Rubio, F., de la Encina, A., Rabanal, P., Rodríguez, I.: Una librería para inteligencia de enjambre basada en la programación funcional. In: Ortega Mallén, Y. (ed.) Actas de las XVIII Jornadas de Programación y Lenguajes (PROLE 2018). Sistedes (2018). https://hdl.handle.net/11705/PROLE/2018/018

Una librería para inteligencia de enjambre basada en la programación funcional*

Fernando Rubio, Alberto de la Encina, Pablo Rabanal, Ismael Rodríguez

Facultad Informática. Universidad Complutense de Madrid.
28040 Madrid, Spain

fernando@sip.ucm.es,albertoe@sip.ucm.es,prabanal@fdi.ucm.es,isrodrig@sip.ucm.es

En este trabajo presentamos una librería de esqueletos paralelos para manejar metaheurísticas basadas en inteligencia de enjambre. La librería está implementada utilizando el lenguaje de programación funcional paralelo Eden, una extensión del lenguaje funcional Haskell. Gracias al orden superior presente en el lenguaje funcional, se simplifican las tareas de desarrollar código genérico, facilitando también la comparación entre distintas metaheurísticas.

Para un mismo problema es posible resolverlo utilizando diversas metaheurísticas bioinspiradas. Además, unas metaheurísticas podrán proporcionar mejores resultados para ciertos problemas, mientras que obtendrán resultados peores en otros, no existiendo ninguna metaheurística que pueda considerarse mejor en el caso general. Es más, a la hora de proporcionar implementaciones paralelas de cada metaheurística, existen múltiples métodos que pueden obtener mejores o peores resultados dependiendo de diversos factores. Por tanto, resulta especialmente interesante disponer de una librería que proporcione al programador facilidades para probar diversas metaheurísticas y diversas paralelizaciones de las mismas, todo ello usando un único marco de trabajo. En este sentido, el uso de lenguajes funcionales facilita dicha tarea. En el presente trabajo se utiliza el lenguaje funcional paralelo Eden [1] para proporcionar una librería de esqueletos paralelos que implementan diversas metaheurísticas, de modo que el programador pueda probar de manera rápida y sencilla el resultado que se obtendría con distintas configuraciones. El artículo completo [4] fue publicado en el congreso IWANN'17, si bien piezas menores del mismo ya habían aparecido publicadas previamente (véase [2, 3]).

References

[1] Rita Loogen (2012): *Eden - Parallel Functional Programming with Haskell*. In: *CEFP'11, LNCS* 7241, Springer, pp. 142–206.

[2] P. Rabanal, I. Rodríguez & F. Rubio (2014): *Parallelizing Particle Swarm Optimization in a Functional Programming Environment. Algorithms* 7(4), pp. 554–581.

[3] F. Rubio, A. de la Encina, P. Rabanal & I. Rodríguez (2013): *Eden's Bees: Parallelizing Artificial Bee Colony in a Functional Environment.* In: *ICCS'13*, pp. 661–670.

[4] Fernando Rubio, Alberto de la Encina, Pablo Rabanal & Ismael Rodríguez (2017): *A Parallel Swarm Library Based on Functional Programming*. In: *Advances in Computational Intelligence, LNCS 10305*, Springer, pp. 3–15.

*This work has been partially supported by projects TIN2012-39391-C04-04, TIN2015-67522-C3-3-R, and S2013/ICE-2731.

Submitted to:
PROLE 2018

© Rubio, Encina, Rabanal, Rodríguez
This work is licensed under the
Creative Commons Attribution License.