

# Estudio del trabajo académico

## *“An ideal team is more than a team of ideal agents”\**

Jose Arias Moncho and Victoria Beltrán Domínguez

Universitat Politècnica de Valencia

**Abstract.** La formación de equipos de agentes para la resolución de tareas es un problema complejo y muy explorado en la literatura. En este trabajo se pretende realizar un estudio del trabajo académico “An ideal team is more than a team of ideal agents” [1]. En concreto, se realiza una presentación de trabajos relacionados dentro del área de investigación para posteriormente comentar de manera detallada los algoritmos presentados para construir equipos, utilizando para ello diferentes métricas y conceptos relacionados. Finalmente, se comenta los resultados obtenidos en este trabajo y se realizan unas conclusiones.

**Keywords:** team formation · algorithms · theoretic study

## 1 Introducción

Para realizar una tarea determinada a través de diversos agentes, necesitamos una manera de poder construir equipos maximizando el rendimiento. De esta manera, este trabajo tiene como objetivo averiguar qué conjunto de agentes conseguirá llevar a cabo una tarea particular de forma óptima.

Asignar agentes a un equipo con el fin de conseguir un propósito común, aunque pueda parecer trivial, es un problema NP-duro, pues aunque la intuición nos diga que lo más sencillo es dividir el problema en pequeñas tareas y asignar cada subtarea a aquel agente que individualmente sea más capaz de desempeñarla, esta no siempre es la mejor solución a causa de las dependencias que existen entre las tareas.

Para mayor entendimiento, vamos a ejemplificar esta situación. Situándonos en el dominio de la búsqueda de información, nuestro objetivo será encontrar una serie de agentes que, de manera conjunta, puedan responder a una consulta sobre libros. Particularmente, queremos saber las propiedades de “nombre”, “autor”, “resumen” y “editor” de un libro. Intuitivamente, dividiríamos la tarea en cuatro consultas más pequeñas y asignaríamos el mejor agente individualmente a cada una de ellas. Por ejemplo, dividiríamos la tarea en subtareas (nombre, autor, resumen y editor) y asignaríamos a cada subtarea el agente que sepa más libros

---

\* Written by **Can Kurtan** and **Pinar Yolum** and **Mehdi Dastani**

de esa subtarea. Así, nuestro equipo estaría formado por el agente que sabe más títulos, el que sabe más autores, el que sabe más resúmenes y el que sabe más editores. Sin embargo, al realizar una consulta nos damos cuenta de que la cantidad de libros que puedes buscar es muy limitada, pues los títulos que se sabe un agente, no coinciden con los editores, los resúmenes o los autores que tienen otros agentes. Esto son las dependencias de la tarea y lo que hace tan difícil nuestro objeto de estudio. De esta manera, en este trabajo, se va a afrontar cómo optimizar la formación de equipos.

Para conseguirlo, se hará uso de grafos expertos y algunas métricas como desempeño, privacidad, justicia, diversidad, etc. También se estudiarán diferentes algoritmos de formación de equipos y su evaluación.

Cabe destacar que para tratar este problema como un problema de optimización, vamos a realizar una serie de asunciones. La primera asunción será que la tarea puede ser subdividida en subtareas independientes. Con esta asunción se pretende establecer que la respuesta de un agente no depende de otros agentes, sólo de su capacidad para resolver esa subtarea. También asumiremos que podemos computar fácilmente la capacidad que tiene un agente para realizar una subtarea, y que un agente o puede o no puede realizar una subtarea, no hay medias tintas. Problemas que no cumplan con estas asunciones no van a ser tratados en este trabajo.

## 2 Trabajos relacionados

En esta sección se va a destacar trabajos relacionados con este, que se enfoquen en este problema de la formación de equipos. De este modo, primero vamos a comentar trabajos que los autores han destacado citándolos de forma directa y después comentaremos otros que están relacionados, pero no directamente citados.

