Universitat Oberta de Catalunya

uoc.edu

Evaluación de métodos de ensamblado *de novo* del genoma de *Ulmus minor*.

Jorge Pallarés Zazo
Máster universitario en Bioinformática y Bioestadística
Evolución Molecular

David Macaya Sanz
Dorcas Orengo Ferriz

Nombre Profesor/a responsable de la asignatura
Laura Calvet Liñan

08/06/2021

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Evaluación de métodos de ensamblado de novo del genoma de Ulmus minor.*
Nombre del autor:	*Jorge Pallarés Zazo*
Nombre del consultor/a:	*Dorcas Orengo Ferriz*
Nombre del PRA:	*Laura Calvet Liñan*
Fecha de entrega (mm/aaaa):	06/2021
Titulación:	Máster universitario en Bioinformática y Bioestadística
Área del Trabajo Final:	*Evolución Molecular*
Idioma del trabajo:	Español
Número de créditos:	15
Palabras clave	*Ulmus,* Genoma*,* Ensamblaje *de novo,*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

El olmo común (*Ulmus minor*) es una especie emblemática del territorio español, en situación comprometida desde el siglo XX por la enfermedad de la grafiosis. Para facilitar los esfuerzos de restauración y conservación de esta especie, este trabajo acomete la creación de una versión preliminar del genoma de *U. minor* mediante el ensamblaje *de novo* de las lecturas largas de secuenciación (PacBio). Se utilizó el programa de ensamblaje *Canu* y se obtuvo un ensamblaje con 67.706 *contigs*, un tamaño total de 1,09 Gb y un valor del estadístico N50 de 57 Kb. En paralelo, se procedió a otro ensamblaje mediante el programa *Falcon*, si bien no se pudo obtener un ensamblaje completo, posiblemente porque la cobertura inicial de las lecturas (29x) fue insuficiente. No obstante, esta versión preliminar se considera vital para la comprensión del proceso a escala general, detectar deficiencias y por tanto articular una estrategia sólida de ensamblaje a futuro con el fin de generar un primer genoma de referencia de calidad aceptable para la especie *U. minor*.

Abstract (in English, 250 words or less):

The common elm (*Ulmus minor*) is emblematic species of the Spanish territory, compromised since the 20th century by the Dutch elm disease. In order to facilitate the restoration and conservation efforts on this species, this work aims to obtain a preliminary draft genome of *U. minor* by the assembly of long sequencing reads (PacBio). Assembly software *Canu* was used to generate 67.706 *contigs* with a total size of 1,09 Gpb and an N50 statistic value of 57 Kb. Likewise, software *Falcon* was employed, although a complete assembly could not be obtained, possibly because the initial coverage of the reads (29x) was insufficient. Nevertheless, this preliminary version is considered vital to better understand the process on a general scale, to detect pitfalls, and thus to articulate a robust assembly strategy for the future to generate a genome of standard quality for *U. minor*.

Índice

Lista de figuras

Lista de tablas

1 Resumen

El olmo común (*Ulmus minor*) es una especie emblemática del territorio español, en situación comprometida desde el siglo XX por la enfermedad de la grafiosis. Para facilitar los esfuerzos de restauración y conservación de esta especie, este trabajo acomete la creación de una versión preliminar del genoma de *U. minor* mediante el ensamblaje *de novo* de las lecturas largas de secuenciación (PacBio). Se utilizó el programa de ensamblaje *Canu* y se obtuvo un ensamblaje con 67.706 *contigs*, un tamaño total de 1,09 Gp y un valor del estadístico N50 de 57 Kb. En paralelo, se procedió a otro ensamblaje mediante el programa *Falcon*, si bien no se pudo obtener un ensamblaje completo, posiblemente porque la cobertura inicial de las lecturas (29x) fue insuficiente. No obstante, esta versión preliminar se considera vital para la comprensión del proceso a escala general, detectar deficiencias y por tanto articular una estrategia sólida de ensamblaje a futuro con el fin de generar un primer genoma de referencia de calidad aceptable para la especie *U. minor*.

2 Introducción

El género *Ulmus* incluye unas 40 especies[1] con una amplia distribución circumboreal, desde las zonas más elevadas de los trópicos hasta las regiones templadas del hemisferio norte, principalmente en el norte de Asia, América del Norte y Europa.

En su día, los olmos fueron especies protagonistas de las masas forestales mixtas de frondosas, dominando principalmente sobre los ecosistemas riparios y en las llanuras aluviales[2]. Se les considera especies clave de los hábitats donde se establecen ya que conforman multitud de asociaciones con un gran número de organismos. Estudios como los de Hans M. Heybroek[3], informan de la existencia de 79 especies de insectos especializados en los olmos en una sola región de Europa.

Desde la antigüedad, los olmos han prestado importantes servicios al ser humano, entre otros la madera de buena calidad. Asimismo, se utilizaba en la ganadería, ya que sus hojas se utilizaban como suplemento alimenticio durante las estaciones más secas; y en la agricultura, por su morfología óptima para el soporte de la vid[4]. Además, son comúnmente utilizados en la restauración de suelos, por sus características óptimas para la bioacumulación de elementos potencialmente tóxicos[5].

El uso constante de los olmos y la frugalidad en cuanto a sus requerimientos edáficos permitió que su propagación en el medio natural se hiciese de forma masiva llegando incluso al ámbito urbano-rural. En beneficio de su porte, el olmo se utilizaba como elemento central para proporcionar sombra en los lugares de reunión pública. Se puede comprobar la importancia cultural de los olmos en el pasado viendo el número de plazas y demás topónimos donde aparece su nombre[2].

En la actualidad, las poblaciones de olmos están muy degradadas debido al impacto negativo de los cambios inducidos por el hombre en los ecosistemas ribereños, y a las pandemias sucesivas provocadas por dos patógenos del género *Ophiostoma.* Ambos se consideran los causantes directos de la enfermedad de la grafiosis: el hongo *Ophiostoma-ulmi*, provocó la primera pandemia; y seguidamente *Ophiostoma novo-ulmi*, más virulento que el anterior, fue el agente causal de la segunda, y actual pandemia. Las redes de comercio propiciaron la rápida expansión de *O. novo-ulmi* a escala mundial, devastando principalmente las poblaciones europeas de olmo[6], ya que los olmos americanos fueron diezmados por *O. ulmi*. La enfermedad de la grafiosis ha provocado uno de los mayores impactos conocidos en el mundo vegetal, y se estima que en la actualidad solo se puede observar el 1% de los ecosistemas dominados por los olmos en el pasado[7].

En Europa se pueden encontrar tres especies de olmo: *Ulmus glabra* Huds., *Ulmus laevis* Pall. y *Ulmus minor* Mill*.* Las tres especies son muy susceptibles a la enfermedad, aunque su afectación en campo oscila en gran medida en función de factores geográficos y climáticos. El vector de la enfermedad de la grafiosis es un insecto perforador de la madera de la familia Curculionidae (subfamilia Scolytinae), que encuentra sus condiciones óptimas para el

desarrollo en latitudes medias. Parece concordar la baja afección encontrada en la especie *U. glabra* con la escasez de insectos vectores en las zonas septentrionales donde se establece. Las poblaciones de *U. laevis* han podido afrontar la enfermedad gracias a sus características anatómicas de la madera, pues resulta poco atractiva para el insecto vector[2]. Sin embargo, el tercer olmo autóctono, *U. minor* (olmo común), es el más abundante en las zonas de latitudes meridionales donde los patógenos se encuentran en condiciones óptimas para su desarrollo y expansión. La Península Ibérica, reúne las condiciones climáticas para el desarrollo tanto del hongo como del insecto vector, y es uno de los territorios donde el *U. minor* se encontraba más ampliamente distribuido. Dadas estas premisas, se puede llegar a la conclusión de que en la Península Ibérica la especie *U. minor* se encontraba en una situación de máxima preocupación.

Ante la situación crítica de la especie, en 1986 nace el "Programa español para la evaluación y conservación de los recursos genéticos de los olmos y la obtención de individuos resistentes a la grafiosis", focalizado en dos objetivos claramente definidos:

- Conservar los recursos genéticos: abarca el proceso desde la recogida de muestras de los ejemplares en campo para su caracterización genética y posterior propagación, hasta la plantación de estos ejemplares una vez propagados en parcelas de conservación y bancos clonales.

- Obtener individuos resistentes: se inicia con la inoculación del hongo *Ophiostoma novo-ulmi* en los ejemplares conservados de cuatro años de edad y una altura superior a los 2 metros para testar su tolerancia a la enfermedad. Además, se realizan cruces controlados entre distintos genotipos para estudiar la heredabilidad en la descendencia de la resistencia a la enfermedad.

El programa ofrece un modelo de actuación biológico completo de un patosistema forestal para las poblaciones de especies comprometidas. Hoy en día la supervivencia del olmo común está garantizada por la presencia de siete genotipos catalogados como resistentes. Aun así, la especie está condicionada por la falta de diversidad, y necesariamente se debe incluir un número mayor de genotipos resistentes (así como cruces de gentotipos resistentes con susceptibles que alcancen el umbral mínimo para su catalogación como "resistente a la grafiosis") para asegurar la supervivencia natural de la especie.

Con el objetivo de continuar la línea de investigación sobre la especie y su resistencia a la enfermedad, se ha propuesto un nuevo enfoque fundamentado en la secuenciación y ensamblaje *de novo* del primer genoma del olmo, con potencial para convertirse en el genoma de referencia del género. Para ello se ha elegido a uno de los genotipos catalogados como resistente (en concreto el genotipo V-AD2 procedente de la región de Ademuz, Valencia). Hasta hoy, se han realizado varios estudios genéticos en relación a las especies de olmos, sin embargo hay muy poca información disponible en comparación con otras especies caducifolias de la misma distribución[6], ya que la mayoría de los estudios se han limitado a la clasificación de las especies de olmo en sus

respectivos subgéneros (subgénero. *Ulmus* y subgénero *Oreoptelea*). Las clasificaciones se basan en técnicas de citometría de flujo que estiman el tamaño medio del genoma, pues la diferencia del tamaño entre ambos subgéneros es significativa. En 2007 Loureiro y colaboradores[8], concluyeron que la especie *U. minor*, del subgénero Ulmus, tiene un tamaño aproximado del genoma de unos 4.25 pg/2C[9] (2C hace referencia a especies diploides), siendo 1 pg DNA equivalente a 978 millones de pares de bases de nucleótidos (Mb).

La complejidad de secuenciar un genoma, en cuanto a tiempo y recursos económicos, ha disminuido gracias al acelerado avance tecnológico que habilita las técnicas de secuenciación de lectura larga. Aun así, sigue siendo un proceso minucioso que engloba varios procedimientos de laboratorio pues supone la secuenciación del ADN de todos los cromosomas de un organismo, así como el contenido en él de mitocondrias y, en el caso de las especies vegetales, plástidos. Esta valiosa información, demanda amplios recursos computacionales y económicos, restringiendo así la escala de trabajo a potentes grupos de investigación. Entre el INIA (Instituto Nacional de Tecnología Agraria y Alimentaria) y el grupo de investigación “Genética, Fisiología e Historia Forestal” (UPM) se reúnen los recursos necesarios para abordar el estudio.

2.1 Contexto y justificación del Trabajo

2.1.1 Descripción general

Este trabajo incluye desde el proceso de filtrado y pulido de las secuencias, hasta el ensamblaje final de éstas para conformar el primer genoma de la especie *Ulmus minor.*

El estudio se inicia con el filtrado de las secuencias y su posterior ensamblaje mediante la tecnología Pacific Biosciences (PacBio); y posteriormente conseguir el pulido final a raíz de las secuencias en formato Ilumina. La tecnología PacBio proporciona una longitud de lectura más larga que las lecturas de Illumina, por lo que se ofrece una mejor oportunidad para el ensamblaje del genoma *de novo*. La desventaja se revela en las altas tasas de error y las bibliotecas de alto coste y bajo rendimiento. Es por ello que se propone una segunda fase para respaldar y corregir el ensamblaje de las secuencias en crudo mediante la tecnología Illumina. La secuenciación en las plataformas de Illumina es mucho menos costosa, tiene menos exigencias en cuanto a la calidad del ADN y ofrece menor tasa de error. Sin embargo, la longitud de las lecturas es reducida (generalmente menor de 200 pares de bases).

Respecto al software utilizado para el ensamblaje, se decidirá entre la utilización exclusiva de *Canu*[10]; o la utilización conjunta de *Canu* y *Falcon*[11] para su posterior comparación.

2.1.2 Justificación del TFG:

El olmo común (*Ulmus minor*) es un árbol emblemático ampliamente repartido en la totalidad del territorio español. Sus poblaciones están actualmente

limitadas por la susceptibilidad que presenta a las enfermedades, especialmente a la enfermedad del olmo holandés *Ophiostoma novo-ulmi*[12].

La enfermedad de la grafiosis es un buen ejemplo de un sistema patológico forestal, compuesto por tres agentes: el olmo (hospedante), el hongo patógeno y el escarabajo (del género *Scolytus*) que actúa como vector de propagación de la enfermedad. Son muchos los estudios enfocados a la resolución del problema desde la perspectiva de la genética. La información genética disponible hoy en día es bastante completa en el caso del hongo, aunque escasa en el caso del olmo y del escarabajo.

Es por ello que el objetivo del presente estudio amplía en gran escala la información genética del olmo, pues se obtendrá el ensamblado *de novo* del genoma del olmo común.

2.2 Objetivos del Trabajo

El objetivo principal de este trabajo es generar una primera versión del genoma de la especie *Ulmus minor,* y posteriormente evaluar los distintos softwares de ensamblaje.

1. Generar una primera versión del genoma de *Ulmus minor* incluye varios objetivos específicos:

    1.1. Realizar un filtrado de las secuencias y posteriormente ensamblar las secuencias mediante la tecnología PacBio.

    1.2. Mejorar el ensamblaje de las secuencias mediante el formato Illumina.

    1.3. Analizar la calidad del genoma obtenido mediante estadísticos tipo N50.

    1.4. Proponer la inclusión de nuevos datos que potencialmente mejorarían el ensamblaje del olmo (mayor cobertura de secuencias, o mapas genéticos)

2. Evaluar los distintos softwares de ensamblaje: en principio se intentará utilizar tanto *Falcon* como *Canu*.

2.3 Enfoque y método seguido

El ADN nuclear del olmo común se encuentra secuenciado y por tanto disponible para el proceso de ensamblaje.

El método a seguir comienza con una revisión bibliográfica de otros estudios relacionados con el ensamblaje *de novo* de genomas en especies forestales (véase por ejemplo el trabajo de Liu y colaboradores[13]). Además, esta lectura bibliográfica sirve para familiarizarse con las plataformas de secuenciación y extensión de las secuencias, permitiendo así planificar y decidir el orden de uso de los formatos disponibles proporcionados por dichas plataformas. Seguidamente el trabajo profundiza en los resultados de secuenciación de la plataforma PacBio, resultados como el tamaño de las lecturas generadas y su posterior comparación con los tamaños estimados del genoma por citometría

de flujo proporcionados por estudios anteriores (como el estudio realizado por Loureiro y colaboradores[8], o el de Whittemore y Xia[14]). Se valorará la eficacia de la plataforma PacBio y la necesidad de complementar el primer proceso de filtrado mediante la plataforma Illumina.

