80
512	47.96	30.70	201.67	81.27	55.57	4041.83	793.73	2137.43	55.57	4861.40	694.63	2519.60
1024	40.57	18.33	179.30	53.13	53.23	1794.37	909.40	912.60	53.25	2195.33	902.47	1339.83
2048	30.55	13.00	129.53	35.57	55.57	936.27	756.73	442.23	53.23	1076.53	959.47	701.50
4096	29.85	10.80	103.23	24.30	41.59	230.63	627.47	276.70	41.54	312.77	802.13	387.10
8192	-	-	-	-	36.87	65.90	493.63	155.73	41.54	151.30	609.10	211.30

Últimos experimentos: GA & Rosenbrock

Es suficiente con fijarnos en el consumo instantáneo medio para poder tener una idea de lo que va a suceder.

Realizamos 30 ejecuciones de cada experimento con un número fijo de generaciones, y comparamos resultados de diferente tamaño de población.

Últimos experimentos: GA & Rosenbrock

100 GENERACIONES		
Tam. Población	Julios	Segundos
64	3,577	0,31
128	3,697	0,9
256	3,864	3,73
512	3,764	15,35
1024	3,808	60,71
2048	3,832	268,88

Últimos experimentos: GA & Rosenbrock

150 GENERACIONES		
Tam. Población	Julios	Segundos
64	3,554	0,46
128	3,652	1,35
256	3,687	5,53
512	3,722	22,12
1024	3,777	90
2048	3,871	408,53

Contenidos

Introducción.

Antecedentes.

Objetivos.

Metodología y Resultados.

Conclusiones.

Contenidos

Introducción.

Antecedentes.

Objetivos.

Metodología y Resultados.

Conclusiones.

Conclusions

Tiempo y energía no son directamente proporcionales.

Se han detectado anomalías que relacionan tamaños de población y cromosoma con el consumo energético.

En ocasiones, valores mayores de tamaños provocan consumos menores, dependiendo del hardware utilizado.

Conclusiones

Hardware, modelos, lenguajes, parámetros.... todos pueden tener influencia en el consumo y eficiencia energética de los algoritmos.

Deberíamos incluir el consumo como un nuevo “objetivo” en los EAs.

Trabajo futuro

Estamos comparando:

- AEs en java, python y C.
- Diferentes paquetes que incluyen AEs.

Thank you.

fcofdez@unex.es