CREACIÓN Y OPERABILIDAD
DE UNA BASE DE DATOS DE
ADN
DISTRIBUIDA
MEDIANTE EL USO DE UN SISTEMA GRID

- **CREACIÓN DE ÁRBOLES DE SUFIJOS**
- **BÚSQUEDA DE PATRONES HABITUALES**
- **COMPRESIÓN DE MOLÉCULAS DE ADN**
- **ESTUDIOS EVOLUTIVOS BASADOS EN ALGORITMOS GENÉTICOS**
- **OPERABILIDAD EN UN SISTEMA GRID**

David Gascón Cabrejas <483969@unizar.es>

http://www.laotracara.com

Índice de contenido

Prólogo

Al iniciar el estudio sobre las posibilidades del uso de un sistema GRID, multitud de problemas de cuyos costes con elevadísimos nos llaman la atención para ser tratados.

Sin embargo hay un tema que desde hace tiempo llama mi atención: el almacenamiento y operabilidad sobre los grandes bancos de ADN.

Es de sentido común el imaginar que dentro de unos años la genética invadirá por completo nuestra vida cotidiana y por ello cada vez más , los centros de investigación Biomolecular se ven en la necesidad de buscar fuentes de **almacenamiento y operabilidad** eficientes para tratar con este tipo de información tan representativa para cualquier ser vivo.

Pienso que una ciudad como Zaragoza ha de estar preparada de cara al futuro, por lo que la creación de un sistema de trabajo distribuido basado en GRID para el manejo de moléculas de ADN es sin duda un reto que ha de ser afrontado como una de las prioridades de estudio e investigación.

Como sabemos el estudio de ADN se basa mayoritariamente en la búsqueda de SUBCADENAS, por lo que se puede ver como un desarrollo eficiente en este campo podría **extrapolarse**, ya no sólo a problemas genéticos sino a todo un mundo de reconocimiento de lenguajes, gramáticas, e incluso a problemas relacionados con algoritmos de compresión y cifrado.

No es de extrañar por ello que cada día tomen más importancia los **algoritmos genéticos** que estudian las mejoras de los resultados basándose en "mutaciones" controladas de los datos de entrada, por lo que hay que ser conscientes que entramos en un mundo de posibilidades infinitas, donde lo más difícil sea, tal vez aunar todas estas ideas para llegar a un fin común.

1- CREACIÓN DE LA BASE DE DATOS

1-1 Construcción de los árboles de sufijos

Según los últimos estudios en algoritmia, se ha demostrado como una de las claves en la eficiencia de una base de datos de ADN se basa en el **preprocesamiento** que le damos a los datos antes de introducirlos en la misma.

Hay que ser conscientes de que una estructura en principio "tridimensionalmente lineal" la vamos a convertir en un árbol, con unas normas específicas para poder operar sobre él de la manera más eficiente posible.

De las muchas posibilidades hemos elegido la ordenación por **árboles de sufijos**, donde el coste relativo de buscar una patrón P, es el tamaño del mismo y NO el de la información almacenada en la base de datos.

Por lo que podemos exponer que:

Dado un patrón P, de tamaño |P| = m; y una base de datos de cadenas S, de tamaño total |S| = n, el coste en tiempo de buscar el patrón P en S es de orden del tamaño de P, es decir:

O(m)

De todas formas en vez de explicar los algoritmos que se usan para la creación de los árboles de sufijos me gustaría centrarme más en la comunicación e interoperabilidad y paralelismo de los diferentes nodos que componen el GRID.

Como primer reto se nos plantea el **preprocesamiento** de una base de datos de moléculas ADN (cuya estructura es lineal) de forma que al final tengamos un árbol por cada molécula de ADN, o incluso si quisiéramos un árbol global que las incluyese a todas.

