



UNIVERSIDAD DE BUENOS AIRES
FACULTAD DE CIENCIAS EXACTAS Y NATURALES
DEPARTAMENTO DE COMPUTACIÓN

Optimización en algoritmos de matching para el proyecto Eratosthenes

Tesis presentada para optar al título de
Licenciado en Ciencias de la Computación

Nicolás Panero

Director: Flavia Bonomo
Buenos Aires, 2016

OPTIMIZACIÓN EN ALGORITMOS DE MATCHING PARA EL PROYECTO ERATOSTHENES

Eratosthenes fue un matemático, geógrafo y astrónomo griego nacido en el 276 a.C.. Durante su cargo como jefe de la Biblioteca de Alejandría, halló un papiro en el cual se efectuaba una observación de un acontecimiento físico, que tomaba lugar en la ciudad de Siena 800km al sur de Alejandría. A partir de dicha información ideó un procedimiento utilizando conceptos de la matemática y la trigonometría para calcular la circunferencia de la tierra. Hoy en día existe un proyecto en donde escuelas de todo el mundo trabajan de forma colaborativa para recrear el procedimiento ideado por Eratosthenes y aprender en el proceso. Para ello las escuelas deben ser emparejadas bajo ciertas restricciones de forma tal que se optimicen sus estimaciones. Este trabajo analiza distintas estrategias para llevar a cabo el armado de dichas parejas, con el fin de lograr los mejores resultados posibles.

Palabras claves: Eratosthenes, Circunferencia de la Tierra, Matcheo Estable, Optimización, Estrategias.

MATCHING ALGORITHMS OPTIMIZATION FOR THE ERATOSTHENES PROJECT

Eratosthenes was a Greek mathematician, geographer and astronomer born in 276 b.C.. While he was in charge of the Library of Alexandria, he found a papyrus that described an observation of a physics event in the city of Syene, 800 km south of Alexandria. With the information he could extract of that papyrus, he thought about a procedure to measure the Earth's circumference using concepts of mathematics and trigonometry. Nowadays there is a project where schools of the entire world can work in a collaborative way to recreate the procedure of Eratosthenes and learn about it. In order to achieve those objectives, the schools must be split in pairs under certain restrictions to optimize the final estimates. This work looks to analyze multiple strategies to split the schools in such a way that let us obtain the best possible results.

Keywords: Eratosthenes, Earth Circumference, Stable Matching, Optimization, Strategies.

AGRADECIMIENTOS

En primer lugar me gustaría agradecer a Flavia Bonomo por darme la posibilidad de llevar a cabo este trabajo, por guiarme en él y responder a mis preguntas durante estos últimos meses.

También a Guillermo Mattei, por haberme otorgado parte de su tiempo para responder todas mis dudas a cerca del problema que abarca este trabajo y estar siempre dispuesto a colaborar con el mismo.

A mis amigos y compañeros de profesión Chris, Yesi y Vicky, que siempre me acompañaron, apoyaron y empujaron para que siga adelante y termine la carrera. Siempre estuvieron ahí dispuestos a poner el hombro y darme ánimos para seguir adelante.

A mis amigos del trabajo y ahora de la vida, siempre dispuestos a darme una mano y ayudarme en lo que necesitara. En especial a Belu que hasta me ayudó a procesar algunos datos, pero también a Juli, Andy, Lari, Alu y Kata, todos siempre con una palabra de apoyo para que no bajara los brazos y siguiera adelante, acompañándome incluso en momentos de mareos y contracturas.

A mis amigos de toda la vida Brai, Andru, Guille y Gasti que compartieron y comparten tanto tiempo e historias conmigo y aún hoy siguen remando a la par a pesar de haberlos dejado plantados más de una vez por tener que hacer la tesis.

A mi familia en general por estar ahí y a mis padres en particular, por brindarme todo el apoyo y el amor necesario para que pudiera crecer no sólo profesionalmente sino también como persona. A mis hermanas Juli y Flor que siempre estuvieron presentes cuando más las necesité. No podría haberlo hecho sin uds.

Y a todas aquellas personas que de una forma u otra, quizá sin saberlo, colaboraron para que este trabajo pudiera llevarse a cabo.

A mi viejo.

Índice general

1..	Introducción	1
2..	Proyecto Eratosthenes	5
3..	Modelo anterior	7
3.1.	Modelo 1	7
3.2.	Modelo 2	8
3.3.	Modelo 3	9
4..	Modelo Nuevo	10
4.1.	Estrategia vinculada al modelo anterior	12
4.2.	Estrategia Uno a Uno	13
4.3.	Estrategia Muchos a Muchos	16
4.4.	Estrategia de Múltiples Capas	19
5..	Mediciones y Resultados	21
5.1.	Resultados 2011	24
5.2.	Resultados 2012	27
5.3.	Resultados 2013	30
5.4.	Resultados 2014	33
5.5.	Resultados 2015	36
5.6.	Resultados 2016	39
6..	Conclusiones	42
7..	ANEXO	44

1. INTRODUCCIÓN

Para comenzar a introducirnos en la temática sobre la cual se basa el presente trabajo, debemos iniciar presentando al matemático, astrónomo y geógrafo griego Eratosthenes. Nacido en el año 276 a.C., Eratosthenes poseía una gran cantidad de conocimientos en múltiples ramas de la ciencia y una gran habilidad para aplicarlos.

Fue así que, siendo el jefe de la Biblioteca de Alejandría, halló un papiro a partir del cual terminaría realizando un descubrimiento de avanzada para su época. En dicho papiro se hacía una observación sobre un evento físico el cual tomaba lugar en la ciudad de Siena (hoy Asuan), aproximadamente 800km al sur de Alejandría. El evento hacía referencia a que, en el día del solsticio de verano, los objetos verticales en dicha ciudad no proyectaban sombra alguna y que se podía observar como la luz solar iluminaba directamente el fondo de los pozos de agua.

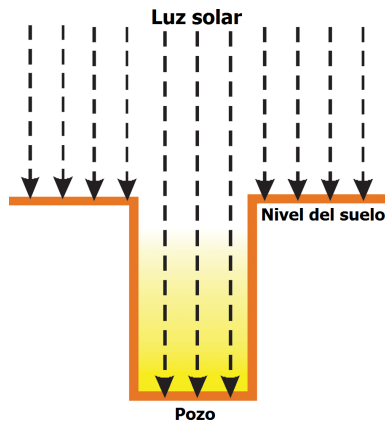
Si bien para otros estudiosos de la época, la información del mencionado papiro podría haber sido considerada una simple observación de algo trivial y cotidiano relacionado a la luz solar, Eratosthenes logró ver más allá de los simples hechos y entendió lo que dicha información implicaba. Que los objetos verticales no proyectaran sombra alguna, no dejaba lugar a otra conclusión más que entender que en ese momento el Sol se encontraba ubicado de manera perpendicular a la ciudad de Siena.

Fue entonces que Eratosthenes decidió comprobar si en Alejandría sucedía exactamente lo mismo, de modo tal que el siguiente solsticio de verano observó el comportamiento de las sombras en su propia ciudad. Para su sorpresa, en Alejandría los objetos verticales sí proyectaban una sombra, permitiendo a Eratosthenes extraer algunas primeras conclusiones.

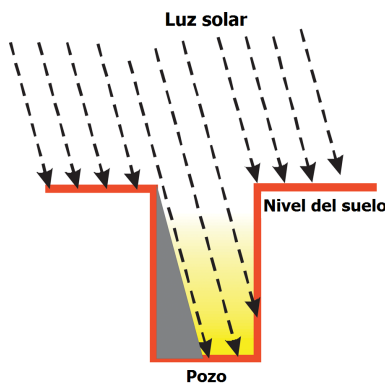
En primer lugar, le hablaba sobre la geometría de la Tierra, ya que si la misma fuera plana como se creía en ese entonces, la perpendicularidad de Siena con respecto al Sol debería sucederse también en Alejandría y por consiguiente el comportamiento durante el solsticio debería ser equivalente en ambas. Al no hacerlo en una y sí en la otra, se asimilaba más al comportamiento de un cuerpo esférico.

En segundo lugar, y el aspecto más importante asociado a este trabajo, sus conocimientos en trigonometría le permitieron idear una forma de calcular la circunferencia de la Tierra, procedimiento que se explica a continuación.

Repasando los hechos, sabemos que en el solsticio de verano en la ciudad de Siena, los elementos verticales no proyectaban ninguna sombra, así como también los pozos de agua reflejaban directamente la luz solar en el fondo de los mismos.

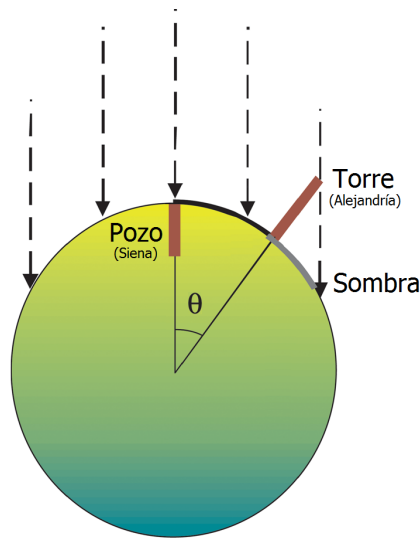


Además sabemos que en ese preciso momento en la ciudad de Alejandría, ocurría lo opuesto y se podía apreciar la sombra de todo elemento vertical proyectada en el suelo así como también en el fondo de los pozos de agua.

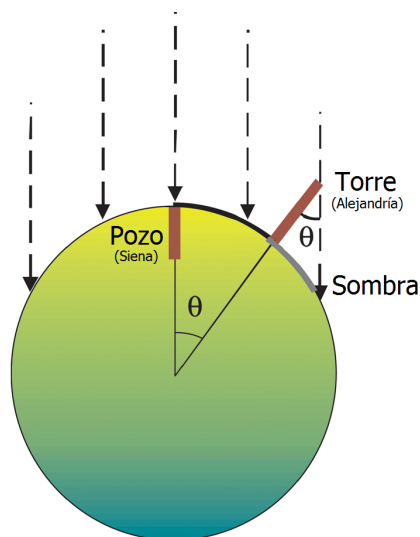


Tomando en consideración la gran distancia que separa a la Tierra del Sol en contraposición con la pequeña distancia que separa a las ciudades de Alejandría y Siena, Eratosthenes entendió que podía suponer que los rayos del sol caían de forma paralela sobre ambas ciudades.

Si entendemos que tanto el pozo en Siena como los elementos verticales en Alejandría están dispuestos de forma perpendicular al suelo y proyectamos la dirección de dichos objetos con una línea recta hacia el infinito, ambas rectas se intersectarán en el centro de la tierra generando un ángulo entre las mismas, que llamaremos Theta.



Por propiedades de la trigonometría, el ángulo que generan los rayos solares al toparse con algún elemento vertical en Alejandría resulta ser un ángulo alterno interno respecto de Theta, razón por la cual se corresponden con el mismo valor. Por lo tanto, si Eratóstenes era capaz de calcular el ángulo que generaban los rayos solares al toparse con los elementos verticales, estaría obteniendo el valor del ángulo Theta.



Restaría entonces conocer la distancia que separa a Siena de Alejandría para conocer la longitud del arco de la circunferencia que se corresponde para el ángulo Theta. Una vez obtenido ese valor y sabiendo que la circunferencia tiene 360° , podremos calcular la longitud total de la circunferencia. En el caso particular de Eratóstenes, obtuvo un valor

aproximado de 7° para Theta, lo cual significa una $1/50$ parte de 360° . Conociendo la distancia entre ambas ciudades solo bastaba hacer las cuentas fáciles.

Si bien los medios y herramientas para medir distancias en esa época eran bastante precarios y poco precisos (al punto de medirlas por el simple hecho de caminar las distancias), el razonamiento al cuál llegó Eratosthenes era totalmente acertado. Existen distintas versiones acerca del valor que utilizó como distancia entre Alejandría y Siena, pero dependiendo de la versión adoptada su error de cálculos oscila entre el 5% y 15%. Si recreamos hoy en día dicho procedimiento aprovechando las nuevas tecnologías para medir distancias y ángulos de manera más precisa, el error en el razonamiento de Eratosthenes es del 0,16%.

2. PROYECTO ERATOSTHENES

En la actualidad, un grupo del Departamento de Física de la Facultad de Ciencias Exactas y Naturales (UBA), organiza junto a otras instituciones, un proyecto a nivel escolar con el fin de enseñar el procedimiento realizado por Eratosthenes¹. En dicho proyecto la propuesta se basa en que estudiantes medios/secundarios de todo el mundo midan el tamaño del planeta Tierra de forma colaborativa, aprendiendo y aplicando nociones sobre trigonometría y matemática en el proceso.

Las escuelas que deseen participar se registran en una web informando los datos necesarios para que los organizadores puedan contactarse con ellos y especificando la latitud y longitud correspondientes a la posición geográfica de la escuela propiamente dicha. Una vez cerrada la etapa de registración, se establece un período (una o dos semanas) en el cual las escuelas deben efectuar sus mediciones, es decir medir el ángulo que establecen los rayos solares con respecto a un elemento vertical a la hora del mediodía solar. Luego cada escuela registra, por medio de la misma web, los valores obtenidos especificando también la fecha y hora en la cual se realizó el experimento.

Una vez que todas las escuelas registraron sus mediciones, se procede a ejecutar un algoritmo el cual se encarga de armar parejas entre las distintas escuelas bajo ciertas condiciones que favorezcan la obtención de mejores resultados. El objetivo del armado de las parejas es representar las ciudades de Alejandría y Siena, utilizadas en el procedimiento de Eratosthenes.

Algunas de las condiciones a tener en cuenta a la hora de armar las parejas son las siguientes:

- Dos escuelas pueden formar una pareja solo si ambas realizaron mediciones en un mismo día.
- A mayor distancia norte-sur entre las escuelas que conforman la pareja, mejores serán los resultados.
- Existe una cota inferior para la distancia norte-sur, hoy establecida en 400km. Dicho valor proviene de experiencias llevadas a cabo en años anteriores pero es ajustado año tras año.
- La distancia este-oeste entre las escuelas que conforman la pareja, no afecta el resultado y puede no ser tenida en cuenta.

Respecto a la primera condición debemos hacer una aclaración. Se pide que las escuelas hayan medido en el mismo día ya que el procedimiento se lleva a cabo generalmente durante el mes de septiembre, mes en el cual se produce el equinoccio. Esto es porque debemos realizar la experiencia dentro del año escolar y como los solsticios se producen durante los meses de Enero y Junio, realizarlo en dichos meses dificultaba la organización. A diferencia del equinoccio, durante el solsticio podemos considerar que el sol mantiene una posición similar por un margen de días. Eso nos permite relajar aún más la restricción y permitir que

¹ <http://eratostenes.df.uba.ar>

las escuelas hayan medido en días diferentes formen pareja. Pero no así con el equinoccio y de ahí proviene la condición antes enunciada.

Una vez conformadas las parejas, se puede proceder a obtener las estimaciones de la circunferencia de la tierra para cada una de ellas. Además se obtendrá también un resultado global de toda la experiencia y se procederá a informar los resultados a las distintas escuelas.

En particular, a diferencia del procedimiento de Eratosthenes, en este proyecto se termina calculando el radio de la tierra en lugar de su circunferencia, pero a fines prácticos se utiliza el mismo procedimiento. En el año 2015, el resultado global de toda la experiencia fue de (6.360 ± 20) km lo cual dista del radio real de la Tierra en un valor muy chico (el radio actual aceptado por la comunidad científica es de 6.371km).

3. MODELO ANTERIOR

En ediciones anteriores del Proyecto Eratosthenes, se utilizó otro algoritmo para el armado de las parejas de escuelas, basado en tres modelos principales, resueltos con técnicas de programación lineal [1].

- **Modelo 1:** El objetivo del primer modelo era obtener una vinculación que maximizara la distancia mínima de norte a sur entre las escuelas asociadas, sujeto a la restricción de que todas las escuelas tengan entre una y seis parejas.
- **Modelo 2:** Tomando como base el resultado del primer modelo, se aplicó un segundo modelo con el fin de lograr un acoplamiento que minimice la distancia máxima de Este a Oeste, manteniendo óptima la distancia mínima Norte-Sur.
- **Modelo 3:** Tomando como base el resultado del segundo modelo y sin perder optimalidad con respecto a los criterios anteriores, la idea en este tercer modelo era reducir al mínimo el número de enlaces de acoplamiento pero intentando maximizar a su vez la suma de las distancias Norte-Sur de las escuelas acopladas.

3.1. Modelo 1

El objetivo del primer modelo era obtener una vinculación que maximizara la distancia mínima de norte a sur entre las escuelas asociadas, sujeto a la restricción de que todas las escuelas tengan entre dos y seis parejas.