De la mano de la revisión bibliográfica, se accederá a la *work-station* recientemente instalada y preparada para su uso en la Escuela Técnica Superior de Ingeniería de Montes, Forestal y del Medio Natural (UPM). Es aquí donde empieza la fase de trabajo técnico, es decir, el filtrado y pulido de las secuencias para conseguir el objetivo final del trabajo: el ensamblado de las secuencias mediante los softwares de ensamblaje *Canu* y *Falcon*.

2.4 Planificación del Trabajo

Para llevar a cabo el estudio, se accederá a la *work-station* del grupo de investigación "Sistemas Naturales e Historia Forestal" (UPM) por ser colaborador del mismo. La *work-station* tiene dos procesadores de 16 núcleos cada uno y 256 Gb de RAM.

El software que se va a usar es *freeware*, es decir, con licencia, pero gratuito para uso científico.

En cuanto a la bibliografía, desde el INIA y la UPM se cuenta con el acceso a plataformas como *Web of Science* y a muchas revistas especializadas e indexadas.

2.4.1 Tareas

1. Generación de una primera versión del genoma de la especie *Ulmus minor.*
    - Comprobación del funcionamiento de la *work-station*.
    - Instalación del software necesario.
    - Filtración de las secuencias crudas.
    - Evaluación de la calidad de las secuencias crudas
    - Manejo de los softwares de ensamblaje *Falcon* y *Canu*.
    - Ensamblaje de las secuencias largas (obtenidas mediante la tecnología PacBio) usando el software *Falcon* y *Canu*.
    - Pulido del ensamblaje usando las secuencias cortas (obtenidas mediante la tecnología de Illumina).
    - Estimación de calidad de los ensamblajes obtenidos mediante estadístico tipo N50.
    - Comparación de la calidad general de los ensamblajes obtenidos con ensamblajes publicados de organismos similares. Valoración de la necesidad de incluir nuevos datos para mejorar el ensamblaje del genoma del olmo.

2. Comparación de los distintos ensamblajes obtenidos con los softwares empleados.

2.4.2 Calendario

Nombre	Fecha de inicio	Fecha de fin
Comprobación del funcionamiento correcto de la work station Familiarizarse con el uso y manejo de la work-station	17/3/21	21/3/21
Instalación del software necesario	18/3/21	21/3/21
Filtración de las secuencias crudas Código informático para automatización	22/3/21	9/4/21
Evaluación de la calidad de las secuencias crudas	10/4/21	16/4/21
Manejo de los softwares Falcon y Canu	17/4/21	13/5/21
Ensamblaje de las secuencias largas (obtenidas mediante la tecnología PacBio) usando el software Falcon y Canu.	17/4/21	3/5/21
Pulido del ensamblaje usando las secuencias cortas (obtenidas mediante la tecnología de Illumina).	4/5/21	13/5/21
Estimación de calidad de los ensamblajes obtenidos mediante estadístico tipo N50.	14/5/21	21/5/21
Comparación de la calidad general de los ensamblajes obtenidos con ensamblajes publicados de organismos similares. Valoración de la necesidad de incluir nuevos datos para mejorar el ensamblaje del genoma del olmo.	4/5/21	21/5/21
Comparación de los distintos ensamblajes obtenidos con los softwares empleados.	4/5/21	21/5/21
Redacción: Introducción y empezar Metodología	19/3/21	9/4/21
Redacción: Metodología	10/4/21	3/5/21
Redacción: Resultados y Discusión	4/5/21	28/5/21
Redacción: Conclusión y Abstract	29/5/21	8/6/21
Elaboración de la presentación	9/6/21	12/6/21
Defensa	16/6/21	23/6/21

Figura 1: Lista de tareas y duración de las mismas para alcanzar los objetivos del trabajo. Fuente: elaboración propia mediante el software GanttProject.

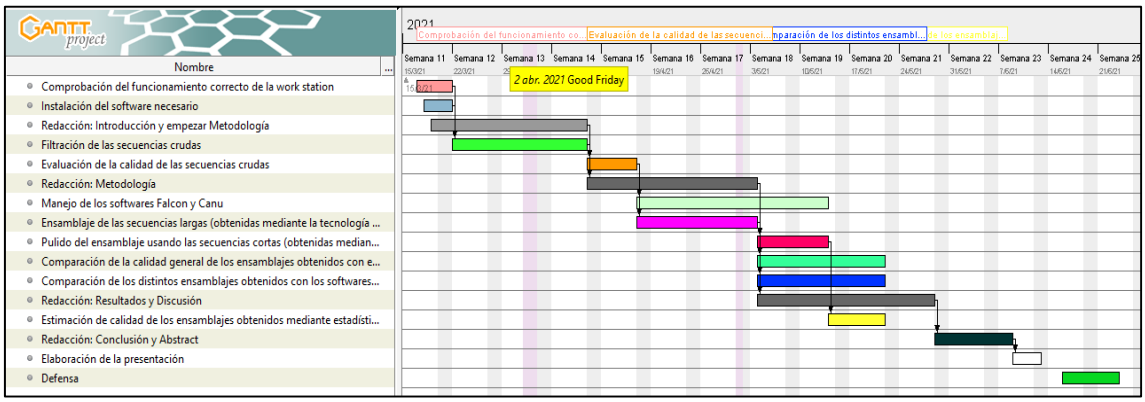

Figura 2: Cronograma de las tareas y duración de las mismas para alcanzar los objetivos del trabajo. Fuente: elaboración propia mediante el software GanttProject.

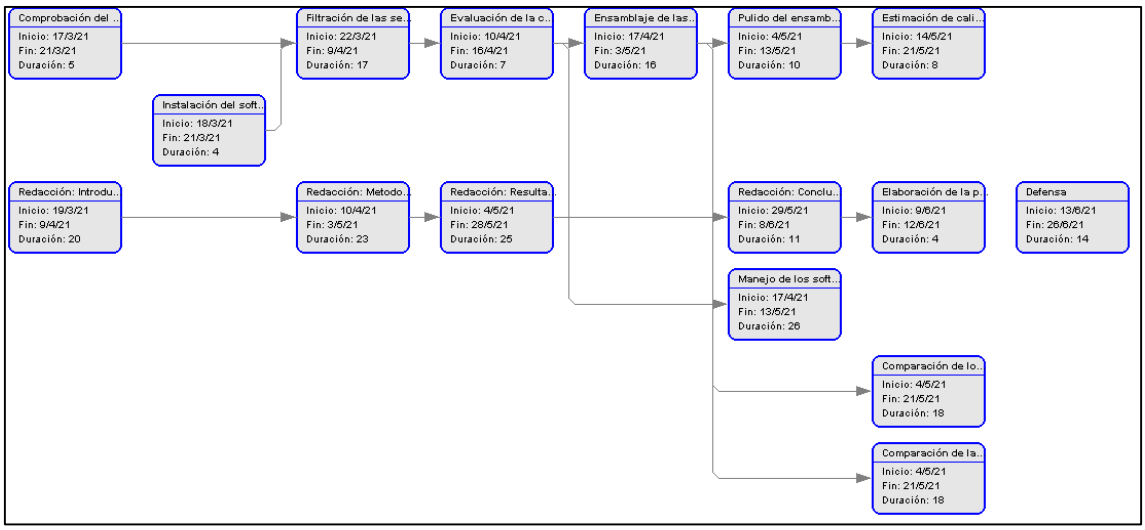

Figura 3: Gráfico PERT de las tareas necesarias para alcanzar los objetivos del trabajo. Fuente: elaboración propia mediante el software GanttProject.

2.4.3 Hitos

PEC2 (19/04): en la parte técnica se debe haber finalizado y evaluado el filtrado de las secuencias. Se debe haber redactado tanto la Introducción, como gran parte de la Metodología.

PEC3 (17/05): en la parte técnica se debe haber ensamblado el genoma mediante ambas plataformas, así como realizado con éxito las carreras con ambos softwares. Del mismo modo, y dado que es parte del proceso, se inicia la evaluación de las plataformas y la necesidad de utilizar ambas. La redacción incluye los apartados de Metodología y los Resultados completos.

PEC4 (18/05): en la parte técnica se debe haber pulido el ensamblado. La redacción se completa con los apartados de Discusión y Conclusión, y se redacta el *Abstract*.

PEC5a (13/06): se elabora la presentación del trabajo en el formato correspondiente.

PEC5b (23/06): se realiza la defensa del trabajo mediante un vídeo explicativo.

2.4.4 Análisis de riesgos

Como en todo proyecto se debe tener en cuenta tanto el factor tiempo, como la escala del mismo. Para ello, se han definido objetivos alcanzables. Además, para cada uno de los objetivos se enumeran las tareas necesarias para completarlos con éxito.

No hay riesgo en cuanto al funcionamiento de la secuenciación, dado que las secuencias ya existen. Aun así, no se ha evaluado la calidad de las secuencias, por lo que hay un pequeño riesgo de que sea baja.

En relación a la complejidad técnica del estudio, se debe tener en cuenta tanto los posibles fallos en el funcionamiento de la *work-station,* como el tiempo necesario para aprender a manejarla eficientemente. En el caso de que algo fallase, o que la capacidad de la *work-station* sea insuficiente, se puede acceder a CESGA, un supercomputador con acceso directo desde el INIA. Además, es posible que los programas de filtrado y ensamblaje de las secuencias se muestren más complejos de lo esperado.

2.5 Breve sumario de contribuciones y productos obtenidos

2.5.1 Plan de trabajo

El plan de trabajo se establece siguiendo las pautas definidas en el apartado de la Planificación del trabajo.

2.5.2 Memoria

La memoria del trabajo se irá redactando a ritmo acorde con los avances en el trabajo técnico. Se plantea el avance lo antes posible en los apartados de Introducción y parte de la Metodología.

2.5.3 Producto

- Generación de un primer ensamblaje del genoma de *Ulmus minor* de carácter preliminar al nivel de *contigs*.
- Informe sobre alternativas de futuro para la mejora de las siguientes versiones en función de los resultados obtenidos.
- Estudio comparativo de distintos softwares para determinar cuál la efectividad y eficacia en este contexto
- Información de diversos parámetros a comparar con el género *Ulmus* y clasificación filogenética.

2.5.4 Presentación virtual

La presentación virtual se realizará mediante un video que muestre, a modo de síntesis, el contenido completo del presente estudio.

2.5.5 Autoevaluación del proyecto

Se definirá una autoevaluación crítica del proyecto en relación al seguimiento y cumplimientos de los objetivos marcados, así como los cambios realizados (previstos o no en el cronograma), hasta la finalización del estudio.

2.6 Breve descripción de los otros capítulos de la memoria

2.6.1 Estado del arte

Se hace hincapié en la complejidad de ensamblar el genoma de un organismo superior mediante ejemplos de los principales genomas disponibles de especies forestales. Por último, se describe brevemente la tecnología de las plataformas de secuenciación.

2.6.2 Metodología

Se inicia con un subapartado dedicado a la tecnología de secuenciación (genotipo utilizado, proceso de extracción del ADN, dónde se ha realizado, con qué medios, etc.). Seguidamente se describe de forma completa el proceso de filtrado así como la tecnología utilizada. Continúa con la explicación técnica de los softwares de ensamblaje, las parametrizaciones que se van a seguir y los estadísticos de calidad que se van a usar.

2.6.3 Resultados

Se adjuntan los resultados obtenidos durante las fases técnicas junto con los estadísticos de calidad. Se respalda los párrafos explicativos de cada uno de los resultados con gráficos específicos ilustrativos.

Para una mejor comprensión, se sigue el orden establecido en el apartado de Metodología. Así, se iniciaría con los resultados del proceso de filtrado de las secuencias, para posteriormente añadir los resultados del ensamblado y pulido de las secuencias.

2.6.4 Discusión

Se determina el alcance de las tecnologías de secuenciación, resaltando la necesidad de utilizar ambas (pros-contras). Igualmente se valoran ambos softwares de ensamblaje en relación al resultado obtenido. Se detallan los problemas encontrados en cada una de las fases y su resolución. Es aquí donde se profundiza sobre la necesidad de re-ensamblar el genoma con otro tipo de datos (ampliar la cobertura de secuencias cortas o largas, incorporar nuevas tecnologías de secuenciación como Oxford Nanopore).

2.6.5 Conclusiones

Se enumeran y describen los hitos más importantes del proyecto. Además se entra en detalle de las posibilidades a futuro que ofrece el estudio.

3 Estado del arte

La secuenciación de genomas de organismos superiores suele presentar complejos retos. Por un lado, los genomas son grandes y no homogéneos. Hay regiones difícilmente accesibles debido a su contenido nucleico. Otras regiones tienen alto grado de repetición, que reducen la eficacia de los algoritmos de ensamblaje. También las duplicaciones de partes del genoma o del genoma completo suelen dificultar el ensamblado, especialmente si son recientes ya que aumentan la probabilidad de eliminación de parte de las secuencias, y dificultan la correcta posición de la secuencia en el genoma. La calidad del ensamblado se establece en función de datos estadísticos como el N50 (longitud del *contig* que junto con todos los mayores que él, abarcan el 50% de todo el genoma), o la longitud máxima.

Los ensambladores de genomas se basan en algoritmos matemáticos complejos para generar los *contigs* mediante la unión de varios fragmentos de secuenciación. Para afrontar secuencias repetidas a lo largo del genoma durante el ensamblaje, se utilizan elementos matemáticos con una estructura capaz de superponer dichas secuencias. Estos recursos matemáticos reciben el nombre de grafos, y en la actualidad se usan dos tipos: Grafos de Bruijin (utilizados por la tecnología Illumina y PacBio) y Grafos en Cadena (utilizados por la tecnología Roche 454). Los genomas de los géneros *Populus, Quercus* y *Fraxinus* son algunos ejemplos de los genomas de especies forestales disponibles que utilizan tecnología de ensamblaje basada en grafos matemáticos. Además, estos genomas sirven como guía para abordar el proceso de ensamblaje actual del *U. minor* ya que todas estas especies son angiospermas y por tanto comparten características específicas, entre otras las duplicaciones completas del genoma debido al proceso de poliploidización[15].

El primer genoma disponible de una especie forestal fue el genoma de *Populus trichocarpa* (familia *Salicaceae*), publicado en 2006[16]. El genoma de esta especie tiene un tamaño aproximado de 0,48Gb/1C. Esta especie se utilizó como modelo en los procesos de secuenciación y ensamblaje de las especies forestales por su pequeño tamaño del genoma, el manejo rápido y sencillo del cultivo, y la amplia disponibilidad de recursos genómicos específicos de la especie. Para llevar a cabo el ensamblaje de *P. trichocarpa*, se integró el ensamblaje de las secuencias *shotgun* con el mapeo genético y se reconstruyó la primera versión del genoma a escala cromosómica. Al ensamblar las secuencias, se conformaron un total de 2.447 *scaffolds* (unión de dos o más *contigs* de forma ordenada y orientada según la estructura del genoma original) que contenían 0,41 Gb de ADN genómco. Los restantes 0,075 Gb resultaron ser secuencias de regiones de ADN heterocromático y por tanto no fue posible incorporarlas en el ensamblaje. El cromosoma más pequeño se constituía de 2 *scaffolds* que abarcaban 0,012 Gb; mientras que el cromosoma más grande se conformaba de 21 *scaffolds* y abarcaba 0,035 Gb. Mediante el mapeo genético se pudo estimar que los géneros *Arabidopsis* y *Populus* se separaron hace unos 100-120 millones de años (Ma); y mediante el estudio de los genes ortólogos entre los géneros *Salix* y *Populus* se pudo apreciar duplicaciones

más recientes (~13 Ma) de afección casi total en los genomas de cada una de las especies.

