

0

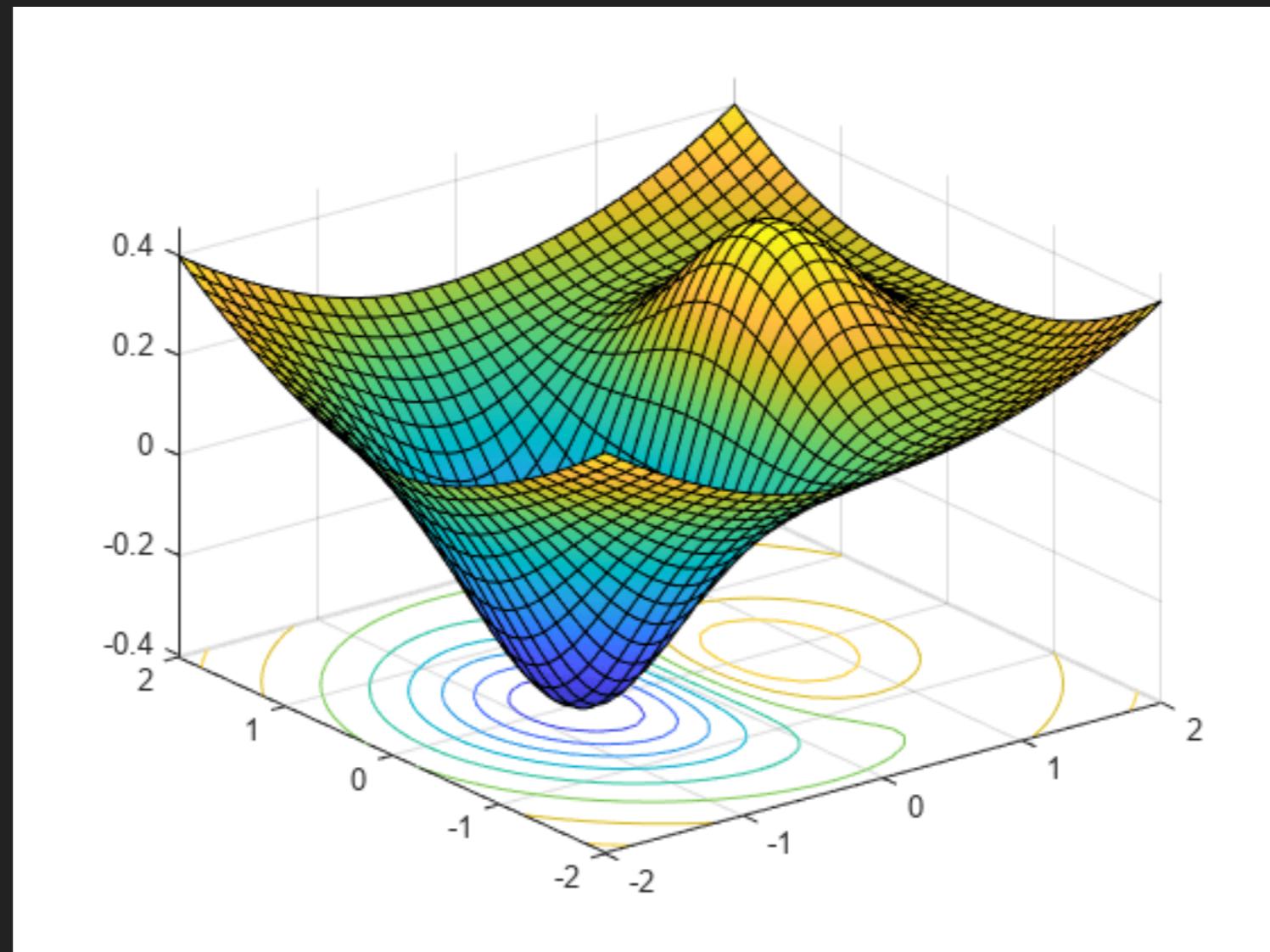
ALGORITMOS EVOLUTIVOS I (CURSADO)

DOCENTE DEL CURSO

- ▶ Miguel Augusto Azar
- ▶ Ingeniero en Informática
- ▶ Especialista en Docencia Superior
- ▶ Docente e investigador

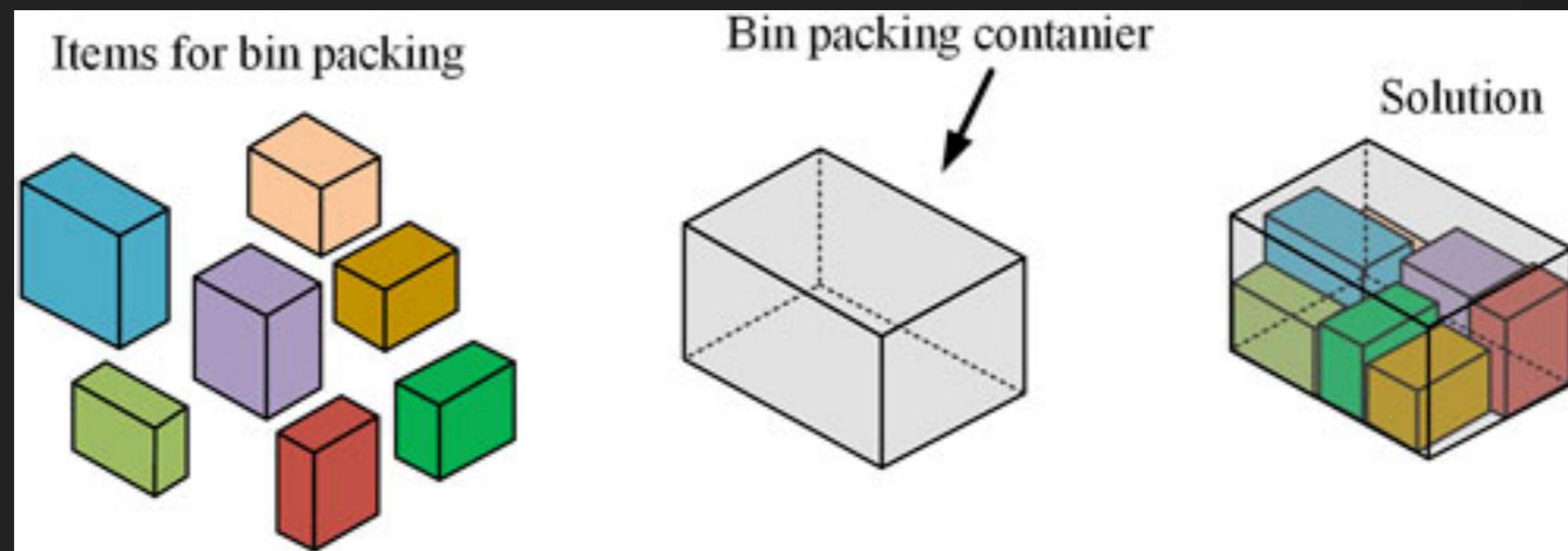


¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?



¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

- ▶ Artículos a embalar en un contenedor

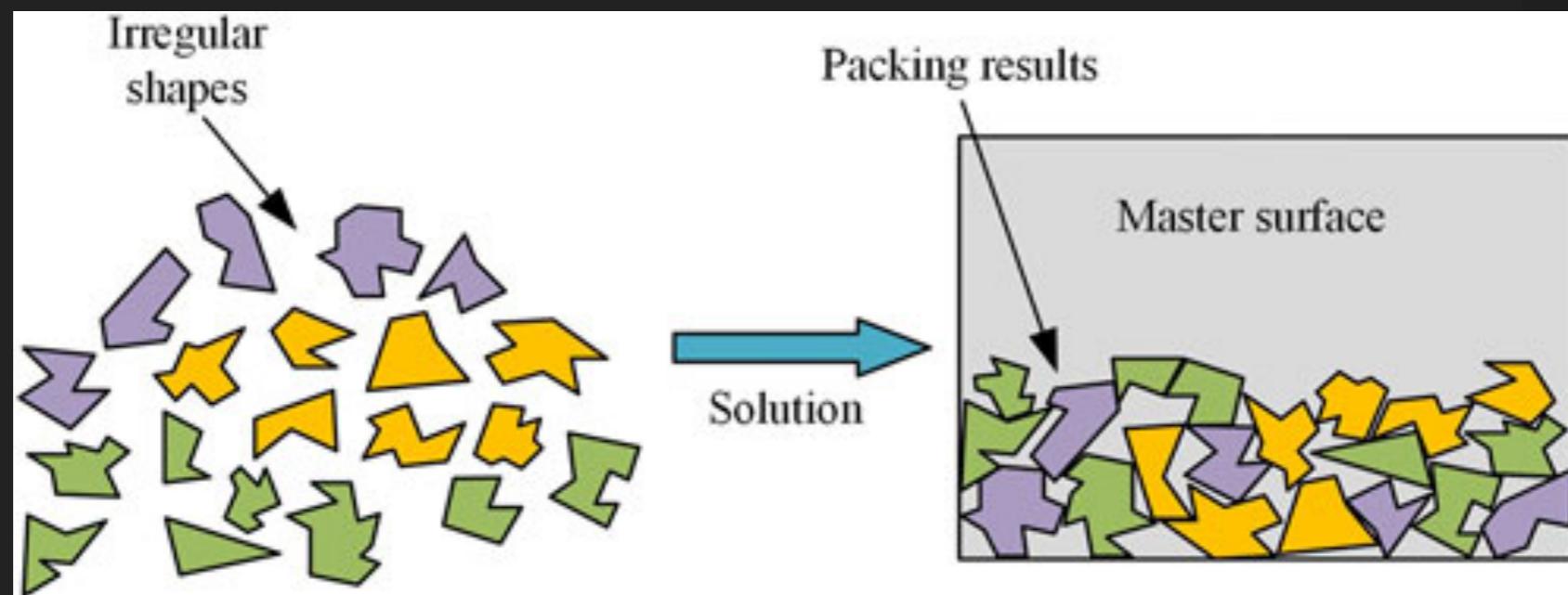


- ▶ 3D bin packing problem (set packing)



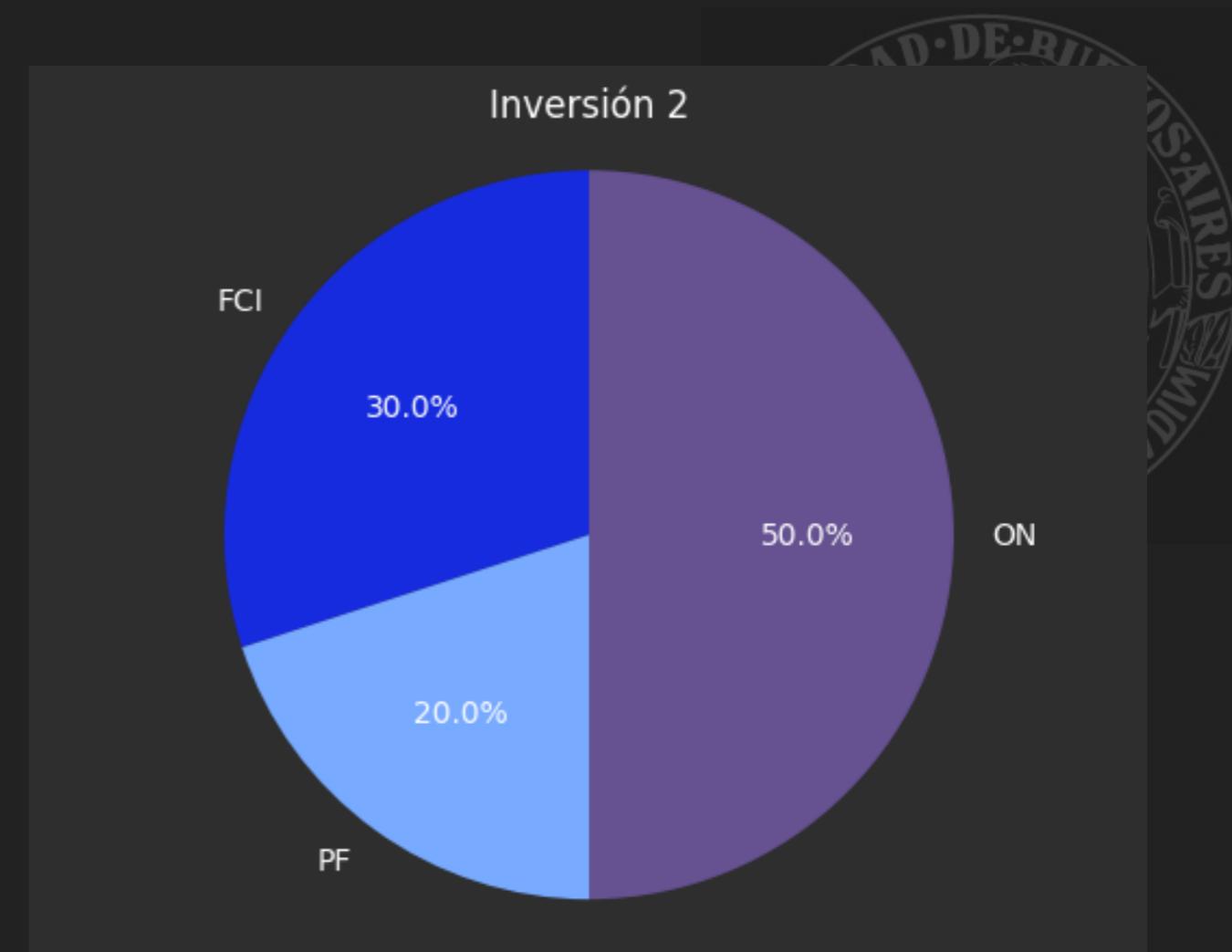
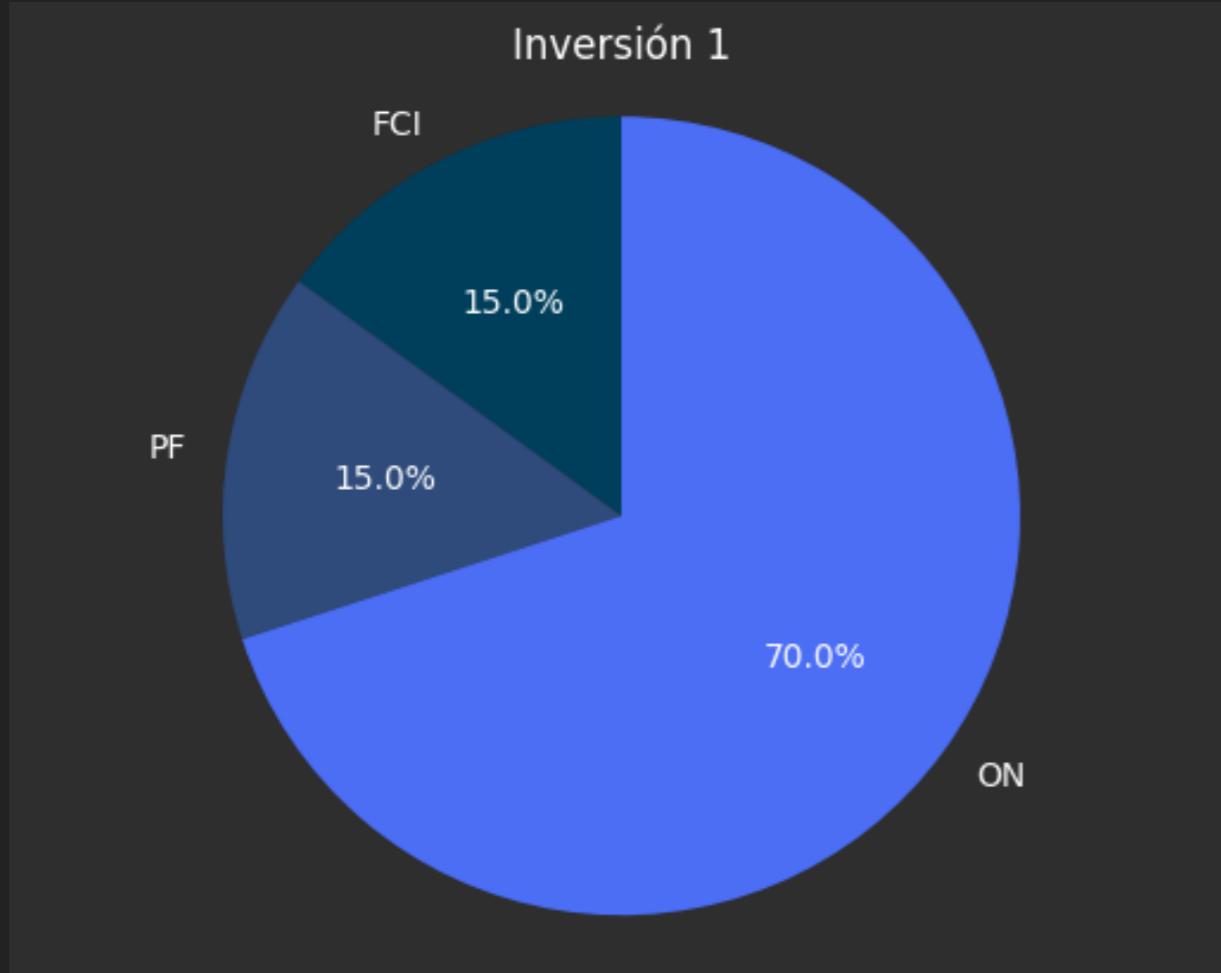
¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