Parámetros del modelo:

- *cant*: cantidad de escuelas;
- *mtot*: máxima cantidad de matchings en total;
- *cmax* = 6: máxima cantidad de escuelas asociadas a una escuela;
- conjunto $I = \{1, \dots, cant\}$: conjunto de escuelas;
- conjunto $P = \{(i, j) \text{ en } I \times I \text{ con } i < j\}$: pares ordenados de escuelas;
- $dv : P \rightarrow \mathbb{Q}$: distancias en latitud entre escuelas;
- *dmax*: distancia máxima entre escuelas (en latitud).

Variables del modelo:

- $x \in \{0, 1\}^P$: $x_{ij} = 1$ si i se relaciona con j y 0 sino;
- $y \in \mathbb{Q}$: variable auxiliar para hacer máximo de mínimo.

Formulación del modelo:

máx y

$$\sum_{j \in I; j > i} x_{ij} + \sum_{j \in I; j < i} x_{ji} \geq 2 \quad \forall i \in I \quad (3.1)$$

$$\sum_{j \in I; j > i} x_{ij} + \sum_{j \in I; j < i} x_{ji} \leq cmax \quad \forall i \in I \quad (3.2)$$

$$\sum_{(i,j) \in P} x_{ij} \leq mtot \quad (3.3)$$

$$y \leq dv_{ij}x_{ij} + dmax(1 - x_{ij}) \quad \forall (i,j) \in P \quad (3.4)$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in P \quad (3.5)$$

Las restricciones 3.1 y 3.2 establecen el mínimo y máximo de parejas por escuela, respectivamente. La restricción 3.3 limita el número total de parejas establecidas. La restricción 3.4 fuerza, para cada par de escuelas (i, j) , a que la variable y sea más chica que la distancia máxima entre dos escuelas si i no se relaciona con j y más chica que dv_{ij} si i se relaciona con j . Luego y es menor o igual que la menor de las dv_{ij} entre los pares de escuelas i y j relacionadas, y se maximiza en la función objetivo.

3.2. Modelo 2

Tomando como base el resultado del Modelo 1, este segundo modelo busca lograr un acoplamiento que minimice la distancia máxima de Este a Oeste, manteniendo óptima la distancia mínima Norte-Sur.

Se agregan como parámetros al modelo:

- $opt_1 \in \mathbb{Q}$: valor óptimo obtenido en el Modelo 1;
- $dh : P \rightarrow \mathbb{Q}$: distancias en longitud entre escuelas.

Formulación del modelo:

mín y

$$\sum_{j \in I; j > i} x_{ij} + \sum_{j \in I; j < i} x_{ji} \geq 2 \quad \forall i \in I \quad (3.6)$$

$$\sum_{j \in I; j > i} x_{ij} + \sum_{j \in I; j < i} x_{ji} \leq cmax \quad \forall i \in I \quad (3.7)$$

$$\sum_{(i,j) \in P} x_{ij} \leq mtot \quad (3.8)$$

$$y \geq dh_{ij}x_{ij} \quad \forall (i,j) \in P \quad (3.9)$$

$$dv_{ij}x_{ij} \geq opt_1x_{ij} \quad \forall (i,j) \in P \quad (3.10)$$

$$x_{ij} \in \{0,1\} \quad \forall (i,j) \in P \quad (3.11)$$

Las restricciones 3.6–3.8 coinciden con las del modelo anterior. La restricción 3.9 fuerza, para cada par de escuelas (i, j) relacionadas, a que la variable y sea más grande que la distancia Este a Oeste entre ellas. Luego y es mayor o igual que la mayor de las dh_{ij} entre los pares de escuelas i y j relacionadas, y se minimiza en la función objetivo. Finalmente, la restricción 3.10 garantiza que la distancia Norte-Sur entre dos escuelas relacionadas es por lo menos el óptimo del Modelo 1.

3.3. Modelo 3

Tomando como base el resultado de los modelos 1 y 2 y sin perder optimalidad con respecto a los criterios anteriores, la idea en este tercer modelo es reducir al mínimo el número de enlaces de acoplamiento pero intentando maximizar a su vez la suma de las distancias Norte-Sur de las escuelas acopladas.

Se agrega como parámetro al modelo:

- $opt_2 \in \mathbb{Q}$: valor óptimo obtenido en el Modelo 2.

Formulación del modelo:

$$\begin{aligned} \text{mín} \quad & \sum_{(i,j) \in P} (dmax + 1 - dv_{ij})x_{ij} \\ \sum_{j \in I; j > i} x_{ij} + \sum_{j \in I; j < i} x_{ji} & \geq 2 \quad \forall i \in I \end{aligned} \quad (3.12)$$

$$\sum_{j \in I; j > i} x_{ij} + \sum_{j \in I; j < i} x_{ji} \leq cmax \quad \forall i \in I \quad (3.13)$$

$$\sum_{(i,j) \in P} x_{ij} \leq mtot \quad (3.14)$$

$$dv_{ij}x_{ij} \geq opt_1x_{ij} \quad \forall (i, j) \in P \quad (3.15)$$

$$dh_{ij}x_{ij} \leq opt_2x_{ij} \quad \forall (i, j) \in P \quad (3.16)$$

$$x_{ij} \in \{0, 1\} \quad \forall (i, j) \in P \quad (3.17)$$

Las restricciones 3.12–3.15 coinciden con las del modelo anterior. La restricción 3.16 garantiza que la distancia Este-Oeste entre dos escuelas relacionadas es a lo sumo el óptimo del Modelo 2. La función objetivo penaliza cada acoplamiento, pero con un valor igual a la diferencia entre la distancia en latitud máxima entre dos escuelas, y la distancia en latitud entre las escuelas acopladas. De esta manera, selecciona un conjunto no redundante de acoplamientos y entre ellos, el más conveniente respecto del criterio de maximizar las distancias Norte-Sur entre escuelas acopladas.

4. MODELO NUEVO

Si bien el software del modelo anterior cumplía con su objetivo, analizando los resultados finales logramos identificar algunos problemas. A su vez realizamos algunas reuniones con Guillermo Mattei, quien es el encargado de coordinar toda la experiencia con las escuelas, para relevar tanto los problemas como los aciertos que notaba al utilizar el sistema anterior. De esta forma la idea era juntar toda la información posible con el fin de planificar el desarrollo de un nuevo sistema que subsane todos o al menos la gran mayoría de dichos problemas.

Algunas de las problemáticas más reconocibles por Guillermo Mattei y por nosotros en nuestro análisis, fueron las siguientes:

- Necesidad de una persona técnica para efectuar las corridas del algoritmo de matching y los ajustes necesarios a dichas ejecuciones.
- No se contemplaba dentro del algoritmo, la restricción respecto del día en que se efectuaba la medición. Una vez establecidas las parejas, las estimaciones estaban basadas en el promedio de todas las mediciones cargadas por cada una de ellas, y no en la medición correspondiente al día en el cual ambas escuelas coincidían haber efectuado el procedimiento. Esto podría agregar error al resultado final para dicha pareja.
- Se restringían posibles parejas de escuelas a partir de intentar minimizar la distancia Este-Oeste, cuando esta última no tiene efecto al momento de obtener la estimación final para cada una de las parejas.
- Necesidad de una mayor diversidad de estrategias de matching que permitan al usuario ajustar la configuración de la ejecución según sea la distribución de escuelas.

En base a la información relevada de las distintas reuniones, planificamos el desarrollo de un sistema que sea de fácil utilización, con una interfaz gráfica simple, con el fin de que pueda ser utilizado directamente por los coordinadores del proyecto sin necesidad de contar con personal técnico. A su vez, notamos la necesidad de contar con múltiples estrategias para atacar el problema de matching, de forma tal de proveer al sistema de una mayor versatilidad para adaptarse a los distintos escenarios posibles, pudiendo combinar dichas estrategias entre sí.

Fue entonces que decidimos investigar algunas técnicas ya existentes para solucionar problemas similares. Uno de dichos problemas es conocido como *Problema de Matrimonios Estables* o *Stable Marriage Problem*. Este problema intenta encontrar para dos grupos de la misma cantidad de personas, uno de hombres y otro de mujeres, parejas estables. Por parejas estables nos referimos a que no debe existir un hombre y una mujer que se prefieran mutuamente por encima de las parejas que les fueron respectivamente asignadas.

En 1962, David Gale and Lloyd Shapley[2], comprobaron que para cualquier número equivalente de hombres y mujeres, siempre es posible encontrar el conjunto de parejas estables. Para ello presentaron un algoritmo el cual consta de varias etapas. En principio, cada uno de los hombres debe tomar el listado de las mujeres y ordenarlo según su preferencia. De la misma forma procederán las mujeres con respecto a los hombres. Una vez

que cada una de las personas tenga su propia lista de preferencias, se procede al armado de parejas. Dicho procedimiento se realiza de forma iterativa hasta que todas las personas terminen teniendo su propia pareja.

En una primera iteración, cada uno de los hombres comenzará realizando una propuesta a la primer mujer que se encuentre en sus respectivas listas de preferencias. Luego, cada mujer podrá haber recibido ninguna, una o múltiples propuestas. Si no recibió ninguna, entonces deberá seguir esperando recibir una. Si recibió una sola propuesta, deberá aceptarla quedando conformada la pareja. Si recibió múltiples propuestas, entonces solo aceptará la propuesta de aquel hombre que se encuentre lo más alto en su lista de preferencias y rechazará las restantes.

En una segunda iteración, los hombres que no hayan formado pareja, realizarán una propuesta a la mujer que se encuentre en el segundo lugar de sus listas de preferencias. Luego, como en el paso anterior, cada mujer podrá haber recibido ninguna, una o múltiples propuestas. Si no recibió ninguna, la mujer quedará en el estado en que se encontraba al finalizar la primer iteración. Si una mujer recibe una única propuesta y todavía no tenía pareja, entonces pasa a estar en pareja con dicho hombre. Si la mujer ya se encontraba en pareja, procederá a verificar el orden en su lista de preferencias del hombre de la nueva propuesta. Si se encuentra por debajo del hombre con el que está actualmente en pareja, entonces rechazará la nueva propuesta. Si se encuentra por encima, entonces anulará la pareja que tenía actualmente y se aceptará la nueva propuesta. En este último caso, el hombre que estaba en pareja pasa a ser soltero nuevamente y volverá a realizar propuestas en la siguiente iteración. Si una mujer recibe múltiples propuestas, procederá de la misma forma que en el caso anterior, solo que considerando a todos los hombres que le realizaron una propuesta en lugar de a uno solo.

De esta manera se continúa iterando, hasta que todas las parejas queden conformadas. Está demostrado que este procedimiento finaliza de manera correcta con todas las parejas armadas, quedando algunos más beneficiados que otros. También es verdad que este procedimiento beneficia al grupo de personas que inicie las propuestas, es decir, en el caso anterior los hombres salen beneficiados por sobre las mujeres al ser ellos quienes efectúan las propuestas (male-optimal).

Bajo este concepto de parejas estables y su solución hallada particularmente para el caso de los matrimonios, se encontraron problemas similares a los cuales se les podía aplicar la misma técnica. Por ejemplo, un problema existente en las universidades de Estados Unidos, era aquel referido al armado de las parejas de compañeros de cuarto (*Stable Roommates Problem*)[3, 4]. La única diferencia con respecto al problema de los matrimonios es que, en el caso de los compañeros de cuarto, no existe la distinción entre los dos grupos de hombres y mujeres, sino que por el contrario todas las personas conforman un único grupo. Estos problemas son catalogados también bajo el concepto de “*uno a uno*” o “*one to one*”, dado que toda persona siempre termina en pareja con una única otra persona.

Pero existen también otras variantes del mismo problema, en donde por ejemplo se permite a las personas tener empates dentro de su lista de preferencias, o no tener la misma cantidad de hombres y mujeres, o permitir a los hombres y mujeres tener más de una pareja. Por ejemplo, encontramos problemas asociados a la asignación de residentes a hospitales [5, 4]. El problema en este caso es asociar para cada hospital un grupo determinado de estudiantes. Podemos observar entonces que, a diferencia del problema de los matrimonios, en este caso tenemos una relación de uno a muchos.

Por último, podemos hacer referencia también a los problemas asociados al armado de Fixtures[4]. En dichos problemas, se precisa que un mismo equipo forme “pareja” con múltiples otros, conformando de esa manera uno de los grupos del fixture. Este último problema entra en otra categoría conocida como “*muchos a muchos*” o “*many to many*”.

Teniendo esto en mente, ideamos y planificamos cinco estrategias para incorporar a nuestro sistema, algunas de las cuales están basadas en los problemas anteriores. En las siguientes secciones procederemos a explicar cada una de ellas, mostrando a su vez un pseudo-código de las mismas para su mejor comprensión.

Pero previamente debemos aclarar un aspecto general, el concepto de preferencia bajo el contexto del proyecto Eratosthenes. Siempre que hagamos referencia al listado de preferencias de una escuela “*e*”, el mismo ha sido creado a partir de tomar el conjunto de escuelas que haya realizado mediciones en los mismos días que “*e*” y ordenarlo de mayor a menor considerando la distancia Norte-Sur que existe entre cada una de ellas y la escuela “*e*”.

Más adelante, cuando se detalle la implementación de las distintas estrategias, se podrá observar en las mismas la utilización de una función mediante la cual se calculan dichos listados de preferencia para las distintas escuelas. A continuación incluimos el pseudo-código de dicha función.

Function *GenerarListadosDePreferencias()*

```

Sea listadoPreferencias ;
foreach e en escuelas do
    Sea oes el listado de las otras escuelas que midieron en el mismo día que e
    con distancia Norte-Sur mayor que distanciaMinima;
    Ordeno oes decrecientemente en relación a su distancia Norte-Sur respecto
    de e;
    Guardo oes en listadoPreferencias asociado a e;
end

```

4.1. Estrategia vinculada al modelo anterior

En primer lugar, respetando y valorando lo realizado en el modelo anterior[1], pensamos en una primer estrategia que busque maximizar la distancia norte-sur para cada escuela. Más específicamente, para cada escuela registrada, se buscaba otra que haya realizado al menos una medición en el mismo día que la primera y que se encuentre a una distancia norte-sur mayor a un mínimo especificado por el usuario y lo más grande posible.

Si bien este procedimiento es simple y se enfoca en una de las características más importantes para optimizar el resultado final, también es verdad que tiene sus riesgos. Si ordenáramos todas las escuelas de norte a sur según su latitud, la mayoría de las escuelas formarían pareja con aquellas pocas que se encuentren en los extremos (suponiendo que midieron en el mismo día). Si por alguna razón estas últimas cometieron errores al realizar el proceso de medición, se verían afectados la mayoría de los resultados, perjudicando en consecuencia al resultado final global de toda la experiencia.

Es por ello que agregamos la posibilidad de que el usuario defina la cantidad máxima de parejas a conformar por las escuelas. De esta forma evitamos que un grupo reducido de escuelas termine formando pareja con todo el resto, disminuyendo entonces la probabilidad

de que una mala medición afecte a un gran porcentaje de resultados.

Nos referiremos a esta estrategia bajo el nombre de *MaxDistance*.

Function *MaxDistance()*

```

Sea maxParejas;
Sea escuelas;

GenerarListadosDePreferencias();

Sea parejas el listado final de parejas;
foreach e en escuelas do
    if maxParejas es cero then // no deseamos limitarlo
        Sea p la primer escuela del listado de preferencias de e;
        Construyo la pareja  $\langle e,p \rangle$ ;
    else
        Sea p la primer escuela del listado de preferencias de e, tal que no tenga
        más parejas que maxParejas;
        Construyo la pareja  $\langle e,p \rangle$ ;
    end
    Agrego la pareja  $\langle e,p \rangle$  al listado de parejas;
end

```

4.2. Estrategia Uno a Uno

Enfocándonos un poco más en los problemas de matching anteriormente enumerados y en particular los de tipo “*uno a uno*”, pensamos en dos posibles estrategias. Si bien ambas estrategias son iterativas, se diferencian en la forma en que generan las parejas. Una de ellas busca mejorar constantemente las parejas formadas en cada iteración, rompiendo parejas anteriormente creadas si fuera necesario. La otra supone que en cada iteración obtuvimos las mejores parejas posibles dado que se hicieron propuestas a las escuelas de mayor preferencia, por lo que en las siguientes iteraciones se buscará armar parejas entre las que quedaron solas (no rompe parejas anteriormente creadas).

De esta forma, el cuerpo principal de ambas variantes está compuesto, en su mayor parte, de los mismos pasos. Como se puede observar a continuación, en primer lugar se generan los listados de preferencias para cada escuela y luego se itera buscando las nuevas posibles parejas para finalmente procesar esas posibilidades y quedarnos solo con aquellas que correspondan.