En primer lugar tenemos el trabajo realizado por Kargar et al. [2], donde intentan resolver la formación de equipos como un problema de optimización con dos criterios: el coste personal y el de comunicación entre los miembros del equipo. Debido a que consideran el coste de comunicación entre agentes, su grafo representará en los nodos el coste del agente y en las aristas el coste de comunicación entre ese par de agentes. Para resolverlo proponen dos enfoques: el primero de ellos es un algoritmo de aproximación  $\alpha - \beta$  donde fijan uno de los dos valores a un máximo y luego buscan optimizar el otro. Su segundo enfoque se basa en utilizar un algoritmo que buscará encontrar todas las soluciones Pareto óptimas, es decir, que no sean dominadas por ninguna otra en ambos valores a optimizar.

A continuación vamos a comentar [3], donde formulan el problema de tal forma que consiguen añadir restricciones al problema como incluir un agente o conjunto de agentes, limitar el tamaño o coste del equipo e imponer cierta localización a la solución final, ya sea a nivel de distancia geográfica o familiaridad. Al

considerar soluciones existentes, comentan la problemática de algunos enfoques con respecto a su robustez, en concreto cuando la red social cambia, ya sea por inserciones o borrados de conexiones entre agentes. En contraposición, ellos proponen utilizar un algoritmo que obtendrá una solución a partir de resolver un problema equivalente relajado, añadiendo términos de penalización  $\gamma$  que relajan el problema y para los que se cumple que:  $\gamma \geq 0$ , obteniendo una solución para el problema no relajado en caso de valer 0.

En último lugar dentro de los trabajos comentados, tenemos el de Liemhetcharat y Veloso [4], donde se busca maximizar la sinergia entre agentes para resolver la tarea, utilizando un grafo de sinergia que marcará las capacidades de los agentes y su desempeño al trabajar codo con codo. Para resolver este problema, su propuesta es utilizar un algoritmo que va a aprender a partir de un grafo de sinergias observando las interacciones entre pequeños grupos de agentes.

A continuación vamos a comentar otros trabajos relacionados con el problema que estamos tratando, pero que no se citan de manera directa.

En primer lugar tenemos el trabajo de Anagnostopoulos et al. [5], donde se establece el problema de forma general, especificando toda la notación y el área de trabajo dentro de este problema de forma muy explicativa y desarrollada. Se comenta además diferentes funciones de cuantificación de la calidad de un equipo para una tarea concreta, algo que no se ha realizado en los trabajos previamente comentados. De este modo, comentan el problema del cubrimiento de tareas en ambientes offline y online. En el caso offline, proponen un algoritmo de aproximación basado en el algoritmo de redondeo aleatorio, conocido dentro del cubrimiento del conjunto. Cuando trabajan con tareas que se obtienen de forma online o a lo largo de un periodo de tiempo, proponen trabajar con diferentes heurísticas para asignar la tarea a los agentes en función de la carga actual de trabajo.

En el trabajo de Datta et al. [6], se comenta la formación de equipos y, de forma novedosa, se evalúa sobre un corpus obtenido a partir del web-crawling de Github. Uno de sus aportes a este problema y algo que consideran en su enfoque es la carga de trabajo de los agentes a nivel individual, impidiendo que este equipo formado ponga demasiada carga en alguno de sus agentes. Cabe destacar que en su trabajo, consideran que la tarea surge a partir de un usuario que quiere realizarla, considerándolo el líder del equipo y estando siempre presente en la solución.

Por último, tenemos [7], el trabajo más antiguo de todos los que hemos encontrado, en el que se hace una introducción al problema y se comenta técnicas que se utilizaba previamente como algoritmos genéticos o enfriamiento simulado. Sin embargo, Lappas et al. indican en su trabajo que son los primeros en utilizar de manera directa información del grafo social entre los agentes, una propuesta que ningún otro trabajo había realizado en el problema de formación de equipos. Este trabajo podría considerarse como el comienzo de esta línea de enfoques,

dado que todos los trabajos previamente citados comentan el enfoque que aquí se desarrolla de trabajar con el diámetro del grafo y el árbol de expansión mínima.

Una vez comentados todos estos trabajos relacionados con el problema de la formación de equipos, pasamos a desarrollar y detallar el trabajo realizado y los conceptos utilizados específicos al trabajo que estamos comentando, "An ideal team is more than a team of ideal agents".

### 3 Descripción

En este apartado se procede a explicar de manera profunda los contenidos del paper "An ideal team is more than a team of ideal agents" [1]. Para ello, y con el objetivo de facilitar la descripción, vamos a dividir esta sección en diferentes partes. Primero, se presentará la nomenclatura que se utilizará durante esta sección. A continuación se presentará el concepto de grafo experto y qué información podemos sacar de él, y finalmente se presentarán diferentes algoritmos para construir equipos.