- ▶ Tela en una fábrica de prendas



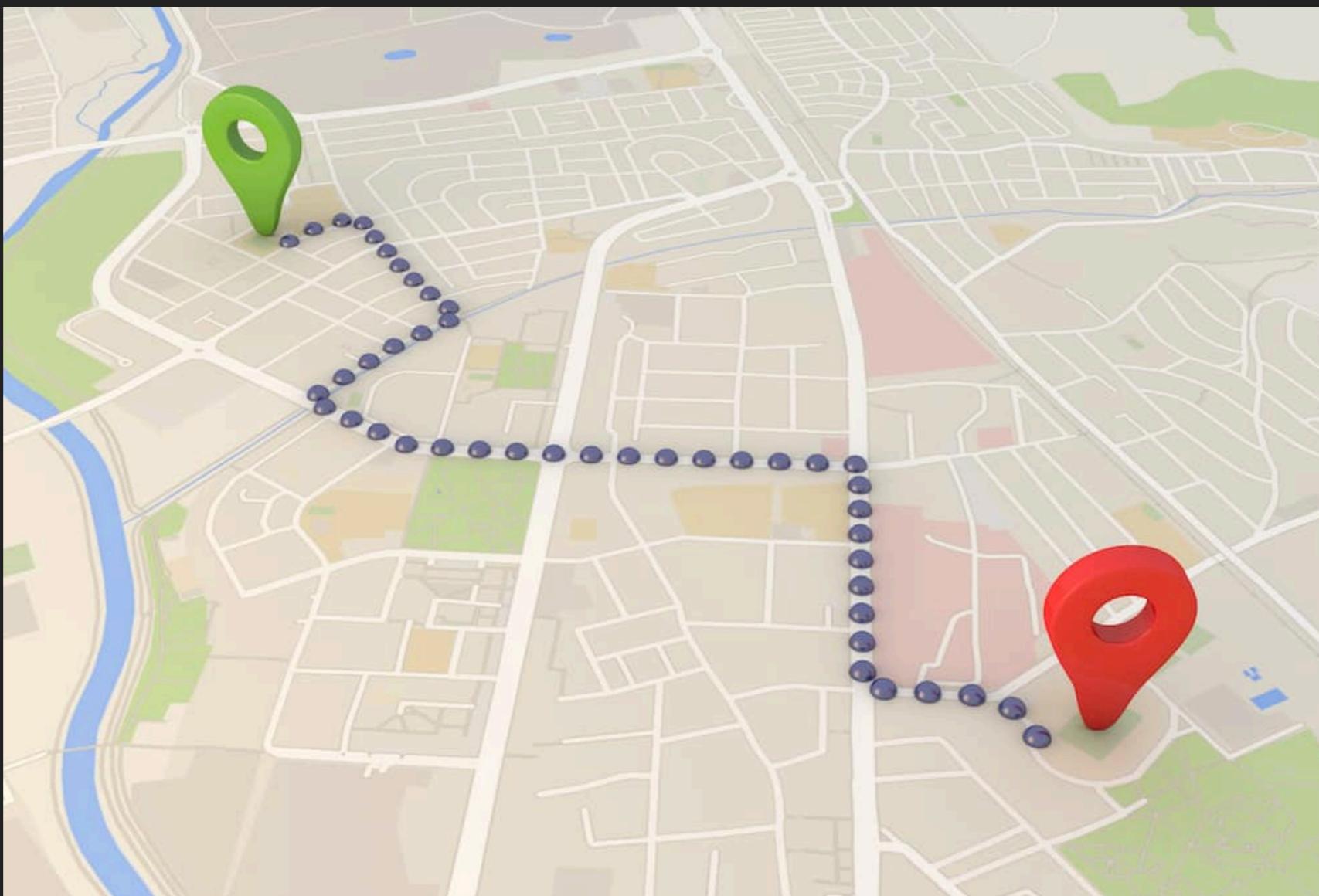
¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

- ▶ Optimización de un portfolio de inversión



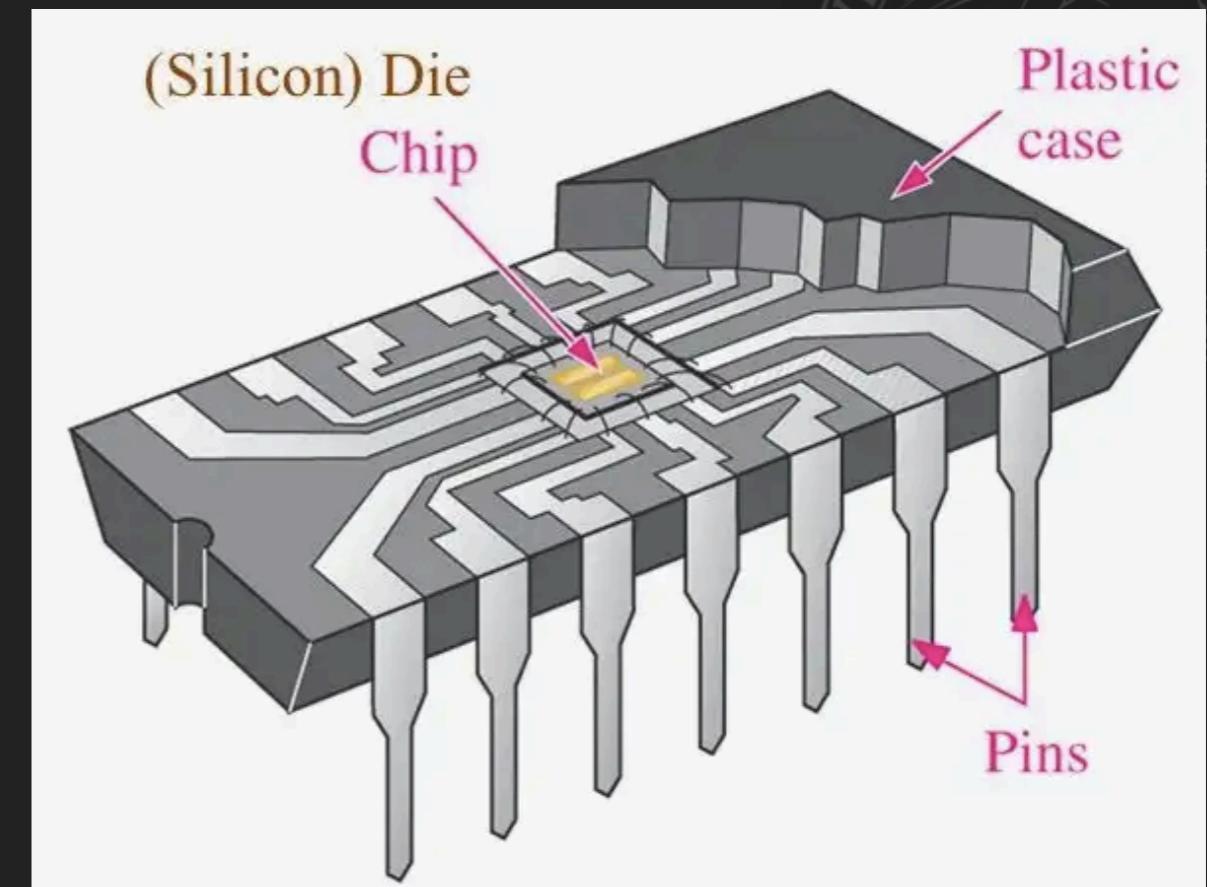
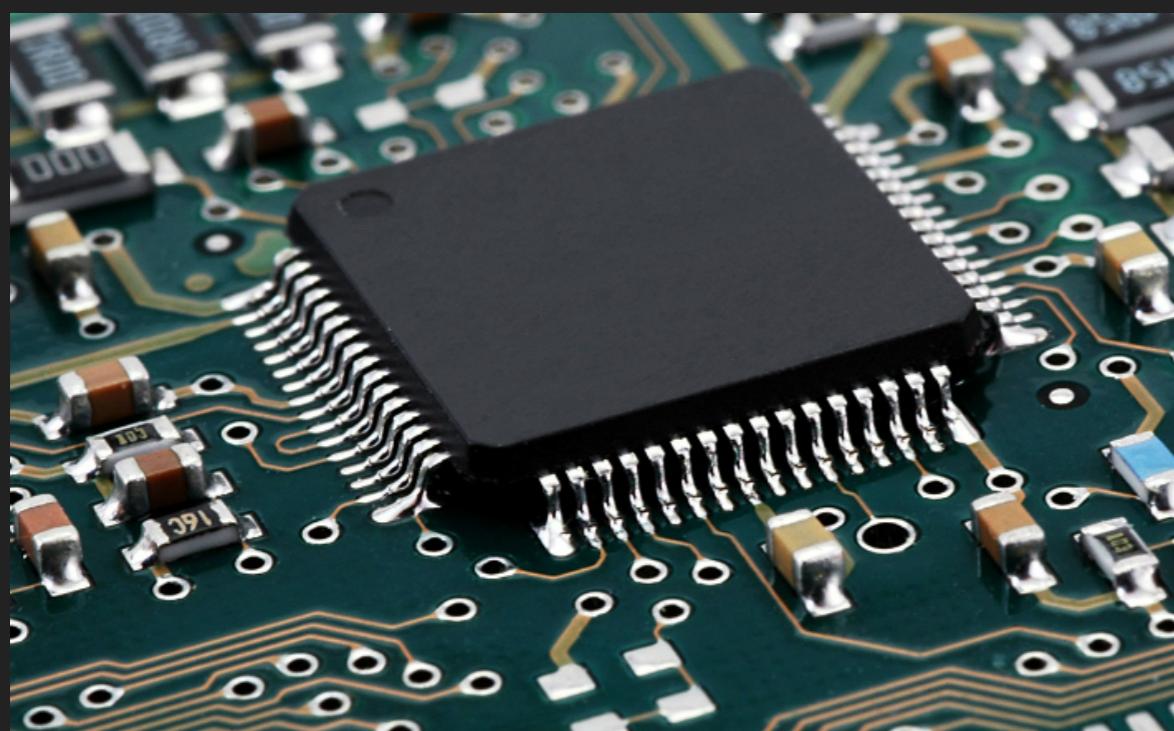
¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