Function *EstrategiaOneToOne()*

```

Sea escuelas;

GenerarListadosDePreferencias();

Sea parejas el listado final de parejas;
Sea siguientesIndices un listado que contiene el siguiente indice a procesar en la
lista de preferencias de cada escuela;

while conseguí armar parejas nuevas do
  | Sea posiblesParejasPorEscuela =
  |   ObtenerPosiblesParejasPorEscuela(siguientesIndices);
  |   ProcesarPosiblesParejasPorEscuela(posiblesParejasPorEscuela);
end

```

Resta entonces explicar la diferencia al momento de obtener las posibles parejas entre una estrategia y la otra. En primer lugar analicemos la estrategia que tiene la habilidad de romper parejas anteriormente creadas (nos referiremos a ella como ***OneToOne-BreakingMatches***). Como bien dijimos previamente, esta estrategia es iterativa y en cada iteración las escuelas que todavía no tengan pareja, proponen conformar una con la siguiente en su lista de preferencias. Una vez que todas realizaron sus propuestas, cada escuela que recibió una o muchas, se quedará sólo con aquella que esté mejor posicionada en su listado de preferencias. De esta forma, cada escuela tendrá siempre una única pareja, descartando o rompiendo parejas anteriormente conformadas siempre que corresponda. El conjunto de escuelas sin parejas se irá reduciendo con cada iteración, dado que a lo sumo podemos romper una pareja para crear una nueva, pero nunca incrementar el número de escuelas sin parejas de una iteración anterior.

A continuación, el detalle de la lógica utilizada para obtener las parejas

Function *ObtenerPosiblesParejasPorEscuela(siguientesIndices)*

```

Sea posiblesParejasPorEscuela;
foreach e en escuelas do
  | Sea posiblePareja;
  | if e no tiene pareja then
  |   | Sea preferencias el listado de preferencias de e;
  |   | if siguientesIndices[e] < preferencias.size() then
  |   |   | Seteo posiblePareja como el siguiente en la lista de preferencias;
  |   |   | Incremento en 1 el siguientesIndices[e];
  |   |   end
  |   | if encontramos una posiblePareja then
  |   |   | Agrego e al listado de posiblesParejasPorEscuela[posiblePareja];
  |   |   end
  |   end
  | end
end
return posiblesParejasPorEscuela;

```

Por otro lado, la otra estrategia de tipo “uno a uno”, procede de la siguiente manera. Cada iteración, las escuelas que todavía no estén en pareja, proponen conformar una a la siguiente escuela en su listado de preferencias que todavía no tenga pareja. Aquí tenemos una diferencia con respecto a la estrategia anterior. Una escuela solo le puede enviar una propuesta a otra que todavía se encuentre sola. Esto ya evita la posibilidad de poder romper parejas previamente creadas, ya que una vez que una escuela forma pareja, nunca más recibirá otra propuesta que amenace su estabilidad. Si una escuela recibe solo una propuesta, entonces formará pareja con la escuela que se lo propuso. Si por el contrario tiene varias, entonces se quedará solo con aquella que se encuentre mejor posicionada en el correspondiente listado de preferencias de la escuela receptora. Las restantes propuestas serán descartadas y las escuelas rechazadas tendrán que intentarlo de nuevo en la próxima iteración. Nos referiremos a esta estrategia bajo el nombre de **OneToOne-NotBreakingMatches** y dejamos a continuación el pseudo-código correspondiente a su lógica de armado de parejas.

```

Function ObtenerPosiblesParejasPorEscuela(siguientesIndices)
  Sea posiblesParejasPorEscuela;
  foreach e en escuelas do
    Sea posiblePareja;
    if e no tiene pareja then
      Sea preferencias el listado de preferencias de e;
      if siguientesIndices[e] < preferencias.size() then
        for p en preferencias con índice mayor a siguientesIndices[e] do
          if p no tiene pareja then
            Seteo posiblePareja como p;
            Incremento en 1 el siguientesIndices[e];
            break;
          end
        end
      end
      if encontramos una posiblePareja then
        Agrego e al listado de posiblesParejasPorEscuela[posiblePareja];
      end
    end
  end
  return posiblesParejasPorEscuela;

```

Finalmente, luego de obtener el listado de posibles parejas para cada escuela, debemos filtrarlo con el objetivo de que cada escuela tenga una única pareja. Como se explicaba anteriormente, si una escuela recibe una única propuesta, estará obligada a aceptarla. Caso contrario, se quedará con aquella que mejor posicionada esté dentro de su listado de preferencias.

```

Function ProcesarPosiblesParejasPorEscuela(posiblesParejasPorEscuela)
  foreach e en escuelas do
    if posiblesParejasPorEscuela[e] es distinto de vacio then
      Sea posiblesParejas = posiblesParejasPorEscuela[e];
      if posiblesParejas tiene una sola opcion then
        Sea p = posiblesParejas[0];
        Construyo la pareja  $\langle e,p \rangle$ ;
      else
        Sea preferencias el listado de preferencias de e;
        Sea p la escuela dentro de posiblesParejas que mejor posición tiene
          en preferencias;
        Construyo la pareja  $\langle e,p \rangle$ ;
      end
      if Si la pareja  $\langle e,p \rangle$  es mejor que las parejas actuales de e y p then
        Rompemos la pareja de e si existe;
        Rompemos la pareja de p si existe;
        Agregó la pareja  $\langle e,p \rangle$  al listado de parejas;
      end
    end
  end

```

4.3. Estrategia Muchos a Muchos

Luego pasamos a analizar la posibilidad de construir una estrategia que permita a las escuelas formar pareja con más de una, es decir orientado a las estrategias de tipo “*muchos a muchos*”. Para ello nos basamos en algunas técnicas utilizadas para el armado de fixtures[4]. Si bien nuestro problema no se asemeja en nada a un fixture, la problemática que enfrentan ambos problemas es similar en sus raíces. En ambos casos tenemos entidades las cuales queremos emparejar, agregando la condición de que una misma entidad no puede quedar emparejada con más de una determinada cierta cantidad. De esta forma podemos apreciar la aparición de un nuevo parámetro que denominaremos *capacidad*. La *capacidad* de una escuela será entonces lo que defina con cuantas otras escuelas puede terminar en pareja. Esto no implica que deba finalizar con esa cantidad, sino más bien actúa de cota superior.

Volviendo al problema puntual al que hace referencia este trabajo, explicaremos más en detalle el funcionamiento detrás de esta estrategia. Se trata de una estrategia que también actúa de forma iterativa y, al igual que en las anteriores, en un primer paso cada escuela crea su propio listado de preferencias con respecto a las demás escuelas. Luego se da inicio a la fase de propuestas en donde permaneceremos iterando siempre que exista alguna escuela la cual no haya llegado a completar su capacidad y que le queden escuelas por recorrer en su lista de preferencias. Al igual que en las estrategias anteriores, cuando una escuela realiza una propuesta, puede generar el quiebre de otra pareja. Pero en el caso de esta estrategia sucede algo más, ya que cuando se rompe una pareja, los integrantes de la misma también se borrarán mutuamente de sus listados de preferencias. De esta forma vamos reduciendo el tamaño de dichas listas al eliminar las opciones que no se pudieron concretar.

Una vez que todas las escuelas alcanzan su capacidad y/o finalizan de recorrer su listado de preferencias, se realizan algunas verificaciones para corroborar que luego de la fase de propuestas podamos encontrar un matching estable. Para ello se calcula, iterando sobre todas las escuelas, la sumatoria del mínimo entre la capacidad y la longitud de listado de preferencias. Como se demuestra en [4], de obtener un valor impar, nos estaría indicando que ya no podremos obtener un matching estable y no podremos seguir con la siguiente fase. Caso contrario se procederá a la fase de eliminación de rotaciones.

Las rotaciones se construyen a partir de las listas de preferencias y el objetivo es encontrar ciclos. Se comienza eligiendo a una escuela que tenga un listado de preferencias con longitud mayor a su capacidad, llamémosla y_i . Luego se busca la escuela que tenga por última opción a y_i , llamémosla x_i . Entonces tenemos el par (x_i, y_i) . Sea y_{i+1} la escuela que se encuentra en la posición $(c+1)$ en la lista de preferencias de x_i , siendo c la capacidad de x_i . Luego Buscamos nuevamente el x_{i+1} que tenga por última opción a y_{i+1} . Continuaremos de esta forma hasta volver a pasar por un par ya recorrido, habiendo formado un ciclo y en consecuencia haber encontrado una rotación.

Una vez encontrada la rotación se procede a su eliminación. Para cada par (x_i, y_i) de la rotación, se eliminan todos los pares (y_i, z) tal que y_i prefiere tanto a x_{i-1} como a $kl(y_i)$, por sobre z . Se define $kl(y_i)$ como la escuela de peor preferencia para y_i , que a su vez contiene a y_i dentro de las primeras c escuelas. Si $kl(y_i)$ no se encuentra definido, entonces solo eliminamos los pares (y_i, z) tal que y_i prefiere a x_{i-1} por sobre z . De esta manera seguiremos reduciendo los listados de preferencias ya que la eliminación del par implica que ambas escuelas se saquen de sus respectivas listas. Una vez eliminada la rotación se vuelve a buscar una nueva hasta que ya no encontremos ninguna, caso en el cual daremos por finalizada la estrategia quedando definidas las parejas a partir de las listas de preferencias.

Function *EstrategiaManyToMany()*

```

Sea escuelas;
Sea parejas el listado final de parejas;
Sea siguientesIndices un listado que contiene el siguiente indice a procesar en la
    lista de preferencias de cada escuela;

GenerarListadosDePreferencias();
Sea posiblesParejasPorEscuela = GenerarPosiblesParejas(siguientesIndices);

if los listados de preferencia no son estables then
    | Se termina la ejecución y no es posible encontrar un matching estable;
end

ProcesarPosiblesParejas(posiblesParejasPorEscuela);

```

Function *GenerarPosiblesParejas(siguientesIndices)*

```

Sea listadoDePosiblesParejas;
while podemosSeguirIterando do
  podemosSeguirIterando = false;
  foreach e en escuelas do
    if e tiene capacidad y no terminamos de iterar su listado de preferencias
    then
      podemosSeguirIterando = true;
      Sea posiblePareja el siguiente en la lista de preferencias de e;
      Incremento en 1 el siguientesIndices[e];
      Agregamos a posiblePareja al listadoDePosiblesParejas de e;
      Agregamos a e al listadoDePosiblesParejas de posiblePareja;
      if posiblePareja tiene mas parejas conformadas que su capacidad
      permitida then
        Ordeno el listadoDePosiblesParejas[posiblePareja] según la
        distancia Norte-Sur respecto de posiblePareja;
        Elimino todas aquellas parejas que tengan un indice mayor a la
        capacidad;
      end
    end
  end
end
return listadoDePosiblesParejas;

```

Function *ProcesarPosiblesParejas(listadoDePosiblesParejas)*

```

Sea existenListasCortas;
Sea escuelasConListasLargas;
while no existenListasCortas y escuelasConListasLargas no sea vacio do
  Sea e una escuela de escuelasConListasLargas;
  Busco rotación a partir de una e;
  Borrarnos la rotacion;
  Actualizo los valores de existenListasCortas y escuelasConListasLargas;
end
if existenListasCortas then
  Se termina la ejecución y no es posible encontrar un matching estable;
else
  Armamos las parejas a partir de los listados de preferencia;
end

```


4.4. Estrategia de Múltiples Capas

Por último, observando el comportamiento de las estrategias anteriores y observando también el dinamismo que tiene año tras año la distribución de las escuelas (a veces con ubicaciones más concentradas o más dispersas), notamos la necesidad de tener que idear una estrategia que nos provea de cierta flexibilidad para poder adaptarnos a esas características. Un ejemplo de lo anterior es lo que explicamos previamente en la estrategia *MaxDistance*, en donde se prioriza maximizar la distancia norte-sur y en donde las escuelas que se encuentran más al sur y más al norte serán elegidas por la mayoría. Si las escuelas de los extremos midieron mal, terminarían perjudicando a todas las demás. Es por ello que necesitamos distribuir de manera más pareja el peso de las mediciones y que la responsabilidad no recaiga toda en solo unas pocas.

La idea entonces es particionar la totalidad de las escuelas en múltiples franjas o capas, para después combinar dichas capas entre si utilizando alguna de las otras estrategias. Esto nos permite evitar que solo se armen parejas con ciertas escuelas al forzar la combinación con otras y, de esa manera, salvar el caso en donde algunas de ellas hayan medido mal. Llegado este último caso, seguramente terminemos descartando esas parejas por otras que nos provean de mejores resultados.

Para esta estrategia el usuario puede definir tres parámetros: la cantidad de capas en las que se quiere separar a las escuelas, el salto o gap a tener en cuenta al querer combinar las capas (un salto de 1 implica que no podremos combinar capas consecutivas) y la estrategia a ser utilizada internamente entre las capas.

En función de estos parámetros, separaremos a las escuelas en las distintas capas, para luego comenzar a combinarlas entre sí haciendo uso de la estrategia interna y considerando el gap suministrado por el usuario. Por cada par de capas que combinemos, obtendremos un listado de parejas las cuales iremos guardando. Al finalizar las combinaciones, tendremos definido el listado final de parejas.

Pero al igual que con *MaxDistance*, daremos la posibilidad al usuario de definir la cantidad máxima de parejas que una escuela puede tener. En caso de definir dicha cantidad, tendremos que filtrar la lista anterior. Para ello primero ordenaremos las parejas obtenidas en forma decreciente según su distancia norte-sur (esto es para priorizar parejas que cumplen con el factor que favorece mejores estimaciones) y adoptaremos todas las parejas posibles hasta alcanzar el límite.

Nos referiremos a esta última estrategia bajo el nombre de *MultipleLayers*.

Function *EstrategiaMultipleLayers()*

Sea *escuelas*;

Ordenamos la lista de *escuelas* de forma ascendente según su latitud;

Sea *cantidadDeEscuelasPorCapa*;

Sea *escuelasPorCapa* el listado de escuelas fraccionado en capas tomando de a *cantidadDeEscuelasPorCapa*;

Sea *posiblesParejas* = *GenerarParejasEntreCapas(escuelasPorCapa)*;

ProcesarParejas(posiblesParejas);

Function *GenerarParejasEntreCapas(escuelasPorCapa)*

```
Sea posiblesParejas el listado posibles parejas;  
for i entre 0 y cantidadCapas do  
  for j entre  $(i + 1) + \textit{gap}$  y cantidadCapas do  
    Sea l el listado de las escuelas de escuelasPorCapa[i] y  
    escuelasPorCapa[j];  
    Aplico la estrategia interna a l;  
    Agrego las distintas parejas obtenidas al listado de posiblesParejas;  
  end  
end  
return posiblesParejas;
```

Function *ProcesarParejas(posiblesParejas)*

```
Sea parejas el listado final de parejas;  
Ordeno el listado de posiblesParejas en forma decreciente según la distancia  
Norte-Sur entre los integrantes de cada pareja;  
foreach pareja en posiblesParejas do  
  if maxima cantidad de parejas por escuela es cero then  
    Agregamos la pareja al listado de parejas;  
  else if ambos miembros de la pareja pueden seguir adoptando parejas then  
    Agregamos la pareja al listado de parejas;  
end  
return parejas;
```

5. MEDICIONES Y RESULTADOS

Habiendo desarrollado las estrategias descritas anteriormente, el objetivo era comprobar su desempeño en comparación con el modelo anterior. Para ello solicitamos los datos registrados por las escuelas en ediciones pasadas del Proyecto Eratosthenes (años 2011 al 2015) y llevamos a cabo la ejecución de las nuevas estrategias.

Al comenzar a plantear los casos de prueba, identificamos tres aristas principales para analizar el comportamiento de las mencionadas estrategias. En primer lugar, la distancia mínima norte-sur que debe existir entre dos escuelas para poder formar una pareja. Esta distancia, como se explicaba en secciones anteriores, se estableció en 400km para el modelo anterior, pero nuestra idea era poder verificar si esa cota era realmente la mejor opción.

Luego tenemos el factor de las escuelas falsas. Dichas escuelas eran creadas anteriormente por los propios organizadores del proyecto, para otorgarle una pareja a aquellas escuelas que terminaban solas luego de ejecutar el algoritmo (ya sea por restricciones propias del algoritmo utilizado o porque, por ejemplo, realizaron las mediciones en un día en el que ninguna otra más lo hizo). Para evitar que las mismas salgan beneficiadas o perjudicadas por la escuela falsa, las creaban en la línea del ecuador en donde sabemos que el sol cae perpendicularmente y por lo tanto no genera sombra alguna. De esta manera, la escuela real terminaba obteniendo su estimación únicamente a partir de su propia medición. Incorporamos entonces este procedimiento a nuestras estrategias y entendimos que es un factor importante a observar ya que a mayor cantidad de escuelas falsas, se estaría atentando contra el aspecto colaborativo del proyecto en sí mismo (a pesar de, quizás, mejorar el desempeño global).

Por último, necesitábamos una forma de corroborar si los valores de las estimaciones obtenidas eran mejores respecto del modelo anterior. Fue entonces que, para cada uno de los años, tomamos las estimaciones obtenidas por las distintas parejas de escuelas y calculamos la diferencia en valor absoluto con respecto al valor real del radio terrestre. Luego sumamos todas esas diferencias y obtuvimos un valor representativo de cuán lejos del valor real estuvo el experimento en su totalidad. Nos referiremos a dicho valor como *diferencia acumulada*. La idea entonces era calcular respectivamente dicho valor para las nuevas estrategias y de esa forma verificar si el resultado general fue mejor o peor.