#### 3.1 Nomenclatura

En este apartado simplemente se pretende destacar una serie de conceptos y sus correspondientes notaciones :

- Conjunto de agentes:  $A = \{a_1, a_2, \dots\}$ .
- Conjunto de capacidades  $C = \{c_1, c_2, \dots\}$ .
- Conjunto de capacidades del agente  $a = a_c \in 2^c$ .
- Una tarea  $T$  se define como un conjunto de capacidades necesarias  $T \in 2^c$ .
- Un equipo  $K$  se define como una asignación de tareas a agentes:  
 $K = \{\langle a, T \rangle | a \in A \ \& \ T \in 2^c\}$ .
- $\mathcal{K}$  representará el conjunto de todos los equipos posibles. Un equipo está definido correctamente si  $\forall \langle a, T \rangle \in K : T \subseteq a_c$ .
- El rendimiento de un agente respecto a una capacidad queda definido como  $P_{cap} : A \times C \rightarrow N$ .
- El rendimiento de un agente respecto a una tarea queda definido como  $P_{agent} : A \times 2^c \rightarrow N$ .
- El rendimiento de un equipo queda definido como  $P_{team} : \mathcal{K} \rightarrow N$ .

#### 3.2 Grafo experto

En el artículo mencionado se propone trabajar con grafos expertos con el fin de poder cuantificar correctamente la cooperación entre agentes al realizar sub tareas de manera conjunta.

Este grafo será un grafo no dirigido cuyos nodos representan pares agente-capacidad y las aristas cuantifican el grado de cooperación entre dos nodos. En este existirán tres tipos de aristas.

- Una aristas de tipo c (*c-edge*) es una arista entre nodos con el mismo agente y diferentes capacidades, indicando el grado de competencia del agente desempeñando estas subtareas.
- Una arista de tipo s (*s-edge*) es una arista entre nodos con diferente agente y la misma capacidad, indicando el grado de apoyo que pueden aportarse mutuamente al desarrollar esta subtarea.
- Una arista de tipo j (*j-edge*) es una arista entre nodos con distintos agentes y capacidades, indicando el grado de trabajo conjunto que pueden realizar para estas subtareas.

Para expresar de manera numérica el desempeño de un nodo o de una relación entre estos, utilizaremos la función pf:

$$pf(x) = \begin{cases} P_{cap}(a, c), & \text{si } x \text{ es un nodo} \\ P_{agent}(a_i, \{c_k, c_l\}), & \text{si } x \text{ es una arista tipo c} \\ P_{team}(\{\langle a_i, \{c_k\}\rangle, \langle a_j, \{c_k\}\rangle\}), & \text{si } x \text{ es una arista tipo s} \\ P_{team}(\{\langle a_i, \{c_k\}\rangle, \langle a_j, \{c_l\}\rangle\}), & \text{si } x \text{ es una arista tipo j} \end{cases}$$

Una vez claros todos estos conceptos, pasamos a definir ciertas métricas que podemos computar a partir del grafo experto:

- La **cooperación** de un nodo mide cómo de bien desempeñaría un agente una capacidad c en relación al resto de capacidades realizadas por otros agentes. Esta métrica se define como el sumatorio de las puntuaciones de los j-edges entre el número de j-edges concretos de ese nodo.

$$cpt(a, c) = \left( \sum_{e \in E_j} pf(e) \right) / |E_j| \quad (1)$$

- La **versatilidad** de un nodo mide cómo de bien desempeñaría un agente otras capacidades además de la capacidad c. Esta métrica se define como el sumatorio de las puntuaciones de los c-edges entre el número de c-edges concretos de ese nodo.

$$vst(a, c) = \left( \sum_{e \in E_c} pf(e) \right) / |E_c| \quad (2)$$

- La **centralidad** de un agente a mide cómo de conectado está un agente al resto del grafo. Para calcularla tenemos que tener en cuenta todos los nodos cuyo agente sea a. Esta métrica se utiliza para cuantificar cómo de bien trabaja este agente en equipos conformados por otros agentes. Esto se hace obteniendo como el sumatorio de aristas de los nodos entre el número de nodos.

$$dct(a) = \left( \sum_{v \in V} |edgesOf(v)| \right) / |V| \quad (3)$$

### 3.3 Algoritmos para construir equipos

Una vez presentados todos los conceptos previos necesarios, pasamos a introducir los dos algoritmos presentados en el artículo: *One-Shot Team Building* y *Iterative Team Building*.