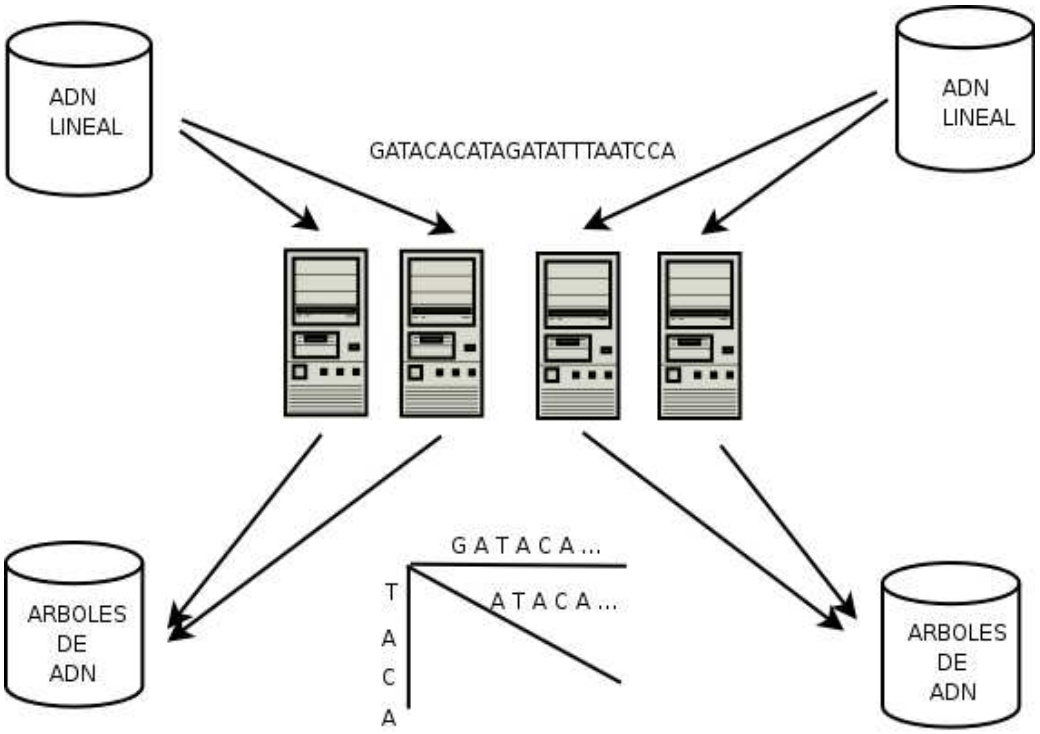
Hay que tener en cuenta de que el coste de la creación de árboles de sufijos es de orden el tamaño lineal de la molécula a procesar.

Por lo que dado una molécula de ADN :M, de longitud |A| = a; podemos decir que el coste en tiempo de crear su árbol de sufijos es relativo al tamaño de esta, es decir:

O(a)

Como es lógico podríamos pensar el una base de datos grande, por ejemplo una que incluyese a todas las personas que habitan Aragón (pensaremos en 1.000.000 de personas); el proceso inicial de crear 1 millón de aboles de sufijos partiendo de la base de datos lineal sería la primera operación que necesitaría ayuda del GRID para poder paralelizar los cálculos.

1-1-1 Implementación Grid de los árboles de sufijos



1-2 Búsqueda de Subcadenas Identificativas

El esquema podría ser más complejo ya que una buena implementación inicial sería el determinar la subcadena de tamaño 'medio' dentro de la molécula de ADN que más se repite.

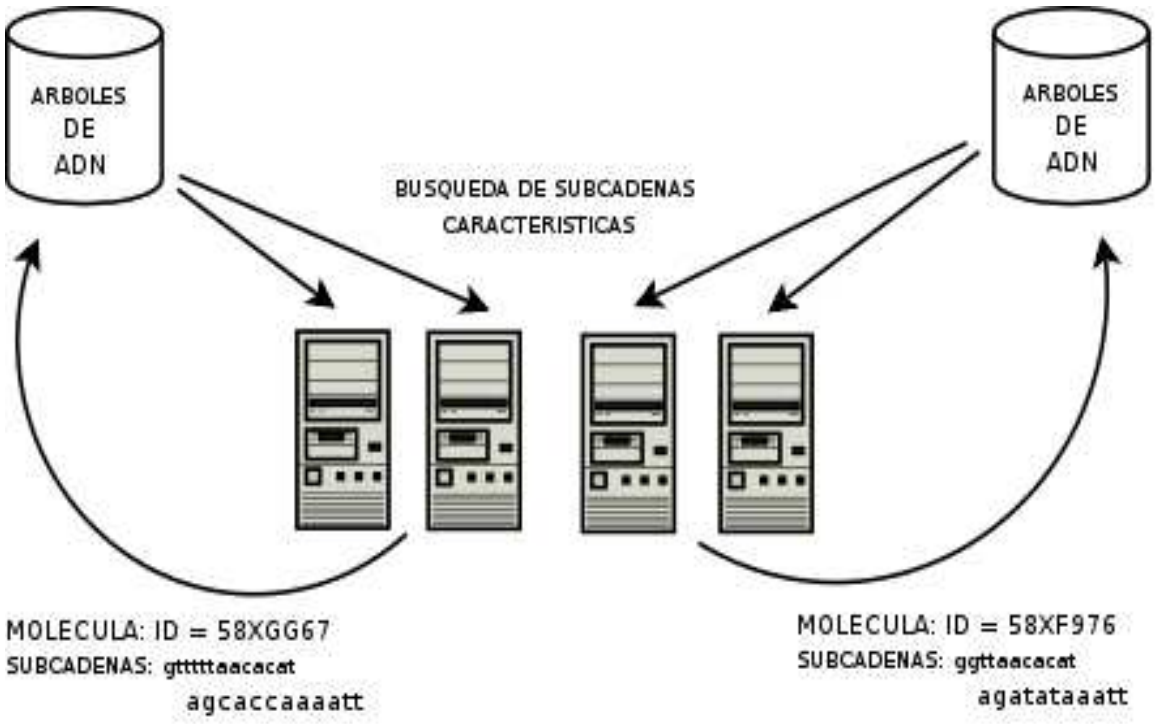
¿por qué buscar esta subcadena?