El genoma de la especie *Quercus robur* (familia *Fabaceae*), se publicó en 2015. El genoma de esta especie tiene un tamaño de 1,5 Gb/2C. Por su vínculo filogenético, se inspira en el modelo genético de la especie *Prunus pérsica* (melocotonero*;* familia *Rosaceae*), y al finalizar el ensamblaje se concluyó que comparten un 84,6% de los genes entre ambas especies[17]. El proceso de secuenciación y ensamblaje del genoma de *Q. robur* fue complejo a razón de su amplio tamaño, y elevada heterozigosidad cromosómica. Se utilizaron varias tecnologías de secuenciación para posteriormente combinar el ensamblaje de lecturas de mayor tamaño y precisión, pero poco rendimiento (Roche 454 GS-FLX) con otra plataforma de gran rendimiento, pero poca longitud (Illumina *paired-end* o lecturas bidireccionales) y, finalmente se aplicó Sanger, técnica no masiva, de bajísimo rendimiento, pero gran precisión y gran control sobre la molécula diana. Así, una vez ensambladas las lecturas largas, se conformaron los *contigs* (alineamiento de múltiples lecturas); y con ayuda de las lecturas cortas se alcanzó un total de 17.910 scaffolds. Se obtuvo un estadístico N50 de 260 kb correspondiente a 1.468 *scaffolds.* Al finalizar el ensamblado, se encontró con un gran número de repeticiones de tamaño variable (desde TEs o transposones, hasta microsatélites y duplicaciones del genoma) que pudieron ser resueltas gracias a una secuenciación de amplia cobertura usando la tecnología Roche 454 GS-FLX.

El genoma de la especie *Fraxinus excelsior* (familia *Oleaceae*) se publicó en 2017. El genoma de la especie tiene un tamaño de 0,64Gb/1C. Para el proceso de secuenciación se seleccionó un individuo con bajo porcentaje de heterozigosidad[18], y durante el proceso de anotación de los genes codificadores se descubrió que, al comparar con otras diez especies vegetales (entre ellos *Populus*), el 25% del total eran específicos de los fresnos. Se utilizaron tecnologías de secuenciación complementarias siguiendo el protocolo habitual para la secuenciación de genomas en organismos superiores. La tecnología HiSeq 2000 (Illumina) proporcionó lecturas cortas de 100 pb y se respaldó con lecturas más largas de tecnología MiSeq (Illumina *paired-end*) de 300pb. Además, se utilizó la tecnología 454 FLX+ (Roche) para generar secuencias con una media de 642 pb utilizadas como marco de secuenciación por su longitud más amplia. Para refinar en el proceso de ensamblaje, se constituyeron 5 moldes o versiones, de las cuales la versión definitiva publicada, se conformaba de 235.463 *contigs* que tras el proceso de filtración se organizaron en un total de 89.487 *scaffolds*. Se obtuvo un estadístico N50 de 104 kb. Una vez ensamblado el genoma del fresno, mediante el software *RepeatMasker* se estimó que más de un tercio del genoma se identificaba como elementos repetidos, aunque “en comparación con otros genomas de eudicotas de tamaño similar este contenido de repeticiones es bajo”[18]. Además, el estudio de los genes parálogos (genes duplicados respecto del último ancestro común), sugieren una duplicación completa del genoma compartida con el olivo (*Olea europea,* familia *Oleaceae*).

El genoma de la especie *Populus alba* (familia *Salicaceae*) se publicó en 2019. El genoma de la especie tiene un tamaño de 0,48 Gb/2C. Para llevar a cabo el proceso de secuenciación y ensamblaje del genoma de *P. alba*, se tomó de

referencia el transcriptoma completo de la misma especie, junto con el genoma de referencia del mismo género. Posteriormente se comprobó la hipótesis de correspondencia ya que sobre el genoma nuclear se pudo mapear el 99,9% de las isoformas presentes en el transcriptoma[13]. La secuenciación se realizó mediante la tecnología SMRT (*single-molecule real-time*; PacBio) de amplia cobertura, en combinación con la tecnología Illumina Hi-Seq2500 para una mayor precisión. Las secuencias se agruparon en 1285 *contigs* los cuales representaban el 83,86% del genoma total de *P. alba* y un estadístico N50 de 1,18 Mb. Al finalizar el ensamblado, se profundizó en la búsqueda de transposones mediante los softwares *TandemRepeatsFinder*, *RepeatMasker*, Open-4.0, e incluso se conformó una librería específica para *P.alba* que, integrando los programas *RepeatModeler*, *LTR_finder* y *PILER*, mostrase la información principal de las secuencias repetidas; concluyendo que un 45,16% del genoma total correspondía a elementos repetidos[13]. Mediante el estudio filogenético del genoma, se pudo comprobar que la separación entre las especies *P. alba* y *P. trichocarpa* se produjo hace 5 Ma.

Por otro lado, también se han secuenciado genomas de gimnospermas (en concreto de coníferas), siendo el primero el de *Pinus taeda* (2010; familia *Pinace*). Los genomas de las coníferas presentan aún más retos de secuenciación que las frondosas por sus genomas extremadamente largos (20-40 Gb); su diversidad en cuanto a las familias de repetición altamente divergentes entre sí; las diferencias en los patrones comunes específicos de elementos promotores necesarios para la trascripción (cajas TATA; CAAT). Aun así, se consiguió secuenciar un 7,5% del genoma total mediante la tecnología de secuenciación Sanger, junto con la tecnología WGS (*whole genome shotgun)* de Genome Analizer II. Actualmente se encuentran disponibles 328.628 secuencias de expresión etiquetadas (ETS) de *P. taeda* en las bases de datos del NCBI. Estas secuencias de expresión, son el resultado de multitud de proyectos de secuenciación de diferentes tejidos bajo condiciones experimentales distintas; y de forma paulatina, se han ido agrupando en 18.921 *unigenes* hasta conformar un mapa genético ideal para la exploración genómica entre las especies de gimnospermas[19].

El software *Canu*, es una de las herramientas de ensamblaje que se ha utilizado para obtener el ensamblaje de las secuencias de *U. minor*. *Canu* es una bifurcación del *Celera Assembler* diseñado de forma específica para el montaje de secuencias PacBio o Oxford Nanopore. La cobertura mínima recomendada para abordar genomas de organismos eucariotas mediante *Canu*, es de 30-60 veces el genoma; y las secuencias de entrada deben estar en formato *fasta* o *fastq*. *Canu* funciona en tres fases diferenciadas. La primera fase se corresponde con el proceso de corrección en cuanto a la mejora en la precisión de las bases nucleótidas en las lecturas. La segunda fase se corresponde con el recorte de las lecturas hasta obtener los fragmentos de secuencia de alta calidad. La tercera y última fase se corresponde con el ordenamiento de las lecturas en *contigs*, generación de secuencias de consenso y gráficos de rutas alternativas.

El software *Falcon*, es la herramienta de ensamblaje propuesta como alternativa a *Canu. Falcon* está diseñado para el ensamblaje de genomas grandes con coincidencia diploide secuenciados con lecturas largas de PacBio.

Funciona de forma jerárquica (*HGAP*, *hierarchical genome assembly process*) en dos fases diferenciadas. En la primera fase, el usuario delimita una longitud de corte en las secuencias para diferenciar secuencias más grandes y pequeñas que dicha longitud definida. Las secuencias pequeñas se van alineando con cada una de las secuencias grandes hasta conformar secuencias corregidas de alta calidad. En la segunda fase estas secuencias corregidas se alinean y ensamblan en *contigs* genómicos.

La disponibilidad de un genoma de referencia de *U. minor* ayudaría por un lado a comprender aspectos evolutivos del género, por medio de estudios de genómica comparativa con otros taxones filogenéticamente emparentados y con otras especies forestales con genomas de referencia disponibles (como *Quercus*, *Fraxinus* o *Populus*). También facilitaría y enriquecería estudios más aplicados de genética cuantitativa, aplicados a la mejora; de genética de poblaciones, aplicados a la conservación; o de transcriptómica, para mejorar el conocimiento sobre los mecanismos de resistencia a la grafiosis y a otros estreses ambientales. El genotipo seleccionado para el proceso de secuenciación y ensamblaje ha sido declarado como “resistente a la grafiosis” pues está adaptado a una amplia gama de condiciones anatómicas y fisiológicas que representan una rica reserva de recursos moleculares útiles para su aplicación en el campo de la bioinformática mediante la comparación, por ejemplo, con otros individuos que no resisten a la enfermedad.

4 Metodología

4.1 Materiales vegetales y preparación de bibliotecas

Para la secuenciación *de novo* del genoma de *U. minor* se cultivó el genotipo registrado como “resistente a la grafiosis” V-AD2 en las instalaciones de la Escuela de Ingeniería de Montes, Forestal y del Medio Natural (UPM) mediante técnicas *in vitro* (sobre medio DKW) para así garantizar la posibilidad de repetir la extracción del ADN del mismo genotipo. Siguiendo varios de los protocolos de extracción de ADN de calidad en especies vegetales[20], se utilizó ADN de origen foliar con distintos prelavados para conseguir aislar el ADN de la mucosidad característica presente en estas especies, y entre los cuales dio mejor resultado el prelavado con una disolución de 100mM Tris-HCl a pH8; 50 mM EDTA; 1M NaCl; 1% PVP K-30 ó PVP K-40 preparado para su uso junto con la adición de 2-mercaptoethanol (1% v/v) justo antes de la extracción.

La preparación de las librerías de secuenciación *de novo* en formato PacBio e Illumina se realizaron en Macrogen, (Geumcheon-gu, Seúl, Corea del Sur). Para la construcción de una sola librería de secuenciación en formato PacBio, se necesitaba enviar como mínimo 1µg total de ADN extraído. Dado que se necesitaban construir ocho librerías, se necesitaban 8 µg totales de ADN. Para la construcción de la librería de secuenciación en formato Illumina, se necesitaba enviar 0,2 µg totales. Finalmente, se optó por enviar 114 µg de ADN total que posteriormente fueron purificados en Macrogen para constituir las librerías de secuenciación *de novo*.

Las secuencias de PacBio (SMRT cell runs) alcanzaron un tamaño de 55,8 Gb y una cobertura de 29x; y las secuencias de Illumina se generaron en la tecnología HiSeq2500 2x250 pb, alcanzando un tamaño de 190 Gb y una cobertura 95x.

4.2 Configuración de la *work-station*

El soporte para el funcionamiento de los softwares de ensamblaje fue una *work-station* con sistema operativo Ubuntu v. 18.04.5 (Linux) mantenida por la Unidad de Patología Forestal de la Escuela de Ingeniería de Montes, Forestal y del Medio Natural (UPM). Esta supercomputadora funciona de forma local mediante 64 CPUs independientes, con un procesador Intel® Xeon(R) Gold 6130 CPU @ 2,10GHz × 64, una memoria RAM de 256 GB, y varios discos duros de almacenamiento. Varios de ellos estaban integrados en un RAID 5 con un total de memoria aprovechable de 8TB. Debido a las necesidades de los softwares empleados fue necesario ampliar esta memoria. Las tres memorias integradas en el RAID 5 fueron reasignadas a un RAID 0 (óptimo para la lectura y escritura a alta velocidad ya que dobla su capacidad a la par que ofrece un rendimiento E/S superior) con una memoria total de 12 TB para ser utilizadas como directorio principal de los resultados de las distintas fases del ensamblaje; mientras que dos nuevas memorias se integraron en un RAID 1 (se almacenan dos veces los datos de escritura en el conjunto de unidades de datos), donde, una vez finalizado el proceso de cada software de

ensamblaje, se fueron depositando los archivos finales de cada una de las fases.

Con el objetivo de prevenir posibles cortes eléctricos, se instaló un SAI que posibilita el funcionamiento de la *work-station* durante un amplio periodo de tiempo; y protege tanto el *hardware* de la *work-station*, como los procesos de ensamblaje llevados a cabo.

4.3 Evaluación de la calidad de las secuencias crudas en generadas en la plataforma PacBio

En un primer lugar se reportan los análisis realizados por el proveedor del servicio de secuenciación (Macrogen, Corea del Sur), que consisten en la distribución de los tamaños de las lecturas de la polimerasa y en la distribución de los tamaños de las sublecturas (*subreads*).

Las lecturas de la polimerasa son el producto más crudo que ofrecen los secuenciadores de la plataforma PacBio. Estos estadísticos son facilitados por el servicio de secuenciación a la entrega de los archivos con las lecturas. Según la propia definición del fabricante: “las lecturas de la polimerasa son recortadas para contener sólo las regiones de alta calidad e incluyen bases de los adaptadores, así como potenciales múltiples pases sobre la plantilla de ADN circular”. Por tanto, si bien son el producto más crudo, ya han sido procesadas para eliminar las regiones de baja calidad.

A continuación se hizo un análisis de dichos ficheros de lecturas usando el programa *SequelQC*[21]. La elección se fundamenta en que este programa es específico para la plataforma PacBio Sequel, plataforma en la que se obtuvieron dichas secuencias, ya que es un programa de fácil instalación y cuyas carreras son rápidas y no consumen excesivos recursos.

Se usó la versión 1.1.0 de *SequelQC* que produce de forma directa múltiples estadísticas y gráficos descriptivos que informan sobre la calidad de las secuencias en crudo.

El parámetro PSR (*Polymerase read to Subread Ratio*) estima las veces que un fragmento ha sido leído dentro de una lectura. Idealmente, cuando se pretende ensamblar un genoma, el valor debería acercarse a 1, indicando que ese fragmento se ha leído solo una vez.

El parámetro ZOR (*ZMW occupancy ratio*) indica cuántos de los ZMWs de la celda SMRT (*zero-mode waveguide;* pozos microscópicos que componen el chip de silicio usado en la secuenciación SMRT con una guía de onda desarrollada para la identificación de cada nucleótido en la fase de polimerización) contienen al menos un fragmento de ADN para ser secuenciado. Valores por encima de 0,5 pueden indicar que se cargaron demasiado las celdas por parte del equipo de secuenciación.

Otro estadístico de interés que ofrece el programa *SequelQC* es el estadístico de calidad de montaje de secuencias N50, cuyo valor va ligado al tamaño de la secuencia para la cual el conjunto de lecturas inferiores en longitud totaliza el 50% de la información total. Mediante la administración a *SequelQC* de los archivos *scraps* (el archivo previo al recorte de los adaptadores y los códigos

de barra), este estadístico N50 se calcula para cuatro procesados de lecturas como descriptores de calidad:

- Lecturas mayores contiguas (*CLRs*): conjunto de las lecturas más largas, es decir, aquellas con longitud de sublecturas equivalente a la longitud de lectura de la polimerasa, y por tanto una secuencia continúa generada a partir de una única plantilla (o inserto).
- CLRs conteniendo sublecturas (*subedCLRs*): añade respecto del estadístico N50CLRs, el resto de sublecturas que no se han generado de forma continua.
- Sublecturas (*subreads*): conjunto de las sublecturas, o fragmentos originarios de la lectura generada por la polimerasa.
- Sublecturas mayores (*longest subreads*): conjunto de las sublecturas de mayor longitud.

