

# Estrategias de planificación para datos y procesos en computación Grid: estado del arte

Data planning strategies and Grid computing processes: state of art

Francisco Arnaldo Vargas Bermúdez<sup>1</sup>, Carmen Inés Báez Pérez<sup>2</sup>

<sup>1</sup>Universidad de Boyacá. franciscovargasb@uniboyaca.edu.co.

<sup>2</sup>Universidad de Boyacá. cibaez@uniboyaca.edu.co.

 OPEN ACCESS



**Copyright:** © 2019 Ingenierías USBmed. La revista *Ingenierías USBmed* proporciona acceso abierto a todos sus contenidos bajo los términos de la licencia creative commons Atribución- no comercial- SinDerivar 4.0 Internacional (CC BY-NC-ND 4.0)

**Tipo de Artículo:** Revisión.

**Recibido:** 15-09-2018.

**Revisado:** 16-01-2019.

**Aprobado:** 30-01-2019

**Doi:** 10.21500/20275846.3805

**Referenciar así:** F. A., Vargas-Bermudez and C., Báez-Perez. "Estrategias de planificación para datos y procesos en computación Grid: estado del arte". *Ingenierías USBMed*, 10(1), pp.40-52, 2019.

**Declaración de disponibilidad de datos:** Todos los datos relevantes están dentro del artículo, así como los archivos de soporte de información.

**Conflicto de intereses:** los autores han declarado que no existen conflicto de intereses.

**Editor:** Yohana López Rivera, Universidad de San Buenaventura, Medellín, Colombia

**Resumen.** La distribución y la naturaleza compartida y heterogénea de la computación grid, hace que su objetivo de ofrecer aplicaciones con poder computacional colectivo, sea un gran reto. El objetivo es presentar el resultado de una investigación sistemática sobre aspectos teóricos de las temáticas que convergen en el diseño y desarrollo de planificadores de datos y procesos en computación grid. Se tomó como referente la metodología planteada para desarrollo de estados de arte, que cuenta con dos fases principales: heurística y hermenéutica. En el proceso se analizaron los fundamentos teóricos, tales como: la arquitectura, las etapas que conforman el proceso realizado por parte de un planificador grid y los tipos de planificación existentes, todo ello con el objeto de abordar los diferentes modelos y enfoques computacionales, heurísticos y metaheurísticos que permiten un funcionamiento eficiente y eficaz de manera óptima de los planificadores grid, permitiendo mitigar en cierto grado los problemas presentados en la planificación grid. La optimización de la planificación de recursos grid, es un área que se encuentra en desarrollo, dado que las problemáticas principales como: demora en la planificación y el proceso de optimización, están aún en etapa de desarrollo.

**Palabras Clave.** Computación grid; heurística; metaheurística; planificación grid; procesos; trabajos.

**Abstract.** The distribution, the shared and heterogeneous nature of grid computing, makes its objective of offering applications with a collective computational power, be a great challenge. The objective is to present the result of the systematic research about theoretical aspects about converge topics in the design and development of data and process planners in GRID computation. The methodology proposed for the development of art states was taken as a reference, which has two main phases: heuristics and hermeneutics. In this process, the theoretical foundations are analyzed like the architecture, the stages that conforms the process performed by a GRID planner and the existing planner types; all this with the aim of addressing the different computational, heuristic and metaheuristic models and approaches that allows the efficient and effective operation of GRID planners, allowing to mitigate in certain degree the problems presented in GRID planning. The optimization of grid resource planning is an area that is under development, given that the main problems such as: delay in planning and the optimization process, are still in the development stage.

**Keywords.** Grid computing, grid scheduling, heuristic, jobs, metaheuristics, processes.

## I. Introducción

El poder computacional de la grid busca aprovechar la interacción entre muchos recursos computacionales (físicos y lógicos) [1], con el fin de estandarizar la interoperabilidad de las tecnologías de información, requiriendo para esto, entre otros, de los planificadores grid [2], es por ello que paulatinamente se han diseñado éstos basados en estrategias que van desde el uso de técnicas manuales hasta llegar al empleo de la inteligencia artificial.

Todo ello implica heterogeneidad en dichos recursos y un nivel de dinámica bastante alto, lo cual imprime complejidad en la forma de gestionarlos [3], así mismo las tareas que deben ser desarrolladas por parte de ellos, hacen necesario el diseño de mecanismos que permitan automatizar el proceso de gestión, mecanismos tales como los planificadores grid; para lo cual se ha hecho uso de las bondades de la computación, de la heurística, de la metaheurística y de la inteligencia artificial entre otras [4]; obteniendo excelentes resultados.

Ofreciendo a todas las áreas de la ciencia el mecanismo para procesar grandes volúmenes de información y con ello permitir la consecución de resultados de forma eficiente y eficaz que apoyarán decisiones que tendrán gran impacto en la sociedad, el medio ambiente y en general en todo lo relacionado con la humanidad.

El proceso de planificación de la grid ha sido estudiado por autores y experimentado en proyectos de investigación donde se busca a través del análisis de técnicas, métodos y modelos, dar una solución óptima a la misma, lo cual sirvió de punto de partida para analizar dichas técnicas, métodos y modelos a través de este estado de arte, con el fin de tener una estructuración sólida que permita proponer nuevas alternativas de solución que aporten a la optimización de dicho proceso.

En este artículo se hace una breve descripción de conceptos tales como: planificación grid, en la cual se aborda el concepto y la arquitectura; seguido de modelos computacionales para la planificación grid, donde se describen los principales modelos que han sido implementados para la solución de problemas de planificación; en la tercera sección se encuentra una descripción de los métodos heurísticos y metaheurísticos para la planificación grid, con el fin de acercar al lector a los enfoques que se han desarrollado en esta temática. En las investigaciones realizadas en planificación grid, se describen algunas de ellas presentando el aporte más relevante para este artículo y por último se plantea un análisis de la temática y algunas conclusiones.

## II. Planificación Grid

El objeto principal de la computación grid [5] es el compartir recursos de diferentes tipos, tales como: computadores personales, supercomputadoras, clúster, ancho de banda de las redes, software y servicios entre otros; todos ellos con diferentes características, lo cual genera un alto nivel de

heterogeneidad [6]. Los grandes volúmenes de datos hacen necesario que dichos recursos sean integrados con el fin de mejorar las capacidades de almacenamiento y procesamiento para dar solución a los nuevos requerimientos del manejo y procesamiento de la información [7]. Es por ello que se comparten los recursos informáticos, con el objeto de permitir realizar tareas, tales como: procesamiento, almacenamiento, entre otras; que son prácticamente imposibles de desarrollar por una máquina o un pequeño grupo de máquinas.

Todos estos elementos y servicios gestionados por middlewares [8], tales como: Glite, Unicore, Gmarte [7, 9], Gridway [10], Condor [11, 12], proyecto que ha sido creado en la Universidad de Wisconsin y Globus Toolkit [13], el cual es considerado como el estándar de facto para esta tecnología.

Se puede hacer un paralelo entre lo que sucede en la vida cotidiana en múltiples procesos donde se requieren recursos que brinden atención a usuarios y/o trabajos, como por ejemplo en procesos industriales y lo que se presenta en la computación grid, donde se atienden tareas haciendo uso de recursos con el fin de ofrecer una solución al proceso masivo de datos, en esta se debe tener en cuenta especificaciones (por ejemplo: prioridad, tiempo de espera, disciplina de atención) de tareas, características y disponibilidad de los recursos.

Lo anterior hace necesario la utilización de mecanismos que permitan el desarrollo de este proceso de una forma óptima, que ofrezca la manera de cumplir con todos o la mayoría de los requerimientos de las tareas haciendo uso adecuado de los recursos disponibles, dicho mecanismo es conocido como planificación de tareas y administración de los recursos en la grid; es por ello que a continuación se hace una descripción de esta temática.

### A. Concepto

Son variados los conceptos que existen a cerca de la planificación, cada uno enfocado según la perspectiva del autor; con algo en común como son las tareas y los recursos, que son básicamente la razón de ser de dicho proceso.

