

Framework de Algoritmos Evolutivos para la Formación Automática de Grupos de Aprendizaje

Joan Sol Roo¹, Marianela Ciolfi Felice¹

¹Facultad de Ciencias Exactas, UNCPBA
{solroo, marianelaciolfi}@gmail.com

Cátedra: Introducción a la Computación Evolutiva

Profesor: Dra. Virginia Yannibelli

Resumen. En el presente artículo, se presenta un framework basado en algoritmos evolutivos para resolver el problema de la formación automática de grupos de aprendizaje. Luego, se describe una instanciación del framework que permite conformar automáticamente grupos de aprendizaje balanceados en cuanto a los roles de sus miembros. Se presentan los experimentos computacionales que han sido desarrollados para evaluar la efectividad y eficiencia de dicha instanciación, y se analizan los resultados obtenidos. Con respecto a dichos resultados, es posible decir que la instanciación desarrollada ha logrado un alto nivel de efectividad y eficiencia. Finalmente, se discute la posibilidad de incorporar al framework otros criterios para la formación automática de grupos de aprendizaje y la utilización simultánea de múltiples criterios de formación.

Palabras clave. Framework, algoritmos evolutivos, algoritmos genéticos, formación automática de grupos, grupos de aprendizaje, criterios de formación de grupos, roles de Belbin.

1 Introducción

En un ambiente de aprendizaje colaborativo, la formación de los grupos de aprendizaje es un aspecto central. Esto se debe a que la manera en la cual se compone un grupo de aprendizaje afecta tanto el nivel de aprendizaje como el comportamiento social de los estudiantes que componen el grupo [1-3]. Los criterios existentes para la formación de grupos de aprendizaje pueden utilizarse de manera individual o de manera combinada. En este último caso el objetivo es considerar diversos factores de forma simultánea como, por ejemplo, estilos de aprendizaje, nivel de conocimiento previo en el tema a aprender, edad, afinidad.

La formación de grupos de aprendizaje constituye un problema que, lejos de ser trivial, requiere por parte de los profesores, una valiosa cantidad de tiempo, esfuerzo y conocimiento acerca de las características de los estudiantes. De este modo, resulta muy útil contar con una solución que automatice este proceso, con el objetivo de ha-

cerlo lo más eficientemente posible y aumentar las posibilidades de éxito de los grupos.

En la literatura relacionada, se han propuesto distintas soluciones que difieren en cuanto al criterio utilizado para formar los grupos de aprendizaje. Sin embargo, de acuerdo a lo investigado, no se han propuesto soluciones que implementen varios criterios de formación de grupos y que permitan combinarlos. Por este motivo, en el presente artículo se propone un framework basado en algoritmos evolutivos al que fácilmente pueden incorporarse nuevos criterios para ser utilizados de manera individual o combinada a la hora de formar los grupos.

Sin embargo, en la literatura relacionada se puede observar la falta de soluciones que tengan en cuenta más de un criterio de formación a la vez. Por este motivo, en el presente artículo se propone un framework basado en algoritmos evolutivos, al que fácilmente pueden incorporarse nuevos criterios para ser combinados a la hora de formar los grupos.

Se decidió utilizar algoritmos evolutivos debido a que el problema abordado es un problema de optimización de tipo NP-Hard y, en este sentido, los algoritmos evolutivos han mostrado ser efectivos y eficientes en la resolución de una amplia variedad de problemas de este tipo [4].

El resto del artículo se organiza de la siguiente manera. En la sección 2, se presentan algunos trabajos relacionados que se han desarrollado en este contexto. En la sección 3, se detalla el diseño del framework desarrollado. En la sección 4, se expone la instanciación realizada del framework y los experimentos computacionales llevados a cabo para evaluar la efectividad y eficiencia de dicha instanciación. En la sección 5, se presentan las conclusiones de este trabajo. Finalmente, en la sección 6, se mencionan diversas líneas de trabajo a futuro.

2 Trabajos Relacionados

Existen diversos criterios para la formación automática de grupos de aprendizaje. Estos criterios han sido utilizados en gran variedad de estudios que pueden encontrarse en la literatura. Dichos estudios usualmente consideran factores como el estilo de aprendizaje de los alumnos [5] [6], su estilo de pensamiento [7], su interés y conocimiento previo sobre el tema a aprender [8] [9], o bien características como su género, habilidades y tiempo disponible [10].