Teniendo en cuenta estos tres aspectos, ideamos los casos de prueba a ejecutar. Con el fin de analizar la cota mínima para la distancia norte-sur entre escuelas, decidimos realizar múltiples corridas para cada estrategia parametrizando justamente dicho valor. Más precisamente, ejecutamos cada estrategia con cotas que variaban desde 0km hasta 1000km con saltos de 100km. La decisión respecto de los 1000km como valor máximo, se debió simplemente al hecho de que en el modelo anterior se tomaba hasta 400km, por lo que pensar en un valor cercano al doble nos pareció suficiente. De la misma forma decidimos también probar con valores inferiores a 400km.

Los otros dos aspectos son valores que iban a ser provistos por los propios algoritmos al ser ejecutados. Por lo que si combinamos estos dos últimos aspectos con el referido en el párrafo anterior, obtendremos dos gráficos comparativos que nos ayudarán a comprender mejor el desempeño de las estrategias utilizadas.

No debemos olvidar que las estrategias *MaxDistance*, *ManyToMany* y *MultipleLayers* exponen un parámetro extra respecto de las estrategias *OneToOne*, el cual puede ser con-

figurado libremente por el usuario. Este parámetro hace referencia a la cantidad máxima de parejas que una escuela puede conformar. En este sentido el modelo anterior solía utilizar un valor cercano a 6, pero con el objetivo de encontrar mejores resultados decidimos realizar pruebas con valores entre 0 y 10 (siendo 0 el valor que indica que no limitaremos la cantidad de parejas). Como consecuencia, se podrá observar por ejemplo en los gráficos asociados a la estrategia *MaxDistance*, múltiples variaciones denominadas *MaxMatchesByLoc_X* las cuales hacen referencia a este parámetro (siendo X el valor entre 0 y 10 respectivamente).

Por otra parte, la estrategia *MultipleLayers*, presenta un gran número de combinaciones y posibilidades a evaluar. En principio contamos con la variable asociada a la cantidad de capas en las que dividiremos el conjunto de escuelas. A esto se suma la posibilidad de variar el salto o gap con el cual combinaremos dichas capas. Además, contamos con la variable respecto de la distancia mínima norte-sur entre escuelas y, como explicábamos en el párrafo anterior, la posible cota máxima para la cantidad de parejas por escuela.

Naturalmente nos vimos forzados a encontrar cotas para las primeras dos variables (las otras dos ya estaban definidas), con el fin de hacer alcanzable al experimento. Explicamos entonces las justificaciones detrás de dichos valores:

- **Cantidad de capas:** para encontrar tanto el extremo inferior como superior a esta variable, analizamos la cantidad de escuelas que se suscribían año tras año al experimento. Notamos entonces que para los años 2012 y 2014 se dieron los valores más bajos, teniendo 160 y 161 escuelas registradas respectivamente (considerando solo aquellas que finalmente realizaron mediciones). Suponiendo un mínimo de 10 escuelas por capa con el fin de lograr un número razonable de parejas al enfrentar unas con otras, resulta una cantidad máxima de 16 capas. Por el contrario, si pensamos en el valor más chico posible para esta variable, es natural pensar en al menos 2 capas.
- **Salto o gap mínimo:** Como bien explicamos anteriormente, el salto o gap mínimo hace referencia a la distancia mínima a tener en consideración a la hora de entrecruzar capas. Un gap de 1 implica que no podremos cruzar capas consecutivas. Teniendo eso en mente, para definir la cota máxima de esta variable, debemos considerar que la capa central debe ser capaz de entrecruzarse con las de los extremos. Siendo que la máxima cantidad de capas es 16, debemos adoptar un gap máximo de 7. De esa forma, las capas 8 y 9 (las capas centrales si consideramos que contamos con 16) podrán cruzarse con las capas 16 y 1 respectivamente. Por el contrario, para el valor mínimo podemos pensar en 0, el cual implicaría cruzar todas contra todas.

Como podemos ver, contamos con muchas variables para esta estrategia, las cuales combinadas generan una gran cantidad de posibles resultados. Esto implica también una gran cantidad de información para revisar y comparar con el fin de encontrar la combinación más efectiva y un tiempo de procesamiento mayor para poder generar todos esos archivos, convirtiendo al sistema en impracticable.

Siendo conscientes de este problema, ideamos dos modos de procesar la información en el sistema. Por un lado, el *Análisis específico*, donde el usuario podrá seleccionar una estrategia determinada, definir sus parámetros y obtener tanto las parejas conformadas como también sus resultados, permitiendo efectuar un análisis de mayor profundidad. Por otro lado, el *Análisis Global*, donde daremos la posibilidad al usuario de realizar un análisis

no tan detallado pero que abarque todas las estrategias con sus respectivas combinaciones. Solo obtendrá los datos asociados a la diferencia acumulada y la cantidad de escuelas falsas creadas para cada caso, lo que agilizará su ejecución. La idea es entonces que el usuario realice en primer lugar este análisis, pre-seleccione aquellos casos que merezcan un análisis más profundo, y entonces sí, efectué un *Análisis específico* sobre dicha selección.

Fue gracias a este último modo, que pudimos ejecutar la estrategia *MultipleLayers* en su totalidad, sin tener que restringir o descartar casos. Naturalmente no incluimos todos los posibles resultados en nuestros gráficos, sino que nos quedamos con dos de ellos. Dichos casos representan la combinación que minimiza la diferencia acumulada y la combinación que minimiza la cantidad de escuelas falsas creadas (en caso de existir varias con el mismo valor, aquella que también minimice la diferencia acumulada). Ambos casos fueron tomados a partir del sub-grupo de resultados que generaban hasta un tope máximo de 10 escuelas falsas. Esto último es para evitar considerar los casos con diferencias acumuladas chicas, pero con un gran número de escuelas falsas.

Por último nos gustaría aclarar, que no se encontrarán representados los resultados correspondientes a la estrategia *ManyToMany*. Esto se debe principalmente a que no logramos obtener ningún resultado a partir de la misma. Si recordamos la forma de proceder de *ManyToMany*, teníamos una primera etapa en la que se realizaban las propuestas; una vez terminada esa etapa se efectuaba una verificación para saber si las listas de preferencias continuaban siendo estables y, en caso de serlo, se efectuaba por último la etapa de eliminación de rotaciones.

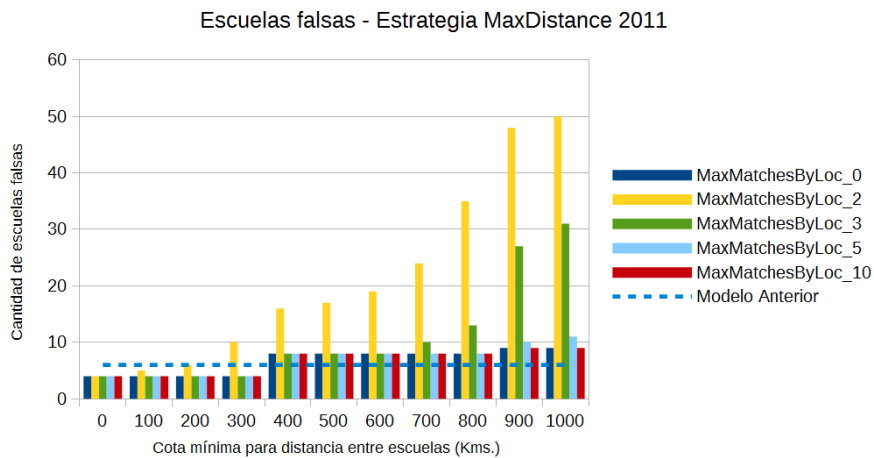
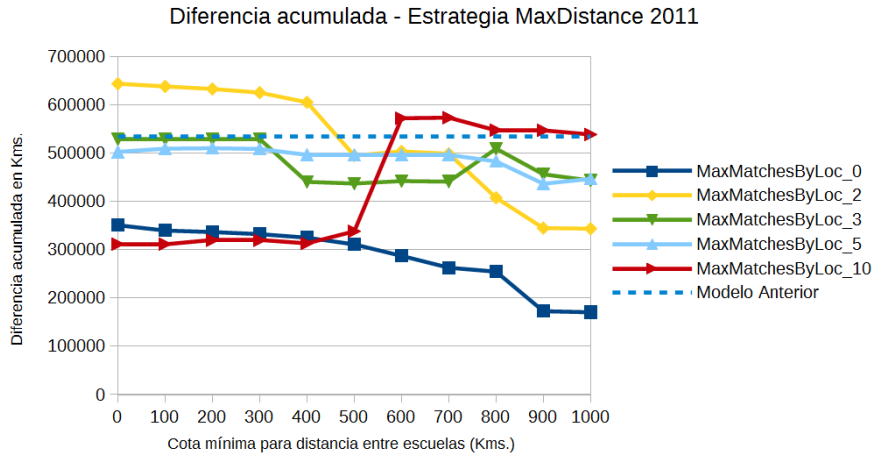
Ante los mismos casos de prueba a los que se expusieron las estrategias *OneToOne*, *MaxDistance* y *MultipleLayers*, nunca logramos terminar el procesamiento de la estrategia, fallando siempre al momento de realizar la verificación de estabilidad posterior a la etapa de propuestas. De modo que por el momento esta estrategia no será considerada para los resultados y quedará pendiente para futuras investigaciones descubrir una variante de la misma que quizás se ajuste mejor al problema de Eratosthenes.

Presentamos entonces a continuación los resultados obtenidos para cada uno de los años, realizando algunas aclaraciones previas que ayudarán al mejor entendimiento de los gráficos.

- Para el cálculo de la diferencia acumulada se utilizaron todos los resultados existentes. Es decir que no se descartaron datos por ser muy lejanos al valor real (outliers), lo cual seguramente afecta al valor final.
- La representación de los valores obtenidos por el modelo anterior con respecto a la diferencia acumulada y la cantidad de escuelas falsas, se visualizarán en los gráficos como rectas constantes con el fin de marcar la cota superior a tener en cuenta para entender si las nuevas estrategias son mejores o no. Pero esto no significa que el modelo anterior haya obtenido el mismo resultado para todas las variantes de distancias mínimas norte-sur entre escuelas. En particular dichos valores están asociados a una distancia de 400km.
- Solo se mostrarán los 5 mejores casos correspondientes a la estrategia *MaxDistance*, con el fin de mejorar la visibilidad y comprensión del gráfico. Estos 5 casos se extraen de los 11 con los que contamos originalmente para cada año, debido a que esta estrategia cuenta con el parámetro extra asociado a la cantidad máxima de parejas que puede formar cada escuela (parámetro al cual hicimos referencia anteriormente).

5.1. Resultados 2011

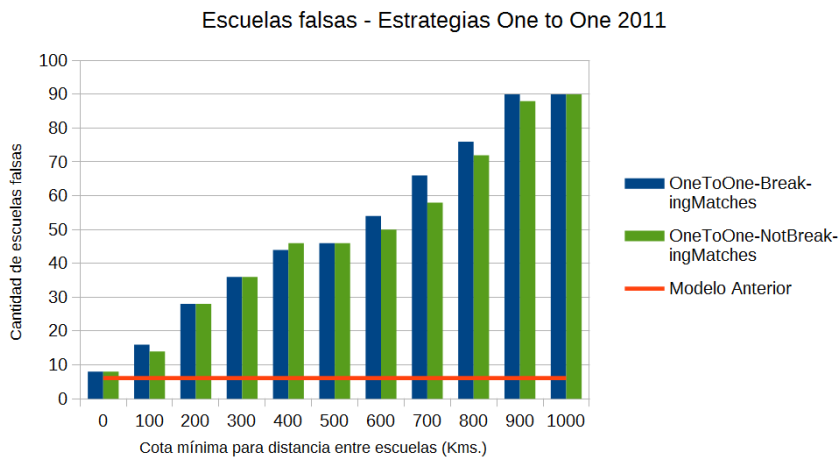
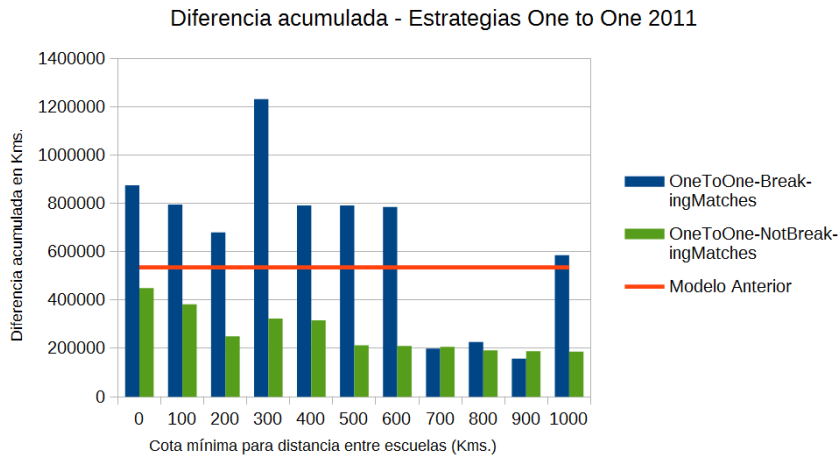
Comenzamos entonces con los gráficos comparativos correspondientes al año 2011, iniciando con aquellos en donde podremos analizar el desempeño respecto de las diferencias acumuladas y cantidad de escuelas falsas creadas, para la estrategia MaxDistance.



Como se puede observar en los gráficos anteriores, los mejores desempeños son aquellos obtenidos por las variantes MaxMatchesByLoc_0 y MaxMatchesByLoc_10. Como explicamos anteriormente, estas variantes hacen referencia a los casos en que no limitamos la cantidad de parejas a conformar por cada escuelas y al caso en que limitamos dicha característica en 10 parejas como máximo respectivamente.

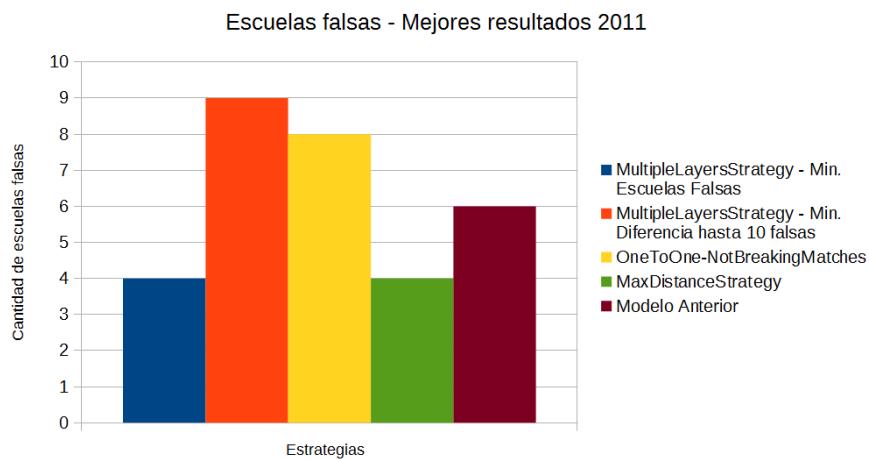
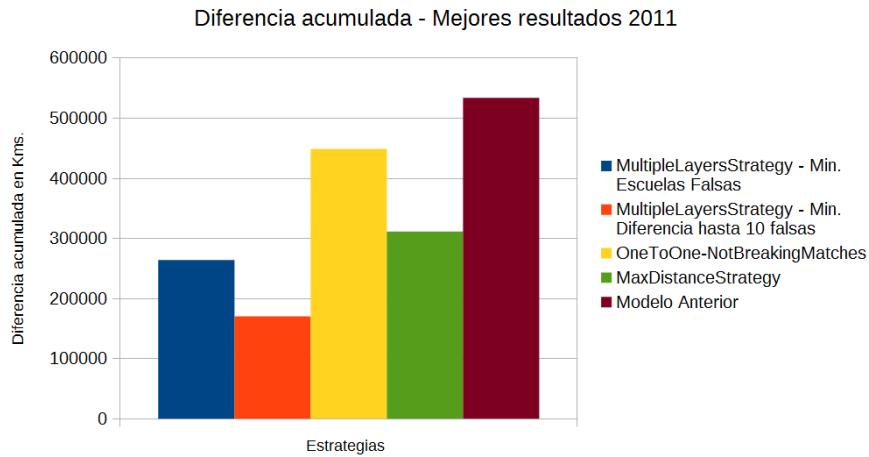
A su vez, al observar el gráfico correspondiente a la creación de escuelas falsas, podemos decir que el mejor desempeño lo tendríamos entre los 0kms y 300kms de distancia mínima norte-sur entre escuelas. Teniendo ambas perspectivas en consideración, la mejor opción sería la asociada a MaxMatchesByLoc_10 considerando 100kms de separación mínima norte-sur, ya que deseamos favorecer el aspecto colaborativo del experimento y los otros casos utilizan una mayor cantidad de escuelas falsas para obtener sus resultados.