- ▶ Transporte



¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

- ▶ Diseño de **circuitos integrados VLSI** (Very Large Scale Integration)



¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

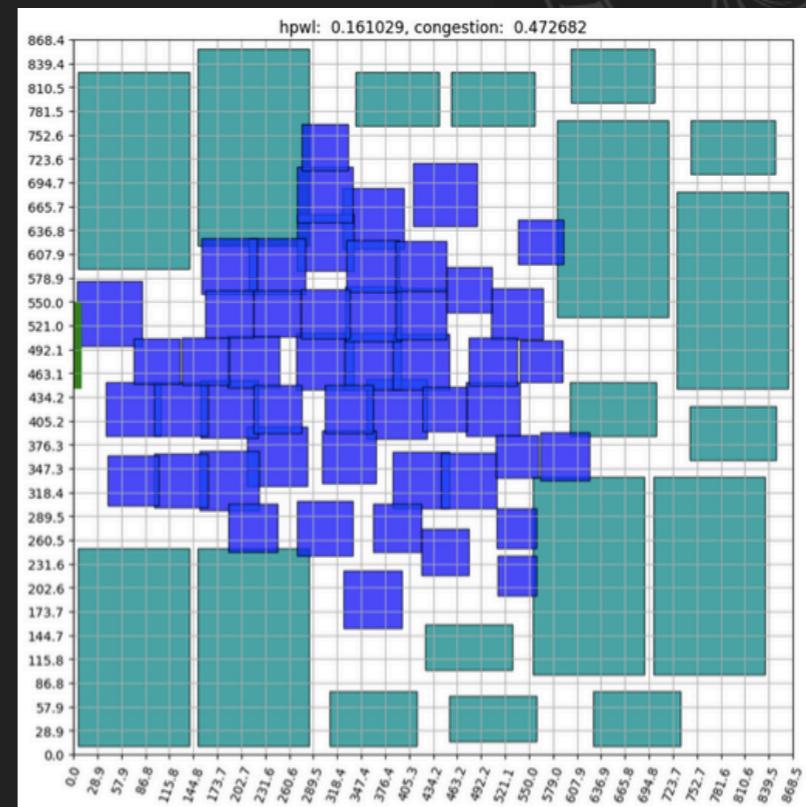
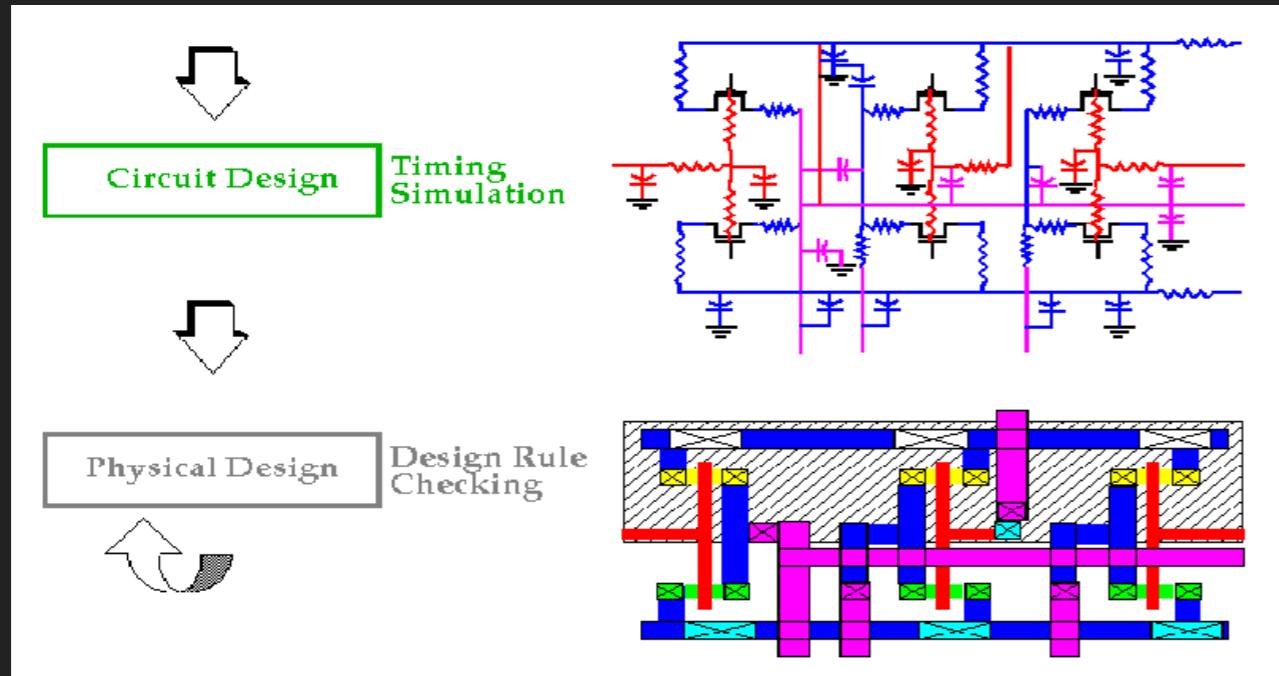
- ▶ Diseño de circuitos integrados VLSI (Very Large Scale Integration):

✓ Routing (enrutamiento de conexiones)