**One-Shot Team Building** En este algoritmo, partimos de una tarea  $T$ , el grafo experto  $G$  y una métrica  $m$  y, de manera iterativa, vamos a ir añadiendo a nuestro equipo aquel agente que satisface una capacidad particular de la tarea maximizando una métrica. En otras palabras, por cada capacidad necesaria para realizar la tarea  $T$ , miraremos qué agentes son capaces de llevarla a cabo. Una vez los tengamos, seleccionaremos aquel que maximice una métrica en particular (por ejemplo, aquel cuyo nivel de cooperación sea mayor, puesto que de esta manera sabemos que este agente trabajará bien en equipo). Añadiremos al agente al equipo y seguiremos con la siguiente capacidad necesaria.

---

**Algorithm 1** buildTeam( $T, G, m$ )

---

**Input:** Task  $T$ , Expert Graph  $G$ , Metric  $m$   
**Output:** Team  $K$  in which metric  $m$  is maximized  
 $K \leftarrow \emptyset$   
**foreach**  $c \in T$  **do**  
     $U \leftarrow \text{capables}(c)$   
     $a \leftarrow \max(c, G, U, m)$   
     $K.add(a, \{c\})$   
**end for**  
**return**  $K$

---

Como ya hemos comentado, este algoritmo requiere apoyarse sobre una métrica  $m$ . Gracias a la información provista por el grafo experto, existen diferentes métricas sobre las que comparar a nuestros agentes: capacidad, cooperación y versatilidad. Estas se detallan a continuación:

- **HIP:** Cada capacidad es asignada al agente que tiene el mejor desempeño individual para esta capacidad. El equipo está compuesto de agentes cuyo desempeño es el mejor de manera individual.
- **HCPT:** Cada capacidad es asignada al agente que tiene mayor cooperatividad. El equipo está compuesto de agentes que tienen la mejor habilidad de cooperación con otros agentes.
- **HVST:** Cada capacidad es asignada al agente que tiene mayor versatilidad para esa capacidad. El equipo está compuesto por agentes que se desempeñan mejor en caso de que se les asigne múltiples capacidades.

De esta manera, dependiendo de la métrica utilizada, obtendremos equipos cuya composición se ha realizado teniendo en cuenta diferentes cualidades.

**Iterative Team Building** En este algoritmo partimos de un equipo inicial, sobre el que vamos a realizar modificaciones siguiendo un criterio particular.

Antes de explicar el algoritmo debemos introducir el concepto de desempeño grupal. El desempeño grupal del equipo es un cálculo no trivial que se acota obteniendo el desempeño máximo y mínimo de un equipo. Para ello se va a utilizar el grafo experto. Para computar el desempeño máximo vamos a considerar todos los pares de asignaciones de tareas a agentes del equipo. Con estos, se puede marcar el desempeño máximo de este equipo como el valor mínimo de las aristas que los conectan.

$$P_{max}(K) = \min\{pf(e) | e \in edges(K)\} \quad (4)$$

Por otro lado, para obtener el desempeño mínimo, vamos a emplear la siguiente fórmula:

$$P_{min}(K) = \max\{I(K, v) | v \in nodes(K)\} \quad (5)$$

donde vamos a utilizar la función *nodes* para obtener todos los nodos del equipo  $K$  y la función  $I$  para generar el desempeño mínimo de un nodo, que definiremos tal que:

$$I(K, v) = \sum_{e \in E} pf(e) - (|E| - 1)pf(v) \quad (6)$$

Una vez tenemos claro cómo computar el desempeño grupal máximo y mínimo, procedemos a presentar dos estrategias de formación de equipos: basada en rendimiento y basada en exploración.

*Formación de equipos basada en rendimiento* Pasamos a enumerar los pasos realizados por el algoritmo basado en rendimiento para facilitar la comprensión.