La verdad es que el tener un dato tan sumamente **identificativo** de una molécula de ADN como la subcadena (o mejor dicho: *subcadenas*) que más se repite en toda su estructura nos ayudará a la hora de realizar búsquedas de patrones externos dentro de la base de datos, pues dependiendo del tipo de búsqueda que impementemos habremos reducido en gran medida las ramas del árbol global a buscar.

Pensemos en que en el momento que adquirimos un patrón podemos saparar estas subcadenas identificativas de forma que en una primera instancia realicemos una búsqueda por los árboles de **patrones característicos** de las diferentes moléculas de ADN de la base de datos.

Supongamos que hemos encontrado 100 moléculas que coinciden en las subcadenas principales del patrón; en este momento podríamos iniciar un estudio completo sobre las relacciones **filogenéticas** del patrón a estudiar con los elementos seleccionados.

1-2-1 Implementación Grid de la búsqueda de subcadenas:



1-3 Comprimiendo moléculas de ADN

Otra posible operación de las moléculas podría ser la compresión de las moléculas de ADN , mediante el método del **"Acierto Aproximado"** (Approximate matching method).

Este método se basa en hacer referencias a cambios en cadenas base almacenadas en diccionarios de datos (similar al método de compresión computacional de Lempel - Ziv).

Veamos un ejemplo:

Subcadena base almacenada en el diccionario: **gatacagataca**

Subcadena encontrada y preparada para comprimir: **gatacagatacc**

Como vemos difieren en la última letra de la cadena, por lo que habría que indicar que se ha producido una variación del último elemento, concretamente que se ha reemplazado la 'a' por la 'c'.

Por lo que si codificamos la secuencia: '**gatacagataca'** mediante el código **'000'** y la operación de cambio: '**C'** mediante el código '**1'** tendríamos que indicar que ha habido un cambio del elemento nº:12 por la letra **'c'** (representada por el código **'10'**) , lo que se indicaría de la siguiente manera:

000 - 1 - 1100 - 10

Como vemos hemos reducido de 12 a 10 el tamaño de codificar esa subcadena; sin embargo la idea principal que queremos exponer sobre el uso de los algoritmos de compresión no es la de ahorrar espacio en el almacenamiento sino el poder ver los cambios que sufren las cadenas y comparar estos cambios entre moléculas de ADN distintas.

Incluso podría tener sentido el aplicar **algoritmos genéticos** de forma que pudiéramos **modelizar** los cambios de las composiciones genéticas de moléculas como el virus del SIDA.

Todo esto supondría un esfuerzo continuado del sistema GRID, ya que habría que gestionar el balanceo de la carga y las prioridades de las diferentes operaciones a ir implementando.

Por supuesto tendríamos la oportunidad de aplicar algoritmos de **gestión y balanceo** de colas.

2- OPERABILIDAD Y PARALELISMO DEL SISTEMA GRID

2-1 Preparando el sistema para la carga que le espera

Como hemos visto hay multitud de operaciones que se pueden desarrollar en paralelo en el GRID, por lo que hay que crear un plan de acción para gestionar de la forma más eficiente posible los ciclos de CPU del sistema.

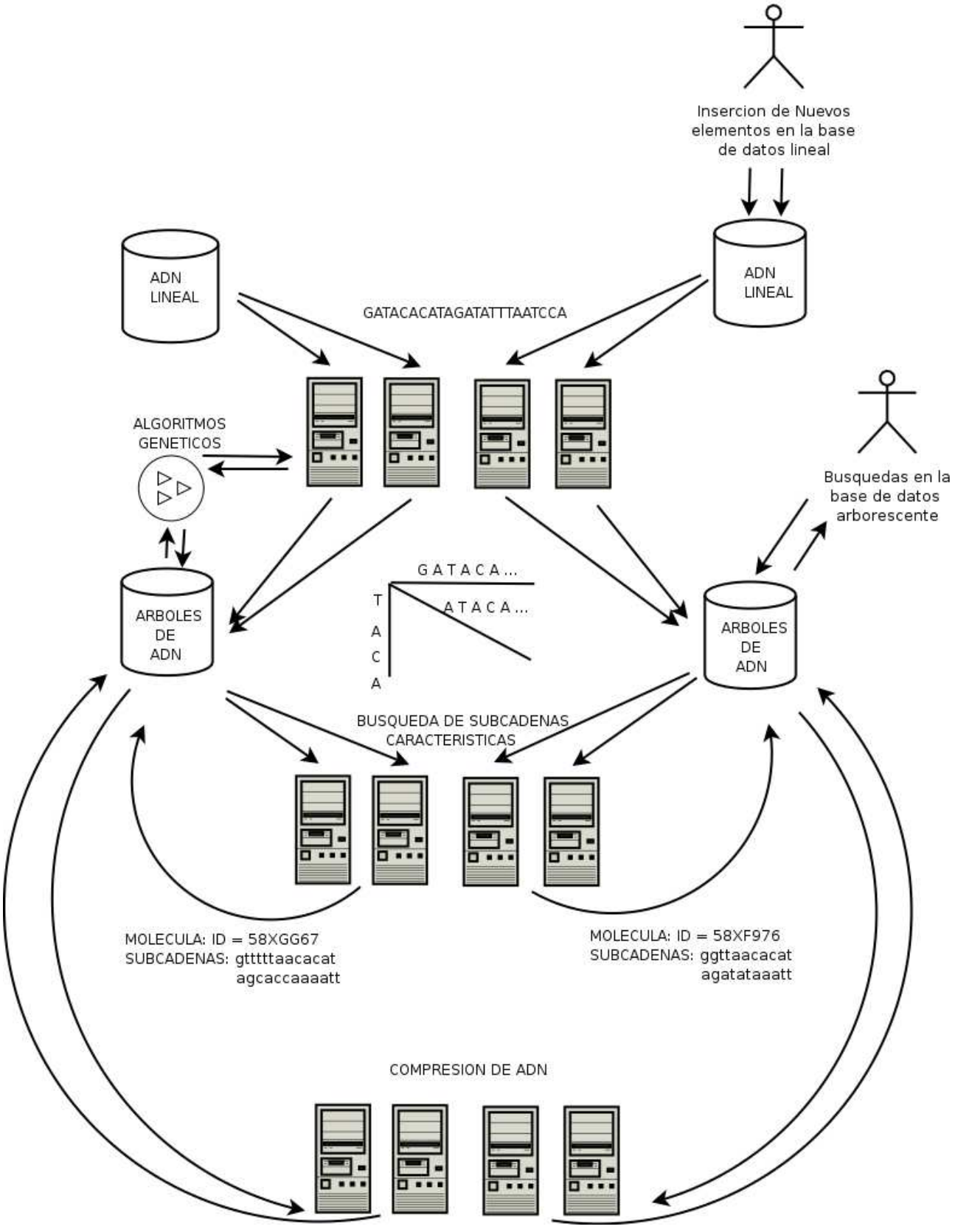
Sin embargo no podemos pensar únicamente en el 'trabajo por lotes' sino que hay que tener en cuenta que en cualquier momento y de forma 'asíncrona' nos pueden pedir que comparemos un patrón con los resultados ya almacenados en la base de datos estructurada.

Con esto lo que quiero decir es que hay que mantener unas colas de trabajo con **prioridades dinámicas** lo que supone todo un reto a la hora de coordinar un sistema donde la **flexibilidad y excalabilidad** no disminuya la **productividad** del mismo.

El objetivo sería conseguir una paralelización máxima en las tareas que se desarrollan:

- Creación de árboles de sufijos
- Búsquedas exhaustivas en la base de datos de árboles ya creados
- Búsqueda de subcadenas principales
- Compresión de las moléculas de ADN
- Desarrollo de Algoritmos Genéticos con el fin de búscar **modelos** representativos de las mutaciones genéticas.

2-2 Posible interacción de paralelismo del GRID



2-3 Conclusión

Todas estas ideas que he ido exponiendo habría que ir refinándolas y ajustando a los modelos reales de disponibilidad de recursos e información.

Simplemente ha pretendido ser un acercamiento a un posible uso de un sistema GRID; con finalidad pragmática y de implantación en un futuro no muy lejano; además de incitar y promover el estudio de algo que está en medio camino entre la biología y la computación : ***la genética***.