El programa se instaló según las instrucciones de los desarrolladores, y para su funcionamiento básico necesita de un archivo de texto de entrada que compile los directorios de las secuencias PacBio en formato *bam*. De esta manera, el programa recoge los archivos *bam* de forma automática, evitando añadir todos los archivos *bam* a una misma carpeta, proceso que demanda tiempo y recursos que estaban siendo utilizados para el proceso de ensamblaje. Además, se puede mejorar la evaluación de la calidad de las secuencias añadiendo de la misma forma los archivos *scraps* de formato *bam*, los cuales incluyen los restos de producto del recorte de las secuencias del adaptador.

4.4 Ensamblaje de las secuencias en formato PacBio mediante el software *Canu*

Para el ensamblaje de las secuencias de un genoma, *Canu* se estructura en tres fases diferenciadas: *correction*, *trimming* y *assembly*. Se puede configurar el software para que se realicen las tres fases de manera consecutiva y de forma automática. Sin embargo, haciendo uso de la posibilidad que ofrece *Canu* de dividir entre fases, se afrontó el ensamblaje manualmente para poder actuar de forma precisa en cada fase frente a un error. Para ello, se generó un archivo log enviado desde la línea de comandos de Linux en el que se indicaba dónde se encontraban las lecturas crudas, el formato de éstas, el tamaño del genoma, y el directorio de almacenaje de los resultados de ensamblaje de las secuencias. Además, desde el archivo *log* fue necesario especificar e instalar una versión de *Java* (versión 8) anterior para el correcto funcionamiento de *Canu*, y limitar el uso de las CPUs de 64 a 54 ya que los procesos ligados al ensamblaje del genoma demandan el uso completo de cada una de las CPUs disponibles. Sin esta limitación de uso, la *work-station* se caía constantemente en pleno funcionamiento.

Frente al error, *Canu* primero finaliza el proceso dando pautas de dónde y por qué se puede haber generado. Cuando se vuelve a cargar el proceso, *Canu* se remite al último archivo generado para continuar, y así no tiene que volver a empezar desde el principio. Aun así, en ocasiones fue necesario eliminar

carpetas temporales a medio generar, pues causaban errores de identificación en el funcionamiento de *Canu.*

Todos los procesos de *Canu* siguen el mismo patrón interno de funcionamiento. En el primer paso se cargan las lecturas obtenidas en la fase anterior (en la fase de corrección, se parte de las lecturas en crudo) en la base de datos *gkpStore*. Seguidamente se calcula la superposición entre secuencias mediante el uso k-meros (se dividen las secuencias en fragmentos pequeños de tamaño constante, usualmente entre 15 y 30 pares de bases, que se comparan entre sí) para poder realizar el proceso denominado como solapamiento (*overlapping*). El producto de esta superposición se carga en la base de datos *OvlStore* desde la cual se completan los objetivos de análisis específicos de cada una de las fases en base a hipótesis de lectura y superposición entre las secuencias.

El primer proceso de *Canu* se denomina *correction* (corrección) y tiene como objetivo mejorar la precisión de las bases nucleótidas en las lecturas (PacBio tiene bastante porcentaje de error en las lecturas que se compensa mediante repetidos pasos de secuenciación). El producto del solapamiento de este proceso es el conjunto de las lecturas corregidas y almacenadas en la base de datos *gkpStore*, desde donde parte el segundo proceso de ensamblaje. Este segundo proceso se denomina *trimming* (recorte) y consigue realizar un recorte de cada una de las lecturas, tomando como criterio la calidad de las distintas áreas que componen dicha lectura. Así, las áreas de menor calidad son eliminadas para obtener el conjunto de secuencias de alta calidad, posteriormente almacenadas en la base de datos *gkpStore*. Desde la base de datos *gkpStore* resultante del proceso *trimming* se inicia el último proceso denominado *assembly* (ensamblaje), que consigue unificar el conjunto de las secuencias de alta calidad en *contigs* mediante superposición constante de las secuencias.

De forma general, se clarificó el tamaño del genoma de la especie *U. minor* para permitir al software realizar una estimación de la profundidad de secuenciación. Existen multitud de parámetros que afectan bien de forma general, o bien de forma particular en cada una de las fases. Por ejemplo, *maxThreads* establece el número máximo de subprocesos a ejecutar en cada fase; *rawErrorRate* se usa para establecer la diferencia máxima esperada entre dos pares de bases en una lectura sin corregir; *correctedErrorRate* marca el umbral entendido como la diferencia máxima esperada entre la lectura después de la corrección de errores; *minReadLength* indica que solo se utilizan secuencias mayores que el umbral definido; *minOverlapLength* indica la longitud mínima de superposición; etc. Aun presentando grandes ventajas, para simplificar, y poder establecer una comparación entre los softwares de ensamblaje y debido a que los procesos son computacionalmente muy exigentes, se decidió dejar por defecto la mayor cantidad de parámetros internos de cada uno de los softwares de ensamblaje (Anexo 1). Debido al gran tamaño del genoma del olmo, cada carrera de ensamblaje duraba unas dos semanas, sin tener en cuenta los eventuales fallos eléctricos y errores del software (*bugs*) que ralentizaron el proceso.

4.5 Ensamblaje de las secuencias en formato PacBio mediante el software *Falcon*

Para ensamblar las secuencias de un genoma, *Falcon* se estructura internamente siguiendo el método jerárquico *HGAP* (*hierarchical genome assembly process*).

La primera fase del ensamblaje se corresponde con el premontaje basado en la corrección de los errores encontrados en las lecturas más largas. Previamente, estas lecturas son seleccionadas en función de la longitud de corte definida. Tras la selección de estas lecturas, los conjuntos de lecturas más cortas a dicha longitud de corte son alineadas con las lecturas largas. En la siguiente fase de ensamblaje se parte del producto resultante del premontaje de las lecturas o *preads.* Este conjunto alineado de lecturas, se vuelve a alinear entre sí para conformar los *contigs* genómicos.

A diferencia de *Canu, Falcon* no ofrece de forma sencilla la posibilidad de trabajar de forma independiente en cada proceso de ensamblaje. Sin embargo, es posible configurar el archivo que sirve como guía para el proceso de ensamblaje de forma que se dirija a una fase indicada del proceso. Frente al error, aun dando pautas de dónde se ha producido, y aún con capacidad de remitirse al último archivo generado para poder reanudar el proceso desde ese punto, los archivos de lectura ligados al error se deben eliminar manualmente de las carpetas que albergan los resultados para obtenerlos nuevamente

Todo el proceso de ensamblaje del software *Falcon* se basa en un *script* denominado *fc_run.py* que se envía, desde la Terminal de Ubuntu. Este *script* tiene como entrada un solo archivo de configuración denominado *fc_run.cfg* (Anexo 2), y en él se especificó que los procesos se iban a realizar de forma local en la *work-station*. Además, el archivo de configuración es el encargado de dirigir la lista de archivos fasta a los procesos de ensamblaje.

Los procesos generales de ensamblaje se pueden dividir a su vez en tres fases. La primera fase se denomina *Overlapdetection and error correction of raw reads,* y se basa en la correcta identificación de todos los solapamientos presentes en las lecturas en crudo. Este proceso necesita que los archivos en formato fasta ligados a las lecturas se transformen en una base de datos *dazzler*. Tras la construcción de la base de datos, se debe realizar una comparación de cada lectura con todas las demás. Esta fase del ensamblaje es la fase que más tiempo consume de todo el proceso. Toda esta información se traduce en multitud de archivos de alineación que contienen información sobre los solapamientos, que a su vez son el objeto de corrección de los errores. Finalmente, todos los archivos son comprimidos en un único archivo fasta (*fc_consensus.py*) que contiene las lecturas corregidas.

La segunda fase se denomina *Overlapdetection of corrected reads*, en la que básicamente vuelven a realizarse los procesos íntegros realizados en la primera fase a excepción del último paso en el que se comprimían las lecturas corregidas en un único archivo fasta (*fc_consensus.py*). Esta fase no enmascara las áreas repetitivas del genoma, como sí que sucede en la primera fase. Tanto los recursos computacionales, como el tiempo ligado al desarrollo

de esta segunda fase dependen de lo bien que se haya realizado el primer proceso de corrección.

La tercera fase se denomina *String Graph assembly* y es el último proceso del ensamblaje. Durante su desarrollo, se genera el ensamblaje final de las secuencias, y los resultados del proceso se sintetizan en dos archivos. El primer archivo se presenta como un archivo fasta de todos los *contigs* primarios (los *contigs* primarios pueden considerarse como los tramos continuos más largos de la secuencia ensamblada de forma contigua). El segundo archivo corresponde a un archivo fasta de los *contigs* asociados a cada *contig* primario, es decir, cada una de las variantes estructurales del *contig* primario conformadas tras el ensamblaje de las secuencias.

Se probaron dos combinaciones de parámetros para la primera fase, y otras dos combinaciones de parámetros para la segunda (Anexo 3). En la tercera fase se probaron múltiples combinaciones.

4.6 Pulido de las secuencias en formato PacBio mediante el uso de las secuencias en formato Illumina.

El archivo final del ensamblaje de las secuencias en formato PacBio, sirvió como objeto de referencia para el pulido con las secuencias de Illumina. Para ello, se utilizó el programa *Racon* el cual necesita como entrada los archivos de Illumina en formato *fastq*, un archivo de alnieamientos en formato sam de las secuencias de Illumina (el cual se generó mediante el programa *bwa mem* [22]), y el archivo que contiene los *contigs* de ensamblaje en formato fasta.

Debido al gran tamaño de los archivos, el proceso se tuvo que paralelizar, ya que la RAM necesaria para hacerlo de una vez era superior al 1TB. Primero se alinearon con *bwa mem* las secuencias originales *fastq* (Illumina HiSeq2500 2x250 bp) sobre el archivo completo de referencia del genoma. Posteriormente se filtraron para dejar solo las que tuvieran suficiente calidad de mapeado (comando *samtools view -f 3*). Se fragmentó el archivo *sam* en archivos menores en cada uno conteniendo un grupo de *contigs* del genoma. De aquí se extrajeron las secuencias incluidas en formato *fastq* (comando *samtools fastq*) y se procedió a hacer el pulido por secciones.

4.7 Comparación de las plataformas de ensamblaje.

La comparación entre ambos softwares de ensamblaje se llevó a cabo a través de los resultados de cada fase de ensamblaje en ambos softwares. Se comparó la demanda de recursos en cada una de las fases, así como el tiempo destinado para la consecución de cada proceso de ensamblaje. Del mismo modo, se registraron las limitaciones necesarias para garantizar el funcionamiento de cada uno de los programas; y si fue necesario cambiar alguno de los valores por defecto en cada archivo de configuración ligado al proceso de ensamblaje. Además, se anotó la diferencia entre los archivos de entrada y de salida en cada una de las fases, para observar los criterios de calidad en cuanto a la corrección y eliminación de las lecturas.

5 Resultados

5.1 Calidad de las secuencias crudas de la plataforma PacBio

5.1.1 *Longitud de las lecturas de la polimerasa*

Como se puede observar en la *Tabla 1*, los valores de N50 oscilaron entre 19,1 kb y 23,7 kb aproximadamente. La longitud media osciló entre 11,7 kb y 14,0 kb.

Tabla 1. Estadístico de las lecturas de la polimerasa (fuente Macrogen)

Nombre de la celda	*N50*	*Longitud media (kb)*
VAD2-PacBio-1_1	19,090	11,689
VAD2-PacBio-1_2	22,659	13,228
VAD2-PacBio-1_3	21,617	13,140
VAD2-PacBio-1_4	22,957	13,427
VAD2-PacBio-1_5	21,456	12,414
VAD2-PacBio-1_6	19,644	12,049
VAD2-PacBio-1_7	23,671	14,027
VAD2-PacBio-1_8	20,254	12,261

5.1.2 Longitud de las sublecturas

Los valores de N50 oscilaron entre 12,8 kb y 14,1 kb aproximadamente. La longitud media osciló entre 9,0 kb y 10,0 kb (*Tabla 2*).

Tabla 2. Estadístico de las sublecturas.

Nombre de la celda	*N50*	*Longitud media (kb)*
VAD2-PacBio-1_1	12,868	8,986
VAD2-PacBio-1_2	13,544	9,650
VAD2-PacBio-1_3	14,132	9,998
VAD2-PacBio-1_4	13,301	9,636
VAD2-PacBio-1_5	13,006	9,157
VAD2-PacBio-1_6	13,602	9,483
VAD2-PacBio-1_7	13,779	10,030
VAD2-PacBio-1_8	13,298	9,315

5.1.3 Descriptores de calidad obtenidos con SequelQC

A continuación, se muestran los resultados de los principales estadísticos de calidad de los ocho archivos en formato binario *bam*, generados en las

respectivas ocho carreras que se realizaron en la plataforma PacBio Sequel con el ADN aislado del genotipo V-AD2.

5.1.4 Estadístico PSR

La calidad referida a la preparación de las bibliotecas de secuenciación es homogénea en cuanto a los valores de PSR. De forma general, se puede asumir que las bibliotecas de secuenciación se prepararon con bastante precisión ya que todos los valores de PSR asociados a las distintas bibliotecas superan el valor de 0,74, por lo que más del 74% del conjunto de hebras molde de ADN coinciden con las lecturas de interés. Cuatro de las bibliotecas de secuenciación (bibliotecas 6, 1, 8 y 3; *Tabla 3*) se aproximan, o superan (biblioteca 6), el valor de 0,8 del estadístico PSR; mientras que las restantes bibliotecas (7, 4, 2 y 5) presentan valores del estadístico PSR en el intervalo comprendido entre los valores de 0,74 y 0,76.

Tabla 3. Valores del estadístico PSR específicos de cada biblioteca de secuenciación ordenados de mayor a menor. Fuente: elaboración propia mediante el software *Microsoft Excel* a partir de los datos obtenidos mediante el software *Canu*.

SMRTcell	*Biblioteca sec.*	*PSR*
VAD2-PacBio-1_6/m54229_191129_130040	*6*	*0,810*
VAD2-PacBio-1_1/m54229_191122_074317	*1*	*0,786*
VAD2-PacBio-1_8/m54229_191130_093132	*8*	*0,785*
VAD2-PacBio-1_3/m54229_191128_061535	*3*	*0,777*
VAD2-PacBio-1_5/m54229_191129_024519	*5*	*0,758*
VAD2-PacBio-1_2/m54229_191122_175739	*2*	*0,744*
VAD2-PacBio-1_4/m54229_191128_163009	*4*	*0,741*
VAD2-PacBio-1_7/m54229_191129_231607	*7*	*0,741*

5.1.5 Estadístico ZOR

Los valores del estadístico ZOR específicos de las bibliotecas de secuenciación son homogéneos entre sí. Siguiendo el patrón de distribución encontrado para el estadístico PSR, las mismas cuatro bibliotecas de secuenciación de mayor calidad (*Tabla 3*) se relacionan con un valor del estadístico ZOR mayor, y por tanto una eficacia muy considerable en el emparejamiento de las moléculas de ADN con las ZMWs. Del mismo modo, las restantes cuatro bibliotecas de secuenciación (5, 2, 4, 7, *Tabla 4*), aun con un valor del estadístico algo menor, demuestran una buena eficacia en términos de emparejamiento de las bibliotecas de secuenciación.