1. Partimos de un equipo  $K$  y un grafo experto  $G$ .
2. Guardamos una copia del equipo actual en  $K_o$ .
3. Obtenemos los valores de cooperación de los nodos del equipo y los guardamos en  $M$ .
4. Calculamos la media de cooperación del equipo.
5. Para cada nodo, obtenemos su cooperación, su agente  $a$ , y su capacidad. Si su cooperación está por debajo de la media previamente calculada, pasamos al paso 6. Si no queda ningún nodo por explorar pasamos a 9.
6. Obtenemos la lista de agentes capaces de realizar la tarea excluyendo al agente previamente seleccionado. Además, creamos un diccionario ordenado de manera descendente llamado *aux*.

7. Para cada agente de los que son capaces, si no pertenece al equipo y su cooperatividad con esa capacidad es mayor que la media, lo añadimos al diccionario aux, normalizando su valor. En caso de pertenecer al equipo, miraremos si su versatilidad es mayor a la media, y en caso de serlo, lo añadimos al diccionario aux normalizando su valor.
8. Si aux no está vacío, reemplazamos al agente  $a$  por el mejor agente del diccionario (el primero).
9. Si no hemos conseguido aumentar o  $P_{min}$  o  $P_{max}$ , paramos. En caso contrario, volvemos al paso 2.

El pseudocódigo del algoritmo se puede encontrar en el propio trabajo [1].

*Formación de equipos basada en exploración* El algoritmo previamente descrito no tiene en consideración ni los agentes que no se han explorado, ni aquellos que se están utilizando de manera reiterada. Por ello, se propone una variación de este algoritmo donde se le da más peso a la exploración, sin ignorar la calidad de los agentes seleccionados. Para ello, en vez de guardar su valor de cooperación o versatilidad, guardaremos las veces que se ha consultado el agente. De esta manera, seleccionaremos al menos consultado de entre los que mejoran la cooperatividad.

### 3.4 Otros aspectos a considerar en la formación de equipos

En la formación de equipos, tal como se destaca en [1], también se pueden considerar la equidad, la privacidad o la diversidad como potenciales cualidades.

El concepto de equidad se refiere a realizar la creación del equipo con el objetivo de que todos los agentes participen. Se basa en la idea de que todos los agentes deberían poder realizar subtareas y no “morirse de hambre”. Vamos a utilizar la inversa de la desviación típica como forma de calcular esta medida.

Por otro lado, la privacidad del equipo es una medida que hace referencia a la información que nuestro sistema potencialmente podría revelar. Se basa en el hecho de que algunas tareas podrían revelar información sobre el propietario que nos gustaría mantener privada. Así, el agente mejor conectado con el resto es el que tiene mayor riesgo de filtrar información. Esta medida se puede calcular con una formula que tiene en cuenta las conexiones de los nodos.

Finalmente, la diversidad es una medida que hace referencia a las diferentes asignaciones que se van a crear entre subtareas y las comunidades de los agentes que las van a realizar. Podemos utilizar algoritmos para identificar las diferentes comunidades del grafo experto y así repartir las tareas de manera variada. Cabe destacar que el mínimo de esta medida es 1, indicando que todas las asignaciones pertenecen a la misma comunidad.

Antes de finalizar, simplemente se quiere destacar que la búsqueda de alguna de estas cualidades en un equipo posiblemente sacrifique desempeño en la tarea, pues estaremos cambiando los agentes de un equipo en base a otros criterios.



## 4 Discusión

En el artículo revisado [1] se ha hecho un experimento artificial en el dominio de los libros con el fin de evaluar los diferentes algoritmos y métricas propuestos.

El experimento que se ha diseñado define la tarea como la capacidad de responder una pregunta relacionada con un total de 10 propiedades de libros. Se tienen un total de 52 agentes y 200 libros.

A estos agentes se les ha asignado capacidades con el fin de imitar el hecho de que pocos agentes son expertos en muchas capacidades y muchos agentes son expertos en pocas capacidades. Para ello, tendremos:

- Tres agentes con 10 capacidades.
- Cuatro agentes con 9 capacidades.
- Cinco agentes con 8 capacidades.
- Seis agentes con 7 capacidades.
- Siete agentes con 6 capacidades.
- Ocho agentes con 5 capacidades.
- Nueve agentes con 4 capacidades.
- Diez agentes con 3 capacidades.

