Universidad Católica San Pablo

Informe del laboratorio 2

Computer Science — Biología Molecular Computacional

Harold Alejandro Villanueva Borda

---

1. Introducción

En este informe se presenta la implementación del algoritmo de alineamiento MSA Estrella basado en la metodología de Gusfield (1997). El algoritmo está diseñado para realizar alineamientos múltiples de secuencias biológicas de forma eficiente, utilizando paradigmas avanzados para optimizar el rendimiento en C++. Para probar la implementación, se utilizaron cadenas sintéticas vistas en clase y muestras biológicas reales del gen BRCA1, obtenidas mediante la técnica de amplificación LR-PCR.

2. Implementación del algoritmo MSA Estrella

A continuación se explica la lógica del código:

2.1. Alineamiento por pares

El algoritmo comienza calculando alineamientos por pares para todas las combinaciones posibles de secuencias. Este alineamiento por pares se realiza utilizando programación dinámica, que es un enfoque eficiente para resolver problemas de optimización, como el alineamiento de secuencias. Se implementa una matriz de score en la que cada celda contiene el mejor score para alinear subsecuencias hasta ese punto. La matriz se llena usando las siguientes reglas:

- Si los nucleótidos o aminoácidos coinciden, se añade una recompensa positiva (ej. +2).
- Si no coinciden, se penaliza con un valor negativo (ej. -1). Si hay un hueco, también se penaliza con otro valor negativo (ej. -1).

El recorrido de la matriz se hace de forma óptima, usando la regla de recurrencia:

$$S(i,j) = \text{máx} \left(\begin{cases} S(i-1,j-1) + \text{match/mismatch} & \text{(diagonal)} \\ S(i-1,j) + \text{gap penalty} & \text{(arriba)} \\ S(i,j-1) + \text{gap penalty} & \text{(izquierda)} \end{cases} \right)$$

Una vez que la matriz está llena, se hace un traceback desde la esquina inferior derecha hasta la esquina superior izquierda para reconstruir la alineación óptima.

2.2. Selección de la secuencia central (centroide)

Después de calcular los alineamientos por pares, el siguiente paso es seleccionar una secuencia que actúe como çentroide". Para ello, se suma el score de cada secuencia en sus alineamientos con las demás, y se elige la secuencia con el score acumulado más alto. Esta secuencia será la base para el alineamiento múltiple.

2.3. Alineamiento múltiple

A continuación, se alinean todas las secuencias con la secuencia central. El proceso de alineamiento múltiple utiliza la misma lógica de programación dinámica que el alineamiento por pares, pero ahora cada secuencia se ajusta de forma que todas queden alineadas con la secuencia central. Se añaden huecos donde sea necesario para alinear todas las secuencias en la misma longitud.

2.4. Uso de estructuras de datos eficientes

El código utiliza vectores para almacenar las secuencias y las matrices de score, así como la matriz de traceback. Se usan bucles anidados para generar todas las combinaciones de secuencias y realizar los alineamientos por pares, lo cual requiere optimizaciones en la gestión de memoria y cálculos eficientes.

2.5. Optimización y técnicas avanzadas

- Programación dinámica: Se aplica para reducir la complejidad del problema de alineamiento de $\mathcal{O}(2^n)$ a $\mathcal{O}(n^2)$ al usar una matriz de memoización que almacena los resultados intermedios.
- Selección de centroide: Elegir la secuencia más representativa"minimiza el número de operaciones en los pasos posteriores.
- Backtracking: Se utiliza para reconstruir las alineaciones óptimas tras llenar la matriz de score.

3. Paradigma utilizado: Programación dinámica

El paradigma principal que sustenta el código es la programación dinámica, una técnica de optimización que descompone problemas complejos en subproblemas más pequeños, guardando los resultados de estos subproblemas para evitar cálculos repetitivos. En este caso, se aplica para llenar una matriz de score en la que cada celda contiene el mejor alineamiento hasta ese punto, reduciendo la complejidad del problema.

La selección del centroide y la alineación múltiple posterior se basan en un enfoque greedy, en el cual se selecciona la secuencia más representativa para alinear todas las demás, optimizando el proceso de alineación global.

4. Pruebas con secuencias sintéticas

Se realizó una prueba utilizando las secuencias vistas en clase para verificar la correcta implementación del algoritmo. Las secuencias utilizadas fueron:

- S1: A T T G C C A T T
- S2: A T G G C C A T T
- S3: A T C C A A T T T T
- S4: A T C T T C T T
- S5: A C T G A C C

Después de ejecutar el algoritmo, se verificó que se obtuvieran todas las soluciones posibles y se calculó el score final del alineamiento.

4.1. Resultado del alineamiento

```
Alignment of example sequences:
ATTGCCATT---
ATGGCCATT---
A-T-CCAATTTT
ATCT-TC-TT--
ACTGACC-----
Total score: 48
```

Figura 1: Resultados obtenidos con el ejemplo realizado en clases

5. Análisis de las secuencias BRCA1

El archivo BRCA1.txt contiene 6 muestras de secuencias obtenidas por amplificación LR-PCR, que incluyen secuencias forward y reverse. El gen BRCA1 es conocido por su influencia en el desarrollo de cáncer de mama y ovario, por lo que el análisis de sus secuencias es de suma importancia.

Las secuencias forward y reverse fueron procesadas por separado para obtener alineamientos múltiples en cada conjunto.

Alineamiento de las secuencias forward y de las secuencias reverse:

```
Forward sequences alignment of BRCA1:
-TGACGTGTC--TGCTCCACTTCCA----
---TGACGTGTCTGC---TCCA-CTTCCA
-TGACGTG-TCTG--CT--CCACTTCCA-
-TGAC-G-T-GTCTG--CTCCACTTCCA-
-TGA--CGTGTC-TGCTCCA-CTTCCA--
-TGACGTGTCTGCTCCACTTCCA------
Total score of forward sequences: 75

Reverse sequences alignment of BRCA1:
-TGCTTGCAGTTTGCTTTCACTGATGGA----
-T-CAGGTACCCTGACCTTCTCTGA---AC--
--G--TG-GGTTGTAAAG--GTCCCAAATGGT
-TGCCTTG--G---G-TCCCTCTGACTGG---
-GTG-GTGCA--TTG-ATGGAAGGAAGCA---
-AG--TG-AGAGGAGC--TC-C-CAGGGC---
Total score of reverse sequences: -63
```

Figura 2: Resultados obtenidos con el ejemplo realizado en clases

6. Interpretación de los resultados

Se observó que las secuencias forward presentaban un score ligeramente más alto que las reverse, lo que podría indicar una mayor similitud entre las muestras forward. La alineación de secuencias genéticas como las de BRCA1 es crucial para entender las mutaciones o variaciones en estas secuencias, y el uso del algoritmo MSA Estrella permite realizar este análisis de manera eficiente.

7. Conclusiones

La implementación del algoritmo MSA Estrella en C++ demostró ser eficiente para la alineación de secuencias múltiples, tanto en el caso de secuencias sintéticas como en las secuencias forward y reverse del gen BRCA1. Los resultados obtenidos indican que el algoritmo es capaz de proporcionar soluciones precisas y con una buena capacidad de optimización, lo cual es vital en el análisis de secuencias biológicas.

8. Implementación en Github

El código fuente del análisis se encuentra disponible en GitHub: GitHub.