Tabla 4. Valores del estadístico ZOR específicos de cada biblioteca de secuenciación ordenados de mayor a menor. Fuente: elaboración propia mediante el software *Microsoft Excel* a partir de los datos obtenidos mediante el software *Canu*.

SMRTcell	*Biblioteca sec.*	*ZOR*
VAD2-PacBio-1_6/m54229_191129_130040	*6*	*0,788*
VAD2-PacBio-1_1/m54229_191122_074317	*1*	*0,770*
VAD2-PacBio-1_3/m54229_191128_061535	*3*	*0,762*
VAD2-PacBio-1_8/m54229_191130_093132	*8*	*0,761*
VAD2-PacBio-1_5/m54229_191129_024519	*5*	*0,739*
VAD2-PacBio-1_2/m54229_191122_175739	*2*	*0,731*
VAD2-PacBio-1_4/m54229_191128_163009	*4*	*0,719*
VAD2-PacBio-1_7/m54229_191129_231607	*7*	*0,716*

5.1.6 Estadísticos N50:

En base a la definición del estadístico N50, se obtienen los cuatro estadísticos específicos de cada biblioteca de secuenciación (*Figura 4*). Los valores específicos de los estadísticos N50CLRs, N50subedCLRs, N50subreads, N50longestsubreads tomados estos de forma individual no presentan diferencias relevantes entre las bibliotecas de secuenciación. Sin embargo, es notable la diferencia en cuanto a los valores en función del estadístico N50 específico al que se hace referencia. Las bibliotecas de secuenciación 5, 4, 7 y 3 se asocian con valores más elevados tomando como referencia los estadísticos N50CLRs y N50subedCLRs, mientras que las bibliotecas de secuenciación 8, 6, 1 y 2 se asocian con valores menores en referencia a los mismos estadísticos. Sin embargo, cuando se hace referencia a los estadísticos N50subreads y N50longestsubreads, resalta el cambio sustancial de la distribución de estos valores, y en especial para la biblioteca de secuenciación 5 (se corresponde con el máximo valor para los estadísticos N50CLRs y N50subedCLRs, y alcanza casi el mínimo valor para los estadísticos N50subreads y N50longestsubreads). Se puede explicar esta diferencia tomando como referencia la biblioteca de secuenciación 5. Esta biblioteca contiene multitud de lecturas del máximo tamaño posible dado que se corresponden con la longitud de lectura de la polimerasa, y por tanto las secuencias han sido generadas de forma continua. Estas lecturas de secuenciación generadas de forma continua representan un alto porcentaje respecto del total de las lecturas generadas en esta biblioteca de secuenciación, y por tanto reducen considerablemente el porcentaje asignado a los fragmentos de lecturas generadas o *subreads*.

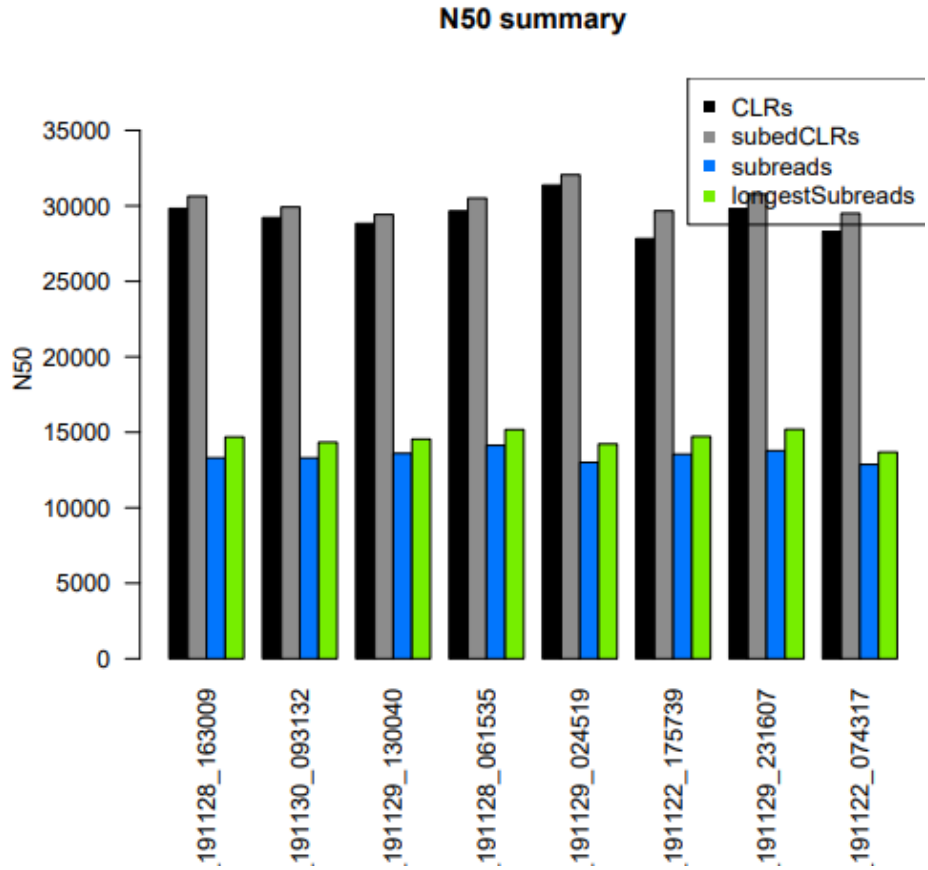

Figura 4: Valores de los estadísticos específicos N50 para cada biblioteca de secuenciación. Fuente: elaboración propia a partir de los datos proporcionados por el software de ensamblaje *Canu* mediante el software *RStudio*.

5.2 Ensamblaje de las secuencias en formato PacBio mediante el software *Canu*

Los resultados obtenidos del ensamblaje de las secuencias en formato PacBio mediante el software *Canu* se pueden estructurar en relación a cada una de las tres fases del proceso de ensamblaje.

5.2.1 Correction

En el proceso de corrección se cargaron un total de 5.311.721 lecturas que contenían un total de 54.686.822.511 pb. Del total de las lecturas cargadas se eliminaron 462.314 lecturas que incluían 220.354.363 pb. Esto supone la eliminación de casi un 9% de las lecturas totales; pero solamente una eliminación del 0,40% del total de las bases nucleótidas. Aunque el conjunto de las lecturas eliminadas supone una reducción alarmante respecto del total de las lecturas cargadas, dada su reducida longitud en términos de nucleótidos, no supone una reducción significativa respecto del total de la información genética disponible. La explicación reside en que, por defecto, el conjunto de lecturas con menos nucleótidos que un valor umbral previamente definido (1.000 pb), se descartaron para los procesos posteriores del ensamblaje. Esta información se especifica para cada una de las bibliotecas de secuenciación, pudiendo observar que las bibliotecas de secuenciación 1 y 5 (*Tabla 5*) se asocian con eliminaciones más drásticas tanto en términos de lecturas como en bases

nucleótidas, ya que estas bibliotecas de secuenciación contenían un amplio conjunto de lecturas de tamaño reducido. En el otro extremo, se puede observar un menor conjunto de lecturas de tamaño reducido en las bibliotecas de secuenciación 3 y 7 (*Tabla 5*) y por tanto un mayor grado de conservación de los nucleótidos para las siguientes fases del ensamblaje.

Tabla 5. Balance de las lecturas cargadas y eliminadas para cada biblioteca de secuenciación durante el proceso de corrección. Fuente: elaboración propia mediante el software *Microsoft Excel* a partir de los datos obtenidos mediante el software *Canu*.

BS	*Reads loaded*	*bp loaded*	*reads skipped*	*bp skipped*	*%read skipped*	*%bp skipped*
1	678237	6643833355	64359	29502503	9,49	0,44
2	500610	5255158990	46142	21102658	9,22	0,40
3	634664	6829072369	50778	24174818	8,00	0,35
4	660376	6865671835	54887	26974476	8,31	0,39
5	779007	7777932372	74168	35320151	9,52	0,45
6	687350	7049380419	58987	28349963	8,58	0,40
7	640305	6886201094	48636	23890743	7,60	0,35
8	731172	7379572077	64357	31039051	8,80	0,42

En la *Figura 5*, se puede observar que un gran porcentaje de las lecturas totales tienen un tamaño reducido. En general, las lecturas más cortas son las más frecuentes, y a medida que aumenta el tamaño de las lecturas la frecuencia relativa de las lecturas de tamaño mayor desciende considerablemente.

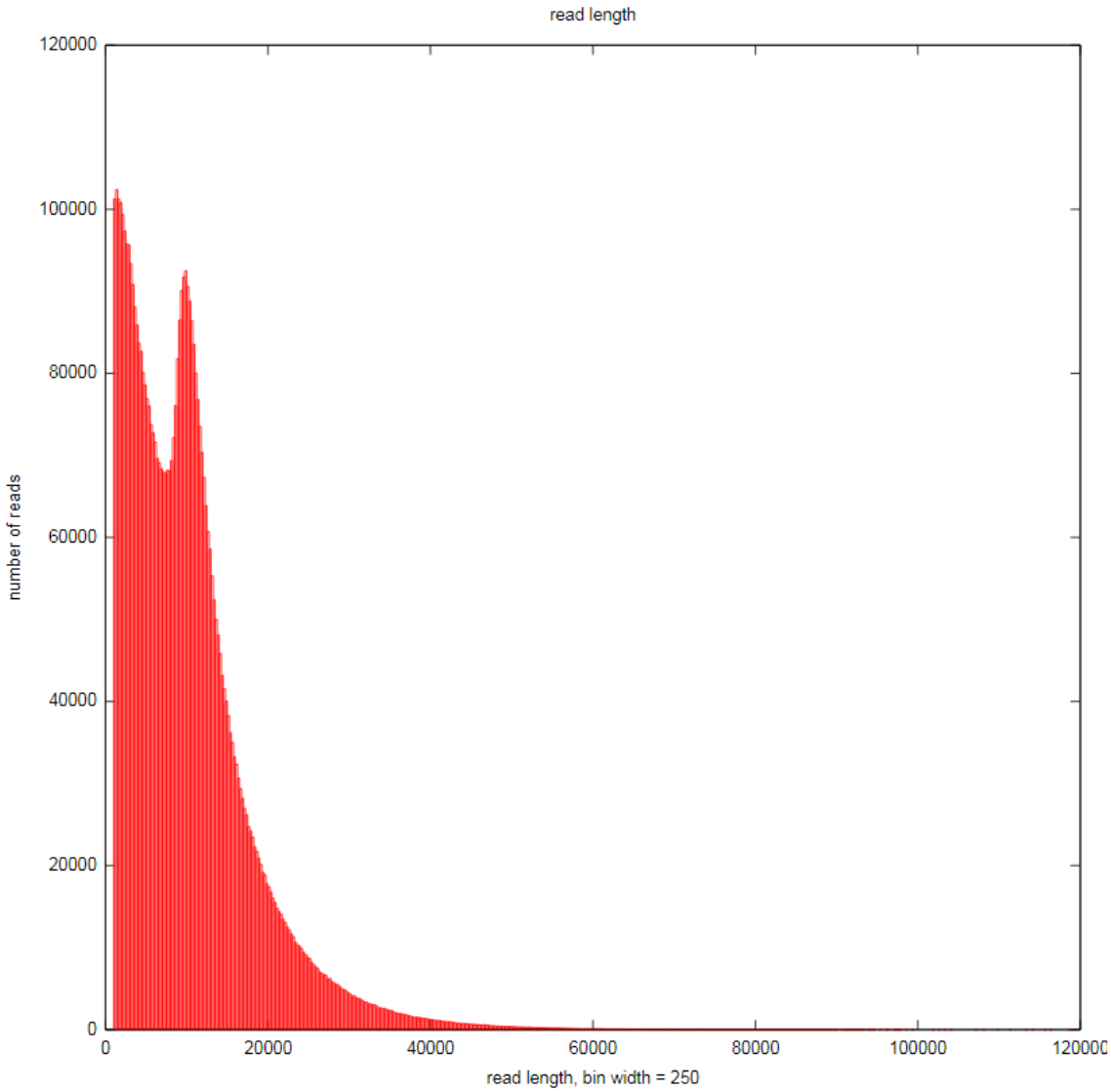

Figura 5: Histograma del tamaño de las lecturas crudas cargadas en la primera fase del ensamblaje. Fuente: elaboración propia a partir de los datos proporcionados por el software de ensamblaje *Canu* mediante el software *RStudio*.

Se definieron fragmentos constantes o k-meros de 16 pb para posteriormente realizar el proceso de solapamiento entre las secuencias. Con esta longitud de k-meros, se obtuvieron 54.607.146.696 mers, de los cuales 2.098.750.006 se identificaron como *distinct mers* (mers que se cuentan solamente una vez aunque aparezcan repetidas veces) y 109.778.090 se identificaron como *single-copy mers* (mers que aparecen una sola vez). El mayor recuento es de 286.211.341, quiere decir que aparece una misma secuencia de 16 pb (muy abundante en el genoma, posiblemente secuencias de telómeros o centrómeros). El conjunto de los mers identificados como *single-copy mers* son los más interesantes biológicamente, y en especial desde el enfoque computacional del ensamblaje ya que permiten el análisis independiente de los fragmentos de longitud definida, y su uso como ancla para el correcto posicionamiento de las lecturas contiguas a estos fragmentos.

En la *Figura 6* se incluyen los criterios definidos por el usuario para la búsqueda de los distintos solapamientos presentes entre las lecturas de secuenciación. Prácticamente todas las lecturas compartían fragmentos de nucleótidos entre sí (*Figura 6*); y de éstas, la mayor parte se solapaba más de un 95% con otras lecturas.

PARAMETERS		
	40	expected coverage
	0	don't use overlaps shorter than this
	0.000	don't use overlaps with erate less than this
	1.000	don't use overlaps with erate more than this
OVERLAPS		
Ignored	0	< 0.0000 fraction error
	0	> 0.4095 fraction error
	0	< 0 bases long
	0	> 2097151 bases long
Filtered	219292678646	too many overlaps, discard these shortest ones
Evidence	212840650	longest overlaps
Total	219505519296	all overlaps
READS		
	45	no overlaps
	8436	no overlaps filtered
	8972	< 50% overlaps filtered
	52339	< 80% overlaps filtered
	333212	< 95% overlaps filtered
	5303240	< 100% overlaps filtered

Figura 6: Porcentaje de solapamientos encontrados entre las lecturas crudas. Fuente: software *Canu*.

Los resultados del proceso de corrección incluyen además un balance entre las lecturas y bases nucleótidas cargadas frente a las lecturas y bases corregidas. Un 55.1% de las lecturas cargadas no generaron corrección alguna; frente a un 49.9% de lecturas que generaron corrección en al menos un nucleótido. El número de correcciones por lectura se puede encontrar en la *Figura 7*, y se puede afirmar que cuanto mayor es el valor de corrección de nucleótidos por lectura, menor es la frecuencia de lecturas asociadas a ese valor de corrección.