Por otro lado, se tienen los criterios que buscan armarlos grupos de manera balanceada en cuanto a los roles de los integrantes, entendiéndose como rol la tendencia a comportarse, contribuir e interrelacionarse con otros de una manera particular [11] [12]. Estos criterios asumen que un nivel de balance alto permite aumentar la colaboración y beneficiar las relaciones sociales. En la literatura existen diversos ejemplos de modelos que proponen esta estrategia [11-18].

Uno de los más ampliamente aceptados es el de Belbin, que se basa en los roles naturales de los participantes. Según Belbin, las personas poseen un nivel de preferencia por cada rol. Además, sostiene que cuando alguien muestra una preferencia alta por determinados roles, es capaz de llevarlos a cabo espontáneamente.

Si bien este criterio fue diseñado para grupos de trabajo, ha sido aplicado a grupos de aprendizaje a lo largo del tiempo con resultados exitosos [19-23]. Recientemente, Yannibelli y Amandi utilizaron las condiciones de balance planteadas por Belbin, en un algoritmo genético [22] y en uno memético [23], para la formación de grupos de aprendizaje en el contexto de estudiantes del área de Ingeniería de Software.

Sin embargo, en la literatura relacionada se puede notar la inexistencia de soluciones que consideren varios criterios de formación. Resulta interesante, entonces, contar con un framework al que fácilmente puedan incorporarse nuevos criterios, permitiendo su aplicación tanto aislada como conjunta.

3 Framework de Algoritmos Evolutivos para Grupos de Aprendizaje

En la sección 3.1 se explica brevemente la estructura y el funcionamiento del algoritmo evolutivo implementado. En la 3.2 se muestra el diseño del framework desarrollado, en que se ejecuta dicho algoritmo. En la 3.3 se detalla el problema de la formación de grupos de aprendizaje y cómo es abordado por el diseño del framework. Finalmente, en la 3.4 se describe la interfaz gráfica del mismo.

3.1 Algoritmo Evolutivo

El algoritmo parte de una población inicial formada por soluciones válidas generadas de manera aleatoria. Se realiza un proceso de selección de padres, en el que algunas son elegidas para aplicarles un operador de cruce. En general, aquellas con mayor fitness (valor que representa cuán buena es una solución con respecto al objetivo de optimización del problema) tienen más chances de ser seleccionadas. A su vez, los individuos pasan por un proceso de mutación, con el fin de introducir diversidad genética. Una vez generada la descendencia, se procede a la selección de sobrevivientes, para crear una nueva población con alguna combinación de individuos de la anterior generación y de la nueva. Como se muestra en la Figura 1, este proceso se repite hasta que se cumple el criterio de terminación del algoritmo.

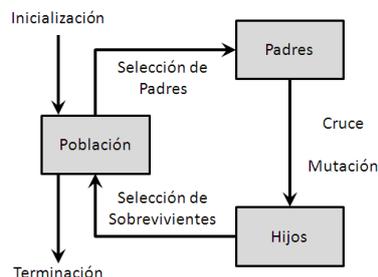


Fig. 1. Diagrama de flujo de un algoritmo evolutivo

3.2 Framework de Algoritmos Evolutivos

Se planteó un diseño orientado a objetos a fin de maximizar la flexibilidad y extensibilidad del framework. En el diagrama de la Figura 2 pueden observarse las interfaces para cada una de las etapas del algoritmo, junto con las implementaciones de algunas de las técnicas más utilizadas. Si bien inicialmente se provee la implementación de un algoritmo genético, el framework cuenta con la interfaz *Algoritmo Evolutivo* que puede implementarse para dar lugar a, por ejemplo, un algoritmo memético.