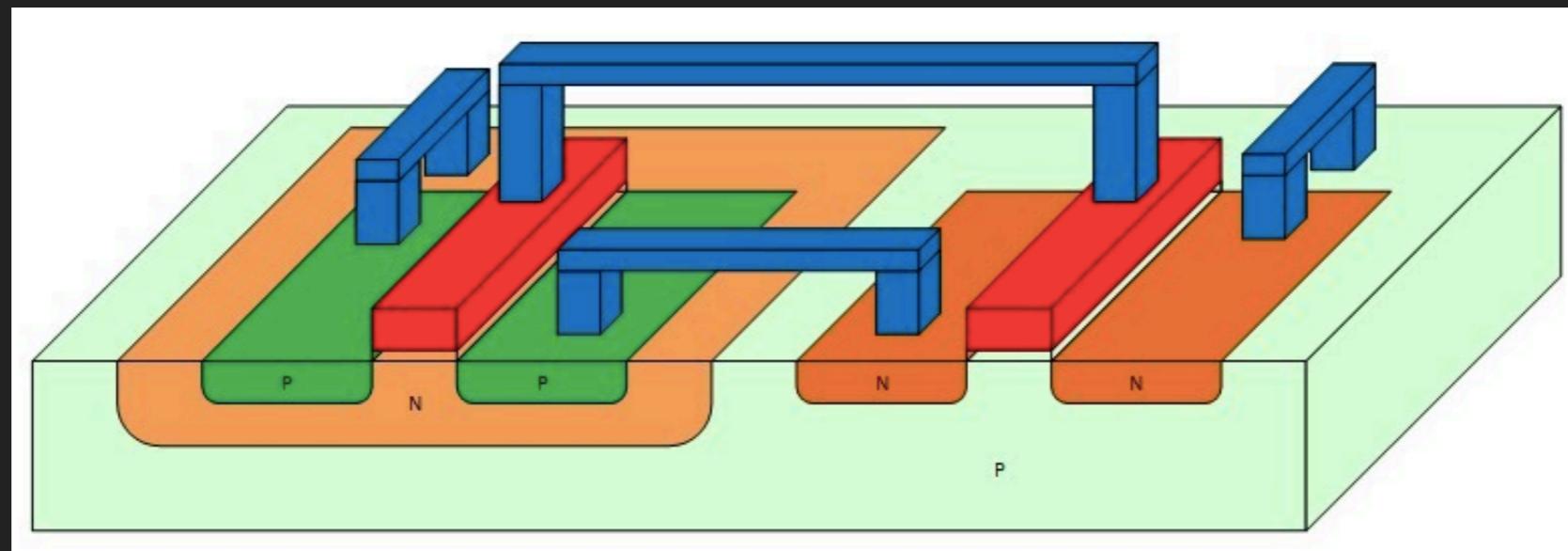
→ Interconexión, max. performance, min. Interferencia, min. longitud “cableado”.

✓ Placement (colocación de componentes macros y clústers)

→ Floorplanning y PPA

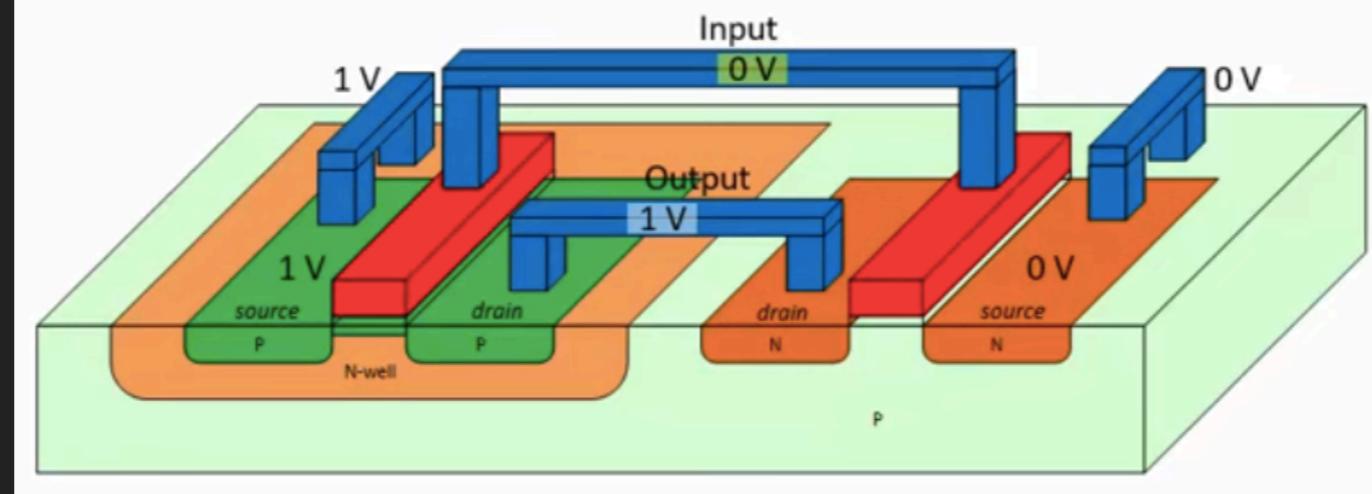


¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

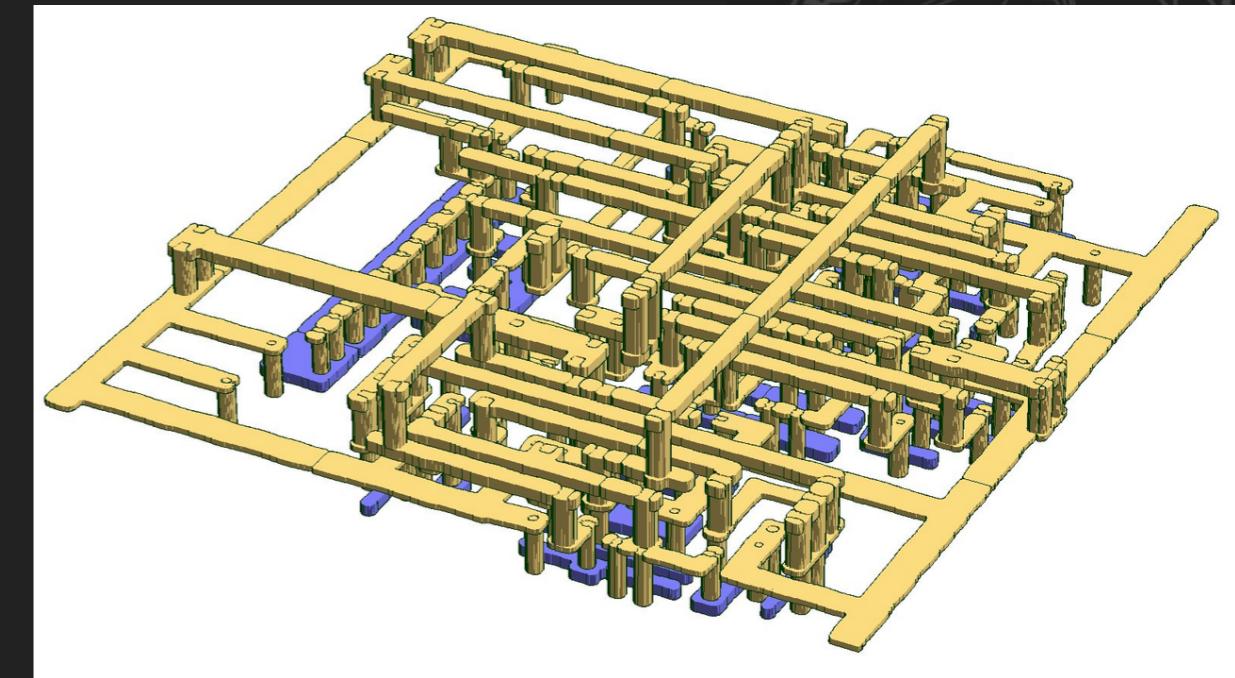
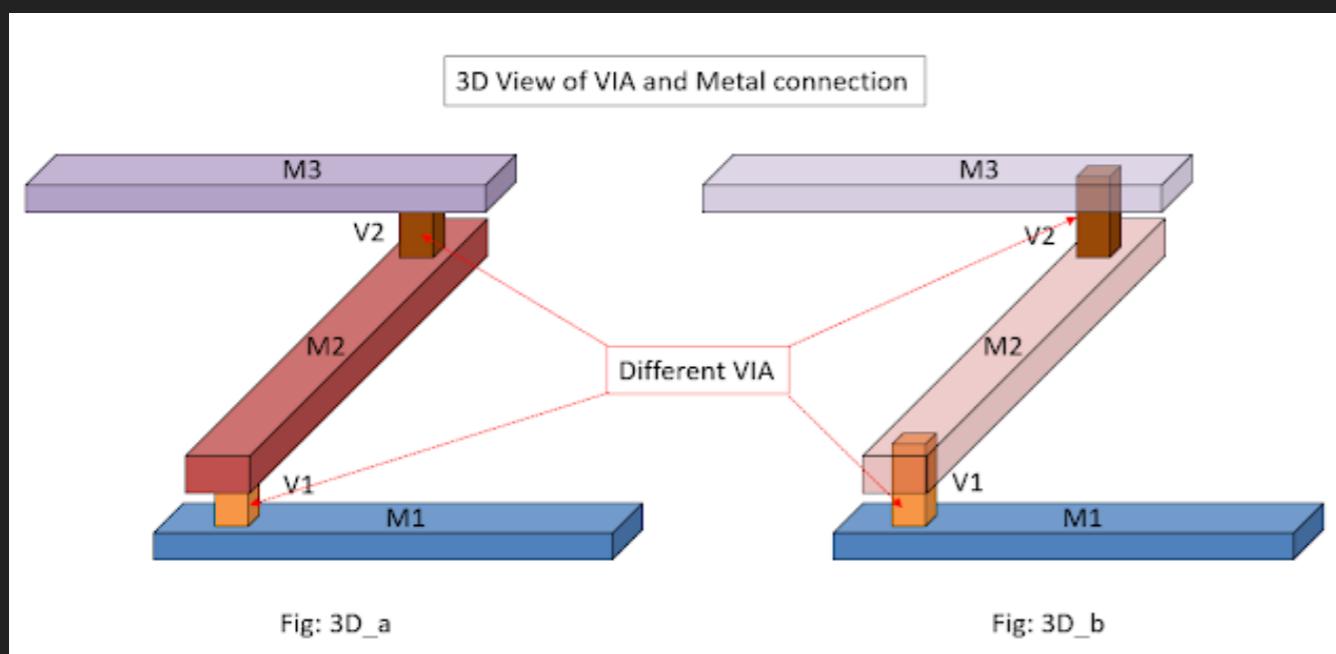