PIECES PER READ	
0	2927066
1	2150876
2	203489
3	26572
4	3241
5	382
6	47
7	3

Figura 7: Número de correcciones por lectura. Fuente: software *Canu*.

En la *Figura 8* se puede observar la distribución de las lecturas en función de su longitud, así como la diferencia de longitud entre las lecturas corregidas frente a las lecturas originales sin corregir. Una vez más, se puede observar que tras el proceso de corrección la cantidad de lecturas se reduce en gran medida; y que el tamaño de las lecturas corregidas se reduce respecto de las lecturas originales o sin corregir.

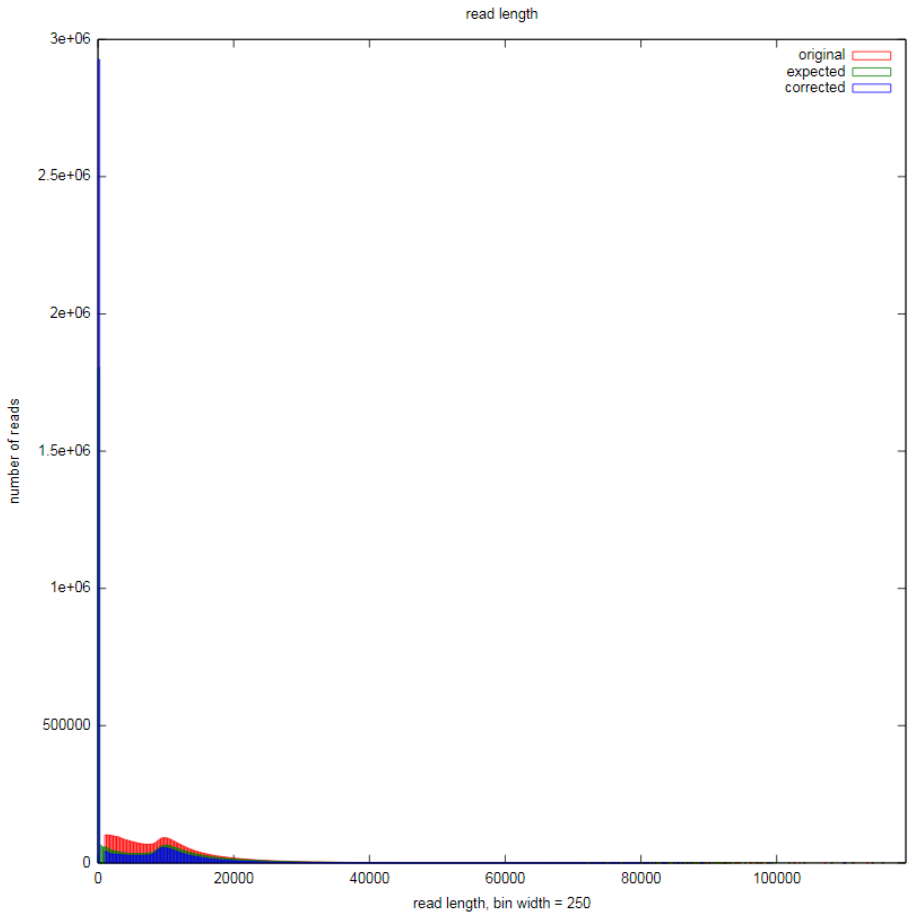

Figura 8: Histograma de longitud original de las lecturas (rojo), esperada (verde) y actual corregida (azul). Fuente: elaboración propia a partir de los datos proporcionados por el software de ensamblaje *Canu*.

5.3 Trimming

En la segunda fase del proceso de ensamblaje solamente se cargaron las lecturas que habían sido corregidas al menos en un nucleótido, es decir, el 49,9% de las lecturas cargadas al comienzo del proceso de ensamblaje. Este subconjunto de lecturas corregidas presentaba una mayor longitud media respecto de las lecturas originales, pues la inmensa mayoría de las lecturas de corta longitud se eliminaron durante el proceso de corrección (*Figura 9*).

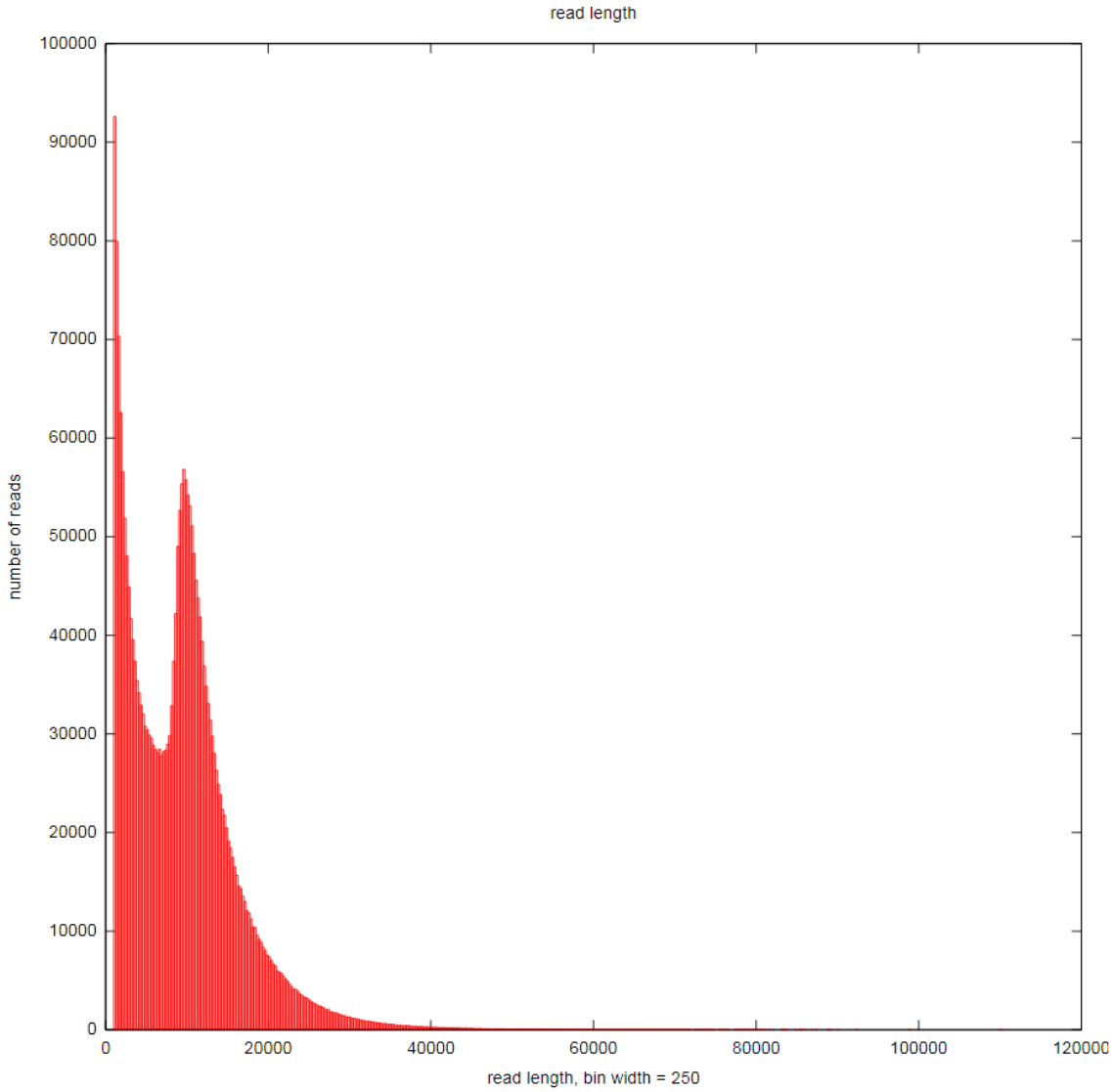

Figura 9: Histograma del tamaño de las lecturas corregidas. Fuente: elaboración propia a partir de los datos proporcionados por el software de ensamblaje *Canu*.

Se volvió a calcular el solapamiento entre las lecturas corregidas para posteriormente eliminar aquellas que no eran compatibles con ninguna de las demás lecturas corregidas. Para ello, previamente se definieron k-meros de 22 pb y se generaron 25.078.405.722 mers, de los cuales 3.668.867.133 se identificaron como *distinct mers;* y 2.191.211.841 como *single-copy mers*.

No hubo un descarte previo de las lecturas corregidas durante el proceso de recorte (*trimming*). En base a los criterios definidos en la *Figura 10*, más del 75% de las lecturas corregidas fueron recortadas, el 20% de las lecturas corregidas se conservaron sin recortar, casi el 1,5% de las lecturas corregidas se eliminaron por no compartir solapamientos con al menos otra lectura corregida, y el 2,16% de las lecturas corregidas se eliminaron de acuerdo a una longitud de recorte menor a la mínima requerida. Un 57,4% de las lecturas fueron recortadas por el extremo 5' de la lectura, mientras que el 54% de las lecturas fueron recortadas por el extremo 3' de la lectura. Dado que la suma de los porcentajes supera el 100%, se concluye que un porcentaje del conjunto de las lecturas fueron recortadas por ambos extremos. La no detección de quimeras puede ser una buena señal (porque son errores) o mala, (porque el programa no es capaz de definirlas) por falta de cobertura (entonces hay que secuenciar más).

PARAMETERS		
1000	reads trimmed below this many bases are deleted	
0.0450	use overlaps at or below this fraction error	
INPUT READS		
reads	**bases**	
2557137	24722495480	reads processed
95610	411617929	reads not processed, previously deleted
0	0	reads not processed, in a library where trimming isn't allowed
PROCESSED		
reads	**bases**	
0	0	no overlaps
7105	19799650	no coverage after adjusting for trimming done already
0	0	processed for chimera
0	0	processed for spur
2550032	24702695830	processed for subreads
READS WITH SIGNALS		
reads	**signals**	
0	0	number of 5' spur signal
0	0	number of 3' spur signal
0	0	number of chimera signal
5687	5790	number of subread signal
SIGNALS		
reads	**bases**	
0	0	size of 5' spur signal
0	0	size of 3' spur signal
0	0	size of chimera signal
5790	2381727	size of subread signal
TRIMMING		
reads	**bases**	
3075	22708921	trimmed from the 5' end of the read
2617	16923352	trimmed from the 3' end of the read

Figura 10: Resumen de las lecturas recortadas. Fuente: software *Canu*.

5.3.1 Unitigging

El conjunto de lecturas corregidas que fueron recortadas de acuerdo a los criterios definidos en el proceso anterior (*Figura 10*), así como el conjunto de lecturas corregidas que se mantuvo sin recortar, conformaron el archivo conjunto de entrada para el proceso final del ensamblaje. Una vez más, se repitió el análisis del solapamiento entre las lecturas de entrada mediante k-meros de 22 pb de longitud. Así, se encontraron 22.932.184.349 mers, 3.135.540.233 *distinct mers*, y 1.761.293.828 *single-copy mers*.

Para identificar los *unitigs* se utilizaron 2.557.124 lecturas que contenían un total de 854.843.232 solapamientos. La *Figura 11* informa del descarte de un 5,3% de las lecturas totales durante la fase de recorte; así como la clasificación de las lecturas y solapamientos utilizados para generar los *contigs*, de entre los cuales un 25,9% corresponde a lecturas correctamente ordenadas y de buena calidad que articulan la generación de *unitigs* por ser consistentes con la mayoría de datos disponibles y de fácil ensamblaje, un 17,4% corresponde a lecturas de baja cobertura y por tanto potencialmente de baja calidad para el ensamblaje, un 24,4% ligado a lecturas de repetición y que por tanto deben colocarse de forma específica, y un 12,7% ligado lecturas con alto grado de repetición.

Category	Reads	%	Read Length			Feature Size or Coverage			Analysis
middle-missing	9321	0.36	11267.52	±	7145.35	1086.94	±	1342.35	bad trimming
middle-hump	17970	0.70	8112.45	±	5920.81	414.60	±	855.72	bad trimming
no-5-prime	56438	2.21	11601.31	±	6751.22	213.91	±	591.35	bad trimming
no-3-prime	51896	2.03	11692.30	±	6918.47	228.69	±	601.56	bad trimming
low-coverage	445593	17.43	4667.79	±	3862.85	4.45	±	1.98	easy to assemble, potential for lower quality consensus
unique	663063	25.93	7608.73	±	4815.37	20.79	±	7.66	easy to assemble, perfect, yay
repeat-cont	212660	8.32	8933.33	±	7200.71	2259.86	±	1979.56	potential for consensus errors, no impact on assembly
repeat-dove	717	0.03	29517.11	±	14837.11	1347.28	±	1519.69	hard to assemble, likely won't assemble correctly or even at all
span-repeat	326330	12.76	12460.66	±	6147.22	5899.34	±	5067.06	read spans a large repeat, usually easy to assemble
uniq-repeat-cont	619740	24.24	9816.81	±	4621.26				should be uniquely placed, low potential for consensus errors, no impact on assembly
uniq-repeat-dove	124249	4.86	16658.09	±	6552.92				will end contigs, potential to misassemble
uniq-anchor	14511	0.57	12346.33	±	6422.49	4645.00	±	4426.35	repeat read, with unique section, probable bad read

Figura 11: Resumen de las lecturas seleccionadas para la conformación de los *contigs* genómicos. Fuente: software *Canu*.

Los resultados finales del proceso de ensamblaje se incluyen en dos archivos en formato fasta. El primer archivo incluye la información más relevante en relación al ensamblaje de las lecturas ordenadas que conforman los *contigs* (*Tabla 6*); y de la misma forma, el segundo archivo incluye las lecturas ordenadas de mayor consistencia que conforman los *unitigs* (*Tabla 7*). Se puede observar que el número de *unitigs* de ensamblaje es mucho mayor que el número de *contigs* de ensamblaje, y que todos ellos, al ser producto de un proceso de ensamblaje *de novo*, no se han incluido como parte de *scaffolds* previamente ensamblados. La proporción de *contigs* de gran longitud (>100 K b) es considerablemente superior a los *unitigs* de la misma longitud, y es una de las razones que sostiene que el estadístico N50 asociado a los *contigs* sea muy superior al estadístico N50 ligado a los *unitigs.* Por otro lado, no hay diferencias significativas en cuanto a la representación de cada nucleótido entre los *contigs* y *unitigs* de ensamblaje.

Tabla 6. Resumen de la información más relevante de los *contigs* de ensamblaje. Fuente: elaboración propia mediante el software *Microsoft Excel* a partir de los datos obtenidos mediante el software *Canu*.

Number of contigs	21404	
Number of contigs in scaffolds	0	
Number of contigs not in scaffolds	21404	
Total size of contigs	1095445729	
Longest contig	643653	
Shortest contig	1172	
Number of contigs > 1Kbp	21404	100.00%
Number of contigs > 10Kbp	21279	99.40%
Number of contigs > 100Kbp	1522	7.10%
Number of contigs > 1Mbp	0	0.00%
Number of contigs > 10Mbp	0	0.00%
Mean contig size	51179	
Median contig size	41289	
N50 contig length	57065	
L50 contig count	5801	
contig %A	32.03	
contig %C	17.97	
contig %G	17.93	
contig %T	32.07	
contig %N	0.00	
contig %non-ACGTN	0	
Number of contig non-ACGTN nt	0	

Tabla 7. Resumen de la información más relevante de los unitigs de ensamblaje. Fuente: elaboración propia mediante el software *Microsoft Excel* a partir de los datos obtenidos mediante el software *Canu*.