**Definición 1.** La planificación consiste en encontrar una adecuada asignación entre jobs y recursos computacionales, considerando diferentes dominios administrativos y la heterogeneidad, dinamismo y control limitado que existe sobre los recursos [14].

**Definición 2.** La planificación es la que realiza la optimización de algunas características y parámetros relacionados con el sistema o el trabajo, tales como: rendimiento, sistema de utilización, tiempo de respuesta, tiempo de espera, tiempo de respuesta, la equidad, nivel de seguridad, disponibilidad [15].

**Definición 3.** La planificación grid consiste en averiguar cuáles recursos están disponibles en un momento determinado y hacer una selección de los mismos para ejecutar la aplicación o tarea [16].

Este concepto lógicamente debe estar en convergencia con un soporte que permita una estructura organizada, la

cual ofrece los mecanismos necesarios para hacer tangible la asignación de tareas y administración de recursos, siendo estos los pilares de la planificación en grid; es por ello que se presenta la arquitectura que permite el funcionamiento de este proceso.

A continuación, se hace una descripción del funcionamiento de la arquitectura.

### B. Arquitectura de la planificación Grid

Tal y como lo expresan [10, 12] la computación grid es un sistema conformado por elementos físicos y lógicos, es decir, es un sistema mayor, el cual cuenta con herramientas que permiten el buen funcionamiento de la misma, siendo los planificadores una de ellas.

Dado lo anterior, deben interactuar con múltiples elementos inmersos en el middleware grid, los cuales permitirán en conjunto con los planificadores realizar un proceso eficiente y eficaz en la atención de tareas y el uso de recursos; he aquí la explicación del porque en la arquitectura se observan múltiples componentes (servicios de información, servicios de monitoreo, servicios de seguridad y otros) externos a la planificación, pero que cumplen un papel importante para la ejecución de dicho proceso [17].

Los siguientes componentes han sido identificados como componentes básicos de la arquitectura de planificación grid [18].

**Servicio de planificación:** se encarga de las consultas de los recursos y calcula una planificación de trabajo(s) para los recursos disponibles, todo ello haciendo uso de los servicios grid de información.

**Servicio supervisor de trabajos:** realiza funciones para monitorear las ejecuciones del trabajo.

**Servicio de información:** maneja información estática y dinámica de los recursos, datos, estado de la red entre otros.

**Servicio de gestión de datos:** maneja la información sobre los datos.

**Servicio de gestión de la red:** ofrece información sobre las redes.

**Servicios de Contabilidad y facturación:** ofrece información sobre el presupuesto para las asignaciones de los recursos.

Los planificadores deben realizar una serie de pasos lógicos que permitirán la obtención de un resultado óptimo en el cual se logre cumplir los requerimientos del usuario haciendo uso eficiente y eficaz de los recursos ofrecidos por la grid; el proceso consta de etapas diseñadas para tal fin, las cuales se describirán a continuación.

### C. Etapas de la planificación Grid

El objetivo de las etapas es recolectar información de manera organizada, evaluar las variables que convergen en el proceso, toma de decisiones, cálculos, monitoreo, entre otras; todo ello necesario para el rendimiento adecuado de este proceso; a continuación, se enumeran y describe brevemente cada una de ellas [16, 19, 20].

#### 1. Preparación y recolección de información

Debido a la heterogeneidad de la grid y a la dinámica de las condiciones de los elementos en ella, como por ejemplo la entrada y salida de los recursos de la grid, las capacidades de procesamiento o de almacenamiento de acuerdo a la cantidad de tareas que se atienden y a otros aspectos; se hace necesario la creación de mecanismos efectivos (servicio grid de información) que permitan mantener actualizada dicha información, con el fin de brindar ésta lo más acertadamente posible y redundar en una buena planificación.

#### 2. Selección de recursos

En la grid convergen gran cantidad de recursos, pero se debe tener presente que no todos ellos se adaptan a los requerimientos de las tareas; es por ello, que la etapa anterior es muy importante y obviamente debe ser previa a ésta, puesto que basado en dicha información se realizará una adecuada selección del recurso o conjunto de recursos que serán candidatos a realizar las tareas pertinentes.

#### 3. Asignación de tareas

Teniendo seleccionados los recursos donde se ejecutará la tarea, se debe realizar a continuación el envío de éstas a los recursos, esta labor tiene dificultad técnica, pero para ello se ha venido trabajando en protocolos, tales como: Grid File Transfer Protocol (GridFTP) [12] y Grid Resource Allocation Management (GRAM) [17, 18, 21] que permiten mitigar esta dificultad.

#### 4. Monitoreo de ejecución de la tarea

Nada se lograría si se realiza la asignación y se olvida el proceso realizado en la ejecución; puesto que se pueden presentar muchas eventualidades que entorpecen la ejecución normal de las tareas, es por ello que se hace de relevante importancia realizar un seguimiento a la ejecución de éstas, con el fin de supervisar en primer lugar, el adecuado desarrollo de ellas y en segundo lugar, identificar posibles fallas de las tareas, en cuyo caso estas serán reprogramadas y asignadas a otros recursos, basado en las políticas del tipo de planificación empleada.

Anteriormente se hizo énfasis en que el proceso de planificación depende o es influenciado por las políticas inherentes al tipo de planificación que se utiliza de acuerdo al caso presentado, es por esto que a continuación se realizará una breve descripción de los tipos de planificación más relevantes.

### D. Tipos de planificación

A continuación, se describe brevemente los principales tipos de planificación grid existentes [19, 22].

#### 1. Flujos de trabajo grid

Existe un tipo de aplicaciones que no permiten bajo acoplamiento, puesto que algunas tareas requieren los resultados de otras para realizar su ejecución, a este tipo de aplicaciones en computación grid se les conoce como flujos

de trabajo, lo cual implica que los flujos de trabajo se ejecuten durante un periodo largo, aumentando las probabilidades de fracaso de uno de sus procesos y a su vez de todo el flujo, esto como consecuencia de la naturaleza dinámica de la grid [23].

De ahí la importancia de la planificación de los flujos de trabajo, en la cual se implementan mecanismos que permiten la ejecución de manera óptima, entre estos mecanismos encontramos tres componentes principales: la programación de flujo de trabajo, el movimiento de datos y la gestión de fallas [23].

## 2. Planificación centralizada, jerárquica y descentralizada

El control sobre los recursos y el conocimiento que se tiene de todo el sistema grid, son parámetros importantes que enmarcan la diferencia entre la planificación centralizada y la descentralizada; esto es importante puesto que los dos tipos de planificación son muy útiles en la computación grid [24].

Centralizada: En ella se presenta un mayor control sobre los recursos, debido a que el planificador tiene conocimiento de la situación de estos a través del sistema de monitoreo; es por ello que es más fácil obtener planificadores eficientes en este tipo de planificación.

Jerárquica: Permite coordinar diferentes planificadores a un cierto nivel. En este caso los planificadores en el nivel más bajo en la jerarquía tienen conocimiento de los recursos[25].  
Descentralizada o distribuida: No hay una entidad central de control de los recursos, la autonomía de los sitios grid hace que sea más difícil obtener planificadores eficientes.

## 3. Planificación adaptativa

La variabilidad en el tiempo del entorno de computación grid, es un aspecto que obliga al diseño de técnicas que permitan una adaptación de la planificación [26], teniendo en cuenta la situación actual de los recursos y las predicciones para su estado futuro, con el objetivo de detectar y evitar el deterioro del rendimiento. Por otro lado, la re-planificación también puede ser vista como una forma de adaptación de la planificación de los trabajos en ejecución, en el que se migran a recursos más adecuados.

Adicionalmente a las ya mencionadas, otros tipos de planificación son: data grids, planificación estática/dinámica [22], planificación inmediata/modo y planificación independiente.

Hasta este momento se ha realizado una descripción del concepto, de las estrategias y de los tipos de planificación grid, lo cual es la fundamentación teórica del tema, a continuación, se hará énfasis en los modelos computacionales que permiten realizar la planificación grid en la práctica, gráficos y tablas.

## III. Modelos computacionales para planificación en Grid

A continuación, se presentan algunos modelos de computación, importantes para la planificación en grid. Es de señalar que estos modelos tienen mucho en común con los modelos de cálculo para la planificación en entornos de computación distribuida.