CMOS Technology: Inverter

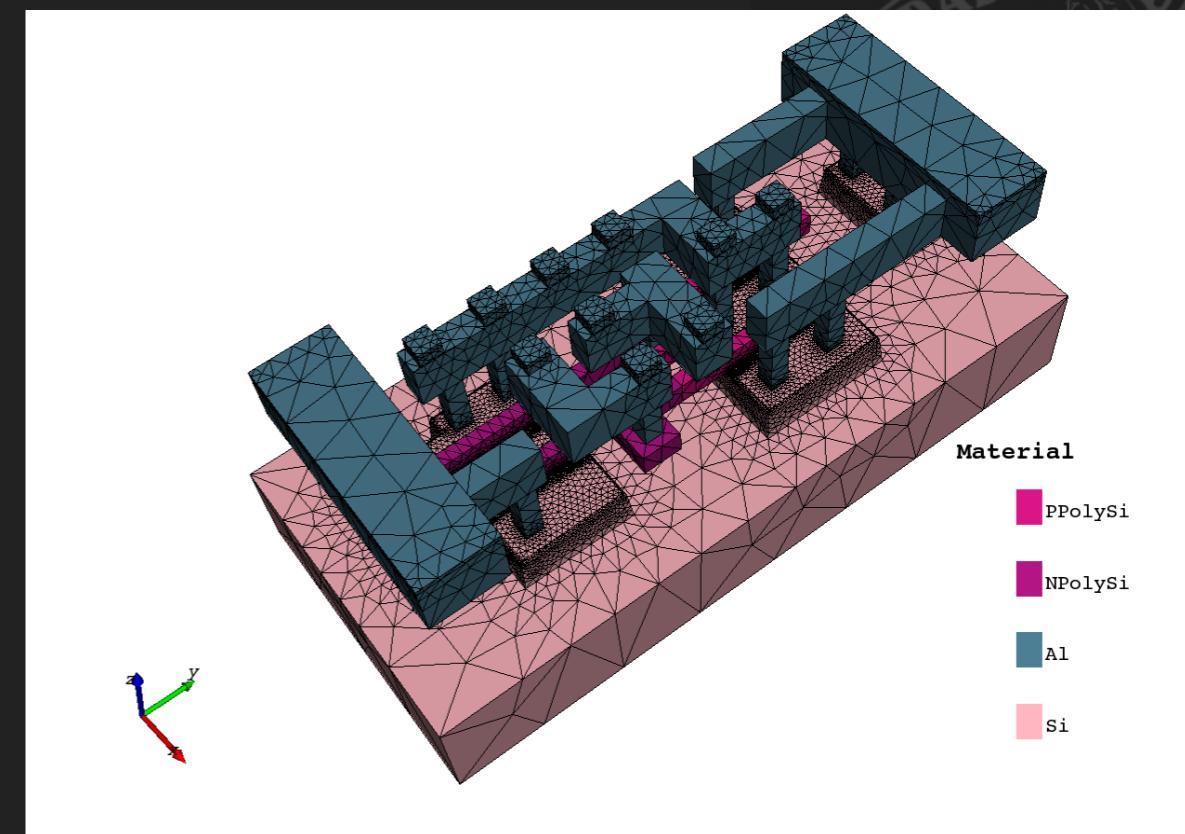
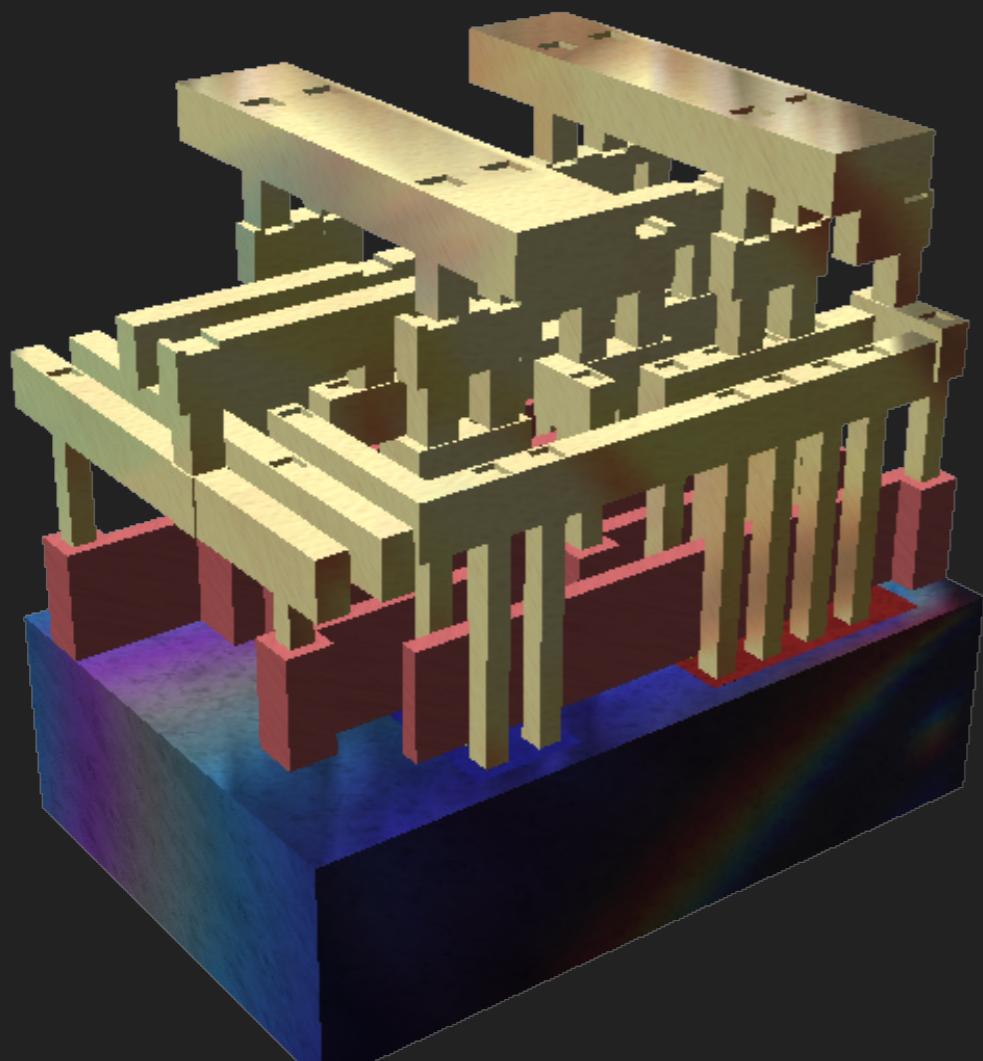
- Input 0: PMOS on, NMOS off, output = 1



¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

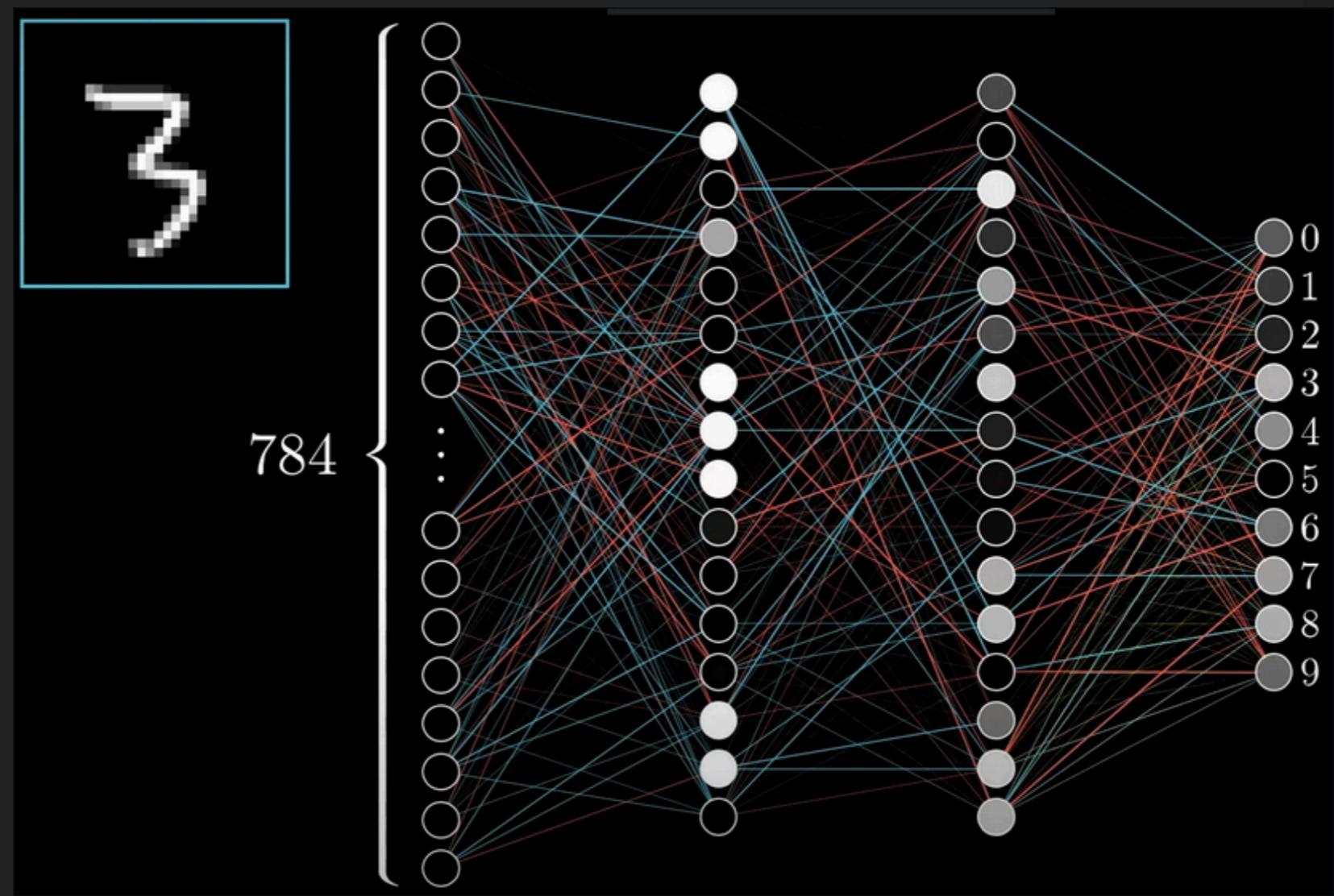


¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?



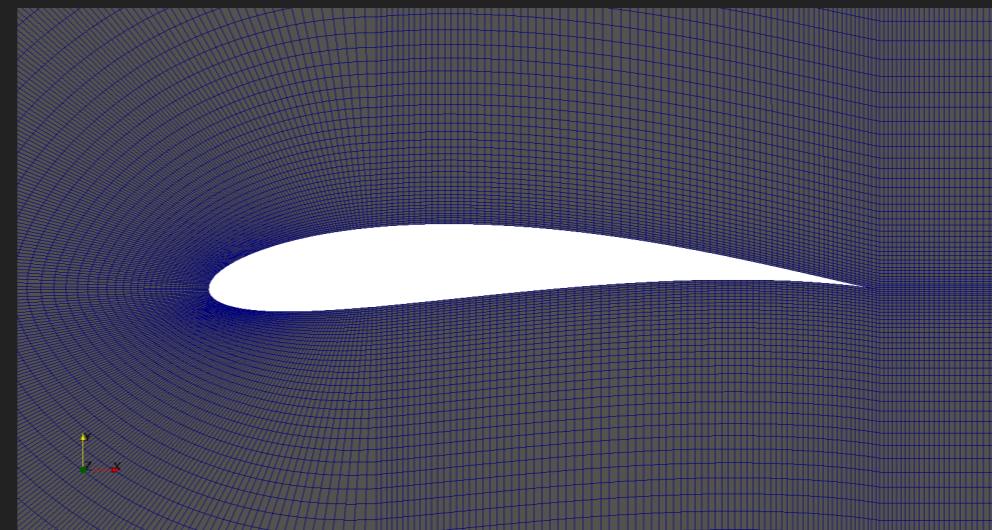
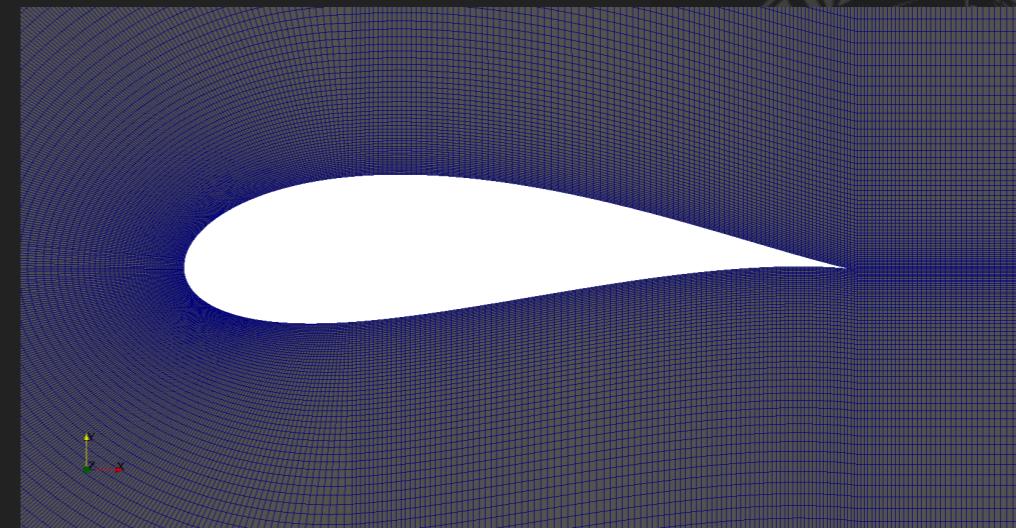
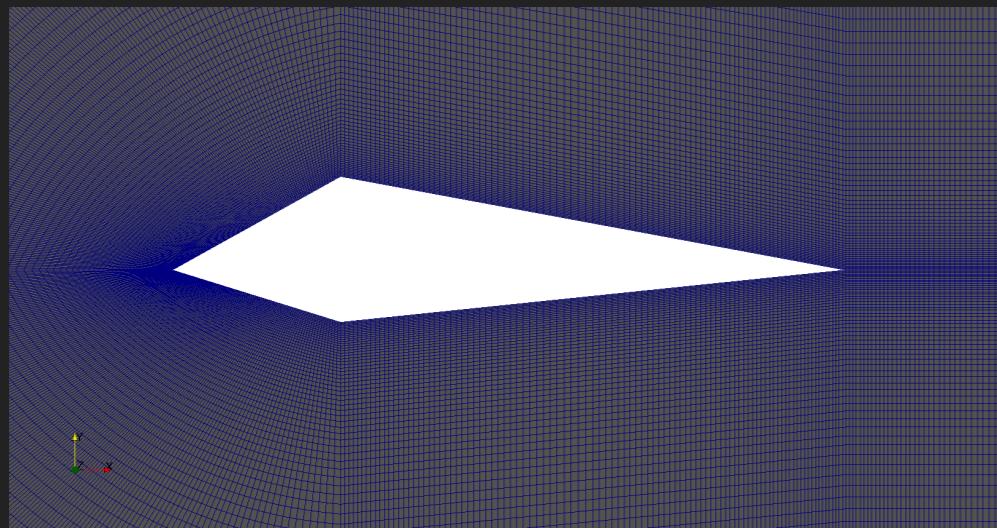
¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

- ▶ Optimización de hiperparámetros en DNN



¿DE QUE SE TRATA ALGORITMOS EVOLUTIVOS I?

- ▶ Diseño óptimo de aerodinámica en vehículos terrestres, aéreos o navales.



TEMAS Y TÉCNICAS A ABORDAR EN EL CURSO

- ▶ Conceptualización de términos utilizados en AE
- ▶ Algoritmos Genéticos (**GA**)
- ▶ Optimización por Enjambre de Partículas (**PSO**)
- ▶ Optimización basada en Enseñanza Aprendizaje (**TLBO**)
- ▶ Optimización basada en Colonia de Hormigas (**ACO**)
- ▶ Búsqueda Tabú (**TS**)
- ▶ Hiperheurísticas



CURSADO

- ▶ Parte 1 (~ 80 min.)
- ▶ Break (~ 10-20 min.)
- ▶ Parte 2 (~ 80 min.)