Number of unitigs	46302	
Number of unitigs in scaffolds	0	
Number of unitigs not in scaffolds	46302	
Total size of unitigs	1330626633	
Longest unitigs	590744	
Shortest unitigs	1000	
Number of unitigs > 1Kbp	46299	100.00%
Number of unitigs > 10Kbp	35831	77.4%
Number of unitigs > 100Kbp	927	2.00%
Number of unitigs > 1Mbp	0	0.00%
Number of unitigs > 10Mbp	0	0.00%
Mean unitig size	28738	
Median unitig size	22534	
N50 unitig length	42238	
L50 unitig count	10085	
unitig %A	31.91	
unitig %C	18.11	
unitig %G	18.09	
unitig %T	31.89	
unitig %N	0.00	
unitig %non-ACGTN	0	
Number of unitig non-ACGTN nt	0	

5.4 Ensamblaje de las secuencias en formato PacBio mediante el software *Falcon.*

Mediante el software *Falcon* se generaron los resultados previos a la generación de los *contigs* genómicos. Para cada carrera, los resultados se presentan como la cantidad de solapamientos encontrados entre las lecturas en cada uno de los procesos. En la primera fase, cuyo objeto son las lecturas sin corregir, se puede observar un incremento notable de las carreras 4 y 5 (mismo valor ya que se mantuvieron los parámetros) respecto de la carrera 2 (*Tabla 8*); y del mismo modo, una vez corregidas las lecturas se encontraron más solapamientos en las carreras 4 y 5 (*Tabla 9*).

5.4.1 Fase de detección de solapamientos y corrección de errores

Tabla 8. Detección de solapamientos de las lecturas sin corregir. Fuente: elaboración propia mediante el software *Microsoft Excel* a partir de los datos obtenidos mediante el software *Falcon*.

Carrera 2	*279.699.992*
Carrera 4	6.800.750.797
Carrera 5	6.800.750.797

5.4.2 Fase de detección de solapamientos de las lecturas corregidas

Tabla 9. Detección de solapamientos de las lecturas corregidas. Fuente: elaboración propia mediante el software *Microsoft Excel* a partir de los datos obtenidos mediante el software *Falcon*.

Carrera 2	*550*
Carrera 4	1.774.626
Carrera 5	402.967.408

5.4.3 Fase de ensamblaje final

Esta fase no se pudo concluir para ninguna de las combinaciones de parámetros debido a que los ficheros de prelecturas se generaban vacíos. Se intentó solucionar el problema accediendo a los comandos del *wrapper* de *Falcon* por separado, pero el error persistió, estando relacionado con el uso por parte del Falcon de otro programa específico para la detección de solapamientos: *DAligner* (ver Discusión).

6 Discusión

6.1 SequelQC

Los tamaños de las lecturas de la polimerasa fueron quizás un poco bajos. Según los manuales de proveedor, la longitud media de la lectura de la polimerasa suele ser 20 kb y el N50 de la longitud de las sublecturas puede alcanzar los 30kb. En este estudio, tanto la longitud media de las lecturas de la polimerasa como el N50 de las sublecturas rondaron los 13 kb. Esto probablemente se deba a que el ADN que se aisló estaba más fragmentado que lo deseable, o más probablemente, a que no estaba suficientemente limpio. Un problema del ADN de olmo, es que tiene mucho mucílago, que es difícil de limpiar. Este mucílago probablemente obture las ZMWs y haga que las lecturas decaigan en calidad muy rápidamente, y por eso no se alcancen valores de N50 cercanos a 30-40 kb que sería lo esperable (*Figura 12*).

Los resultados del análisis sin la integración de los archivos *scraps,* incluyen la mayor parte de la información en cuanto a la evaluación de la calidad de las secuencias crudas*;* aunque la integración de los archivos *scraps* se tradujo en un proceso computacional más completo en cuanto a la cantidad de información que se pudo extraer. Mediante la integración de los archivos *scraps* se añaden estadísticos de calidad en relación a las secuencias generadas de forma continua por la polimerasa durante la secuenciación (*Figura 13*).

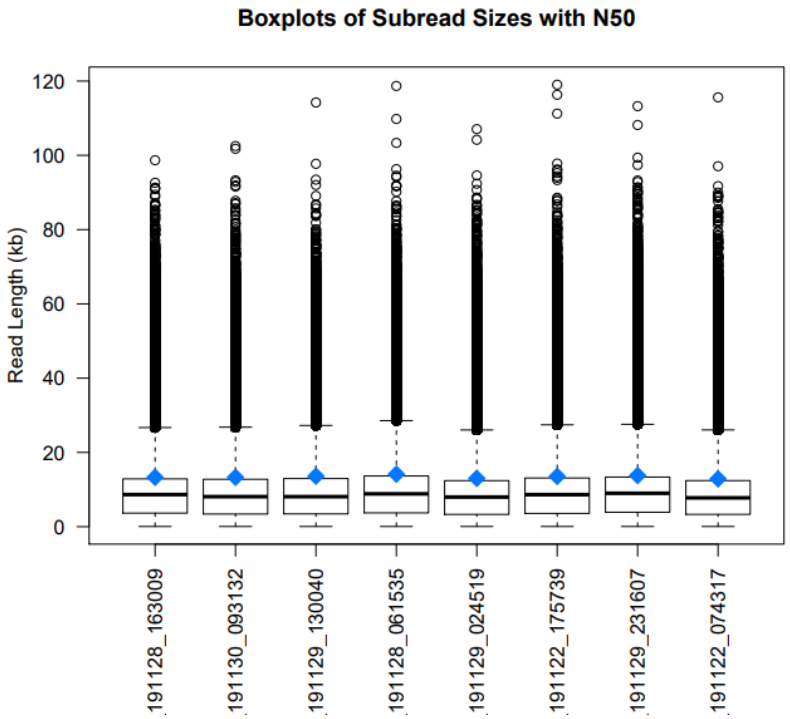

Figura 12: Variación de los tamaños de las lecturas por biblioteca de secuenciación, y el valor que toma el estadístico N50 en cada una de ellas. La distribución de los datos parece seguir el mismo esquema en cada una de las bibliotecas de secuenciación. La dispersión se muestra como similar. Fuente: elaboración propia a partir de los datos proporcionados por el software de ensamblaje *Canu* mediante el software *RStudio*.

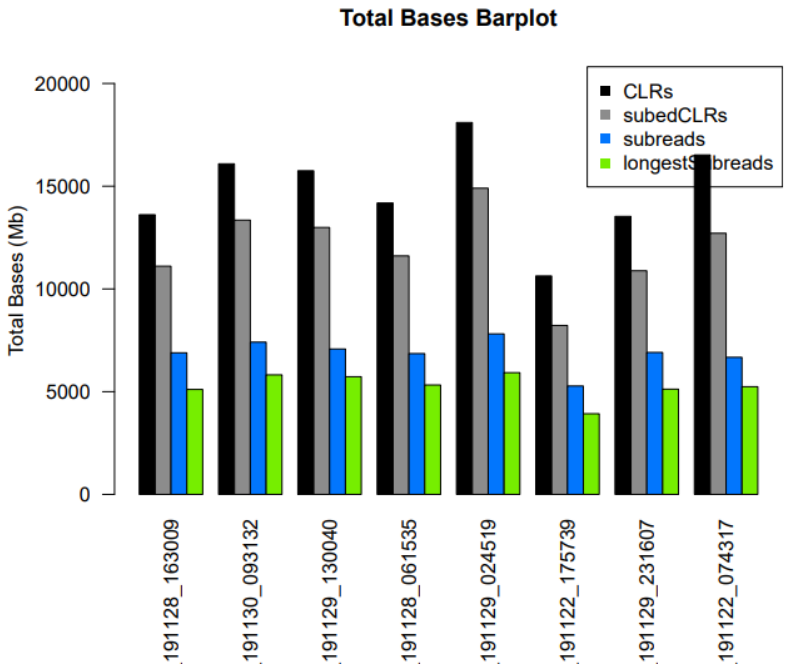

Figura 13. Gráfico de barras múltiple del tamaño total de bases para cada estadístico y biblioteca de secuenciación. Mediante el uso de los archivos scraps, se añade a la gráfica los estadísticos CLRs y subedCLRs

6.2 Canu

6.2.1 Dificultades técnicas

El ensamblaje de las lecturas de secuenciación mediante el uso del software *Canu* se consiguió realizar de forma íntegra mediante la configuración de 3 carreras independientes. La repetición del ensamblaje fue necesaria de acuerdo a la adaptación de las carreras a los recursos computacionales de la *work-station,* junto con la configuración específica de las carreras al formato del conjunto de lecturas de secuenciación. La primera carrera, se generó con los valores por defecto del programa de ensamblaje como base para la adaptación progresiva de estos valores hasta su correcta configuración. Frente al error temprano, fue necesario limitar el proceso de ensamblaje a los recursos locales mediante el uso del parámetro de configuración *useGrid=false,* y limitar el uso de las 64 CPUs que componen la *work-station* a un uso parcial de 54 CPUs. Mediante el ajuste de estos parámetros, se lanzó una segunda carrera independiente (es decir, para no perder información se generó una nueva carpeta de ensamblaje) que comenzó a generar información ligada a la primera fase de corrección de las lecturas de secuenciación. En la misma fase de corrección, se alcanzó un punto en el que constantemente se mostraba un error en la Terminal asociado a la versión del programa *Java* instalado, posteriormente resuelto mediante el script detallado a continuación.

```
#Comando para listar todas las versiones de java instaladas:
sudo update-alternatives --config java
#Comando para cambiar la version default de java:
export JAVA_HOME="$(jrunscript -e 'java.lang.System.out.println(java.lang.System.getProperty("java.home"));')"
```

El proceso de ensamblaje se realizó de forma satisfactoria mediante el uso de una tercera carrera hasta alcanzar los *contigs* y *unitigs* de ensamblaje. Aun así, el proceso generó tal cantidad de información que hubo que instalar dos nuevas memorias NAS de forma específica (como se describe en el apartado de Metodología, apartado de configuración de la *work-station*), para conseguir almacenar todos los archivos de ensamblaje generados; y en determinados procesos computacionales, los requisitos mínimos demandaban más que los máximos definidos en la configuración dando lugar a interrupciones inesperadas. En estos momentos fue más operativo ajustar y solventar las tareas de determinados archivos manualmente a través de la línea de comandos de Linux.

6.2.2 Calidad del ensamblaje

Casualmente, en febrero de 2021 se liberó el borrador de un pre-ensamblaje genoma de *Ulmus americana* en la base de datos NCBI (*Accession number* PRJNA390847). Dado que la liberación del pre-ensamblaje no se acompañó de un artículo, la información del proceso del ensamblaje del genoma de *U. americana* es escasa. Aun así, se indica que fue ensamblado usando lecturas de Illumina mediante el uso del programa *SPAdes*. También se incluye la longitud total del genoma, junto con el estadístico N50. La longitud total una vez ensamblado el genoma de *U. americana* es de 1,26 Gb, por lo que concuerda en gran medida con los resultados obtenidos en el ensamblaje de la especie *U. minor*, ya que la longitud total del genoma ensamblado es de 1,21 Gbp. Sin embargo, es importante recordar que el genoma de *U. americana*, del subgénero Oreoptelea, es de aproximadamente 1,5 Gbp mientras que el de *U. minor*, del subgénero Ulmus, ronda los 2,1 Gbp. El estadístico N50 asociado al genoma de *U. americana* es de 60 Kb, y el estadístico N50 del genoma de *U. minor* es de 57 Kb. Mediante la comparación de estos datos, se puede considerar que nuestro ensamblaje preliminar comparte calidad con el pre-ensamblaje liberado del *U. americana*.

6.3 Falcon

6.3.1 Dificultades técnicas

El ensamblaje de las lecturas de secuenciación mediante el uso del software *Falcon* se consiguió realizar casi al completo mediante la configuración de 6 carreras independientes. En un primer intento, se tomó como referencia la configuración computacional definida en *Canu* para abordar el proceso de ensamblaje. *Falcon*, en general utiliza más memoria RAM, por lo que el uso de las CPUs hubo que limitarlo en varias ocasiones. Por defecto, la configuración interna del software *Falcon* es más extensa y restrictiva que la configuración del software *Canu*.

En las cuatro primeras carreras se encontró el mismo patrón problemático en el funcionamiento del software: los archivos generados, al avanzar de fase se

esperaba que aumentasen en gran medida su tamaño, pero no funcionó como se esperaba y por tanto al alcanzar la fase final del ensamblaje disponían de poca información. La explicación se encontró en los restrictivos criterios de configuración definidos, posteriormente reflejados en una baja calidad de las lecturas tras el proceso de corrección de las lecturas en crudo, solapamientos mal identificados entre las lecturas corregidas, y la imposibilidad de definir lecturas molde como pilares fundamentales del ensamblaje. Poco a poco se fueron ajustando distintos parámetros de configuración de manera específica para cada uno de las cuatro carreras, para así comparar y entender cuánto y cómo afectaban los ajustes en cada proceso del ensamblaje.

En el Anexo 3 se puede encontrar la lista de los principales parámetros que fueron ajustados de manera específica para cada *run*, y que, por tanto, generaron diferencias significativas en los procesos del ensamblaje. La escala de afección de cada uno de los parámetros se encuentra definida en las casillas correspondientes a la fila de “Configuración”.

A escala general, se decidió que tras el error generado en el *run1* asociado al parámetro *input_fofn=input_fofn* y *pa_fasta_filter_option=median*, los parámetros debían sustituirse por *input_fofn= subreads.fasta.fofn* (ya que es el formato de los archivos de entrada) y *pa_fasta_filter_option=streamed-median* (aplica el filtro ZMW de longitud mediana ejecutando una sola pasada sobre los datos). Al ser un genoma de gran tamaño (>10Mb) la configuración en relación a la partición de las lecturas se fue ajustando en función de dos marcadores: el marcador -x se focaliza en el filtrado de secuencias inferiores a un valor de longitud establecido, mientras que el marcador –s controla el tamaño de los bloques almacenados en la DB dazzler (los datos de secuenciación deben estar almacenados en la DB dazzler para la primera y segunda etapa del ensamblaje). El ajuste del parámetro *pa_REPmask_code* hace referencia a la configuración de la búsqueda de repeticiones intercaladas en las lecturas y, salvo en el primer *run*, se realizó tres veces consecutivas (separados por “;”) definiendo el tamaño del grupo y la cobertura.