### A. Modelo de cómputo para tiempo de espera – expected time to compute (ETC)

Las máquinas que hacen parte de la grid, trabajan en conjunto para ejecutar tareas que varían en sus requisitos computacionales; por ello se requiere la optimización del tiempo que estas deben esperar para ser atendidas por dichas máquinas. En [27], se define la importancia de tener conocimiento de la carga computacional de la tarea (por ejemplo, número de instrucciones), de la capacidad de cálculo de cada recurso (por ejemplo, en millones de instrucciones por segundo, Microprocessor without Interlocked Pipeline Stages-MIPS), y una estimación de la carga previa de los recursos.

Esta información permite calcular la matriz Expected Time to Compute ETC [28], cuya dimensión es número de tareas por el número de máquinas, donde cada posición  $ETC[t][m]$  indica el tiempo de espera para calcular la tarea  $t$  en el recurso  $m$ , el cual se supone que se conoce o bien se calcula en este modelo.

### B. Modelo de consumo de los ciclos totales del procesador - Consumption Model of the Total Cycles of the Processor (TPCC)

A pesar de sus interesantes propiedades, el modelo de ETC tiene una limitación importante, la capacidad de cálculo de los recursos se asume sin cambios durante el cálculo de tareas. Esta limitación se hace más evidente si se tiene en cuenta los sistemas grid en el que no sólo los recursos tienen diferentes capacidades de computación, sino también que podría cambiar con el tiempo debido a la sobrecarga del sistema de computación grid. La velocidad de cálculo de los recursos puede suponerse constante sólo por periodos cortos o muy cortos de tiempo. Con el fin de mitigar este problema se introdujo el modelo de consumo de los ciclos totales del procesador (TPCC) [16, 29].

Se ha propuesto un nuevo criterio en la planificación grid, el cual es el consumo de energía por un planificador [29]; he aquí el origen del TPCC, que representa el total de energía consumido por una aplicación al ejecutar las tareas, el consumo de ciclos totales del procesador se define como: el número total de instrucciones que los recursos grid podrían completar, desde el momento de inicio de la ejecución de la planificación hasta la hora de la finalización.

### C. Modelo de sistema de información Grid

Los modelos abordados hasta el momento, brindan una descripción detallada del problema y de la solución a este, sin embargo, éstos se basan en las predicciones, distribu-

ciones y simulaciones; lo cual no los hace muy atractivos para su aplicación a escenarios reales; es por ello, que actualmente se han diseñado otro tipo de modelos, como por ejemplo el de sistema de información grid (SIG), en el cual el planificador grid utiliza archivos de descripción para las tareas y/o aplicaciones y para los recursos [30], además de información acerca del estado de los recursos (uso de cpu, número de trabajos en ejecución, entre otros); todo ello proporcionado por el SIG. Seguido de un cálculo que busca la mejor correspondencia entre las tareas y los recursos, dicho cálculo es alimentado por información actualizada al día de la carga de trabajo de los recursos. Por ello, este modelo es mucho más útil en los entornos grid reales y es especialmente adecuado para aplicar heurísticas simples, tales como: primero que llegó - primero servido, primero las de prioridad alta, trabajo más corto - primero en ser atendido, entre otros.

#### D. Rendimiento del sistema Grid y optimización de criterios

Existen diferentes criterios que deben ser tenidos en cuenta en el momento de realizar una planificación en grid, la obtención de ellos permitirá un funcionamiento óptimo del sistema grid en general; estos criterios se pueden clasificar en dos grandes grupos de acuerdo al objetivo que se busca, ellos son: de rendimiento y de optimización; lo cual permite definir una planificación multi-objetivo que se aborda en el siguiente numeral.

Entre los criterios de rendimiento se incluyen: utilización de CPU de los recursos grid, balanceo de carga, uso del sistema, tiempo de cola, tiempo de respuesta, rendimiento acumulado, tiempo de espera, disponibilidad, escalabilidad, eficiencia de la planificación [31] entre otros; y entre los criterios de optimización se incluyen el Makespan Bassem[32], Mungwattana [33], tiempo de flujo, utilización de recursos, equilibrio de carga, tiempo de respuesta, tiempo total de realización ponderado, ponderado del número de trabajos tardíos, tiempo de respuesta ponderado, entre otros.

#### E. Enfoque de optimización multiobjetivo

De acuerdo a lo tratado anteriormente y debido a la necesidad de optimizar el conjunto de criterios que pueden ser tenidos en cuenta a la hora de planificar[33], se puede decir que la planificación grid es multi-objetivo [34] en su formulación general, es por ello que se busca una buena solución de compromiso entre los criterios cuando se consideren en conjunto. En [35] se observa el tratamiento que se da a criterios tales como: el tiempo de ejecución, el consumo de energía y el coste económico.

Con lo anterior se puede concluir que el problema es multi-criterio, lo cual crea la necesidad de diseñar soluciones de optimización multi-objetivo; entre estas soluciones se encuentran las jerárquicas y las simultáneas, las cuales se tratarán de forma breve a continuación.

#### 1. Enfoque jerárquico

La jerarquía implícitamente define prioridades, en este caso la prioridad es entre criterios. Por ejemplo, en computación de alto rendimiento se debe evaluar a que se le da mayor prioridad entre el makespan y el tiempo de respuesta, claro está que dicha prioridad puede ser modificada de acuerdo a las necesidades del usuario. Este enfoque ha sido considerado para la planificación de trabajo independiente bajo el modelo de ETC [36, 37].

#### 2. Enfoque simultáneo

En este enfoque, una planificación óptima es aquella en la que una mejora con respecto a un criterio y con respecto a otro se deteriora. Tratar con muchos criterios de optimización en conflicto, al mismo tiempo sin duda tiene un costo de computación alto [38], lo cual debe abordarse a través de la teoría de optimización de Pareto Liefoghe [39]. Sin embargo, en el problema de planificación grid, en lugar de conocer muchos puntos de Pareto en el espacio de la solución, se está interesado en conocer un planificador que tiene un equilibrio entre los criterios considerados y que realiza cálculos rápidamente.

Muchas clases de algoritmos metaheurísticos han sido desarrollados para la optimización multi-objetivo, entre ellos el Algoritmo Genético Priya Multi-objetivo (Multiobjective Optimization Genetic Algorithm - MOGA) [40].

### IV. Métodos Heurísticos y Meta Heurísticos para planificación en Grid

Se ha observado la complejidad que implica el desarrollo de un planificador grid, y también se ha realizado un estudio del funcionamiento de los modelos computacionales que permiten realizar a los planificadores un trabajo con el fin de optimizar el funcionamiento de éstos; pero también es claro que estos modelos no presentan el óptimo adecuado en la planificación; es por ello que se ha puesto a disposición de este problema las bondades de la heurística y la metaheurística.

A continuación, se describe brevemente los enfoques más importantes heurísticos y metaheurísticos y los beneficios de su uso para el problema de planificación grid [41, 42, 43].

#### A. Enfoque de heurística basada en búsqueda local

En este enfoque el proceso de búsqueda se construye con base en un conjunto de soluciones encontradas, haciendo uso de métodos diseñados para las búsquedas locales [27]. Algunos de éstos son: Hill Climbing (HC), Simulated Annealing (SA) and Tabu Search (TS), entre otros.

HC [27]: la optimización discreta se utiliza para modelar una serie de problemas del mundo real, dichos problemas se definen por un conjunto de soluciones (es decir, espacio de soluciones) y también por un valor de la función objetivo asignada a cada solución. Resolver este tipo de problemas consiste en encontrar soluciones que globalmente optimi-

cen la función objetivo dentro del espacio de soluciones, el objetivo de este algoritmo es resolverlas.

SA: es utilizado para resolución de problemas continuos de optimización y su característica principal es proporcionar un medio para descartar óptimos locales no acordes, permitiendo movimientos en HC con la esperanza de encontrar un óptimo global, es mejor que HC. Una breve historia de SA se presenta en [44, 45], las cuales incluyen una revisión de su aplicación a problemas de optimización discreta y continua.

TS: es más sofisticado y por lo general requiere más tiempo de cálculo para llegar a buenas soluciones. Sin embargo, sus mecanismos de listas tabú, los criterios de aspiración, la intensificación y diversificación pueden hacer que los algoritmos de búsqueda sean muy potentes [46]. En la obra de Liu [47], se puede observar el desarrollo de un algoritmo para explorar el agrupamiento adecuado de los conjuntos de datos.

Algunos de los métodos de búsqueda local para la planificación, basados en el enfoque de la vecindad son: movimiento de la base con búsqueda local [48], intercambio de base basado en búsqueda local [49] y por último se cita al método de Búsqueda Variable del Vecindario - Variable Neighborhood Search (VNS) [50].

## B. Heurística basada en la población

Estos métodos son óptimos para solucionar los problemas de optimización combinatoria, pero presentan una desventaja, la cual consiste en que requieren tiempos muy altos de funcionamiento para encontrar soluciones adecuadas. Así mismo, los mecanismos utilizados en estos métodos pueden ser empleados para aumentar la convergencia del método, cuando se trata de encontrar soluciones de buena calidad en un corto tiempo.

En esta familia se pueden distinguir 3 categorías de métodos basados en la población: Algoritmos evolutivos [51, 52] (algoritmos genéticos - AG, Memetic algorithms - MAS), optimización de colonia de hormigas (ACO) y optimización de enjambre de partículas (PSO).

A continuación, se hace una muy breve descripción de estos:

Algoritmos Genéticos Leal. Estos son bastante lentos debido al tiempo que emplean en cada iteración, teniendo en cuenta que el factor tiempo empleado para evaluar una solución es determinante [24]; pero para mitigar este problema se ha diseñado el algoritmo genético mejorado (improved genetic algorithm - IGA); aun así, son bastante utilizados en la solución del problema de planificación grid. Algunos autores han abordado esta problemática a través de trabajos tales como los descritos por Kamkar [53] y Qing [54].

Optimización de colonia de hormigas (Ant colony optimization - ACO). Este tipo de algoritmo es basado en el comportamiento de las colonias de hormigas y ha sido aplicado a problemas tales como asignación cuadrática, ruteo de vehículos, enrutamiento y muchos otros; es tan útil que ha sido aplicado a la optimización combinatoria para el tema

de planificación grid [15, 55].

Optimización de enjambre de partículas (Optimization of particle swarm - PSO). Es un algoritmo de búsqueda basado en la población y el comportamiento social de la congregación de aves y peces. El planificador tiene como objetivo reducir al mínimo makespan y el tiempo de flujo simultáneamente. Los estudios experimentales demuestran una gran eficiencia de éste [56, 57]. Según lo expuesto en [58], se han desarrollado algoritmos para minimizar el makespan pero estos no son la mejor solución para todos los ambientes grid.

## C. Enfoques heurísticos híbridos

Actualmente la hibridación de diferentes metaheurísticas está muy extendida, especialmente en lo que respecta a la utilización de métodos de búsqueda local dentro de los métodos basados en la población.

Se observa que los métodos de optimización cuentan con unas fortalezas y debilidades, al igual que los enfoques metaheurísticos; entonces porque no unir las fortalezas de estos con el objeto de lograr el diseño de un enfoque más adecuado para la solución del problema de planificación grid. Por ejemplo, se puede combinar las búsquedas evolutivas con búsquedas locales; de hecho, la hibridación entre diferentes metaheurísticos ha demostrado eficacia para muchos problemas difíciles de superar con un método único [59, 60, 61].

Tanto la computación evolutiva como la optimización de colonia de hormigas, a menudo hacen uso de los procedimientos de búsqueda local, para refinar las soluciones que se generan durante el proceso de búsqueda; este tipo de hibridación, es generalmente de mucho éxito, obteniéndose algoritmos de búsqueda muy potentes con la hibridación de: algoritmos de búsqueda tales como: Local Search (LS), simulated annealing (SA), tabu search (TS); con algoritmos evolutivos como genetic algorithms (GA), ant colonies (AC) [62]. Estos algoritmos se basan en fenómenos de la naturaleza no determinísticos [63].

A continuación, se enumeran algunas combinaciones que han aportado muy buenos resultados en lo que se refiere a la optimización de soluciones; se direcciona hacia la hibridación de heurísticos como GA, SA y TS, por ejemplo, GA+SA, GA+TS [50].

## D. Enfoques de hyper-heurística

Uno de los objetivos clave del enfoque híper- heurístico [64], es elevar el nivel de generalidad en la que los sistemas de optimización pueden funcionar. La híper- heurística producirá un aumento de los sistemas generales que son capaces de manejar una amplia gama de dominios de problemas, en lugar de la tecnología metaheurística que tiende a ser personalizada para un problema particular o una clase limitada de problemas. Las híper-heurísticas están interesadas en elegir de forma inteligente el algoritmo heurístico adecuado en una situación dada. Por supuesto, una híper-heurística puede ser

(a menudo lo es) una (meta) heurística y puede funcionar en (meta) heurística. Algunos trabajos desarrollados en este campo que demuestran buenos resultados en su aplicación a la planificación grid, son:

Kendall [65], presenta una búsqueda tabú basada en hiper-heurísticas para la solución de problemas de tiempos. El marco de la hiper-heurística, utiliza una lista tabú para supervisar el rendimiento de un conjunto de heurísticas de bajo nivel y luego hacer heurística tabú que ha sido aplicada muchas veces, lo que permite que otras heurísticas sean aplicadas.

Xhafa [66] presentó un hiper-heurístico simple para solucionar el problema, el cual utiliza como heurística subyacente de un conjunto ad hoc (de modo inmediato y por lotes) los métodos de planificación de trabajo de los recursos grid de acuerdo a las características de grid y de trabajo. El hiper-heurístico es un algoritmo de alto nivel, que examina el estado y las características del sistema grid, selecciona y aplica el método ad hoc que produce la mejor planificación de los trabajos [67, 68].

Así como los trabajos mencionados existen muchos otros, lo importante es resaltar que la hiper-heurística no busca solucionar el problema como tal, su objetivo es seleccionar la mejor heurística para ofrecer dicha solución, basándose en una serie de criterios, haciéndolo prácticamente de forma inteligente.

#### E. Enfoque planificación basada en economía

Al igual que en la economía o mercado normal, donde se emplean conceptos como: costos, ofrecimiento, presupuestos, cambio; este enfoque, busca trasladar toda esta temática al área de la planificación grid debido a que esta lo que realiza es una negociación, donde las tareas presentan las solicitudes de requerimientos para la utilización de recursos, por ejemplo, durante un tiempo determinado, con determinadas capacidades de cómputo o almacenamiento, y así, con muchas otras características que se han enumerado anteriormente [69].

Términos como las ofertas también pueden aparecer en esta área, se presentan negociaciones, tales como: que el usuario define qué quiere y cómo lo quiere y el proveedor ofrece lo que se adapte al usuario, el usuario verifica qué ofrece el proveedor y cuál es su costo y decide si adquiere el producto o no; todo esto se presenta en la planificación grid con enfoque de economía.

Esta similitud hace que aparezcan mecanismos que permitan dar el tratamiento de la economía normal, a los componentes que convergen en la planificación grid. En esta línea, se han desarrollado trabajos que permiten hacer tangible esta gran idea, y con ello lograr la eficiencia y eficacia de este proceso; algunos trabajos se enumerarán posteriormente.

La integración de la economía [70] de cómputo, como parte de un sistema de planificación, ofrece gran influencia en la manera como los recursos computacionales son seleccionados, para cumplir con los requisitos del usuario.

Algo complicado en este enfoque, es la fijación de los precios de los recursos, un trabajo enfocado en esta dirección es el grid service scheduling architecture based on economic model (GSAEM) [71]. También en [72], se presenta una descripción detallada de las fortalezas de los modelos económicos y una visión comparativa entre los diferentes modelos económicos, en computación grid.

#### V. Investigaciones realizadas en planificación para Grid

Como se ha expuesto anteriormente, se observa que la planificación grid es uno de los componentes con mayor impacto en la computación grid, y también se sabe que es de gran complejidad debido a la heterogeneidad de los recursos, y tareas que convergen en la grid y a la característica dinámica de la misma; es por ello que investigadores se han concentrado en el diseño y desarrollo de: algoritmos, mecanismos y trabajos donde se aplican éstos. A continuación, se realizará una descripción breve de algunos de estos trabajos realizados en esta área.

En [73], se presenta un enfoque novedoso basado en la optimización enjambre de partículas - particle swarm optimization (PSO), para la planificación de trabajos en computación grid. Las representaciones de la posición y la velocidad de las partículas en PSO se extienden desde los vectores reales a las matrices difusas. El enfoque propuesto, es el de generar de forma dinámica un planificador óptimo para ejecutar las tareas dentro de un período mínimo de tiempo, así como, la utilización de los recursos de manera eficiente.

Baghban & Rahmani [74], presentan un modelo y un algoritmo de planificación de tareas en entornos de computación grid. El algoritmo propuesto es evaluado en un ambiente grid simulado, con los patrones estadísticos de inserción de trabajos en el sistema siguiendo cada uno de ellos una distribución normal, de Poisson y exponencial. Los resultados muestran que el nuevo algoritmo propuesto, es eficiente en comparación con los resultados obtenidos a partir de otros algoritmos conocidos.

El modelo descentralizado presentado por Leal & Llorente [75] para la planificación de tareas independientes en grids federadas, consiste en un conjunto de metaplanificadores [76], en cada una de las infraestructuras grid que conforman la grid federada. Cada metaplanificador tiene que implementar una estrategia de asignación, con el fin de mejorar dos de las funciones objetivo más comunes de los problemas en la planificación de tareas: makespan y el rendimiento de los recursos.

El artículo presentado sobre planificación avanzada y reserva en un sistema grid [77], expone la realización de una planificación anticipada, reservando recursos por parte de las tareas, con lo cual a futuro pueden tener acceso simultáneo a recursos suficientes para su ejecución. Para lograr esto se introduce la noción de puesto de trabajo del usuario en un nodo de computación virtual, llamado vista lógica, el cual

posteriormente se asigna en el momento de la ejecución a un nodo de trabajo real, conocido como vista física.

En [78] se presenta un trabajo, cuyo objetivo es proporcionar pronósticos exactos, con la posibilidad de, cambiar dinámicamente las características de rendimiento de un conjunto distribuido de recursos en meta computación.

Ofreciendo un servicio ubicuo, que puede realizar seguimiento con cambios dinámicos y estables, para lo cual utiliza: técnicas de programación adaptativa, una arquitectura compatible con la extensibilidad y abstracciones internas que pueden ser implementadas de manera eficiente y portable.

También, se describe la implementación actual de la Red de Servicio de Clima - Climate Service Network (NWS) para Unix y TCP/IP; proporcionando ejemplos de monitoreo del rendimiento y capacidad de predicción.

En [79], se describe la implementación del Median Deviation Based Task Scheduling Algorithm (MDTS), el cual se enfoca en dos puntos de vista: clasificación de tareas y selección del procesador usando políticas basadas en inserción, para ello, se tiene en cuenta la heterogeneidad de las tareas y las máquinas para calcular los costos computacionales y de comunicaciones.

Trabajos realizados en otras áreas (cloud computing) que comparten la fundamentación de la grid computing, aportan elementos muy interesantes para el diseño de soluciones en el ámbito de la planificación grid. Uno de ellos, es el trabajo expuesto en [80], el cual busca estrategias que permitan solventar las limitantes que se encuentran en la planificación de trabajos a nivel computacional, en este se realiza una propuesta, partiendo de que los trabajos llegan a una nube híbrida estocásticamente, buscando minimizar el número de instancias computacionales.

Otras investigaciones en el área son: A Grid Task Scheduling Algorithm Based on QoS Priority Grouping [81].

A Job Scheduling Optimization Model based on Time Difference in Service Grid Environments [82], Research on an Effective Mechanism of Task-scheduling in Grid Environment [83], An Improved Genetic Algorithm with Limited Iteration for Grid Scheduling [84], An Intranet Grid Computing Tool for Optimizing Server Loads [85], Relaxed Dijkstra and A with linear complexity for robot path planning problems in large-scale grid environments [86], A survey of job scheduling, resource management in grid computing [87] and An improved task scheduling and load balancing algorithm under the heterogeneous cloud computing network [88].

Después de analizada la temática y los diferentes avances que se han realizado en el área, se observa que se podría trabajar para aportar a la optimización de este proceso, desde diferentes aspectos, tales como:

Análisis de la computación cuántica como propuesta de mejora a la planificación en computación grid.

Análisis comparativo de los mecanismos de planificación implementados en la computación grid y cloud computing con el objeto de establecer las ventajas competitivas en cada uno de ellos.

Análisis comparativo de los elementos de la economía, utilizados en el enfoque planificación basada en economía; frente a los nuevos lineamientos de la economía.

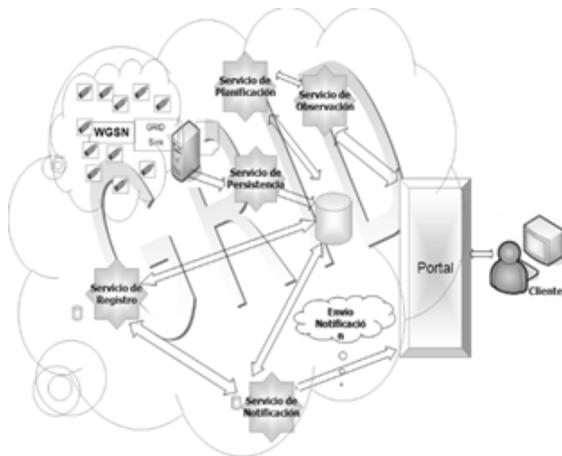
Los beneficios de estas estrategias de planificación, se observan en trabajos de investigación, en los cuales se generan grandes volúmenes de datos y por ende aumenta la necesidad de recursos para procesamiento de los mismos; lo cual es directamente proporcional con los trabajos que deben ser atendidos por la grid.

Prueba de ello, las investigaciones realizadas en el Grupo Internacional de Investigación en Informática, Comunicaciones y Gestión del Conocimiento (GICOGE) de la Universidad Distrital Francisco José de Caldas (Bogotá); en el marco del proyecto denominado “Modelo para el manejo de la información geográfica generada por una red inalámbrica de Geosensores”.

Este proyecto se basa en la propuesta del Open Geospatial Consortium (OGC) a través del Sensor Web Enablement (SWE), el cual define una solución que permite el desarrollo de diferentes actividades que ofrecen servicios a la sociedad, lo cual se logra a través de la inclusión de servicios que permiten realizar actividades tales como: verificación de datos y envío de notificaciones a usuarios, registro de usuarios y sensores; todas ellas realizadas por servicios web conocidos como: notificación, registro, observación y planificación [89].

La propuesta no cuenta con un servicio web, que ofrezca la funcionalidad de trasladar los datos recolectados por las redes inalámbricas de geosensores a un almacén de datos, como por ejemplo una base de datos; es por ello que este proyecto dentro de sus objetivos define la inclusión de un servicio grid llamado Servicio de Persistencia (Persistence Service, PS) [90] que permita almacenar las lecturas realizadas por la red inalámbrica de geosensores, en un recurso de datos ubicado en ambiente grid.

Siendo necesario para su desarrollo las siguientes investigaciones: Redes Inalámbricas de Geosensores Aplicadas en Sistemas de Observación y Monitoreo Ambiental [91], Plataforma de Computación Grid para Redes Inalámbricas de Geosensores [92], Planteamiento de un Modelo para los Servicios Grid de Notificación y Registro de Información Geográfica [93], Habilitamiento de la Web para Manejo de Información de Geosensores: Servicio de Observación de Sensores y Servicio de Planificación de Sensores. Una Mirada hacia Sensor Grid [94], De las Redes Inalámbricas de Geosensores a la Web de Sensores [95]; las cuales están enmarcadas en el modelo que se puede observar en la Figura 1.



**Figura 1.** Adaptación Modelo OGC. Fuente: [96]. Adaptado por: Grupo de Trabajo Geo-Sensores. Grupo de Investigación GICOGE. Universidad Distrital “Francisco José de Caldas”. 2008.