Además de esto, también debemos asignar cuántos libros sabrá cada agente. Para ello, se ha asignado de manera aleatoria cuántos libros sabe entre 10 y 100. Si el agente se sabe más de 50, entonces los libros serán escogidos al azar. Por el contrario, si sabe menos, se le asignará un subconjunto preseleccionado de 50 libros. Así, conseguimos que existan agentes con mucha información general y agentes especialistas.

A continuación, se han simulado diferentes experimentos para sacar el grafo experto y todos los valores necesarios.

Una vez tenemos el grafo experto, podemos evaluar la creación de los equipos. Para ello se han hecho tareas de diferentes tamaños (de 3 a 7). Para cada tamaño, se han creado diez tareas diferentes que requieren de capacidades diferentes.

Una vez definida la forma en la que se recogen los resultados, se procede a crear equipos y evaluarlos. Para el algoritmo iterativo, como necesitamos un equipo inicial sobre el que mejorar, se inicializarán varios equipos asignando aleatoriamente a cada subtarea un agente capaz de llevarla a cabo.

Los resultados obtenidos podemos verlos en la Tabla 1. Cabe destacar que en este experimento entendemos como desempeño el número medio de libros devueltos.

| Size | One-Shot |      |      | Iterative             |                       |
|------|----------|------|------|-----------------------|-----------------------|
|      | HIP      | HCPT | HVST | Basado en rendimiento | Basado en exploración |
| 3    | 21.44    | 20   | 12.4 | 16.78                 | 12.88                 |
| 4    | 9.6      | 10.5 | 5.7  | 14.54                 | 6.82                  |
| 5    | 5        | 5.6  | 2.6  | 10.83                 | 3.79                  |
| 6    | 2.3      | 3.4  | 1.3  | 7.71                  | 2.21                  |
| 7    | 1.3      | 2    | 0.9  | 5.98                  | 1.41                  |

Table 1: Desempeño de los equipos para diferentes tallas de tarea

Como se puede observar, a medida que aumenta el tamaño de la tarea, el rendimiento es menor, porque la tarea es más compleja. También podemos ver que para tareas más pequeñas, la formación de equipos por one-shot funciona generalmente mejor (sobre todo HIP y HCPT), pero a medida que el tamaño del problema aumenta, es el algoritmo iterativo el que consigue mejores resultados (sobre todo el que está basado en rendimiento).

Al ver que el iterativo consigue generalmente mejores resultados (sobre todo en tareas más complejas), se explora la evolución del rendimiento del algoritmo iterativo basado en rendimiento en la Tabla 2. En la parte inicial, vemos el desempeño máximo, mínimo y exacto del equipo provisto inicialmente. En la parte Final, vemos los desempeños conseguidos después de aplicar el algoritmo iterativo. Podemos observar que se consigue mejorar sustancialmente los equipos. Fijándonos bien, también nos damos cuenta de que a medida de que un equipo es más grande, el desempeño mínimo y máximo comprende un mayor rango.

| Size | Inicial   |           |       | Final     |           |       |
|------|-----------|-----------|-------|-----------|-----------|-------|
|      | $P_{min}$ | $P_{max}$ | exact | $P_{min}$ | $P_{max}$ | exact |
| 3    | 2.54      | 10.37     | 6.14  | 12.64     | 22.24     | 16.78 |
| 4    | 0.5       | 8.29      | 2.80  | 10.55     | 22.75     | 14.54 |
| 5    | 0.06      | 6.35      | 1.21  | 6.54      | 20.43     | 10.83 |
| 6    | 0         | 5.45      | 0.57  | 2.88      | 18.73     | 7.71  |
| 7    | 0         | 4.60      | 0.27  | 1.53      | 17.79     | 5.98  |

Table 2: Evolución del desempeño de los equipos formados

Finalmente, se comparan el algoritmo iterativo basado en rendimiento con el de exploración en términos de equidad. Se espera que el de exploración incremente la equidad, al escoger aquellos agentes menos solicitados, mientras que el de rendimiento deje a aquellos menos productivos de lado. Se puede ver esta situación reflejada en la Figura 1.