Durante el pre-ensamblaje de las secuencias se ajustó el valor de tres parámetros diferentes: el valor del parámetro *seed coverage* osciló en el umbral comprendido entre el valor 20 y 40 (salvo en el primer *run* que no se definió) en relación a la cobertura esperada de las lecturas más largas o semilla; el valor del parámetro *length cutoff* se fue ajustando en los diferentes *carreras* en base al tamaño mínimo de las lecturas que se proponen para su uso, y cuando el valor es -1 se está forzando el auto-cálculo de la cobertura de las lecturas más largas o semilla; y el valor *pa_daligner_option* se compone de los parámetros específicos ajustables –k (tamaño de los k-meros), -e (tasa de correlación media; un valor de 0,7 es óptimo para lecturas de baja calidad, mientras que un valor de 0,8 se ajusta bien a lecturas de alta calidad), -l (longitud mínima del solapamiento; un valor pequeño se asocia con librerías de poco tamaño y vicerversa), -h (número total de bases cubiertos por los encuentros de los k-meros). Finalmente, se ajustaron y añadieron parámetros internos que componen *ovlp_daligner_option* en la configuración interna de *Pread-overlapping* como -k, -e, y -l para comparar los resultados obtenidos durante la fase del pre-ensamblaje.

Mediante el ajuste de los parámetros específicos de cada proceso de ensamblaje se consiguieron generar los archivos de lectura corregidos con la información de los solapamientos encontrados en el subconjunto de éstas. Aun así, no se pudo establecer el orden de las lecturas necesario para generar los *contigs* genómicos. El proceso se detenía e informaba de un error complejo y poco común. Se sabe que este error es poco habitual por la inexistencia de información al respecto, ya que la mayoría de los errores computacionales se incluyen en foros específicos de consulta, en el cual los desarrolladores del programa se encargan de resolver la cuestión, o proponen las causas más probables del problema. Por lo que se ha visto el error se genera desde el comienzo del proceso y parece guardar una estrecha relación con la cobertura de las lecturas de secuenciación y el uso por parte de *Falcon* de programa *DAligner*. Si bien el problema podría ser técnico en la parte computacional, el hecho de que se encontrase gran dificultad para encontrar solapamientos a no ser que se redujese el error a valores muy bajos (0,75 para la primera fase), indica que el problema se debe a que la cobertura que se tiene por el momento es baja, unido a la posibilidad de que la calidad de las lecturas también sea menor de la deseable. Como referencia, generalmente en los genomas de especies forestales se considera una cobertura mínima de las lecturas de al menos 60x (Stephen DiFazio, comunicación personal), muy por encima del valor 30x.

6.4 Necesidad de utilizar o no ambos softwares de ensamblaje.

En relación a la complejidad de generar un genoma de alta calidad que sirva de referencia para las especies del género *Ulmus*, se afirma la necesidad de haber utilizado ambos softwares de ensamblaje. El aprendizaje del funcionamiento de los parámetros que pueden afectar en cada proceso específico del software de ensamblaje se considera vital para la comprensión del proceso a escala general, y que por tanto articula una estrategia de ensamblaje sólida a futuro.

Se ha podido comprobar que entre ambos softwares existen diferencias sustanciales que afectan directamente en el proceso de ensamblaje. La principal diferencia, reflejada directamente en los resultados de este proyecto, se podría considerar el valor de cobertura mínima (*coverage*) necesaria para transformar el conjunto de lecturas en crudo hasta *contigs* genómicos. Otra de las más visibles es la diferencia en cuanto a la demanda de memoria de almacenamiento. De media, las carreras de *Falcon* ocuparon poca memoria en el disco duro, mientras que las de *Canu* fueron muy exigentes. El consumo de memoria RAM que se asocia con los procesos internos del software *Falcon* se ha podido estimar muy por encima del necesario para el ensamblaje realizado mediante el software *Canu.* Otra de las diferencias fácilmente observables se asocia con los archivos de configuración generales de cada uno de los softwares (Anexo 1; Anexo 2). Por un lado, *Canu* utiliza un archivo de configuración simple, es decir, no necesariamente se le deben especificar cada uno de los parámetros particulares del proceso en cuestión, y éstos si no se especifican son asignados por defecto; mientras que *Falcon* necesita un archivo de configuración complejo, que compile en detalle la información paramétrica de cada proceso de ensamblaje, e incluso de las fases que conforman la consecución de cada uno de los procesos del ensamblaje (Anexo 2).

La mayor versatilidad de *Falcon* se obtiene a coste de una mayor complejidad para su ejecución. Esto permite que este ensamblador pueda ser usado para proyectos a largo plazo, con un elevado respaldo económico, en los que de tiempo y haya recursos computacionales suficientes para realizar múltiples pruebas con diferentes valores paramétricos. Sin embargo, la automatización de *Canu* permite que pueda ser utilizado en proyectos de menor escala, o en los que el genoma de referencia sea un medio para conseguir otros objetivos, y no el fin último del proyecto.

Sin embargo, para progresar en la obtención de un genoma de referencia de calidad aceptable para *Ulmus minor*, se considera la necesidad de utilizar más de un software de ensamblaje, aunque sea necesario ampliar la cobertura de las lecturas, y así poder liberar ambos resultados para diferentes aplicaciones, además de compararlos y detectar posibles sesgos de alguno de los programas.

6.5 Inclusión de nuevos datos y perspectivas a futuro

En futuras versiones se plantea empezar con el ensamblaje de las secuencias de Illumina mediante el software *SPADes*, y posteriormente realizar el *scaffolding* mediante el uso de las secuencias de PacBio. En todo caso parece necesario aumentar la cobertura de las lecturas de secuenciación disponibles a un mínimo de 60x para resolver los problemas de ensamblaje anteriormente presentados, y obtener un ensamblaje de buena calidad. Además, se plantea la posibilidad de aumentar la amplitud en cuanto a tamaños de las bibliotecas de secuenciación. Por último, la inclusión de lecturas muy largas en formato Oxford Nanopore permitiría la combinación de diferentes tecnologías de secuenciación que potencialmente mejorarían la calidad del ensamblaje.

7 Conclusiones

7.1 Conclusiones

- ✓ El ensamblaje del genoma de un organismo superior (en este caso una especie forestal como es el olmo) es un proceso complejo y laborioso que integra multitud de procesos computacionales.
- ✓ Según los análisis realizados con el software *SequelQC*, la información disponible de PacBio fue de calidad marginalmente inferior de la esperada.
- ✓ Con la información disponible de las plataformas PacBio (29x) e Illumina (95x) se consiguió hacer un ensamblaje del genoma exclusivamente con el software *Canu*, pero no se pudo obtener un ensamblaje con *Falcon*, posiblemente porque la cobertura fue insuficiente.
- ✓ *Canu*, por defecto, es capaz de abordar los procesos de ensamblaje de manera sencilla e íntegra a pesar de la baja cobertura encontrada en las lecturas originales, aunque demanda una amplia memoria de almacenamiento para la escritura de archivos de salida en cada uno de los procesos de ensamblaje.
- ✓ *Falcon* necesita de un archivo de configuración en el que se definan y acoten de forma precisa los parámetros necesarios para cada proceso del ensamblaje, y necesita utilizar mucha memoria RAM para la escritura de archivos de salida.
- ✓ Para solventar el problema, a corto plazo se plantea cambiar la aproximación, haciendo un ensamblaje inicial con las lecturas de Illumina usando *SPADes* y haciendo un *scaffolding* posteriormente con las lecturas de PacBio.
- ✓ Se considera necesario ampliar la cobertura de las secuencias de tamaño largo, o bien haciendo nuevas carreras en PacBio, o bien haciendo carreras con Oxford Nanopore.

7.2 Líneas de futuro

Como se indica en las conclusiones, para las versiones posteriores se plantea la posibilidad de integrar nuevos datos de secuenciación (Oxford Nanopore), ampliar la cobertura de las secuencias utilizadas de PacBio (de 29x a 60x), y probar el orden inverso del proceso de ensamblaje (en primer lugar, ensamblar las lecturas en formato Illumina y completar el ensamblaje mediante las secuencias de PacBio).

7.3 Seguimiento de la planificación

El calendario propuesto incluye una lista de tareas definidas que han guiado mediante objetivos temporales el avance del proyecto. Se sabía desde el principio que los objetivos del proyecto eran ambiciosos debido a la complejidad computacional de los procesos de ensamblaje. Aun así, se han

cumplimentado todos y cada uno de los objetivos planteados. Se es consciente de que el cronograma propuesto en un principio se ha tenido que modificar por las sucesivas repeticiones del ensamblaje mediante ambos softwares, cuyo único objetivo consistía en la mejora de la calidad del ensamblaje.

8 Glosario

1C: valor que define la longitud del genoma incluyendo solamente un juego de cromosomas.

2C: valor que define la longitud del genoma incluyendo los dos juegos de cromosomas completos.

Canu: software de ensamblaje que surge de la bifurcación del *Celera Assembler,* y está diseñado de forma específica para el montaje de secuencias PacBio o Oxford Nanopore.

Carrera: procesos independientes del ensamblaje ligados específicamente a un archivo de configuración que incluye los valores definidos de los parámetros de ensamblaje.

CLR: lecturas con una longitud de sublectura aproximadamente equivalente a la longitud de lectura de la polimerasa, lo que indica que la secuencia se genera a partir de una única plantilla continua desde el principio.

Cobertura: número de veces que se lee una misma posición nucleótida.

Contig: secuencias superpuestas de ADN utilizadas para hacer un mapa físico que reconstruye la secuencia original de ADN de un cromosoma o de una región de un cromosoma.

Falcon: software de ensamblaje configurado para procesar genomas grandes con coincidencia diploide secuenciados con lecturas largas de PacBio.

Genoma: totalidad del material genético que posee un organismo o una especie en particular.

Grafiosis: pandemia que entró en España a mediados de la década de los 30, y cuyo patógeno responsable es el hongo vascular *Ophiostoma ulmi*. A principios de los 70 sobrevino una nueva especie de mayor virulencia denominada *Ophiostoma novo-ulmi* que diezmó las poblaciones autóctonas de la especie *U. minor.*

N50: longitud de lectura en la que el 50% de las bases están incluídas en lecturas más largas o iguales a este valor.

PSR (Polymerase read to Subread Ratio): estadístico de calidad que estima las veces que un fragmento de secuenciación ha sido leído dentro de una lectura.

Unitig: secuencias superpuestas de ADN consistentes con la mayoría de datos disponibles y de fácil ensamblaje utilizadas para hacer un mapa físico que reconstruye la secuencia original de ADN de un cromosoma o de una región de un cromosoma.

Scaffold: conjunto ordenado de contigs en los que puede existir huecos sin secuenciar conectados mediante lecturas de secuencias de extremos emparejados.

SMRT (zero-mode waveguide): pozos microscópicos que componen el chip de silicio usado en la secuenciación SMRT con una guía de onda desarrollada para la identificación de cada nucleótido en la fase de polimerización.

ZMWs: dispositivo nanofotónico incluido en las células SMRT, utilizado para confinar la luz a un pequeño volumen de observación.

ZOR (ZMW occupancy ratio): indica cuántos de los ZMWs de la celda contienen al menos un fragmento de ADN para ser secuenciado.

9 Bibliografía

1. Fu, L., Xin, Y. & Whittemore AT. *Ulmus*. In: (eds.) WZ and PHR, ed. *Flora of China. Vol. 5. Missouri Botanical Garden Press, St. Louis, MO*. ; 2004:1–19.
2. Martín JA, Sobrino-Plata J, Rodríguez-Calcerrada J, Collada C, Gil L. Breeding and scientific advances in the fight against Dutch elm disease: Will they allow the use of elms in forest restoration? *New For*. 2019;50(2):183-215. doi:10.1007/s11056-018-9640-x
3. Hans M. Heybroek (1993). Why bother about the elm? *Dutch elm Dis Res Cell Mol approaches Springer, Berlin*.:pp 1–8.
4. Foster, E.S. HE. *Lucius junius moderatus columella*: on agriculture II. *Harvard Univ Press Cambridge*. Published online 1954.
5. Mataruga Z, Jarić S, Kostić O, et al. The potential of elm trees (*Ulmus glabra* Huds.) for the phytostabilisation of potentially toxic elements in the riparian zone of the Sava River. *Environ Sci Pollut Res*. 2020;27(4):4309-4324. doi:10.1007/s11356-019-07173-9
6. Bernier L, Aoun M, Bouvet GF, et al. Genomics of the dutch elm disease pathosystem: Are we there yet? *IForest*. 2014;8:149-157. doi:10.3832/ifor1211-008
7. GL M. Elm Losses and their Causes over a 20 Year period—A long-term Study of *Ulmus* in Saxony, Germany. In: *3rd International Elm Conference, Florence (Italy) 9–11 October 2013. Book of Abstracts, CNR-IPP Institute of Plant Protection*. ; 2013.
8. Loureiro J, Rodriguez E, Gomes Â, Santos C. Genome size estimations on *Ulmus minor* Mill., *Ulmus glabra* Huds., and *Celtis australis* L. using flow cytometry. *Plant Biol*. 2007;9(4):541-544. doi:10.1055/s-2007-965165
9. Doležel J, Greilhuber J, Suda J. Estimation of nuclear DNA content in plants using flow cytometry. *Nat Protoc*. 2007;2(9). doi:10.1038/nprot.2007.310
10. Koren S, Walenz BP, Berlin K, Miller JR, Bergman NH, Phillippy AM. Canu: Scalable and accurate long-read assembly via adaptive κ-mer weighting and repeat separation. *Genome Res*. 2017;27(5):722-736. doi:10.1101/gr.215087.116
11. Chin CS, Peluso P, Sedlazeck FJ, Nattestad M, Concepcion GT, Clum A, Dunn C, O'Malley R, Figueroa-Balderas R, Morales-Cruz A, Cramer GR, Delledonne M, Luo C, Ecker JR, Cantu D, Rank DR SM. Phased diploid genome assembly with single-molecule real-time sequencing. *Nat Methods*. Published online 2016. doi:10.1038/nmeth.4035
12. Whittemore AT, Xia ZL. Genome size variation in Elms (*Ulmus* spp.) and related genera. *HortScience*. 2017;52(4):547-553. doi:10.21273/HORTSCI11432-16
13. Liu YJ, Wang XR, Zeng QY. De novo assembly of white poplar genome and genetic diversity of white poplar population in Irtysh River basin in China. *Sci China Life Sci*. 2019;62(5):609-618. doi:10.1007/s11427-018-9455-2
14. Whittemore AT, Xia ZL. Ploidy of seeds from odd-polyploid American Elm. *J Am Soc Hortic Sci*. 2020;145(3):186-192. doi:10.21273/JASHS04828-19
15. Salse J. In silico archeogenomics unveils modern plant genome

organisation, regulation and evolution. *Curr Opin Plant Biol*. 2012;15(2):122-130. doi:10.1016/j.pbi.2012.01.001
16. Tuskan GA, DiFazio S, Jansson S, et al. The genome of black cottonwood, *Populus trichocarpa* (Torr. & Gray). *Science (80-)*. 2006;313(5793):1596-1604. doi:10.1126/science.1128691
17. Plomion C, Aury JM, Amselem J, et al. Decoding the oak genome: Public release of sequence data, assembly, annotation and publication strategies. *Mol Ecol Resour*. 2016;16(1):254-265. doi:10.1111/1755-0998.12425
18. Sollars ESA, Harper AL, Kelly LJ, et al. Genome sequence and genetic diversity of European ash trees. *Nature*. 2017;541(7636):212-216. doi:10.1038/nature20786
19. