

Ant Colony Optimization

Martín Pedemonte
Instituto de Computación, Facultad de Ingeniería
Universidad de la República
Montevideo, Uruguay
mpedemon@fing.edu.uy

Agosto de 2007

En los últimos años, la comunidad científica ha realizado una gran cantidad de propuestas de nuevas metaheurísticas que prometían resolver un amplio espectro de problemas de optimización del tipo NP. Sin embargo, en la práctica solamente un grupo pequeño de esas propuestas han logrado consolidarse, demostrando una amplia aplicabilidad sobre problemas de muy diversas características y adquiriendo la madurez necesaria como técnica de optimización para ser una alternativa real al momento de resolver un problema de optimización. Ant Colony Optimization (ACO) es una metaheurística sobre la que se ha trabajado ampliamente en los últimos 15 años. Se ha aplicado con éxito sobre varios de los problemas estándares de optimización demostrando su potencial.

El presente reporte es un relevamiento de las diversas variantes de ACO que han sido propuestas en estos 15 años. El eje central de este relevamiento es el estudio de las propuestas existentes para problemas estáticos de optimización combinatoria.

1. Introducción

No existe en la comunidad científica un consenso sobre la definición exacta del término “metaheurística”. Sin embargo, la más satisfactoria y comúnmente usada es considerarla como un conjunto de patrones empleados para definir métodos heurísticos que pueden ser aplicados a una amplia gama de problemas. Es posible interpretar a las metaheurísticas como un framework general de algoritmos que pueden ser aplicados a diferentes problemas de optimización con relativamente pocas modificaciones para su adaptación a las particularidades del problema considerado.

En los últimos años ha existido un aumento en el interés en la aplicación de metaheurísticas para la resolución de problemas de optimización combinatoria del tipo NP-difícil. Varias taxonomías para metaheurísticas han sido desarrolladas pero la más común distingue entre Métodos basados en Trayectoria (Trajectory Methods) y Métodos Poblacionales o basados en Población (Population-Based Methods) [7, 46]. La primera de dichas clases incluye todos los métodos que utilizan un mecanismo de exploración caracterizado por recorrer una trayectoria en el espacio de búsqueda, mientras que la segunda clase incluye todos los métodos que utilizan un conjunto de soluciones potenciales (la población) en lugar de explotar una única solución.

En el presente documento se aborda la propuesta de la metaheurística basada en población Ant Colony Optimization (ACO), presentándose una visión general del framework y abordando las principales variantes propuestas. El eje central de este relevamiento es el estudio de las propuestas existentes para abordar problemas estáticos de optimización combinatoria.

El resto del reporte se estructura de la siguiente forma. En las próximas dos secciones se presenta una primera aproximación al framework. Las secciones subsiguientes presentan un amplio repaso de las diversas variantes de algoritmos sobre este framework que han sido propuestas por la comunidad científica. Están organizadas intentando ser fieles al orden cronológico de aparición de las propuestas. Este enfoque permite explicar, en forma más sencilla, las mejoras que pretende incorporar cada una de las variantes. En cada una de las secciones, además de describirse las principales características de la idea propuesta, se presenta los parámetros que utilizan, sus valores recomendados, y una evaluación de la variante, a partir de la bibliografía relevada. En algunas secciones se realiza un agrupamiento de algoritmos que presentan la incorporación de una misma idea o concepto. En la sección 4, se presenta el algoritmo original y aquellas variantes que incorporan conceptos de elitismo. En las secciones 10 y 11 se introducen las variantes que utilizan una población auxiliar y las que utilizan una solución parcial para construir la solución respectivamente. Finalmente, la última sección presenta un resumen

de la evaluación de las variantes y las conclusiones de este reporte.

2. Antecedentes

El comportamiento de las colonias de hormigas ha sido fuente de inspiración para el diseño de múltiples algoritmos. A pesar de las limitadas capacidades de trabajo de cada individuo, la colonia es capaz de realizar tareas de gran complejidad a través de la coordinación entre los individuos. Una de las formas de comunicación que utilizan las hormigas para lograr tal coordinación de tareas es indirecta y consiste en realizar modificaciones en su ambiente¹. Debido a su escasa visión, esta comunicación se basa en la liberación de una sustancia química (feromona) que depositan para marcar rastros en el suelo.

ACO es una metaheurística basada en población formulada por Dorigo [24, 25, 29] que unifica en un framework común las diversas variantes propuestas sobre la utilización de hormigas artificiales para la solución de problemas de optimización. La primera propuesta que se enmarca dentro del framework de ACO fue realizada por el propio Dorigo [12, 23, 28] en el contexto de su tesis de doctorado y establece las características fundamentales de este tipo de técnicas:

1. Se utilizan agentes (las hormigas artificiales) para construir soluciones en forma incremental. Cada una de las hormigas construye una solución mediante la incorporación de componentes sobre una solución parcial en forma independiente.
2. Para la incorporación de los componentes se realiza una elección mediante una regla probabilística que tiene en cuenta la experiencia adquirida en etapas anteriores de la búsqueda e información heurística del problema que está siendo resuelto.
3. Para incorporar la experiencia adquirida en la construcción de soluciones se utiliza una matriz de feromona, a modo de memoria que almacena el rastro depositado por las hormigas en la construcción de soluciones de buena calidad.

Se han planteado variantes que incorporan este tipo de mecanismos para abordar tanto problemas estático como dinámicos [16, 17, 18, 19, 20]. Los estáticos se caracterizan por estar completamente definidos a priori, conociéndose todas sus características antes de su resolución y manteniéndose

¹A este concepto se llama *stigmergy* y se suele traducir al castellano como estigmergia.

inalteradas durante la resolución del mismo. En cambio los dinámicos presentan características que evolucionan mientras que el problema es resuelto, por lo cual algoritmos que aborden este tipo de problemas deben incorporar mecanismos que permitan la adaptación a esos cambios.

También existen variantes planteadas para los problemas de optimización multiobjetivo [4, 21, 22, 35, 37]. Los problemas de optimización multiobjetivo difieren de los tradicionales en que no se dispone de una única función a optimizar sino de varias y generalmente implican el balance entre intereses contrapuestos.

Si bien el planteo inicial se realizó para problemas de optimización combinatoria se han formulado extensiones para su aplicación a problemas de optimización de funciones con dominio continuo [5, 30, 56, 60].

3. Ant Colony Optimization (ACO)

Existen diferencias mínimas entre el framework propuesto según sea aplicado sobre un problema estático o dinámico. Por ser el objeto de este relevamiento la aplicación de la técnica sobre problemas estáticos solamente se presenta el esquema general para un algoritmo ACO aplicado a un problema de optimización combinatoria estático en el algoritmo 1 [29].

Algoritmo 1 ACO aplicado a un problema estático

```

SetParameters
InitializePheromoneTrails
while termination condition not met do
    ConstructAntsSolutions
    ApplyLocalSearch    %optional
    UpdatePheromones
end while
return best solution found

```

El procedimiento *ConstructAntsSolutions* utiliza una colonia de hormigas en forma concurrente y asíncrona para construir incrementalmente soluciones al problema de optimización que esté siendo considerado. Las decisiones para construir las soluciones son tomadas en forma estocástica a partir de información histórica sobre recorridas previas del espacio de búsqueda (el rastro de la feromona) e información heurística para asegurar la exploración de nuevas regiones del espacio de búsqueda.

El procedimiento *ApplyLocalSearch* es opcional y se utiliza para mejorar las soluciones encontradas mediante la utilización de una búsqueda local.

Evidentemente, este procedimiento es específico del problema atacado y por ser el eje de este estudio la mecánica de los algoritmos y no su aplicación a un problema en concreto, no se entrará en mayores detalles al respecto.

El procedimiento *UpdatePheromones* es el proceso por el cual se actualiza el rastro de feromona depositada. El valor de los rastros puede decrementarse por vía de la evaporación o incrementarse por el depósito en las componentes que las hormigas utilizaron para construir su solución. La evaporación sirve para evitar la convergencia prematura del algoritmo a regiones no óptimas, permitiendo que la componente heurística siga explorando el espacio de búsqueda. El depósito de feromona en una componente utilizada para construir una buena solución, la vuelve más atractiva para el futuro, aumentando su probabilidad de ser seleccionada.

Múltiples variantes han sido propuestas que instancian el esquema general descrito. En las próximas secciones se presentan las variantes más destacadas. La notación adoptada por gran parte de la bibliografía consultada para la presentación de las variantes está fuertemente influenciada por su aplicación sobre el Travelling Salesman Problem (TSP)². En este reporte se utiliza la notación presentada originalmente en [7] y posteriormente ampliada en [6] por considerarse más clara al ser completamente independiente del problema considerado.

Para la comprensión del resto del documento se presentan algunas convenciones de notación extraídas de [6]. Dada la función objetivo $f : D_1 \times \dots \times D_n \rightarrow \mathbb{R}$ del problema combinatorio a resolver, el conjunto de variables discretas X_i con valores $x_i \in D_i = \{d_1^i, \dots, d_{|D_i|}^i\}$, $i = 1, \dots, n$ y el conjunto de todas las asignaciones factibles $S = \{s = \{(X_1, x_1), \dots, (X_n, x_n)\} | x_i \in D_i, s \text{ satisface las restricciones del problema}\}$ se establece que:

- s es una solución al problema de optimización combinatoria.
- la componente c_{i,x_i} de una solución s es la combinación de una variable X_i con uno los valores de su dominio $x_i \in D_i$.

4. Ant System (AS) y variantes similares

La primera propuesta de este tipo de algoritmos fue realizada por Dorigo en 1991 [12, 23, 28]. Consistía en tres propuestas diferentes llamadas *ant-density*, *ant-quantity* y *ant-cycle*. Las dos primeras variantes cayeron rápidamente en desuso ya que en la práctica presentaban peores resultados, la principal diferencia que tiene con la tercera variante es en la forma en la cual

²Problema del viajante de comercio.

realizan la actualización de la feromona. Al no haberse extendido la utilización de las otras variantes, en general al referirse a AS se hace referencia a la variante *ant-cycle*.

Algoritmo 2 Ant System

```

InitializePheromoneValues( $T$ )
while termination condition not met do
  for all ants  $a \in A$  do
     $s_a = \text{ConstructAntSolution}(T, H)$ 
    ApplyOnlineDelayedPheromoneUpdate( $T, s_a | a \in A$ )
  end for
end while
return best solution found

```

En el algoritmo 2 se presenta el AS, A representa el conjunto de hormigas (colonia), T la matriz de feromona y s_a la solución construida por una hormiga $a \in A$ a partir de T y un criterio heurístico H . Para cada paso del bucle principal, se construyen soluciones para cada una de las hormigas que posteriormente serán utilizadas para actualizar el valor de la feromona.

En el procedimiento *InitializePheromoneValues* se inicializa el rastro de feromona para cada una de las posibles componentes de la solución. Si se utilizan valores muy grandes se precisarán varias iteraciones antes de que la evaporación reduzca lo suficiente su efecto para que la feromona depositada por las hormigas efectivamente guíe la búsqueda. En cambio si los valores son muy pequeños, la búsqueda estará fuertemente orientada por las primeras soluciones generadas. La experiencia indica que conviene inicializar este valor con un valor similar a la cantidad de feromona que será depositada por la colonia de hormigas en una iteración. Para esto es necesario conocer el costo de una solución al problema considerado. Típicamente se utiliza el costo asociado a una solución construida mediante alguna heurística auxiliar sencilla o simplemente la estimación del costo de una buena solución. El valor recomendado [29] para este parámetro es $\forall \tau_{i,x_i} \in T, \tau_{i,x_i} = \tau_{init} = \frac{m}{\text{Costo}(\text{SolucionAuxiliar})}$, siendo m la cantidad de hormigas y $\text{Costo}(\text{SolucionAuxiliar})$ es el costo de una solución construida utilizando alguna heurística auxiliar ³.

En el procedimiento *ConstructAntsSolutions* una hormiga construye incrementalmente una solución a partir de agregar componentes en una solución parcial s^p , inicialmente vacía. La elección de la próxima componente a agregar es realizada de acuerdo a la regla de transición de estados (state transition rule, también llamada random proportional rule):

³En el caso del TSP se suele construir una solución mediante la heurística nearest-neighbor.

$$p(c_{i,x_i} | s^p) = \begin{cases} \frac{[\tau_{i,x_i}]^\alpha [\eta_{i,x_i}]^\beta}{\sum_{c_{j,x_j} \in J(s^p)} [\tau_{j,x_j}]^\alpha [\eta_{j,x_j}]^\beta} & \text{si } c_{i,x_i} \in J(s^p) \\ 0 & \text{en otro caso} \end{cases} \quad (1)$$

En la ecuación 1 los parámetros β y α son utilizadas para ajustar la influencia relativa de la información heurística (η_{i,x_i}) y de los valores de feromona (τ_{i,x_i}) y $J(s^p)$ es el conjunto de componentes que pueden ser agregados a la solución parcial s^p . Típicamente se utiliza $\eta_{i,x_i} = \frac{1}{\text{Costo}(c_{i,x_i})}$.

Si se anula el parámetro α , solamente se considera el componente heurístico, por lo cual se podría interpretar el algoritmo resultante como un algoritmo greedy aleatorio con múltiples puntos de arranque. Si se anula el parámetro β , rápidamente se observa la aparición del fenómeno conocido como estancamiento que consiste en la situación en la cual todas las hormigas siguen exactamente el mismo camino y construyen exactamente la misma solución. El mismo comportamiento se suele observar si se trabaja con valores de α mayores a 1 [28].

Una vez que todas las hormigas han construido sus soluciones, se realiza la actualización de la feromona mediante la aplicación de la regla de actualización conocida como online delayed pheromone update rule:

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \sum_{a \in A} \Delta \tau_{i,x_i}^{s_a} \quad \forall \tau_{i,x_i} \in T \quad (2)$$

$$\text{siendo } \Delta \tau_{i,x_i}^{s_a} = \begin{cases} F(s_a) & \text{si } c_{i,x_i} \in s_a \\ 0 & \text{en otro caso} \end{cases} \quad (3)$$

El objetivo de la regla es aumentar la cantidad de feromona en los componentes de la solución que han sido utilizados por las soluciones de calidad alta.

En la ecuación 2, ρ es la tasa de evaporación de feromona ($0 \leq \rho \leq 1$). En la ecuación 3, $F : S \mapsto \mathfrak{R}^+$ es una función que indica la calidad de la solución considerada (función de calidad). Típicamente se utiliza $F(s_a) = \frac{1}{\text{Costo}(s_a)}$ [29]. Originalmente se utilizaba $F(s_a) = \frac{Q}{\text{Costo}(s_a)}$, siendo Q una constante. Sin embargo se ha comprobado que el valor de Q no es significativo [28], por lo cual resulta natural considerarlo como 1, eliminando un parámetro.

Ant-density y *ant-quantity* se diferencian de la variante anterior en que cada una de las hormigas deposita la feromona al utilizar cada componente, en lugar de realizarlo cuando ya han completado la solución. La variante *ant-density* utiliza la regla de la ecuación 4, mientras que la variante *ant-quantity* utiliza la regla de la ecuación 5. Como se indicó anteriormente ambas propuestas cayeron rápidamente en desuso.

$$\Delta\tau_{i,x_i}^a = \begin{cases} Q & \text{si la hormiga } a \text{ agrega } c_{i,x_i} \text{ en ese paso} \\ 0 & \text{en otro caso} \end{cases} \quad (4)$$

$$\Delta\tau_{i,x_i}^a = \begin{cases} \frac{Q}{\text{Costo}(c_{i,x_i})} & \text{si la hormiga } a \text{ agrega } c_{i,x_i} \text{ en ese paso} \\ 0 & \text{en otro caso} \end{cases} \quad (5)$$

Los parámetros que son utilizados por esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el TSP [28] se muestran en la tabla 1. En el caso del tamaño de la población se recomienda que sea igual a la cantidad de ciudades (n) del TSP considerado y con una distribución uniforme o aleatoria del lugar de inicio de las hormigas.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	n
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{m}{\text{Costo}(\text{Solucion.Auxiliar})}$
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	$2 \leq \beta \leq 5$
ρ	Tasa de evaporación de la feromona	0,5

Tabla 1: Parámetros para el AS

Existen dos propuestas posteriores que intentan mejorar los resultados obtenidos por AS y que presentan grandes similitudes con esta variante y que por tanto serán comentadas en esta misma sección, Elitist Ant System (EAS) y Rank-Based Ant System (AS_{rank}).

4.1. Elitist Ant System (EAS)

La idea de la variante consiste en darle mayor énfasis al mejor camino encontrado ponderándolo en forma particular al momento de realizar la actualización de los rastros de feromona [28]. Podría verse como si el mejor camino hubiera sido recorrido por un cierto número de hormigas, las que se dan en llamar hormigas elitistas.

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \sum_{a \in A} \Delta\tau_{i,x_i}^{sa} + e * \Delta\tau_{i,x_i}^{sbs} \quad \forall \tau_{i,x_i} \in T \quad (6)$$

Siendo s_{bs} la mejor solución encontrada hasta el momento (best-so-far) y e la cantidad de hormigas elitistas utilizadas. Se mantiene de la variante AS la regla para calcular la cantidad de feromona a depositar (ver ecuación 3).

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	n
e	Cantidad de hormigas elitistas	n
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{e+m}{\rho * Costo(SolucionAuxiliar)}$
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	$2 \leq \beta \leq 5$
ρ	Tasa de evaporación de la feromona	0,5

Tabla 2: Parámetros para el EAS

Los parámetros que son utilizados por esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el TSP [29] se muestran en la tabla 2.

Se ha propuesto sustituir la utilización de la mejor solución hasta el momento por la de la mejor solución generada por cada una de las hormigas hasta el momento [41, 61]. Esta propuesta se conoce como Ant System Local Best Tour (AS-LBT) y solamente ha obtenido leves mejorías con respecto a EAS al ser aplicado sobre el TSP. Los autores señalan como una virtud de su propuesta que puede funcionar en forma completamente distribuida, ya que evita realizar comparaciones entre las soluciones generadas por hormigas diferentes. Esta virtud es relativa porque no se propone ningún mecanismo particular para trabajar sobre la matriz de feromona, con lo cual sigue siendo necesario manejar en forma centralizada dicha matriz.

También se han realizado propuestas posteriores para aumentar la diversidad, como por ejemplo la realizada por Fidanova [31]. Dicha propuesta consiste en incrementar la diversidad mediante un refuerzo adicional de feromona sobre las componentes que no se hayan utilizado para construir soluciones en la iteración. La idea es volver atractivas componentes que no forman parte ni de las buenas, ni de las malas soluciones. Conviene señalar que el mecanismo habitual de depósito de feromona asegura que las componentes pertenecientes a las soluciones de buena calidad se volverán más atractivas y que las pertenecientes a las soluciones de mala calidad se volverán menos atractivas.

4.2. Rank-Based Ant System (AS_{rank})

La idea de incorporar elitismo tiene similitudes con la que está presente en los algoritmos genéticos y potencialmente tiene los mismos inconvenientes. Un posible inconveniente es que se otorgue un excesivo énfasis a la explotación, con lo cual se privilegian demasiado las regiones del espacio de búsqueda próximas a la mejor solución. Otro aspecto negativo de las propuestas ya comentadas es que la distribución de la feromona es realizada por todas las hormigas que construyen soluciones, con lo cual es posible utilizar información de soluciones de muy baja calidad.

Intentado paliar estos aspectos negativos Bullnheimer, Hartl y Straub proponen en 1997 [9, 10, 11] la variante que se dio en llamar Rank-Based Ant System AS_{rank} ⁴. La idea consiste en ordenar las soluciones construidas por las hormigas de acuerdo a su costo y ponderar su contribución en la actualización del rastro de acuerdo a la posición μ que ocupe cada solución en el ranking. Adicionalmente, únicamente las mejores $\omega - 1$ soluciones son tenidas en cuenta para dicha actualización. Se mantiene el elitismo de la propuesta anterior, asignándole el peso máximo ω . Se considera como el factor de ponderación mínimo el 1, por lo cual se deduce que el peso para la μ -ésima hormiga es $\omega - \mu$, con lo cual resulta:

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \sum_{\mu=1}^{\omega-1} (\omega - \mu) \Delta\tau_{i,x_i}^{s^\mu} + \omega * \Delta\tau_{i,x_i}^{s^{bs}} \quad \forall \tau_{i,x_i} \in T \quad (7)$$

Se mantiene de la variante AS la regla para calcular la cantidad de feromona a depositar (ver ecuación 3).

Los parámetros que son utilizados por esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el TSP [29] se muestran en la tabla 3. Conviene destacar que partiendo de las premisas que se utilizan solamente se debe considerar un parámetro pudiendo ser ω , μ o e , ya que el resto quedan determinados a partir de él (como se aprecia en la ecuación 7).

En [9] se realizan los experimentos utilizando $\rho = 0,5$. El valor recomendado para τ_{init} en [29] no es correcto, ya que μ es el índice de la sumatoria⁵. El valor correcto debería ser $\frac{0,5 * \omega * (\omega + 1)}{\rho * Costo(SolucionAuxiliar)}$.

La incorporación de un mecanismo que permite aumentar la diversidad en la generación de las soluciones ha mostrado alguna mejora sobre los resultados obtenidos por el AS_{rank} sobre el TSP [47, 48]. El mecanismo propuesto

⁴El nombre original de la variante fue Ant system with elitist strategy and ranking.

⁵Consultar páginas 71 y 74 en [29]. Hay una leve diferencia de notación debido a que en este reporte se adopta la del artículo original, siendo $\mu = r$ y $\omega = e$.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	n
ω	Cantidad de hormigas que depositan feromona	6
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{0,5*\mu*(\mu-1)}{\rho*Costo(SolucionAuxiliar)}$
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	$2 \leq \beta \leq 5$
ρ	Tasa de evaporación de la feromona	0,1

Tabla 3: Parámetros para el AS_{rank}

consiste en permitir que la siguiente componente sea seleccionada aleatoriamente entre las componentes disponibles en forma equiprobable. Se incorpora un parámetro r a partir del cual y en forma aleatoria se determina si para la próxima componente, se debe utilizar la regla de transición de estados (ver ecuación 1) o la nueva regla de selección aleatoria.

4.3. Evaluación

Las variantes presentadas en esta sección han sido ampliamente utilizadas para resolver el TSP, Asymmetric Travelling Salesman Problem (ATSP), Quadratic Assignment Problem (QAP) y Job-Shop Scheduling Problem (JSSP) entre otros problemas, lo cual muestra su versatilidad como metaheurística.

En general se ha utilizado como problema de validación de las propuestas el TSP obteniéndose excelentes resultados para instancias pequeñas de los problemas, inclusive aventajando a metaheurísticas ampliamente usadas como Simulated Annealing (SA), Tabu Search (TS) y Genetic Algorithms (GA). Al considerarse instancia más grandes, es necesario incluir mecanismos como el ranking y la utilización de operadores de búsqueda local⁶ para mantener su potencia. En general los resultados de AS_{rank} aventajan a los de EAS, que a su vez son mejores que los de la variante AS.

Finalmente, conviene destacar que está disponible el código fuente de las variantes AS, EAS y AS_{rank} en [1].

⁶Típicamente 3-opt para el TSP.

5. $\mathcal{MAX} - \mathcal{MIN}$ Ant System (\mathcal{MMAS})

La buena calidad de los resultados obtenidos al aplicar AS a instancias pequeñas de un problema no escalaba al incrementarse las dimensiones de las instancias para el problema considerado. En 1996, Stützle y Hoos formulan la propuesta $\mathcal{MAX} - \mathcal{MIN}$ Ant System (\mathcal{MMAS}) [49, 50, 51, 52] que pretende solucionar algunos aspectos no considerados en AS. La idea fundamental detrás de la propuesta es lograr una mayor explotación de las mejores soluciones encontradas, pero incorporando a su vez mecanismos que aseguren que no se llegue a un estado de estancamiento en etapas tempranas de la búsqueda.

Las principales características de la propuesta son las siguientes:

1. **Actualización de la feromona:** Solamente una hormiga por iteración actualiza el rastro de feromona. Existen dos posibilidades para la actualización del rastro, utilizar la mejor solución de la iteración o la mejor solución hasta el momento.

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \Delta\tau_{i,x_i}^{s_{best}} \quad \forall \tau_{i,x_i} \in T \quad (8)$$

Se mantiene de la variante AS la regla para calcular la cantidad de feromona a depositar (ver ecuación 3). Considerando s_{best} como la mejor solución de la iteración o la mejor solución hasta el momento. En general, la idea de utilizar exclusivamente la mejor solución hasta el momento presenta peores resultados, pero se considera conveniente utilizar en algunas iteraciones la mejor solución de la iteración y en otras la mejor hasta el momento [49].

2. **Límites en el rastro de feromona:** Se incorporan límites explícitos para la cantidad de feromona que puede haber en una componente $\tau_{min} \leq \tau_{i,x_i} \leq \tau_{max}$. Después de cada iteración se debe asegurar que el rastro de la feromona quede comprendido en el umbral, por lo cual si $\tau_{i,x_i} > \tau_{max}$ se debe realizar el ajuste $\tau_{i,x_i} = \tau_{max}$. Del mismo modo, si $\tau_{i,x_i} < \tau_{min}$ se debe realizar el ajuste $\tau_{i,x_i} = \tau_{min}$. Si adicionalmente se imponen las condiciones naturales $\tau_{min} > 0$ y $\eta_{i,x_i} < \infty$, la probabilidad de elegir un componente específico siempre es > 0 .

Estimación de τ_{max}

El valor máximo se calcula adaptando el mayor valor que teóricamente podría tener $\tau_{max}^{teo} = \frac{1}{\rho * Costo(s_{opt})}$. En la práctica, como evidentemente no se dispone de $Costo(s_{opt})$, se utiliza el valor correspondiente a la mejor

solución encontrada hasta el momento $Costo(s_{bs})$. Por lo tanto, para cada iteración en la que se obtenga una mejor solución se debe realizar la actualización de τ_{max} .

Estimación de τ_{min}

La estimación de τ_{min} es bastante más compleja. En [52] se presenta el desarrollo completo de dicha estimación. A continuación se presenta un breve resumen de los conceptos esenciales presentes en el mencionado artículo para lograr la comprensión de la ecuación resultante (9).

Se define la convergencia de \mathcal{MMAS} como la distribución de feromona en la cual para cada punto de selección de componentes, una de las posibles componentes tiene τ_{max} y el resto tiene τ_{min} como rastro de feromona. Cuando se ha producido la convergencia, la mejor solución hasta el momento se debe construir con probabilidad $p_{best} > 0$. Una hormiga construirá la mejor solución hasta el momento, si en cada selección de componente realiza la elección correcta, es decir si selecciona la componente que tiene asociado como rastro de feromona τ_{max} . Adicionalmente, la hormiga debe realizar n decisiones correctas, es decir una por cada componente que selecciona⁷. Finalmente se asume que las opciones entre las que debe elegir en cada paso es constante (*avg*) y que la probabilidad de tomar la decisión correcta en cada paso es constante. Bajo estos supuestos se determina la siguiente estimación:

$$\tau_{min} = \frac{\tau_{max} * (1 - \sqrt[n]{p_{best}})}{(avg - 1) * \sqrt[n]{p_{best}}} \quad (9)$$

- 3. Inicialización de los rastros:** Los rastros son inicializados con el máximo valor posible para todas las componentes (τ_{max}). Antes de la primera iteración no se conoce una solución por lo cual se utiliza un valor alto, para que tras la primera iteración todas las componentes tengan asociado el valor τ_{max} . Se utilizan tasas de evaporación lenta, logrando aumentar la exploración que se realiza en etapas tempranas.

Adicionalmente, se propone el mecanismo de suavizamiento de los rastros de feromona⁸ que no solamente es útil para \mathcal{MMAS} , sino para todos los algoritmos propuestos para ACO que incorporan elitismo. La idea consiste en fomentar la exploración mediante el incremento de la probabilidad de seleccionar componentes con una baja cantidad de feromona cuando el \mathcal{MMAS}

⁷El problema sobre el que se realiza el planteo en [52] es el TSP y n es la cantidad de ciudades.

⁸Pheromone Trail Smoothing, PTS.

ha convergido o está próximo a la convergencia⁹. Para esto se incrementa el rastro en forma proporcional a la diferencia al máximo rastro posible [52], con lo cual resulta:

$$\tau_{i,x_i}^* \leftarrow \tau_{i,x_i} + \delta * (\tau_{max} - \tau_{i,x_i}) \quad \text{con} \quad 0 < \delta < 1 \quad \forall \tau_{i,x_i} \in T \quad (10)$$

La propuesta también incorpora el concepto de lista de candidatos para obtener mejoras en el desempeño al crecer el tamaño de los problemas. La idea consiste en no considerar todas las posibles componentes al momento de tener que agregar una a la solución parcial, sino solamente un subconjunto. En caso de que todas las componentes consideradas ya estuvieran incluidas, se procede a considerar las que no fueron incluidas en la lista de candidatos.

En la tabla 4 se muestran los parámetros que utiliza esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el TSP [29], con la excepción de δ sobre el cual no se han realizado estudios de calibración. El valor que figura en la tabla como recomendado para δ es el que fue utilizado en [52] para evaluar la incorporación del mecanismo PTS.

Si bien puede parecer una gran cantidad de parámetros, varios están determinados analíticamente y se debe realizar simplemente el cálculo (τ_{min} , τ_{max} y por lo tanto también τ_{init}) y avg surge directamente del problema considerado. Los parámetros que requieren calibración son los usuales, p_{best} y δ (si se incorpora el mecanismo PTS).

5.1. Evaluación

Es una variante que presenta muy buenos resultados y que ha sido ampliamente utilizada, dejando en segundo plano a las variantes descritas anteriormente. Por ejemplo, en [29] se realiza un amplio estudio sobre las variantes AS, EAS, AS_{rank} , \mathcal{MMAS} y ACS aplicadas al TSP concluyendo que \mathcal{MMAS} obtiene los mejores resultados.

Es una variante que ha demostrado su versatilidad al haber sido utilizada exitosamente para resolver el TSP, ATSP y QAP. Aunque al incrementarse el tamaño de los problemas considerados, también requiere la utilización de operadores de búsqueda local para mantenerse competitiva con otras metaheurísticas.

En [49] se señala que el límite inferior del rastro de feromona juega un rol más importante que el límite superior para la obtención de soluciones de buena calidad. La fórmula planteada (ecuación 9) para el límite inferior

⁹Existen mecanismos que permiten determinar cuando se está en esas condiciones, por ejemplo el Average Branching Factor, presentado originalmente en [32] como el Mean λ -Branching Factor.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	n
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	τ_{max}
τ_{min}	Cantidad mínima de feromona en las componentes de las soluciones	$\frac{\tau_{max}*(1-\sqrt[p_{best}]{avg-1})}{(avg-1)*\sqrt[p_{best}]{avg-1}}$
τ_{max}	Cantidad máxima de feromona en las componentes de las soluciones	$\frac{1}{\rho*Costo(MejorSolucion)}$
p_{best}	La probabilidad de que una hormiga construya la mejor solución	0,05
avg	El promedio de la cantidad de componentes entre los que debe elegir una hormiga en cada paso para armar la solución completa	$\frac{n}{2}$
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	$2 \leq \beta \leq 5$
ρ	Tasa de evaporación de la feromona	0,02
δ	Tasa para el PTS	0,5

Tabla 4: Parámetros para el \mathcal{MMAS}

puede presentar dificultades al intentar aplicarla en otros problemas, debido a la dificultad de estimar el valor de algunas variables necesarias (por ejemplo: avg) para dicho cálculo.

Finalmente, conviene destacar que está disponible el código fuente de la variante \mathcal{MMAS} en [1].

6. Ant Colony System (ACS)

Al abordarse instancias de los problemas de mayor porte, las variantes simples de AS presentaron un empeoramiento de la calidad de las soluciones obtenidas. Explorando alternativas para este problema Gambardella y Dorigo propusieron en 1995 la incorporación de ideas presentes en técnicas de aprendizaje automático [32]. En particular se propuso el algoritmo ANT-Q que incorporaba mecanismos similares a los de la técnica de aprendizaje por refuerzo conocida como Q-Learning. Posteriormente, la propuesta fue re-

formulada en algunos de sus detalles y es conocida actualmente como Ant Colony System (ACS) [26, 27].

Se diferencia de AS fundamentalmente en los siguientes aspectos:

1. **Regla de transición de estados:** Se modifica la regla de transición de estados incorporando un mecanismo directo para balancear entre la explotación del conocimiento acumulado del problema y la exploración controlada de nuevas componentes. Para eso se incorpora el parámetro $q_0 \in [0, 1]$. Cuando cada hormiga debe incorporar una nueva componente genera un número aleatorio $q \in [0, 1]$ y aplica la regla de transición de estados conocida como regla proporcional pseudoaleatoria¹⁰ que se muestra en las ecuaciones 11 y 12.

$$\begin{aligned} &\text{Si } q \leq q_0: \\ p(c_{i,x_i} | s^p) &= \begin{cases} 1 & \text{si } c_{i,x_i} = \mathit{argmax}([\tau_{j,x_j}]^\alpha [\eta_{j,x_j}]^\beta) \text{ con } c_{j,x_j} \in J(s^p) \\ 0 & \text{en otro caso} \end{cases} \end{aligned} \quad (11)$$

$$\begin{aligned} &\text{Si } q > q_0: \\ p(c_{i,x_i} | s^p) &= \begin{cases} \frac{[\tau_{i,x_i}]^\alpha [\eta_{i,x_i}]^\beta}{\sum_{c_{j,x_j} \in J(s^p)} [\tau_{j,x_j}]^\alpha [\eta_{j,x_j}]^\beta} & \text{si } c_{i,x_i} \in J(s^p) \\ 0 & \text{en otro caso} \end{cases} \end{aligned} \quad (12)$$

Se suele eliminar el parámetro α , considerándolo como 1. En ninguno de los artículos considerados aparece una justificación directa sobre la eliminación de dicho parámetro. Sin embargo, conviene notar que en el algoritmo AS original α y β ponderan a los términos que reflejan la información histórica (explotación) y heurística (exploración) respectivamente. En esta propuesta existe un parámetro (q_0) que explícitamente realiza el balance entre la explotación (ecuación 11) y una exploración controlada (ecuación 12), con lo cual se desdibuja el sentido que originalmente tenían ambos parámetros, fundamentalmente α .

2. **Actualización global de la feromona:** Solamente una hormiga por iteración actualiza el rastro de feromona. Si bien esta idea tiene similitudes con la de \mathcal{MMAS} , conviene notar que en este caso solamente se realiza el depósito y la evaporación de feromona sobre las componentes que están presentes en la solución de la hormiga considerada ($\tau_{i,x_i} \in T^a$). Existen dos opciones sobre que hormiga considerar, la que

¹⁰pseudorandom proportional rule.

construyó la mejor solución hasta el momento o la mejor solución de la iteración. Típicamente se utiliza la mejor solución hasta el momento porque es la que ha presentado los mejores resultados en la práctica [26].

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \rho * \Delta\tau_{i,x_i}^{sbs} \quad \forall \tau_{i,x_i} \in T^{bs} \quad (13)$$

Se mantiene de la variante AS la regla para calcular la cantidad de feromona a depositar (ver ecuación 3). Solamente se realizan cambios en las componentes que forman parte de la mejor solución hasta el momento, para el resto de las componentes no se realiza actualización. La feromona depositada es multiplicada por el factor ρ , con lo cual la nueva cantidad de feromona es en realidad un promedio ponderado entre la cantidad anterior y la cantidad depositada.

3. **Actualización local de la feromona:** Cada hormiga al incorporar una componente a su solución parcial actualiza el rastro de feromona de acuerdo a la siguiente ecuación:

$$\tau_{i,x_i} \leftarrow (1 - \xi) * \tau_{i,x_i} + \xi * \tau_{init} \quad (14)$$

Siendo $0 < \xi < 1$. Se evaluaron alternativas a τ_{init} como la cantidad de feromona depositada localmente. El algoritmo ANT-Q utilizaba $\gamma * \max(\tau_{i,x_i})$ con $c_{i,x_i} \in J(s^p)$, es decir el valor máximo de feromona para el próximo componente a introducir en la solución multiplicado por el parámetro γ [32]. La otra opción considera fue 0, es decir no realizar depósito. Ambas opciones se consideran inferiores a la de la ecuación 14 [26]. En el primer caso los resultados son similares pero su cálculo es más costoso computacionalmente. En el segundo caso la calidad de las soluciones obtenidas fue inferior.

La idea detrás de esta actualización en línea es provocar pequeñas reducciones en la feromona a medida que las hormigas utilizan las componentes para construir sus soluciones en un mismo paso de la iteración. De esta forma se vuelve menos interesante su inclusión por parte del resto de las hormigas, permitiendo así la exploración de componentes no utilizadas. A raíz de este comportamiento es evidente que el algoritmo no presenta el mismo resultado si las hormigas construyen sus soluciones en forma secuencial o paralela. En general se ha optado por permitir la construcción en paralelo, pero no existen estudios que comparen ambas opciones [29].

Adicionalmente, la propuesta cuenta con el mérito de haber sido la primera que incorporó la utilización de lista de candidatos [29] (presentada en la sección 5).

En la tabla 5 se muestran los parámetros que utiliza esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el TSP [29]. En [26] se realiza un amplio estudio teórico sobre varios de los parámetros justificándose la utilización de $m = 10$.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	10
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{1}{n * Costo(SolucionAuxiliar)}$
q_0	Parámetro de la regla de transición de estados	0,9
β	Influencia relativa del componente heurístico	$2 \leq \beta \leq 5$
ρ	Tasa global de evaporación de la feromona	0,1
ξ	Tasa local de evaporación de la feromona	0,1

Tabla 5: Parámetros para el ACS

6.1. Evaluación

Es una variante que presenta resultados competitivos con otras técnicas aunque también requiere la incorporación de un operador de búsqueda local para instancias grandes de los problemas considerados. Presenta resultados levemente inferiores en calidad a los de $MMAS$, pero en general se constata que la evolución en la calidad de las soluciones obtenidas por $MMAS$ es más lenta que la de ACS [29]. Es posible que este comportamiento se deba a que ACS realiza una mayor explotación de las mejores soluciones encontradas y por tanto no logra explorar correctamente el espacio de búsqueda.

El algoritmo ANT-Q ha caído en desuso ya que presenta resultados similares en calidad a los de ACS y la forma de cálculo de la actualización local de feromona es más costosa computacionalmente.

Finalmente, conviene destacar que está disponible el código fuente de la variante ACS en [1].

7. Approximate Nondeterministic Tree Search (ANTS)

Es una propuesta de Maniezzo realizada en 1998 [42, 43, 44, 45] que incorpora conceptos de programación matemática. La idea presenta similitudes con las de la técnica Branch & Bound.

Se plantean las siguientes modificaciones al mecanismo básico del AS:

1. **Utilización de cotas inferiores:** Se modifica el mecanismo provisto para considerar la información heurística. En vez de tener en cuenta solamente el costo de la componente que se va a incluir, se calcula una cota inferior del costo de completar la solución parcial agregando dicha componente. Evidentemente a menor costo de la cota inferior, más atractiva tiene que ser la componente, ya que parece ser más prometedora incluirla en la solución que las otras componentes entre las que se puede optar.

Este mecanismo, adicionalmente permite evitar la construcción de soluciones que superarán el costo de la mejor solución hasta el momento. Si para una solución parcial, al considerar las cotas inferiores calculadas para cada una de las componentes disponibles, todas superan el costo de la mejor solución encontrada hasta el momento, la hormiga no debe proseguir con la construcción de la solución.

La ventaja sustancial que tiene este mecanismo es que aporta una visión global de la solución en la información heurística considerada, al tener en cuenta como repercute la inclusión de la componente en la construcción completa de la solución y en el costo de esta solución. En cambio en el esquema original de AS, la visión que se tiene es sumamente local al considerar exclusivamente el costo de la componente a incluir.

Es un mecanismo que puede ser costoso en lo que respecta al desempeño porque implica que para cada solución parcial, en cada paso en el cual se deba incorporar una componente a la solución, y para cada una de ellas se deben realizar los cálculos para obtener las cotas inferiores.

2. **Regla de transición de estados:** Se modifica la regla de transición de estados, utilizando una expresión que es más simple desde el punto de vista computacional y de los parámetros considerados, ya que evita el cálculo de potencias y utiliza solamente un parámetro en vez de los dos habituales.

$$p(c_{i,x_i}|s^p) = \begin{cases} \frac{\zeta\tau_{i,x_i} + (1-\zeta)\eta_{i,x_i}}{\sum_{c_{j,x_j} \in J(s^p)} \zeta\tau_{j,x_j} + (1-\zeta)\eta_{j,x_j}} & \text{si } c_{i,x_i} \in J(s^p) \\ 0 & \text{en otro caso} \end{cases} \quad (15)$$

Siendo $0 \leq \zeta \leq 1$.

3. **Actualización del rastro de feromona:** Se plantea un mecanismo diferente al usual que se caracteriza por "no tener una forma explícita" para la evaporación de feromona. La fórmula de actualización de la feromona tiene la siguiente forma:

$$\tau_{i,x_i} \leftarrow \tau_{i,x_i} + \sum_{a \in A} \Delta\tau_{i,x_i}^{s_a} \quad \forall \tau_{i,x_i} \in T \quad (16)$$

Siendo:

$$\Delta\tau_{i,x_i}^{s_a} = \begin{cases} \vartheta \left(1 - \frac{Costo(s_a) - LB}{Costo(s_{avg}) - LB} \right) & \text{si } c_{i,x_i} \in s_a \\ 0 & \text{en otro caso} \end{cases} \quad (17)$$

con $Costo(s_a)$ el costo de la solución considerada, $Costo(s_{avg})$ el costo promedio de las últimas l soluciones construidas, LB el valor de una cota inferior al costo de la solución óptima del problema que debe ser calculada al comienzo de la ejecución del algoritmo ($LB \leq Costo(s_{opt})$), l y ϑ son parámetros del algoritmo.

Conviene notar que si $Costo(s_a) > Costo(s_{avg}) \Rightarrow \Delta\tau_{i,x_i}^{s_a} < 0$, con lo cual se produce un decremento en la cantidad de feromona del componente que puede ser visto como una evaporación implícita. En el caso contrario, se produce el aumento en la cantidad de feromona de la componente.

El mecanismo planteado provoca un escalado dinámico de la feromona. En etapas avanzadas de la búsqueda, cuando la calidad de las distintas soluciones sea similar, permite la detección de mejoras leves y el refuerzo de las correspondientes componentes.

En la tabla 6 se muestran los parámetros que utiliza esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el QAP [42]. El parámetro τ_{init} no es necesario determinarlo a priori, se suele utilizar la cantidad de ocurrencias de las componentes en las cotas inferiores como estimación de la probabilidad de aparición en las soluciones. En

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	$\frac{n}{2}$
ζ	Influencia relativa entre la componente de feromona y la componente heurística	0,5
ϑ	Factor de la cantidad de feromona depositada por una hormiga	τ_{init}
l	Cantidad de soluciones consideradas para calcular el costo promedio	$4n$

Tabla 6: Parámetros para el ANTS

este caso por tratarse del problema QAP, n es la cantidad de recursos y de ubicaciones.

En [43, 44] se realizaron pruebas para el Frequency Assignment Problem (FAP) obteniendo valores diferentes para los parámetros. En este caso, los valores recomendados son: $m = \frac{n}{100}$, $l = \frac{n}{10}$ y $\vartheta = 0,3$.

7.1. Evaluación

La variante ha sido aplicada con éxito al QAP, FAP y sobre problemas de diseño de Data Warehouse [45]. En el caso del QAP, ha aventajado en calidad a los resultados obtenidos mediante TS y GRASP, sin requerir mayores tiempos de ejecución para obtener esa mejora [42].

No existen artículos que hagan referencia a su aplicación sobre el TSP, solamente en [29] se señala que se han realizado experimentos limitados sobre el ATSP.

Existen aspectos que presenta el algoritmo que pueden comprometer su éxito al abordar un problema concreto. La variante requiere de un mecanismo que a partir de una solución parcial permita estimar una cota inferior del costo. Es posible que al considerar un problema en particular no se conozcan este tipo de mecanismos, lo cual invalidaría la utilización de este algoritmo. Si para el problema considerado existiera este mecanismo, debería obtener una cota con una precisión aceptable sin comprometer el desempeño debido a un alto costo computacional asociado a su cálculo. Conviene recordar que la estimación de las cotas inferiores, se debe realizar en cada paso que hay que elegir componentes para cada una de las componentes consideradas, con lo cual un mecanismo costoso computacionalmente es prohibitivo.

8. Hyper-Cube Framework for ACO (HCF-ACO)

Es una propuesta realizada por Blum, Roly y Dorigo en 2001 [8] y consiste en realizar un escalado de los valores de rastro de feromona, de forma que permanezcan en el intervalo $[0,1]$. La motivación detrás de esta propuesta se vincula a la estrecha relación existente entre los valores de feromona de los componentes y la escala del problema. Se ha comprobado en la práctica que un mismo algoritmo ACO puede tener resultados distintos al ser aplicado sobre un mismo problema, teniendo como única diferencia que la función objetivo sea multiplicada por una constante [6].

En el planteo de la propuesta se estudia el conjunto de los valores de feromona a partir de considerar el vector $\vec{\tau} = (\tau_{i,x_i})$, $\forall \tau_{i,x_i} \in T$. Si se considera el problema de optimización combinatoria subyacente como un problema de decisión (0-1) o un problema de programación entera y se relaja permitiendo valores en el intervalo $[0, 1]$ para las variables de decisión, $\vec{\tau}$ es una solución al nuevo problema propuesto. A partir de lo cual, los cambios en $\vec{\tau}$ implican desplazamientos en el hiperespacio de las soluciones del problema relajado.

En [8] se presentan fórmulas que permiten realizar la actualización de la feromona de forma de cumplir con la premisa que los valores pertenezcan al intervalo $[0,1]$.

Para el AS:

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \rho \sum_{a \in A} \Delta \tau_{i,x_i}^{s_a} \quad \forall \tau_{i,x_i} \in T \quad (18)$$

$$\text{Siendo } \Delta \tau_{i,x_i}^{s_a} = \begin{cases} \frac{\frac{1}{F(s_a)}}{\sum_{l \in A} \frac{1}{F(s_l)}} & \text{si } c_{i,x_i} \in s_a \\ 0 & \text{en otro caso} \end{cases} \quad (19)$$

y para el MMAS:

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \rho \Delta \tau_{i,x_i}^{s_{best}} \quad \forall \tau_{i,x_i} \in T \quad (20)$$

$$\text{Siendo } \Delta \tau_{i,x_i}^{s_{best}} = \begin{cases} 1 & \text{si } c_{i,x_i} \in s_{best} \\ 0 & \text{en otro caso} \end{cases} \quad (21)$$

Una ventaja de esta formulación es que no requiere el recálculo de τ_{min} y τ_{max} al encontrarse una mejor solución como sucede en MMAS. Estos valores se conocen a priori y son 0 y 1 respectivamente. Se debe realizar el ajuste de los valores para asegurar que queden dentro del umbral, es decir si el

rastros de feromona en una componente es mayor que τ_{max} debe ser ajustado a τ_{max} . Lo mismo sucede si el rastro es inferior a τ_{min} debiendo ajustarse a τ_{min} .

No hay disponibles estudios sobre calibración de parámetros para esta variante. Simplemente se señala que τ_{init} debe ser 0,5 porque da igual importancia a ambas direcciones de búsqueda [6].

Adicionalmente se presenta un mecanismo para detectar la convergencia que consiste en contar para cuantas componentes se ha alcanzado el valor de τ_{min} o τ_{max} . Cuando un porcentaje significativo de ellas (99.9% [6]) han alcanzado uno de esos valores se producen la reinicialización de los rastros al valor τ_{init} .

8.1. Evaluación

No es exactamente otra variante, sino un concepto que puede ser aplicado sobre las variantes. La dificultad radica en hallar la ecuación de actualización de los rastros de feromona para que se mantengan en el intervalo $[0,1]$. Para los algoritmos AS, ACS y \mathcal{MMAS} las ecuaciones para actualizar los rastros ya fueron obtenidas [6]. A su vez presenta como ventaja que vuelve más explícita la relación existente entre la feromona depositada y las soluciones concretas al problema.

En [6] se reportan resultados de la comparación entre AS y AS con HCF-ACO, pudiendo comprobarse que efectivamente los resultados de la nueva variante son independientes del rango de los valores de las funciones objetivos de la instancia del problema considerado. Del mismo modo se realizan comparaciones con otros metaheurísticas obteniendo resultados competitivos. El problema utilizado para realizar la comparación es el Unconstrained Binary Quadratic Programming (UBQP).

Finalmente, conviene destacar que está disponible el código fuente de esta variante en [1].

9. Best-Worst Ant System (BWAS)

El algoritmo Population-Based Incremental Learning (PBIL) de la familia de los algoritmos evolutivos presenta fuertes similitudes con los algoritmos del framework ACO. Ambos algoritmos utilizan un mecanismo de memoria que se adapta con el paso de las iteraciones y generan las soluciones a partir de esa memoria. En ambos casos la adaptación de la memoria se produce considerando la calidad de las soluciones generadas. Esta variante fue propuesta por Cordón et al. en 2000 [15] y propone la incorporación en ACO

de mecanismos presentes en PBIL. La inclusión de ideas presentes en otros algoritmos evolutivos ha probado ser positiva en las variantes EAS y AS_{rank} .

Las principales diferencias con AS son las siguientes:

1. **Actualización de la feromona:** Se consideran dos hormigas para la actualización, la mejor hasta el momento para reforzar su rastro y la peor de la iteración para penalizarlo. El mecanismo de penalización consiste en aplicar una evaporación extra sobre las componentes que están presentes en la peor solución y que no están en la mejor solución hasta el momento. Con lo cual resulta el siguiente mecanismo de actualización de la feromona:

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \Delta\tau_{i,x_i}^{s_{bs}} \quad \forall \tau_{i,x_i} \in T \quad (22)$$

$$\tau_{i,x_i} \leftarrow (1 - \rho)\tau_{i,x_i} \quad \forall \tau_{i,x_i} \in T | c_{i,x_i} \in s_{wi} \wedge c_{i,x_i} \notin s_{bs} \quad (23)$$

Siendo s_{wi} la peor solución de la iteración. Se mantiene de la variante AS la regla para calcular la cantidad de feromona a depositar (ver ecuación 3)

2. **Reinicialización del rastro de feromona:** Se utiliza para provocar diversidad y se realiza cuando se detecta estancamiento. El rastro de feromona se reinicializa a τ_{init} . Originalmente se propuso como criterio para detectar el estancamiento, un porcentaje de componentes diferentes entre la mejor y la peor solución generada en la iteración [15]. Sin embargo, en la práctica no presentaba buenos resultados para instancias grandes, ya que no se alcanzaba el umbral y no se producían reinicializaciones. Posteriormente el criterio fue modificado por un porcentaje de la cantidad total de iteraciones [14]. Este criterio no considera explícitamente la detección de estancamiento, pero asegura la ejecución del reinicio.
3. **Incorporación de la mutación del rastro de feromona:** Se introduce un mecanismo para incorporar diversidad que no está presente en ninguna otra variante, la mutación de los rastros de feromona. El mecanismo está diseñado de forma de provocar pequeñas variaciones en las primeras etapas de la búsqueda, y grandes variaciones en las últimas etapas. En etapas avanzadas, cuando el algoritmo ha convergido, se provocan saltos en el espacio de búsqueda para encontrar zonas no visitadas, fomentando así la exploración.

Cada componente de la matriz de feromona es mutada con probabilidad P_m de acuerdo a la ecuación 24.

$$\tau_j = \begin{cases} \tau_j + mut(it, \tau_{threshold}), & \text{si } a = 0 \\ \tau_j - mut(it, \tau_{threshold}), & \text{si } a = 1 \end{cases} \quad (24)$$

Siendo a un valor aleatorio en $\{0, 1\}$, it es el número de la iteración que se está ejecutando y $\tau_{threshold}$ es el promedio del rastro de las componentes que están presentes en la mejor solución (ver ecuación 25). Conviene notar que al restarse un valor al rastro actual de feromona, podría ocurrir que el valor resultante sea negativo. Se debe realizar un chequeo explícito de esa condición, corrigiendo el valor a 0.

$$\tau_{threshold} = \frac{\sum_{c_i, x_i \in s_{bs}} \tau_{i, x_i}}{|s_{bs}|} \quad (25)$$

La función mut se define en la ecuación 26. Dicha función cumple con la propiedad de volver a su valor inicial cada vez que se produzca un reinicio.

$$mut(it, \tau_{threshold}) = \frac{it - it_r}{Nit - it_r} * \sigma * \tau_{threshold} \quad (26)$$

Siendo Nit el máximo número de iteraciones del algoritmo, it_r la última iteración en la que se produjo una reinicialización del rastro de feromona y σ un parámetro que permite ajustar el peso máximo de la mutación.

En la tabla 7 se muestran los parámetros que utiliza esta variante y sus valores recomendados para el TSP aunque se señala que no se ha realizado un análisis profundo de la influencia de los distintos parámetros [14]. El valor reportado para τ_{init} no parece ser correcto al considerar los valores usuales que se utilizan para AS (ver Tabla 1).

Existe una propuesta de hibridización con el algoritmo ACS en [13].

9.1. Evaluación

La variante ha sido aplicada con éxito sobre el QAP [13] y el TSP [14, 15] mostrando resultados competitivos contra las variantes AS y ACS para instancias de variados tamaños.

Se ha realizado un estudio sobre la incidencia en la calidad de los resultados obtenidos de cada una de las tres características del algoritmo propuesto.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	15
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{1}{n * Costo(SolucionAuxiliar)}$
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	2
ρ	Tasa de evaporación de la feromona	0,2
P_m	Probabilidad de mutación del rastro de feromona	0,3
σ	Parámetro del operador de mutación	4

Tabla 7: Parámetros para el BWAS

Los resultados demuestran que existe un balance entre dichas características y la eliminación de cualquiera de ellas empeora la calidad de los resultados obtenidos [14]. No se ha realizado un análisis profundo de la influencia de los distintos parámetros, solamente se han usado las configuraciones recomendadas para los otros algoritmos incorporando los nuevos parámetros.

Finalmente, conviene destacar que está disponible el código fuente de la variante BWAS en [1].

10. Utilización de una población auxiliar

Las variantes presentadas en esta sección, adicionalmente a la población habitual que utiliza la técnica para realizar múltiples búsquedas en cada iteración, incorporan una población auxiliar que permite almacenar soluciones de iteraciones anteriores. La población auxiliar se considera al momento de realizar la actualización del rastro de feromona. En estas variantes no es necesario almacenar explícitamente la matriz de rastros de feromona, sino que puede ser calculada en cada una de las iteraciones a partir de la población auxiliar. La principal motivación detrás de este cambio es permitir una mejor adaptación sobre problemas dinámicos, permitiendo que se reflejen los ajustes en la matriz de feromona en forma más rápida que con el manejo tradicional.

10.1. Population Based ACO (P-ACO)

La primera de las variantes se llama Population Based ACO (P-ACO) y fue propuesta en 2002 por Michael Guntsch y Martin Middendorf [38].

La idea consiste en mantener una población auxiliar P de soluciones de tamaño k (más pequeña que la población de hormigas) con las mejores soluciones generadas en las iteraciones pasadas. Inicialmente P está vacía. En cada iteración, cuando todas las hormigas han construido su respectiva solución, se realiza la actualización de P mediante la incorporación de la mejor solución de la iteración. Si como resultado de la actualización P contiene $k+1$ soluciones, la solución más antigua de P es borrada. En trabajos posteriores [39] se evaluaron otras opciones sobre que solución retirar de la población. Los criterios evaluados fueron: edad (la más antigua) y calidad (la peor) con el mismo funcionamiento; y calidad y probabilística (a partir de la calidad de las soluciones) considerando también como posible solución a retirar, la que se está insertando en la población. Sin embargo solamente se estudió su efecto en problemas de optimización combinatoria dinámicos.

Del mismo modo que en el resto de las variantes de ACO se utiliza una matriz de feromona para que cada una de las hormigas construya su solución. La diferencia fundamental consiste en que dicha matriz se calcula en cada iteración a partir de P . La matriz de feromona es inicializada con el valor $\tau_{init} > 0$ y se incrementa si la componente pertenece a soluciones de la población auxiliar de acuerdo a la siguiente ecuación:

$$\tau_{i,x_i} = \tau_{init} + \zeta_{i,x_i} * \Delta \quad \forall \tau_{i,x_i} \in T \quad (27)$$

siendo ζ_{i,x_i} el número de soluciones en P que contienen dicha componente. De esta forma se discretiza la cantidad de feromona de una forma conocida a priori e independiente de la calidad de cada una de las soluciones consideradas. Los valores posibles son $\tau_{init}, \tau_{init} + \Delta, \dots, \tau_{init} + k * \Delta$.

Debido a la forma en que se maneja el rastro de feromona no resulta necesario tener un mecanismo explícito de evaporación y por tanto no se requiere la utilización de un parámetro para ello. Δ se calcula a partir de establecer el valor máximo de feromona para cada una de las componentes τ_{max} , de acuerdo a la siguiente ecuación:

$$\Delta = \frac{\tau_{max} - \tau_{init}}{k} \quad (28)$$

Una de las premisas seguidas al momento de diseñar la variante fue reducir el tiempo de ejecución asociado al calculo de la actualización de la matriz de feromona [40]. En las variantes usuales, la cantidad de feromona a depositar se determina a partir de la calidad de cada una de las soluciones consideradas,

por lo cual se deben realizar operaciones para cada una de ellas. Para evitar recalcular en cada iteración toda la matriz de feromona y considerando que solamente una solución puede entrar y otra puede salir de la población, resulta conveniente almacenar la matriz de feromona aunque estén almacenadas las soluciones que permiten su cálculo. Para la actualización son necesarias $2n$ operaciones, ya que al ingresar una solución a la población para cada una de sus componentes se debe sumar Δ y para la solución que es eliminada de la población se debe restar Δ para cada una de sus componentes.

Se utiliza la regla de transición de estados de ACS que realiza un balance explícito entre la explotación determinística y la exploración probabilística incorporando el parámetro q_0 (ver ecuaciones 11 y 12).

En la tabla 8 se resumen los parámetros que utiliza esta variante y los valores recomendados determinados a partir de pruebas realizadas para el TSP [38]. En el mismo artículo se señala que para el problema QAP se utilizaron $\tau_{init} = \frac{1}{n}$, $\tau_{max} = 5$ y $\beta = 0$.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	10
k	Tamaño de la población auxiliar de hormigas	1
q_0	Parámetro de la regla de transición de estados	0,9
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{1}{n-1}$
τ_{max}	Cantidad máxima de feromona en las componentes de las soluciones	1
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	5

Tabla 8: Parámetros para el P-ACO

10.2. Omicron ACO (OA)

La segunda variante de esta familia se llama Omicron ACO y fue propuesta en 2004 por Osvaldo Gómez y Benjamín Barán [36]. Está fuertemente inspirada en P-ACO, presentando algunas diferencias con dicha variante. Del

mismo modo que en la anterior propuesta se mantiene una población auxiliar P de soluciones de tamaño k con las mejores soluciones generadas en las iteraciones pasadas.

Las principales diferencias que presenta con la variante P-ACO son las siguientes:

1. **Valor inicial de feromona:** Se elimina este parámetro, considerándose siempre en 1.
2. **Actualización de la feromona:** Se incorpora el parámetro O (Omicron) que representa la cantidad máxima de feromona que se puede agregar a una componente, es decir que se suma $\frac{O}{k}$ por cada solución que contenga la componente. En este caso los valores de feromona posibles para una componente τ_{i,x_i} son $1, 1 + \frac{O}{k}, \dots, 1 + \frac{(k-1)O}{k}, 1 + O$. La actualización de la matriz de rastro de feromona se realiza cada K iteraciones.
3. **Componentes de la población P :** La población auxiliar no podrá contener elementos repetidos.
4. **Política de reemplazo en la población P :** Se incorpora la mejor solución de la iteración, siempre que no esté presente en P y que sea mejor que el peor elemento de P .
5. **Regla de Transición:** Se utiliza la regla de transición de estados de la variante AS (ver ecuación 1).

En la tabla 9 se resumen los parámetros que utiliza esta variante y los valores recomendados determinados a partir de pruebas realizadas para el TSP [36]. El parámetro K podría ser eliminado y realizarse la actualización en todas las iteraciones aunque podría degradarse el desempeño.

10.3. Evaluación

La utilización de una población auxiliar para realizar la actualización del rastro de feromona es una idea relativamente reciente dentro de la comunidad científica y todavía no muy difundida. Presenta varios aspectos que vuelven atractivo su estudio.

El primero de ellos consiste en que permite una visualización más clara sobre en que estado se encuentra la matriz de feromona, ya que la población auxiliar representa una fotografía de que soluciones se consideraron para su armado. En las otras variantes con el paso de las iteraciones se pierde visión

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	n
k	Tamaño de la población auxiliar de hormigas	25
O	Cantidad máxima de feromona en las componentes de las soluciones	600
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	2
K	Cantidad de iteraciones para actualizar la matriz de feromona	1000

Tabla 9: Parámetros para el OA

sobre la incidencia real que tuvieron las soluciones en la matriz de feromona resultante. También resulta más clara la utilización de los parámetros y más simple la interpretación de que función cumplen en el algoritmo que para otras variantes.

Otro aspecto a considerar es la fuerte similitud que presenta con la metaheurística GA al incluir una población de soluciones. Esta similitud es una ventaja al momento de realizar implementaciones paralelas pudiendo adoptarse mecanismos similares a los utilizados por GA. Esta ventaja también se extiende a la posible hibridización con otras técnicas poblacionales¹¹ ya que resulta más natural realizar intercambios con las soluciones de la población auxiliar, que representan la experiencia adquirida sobre el problema, en vez de utilizar una de las soluciones construidas en la iteración.

La eficiencia computacional en lo que respecta al manejo de los rastros de feromona es mejor para la variante P-ACO que para las variantes usuales. Si se utilizan criterios distintos a la antigüedad como política de reemplazo en la población auxiliar de P-ACO o en el caso de OA, se deben realizar comparaciones adicionales entre soluciones que pueden atentar contra esta premisa.

Un aspecto que podría resultar cuestionable de las variantes de esta familia es que no utilizan en forma explícita la calidad de las soluciones para calcular el rastro de feromona. Sin embargo, las soluciones que se consideran en P son las últimas k mejores¹², con lo cual se asegura en cierta medida que

¹¹por ejemplo: GA.

¹²para el P-ACO. Para el OA son las mejores k .

la contribución es realizada por una solución de buena calidad. En las variantes más simples es necesario incluir la calidad de las soluciones ya que la actualización es realizada por todas las hormigas que construyen soluciones, sean estas de buena o mala calidad.

Para la variante P-ACO se realizaron comparaciones contra EAS y \mathcal{MMAS} sobre los problemas TSP y QAP [38]. Se señalan las dificultades que implica comparar en forma justa dos variantes que tengan diferentes parámetros. La metodología usada consiste en armar un ranking de acuerdo a los resultados obtenidos para todas las configuraciones de cada una de las variantes. Posteriormente realizan una clasificación de acuerdo a los parámetros comunes de ambas variantes que se están evaluando, obteniendo la posición promedio del ranking para cada una de las variantes y clasificaciones.

Para realizar las pruebas utilizaron como criterio usar el mismo tiempo de ejecución para cada una de las variantes. A partir de la metodología de evaluación propuesta, el promedio en el ranking de P-ACO es en general mejor que el de las otras variantes. A partir de la mejor posición obtenida en promedio en el ranking justifican que los resultados deben ser por lo menos igual de buenos en promedio que los de las otras variantes [38]. Conviene señalar que en ninguno de los casos se incluyeron operadores de búsqueda local.

En [39] se realiza un estudio sobre problemas de optimización combinatoria dinámicos (TSP y QAP dinámicos), comprobándose que P-ACO muestra una gran adaptación a cambios en las condiciones del problema.

Para la variante OA se realizaron comparaciones contra \mathcal{MMAS} sobre el TSP [36]. Para realizar las pruebas utilizaron como criterio usar el mismo tiempo de ejecución para cada una de las variantes. Se evaluó la evolución de la calidad promedio de las soluciones sobre un par de instancias del TSP, una de tamaño pequeño y otra de tamaño mediano. Los resultados obtenidos por OA fueron superiores a los de \mathcal{MMAS} . Conviene señalar que la comparación se realizó entre los algoritmos puros, es decir que no se incluyeron operadores de búsqueda local en ninguna de las dos variantes.

11. Utilización de soluciones parciales

En las variantes presentadas hasta esta sección, cada hormiga que construye una solución al problema (en el procedimiento *ConstructAntSolution*¹³) comienza con una solución vacía. En cada paso de la construcción de la solución incrementa en una componente la solución parcial que ha construido

¹³ver el algoritmo 2

hasta obtener una solución completa al problema¹⁴.

Las variantes presentadas en esta sección parten de una premisa diferente, la reutilización de porciones de soluciones generadas anteriormente a partir de las cuales comenzar la construcción de la solución completa.

Las variantes vistas hasta el momento suelen tener una alta explotación de las mejores soluciones encontradas. Esto puede ocasionar un estancamiento prematuro en la evolución de la calidad de las soluciones. Algunas de las variantes ya presentadas incorporan mecanismos explícitos para evitar ese tipo de inconvenientes. Las variantes de esta sección deben considerar especialmente esta posibilidad, ya que utilizan partes de soluciones ya construidas para construir las nuevas soluciones. Por lo cual deberán incluir mecanismos que aseguren un correcto balance entre exploración y explotación.

11.1. Variantes con memoria externa de Acan

La primera propuesta sobre la incorporación de soluciones parciales en la etapa de generación de soluciones fue realizada por Acan [2]. La idea esencialmente consiste en mantener una memoria externa a la población de hormigas en la cual se almacenan segmentos provenientes de soluciones de buena calidad, obtenidos en etapas anteriores de la búsqueda.

El mecanismo de funcionamiento puede dividirse en dos etapas, la primera para el llenado de la memoria externa y la segunda de funcionamiento en régimen de la variante.

En la primera etapa se comienza con una memoria externa de M segmentos de soluciones inicialmente vacía. Se ejecutan sucesivas iteraciones del algoritmo ACO hasta completar los M segmentos. En cada una de las iteraciones, se consideran las k mejores soluciones de la iteración y para cada una de ellas se selecciona en forma aleatoria un segmento de largo variable que se incorpora a la memoria. También se almacena en la memoria el largo del segmento y el costo asociado a la solución de la cual provino.

En el algoritmo 3 se presenta el funcionamiento de la segunda etapa, es decir cuando la variante está funcionando en régimen.

Se incorpora el procedimiento *TournamentSelection* que realiza la selección de la solución parcial mediante un torneo entre Q segmentos.

El procedimiento *ConstructAntSolution* incorpora como parámetro la solución parcial a partir de la cual se realiza la construcción. La construcción de la solución se realiza desde el fin del segmento considerado y podría ser realizada mediante cualquiera de las variantes de ACO. Acan utiliza una va-

¹⁴En general la solución completa es factible ya que no se suele trabajar en ACO con soluciones no factibles.

Algoritmo 3 Funcionamiento en régimen de la variante de Acan

```
while termination condition not met do
  for all ants  $a \in A$  do
     $s^p := \text{TournamentSelection}(M)$ 
     $s_a := \text{ConstructAntSolution}(s^p, T, H)$ 
  end for
   $\text{UpdatePheromone}(T)$ 
   $\text{UpdateExternalMemory}(M)$ 
end while
return best solution found
```

riante de \mathcal{MMAS} que incorpora una regla de transición de estados del estilo de ACS, es decir con balance explícito entre explotación y exploración controlada [2]. La forma en que se realiza la actualización debe guardar cierta coherencia con el mecanismo que se adopte para la actualización del rastro de feromona.

El procedimiento *UpdateExternalMemory* realiza la actualización de la memoria externa. Para esto se consideran las k mejores soluciones generadas en la iteración y se realiza el corte de segmentos del mismo modo que en la etapa inicial. En este caso se debe decidir que segmentos pertenecientes a la memoria deben ser sustituidos por los nuevos. Para cada segmento nuevo se consideran primero aquellos elementos de la memoria que tengan un costo superior, sustituyendo al peor de ellos. Si no existieran elementos con costo más alto que el nuevo segmento considerado, para obtener diversidad se concatena el segmento con el elemento de costo más alto. Al realizarse la concatenación pueden repetirse las componentes y se procede a su eliminación.

Posteriormente Acan realizó una segunda propuesta con muchas similitudes con la anteriormente presentada [3]. Los principales aspectos en los que se diferencia con su propuesta original son los siguientes:

1. **Utilización de permutaciones:** En vez de segmentos de soluciones, se utilizan permutaciones, liberándose la restricción original de que las componentes de las soluciones parciales sean contiguas. Se almacena el costo asociado a la solución de la cual provino y un tiempo de vida de la permutación. Cuando una permutación supera su tiempo de vida es eliminada de la memoria externa.
2. **Selección por torneo:** Se realiza a partir de un puntaje que vincula el costo (a través del fitness¹⁵) y la edad de la permutación. En la

¹⁵se utiliza en GA para guiar la búsqueda.

ecuación 29 se presenta la forma de obtener ese puntaje. La fórmula busca darle mayor prioridad al momento de ser seleccionadas a las que fueron incluidas antes en la memoria porque están más próximas a ser borradas.

$$Puntaje(i) = Fitness(i) + Edad^2(i) \quad (29)$$

3. **Mecanismo de sustitución de elementos de la memoria:** Se modifica, eliminándose la concatenación de elementos. Cuando no se encuentran en la memoria elementos con peor costo, se realiza la sustitución por el elemento de la memoria con peor costo dentro de los que tienen mayor edad.
4. **Búsqueda local:** Como en la gran mayoría de variantes de ACO, se incorpora un mecanismo de búsqueda local.

En la tabla 10 se muestran los parámetros que utiliza esta variante y los valores utilizados en [2] para el TSP. La segunda variante introduce el parámetro tiempo de vida de una permutación¹⁶, que fue utilizado con el valor 5 sobre el QAP [3].

11.2. Iterated Ants (ia)

La variante fue propuesta por Wiesemann y Stützle en 2006 [62] y consiste en incorporar la mecánica utilizada por la metaheurística Iterated Greedy (IG) al proceso de construcción de la solución. El procedimiento de construcción de una solución por parte de una hormiga se presenta en el algoritmo 4.

Algoritmo 4 Construcción de una solución para Iterated Ants

```

 $s^p := Destruct(s)$ 
 $s' := Construct(s^p)$ 
 $s' := LocalSearch(s')$  %optional
 $s := AcceptanceCriterion(s, s')$ 
return  $s$ 

```

En la primera etapa, el procedimiento *Destruct*, a partir de una solución completa s , elimina algunas componentes obteniendo una solución parcial s^p . Un aspecto que no está claro en la propuesta es que solución inicial (s) considera cada hormiga. Si se siguiera un mecanismo similar al de IG, la

¹⁶lifetime.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	n
M	Tamaño de la memoria externa	m
Q	Elementos participantes del torneo	5
k	Soluciones consideradas por iteración para la memoria externa	5
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{1}{n-1}$
τ_{max}	Cantidad máxima de feromona en las componentes de las soluciones	3
q_0	Parámetro de la regla de transición de estados	0,9
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	5
ρ	Tasa de evaporación de la feromona	0,1

Tabla 10: Parámetros para las variantes con memoria externa de Acan

hormiga debería generar esa solución antes de aplicar *Destruct*. Sin embargo, esta alternativa parece ir en sentido contrario de lo que se pretende con el cambio, es decir evitar que las hormigas construyan soluciones a partir de la solución vacía. Otras opciones que parecen tener mayor sintonía con la propuesta son considerar s como la mejor solución generada hasta el momento o la mejor solución generada por la misma hormiga hasta el momento. Este aspecto intuitivamente es importante para el éxito o no de la propuesta, sin embargo, en [62] no se menciona nada al respecto.

Existen varias alternativas para el procedimiento *Destruct* en lo que respecta a cómo se realiza la selección de las componentes a eliminar y cuántas se deben eliminar. La selección de las componentes a eliminar se realiza en forma aleatoria, proponiéndose las siguientes alternativa:

- rand: de acuerdo a una distribución uniforme.
- prob: en forma proporcional al rastro de feromona presente en la componente. Es decir, a mayor rastro, mayor probabilidad de ser eliminada.
- iprob: en forma inversamente proporcional al rastro de feromona presente en la componente. Es decir, a menor rastro, mayor probabilidad de ser eliminada.

Para la cantidad de componentes a eliminar se propusieron las siguientes alternativas:

- fija: Se incorpora un nuevo parámetro l que es la cantidad de componentes a eliminar.
- variable: es variable durante la ejecución del algoritmo. Si para el valor actual de l no se puede mejorar la solución se actualiza $l := l + 1$, en otro caso l se actualiza al valor mínimo.

A partir de s^p se realiza la construcción de la solución completa en el procedimiento *Construct*. Para esto se puede seguir cualquiera de los mecanismos utilizados en las variantes presentadas en las secciones anteriores. En particular en [62] se utiliza la de la variante *MMAS*. Se puede incorporar una búsqueda local que permita mejorar la solución generada mediante el procedimiento *LocalSearch*. Conviene recordar que en el framework está posibilidad existía mediante el procedimiento *ApplyLocalSearch*¹⁷.

Finalmente se aplica un criterio de aceptación para decidir cual es la solución que efectivamente se considera construida por la hormiga en el procedimiento *AcceptanceCriterion*. El criterio más razonable es utilizar la mejor solución de las dos consideradas.

El manejo de feromona de la variante se realiza del mismo modo que en *MMAS*. Esto y el hecho de utilizar el mecanismo de construcción de la misma variante llevó a los autores a darle el nombre *Iterated Ants MMS - MIN Ant System* (*iaMMAS*) a la propuesta.

Los parámetros que utiliza esta variante y los valores que se utilizaron sobre el QAP [62] son los mismo que con la variante *MMAS* (ver tabla 4) con la excepción de ρ que se utilizó 0,1. Sobre esos parámetros no se realizó una calibración específica. En cambio se realizó un análisis completo sobre el criterio de selección y la cantidad de componentes a eliminar. En el primer caso, *iprob* se mostró como la mejor opción, mientras que en el segundo caso no existe una opción que sea claramente mejor que la otra. Cuando se consideró un largo fijo (l), pudo comprobarse una gran variabilidad de la calidad de los resultados dependiendo de la instancia del problema considerado.

11.3. Cunning Ants (*cAS*)

La variante fue propuesta por Tsutsui en 2006 [57] y consiste en que cada hormiga genere la nueva solución usando partes de una solución generada en

¹⁷ver el algoritmo 1

una iteración previa. La idea incorpora el manejo de dos tipos de hormigas: las astutas (*c-ants*) y las donantes (*d-ants*)¹⁸. Las *c-ants* se apropian de parte de las soluciones construidas por las *d-ants*, completando la solución con el mecanismo usual.

La colonia de hormigas se organiza en m unidades que contienen una única solución. En cada iteración y en cada una de las unidades, una *c-ant* genera una nueva solución utilizando a la solución de la unidad como *d-ant*. Posteriormente se compara la nueva solución generada con la ya existente en la unidad, conservándose la mejor como la solución de la unidad. En un sentido amplio puede considerarse como una variante que incorpora una memoria adicional (las soluciones en cada una de las unidades) a la memoria que usualmente utilizan este tipo de algoritmos (la matriz de rastros de feromona).

La actualización del rastro de feromona utiliza las soluciones presentes en cada una de las unidades de acuerdo a la ecuación 30. Se mantiene de la variante AS la regla para calcular la cantidad de feromona a depositar (ver ecuación 3).

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \sum_{a \in A} \Delta \tau_{i,x_i}^{s_a} \quad \forall \tau_{i,x_i} \in T \quad (30)$$

La variante también incorpora los límites en el rastro de feromona τ_{min} y τ_{max} propios de la variante \mathcal{MMAS} . La fórmula para calcular τ_{min} (ver ecuación 31) no sufre modificaciones, pero la de τ_{max} debe ser corregida para tener en cuenta el depósito por parte de feromona de todas las hormigas de la colonia (ver ecuación 32).

$$\tau_{min} = \frac{\tau_{max} * (1 - \sqrt[n]{p_{best}})}{(\frac{n}{2} - 1) * \sqrt[n]{p_{best}}} \quad (31)$$

$$\tau_{max} = \frac{1}{\rho} * \sum_{a \in A} \frac{1}{Costo(s_a)} \quad (32)$$

Sobre el algoritmo original del AS (ver algoritmo 2) se realizan algunas modificaciones que se describen a continuación. En primer lugar, la etapa de inicialización del rastro de feromona debe incluir la generación de la solución inicial de cada una de las unidades, como se muestra en el algoritmo 5. El procedimiento *ConstructAntSolution* incorpora como parámetro la solución parcial a partir de la cual se realiza la construcción. Para el algoritmo 5, la solución parcial es vacía.

¹⁸cunning ants y donor ants respectivamente. Esto motiva el nombre del algoritmo.

Algoritmo 5 Inicialización del rastro de feromona para *cAS*

```
Initialize pheromone trails
for each unit  $k$  do
     $s'$  := ConstructAntSolution(empty,  $T$ ,  $H$ )
     $s''$  := ConstructAntSolution(empty,  $T$ ,  $H$ )
     $s_k$  := Best( $s'$ ,  $s''$ )
end for
UpdatePheromone
```

Un segundo aspecto que se modifica, es la forma en la cual la colonia de hormigas construye las nuevas soluciones. En el algoritmo 6 se refleja el manejo de las *c-ants* y las *d-ants*.

Algoritmo 6 ConstructAntsSolutions para *cAS*

```
for each unit  $k$  do
     $d$ -ant :=  $s_k$ 
     $c$ -ant := BorrowSolution( $d$ -ant)
     $s'$  := ConstructAntSolution( $c$ -ant,  $T$ ,  $H$ )
     $s_k$  := Best( $s_k$ ,  $s'$ )
end for
```

Adicionalmente se presenta una función densidad de probabilidad para determinar la cantidad de componentes que se toman de la solución anterior. Este mecanismo requiere la incorporación de un parámetro adicional (γ) [57].

En la tabla 11 se muestran los parámetros que utiliza esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el TSP [57]. Básicamente se modifica el valor del parámetro p_{best} de la calibración para el *MMAS* [52] y se incorpora el parámetro γ sobre el que se realiza un amplio estudio.

11.4. Evaluación

Las propuestas existentes que incorporan la utilización de soluciones parciales en el proceso de construcción de la solución son muy recientes y en algunos casos no han logrado una maduración definitiva. Todavía no es posible realizar una valoración definitiva sobre la bondad de la incorporación de este tipo de mecanismos.

Las variantes propuestas por Acan fueron comparadas contra *MMAS* sobre TSP [2] y QAP [2, 3]. Si bien los resultados reportados por sus variantes se muestran como superiores a los obtenidos por *MMAS*, están bastante

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	n
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	10
τ_{min}	Cantidad mínima de feromona en las componentes de las soluciones	$\frac{\tau_{max}*(1-\sqrt[n]{p_{best}})}{(\frac{n}{2}-1)*\sqrt[n]{p_{best}}}$
τ_{max}	Cantidad máxima de feromona en las componentes de las soluciones	$\frac{1}{\rho} * \sum_{a \in A} \frac{1}{Costo(s_a)}$
p_{best}	La probabilidad de que una hormiga construya la mejor solución	0,005
α	Influencia relativa de la componente de feromona	1
β	Influencia relativa del componente heurístico	2
ρ	Tasa de evaporación de la feromona	0,02
γ	Parámetro para la cantidad de componentes apropiados	0,4

Tabla 11: Parámetros para el cAS

lejos de los mejores resultados reportados con esa variante por otros autores. En [62] se señala, a partir de los resultados que presenta Acan, que la versión implementada de \mathcal{MMAS} contra la que realizó las comparaciones dista de ser buena. Independientemente de este hecho y del éxito o no que pueda presentar en la práctica, la propuesta de Acan cuenta con el mérito de haber sido la primera en incorporar la construcción de soluciones a partir de soluciones parciales obtenidas en iteraciones anteriores.

En el caso de $ia\mathcal{MMAS}$ se realizaron pruebas comparativas sobre el QAP contra \mathcal{MMAS} para todas las alternativas propuestas incorporando búsquedas locales en todos los casos. La utilización de soluciones parciales no pudo igualar la calidad de los resultados obtenidos por \mathcal{MMAS} [62]. Sin embargo, los autores señalan que existe evidencia que en contextos para los cuales no se conocen buenos algoritmos de búsqueda local, los resultados obtenidos por esta variante se muestran más promisorios. En cuanto al desempeño se constata que para un tiempo de ejecución fijo, la variante propuesta logra construir una mayor cantidad de soluciones. Este hecho parece natural debido a que se reutiliza parte de la solución.

En el caso de cAS las pruebas se realizaron sobre el TSP y el ATSP para instancias de variados tamaños [57]. Se compararon los resultados con-

tra \mathcal{MMAS} y ACS, comprobándose que esta variante aventajaba a ambas. Se extendieron las pruebas para considerar la utilización de operadores de búsqueda local. En todos los casos la calidad de las soluciones obtenidas fue superior al de las otras variantes.

Recientemente se ha aplicado el cAS sobre el QAP [58, 59] pudiendo comprobarse que se mantienen los buenos resultados obtenidos sobre el TSP. Los resultados experimentales obtenidos sobre dos de los problemas que más han sido abordados con este tipo de técnicas, posicionan a esta moderna variante, como una de las más promisorias. Un aspecto importante para dicho éxito parece ser la utilización de las unidades que pueden ser vista como una población de soluciones. Al mantenerse dicha población se asegura un cierto grado de diversidad en la búsqueda. Otro aspecto señalado como positivo de la propuesta, que podría ser generalizable a otras variantes de esta familia, es que el mecanismo de construcción de las soluciones evita el estancamiento prematuro al realizar actualizaciones más suaves del rastro de feromona [57].

12. Otras variantes

Las últimas dos variantes que se presentan en este reporte, modifican alguno de los aspectos esenciales del funcionamiento del framework ACO, al punto de ser consideradas por fuera de este. El primero de ellos, Hybrid Ant System (HAS), utiliza el rastro de feromona para provocar modificaciones en las soluciones y no para su construcción. El segundo de ellos, Fast Ant System (FANT), utiliza números enteros para el rastro de feromona, en vez de números reales, y no utiliza ningún mecanismo de evaporación explícita de feromona.

12.1. Hybrid Ant System (HAS)

La variante fue propuesta por Gambardella y Dorigo en 1997 [33]. Presenta similitudes con la metaheurística Iterated Local Search (ILS), ya que se aplican perturbaciones sobre la solución actual para posteriormente realizar una búsqueda local. El funcionamiento básico de la variante se presenta en el algoritmo 7.

En el bucle principal para cada una de las hormigas se realiza la modificación de la solución que tiene asociada, en el procedimiento *Modification* a partir del rastro de feromona. Esta es la principal diferencia que presenta la variante con el resto de los algoritmos comentados, ya que no tiene un procedimiento de construcción de soluciones. El procedimiento realiza R modificaciones sobre la solución.

Algoritmo 7 Hybrid Ant System

```
InitializePheromoneValues( $T$ )
for all ants  $a \in A$  do
     $s_a = \text{ConstructRandomSolution}()$ 
end for
while termination condition not met do
    for all ants  $a \in A$  do
         $s' = \text{Modification}(s_a, T)$ 
         $s'' = \text{ApplyLocalSearch}(s')$ 
         $s_a = \text{Best}(s_a, s'')$ 
    end for
    ApplyOnlineDelayedPheromoneUpdate( $T, s_a | a \in A$ )
    if converged then
        Reinitialize pheromone
    end if
end while
return best solution found
```

Se introduce un nuevo parámetro q_0 similar al de la variante ACS, para realizar un balance explícito entre la explotación determinística y la exploración probabilística. En el caso de la explotación se considera el mejor cambio posible para la componente (ver ecuación 33). Mientras que para la exploración se calculan las probabilidades para cada una de las opciones como las proporciones correspondientes para cada una de las opciones (ver ecuación 34). Se eliminan los parámetros α y β .

Si $q \leq q_0$:

$$p(c_{i,x_i} | s^p) = \begin{cases} 1 & \text{si } c_{i,x_i} = \text{argmax}(\tau_{j,x_j} \eta_{j,x_j}) \text{ con } c_{j,x_j} \in J(s^p) \\ 0 & \text{en otro caso} \end{cases} \quad (33)$$

Si $q > q_0$:

$$p(c_{i,x_i} | s^p) = \begin{cases} \frac{\tau_{i,x_i} \eta_{i,x_i}}{\sum_{c_{j,x_j} \in J(s^p)} \tau_{j,x_j} \eta_{j,x_j}} & \text{si } c_{i,x_i} \in J(s^p) \\ 0 & \text{en otro caso} \end{cases} \quad (34)$$

Después de la aplicación de la búsqueda local, se evalúa la nueva solución y la original y se conserva la mejor de ellas. La actualización del rastro de feromona es realizada solamente por una hormiga por iteración de acuerdo a la siguiente ecuación:

$$\tau_{i,x_i} \leftarrow (1 - \rho) * \tau_{i,x_i} + \rho \tau_{i,x_i}^{s_{bs}} \quad \forall \tau_{i,x_i} \in T \quad (35)$$

Se mantiene de la variante AS la regla para calcular la cantidad de feromona a depositar (ver ecuación 3).

Por último el algoritmo incorpora un mecanismo de diversificación que consiste en la reinicialización de los rastros cuando se detecta convergencia.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	10
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	$\frac{1}{n * Costo(SolucionAuxiliar)}$
q_0	Parámetro de la regla de transición de estados	0,9
R	Cantidad de modificaciones sobre la solución	$\frac{n}{3}$
ρ	Tasa global de evaporación de la feromona	0,1

Tabla 12: Parámetros para el HAS

En la tabla 12 se muestran los parámetros que utiliza esta variante y sus valores recomendados determinados a partir de pruebas realizadas para el QAP [34].

12.2. Fast Ant System (FANT)

La variante Fast Ant System (FANT) fue propuesta por Taillard y Gambardella en 1997 [54, 55] e incorpora un par de ideas novedosas en el manejo del rastro de feromona. Estas ideas buscan simplificar el mecanismo de actualización de los rastros con el objetivo de mejorar el tiempo de ejecución de este tipo de algoritmos.

Se diferencia de AS fundamentalmente en los siguientes aspectos:

1. **No utilización de colonia:** La variante FANT no utiliza una población de individuos (colonia) sino un único individuo con lo cual logra rápidamente encontrar buenas soluciones. Aunque en [53] se señala que no es una característica intrínseca a la variante, ya que si se utiliza una colonia se obtienen mejores resultados.

2. **Actualización del rastro de feromona:** La utilización de números enteros para los rastros de feromona y la no inclusión de un mecanismo explícito de evaporación de los rastros, simplifica en forma significativa las operaciones que se deben realizar para la actualización de la feromona. Debido a la no utilización de evaporación, periódicamente se deben realizar reinicializaciones de los rastros.

El mecanismo para el manejo de los rastros de feromona requiere dos parámetros: r es la cantidad de feromona que se agrega en las componentes de la solución que se genera en la iteración y r^* es la cantidad de feromona que se agrega en las componentes de la mejor solución hasta el momento. El parámetro r^* se mantiene fijo durante la ejecución mientras que r es variable. Al inicio de la ejecución se asigna $r = 1$ y $\tau_{i,x_i} = r \quad \forall \tau_{i,x_i} \in T$. Se aplican las siguientes ecuaciones para realizar la actualización del rastro de feromona:

$$\tau_{i,x_i} \leftarrow \tau_{i,x_i} + \Delta\tau_{i,x_i}^{sit} + \Delta\tau_{i,x_i}^{sbs} \quad \forall \tau_{i,x_i} \in T \quad (36)$$

$$\text{siendo } \Delta\tau_{i,x_i}^{sit} = \begin{cases} r & \text{si } c_{i,x_i} \in sit \\ 0 & \text{en otro caso} \end{cases} \quad (37)$$

$$\text{y } \Delta\tau_{i,x_i}^{sbs} = \begin{cases} r^* & \text{si } c_{i,x_i} \in sbs \\ 0 & \text{en otro caso} \end{cases} \quad (38)$$

Existen dos excepciones en las cuales no se siguen las ecuaciones presentadas anteriormente y son:

- Si se mejora sbs , se asigna $r = 1$ y $\tau_{i,x_i} = r \quad \forall \tau_{i,x_i} \in T$ con el objetivo de intensificar la búsqueda en las cercanías de sbs .
- Si la solución construida en la iteración coincide con la mejor solución hasta el momento ($s_{it} = s_{bs}$), se incrementa r en una unidad y $\tau_{i,x_i} = r \quad \forall \tau_{i,x_i} \in T$ con el objetivo de diversificar la búsqueda.

3. **Regla de transición de estados:** Se modifica la regla de transición de estados eliminando la componente heurística y los parámetros asociados.

$$p(c_{i,x_i} | s^p) = \begin{cases} \frac{\tau_{i,x_i}}{\sum_{c_{j,x_j} \in J(s^p)} \tau_{j,x_j}} & \text{si } c_{i,x_i} \in J(s^p) \\ 0 & \text{en otro caso} \end{cases} \quad (39)$$

En la tabla 13 se muestran los parámetros que utiliza esta variante. El único que realmente es un parámetro a ajustar es r^* y su valor recomendado está extraído de [55] a partir de la aplicación de la variante sobre el QAP.

Parámetro	Descripción	Valor recomendado
m	Tamaño de la población de hormigas	1
τ_{init}	Cantidad inicial de feromona en las componentes de las soluciones	r
r^*	Cantidad de feromona que se agrega en las componentes de la mejor solución hasta el momento	4
r	Cantidad de de feromona que se agrega en las componentes de la solución que se genera en la iteración	1(inicialmente)

Tabla 13: Parámetros para el FANT

12.3. Evaluación

La variante HAS fue aplicada originalmente sobre el Secuential Ordering Problem (SOP) [33] obteniendo muy buenos resultados en cuanto a calidad y desempeño. Inclusive sobre algunas instancias se lograron mejorar las mejores soluciones conocidas. Posteriormente se aplicó sobre sobre el QAP [34] mostrando buenos resultados comparativos sobre instancias reales, pero siendo superado ampliamente en instancias aleatorias por otras metaheurísticas (TS y GA). Algunos años después se intentó con poco éxito utilizar esta variante sobre el problema TSP [53]. Se comprobó que se obtenían mejores resultados realizando perturbaciones aleatorias que a partir de la matriz de feromona. No es una variante que se considere dentro de las que presentan los mejores resultados.

La variante FANT fue aplicada originalmente sobre el QAP [55] demostrando buenos resultados para ejecuciones cortas. Sin embargo al incrementarse el tiempo de ejecución es derrotado por otras metaheurísticas e inclusive la variante HAS. En el artículo se señala que la variante presenta muy buenos resultados para instancias muy grandes de los problemas al tener una estrategia de búsqueda muy rápida e intensa, inclusive pudo encontrar mejores soluciones que las que se conocían para algunas instancias. Al intentar utilizar la variante sobre el problema TSP no se obtuvieron buenos resultados [53], inclusive utilizando una colonia en lugar de un único individuo como se

establece en la propuesta original. El motivo para el mal desempeño es que hay una rápida convergencia a soluciones no óptimas.

En [53] se presenta la hibridización de las ideas detrás de HAS y FANT obteniendo mejores resultados para el TSP que cada una de las variantes individualmente.

13. Conclusiones

Ant Colony Optimization es una metaheurística que se ha consolidado en los últimos años transformando en una alternativa válida al momento de resolver un problema de optimización combinatoria. Ha demostrado ser competitiva con otras propuestas al ser aplicada sobre varios problemas estándares de optimización combinatoria como por ejemplo: TSP, ATSP, QAP, varios problemas de ruteo, etc. El desarrollo teórico no fue considerado en este reporte, pero conviene señalar que existen demostraciones para casos particulares sobre la convergencia de ACO.

En este reporte se ha realizado un amplio repaso de las distintas variantes propuesta de ACO para la resolución de problemas estáticos de optimización combinatoria. Se incluyeron las propuestas más antiguas, y por lo tanto las más difundidas, estudiadas y utilizadas, y nuevas propuestas de reciente formulación.

Dentro de las variantes históricas \mathcal{MMAS} y ACS se destacan ampliamente por sobre el resto, al haber demostrado su versatilidad al ser aplicadas sobre una gran variedad de problemas en forma exitosa. El resto de las propuestas o bien han sido superados por ambas variantes (AS, EAS, AS_{rank}), o son superiores para algún problema en particular debido a la explotación de algún tipo de particularidad que no siempre es posible extender hacia otros problemas (ANTS y el requisito de existencia de algún mecanismo para obtener buenas cotas inferiores).

La variante HCF-ACO no es exactamente una propuesta distinta, sino una forma de manejar las actualizaciones que puede ser aplicada sobre las otras variantes. El gran mérito de las ideas planteadas en HCF-ACO es que permite obtener resultados de calidad similar independientemente del escalado del problema considerado.

La variante BWAS incorpora conceptos presentes en un algoritmo evolutivo (PBIL). La gran diferencia conceptual que introduce es la utilización de mutación sobre los rastros de feromona para incorporar diversidad en la búsqueda. No es una propuesta que haya sido muy utilizada fuera del grupo que la formuló.

Las variantes que incorporan la utilización de una población auxiliar pa-

recen no haber sido suficientemente explotadas por la comunidad científica. Las propuestas existentes presentan aspectos interesantes como ser su claridad en lo que respecta al manejo de la memoria de búsquedas anteriores y a las posibilidades que brinda el manejo explícito de la población de soluciones.

La incorporación de soluciones parciales al proceso de construcción de las soluciones es la idea más reciente de las relevadas, por lo cual no se puede realizar una valoración definitiva sobre las virtudes de incorporar este tipo de mecanismos. Sin embargo, *cAS* se perfila como una variante de las más promisorias, a partir de los muy buenos resultados que ha obtenido sobre el TSP, ATSP y QAP.

Finalmente, HAS y FANT son propuestas que si bien presentan ideas que pueden ser consideradas interesantes, han sido poco exploradas en parte debido a que su éxito se restringe a algún problema en particular.

Bibliografia

- [1] The ant colony optimization home page. <http://iridia.ulb.ac.be/~mdorigo/ACO/ACO.html> consultada julio 2007.
- [2] Adnan Acan. An External Memory Implementation in Ant Colony Optimization. In M. Dorigo, M. Birattari, C. Blum, L.M. Gambardella, F. Mondada, and T. Stützle, editors, *Proceedings of ANTS 2004 – Fourth International Workshop on Ant Colony Optimization and Swarm Intelligence*, volume 3172 of *Lecture Notes in Computer Science*, pages 73–82, Brussels, Belgium, September 5-8 2004. Springer Verlag.
- [3] Adnan Acan. An External Partial Permutations Memory for Ant Colony Optimization. In *Fifth International Conference on Evolutionary Computation in Combinatorial Optimization - EvoCOP 2005*, Lecture Notes in Computer Science, pages 1–11, Lausanne, Switzerland, 2005. Springer Verlag.
- [4] Adil Baykasoğlu, Türkey Dereli, and Ibrahim Sabuncu. A Multiple Objective Ant Colony Optimization Approach to Assembly Line Balancing Problems. In *35th International Conference on Computers and Industrial Engineering (CIE35)*, pages 263–268, Istanbul, Turkey, June 2005.
- [5] George Bilchev and Ian C. Parmee. The Ant Colony Metaphor for Searching Continuons Design Spaces. In T. Fogarty, editor, *Proceedings of the Seventh International Conference on Parallel Problem Solving from Nature (PPSN VII)*, volume 993 of *Lecture Notes in Computer Science*, pages 24–39. Springer Verlag, 1995.
- [6] Christian Blum and Marco Dorigo. The Hyper-Cube Framework for Ant Colony Optimization. *IEEE Transactions on Systems, Man, and Cybernetics - Part B*, 34(2):1161–1172, 2004.
- [7] Christian Blum and Andrea Roli. Metaheuristics in Combinatorial Optimization: Overview and Conceptual Comparison. *ACM Comput. Surv.*, 35(3):268–308, 2003.
- [8] Christian Blum, Andrea Roli, and Marco Dorigo. HC-ACO: The Hyper-Cube Framework for Ant Colony Optimization. In J.P. Sousa, editor, *Proceedings of the fourth Metaheuristics International Conference (MIC'01)*, pages 399–403, Porto, Portugal, July 16-20 2001.

- [9] Bernd Bullnheimer, Richard F. Hartl, and Christine Strauss. A New Rank Based Version of the Ant System - A Computational Study. Working Paper 3/97, Institute of Management Science, University of Vienna, Austria, april 1997.
- [10] Bernd Bullnheimer, Richard F. Hartl, and Christine Strauss. Applying the Ant System to the Vehicle Routing Problem. In *Proceedings of the second Metaheuristics International Conference (MIC'97)*, Sophia-Antipolis, France, 1997.
- [11] Bernd Bullnheimer, Richard F. Hartl, and Christine Strauss. A New Rank Based Version of the Ant System: A Computational Study. *Central European Journal for Operations Research and Economics*, 7(1):25–38, 1999.
- [12] Alberto Colomi, Marco Dorigo, and Vittorio Maniezzo. Distributed Optimization by Ant Colonies. In F. Varela and P. Bourguine, editors, *Proceedings of the First European Conference on Artificial Life (ECAL)*, pages 134–142. MIT Press, Cambridge, Massachusetts, 1991.
- [13] Oscar Cordón, Iñaki Fernández de Viana, and Francisco Herrera. Analysis of the Best-Worst Ant System and its Variants on the QAP. In M. Dorigo, G. Di Caro, and M. Sampels, editors, *Proceedings of the Third International Workshop on Ant Algorithms (ANTS'2002)*, volume 2463 of *Lecture Notes in Computer Science*, pages 228–234, Brussels, Belgium, September 12-14 2002. Springer Verlag.
- [14] Oscar Cordón, Iñaki Fernández de Viana, and Francisco Herrera. Analysis of the Best-Worst Ant System and its Variants on the TSP. *Mathware & Soft Computing*, (9):177–192, 2002.
- [15] Oscar Cordón, Iñaki Fernández de Viana, Francisco Herrera, and Llanos Moreno. A New ACO Model Integrating Evolutionary Computation Concepts: The Best-Worst Ant System. In M. Dorigo, M. Middendoff, and T. Stützle, editors, *From Ant Colonies to Artificial Ants: Proceedings of the Second International Workshop on Ant Algorithms (ANTS'2000)*, pages 22–29, Brussels, Belgium, September 8-9 2000.
- [16] Gianni Di Caro. *Ant Colony Optimization and its Application to Adaptive Routing in Telecommunication Networks*. Phd thesis, Faculé des Sciences Appliquées, Université Libre de Bruxelles, Brussels, Belgium, 2004.

- [17] Gianni Di Caro and Marco Dorigo. AntNet: Distributed Stigmergetic Control for Communications Networks. *Journal of Artificial Intelligence Research*, 9:317–365, 1998.
- [18] Gianni Di Caro, Frederick Ducatelle, and Luca Maria Gambardella. Ant-HocNet: an Ant-Based Hybrid Routing Algorithm for Mobile Ad Hoc Networks. In X. Yao, E. Burke, J.A. Lozano, J. Smith, J.J. Merelo-Guervós, J.A. Bullinaria, J. Rowe, P. Tino, A. Kabán, and H.-P. Schwefel, editors, *Proceedings of the 8th International Conference on Parallel Problem Solving from Nature (PPSN VIII)*, volume 3242 of *Lecture Notes in Computer Science*, pages 461–470, Birmingham, UK, September 18-22 2004. Springer-Verlag.
- [19] Gianni Di Caro, Frederick Ducatelle, and Luca Maria Gambardella. Studies of Routing Performance in a City-like Testbed for Mobile Ad Hoc Networks. Technical Report 07-06, IDSIA, Lugano (Switzerland), March 2006.
- [20] Gianni Di Caro and Thanos Vasilakos. Ant-SELA: Ant-Agents and Stochastic Automata Learn Adaptive Routing Tables for QoS Routing in ATM Networks. In M. Dorigo, M. Middendoff, and T. Stützle, editors, *From Ant Colonies to Artificial Ants: Proceedings of the Second International Workshop on Ant Algorithms (ANTS'2000)*, Brussels, Belgium, September 8-9 2000.
- [21] Karl Doerner, Walter J. Gutjahr, Richard F. Hartl, Christine Strauss, and Christian Stummer. Pareto Ant Colony Optimization: A Metaheuristic Approach to Multiobjective Portfolio Selection. *Annals of Operations Research*, 131(1–4):79–99, October 2004.
- [22] Karl Doerner, Richard F. Hartl, and Marc Reimann. Are COMPETants More Competent for Problem Solving? - The Case of Full Truckload Transportation. *Central European Journal of Operations Research*, 11(2):115–141, 2003.
- [23] Marco Dorigo. *Optimization, Learning and Natural Algorithms (in Italian)*. PhD thesis, Politecnico di Milano, Italy, 1992.
- [24] Marco Dorigo and Gianni Di Caro. Ant Colony Optimization: A New Meta-Heuristic. In P.J. Angeline, Z. Michalewicz, M. Schoenauer, X. Yao, and A. Zalzal, editors, *Proceedings of Congress on Evolutionary Computation (CEC99)*, Washington DC, July 6-9 1999. IEEE Press.

- [25] Marco Dorigo and Gianni Di Caro. The Ant Colony Optimization Meta-Heuristic. In D. Corne, M. Dorigo, and F. Glover, editors, *New Ideas in Optimisation*, pages 11–32. McGraw-Hill, London, UK, 1999. Also available as Tech.Rep.IRIDIA/99-1, Université Libre de Bruxelles, Belgium.
- [26] Marco Dorigo and Luca Maria Gambardella. Ant Colony Sytem: A Cooperative Learning Approach to the Travelling Salesman Problem. Technical Report 1996-5, IRIDIA, Université Libre de Bruxelles, Belgium, 1996.
- [27] Marco Dorigo and Luca Maria Gambardella. Ant Colony Sytem: A Cooperative Learning Approach to the Travelling Salesman Problem. *IEEE Transactions on Evolutionary Computation*, 1(1):53–66, 1997.
- [28] Marco Dorigo, Vittorio Maniezzo, and Alberto Colorni. The Ant System: Optimization by a Colony of Cooperating Agents. *IEEE Transactions on Systems, Man, and Cybernetics-Part B*, 26(1):29–41, 1996.
- [29] Marco Dorigo and Thomas Stützle. *Ant Colony Optimization*. MIT Press, Cambridge, MA, USA, 2004.
- [30] Johann Dréo and Patrick Siarry. A New Ant Colony Algorithm Using the Heterarchical Concept Aimed at Optimization of Multim minima Continuous Functions. In M. Dorigo, G. Di Caro, and M. Sampels, editors, *Proceedings of the Third International Workshop on Ant Algorithms (ANTS'2002)*, volume 2463 of *Lecture Notes in Computer Science*, pages 216–221, Brussels, Belgium, September 12-14 2002. Springer Verlag, Berlin, Germany.
- [31] Stefka Fidanova. ACO Algorithm with Additional Reinforcement. In M. Dorigo, G. Di Caro, and M. Sampels, editors, *Proceedings of the Third International Workshop on Ant Algorithms (ANTS'2002)*, volume 2463 of *Lecture Notes in Computer Science*, pages 292–293, Brussels, Belgium, September 12-14 2002. Springer Verlag.
- [32] Luca Maria Gambardella and Marco Dorigo. Ant-Q: A Reinforcement Learning Approach to the Travelling Salesman Problem. In A. Prieditis and S. Russell, editors, *Proceedings of the Twelfth International Conference on Machine Learning (ICML)*, pages 252–260. Morgan Kaufmann, San Mateo, California, 1995.
- [33] Luca Maria Gambardella and Marco Dorigo. HAS-SOP: An Hybrid Ant System for the Sequential Ordering Problem. Technical Report 97-11, IDSIA, Lugano, Switzerland, 1997.

- [34] Luca Maria Gambardella, Éric Taillard, and Marco Dorigo. Ant Colonies for the Quadratic Assignment Problem. *Journal of the Operational Research Society*, 50:167–176, 1999.
- [35] Carlos García-Martínez, Oscar Cordón, and Francisco Herrera. An Empirical Analysis of Multiple Objective Ant Colony Optimization Algorithms for the Bi-criteria TSP. In Marco Dorigo, Mauro Birattari, Christian Blum, Luca M. Gambardella, Francesco Mondada, and Thomas Stützle, editors, *Proceedings of the 4th International Workshop on Ant Colony Optimization and Swarm Intelligence*, pages 61–72. Springer. Lecture Notes in Computer Science Vol. 3172, 2004.
- [36] Osvaldo Gómez and Benjamín Barán. Omicron ACO. In Mauricio Solar, David Fernández-Baca, and Ernesto Cuadros-Vargas, editors, *30ma Conferencia Latinoamericana de Informática (CLEI2004)*, pages 932–939. Sociedad Peruana de Computación, Septiembre 2004.
- [37] Marc Gravel, Wilson L. Price, and Cariline Gagné. Scheduling Continuous Casting of Aluminum Using a Multiple Objective Ant Colony Optimization Metaheuristic. *European Journal of Operational Research*, 143(1):218–229, November 2002.
- [38] Michael Guntsch and Martin Middendorf. A Population Based Approach for ACO. In Stefano Cagnoni, Jens Gottlieb, Emma Hart, Martin Middendorf, and Günther Raidl, editors, *Applications of Evolutionary Computing, Proceedings of EvoWorkshops 2002: EvoCOP, EvoIASP, EvoSTim*, volume 2279, pages 71–80, Kinsale, Ireland, 3-4 2002. Springer-Verlag.
- [39] Michael Guntsch and Martin Middendorf. Applying Population Based ACO to Dynamic Optimization Problems. In M. Dorigo, G. Di Caro, and M. Sampels, editors, *Proceedings of the Third International Workshop on Ant Algorithms (ANTS'2002)*, volume 2463 of *Lecture Notes in Computer Science*, pages 97–104, Brussels, Belgium, September 12-14 2002. Springer-Verlag.
- [40] Stefan Janson, Daniel Merkle, and Martin Middendorf. *Parallel Metaheuristics*, chapter Parallel Ant Colony Algorithms, pages 171–201. John Wiley & Sons, 2005.
- [41] Simon Kaegi and Tony White. Using Local Information to Guide Ant Based Search. In Paul W. H. Chung, Chris J. Hinde, and Moonis Ali, editors, *Proceedings of the 16th International Conference on Industrial and*

- Engineering Applications of Artificial Intelligence and Expert Systems (IEA/AIE 2003)*, volume 2718 of *Lecture Notes in Computer Science*, pages 692–701, Laughborough, UK, June 23-26 2003. Springer.
- [42] Vittorio Maniezzo. Exact and Approximate Nondeterministic Tree-Search Procedures for the Quadratic Assignment Problem. Technical Report CSR 98-1, C. L. in Scienze dell'Informazione, Università di Bologna, Sede di Cesena, Italy, 1998.
 - [43] Vittorio Maniezzo and Antonella Carbonaro. An ANTS Heuristic for the Frequency Assignment Problem. Technical Report CSR 98-4, C. L. in Scienze dell'Informazione, Università di Bologna, Sede di Cesena, Italy, 1998.
 - [44] Vittorio Maniezzo and Antonella Carbonaro. An ANTS Heuristic for the Frequency Assignment Problem. *Future Generation Computer Systems*, 16(8):927–935, 2000.
 - [45] Vittorio Maniezzo, Antonella Carbonaro, Matteo Golfarelli, and Stefano Rizzi. ANTS for Data Warehouse Logical Design. In J.P. Sousa, editor, *Proceedings of the fourth Metaheuristics International Conference (MIC'01)*, pages 249–254, Porto, Portugal, July 16-20 2001.
 - [46] Belén Melián, José Andrés Moreno Perez, and José Marcos Moreno-Vega. Metaheurísticas: Una visión global. *Inteligencia Artificial. Revista Iberoamericana de IA*, (19):7–28, 2003.
 - [47] Yoshiyuki Nakamichi and Takaya Arita. Diversity Control in Ant Colony Optimization. In *Proceedings of the Inaugural Workshop on Artificial Life (AL'01)*, pages 69–78, 2001.
 - [48] Yoshiyuki Nakamichi and Takaya Arita. Diversity Control in Ant Colony Optimization. *Artificial Life and Robotics*, 7(4):198–204, 2004.
 - [49] Thomas Stützle and Holger Hoos. Improving the Ant System: A Detailed Report on the $\mathcal{MAX} - \mathcal{MIN}$ Ant System. Technical Report 96-12, AIDA, Germany, 1996.
 - [50] Thomas Stützle and Holger Hoos. $\mathcal{MAX} - \mathcal{MIN}$ Ant System and Local Search for the Traveling Salesman Problem. In IEEE, editor, *Proceedings of the fourth International Conference on Evolutionary Computation (ICEC)*, pages 308–313. IEEE Press, 1997.

- [51] Thomas Stützle and Holger Hoos. The $\mathcal{MAX} - \mathcal{MIN}$ Ant System and Local Search for Combinatorial Optimization Problems: Towards Adaptive Tools for Global Optimization. In *Proceedings of the second Metaheuristics International Conference (MIC'97)*, Sophia-Antipolis, France, 1997.
- [52] Thomas Stützle and Holger Hoos. $\mathcal{MAX} - \mathcal{MIN}$ Ant System. *Future Generation Computer Systems*, 16(8):889–914, 2000.
- [53] Thomas Stützle and Sebastian Linke. Experiments with Variants of Ant Algorithms. *Mathware & Soft Computing*, 9 (2-3):193–207, 2002.
- [54] Éric Taillard. FANT: Fast Ant System. Technical Report 46-98, IDSIA, IDSIA, Lugano, 1998.
- [55] Éric Taillard and Luca Maria Gambardella. Adaptive Memories for the Quadratic Assignment Problem. Technical Report 97-87, IDSIA, Lugano, Switzerland, 1997.
- [56] Shigeyoshi Tsutsui. Ant Colony Optimization for the Continuous Domains with Aggregation Pheromones Metaphor. In *5th International Conference on Recent Advances in Soft Computing (RASC'04)*, pages 207–212, Nottingham, United Kingdom, 2004.
- [57] Shigeyoshi Tsutsui. *cAS*: Ant Colony Optimization with Cunning Ants. In Thomas Philip Runarsson, Hans-Georg Beyer, Edmund K. Burke, Juan J. Merelo Guervós, L. Darrell Whitley, and Xin Yao, editors, *Parallel Problem Solving from Nature - PPSN IX, 9th International Conference, Reykjavik, Iceland, September 9-13, 2006, Proceedings*, volume 4193 of *Lecture Notes in Computer Science*, pages 162–171. Springer, 2006.
- [58] Shigeyoshi Tsutsui. Cunning Ant System for Quadratic Assignment Problem with Local Search and Parallelization. MEDAL Report No. 2007006, Missouri Estimation of Distribution Algorithms Laboratory, University of Missouri, St. Louis, MO, 2007.
- [59] Shigeyoshi Tsutsui and Martin Pelikan. *cAS*: The Cunning Ant System. MEDAL Report No. 2007007, Missouri Estimation of Distribution Algorithms Laboratory, University of Missouri, St. Louis, MO, 2007.
- [60] Shigeyoshi Tsutsui, Martin Pelikan, and Ashish Ghosh. Performance of Aggregation Pheromone System on Unimodal and Multimodal

- Problems. In *The IEEE Congress on Evolutionary Computation, 2005 (CEC2005)*, volume 1, pages 880–887, 2-5 September 2005.
- [61] Tony White, Simon Kaegi, and Terri Oda. Revisiting Elitism in Ant Colony Optimization. In E. Cantú-Paz, J. A. Foster, K. Deb, D. Davis, R. Roy, U.-M. O’Reilly, H.-G. Beyer, R. Standish, G. Kendall, S. Wilson, M. Harman, J. Wegener, D. Dasgupta, M. A. Potter, A. C. Schultz, K. Dowsland, N. Jonoska, and J. Miller, editors, *Genetic and Evolutionary Computation – GECCO-2003*, volume 2723 of *Lecture Notes in Computer Science*, pages 122–133, Chicago, 12-16 July 2003. Springer-Verlag.
- [62] Wolfram Wiesemann and Thomas Stützle. An Experimental Investigation of Iterated Ants for the Quadratic Assignment Problem. Technical Report 2006-3, IRIDIA, Université Libre de Bruxelles, Belgium, 2006.