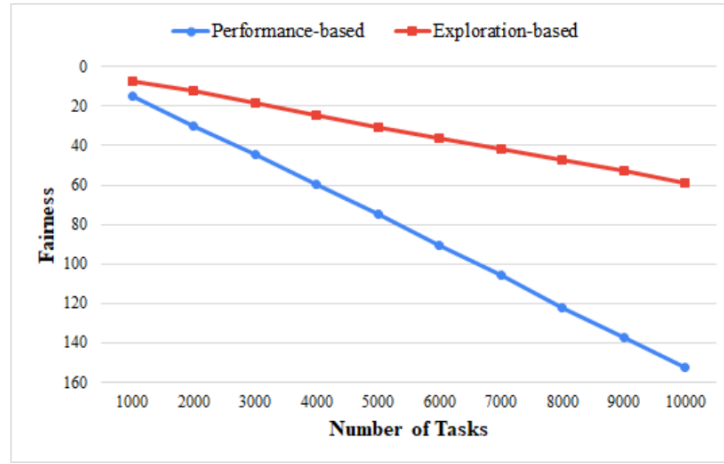


Fig. 1: Imagen obtenida del paper [1].

Finalmente destacar que pese a que los resultados del algoritmo iterativo basado en exploración no consigue resultados tan buenos como el que está basado en productividad, para tareas complejas conseguirá mayor equidad entre agentes.

## 5 Conclusiones

Dentro de las conclusiones de este trabajo queremos indicar de forma breve los aspectos clave del trabajo realizado.

En primer lugar, hemos realizado una introducción al problema de la formación de equipos, un problema complejo y abierto que es bastante interesante, con una gran cantidad de variantes y enfoques posibles a la hora de plantearlo.

Posteriormente, hemos revisado trabajos relacionados con el artículo que hemos estudiado, comprobando que otras opciones se habían explorado y comprobando que soluciones proponían otros expertos a este complejo problema.

A continuación, hemos explicado con bastante detalle y esmero el trabajo realizado en [1], así como los conceptos establecidos.

Finalmente, hemos establecido y discutido los resultados obtenidos en el artículo a estudiar, comprobando que enfoque obtenía los mejores resultados e indicando a qué se deben estos resultados obtenidos.

Para concluir este trabajo, nos gustaría destacar que este problema de la formación de equipos nos ha resultado realmente interesante, ya que tiene una gran variedad de aplicaciones a casos concretos del mundo actual. Además, nos resulta impactante el hecho de que siempre podremos mejorar la resolución del

problema si se busca considerar otros criterios. Consideramos que esto es bastante importante, pues conforme avanza la sociedad, los criterios considerados para la resolución de este problema pueden ir variando. Por ello, es de vital importancia que se explore la resolución de estos problemas con criterios más allá de la eficiencia, ya que en el mundo real no todo se basa en ésta.

## Referencias

- [1] Can Kurtan, Pinar Yolum, and Mehdi Dastani. “An Ideal Team Is More than a Team of Ideal Agents”. In: *ECAI*. 2020.
- [2] Mehdi Kargar, Morteza Zihayat, and Aijun An. “Finding Affordable and Collaborative Teams from a Network of Experts”. In: May 2013, pp. 587–595. ISBN: 978-1-61197-262-7. DOI: 10.1137/1.9781611972832.65.
- [3] Syama Sundar Rangapuram, Thomas Bühler, and Matthias Hein. *Towards Realistic Team Formation in Social Networks based on Densest Subgraphs*. 2015. arXiv: 1505.06661 [cs.SI].
- [4] Somchaya Liemhetcharat and Manuela Veloso. “Modeling and learning synergy for team formation with heterogeneous agents”. In: June 2012, pp. 365–374.
- [5] Aris Anagnostopoulos et al. “Power in Unity: Forming Teams in Large-Scale Community Systems”. In: Jan. 2010, pp. 599–608. DOI: 10.1145/1871437.1871515.
- [6] Samik Datta, Anirban Majumder, and KVM Naidu. “Capacitated Team Formation Problem on Social Networks”. In: *SIGKDD* (May 2012). DOI: 10.1145/2339530.2339690.
- [7] Theodoros Lappas, Kun Liu, and Evimaria Terzi. “Finding a team of experts in social networks”. In: June 2009, pp. 467–476. DOI: 10.1145/1557019.1557074.