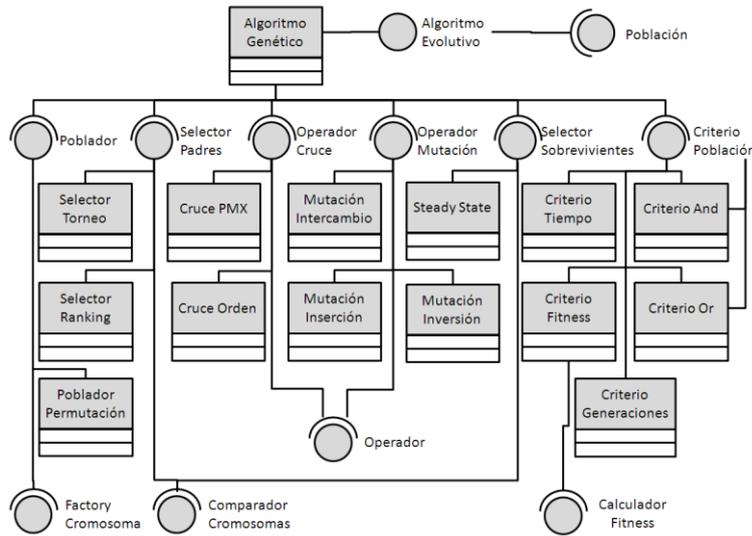


Fig. 2. Clases vinculadas al control del algoritmo

En la Figura 3 se presentan las interfaces para la manipulación y evaluación de poblaciones, junto con dos índices implementados para la mejora de performance.

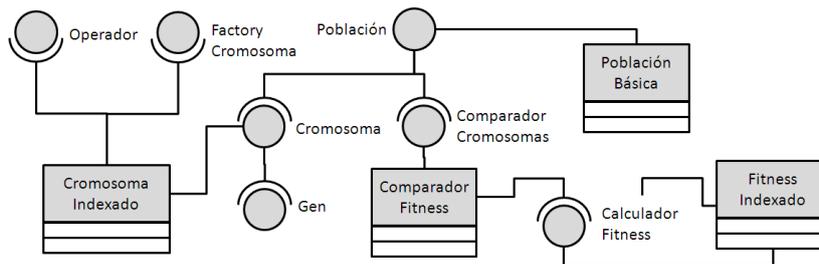


Fig. 3. Clases vinculadas al manejo de la población

3.3 Grupos de Aprendizaje

El problema a resolver a partir del framework consiste en la formación automática de grupos de aprendizaje maximizando el cumplimiento del criterio tenido en cuenta. Se tiene un conjunto de n estudiantes: $E = \{e_1, e_2, \dots, e_n\}$, que se quiere dividir en g grupos: $G = \{G_1, G_2, \dots, G_g\}$ del mismo tamaño (en caso de que esto no sea posible, las cantidades podrán diferir en a lo sumo 1). Cada alumno debe pertenecer a un único equipo. Los valores de E , n y g son conocidos y conforman la entrada.

En la Figura 4 se encuentran las clases que representan el grupo de aprendizaje, junto al individuo y el criterio de evaluación.

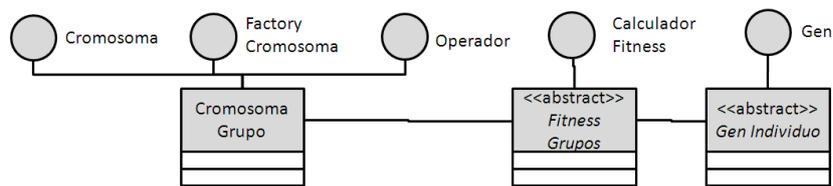


Fig. 4. Clases vinculadas al manejo de grupos

Lo presentado en las secciones 3.2 y en la actual, permite afirmar que gracias a la abstracción y al uso de interfaces, es sencillo incorporar un nuevo criterio, así como utilizar el framework para resolver otros problemas que puedan beneficiarse de los algoritmos genéticos.

3.4 Interfaz

Se provee una interfaz visual que permite la configuración de cada uno de los parámetros del algoritmo, así como la selección del criterio a utilizar, y para un mismo criterio, la función de fitness a emplear. El agregado de nuevas clases de configuración, así como de funciones de fitness no presenta complicaciones.

La interfaz permite el seguimiento de la población en tiempo real, generación a generación, mediante una tabla de colores (Figura 5, izquierda). Se representa la población en orden descendente; cada solución es una fila, en la que el brillo representa el fitness; cada característica –por ejemplo, para el criterio de Belbin, el rol preferido por el estudiante- tiene un color único.

Además, se puede visualizar la evolución de la función de fitness a través de un gráfico de líneas (Figura 5, derecha), pudiendo adicionalmente compararla con ejecuciones previas. Por último, los datos generacionales se pueden almacenar para su futuro estudio y análisis.

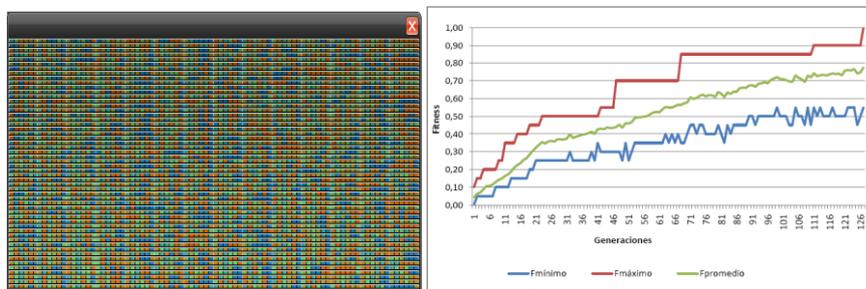


Fig. 5. Progreso de la población. Izquierda: Tablade colores. Derecha: Gráfico de líneas

4 Instanciación: Roles de Belbin

Se realizó una instanciación del framework para resolver el problema de la formación automática de grupos utilizando el criterio de los roles de Belbin.

4.1 Formalización del Criterio de Belbin

Se tiene información sobre la preferencia de cada estudiante por cada uno de los roles definidos por Belbin[11][12], los cuales se muestran en la Tabla 1. Estos datos se obtienen a través de autoevaluaciones que deben completar los alumnos, en el marco del método BTRSPI (BelbinTeam-Role Self-PerceptionInventory)[11][12].

De acuerdo al criterio de Belbin, un grupo está balanceado si cada rol es interpretado naturalmente por sólo un miembro del grupo [11]. Por el contrario, un grupo está desbalanceado si hay roles que no son llevados a cabo naturalmente por ningún integrante, o si varios de ellos juegan el mismo rol [11], [12].

Tabla 1. Roles de Belbin

Rol	Contribución
Cerebro (CE)	Creativo, poco ortodoxo. Resuelve problemas difíciles.
Investigador de Recursos (IR)	Extrovertido, entusiasta, comunicativo. Busca nuevas oportunidades. Desarrolla contactos.
Coordinador (CO)	Maduro, seguro de sí mismo. Aclara las metas a alcanzar. Promueve la toma de decisiones. Delega bien.
Impulsor (IS)	Dinámico, trabaja bien bajo presión. Tiene iniciativa y coraje para superar obstáculos.
Monitor Evaluador (ME)	Serio, perspicaz y estratega. Distingue todas las opciones. Juzga con exactitud.
Cohesionador (CH)	Cooperador, apacible, perceptivo y diplomático. Escucha e impide los enfrentamientos.
Implementador (ID)	Disciplinado, leal, conservador y eficiente. Transforma las ideas en acciones.

Finalizador (FI)	Esmerado, concienzudo, ansioso. Busca los errores y las omisiones. Realiza las tareas en el plazo establecido.
Especialista (ES)	Sólo le interesa una cosa a la vez. Aporta cualidades y conocimientos específicos.

4.2 Funciones de Fitness

Como primera función de fitness, se consideró el siguiente modelo [23]: se asigna un puntaje a cada rol r en el grupo G , según la cantidad de individuos que lo desempeñan naturalmente, a saber:

$$nr(G, r) = \begin{cases} 1 & \text{si } r \text{ se desempeña naturalmente por solo un individuo en } G \\ -2 & \text{si } r \text{ no se desempeña naturalmente por ningún individuo en } G \\ -p & \text{si } r \text{ es desempeñado por } p \text{ individuos en } G \end{cases}$$

Entonces, el fitness de cada grupo puede expresarse como:

$$fitness_{grupo} = \frac{\text{únicos} - \text{repetidos} - 2 * \text{faltantes}}{\text{únicos} + \text{faltantes}} \tag{1}$$

A fines comparativos, se planteó un segundo modelo para el criterio: para cada grupo, se considera el factor que representan los roles únicos encontrados sobre el total de roles encontrados más los roles faltantes; el modelo se refleja en la Fórmula 2:

$$fitness_{grupo} = \frac{\text{únicos}}{\text{únicos} + \text{repetidos} + \text{faltantes}} \tag{2}$$

El Fitness de una solución será entonces el promedio de los fitness de los grupos que la componen.

4.3 Implementación

Debido al diseño utilizado, sólo es necesario implementar la representación de los individuos (como el conjunto de sus roles preferidos) y la función de fitness correspondiente a cada una de las fórmulas propuestas en el inciso anterior, como puede observarse en la Figura 6.

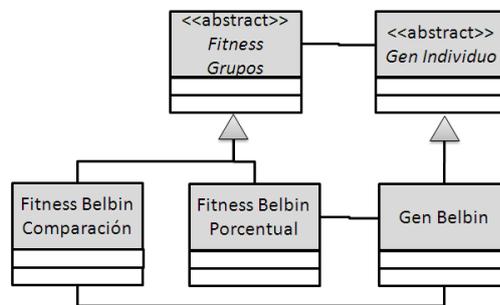


Fig. 6. Implementación del criterio de roles de Belbin

4.4 Experimentación

Se tomó un grupo de 120 individuos, a dividir en 20 grupos, utilizando ambas funciones de fitness (fórmulas (1) y (2)). Se evaluaron las distintas configuraciones de los parámetros disponibles (Tabla 2), para determinar cuáles ofrecían potencial para ser estudiadas en detalle.

Tabla 2. Configuración de parámetros de la primera tanda de pruebas

Parámetro	Configuraciones	
	Métodos posibles	Valores posibles
Generación de la población inicial	Permutación aleatoria	Tamaño: 50 individuos
Selección de padres	Ranking, Torneo	Monógamo, Polígamo
Cruce	Orden, PMX	Probabilidad: {50, 60, 70, 80, 90, 100}%
Mutación	Inserción, Intercambio, Inversión	Probabilidad: {0, 10, 20, 30, 40, 50}%
Selección de Sobrevivientes	Steadystate	K: {30, 40, 50, 60, 70}%
Criterio de Finalización	Fitness máximo	1.00
	Tiempo máximo	120 segundos

Las configuraciones más prometedoras fueron ejecutadas repetidas veces con pequeñas variaciones de sus parámetros. Los resultados finales de las pruebas, para una población de tamaño 100, se muestran en las tablas 3 y 4, donde Función 1 corresponde a la dada por la fórmula (1) y Función (2), por la fórmula (2).

Tabla 3. Mejor configuración para las funciones de fitness implementadas

	Función 1	Función 2
Selección de padres	Torneo polígamo	Torneo polígamo
Operador de cruce	Orden (prob.: 100%)	Orden (prob.: 90%)
Operador de mutación	Inversión (prob.: 5%)	Inversión (prob.: 7%)
Selección de sobrevivientes	Steadystate (k: 50%)	Steadystate (k: 40%)

Tabla 4. Resultados obtenidos con la mejor configuración para las funciones de fitness implementadas

	Función 1	Función 2
Tiempo de ejecución (segundos)	36,2	11,82
Iteraciones (promedio de 20 ejecuciones)	644,6	212,75
Fitness mínimo	0,66	0,52
Fitness máximo	1,00	1,00
Fitness promedio	0,89	0,78

Como puede observarse, el tiempo promedio de ejecución es considerablemente menor para la Función 2, así como el número de iteraciones que toma llegar al fitness máximo. En cambio, tanto el fitness mínimo como el promedio son menores que los que arroja la Función 1, sin embargo, estos factores no tienen peso en el problema pues el objetivo es encontrar una solución con fitness óptimo.

El análisis realizado indica que la utilización del cruce de orden con probabilidad alta (90 a 100%), en conjunto con mutación por inversión (de 0 a 10%) y una supervivencia de alrededor de la mitad de los mejores padres, presenta la mayor eficiencia al resolver la instancia dada. Una posible explicación del buen desempeño es la preservación de las posiciones relativas, fomentando la asociación progresiva de individuos cuyos roles son complementarios.

5 Conclusiones

Se planteó el desarrollo de un framework de algoritmos evolutivos, con el objetivo de proporcionar una solución al problema de la formación automática de grupos de aprendizaje. Se instanció el framework con el criterio de roles de Belbin, implementado dos funciones de fitness distintas. Se desarrollaron experimentos computacionales para evaluar la efectividad y eficiencia de la instanciación desarrollada. Se encontró que la segunda función de fitness implementada resuelve la instancia del problema en un tiempo aceptable. Como se mostró en la sección 3, gracias a la abstracción y al uso de interfaces, la implementación de otros criterios de formación de grupos, así como su utilización conjunta, es una tarea muy sencilla.

6 Trabajos futuros

Como trabajo futuro, se pretende realizar más pruebas en el contexto del criterio de Belbin, con distintas instancias del problema, para aumentar la confianza del análisis y verificar así la efectividad de las funciones de fitness implementadas.

Se implementarán, además, otros criterios para la formación de grupos de aprendizaje, tarea que debido al diseño del framework no presentará inconvenientes.

Por último, se extenderá el algoritmo genético desarrollado para generar un algoritmo memético (algoritmo híbrido que combina un algoritmo evolutivo y un algoritmo de búsqueda local) y se analizará si la búsqueda local que el algoritmo memético incluye mejora los resultados obtenidos.

7 Referencias

- [1] Beane, W. E., Lemke, E. A.: Group variables influencing the transfer of conceptual behavior. *Journal of Educational Psychology*. 215–218 (1971)
- [2] Dalton, D. W., Hannafin, M. J., Hooper, S.: Effects of individual and cooperative computer-assisted instruction on student performance and attitudes. *Educational Technology Research and Development*. 15–24 (1989)

- [3] Hooper, S., Hannafin, M. J.: Cooperative CBI: The effects of heterogeneous versus homogeneous group on the learning of progressively complex concepts. *Journal of Educational Computing Research*. 413–424 (1988)
- [4] Eiben, A. E., Smith, J. E.: *Introduction to evolutionary computing* (2nd ed.). Springer (2007)
- [5] Felder, R.M., Silverman, L.K.: Learning Styles and Teaching Styles in Engineering Education. *Engr. Education*. 674-681 (1988)
- [6] Honey, P., Mumford, A.: *The learning styles helper's guide*. Peter Honey Publications Ltd., Maidenhead (2000)
- [7] Wang, D. Y., Lin, S. S. J., Sun, C. T.: DIANA: A computer-supported heterogeneous groupings system for teachers to conduct successful small learning groups. *Computers in human Behaviors*. (2007)
- [8] Graf, S., Bekele, R.: *Forming Heterogeneous Groups for Intelligent Collaborative Learning Systems with Ant Colony Optimization*. *Lecture Notes in Computer Science*. 217-226 (2006)
- [9] Lin, Y. T., Huang, Y. M., Cheng, S. C.: A automatic group composition system for composing collaborative learning groups using enhanced particles swarm optimization. *Computers & Education*. 1483-1493 (2010)
- [10] Cavanaugh, R., Ellis, M., Layton, R., Ardis, M.: *Automating the Process of Assigning Students to Cooperative-Learning Teams*. 2004 American Society for Engineering Education Annual Conference & Exposition (2004)
- [11] Belbin, R. M.: *Management teams: Why they succeed or fail*. Butterworth-Heinemann, Oxford (1981)
- [12] Belbin, R. M.: *Team roles at work*. Butterworth-Heinemann, Oxford (1993)
- [13] Felder R. M.: *Reaching the second tier*. *Journal of College Science Teaching*. 286–290 (1993)
- [14] Bridges W.: *The Character of Organizations: Using Personality Type in Organization Development*. Davies-Black Publishers (2000)
- [15] Jensen D., Feland J., Bowe M., Self B.: A 6-Hats based team formation strategy: Development and comparison with an MBTI based approach. In: *Proceedings of the ASEE Annual Conference* (2000)
- [16] Margerison, C.J., McCann D.J.: *How to Lead a Winning Team*. University Press, Bradford (1985)
- [17] McGregor, D.: *The Human Side of Enterprise*. McGraw Hill, New York (1960)
- [18] Buchanan, D. Huczynski, H.: *Organisational Behaviour*. Prentice Hall, London (1985)
- [19] Alban, L. Implementation of Belbin's model for the creation of teams in project based courses. In: *Proceedings of 20th Australasian Association for Engineering Education Conference*. 429-434 (2009)
- [20] Ounnas, A., Davis, H. C., Millard, D. E.: A Framework for Semantic Group Formation in Education. *Educational Technology & Society*. 3–55 (2009)
- [21] Jeffries, P., Grodzinsky, F., Griffin, J.: Building successful on-line learning communities across international boundaries: a Case Study. In: *Proceedings of ETHICOMP 2004*. University of the Aegean, Syros (2004)
- [22] Yannibelli, V., Amandi, A.: A deterministic crowding evolutionary algorithm to form learning teams in a collaborative learning context. *Expert Systems with Applications*. Pergamon Elsevier Science. Vol. 39, pp. 8584–8592, 2012. (2012).
- [23] Yannibelli, V., Amandi, A.: A metaheuristic algorithm for collaborative learning team formation in the context of software engineering courses. In: *LNCS*, Springer. In press (Acceptance date: November 2011)