Continuando con el resto de las estrategias para el año 2011, mostramos a continuación los resultados obtenidos por las estrategias OneToOne.



A partir de estos gráficos podemos inferir que los mejores resultados, respecto de la diferencia acumulada, son aquellos en donde la distancia mínima entre escuelas está definida entre los 700kms y 900kms. Pero por el contrario, respecto de la cantidad de escuelas falsas, solo sería aceptable considerar aquel caso en donde la distancia mínima entre escuelas es de 0kms, ya que es el único caso en donde dicha cantidad está lo más cerca posible del valor del modelo anterior. Afortunadamente dicho caso también mejora la diferencia acumulada en comparación con el modelo anterior. Por lo tanto, de las estrategias OneToOne, nos quedaremos con el caso de OneToOne-NotBreakingMatches para una distancia mínima norte-sur de 0kms.

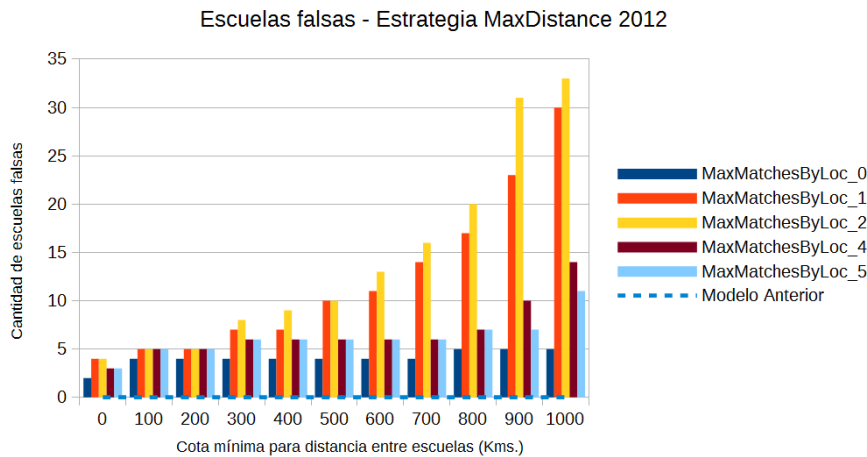
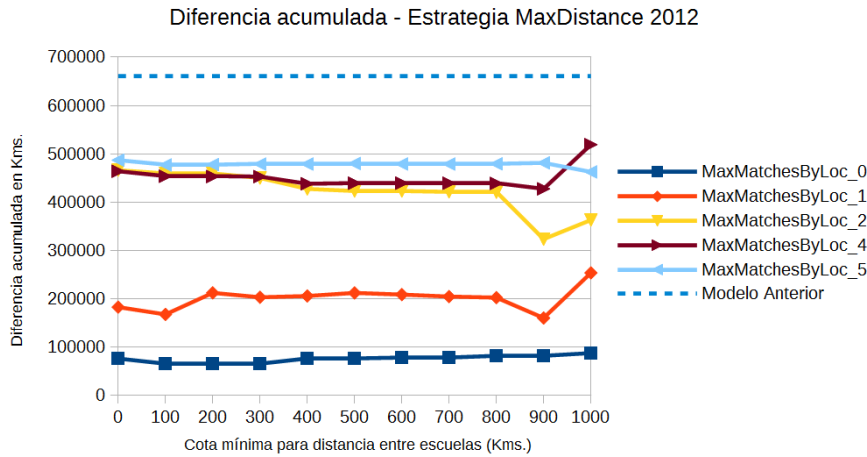
Por último, habiendo realizado un análisis previo de las estrategias MaxDistance y OneToOne y seleccionado los mejores casos para cada una de ellas, vamos a comparar dicha selección con los mejores casos encontrados para la estrategia MultipleLayers.



De estos últimos gráficos, podemos inferir que los resultados obtenidos por la estrategia MultipleLayers, mejoran los resultados obtenidos por el resto de las estrategias. Pero en particular, el primero de ellos necesitó crear una menor cantidad de escuelas falsas, destacándolo por sobre el segundo y definiéndolo como la mejor opción para el año 2011. Más específicamente, este último, fue configurado de la siguiente manera: una división de 11 capas, un gap entre capas de 0 (es decir podemos combinar cualquier capa con cualquier otra), una cantidad máxima de 10 parejas por escuela y usando como estrategia interna a MaxDistance, con una cantidad máxima de 3 parejas por escuela y una distancia mínima norte-sur de 100kms.

5.2. Resultados 2012

Pasando al año 2012, volvemos a analizar en primer lugar los mejores resultados obtenidos para la estrategia MaxDistance.

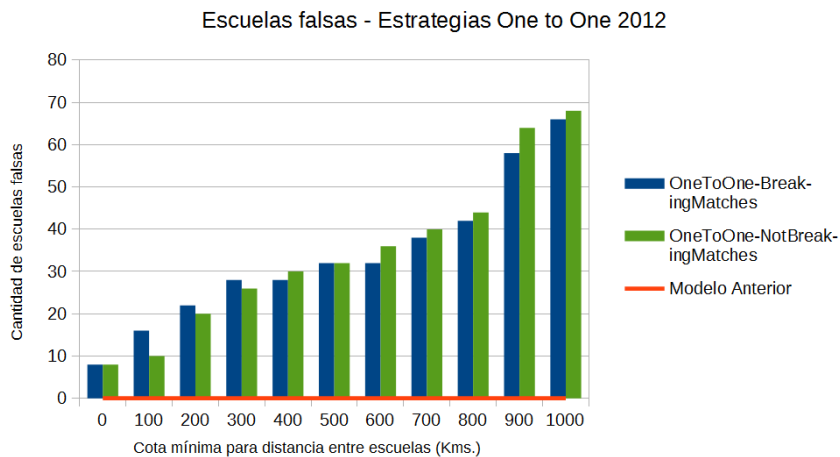
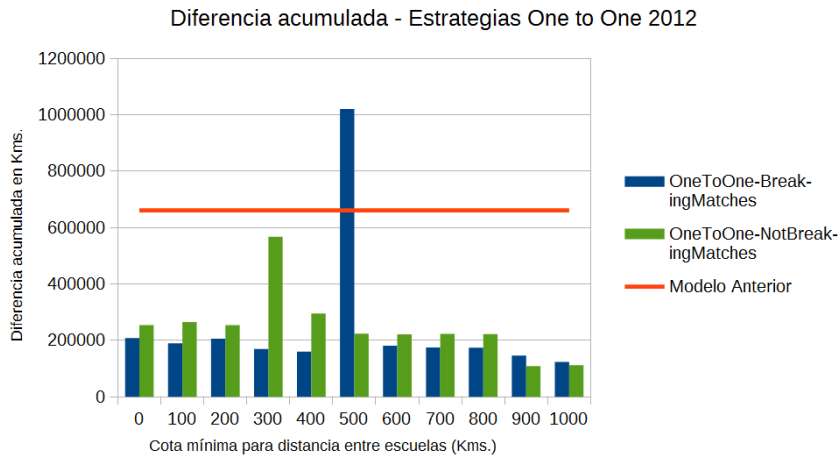


A diferencia del año 2011, las 5 mejores configuraciones de MaxDistance se encuentran por debajo del resultado obtenido por el modelo anterior. Sin embargo, aquella que no limita la cantidad de parejas a conformar por cada escuela, es la que evidencia menor diferencia acumulada.

A su vez, si observamos el segundo gráfico, podemos notar que el modelo anterior no había tenido la necesidad de crear ninguna escuela falsa. Por el contrario, todos los casos de MaxDistance tuvieron que crear, por lo menos, 2 escuelas falsas en el mejor de los casos.

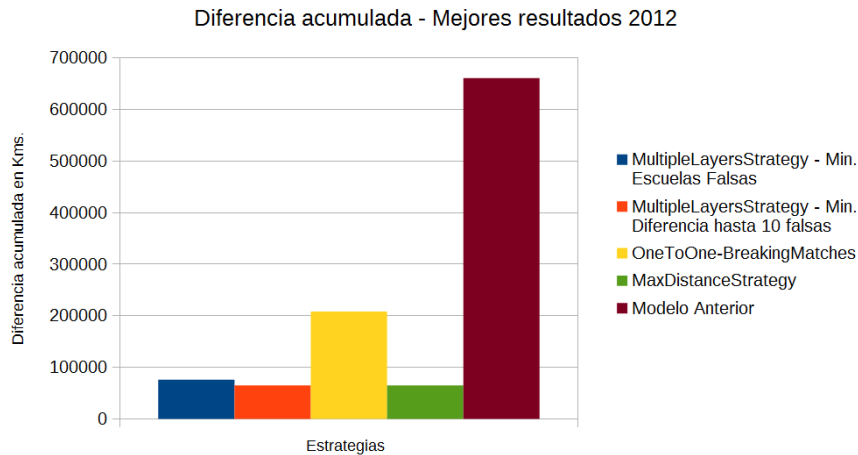
De esto se desprende que, para el año 2012, la mejor configuración para esta estrategia es aquella en donde no limitamos la cantidad de parejas máximas a crear (MaxMatchesBy-Loc.0) y donde la distancia mínima norte-sur entre escuelas es de 300kms, teniendo como consecuencia la creación de 4 escuelas falsas respecto de las 0 creadas por el modelo anterior.

En relación a las estrategias OneToOne, vamos a encontrar muy buenos resultados referidos a la diferencia acumulada, pero continuamos con la tendencia de un índice de creación de escuelas falsas casi exponencial.



La particularidad del gran número de escuelas falsas creadas, nos limita y sesga en el análisis que podamos realizar. A pesar de contar con excelentes resultados en el primer gráfico, muchos de ellos conllevan la creación de una gran cantidad de escuelas. Esto es interpretado de forma negativa ya que el objetivo del proyecto es fomentar que las escuelas trabajen de forma colaborativa. En consecuencia, las mejores opciones con las que contamos, son aquellas en donde la distancia mínima norte-sur entre escuelas es de 0kms, caso en el cual tenemos la menor cantidad de escuelas falsas. En dicho caso, la mejor de las estrategias es la OneToOne-BreakingMatches.

Comparemos entonces estos dos últimos resultados con los obtenidos en la estrategia MultipleLayers. Nuevamente vamos a poder observar dos variantes respecto a esta última estrategia, que minimizan tanto la creación de escuelas falsas como la diferencia acumulada.

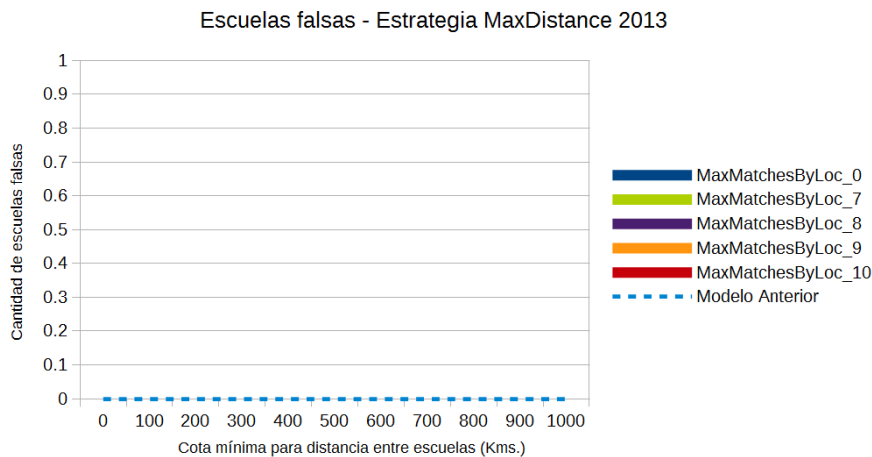
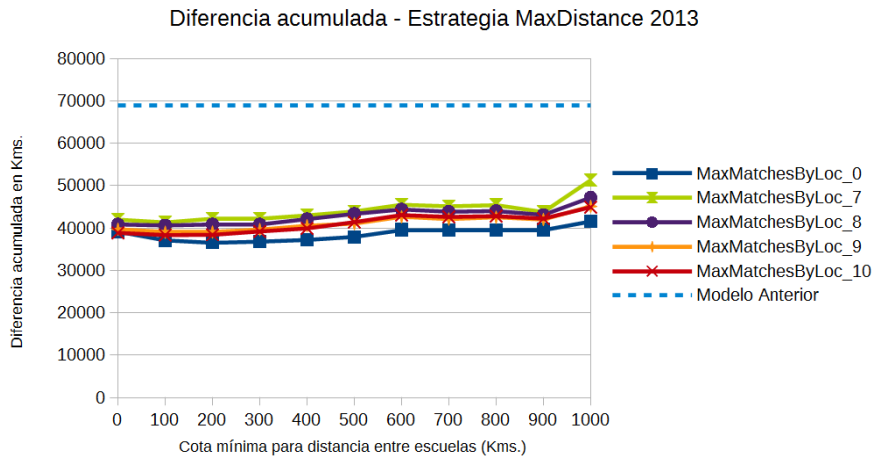


Nuevamente las estrategias MultipleLayers demuestran un buen desempeño en ambos aspectos, igualando a la estrategia MaxDistance respecto de la diferencia acumulada y mejorando por sobre todas las demás respecto de las escuelas falsas. En particular, el caso en que minimizamos principalmente la construcción de escuelas falsas para luego minimizar la diferencia acumulada, presenta un mejor comportamiento si consideramos ambos gráficos.

Específicamente, el detalle de la configuración de dicha estrategia es el siguiente: dividimos a las escuelas en 2 capas, considerando un gap de 0 para los cruces entre las mismas y sin limitar la cantidad de parejas a conformar por cada escuela. Además, internamente contamos con la estrategia MaxDistance, a la cual tampoco le limitamos la cantidad de parejas para cada escuela y tampoco impondremos un límite inferior a los kms necesarios norte-sur para emparejar a las mismas.

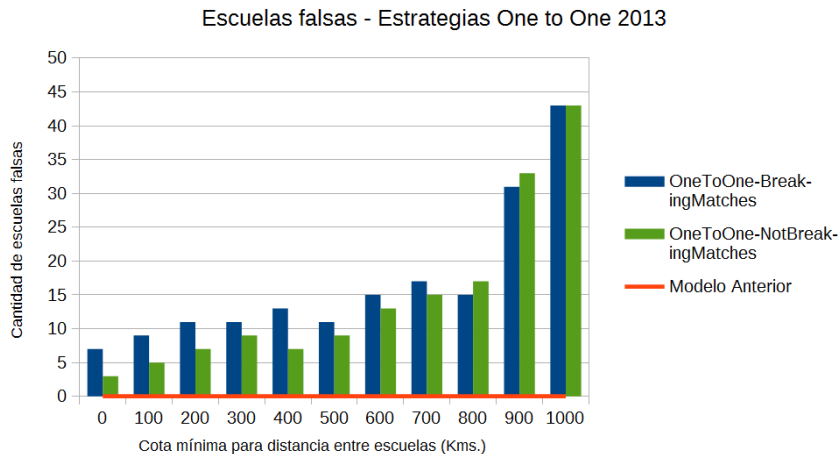
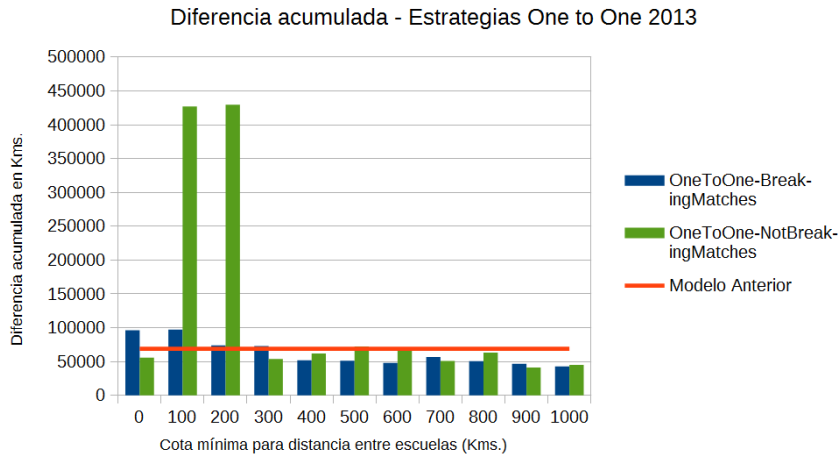
5.3. Resultados 2013

Para el año 2013, la relación entre los resultados del modelo anterior y los nuevos provistos por la estrategia MaxDistance, siguen mostrando grandes diferencias a favor de las nuevas estrategias. A diferencia de los resultados correspondientes al año anterior, en este caso todas las variantes se encuentran compactas y similares.