## VI. Consideraciones finales

Los modelos analizados en el capítulo dos, han sido diseñados partiendo de unas condiciones específicas, que deben ser tenidas en cuenta para poderlos aplicar, dado que su uso eficiente permitirá una mejor respuesta a los procesos de planificación, debido a que éstos generan carga adicional en cuanto a procesamiento y uso de recursos para la adecuada ejecución de los trabajos dentro de la grid.

Es importante, conocer en detalle las características particulares de los diferentes trabajos, que se van a ejecutar dentro de la grid, con el fin de buscar el modelo óptimo necesario que garantice la mejor eficiencia y eficacia en la ejecución de los mismos.

De acuerdo con lo expuesto hasta el momento, se puede analizar que los métodos heurísticos y metaheurísticos para la planificación en grid, se pueden aplicar de acuerdo con las características particulares de cada caso.

Cada uno de los enfoques tiene un objetivo específico, pero a medida que estos son más sofisticados, presentan como ventaja que se acercan más a una solución óptima del problema de planificación, lo cual genera como consecuencia mayor carga en el consumo de potencia de la grid y aumento del tiempo de respuesta.

Siendo la planificación un aspecto relevante dentro de la ejecución de una tarea, se han adelantado investigaciones que buscan mediante la aplicación de diferentes técnicas, la optimización de dicho proceso con el fin de dirigir la mayor cantidad de recursos hacia la ejecución de la tarea y no hacia la ejecución de la planificación.

## VII. Conclusiones

En la computación grid, convergen grandes temáticas que son necesarias en el ofrecimiento de una infraestructura adecuada para la prestación de servicios, tales como: procesamiento y almacenamiento. Una de esas temáticas es la planificación grid, absolutamente necesaria para que

todos los procesos relacionados con estos servicios, sean desarrollados de forma eficiente y eficaz.

Debido a que la planificación grid, es uno de los pilares fundamentales para la correcta y adecuada ejecución de trabajos en la grid, se han realizado investigaciones en las cuales se hace uso de modelos computacionales, heurística y metaheurísticas, obteniendo buenos resultados que se acercan a la solución de este problema, pero aún no se cuenta con la existencia de un planificador que permita realizar dicha labor de manera efectiva.

Los trabajos de investigación analizados en el desarrollo de este artículo, brindan bases y fundamentos para lograr obtener un planificador que cumpla dicha labor con un nivel de efectividad alto.

Como todos los procesos relacionados con información, la seguridad es un tema bastante importante en esta temática de la planificación grid; es por ello, que se recomienda tenerla presente en los trabajos de investigación que se realizan en esta dirección.

## Referencias

- [1] J. M. NICK, “The physiology of the grid: An open grid services architecture for distributed systems integration,” *DRAFT document*, 2002.
- [2] J. Yu and R. Buyya, “A taxonomy of workflow management systems for grid computing,” *Journal of Grid Computing*, vol. 3, no. 3-4, pp. 171–200, 2005.
- [3] M. Roffilli, V. Maniezzo, and M. Boschetti, “Grid-based services for optimized freight distribution,” in *2006 International Conference on Service Systems and Service Management*, vol. 1, pp. 558–563, IEEE, 2006.
- [4] S. Hutterer, A. Beham, M. Affenzeller, F. Auinger, and S. Wagner, “Software-enabled investigation in metaheuristic power grid optimization,” *IEEE Transactions on Industrial Informatics*, vol. 10, no. 1, pp. 364–372, 2014.
- [5] I. Foster, C. Kesselman, and S. Tuecke, “The anatomy of the grid: Enabling scalable virtual organizations,” *The International Journal of High Performance Computing Applications*, vol. 15, no. 3, pp. 200–222, 2001.
- [6] K. Yadav, D. Jindal, and R. Singh, “Job scheduling in grid computing,” 2013.
- [7] J. Taheri, A. Zomaya, and S. U. Khan, “Grid simulation tools for job scheduling and datafile replication,” 2013.
- [8] R. H. M. y A. Rodas VázquezU, “El grid: Una manera de aparecer en el mundo. entre ciencia e ingeniería,” 2009.
- [9] H. B. Prajapati and V. A. Shah, “Scheduling in grid computing environment,” in *2014 Fourth International Conference on Advanced Computing & Communication Technologies*, pp. 315–324, IEEE, 2014.
- [10] R. LÓPEZ HERRERO, “Aplicación de las tecnologías grid y de las arquitecturas orientadas a servicios en el análisis de estructuras de edificación,” 2007.

- [11] A. R. Butt, R. Zhang, and Y. C. Hu, "A self-organizing flock of condors," *Journal of parallel and distributed computing*, vol. 66, no. 1, pp. 145–161, 2003.
- [12] G. Moltó Martínez, *Computación de altas prestaciones sobre entornos grid en aplicaciones biomédicas: simulación de la actividad eléctrica cardiaca y diseño de proteínas*. PhD thesis, 2008.
- [13] I. Foster and C. Kesselman, "The globus project: A status report," in *Proceedings Seventh Heterogeneous Computing Workshop (HCW'98)*, pp. 4–18, IEEE, 1998.
- [14] N. B. Trejo, "Aplicación de multicast ipv6 seguro a servicios de información en entornos grid," 2008.
- [15] P. K. Tiwari and D. P. Vidyarthi, "Improved auto control ant colony optimization using lazy ant approach for grid scheduling problem," *Future Generation Computer Systems*, vol. 60, pp. 78–89, 2016.
- [16] Á. Fernández, E. Heymann, J. Salt, and M. A. Senar, "Servicios de asignación y planificación de recursos grid," *RedIRIS: boletín de la Red Nacional de I+ D RedIRIS*, no. 66, pp. 7–11, 2003.
- [17] A. Casan, *Arquitecturas GRID orientada a la gestión de recursos*. PhD thesis, 2004.
- [18] T. Çavdar and M. T. Kakız, "Threshold-based negotiation framework for grid resource allocation," *IET Communications*, vol. 11, no. 14, pp. 2236–2243, 2017.
- [19] K. Srikala and S. Ramachandram, "Fault tolerant scheduling of workflows in grid computing environment (ftsw)," in *2015 Global Conference on Communication Technologies (GCCT)*, pp. 343–347, IEEE, 2015.
- [20] M. B. Qureshi, M. M. Dehnavi, N. Min-Allah, M. S. Qureshi, H. Hussain, I. Rentifis, N. Tziritas, T. Loukopoulos, S. U. Khan, C.-Z. Xu, *et al.*, "Survey on grid resource allocation mechanisms," *Journal of Grid Computing*, vol. 12, no. 2, pp. 399–441, 2014.
- [21] K. Loheswaran, B. Navin, S. S. Devi, and S. Saranya, "Scheduling and resource management using pso in p-grid," in *2010 International Conference on Communication and Computational Intelligence (INCOCCI)*, pp. 399–403, IEEE, 2010.
- [22] E. Torres, *Técnicas de monitorización y diferenciación de servicios para la asignación de recursos en entornos de computación Grid, en base a indicadores de nivel de servicio*. PhD thesis, 2010.
- [23] U. Schwiegelshohn, R. M. Badia, M. Bubak, M. Danelutto, S. Dustdar, F. Gagliardi, A. Geiger, L. Hluchy, D. Kranzlmüller, E. Laure, *et al.*, "Perspectives on grid computing," *Future Generation Computer Systems*, vol. 26, no. 8, pp. 1104–1115, 2010.
- [24] K. Krauter, R. Buyya, and M. Maheswaran, "A taxonomy and survey of grid resource management systems for distributed computing," *Software: Practice and Experience*, vol. 32, no. 2, pp. 135–164, 2002.
- [25] K. Leal Algara, *Estrategias de planificación en infraestructuras grid federadas*. Universidad Complutense de Madrid, Servicio de Publicaciones, 2010.
- [26] L.-T. Lee, C.-H. Liang, and H.-Y. Chang, "An adaptive task scheduling system for grid computing," in *The Sixth IEEE International Conference on Computer and Information Technology (CIT'06)*, pp. 57–57, IEEE, 2006.
- [27] H. H. Hoos, K. Smyth, and T. Stützle, "Search space features underlying the performance of stochastic local search algorithms for max-sat," in *International Conference on Parallel Problem Solving from Nature*, pp. 51–60, Springer, 2004.
- [28] F. Xhafa and J. Kolodziej, "Game-theoretic, market and meta-heuristics approaches for modelling scheduling and resource allocation in grid systems," in *2010 International Conference on P2P, Parallel, Grid, Cloud and Internet Computing*, pp. 235–242, IEEE, 2010.
- [29] N. Fujimoto and K. Hagihara, "Near-optimal dynamic task scheduling of precedence constrained coarse-grained tasks onto a computational grid," in *null*, p. 80, IEEE, 2003.
- [30] E. Ambrosi, A. Ghiselli, and G. Taffoni, "Gdse: a new data source oriented computing element for grid," in *Proceedings of the 24th IASTED international conference on Parallel and distributed computing and networks*, pp. 53–57, ACTA Press, 2006.
- [31] Y. Zhu and L. Ni, "A survey on grid scheduling systems, technical report sjtu\_cs\_tr\_200309001," 2013.
- [32] B. Jarbouai, M. Eddaly, and P. Siarry, "A hybrid genetic algo-rithm for solving no-wait flowshop," *International Journal of Ad-vanced Manufacturing Technolgy*, vol. 11, pp. 472–478, 2010.
- [33] A. Mungwattana and K. Ploydanai, "Future makespan heuristic for job shop scheduling problem," in *The 40th International Conference on Computers & Industrial Engineering*, pp. 1–5, IEEE, 2010.
- [34] H. Viswanathan, E. K. Lee, and D. Pompili, "A multi-objective approach to real-time in-situ processing of mobile-application workflows," *IEEE Transactions on Parallel and Distributed Systems*, vol. 27, no. 11, pp. 3116–3130, 2016.
- [35] M. Arsuaga Ríos, "Optimización multiobjetivo para la planificación de trabajos en entornos de computación distribuida," 2015.
- [36] F. Xhafa, J. S. Carretero Casado, and A. Abraham, "Genetic algorithm based schedulers for grid computing systems," *International journal of innovative computing information and control*, vol. 3, no. 5, pp. 1053–1071, 2007.
- [37] F. Xhafa, J. A. Gonzalez, K. P. Dahal, and A. Abraham, "A ga (ts) hybrid algorithm for scheduling in computational grids," in *International Conference on Hybrid Artificial Intelligence Systems*, pp. 285–292, Springer, 2009.