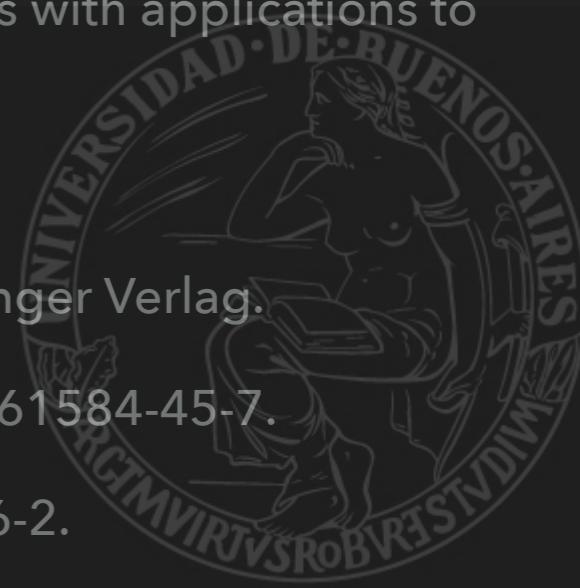
EVALUACIÓN

- ▶ Se aprueba la materia implementando **2 desafíos prácticos**.
 - ✓ Individual.
 - ✓ Grupal (hasta 2 integrantes).
 - ✓ Informe analítico y explicativo de al menos 3 carillas (.pdf).
 - ✓ Código fuente en un repositorio (enlace al repositorio en el informe, .py, .ipynb, otras extensiones asociadas al lenguaje Python solo si es necesario).
 - ✓ Se pueden presentar durante la cursada o al final.
 - ✓ Los desafíos son abiertos. El estudiante (o el grupo) elige un caso de uso según la técnica estudiada y elabora el desafío práctico.



REFERENCIAS BIBLIOGRÁFICAS Y WEB (I)

- ▶ Eiben, A. E., & Smith, J. E. (2015). *Introduction to evolutionary computing*. Springer-Verlag Berlin Heidelberg.
- ▶ Holland, J. H. (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press.
- ▶ Mitchell, M. (1998). *An introduction to genetic algorithms*. MIT press.
- ▶ Michalewicz Z. (1996). *Genetic Algorithms + Data Structures = Evolution Programs*. Springer Verlag.
- ▶ Udayan Das et al. (2024). *Introduction to Python Programming*. OpenStax. ISBN 978-1-961584-45-7.
- ▶ Introducing Python (Third release). Lubanovic B. (2019). O'Reilly. ISBN 978-1-449-35936-2.
- ▶ <https://www.python.org/downloads/release/python-3130b1/>
- ▶ <https://matplotlib.org>
- ▶ Roy, S., & Chakraborty, U. (2013). *Introduction to soft computing: neuro-fuzzy and genetic algorithms*. Pearson.
- ▶ Hillier, F. S. Lieberman, G. J. (2010). *Introducción a la Investigación de Operaciones*.



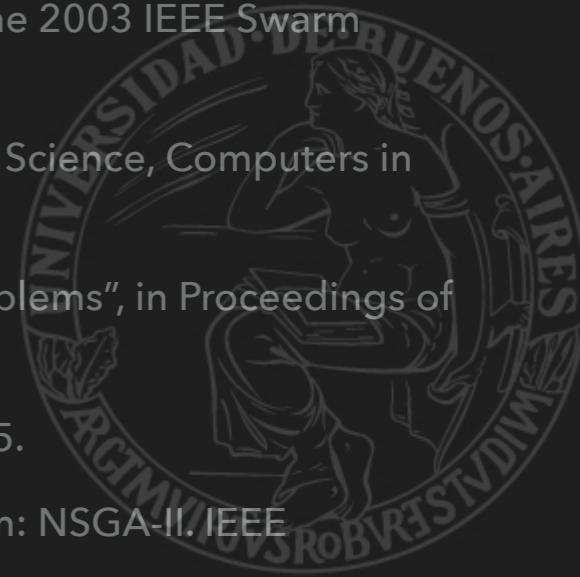
REFERENCIAS BIBLIOGRÁFICAS Y WEB (II)

- ▶ Lu Y, Hao J, Wu Q. (2022). Solving the clustered traveling salesman problem via traveling salesman problem methods. *PeerJ Computer Science* 8:e972
- ▶ Pratihar, D. K. (2007). Soft computing. Alpha Science International, Ltd.
- ▶ A. Cauchy. (1847). Méthode générale pour la résolution des systèmes d'équations simultanées, *Comp. Rend. Sci. Paris*, 25. 536-538.
- ▶ Chong, E. K., & Żak, S. H. (2013). An introduction to optimization (Vol. 75). John Wiley & Sons.
- ▶ Glover, F. (1986). Future paths for integer programming and links to artificial intelligence. *Computers & operations research*, 13(5), 533-549.
- ▶ J. Kennedy, R. Eberhart. (1995). Particle swarm optimization (in Neural Networks). *Proceedings., IEEE International Conference on*, vol. 4, pp. 1942 -1948 vol.4.
- ▶ Engelbrecht, A. P. (2007). Computational intelligence: an introduction. John Wiley & Sons.
- ▶ Clerc, M. (2010). Particle swarm optimization (Vol. 93). John Wiley & Sons.
- ▶ Shi, Y., & Eberhart, R. (1998). A Modified Particle Swarm Optimizer. *Proceedings of the IEEE International Conference on Evolutionary Computation*, 1998. IEEE World Congress on Computational Intelligence (Cat. No.98TH8360), Anchorage, AK, USA, 1998, pp. 69-73. DOI: 10.1109/ICEC.1998.699146.
- ▶ Bratton, D., & Kennedy, J. (2007, April). Defining a standard for particle swarm optimization. In 2007 IEEE swarm intelligence symposium (pp. 120-127). IEEE.



REFERENCIAS BIBLIOGRÁFICAS Y WEB (III)

- ▶ Parsopoulos, K. E., & Vrahatis, M. N. (2002). Particle swarm optimization method for constrained optimization problems. *Intelligent technologies-theory and application: New trends in intelligent technologies*, 76(1), 214-220.
- ▶ Hu, X., Eberhart, R. C., & Shi, Y. (2003, April). Engineering optimization with particle swarm. In *Proceedings of the 2003 IEEE Swarm Intelligence Symposium. SIS'03* (Cat. No. 03EX706) (pp. 53-57). IEEE.
- ▶ Coello Coello, C. A., "Use of a self-adaptive penalty approach for engineering optimization problems", Elsevier Science, Computers in Industry 41, 2000, pp. 113-127.
- ▶ Parsopoulos, K. E., and Vrahatis, M. N., "Particle Swarm Optimization Method for Constrained Optimization Problems", in *Proceedings of the Euro-International Symposium on Computational Intelligence*, 2002.
- ▶ Engelbrecht, A. P., "Fundamentals of Computational Swarm Intelligence", John Wiley & Sons Ltd, England, 2005.
- ▶ Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. A. M. T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE transactions on evolutionary computation*, 6(2), 182-197.
- ▶ Zitzler, E., Laumanns, M., & Thiele, L. (2001). SPEA2: Improving the strength Pareto evolutionary algorithm. *TIK report*, 103.
- ▶ Coello, C. C., & Lechuga, M. S. (2002, May). MOPSO: A proposal for multiple objective particle swarm optimization. In *Proceedings of the 2002 Congress on Evolutionary Computation. CEC'02* (Cat. No. 02TH8600) (Vol. 2, pp. 1051-1056). IEEE.
- ▶ López, J. (2013). Optimización multiobjetivo: aplicaciones a problemas del mundo real. Buenos Aires, Argentina, Universidad Nacional de la Plata.
- ▶ Yucra López, Carlos Enrique (2021). Biblioteca para la comparación estadística de algoritmos evolutivos. Proyecto Fin de Carrera / Trabajo Fin de Grado, Madrid, España.



REFERENCIAS BIBLIOGRÁFICAS Y WEB (IV)

- ▶ <https://www.youtube.com/watch?v=oSrUsM0hoPs> CMOS Tech: NMOS and PMOS Transistors in CMOS Inverter (3-D View)
- ▶ <https://www.youtube.com/watch?v=1Lad28K3Xi0> CMOS Fabrication Process (Animation)
- ▶ <https://www.makinarocks.ai/en/application-specific-integrated-circuit-asic-floorplan-automation-part-ii/>