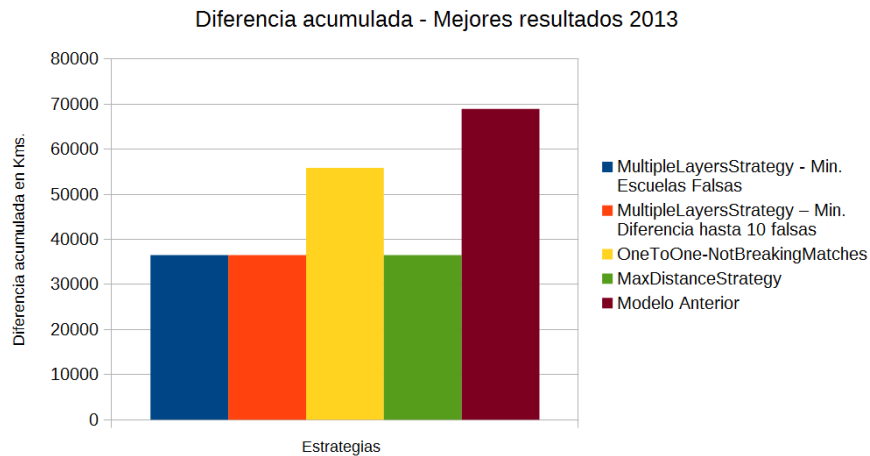
Si bien todas aparentan tener la misma distribución, la correspondiente a MaxMatchesByLoc.0 toma una leve ventaja por sobre el resto. Para el caso de la creación de escuelas falsas, si bien el gráfico parece no aportar demasiada información, representa a su vez un panorama ideal en el cual nuestro sistema fue capaz de emparejar a todas las escuelas sin necesidad de crear nuevas. Por lo que, para el año 2013, la decisión solo quedará en función del primer gráfico. Razón por la cual, nos quedaremos con la configuración correspondiente a MaxMatchesByLoc.0, con una distancia mínima nort-sur de 200kms.

Respecto de las estrategias OneToOne, el año 2013 no parece ser la excepción en cuanto a su comportamiento respecto de los años anteriores. Si bien los márgenes de diferencia con relación al modelo anterior son menores en cuando a la diferencia acumulada, el factor determinante es el que se extrae del segundo gráfico.



Si bien la curva de crecimiento tiene un quiebre llegando a los 500-600 kms (mientras que en años anteriores a los 200kms ya teníamos valores importantes), el problema se acentúa aún más por el hecho de que el modelo anterior no tuvo necesidad de crear ninguna. Bajo estas condiciones, el mejor resultado que podemos extraer es el correspondiente a OneToOne-NotBreakingMatches tomando 0kms como cota mínima para la distancia nort-sur entre escuelas.

Finalmente observemos entonces la comparativa considerando los resultados de la estrategia MultipleLayers y los resultados anteriormente destacados para las estrategias MaxDistance y OneToOne.

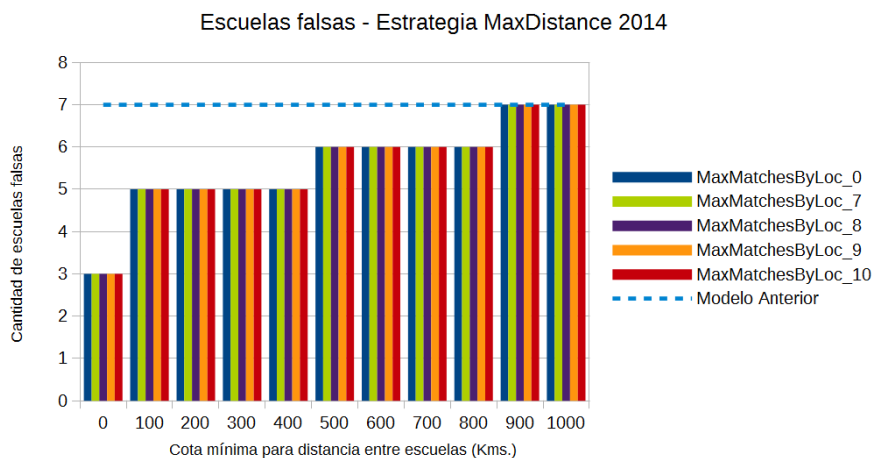
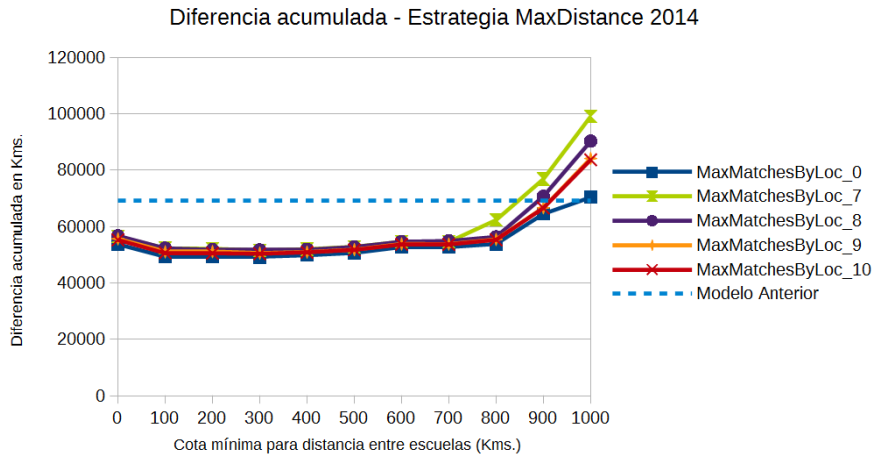


Como era de esperarse, ambas estrategias `MultipleLayers` optaron por utilizar como estrategia interna a `MaxDistance`, dado que bajo esa estrategia no tuvimos la necesidad de crear escuelas falsas. Es por ello que el segundo gráfico se observa casi sin información, excepto por el valor de la estrategia `OneToOne`.

Respecto de la diferencia acumulada, tenemos una paridad para las estrategias `MultipleLayers` y `MaxDistance`. Si observamos el detalle de las configuraciones de dichas estrategias, nos daremos cuenta de que básicamente es la misma en los tres casos. El algoritmo en este caso se mostró consistente en su elección, optando por la estrategia `MaxDistance` con `MaxMatchesByLoc_0` y 200kms de distancia norte-sur. De modo que, podemos considerar que en este caso el uso de la estrategia `MultipleLayers` no cumple función alguna más que la de un encapsular a `MaxDistance`, razón por la cual descartaremos dichas opciones y nos quedaremos simplemente con la correspondiente a `MaxDistance`.

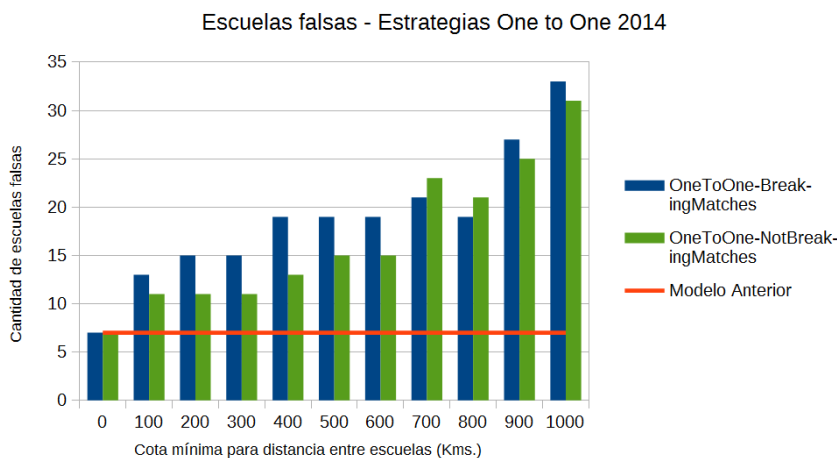
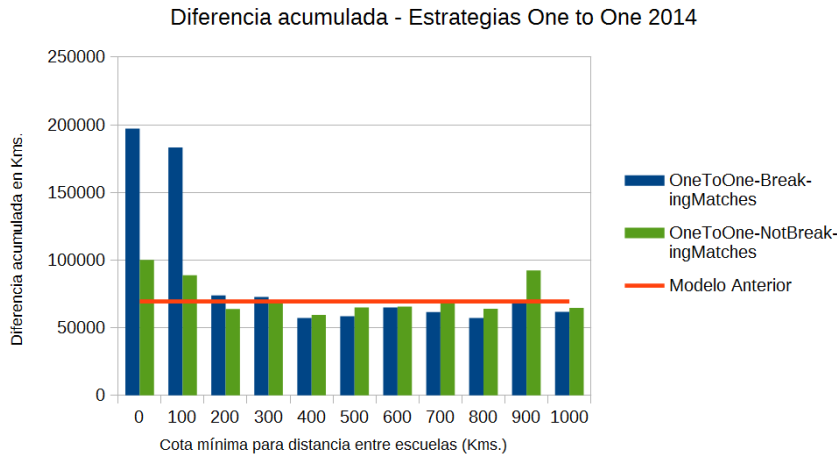
5.4. Resultados 2014

En el año 2014, volvemos a encontrar un desempeño muy parejo para las distintas variantes de MaxDistance, mostrando mejoras en ambos gráficos respecto del modelo anterior.



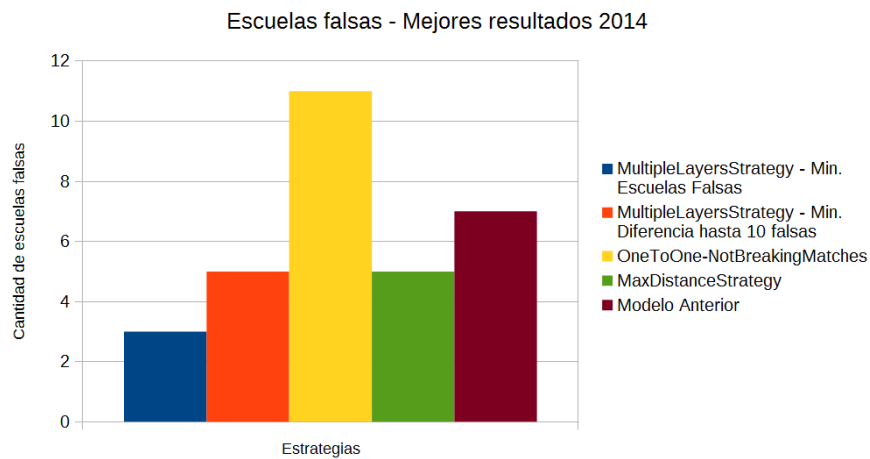
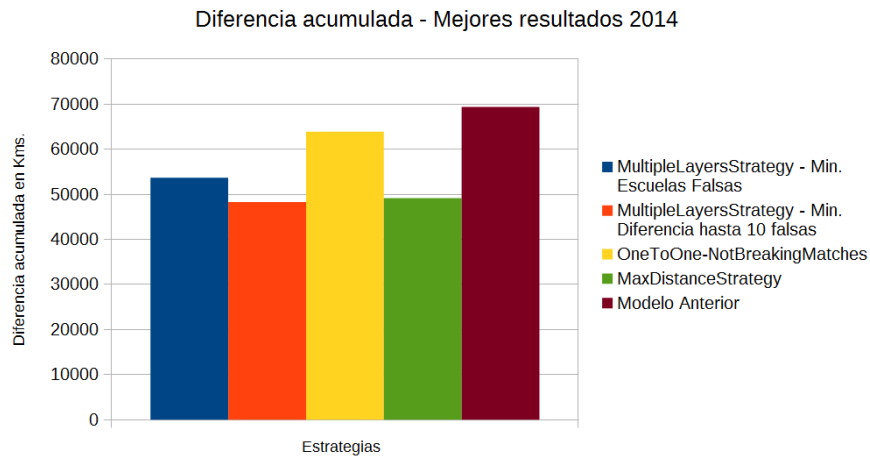
Si bien la opción MaxMatchesByLoc_0 con 0kms de distancia norte-sur es la que menos escuelas falsas crea, los subsiguientes casos que abarcan entre los 100kms y 400kms, obtienen mejores resultados para la diferencia acumulada. Como, para dichos casos, aún nos encontramos por debajo de la cantidad de escuelas falsas creadas por el modelo anterior, optamos por mejorar un poco el segundo aspecto en detrimento del primero. Es por ello que elegimos como mejor opción a aquella correspondiente a MaxMatchesByLoc_0 con 300kms de distancia mínima norte-sur.

De la misma forma, los gráficos correspondientes a la estrategia OneToOne, vuelven a evidenciar problemas. Si bien contamos con un caso en donde se iguala la cantidad de escuelas falsas creadas por el modelo anterior, ese mismo caso genera una diferencia acumulada mayor. Resulta inevitable entonces tener que sacrificar uno de dichos aspectos en beneficio del otro, tal como hicimos con la estrategia anterior.



Optamos entonces por el caso de OneToOne-NotBreakingMatches con 200kms de distancia norte-sur entre escuelas. Como podemos observar dicho caso crea cerca del doble de escuelas falsas respecto del modelo anterior. Sin embargo, creemos es la mejor opción que nos provee este tipo de estrategias para este año.

Unificando dichas observaciones con los casos seleccionados para MultipleLayers, se observa que la mejor opción es quedarnos nuevamente con la combinación entre las estrategias MultipleLayers y MaxDistance. Más específicamente, aquella en donde minimizamos la diferencia acumulada (segunda opción en el gráfico).

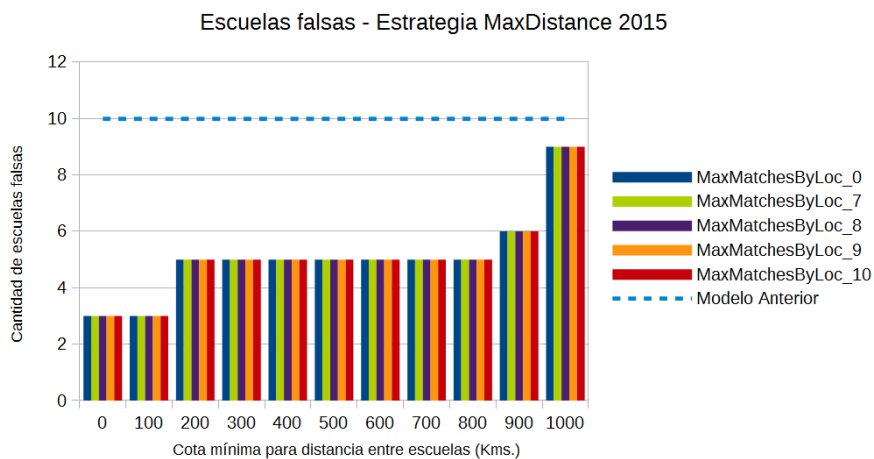
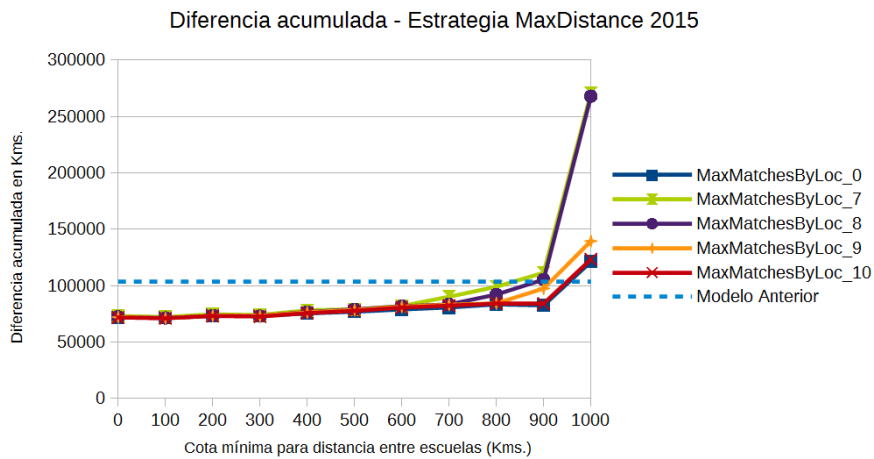


Si bien la primer opción genera una menor cantidad de escuelas, la diferencia acumulada es mayor a la obtenida por la mencionada anteriormente. Además como aún nos mantenemos por debajo de la cantidad creada por el modelo anterior, entendemos que es un margen que resulta beneficioso adoptar.

La configuración seleccionada para el año 2014, será: conformar 16 capas, no permitiendo cruces de capas consecutivas (un gap igual a 1) y con una cantidad de parejas máxima de 9 por escuela. Internamente contaremos con una estrategia MaxDistance, la cual permitirá 3 parejas por escuela y una distancia norte-sur mínima de 300kms.