- [38] E. N. Alkhanak, S. P. Lee, R. Rezaei, and R. M. Parizi, "Cost optimization approaches for scientific workflow scheduling in cloud and grid computing: A review, classifications, and open issues," *Journal of Systems and Software*, vol. 113, pp. 1–26, 2016.
- [39] A. Liefoghe, S. Mesmoudi, J. Humeau, L. Jourdan, and E.-G. Talbi, "A study on dominance-based local search approaches for multiobjective combinatorial optimization," in *International Workshop on Engineering Stochastic Local Search Algorithms*, pp. 120–124, Springer, 2009.
- [40] S. B. Priya, M. Prakash, and K. Dhawan, "Fault tolerance-genetic algorithm for grid task scheduling using check point," in *Sixth International Conference on Grid and Cooperative Computing (GCC 2007)*, pp. 676–680, IEEE, 2007.
- [41] M. Gendreau and J.-Y. Potvin, "Metaheuristics in combinatorial optimization," *Annals of Operations Research*, vol. 140, no. 1, pp. 189–213, 2005.
- [42] Z. Michalewicz and D. B. Fogel, *How to solve it: modern heuristics*. Springer Science & Business Media, 2013.
- [43] M. Rahman, R. Hassan, R. Ranjan, and R. Buyya, "Adaptive workflow scheduling for dynamic grid and cloud computing environment," *Concurrency and Computation: Practice and Experience*, vol. 25, no. 13, pp. 1816–1842, 2013.
- [44] D. Henderson, S. H. Jacobson, and A. W. Johnson, "The theory and practice of simulated annealing," in *Handbook of metaheuristics*, pp. 287–319, Springer, 2003.
- [45] F. J. Rodriguez-Diaz, C. Garcia-Martinez, and M. Lozano, "A ga-based multiple simulated annealing," in *IEEE Congress on Evolutionary Computation*, pp. 1–7, IEEE, 2010.
- [46] S. Meeran and M. Morshed, "A hybrid genetic tabu search algorithm for solving job shop scheduling problems: a case study," *Journal of intelligent manufacturing*, vol. 23, no. 4, pp. 1063–1078, 2012.
- [47] Y. Liu, Y. Liu, L. Wang, and K. Chen, "A hybrid tabu search based clustering algorithm," in *International Conference on Knowledge-Based and Intelligent Information and Engineering Systems*, pp. 186–192, Springer, 2005.
- [48] F. Angiulli and G. Folino, "A grid-based architecture for nearest neighbor based condensation of huge datasets," in *Proceedings of the third international workshop on Use of P2P, grid and agents for the development of content networks*, pp. 13–20, ACM, 2008.
- [49] F. Zamfirache, M. Frincu, and D. Zaharie, "Population-based metaheuristics for tasks scheduling in heterogeneous distributed systems," in *International Conference on Numerical Methods and Applications*, pp. 321–328, Springer, 2010.
- [50] A. Abraham, H. Liu, C. Grosan, and F. Xhafa, "Nature inspired meta-heuristics for grid scheduling: single and multi-objective optimization approaches," in *Metaheuristics for Scheduling in Distributed Computing Environments*, pp. 247–272, Springer, 2008.
- [51] S. Xue, C. Li, M. Yang, and J. Nie, "Grid resource scheduling based on improved differential evolution algorithms," in *2010 Sixth International Conference on Natural Computation*, vol. 8, pp. 4030–4034, IEEE, 2010.
- [52] Z. Pooranian, M. Shojafar, J. H. Abawajy, and M. Singhal, "Gloa: a new job scheduling algorithm for grid computing," *IJIMAI*, vol. 2, no. 1, pp. 59–64, 2013.
- [53] I. Kamkar, M. Poostchi, and M. R. A. Totonchi, "A cellular genetic algorithm for solving the vehicle routing problem with time windows," in *Soft Computing in Industrial Applications*, pp. 263–270, Springer, 2010.
- [54] Q.-w. FAN, P. WANG, H.-q. ZHANG, and X.-j. GAO, "Analysis of running mechanism of crossover operators in genetic algorithm [j]," *Journal of Beijing University of Technology*, vol. 10, 2010.
- [55] J. D. Bunton, A. T. Ernst, and M. Krishnamoorthy, "An integer programming based ant colony optimisation method for nurse rostering," in *2017 Federated Conference on Computer Science and Information Systems (FedCSIS)*, pp. 407–414, IEEE, 2017.
- [56] H. Izakian, B. T. Ladani, K. Zamanifar, and A. Abraham, "A novel particle swarm optimization approach for grid job scheduling," in *International Conference on Information Systems, Technology and Management*, pp. 100–109, Springer, 2009.
- [57] Z. Pooranian, M. Shojafar, J. H. Abawajy, and A. Abraham, "An efficient meta-heuristic algorithm for grid computing," *Journal of Combinatorial Optimization*, vol. 30, no. 3, pp. 413–434, 2015.
- [58] C. Jiang, C. Wang, X. Liu, and Y. Zhao, "A survey of job scheduling in grids," in *Advances in Data and Web Management*, pp. 419–427, Springer, 2007.
- [59] E.-G. Talbi, "A taxonomy of hybrid metaheuristics," *Journal of heuristics*, vol. 8, no. 5, pp. 541–564, 2002.
- [60] C. Blum, J. Puchinger, G. R. Raidl, and A. Roli, "Hybrid metaheuristics in combinatorial optimization: A survey," *Applied Soft Computing*, vol. 11, no. 6, pp. 4135–4151, 2011.
- [61] M. T. Younis and S. Yang, "Hybrid meta-heuristic algorithms for independent job scheduling in grid computing," *Applied Soft Computing*, vol. 72, pp. 498–517, 2018.
- [62] X. Sun, K. Zhang, M. Ma, and H. Su, "Multi-population ant colony algorithm for virtual machine deployment," *IEEE Access*, vol. 5, pp. 27014–27022, 2017.
- [63] F. Dong and S. G. Akl, "Scheduling algorithms for grid computing: State of the art and open problems," tech.

- rep., Technical report, 2006.
- [64] E. Burke, G. Kendall, J. Newall, E. Hart, P. Ross, and S. Schulenburg, "Hyper-heuristics: An emerging direction in modern search technology," in *Handbook of metaheuristics*, pp. 457–474, Springer, 2003.
- [65] G. Kendall and N. M. Hussin, "An investigation of a tabu-search-based hyper-heuristic for examination timetabling," in *Multidisciplinary Scheduling: Theory and Applications*, pp. 309–328, Springer, 2005.
- [66] F. Xhafa, "A hyper-heuristic for adaptive scheduling in computational grids," *Neural Network World*, vol. 17, no. 6, p. 639, 2007.
- [67] I. Chana *et al.*, "Bacterial foraging based hyper-heuristic for resource scheduling in grid computing," *Future Generation Computer Systems*, vol. 29, no. 3, pp. 751–762, 2013.
- [68] R. Aron, I. Chana, and A. Abraham, "A hyper-heuristic approach for resource provisioning-based scheduling in grid environment," *The Journal of Supercomputing*, vol. 71, no. 4, pp. 1427–1450, 2015.
- [69] E. Reihani, M. Motalleb, M. Thornton, and R. Ghorbani, "A novel approach using flexible scheduling and aggregation to optimize demand response in the developing interactive grid market architecture," *Applied energy*, vol. 183, pp. 445–455, 2016.
- [70] R. Buyya, D. Abramson, J. Giddy, and G. Nimrod, "An architecture for a resource management and scheduling system in a global computational grid," in *Proceedings 4th International Conference and Exhibition on High Performance Computing in Asia-Pacific Region (HPC ASIA 2000)*, 2002.
- [71] L. Zou, F.-a. Liu, and Y. Ma, "Grid service scheduling algorithm based on marginal principle," in *2008 Seventh International Conference on Grid and Cooperative Computing*, pp. 175–179, IEEE, 2008.
- [72] A. Haque, S. M. Alhashmi, and R. Parthiban, "A survey of economic models in grid computing," *Future Generation Computer Systems*, vol. 27, no. 8, pp. 1056–1069, 2011.
- [73] A. A. y. H. A. H. Liu, "Scheduling jobs on computational grids using a fuzzy particle swarm optimization algorithm," *Future Generation Computer Systems*, vol. 26, no. 8, pp. 1336–1343, 2009.
- [74] H. Baghban and A. M. Rahmani, "A heuristic on job scheduling in grid computing environment," in *2008 Seventh International Conference on Grid and Cooperative Computing*, pp. 141–146, IEEE, 2008.
- [75] K. Leal, E. Huedo, and I. M. Llorente, "A decentralized model for scheduling independent tasks in federated grids," *Future Generation Computer Systems*, vol. 25, no. 8, pp. 840–852, 2009.
- [76] J. C. Blanco Real *et al.*, "Agregación de infraestructuras computacionales usando técnicas de meta-planificación centradas en el usuario," 2017.
- [77] "Planificación anticipada y reserva en un sistema de cuadrícula," in *Conferencia Internacional sobre Tecnologías Digitales en Red*.
- [78] R. Wolski, N. T. Spring, and J. Hayes, "The network weather service: A distributed resource performance forecasting service for metacomputing," *Future Generation Computer Systems*, vol. 15, no. 5-6, pp. 757–768, 1999.
- [79] M. F. Akbar, E. U. Munir, M. M. Rafique, Z. Malik, S. U. Khan, and L. T. Yang, "List-based task scheduling for cloud computing," in *2016 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPS-Com) and IEEE Smart Data (SmartData)*, pp. 652–659, IEEE, 2016.
- [80] J. Zhu, X. Li, R. Ruiz, X. Xu, and Y. Zhang, "Scheduling stochastic multi-stage jobs on elastic computing services in hybrid clouds," in *2016 IEEE International Conference on Web Services (ICWS)*, pp. 678–681, IEEE, 2016.
- [81] F. Dong, J. Luo, L. Gao, and L. Ge, "A grid task scheduling algorithm based on qos priority grouping," in *2006 Fifth International Conference on Grid and Cooperative Computing (GCC'06)*, pp. 58–61, IEEE, 2006.
- [82] J. Liu, C. Zhou, J. Chen, H. Liu, and Y. Wen, "A job scheduling optimization model based on time difference in service grid environments," in *Sixth International Conference on Grid and Cooperative Computing (GCC 2007)*, pp. 283–287, IEEE, 2007.
- [83] H. Zhang, C. Wu, Q. Xiong, L. Wu, and G. Ye, "Research on an effective mechanism of task-scheduling in grid environment," in *2006 Fifth International Conference on Grid and Cooperative Computing (GCC'06)*, pp. 86–92, IEEE, 2006.
- [84] H. Yin, H. Wu, and J. Zhou, "An improved genetic algorithm with limited iteration for grid scheduling," in *Sixth International Conference on Grid and Cooperative Computing (GCC 2007)*, pp. 221–227, IEEE, 2007.
- [85] P. Lukasik and M. Sysel, "An intranet grid computing tool for optimizing server loads," in *Modern Trends and Techniques in Computer Science*, pp. 467–474, Springer, 2014.
- [86] A. Ammar, H. Bennaceur, I. Châari, A. Koubâa, and M. Alajlan, "Relaxed dijkstra and a\* with linear complexity for robot path planning problems in large-scale grid environments," *Soft Computing*, vol. 20, no. 10, pp. 4149–4171, 2016.
- [87] R. Sharma, V. K. Soni, M. K. Mishra, and P. Bhuyan, "A survey of job scheduling and resource management in grid computing," *world academy of science, engineering and technology*, vol. 64, pp. 461–466, 2010.
- [88] M.-L. Chiang, H.-C. Hsieh, W.-C. Tsai, and M.-C. Ke, "An improved task scheduling and load balancing algo-

- rithm under the heterogeneous cloud computing network,” in *2017 IEEE 8th International Conference on Awareness Science and Technology (iCAST)*, pp. 290–295, IEEE, 2017.
- [89] X. Chu and R. Buyya, “Service oriented sensor web,” in *Sensor networks and configuration*, pp. 51–74, Springer, 2007.
- [90] F. V. B. y J.N. Pérez., “Prototipo de modelo de persistencia para la variable temperatura irradiada, sensada por un geosensor para ser almacenada en ambiente grid,” pp. 117–124, 2008.
- [91] J. A. B. Velandia and J. N. P. Castillo, “Redes inalámbricas de geosensores aplicadas en sistemas de observación y monitoreo ambiental,” *Revista GTI*, vol. 11, no. 29, pp. 59–68, 2012.
- [92] J. A. B. Velandia and J. N. P. Castillo, “Plataforma de computación grid para redes inalámbricas de geosensores,” *Revista GTI*, vol. 11, no. 31, pp. 35–43, 2012.
- [93] C. I. B. Pérez and J. N. P. Castillo, “Planteamiento de un modelo para los servicios grid de notificación y registro de información geográfica,” *Revista GTI*, vol. 8, no. 20, pp. 13–22, 2009.
- [94] J. F. G. Estupiñan, J. N. P. Castillo, *et al.*, “Habilitamiento de la web para manejo de información de geosensores: servicio de observación de sensores y servicio de planificación de sensores. una mirada hacia sensor grid,” *Revista GTI*, vol. 10, no. 26, pp. 55–65, 2011.
- [95] C. I. B. Pérez and J. N. P. Castillo, “De las redes inalámbricas de geosensores a la web de sensores,” *Revista GTI*, vol. 6, no. 16, pp. 63–70, 2007.
- [96] M. Botts, A. Robin, J. Davidson, and I. Simonis, “Opengis sensor web enablement architecture document,” *Open Geospatial Consortium Inc, Tech. Rep.*, 2006.