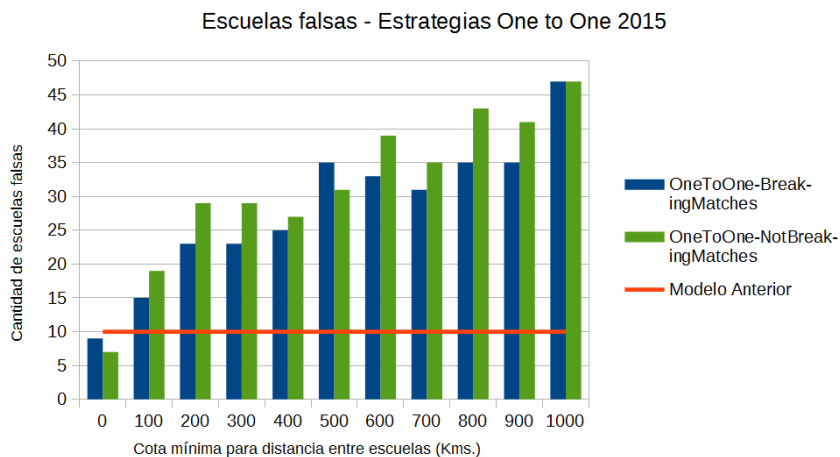
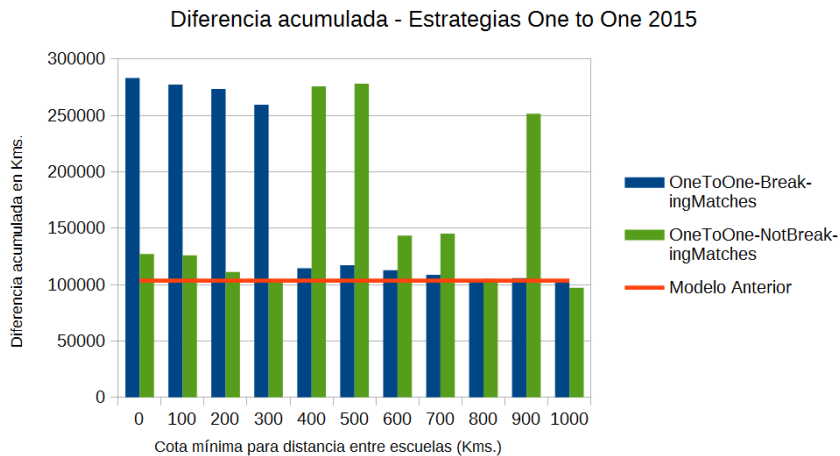
5.5. Resultados 2015

Para el año 2015, observamos el siguiente comportamiento.



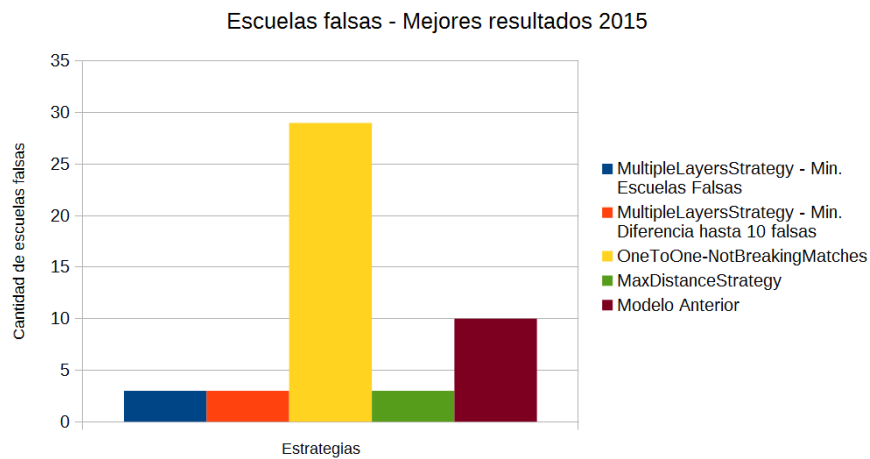
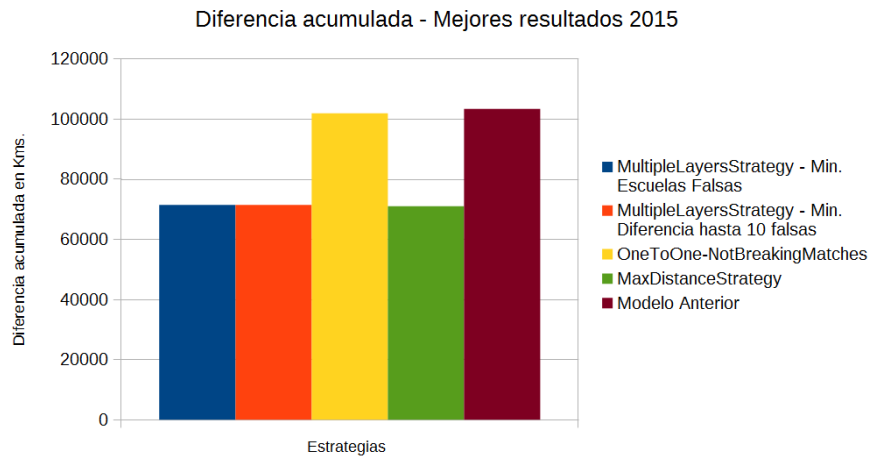
Nuevamente evidenciando resultados por debajo de los obtenidos en el modelo anterior, la estrategia MaxDistance cumple con el objetivo buscado. Creando a su vez un bajo índice de escuelas falsas, las distintas variantes de la estrategia tienen un comportamiento muy similar. Sin embargo, la que obtuvo una leve ventaja por sobre el resto en este caso, es la MaxMatchesByLoc_10 utilizando una cota mínima de 100kms.

Observemos entonces lo obtenido a partir de las estrategias OneToOne.



Al igual que en el 2014, la estrategia OneToOne no parece identificarse como una buena opción a utilizar. En los pocos casos en los que la diferencia acumulada se encuentra por debajo del valor obtenido en el modelo anterior, la cantidad de escuelas que fueron necesarias crear es el doble o el triple de lo esperado. Bajo estas condiciones se dificulta poder optar por una variante, pero si priorizamos la diferencia acumulada, entonces nos quedaremos con OneToOne-NotBreakingMatches y 300kms de distancia norte-sur.

Finalmente nos queda solo analizar los resultados obtenidos por la estrategia MultipleLayers y compararlo con los anteriores.

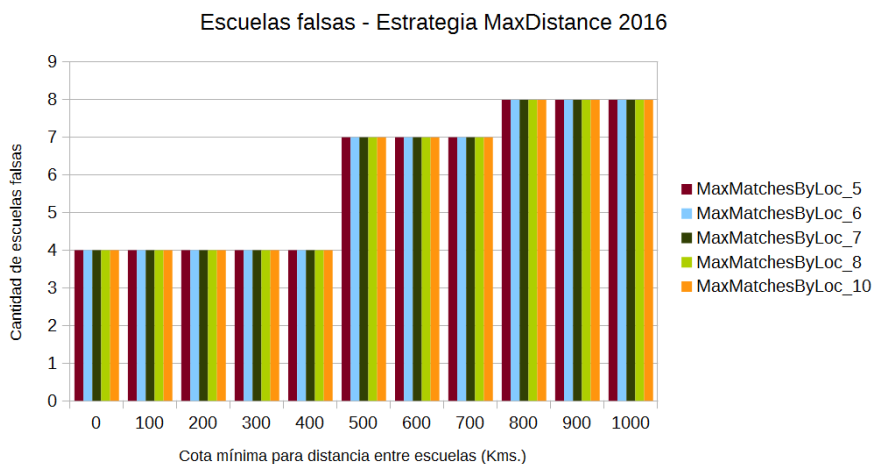
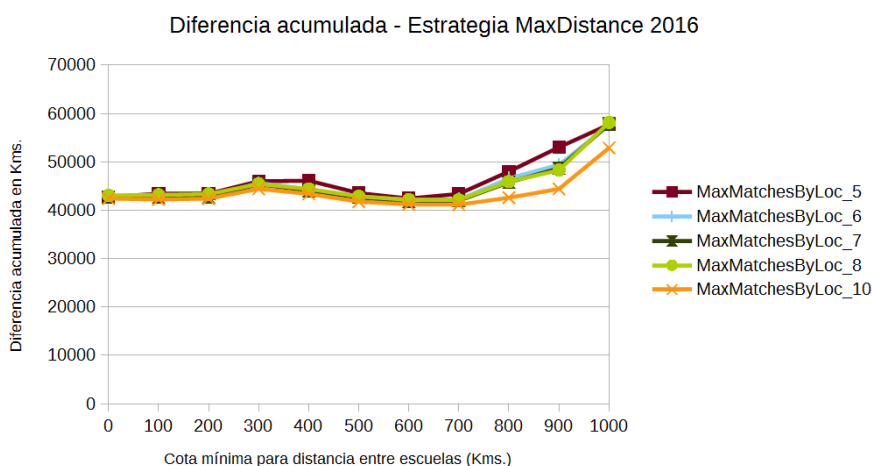


Observando estos últimos gráficos, pareciera existir una paridad entre las estrategias MaxDistance y MultipleLayers. Sin embargo la diferencia acumulada para la primera de ellas, es levemente inferior. Por lo que para el año 2015, la mejor estrategia a utilizar es la MaxDistance en su versión MaxMatchesByLoc_10 con 100kms de distancia mínima norte-sur y 3 escuelas falsas creadas.

5.6. Resultados 2016

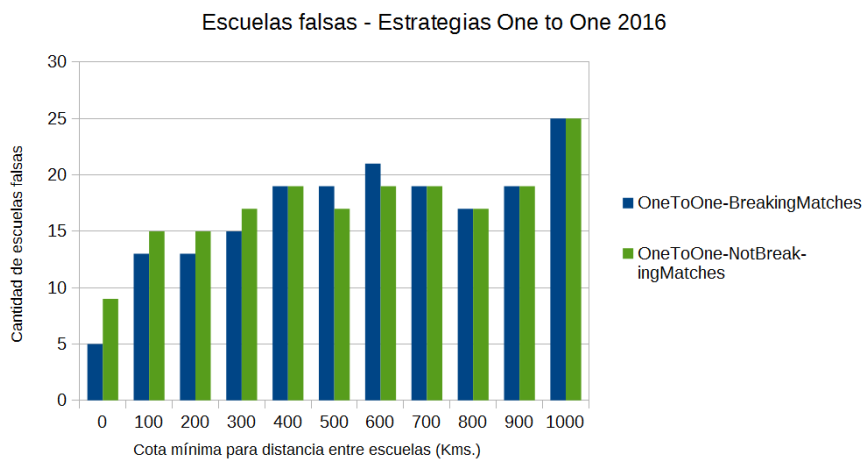
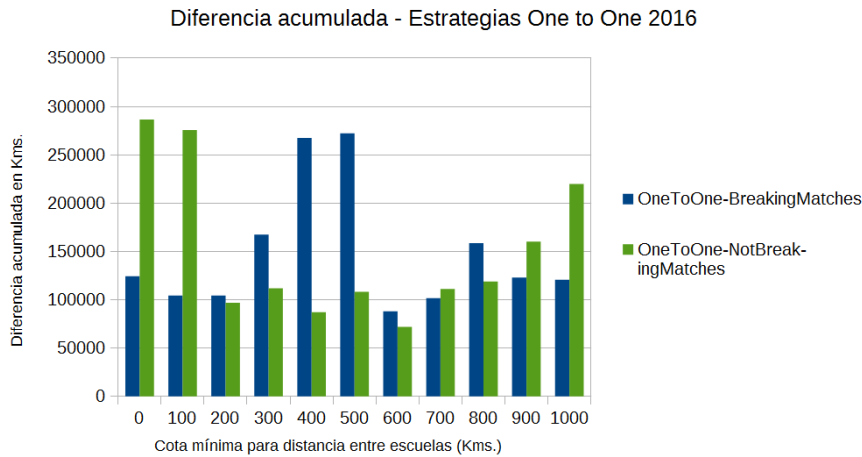
Finalmente para el año 2016, última realización del Proyecto Eratosthenes a la fecha, se decidió utilizar únicamente nuestro nuevo sistema con el fin de ponerlo a prueba. Es a causa de esto, que para este año no contaremos con la comparación de resultados respecto del modelo anterior. Solamente se hará un análisis respecto de las distintas estrategias planteadas en este trabajo.

Comenzamos entonces observando los resultados asociados a la estrategia MaxDistance.



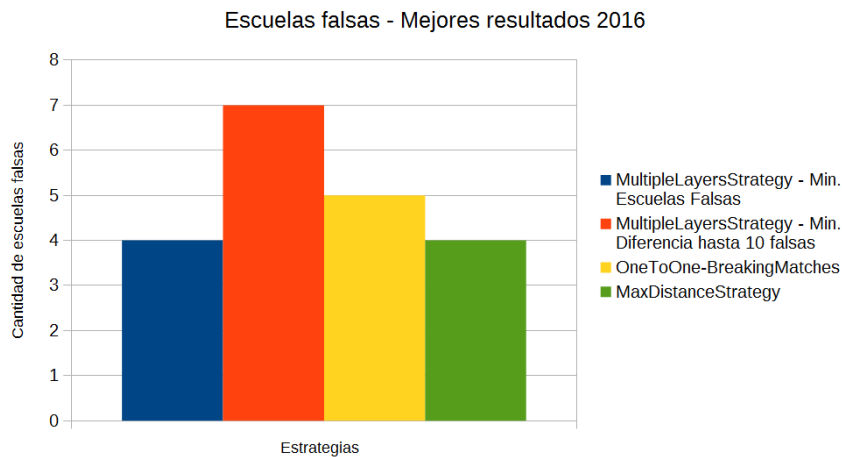
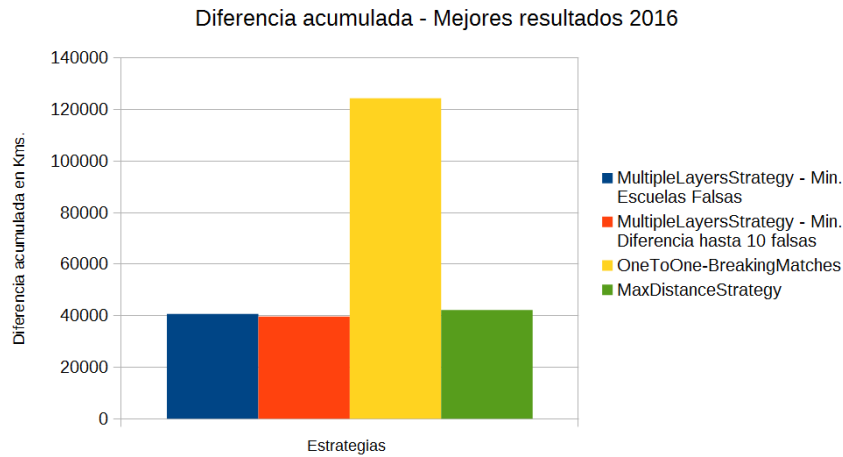
Observando ambos gráficos podemos concluir que el mejor valor respecto de la diferencia acumulada se encuentra entre los 600kms y 700kms de distancia mínima entre escuelas. Sin embargo, para dichos casos, la cantidad de escuelas falsas creadas es mayor. Por lo que optamos por sacrificar un poco el valor de la diferencia acumulada en favor de la cantidad de escuelas falsas, y nos quedaremos con la opción asociada a MaxMatchesByLoc_10 utilizando una cota mínima de 100kms.

Por otro lado para el caso de las estrategias OneToOne, se observa



Al igual que en muchos de los años anteriores, los mejores resultados de uno de los gráficos no se condicen con los mejores resultados del otro. Si en uno de ellos los resultados mejoran, suele suceder lo contrario en el otro. Es por ellos que para esta estrategia elegimos quedarnos con la especialidad OneToOne-BreakingMatches en su configuración de 0kms de cota mínima para la distancia entre escuelas.

Finalmente, recopilando todos los resultados de este año, podemos observar aquellos mencionados anteriormente junto con los obtenidos para la estrategia MultipleLayers.



De estos se desprende que existe una leve mejora por parte de las estrategias MultipleLayers respecto del resultado obtenido por la estrategia MaxDistance. Particularmente para el caso en donde minimizamos la creación de escuelas falsas, la diferencia acumulada es menor y se mantiene el mismo índice de creación de escuelas falsas.

Más específicamente, la configuración seleccionada para el año 2016, será: conformar 14 capas, no permitiendo cruces de capas consecutivas (un gap igual a 1) y con una cantidad de 5 parejas máximas por escuela. Internamente contaremos con una estrategia MaxDistance, la cual permitirá también hasta 5 parejas por escuela y una distancia norte-sur mínima de 200kms.

6. CONCLUSIONES

Habiendo realizado las pruebas correspondientes a lo largo de los 6 años, pudimos observar en la sección anterior que las nuevas estrategias mostraron mejoras en todas las características que definimos en un comienzo.

Si bien una de las estrategias ideadas no pudo ser puesta a prueba realmente como deseábamos, el resto obtuvo sus resultados y fuimos capaces de compararlos contra los obtenidos por el modelo anterior. A pesar de que en la mayoría de los casos concluimos que la estrategia más conveniente era MaxDistance o en su defecto la combinación MaxDistance + MultipleLayers, esto no implica que las estrategias OneToOne hayan resultado deficientes en su totalidad. Estas últimas lograron obtener buenos resultados respecto de las diferencias acumuladas, pero presentaron problemas a la hora de bajar el nivel de escuelas falsas creadas.

Por otro lado la estrategia MultipleLayers demostró ser eficiente al ser combinada con el resto de las estrategias. En muchos de los años, al combinarse particularmente con la estrategia MaxDistance, lograba decrementar aún más la diferencia acumulada denotando haber conformado parejas más óptimas con resultados más próximos al real.

A modo de resumen de las mejoras obtenidas, obtuvimos los porcentajes de mejora para cada uno de los años, con el fin de hacer gráfico lo anteriormente enumerado.

Año	Diferencia acumulada nueva	Diferencia acumulada anterior	Mejora
2011	264.082,07	533.838,94	50,53%
2012	76.240,63	661.275,37	88,47%
2013	36.510,84	68.948,47	47,05%
2014	48.301,74	69.379,26	30,38%
2015	71.068,43	103.422,32	31,28%
2016	40.685,23	N/A	N/A

A partir de estos porcentajes podemos decir que en promedio tenemos una mejora del 49,54 %, lo cual consideramos más que satisfactorio y lo cual creemos que se puede seguir mejorando.

Resta entonces para futuras investigaciones, lograr encontrar una optimización que logre bajar el índice de creación de escuelas falsas para las estrategias OneToOne y también buscar la manera de intentar ajustar la estrategia ManyToMany con el fin de conocer si es capaz de encontrar mejores resultados a los ya obtenidos.

Respecto al desarrollo del sistema propiamente dicho, logramos obtener una herramienta sencilla y de fácil utilización, la cual podrá ser utilizada directamente por los organizadores del evento, sin que estos necesiten de la asistencia técnica de una tercera persona. Esto les brindará independencia a la hora de poder realizar la cantidad de pruebas que crean necesarias y en el momento que lo deseen, evitando retrasos y demoras producidas al tener que coordinar con otras personas.

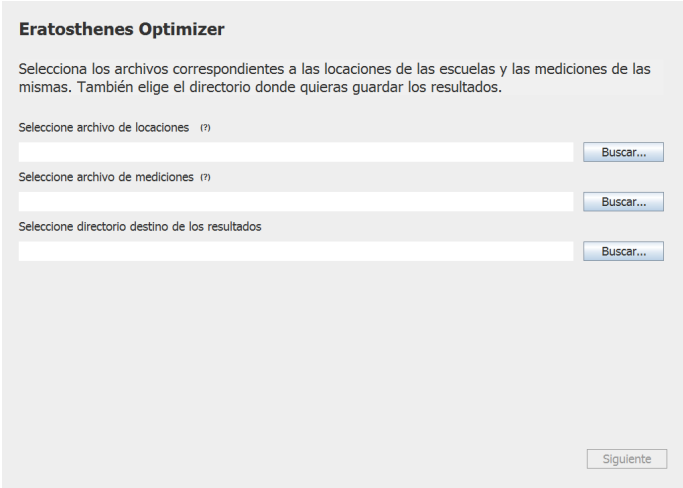
A su vez logramos el desarrollo de un sistema el cual puede ser utilizado durante todo el año. Esto se debe a que, como explicamos anteriormente, durante los solsticios el sol preserva la misma posición durante un margen mayor de días. Esto nos permitiría relajar la restricción impuesta sobre las escuelas para que solo formen parejas con aquellas otras que midieron en el mismo día. Sin embargo, si el sistema funciona durante los equinoccios

en donde necesitamos que las parejas de escuelas hayan realizado sus mediciones en el mismo día, se desprende que también funcionará para el período de solsticios en donde no existe tal restricción.

De esta forma le aportamos mayor generalidad al sistema y podemos afirmar que si obtuvimos buenos resultados para el caso de los equinoccios, la misma mejora debería presentarse en caso de utilizar el sistema durante los solsticios.

7. ANEXO

A continuación presentaremos algunas capturas de pantalla del sistema a modo explicativo, comenzando con la pantalla principal que el usuario se encontrará al ejecutarlo.



Eratosthenes Optimizer

Selecciona los archivos correspondientes a las locaciones de las escuelas y las mediciones de las mismas. También elige el directorio donde quieras guardar los resultados.

Selecciónese archivo de locaciones (?)

Selecciónese archivo de mediciones (?)

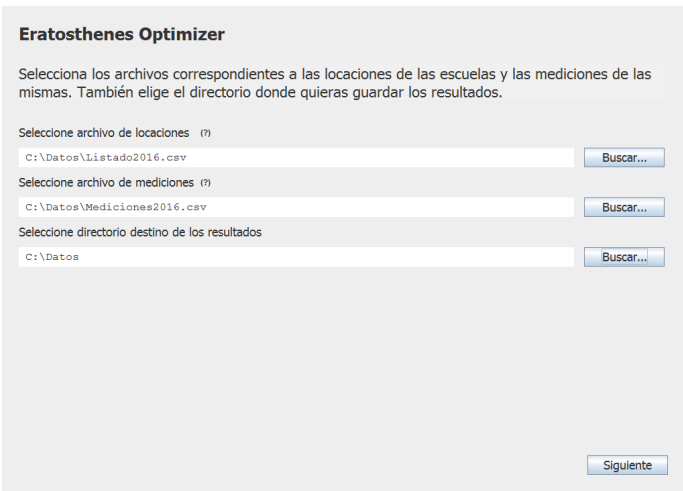
Selecciónese directorio destino de los resultados

Buscar... (3x)

Siguiente

Como se puede observar se presentan tres campos en los cuales debemos ingresar las rutas de los archivos que contienen la información asociada al listado de escuelas y las mediciones efectuadas por las mismas, como también el directorio destino donde deseamos dejar los resultados del análisis.

Haciendo click en el botón *Buscar* se desplegará una ventana que nos permitirá navegar nuestro sistema de archivos para encontrar los mismos. Una vez ingresados los datos, se habilitará el botón para continuar



Eratosthenes Optimizer

Selecciona los archivos correspondientes a las locaciones de las escuelas y las mediciones de las mismas. También elige el directorio donde quieras guardar los resultados.

Selecciónese archivo de locaciones (?)

C:\Datos\Listado2016.csv

Selecciónese archivo de mediciones (?)

C:\Datos\Mediciones2016.csv

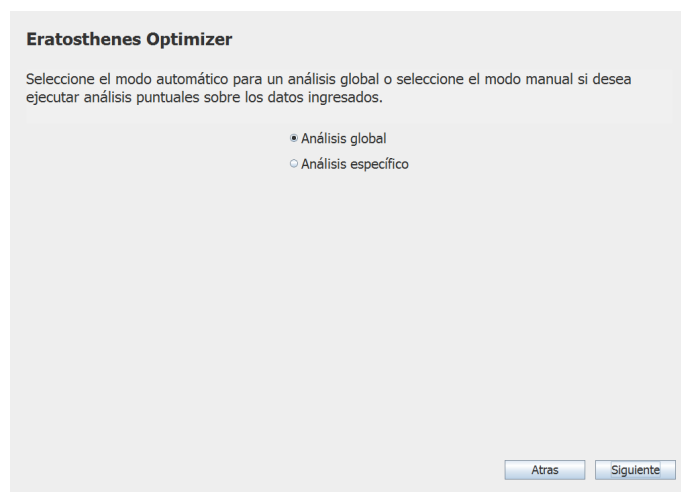
Selecciónese directorio destino de los resultados

C:\Datos

Buscar... (3x)

Siguiente

Una vez que pasamos a la siguiente pantalla, se nos presentarán dos opciones. O bien realizar el *Análisis global* o bien un *Análisis específico*.



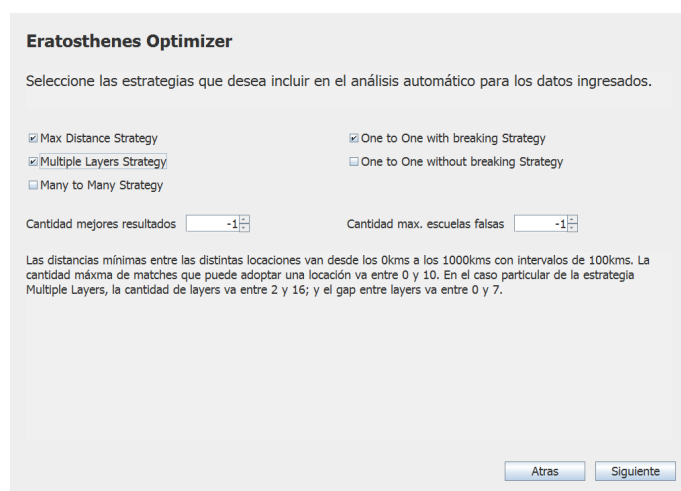
Eratosthenes Optimizer

Seleccione el modo automático para un análisis global o seleccione el modo manual si desea ejecutar análisis puntuales sobre los datos ingresados.

Análisis global
 Análisis específico

Atras Siguiete

Si optamos por realizar un *Análisis global* y hacemos click en el botón *Siguiete*, veremos una serie de opciones para configurar dicho análisis.



Eratosthenes Optimizer

Seleccione las estrategias que desea incluir en el análisis automático para los datos ingresados.

Max Distance Strategy
 Multiple Layers Strategy
 Many to Many Strategy

One to One with breaking Strategy
 One to One without breaking Strategy

Cantidad mejores resultados
Cantidad max. escuelas falsas

Las distancias mínimas entre las distintas locaciones van desde los 0kms a los 1000kms con intervalos de 100kms. La cantidad máxima de matches que puede adoptar una locación va entre 0 y 10. En el caso particular de la estrategia Multiple Layers, la cantidad de layers va entre 2 y 16; y el gap entre layers va entre 0 y 7.

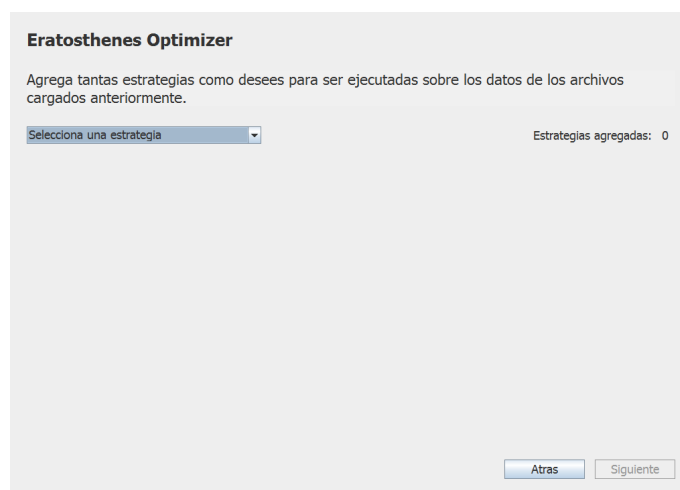
Atras Siguiete

En primer lugar podemos observar que se listan todas las estrategias disponibles, a partir de las cuales el usuario puede ir seleccionando aquellas que desea incluir dentro del análisis a ejecutar. A su vez se ofrecen dos campos numéricos los cuales limitan el accionar del análisis.

El primero de ellos limita la cantidad de resultados que vamos a obtener al finalizar la ejecución. En caso de dejar un valor de -1, implica que deseamos obtener todos los resultados. Caso contrario se debe ingresar otro número mayor a cero.

El segundo nos permite limitar la cantidad de escuelas falsas a crear. En caso de dejar un valor de -1, implica que no deseamos limitar esa propiedad. Caso contrario debemos ingresar un número mayor a cero y se descartarán aquellos casos en donde los resultados superen ese valor.

Por otro lado, si decidimos hacer un *Análisis específico*, observaremos otra pantalla distinta.

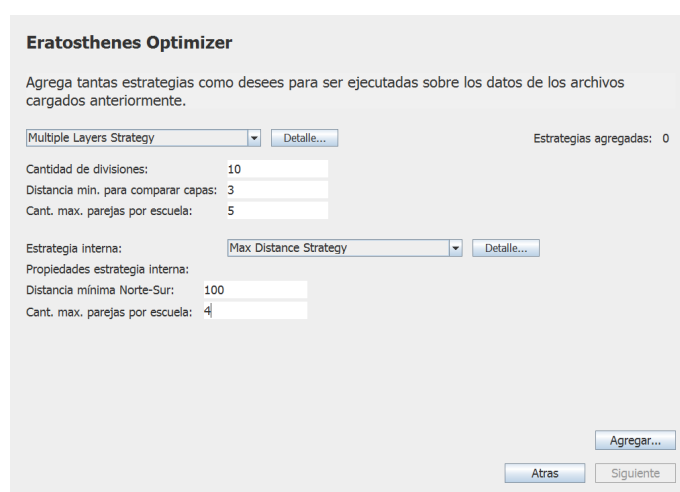


The screenshot shows the 'Eratosthenes Optimizer' window. At the top, it says 'Eratosthenes Optimizer' and 'Agrega tantas estrategias como desees para ser ejecutadas sobre los datos de los archivos cargados anteriormente.' Below this is a dropdown menu labeled 'Selecciona una estrategia' and a counter 'Estrategias agregadas: 0'. At the bottom right, there are two buttons: 'Atras' and 'Siguiete'.

En esta pantalla se nos permitirá armar nuestro propio set de estrategias a ejecutar como deseemos. En primer lugar debemos seleccionar la estrategia a utilizar. Al hacerlo se nos desplegarán nuevos campos de texto mediante los cuales terminaremos de configurar la estrategia propiamente dicha.

Una vez finalizamos de configurar la misma, se nos habilitará el botón de *Agregar* dicha estrategia al set a ejecutar. Podemos entonces repetir los pasos anteriores cuantas veces queramos, agregando tantas como deseemos.

A modo de ejemplo mostramos un caso en el que se está construyendo una estrategia *MultipleLayers* utilizando a *MaxDistance* como estrategia interna.



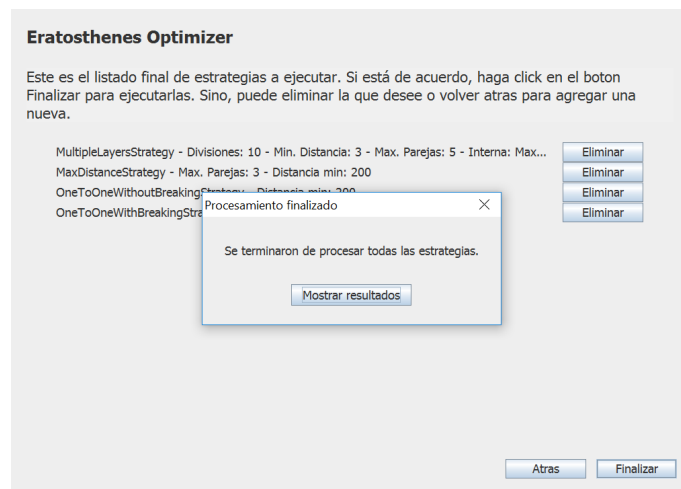
The screenshot shows the 'Eratosthenes Optimizer' window with the 'Multiple Layers Strategy' selected. It includes a 'Detalle...' button next to the strategy name. The configuration fields are: 'Cantidad de divisiones: 10', 'Distancia min. para comparar capas: 3', and 'Cant. max. parejas por escuela: 5'. Below these, the 'Estrategia interna:' is set to 'Max Distance Strategy' with another 'Detalle...' button. Further configuration includes 'Propiedades estrategia interna:', 'Distancia mínima Norte-Sur: 100', and 'Cant. max. parejas por escuela: 4'. At the bottom right, there are three buttons: 'Agregar...', 'Atras', and 'Siguiete'.

Cuando terminemos de agregar las mismas, entonces si podremos hacer click en el botón *Siguiete* y pasaremos a observar la siguiente pantalla.



Esta pantalla nos permite confirmar las estrategias a ejecutar y es acá donde veremos listadas todas aquellas estrategias que hayamos ido agregando en el paso anterior. Si nos arrepentimos de alguna de las mismas, podemos eliminarla directamente desde esta pantalla, o bien volver a la anterior para continuar agregando nuevas.

Finalmente si hacemos click en el botón *Finalizar*, procederemos a ejecutar todas las estrategias y en caso de éxito nos mostrará un mensaje como el siguiente.



Si hacemos click en el botón *Mostrar resultados* nos abrirá en una nueva ventana el destino de los resultados que habíamos configurado en la pantalla principal. Desde ahí podremos acceder a las planillas con los resultados finales.

Bibliografía

- [1] Victoria Bekeris, Flavia Bonomo, Edgardo Bonzi, Beatriz García, Guillermo Mattei, Diego Mazzitelli, Silvina Ponce Dawson, Constanza Sánchez Fernández de La Vega, and Francisco Tamarit. Eratosthenes 2009/2010: An old experiment in modern times. *Astronomy Education Review*, 10(1), 2011.
- [2] David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.
- [3] Robert W Irving. An efficient algorithm for the “stable roommates” problem. *Journal of Algorithms*, 6(4):577–595, 1985.
- [4] Sandy Scott. *A study of stable marriage problems with ties*. PhD thesis, University of Glasgow, 2005.
- [5] Robert W Irving, David F Manlove, and Sandy Scott. The hospitals/residents problem with ties. In *Scandinavian Workshop on Algorithm Theory*, pages 259–271. Springer, 2000.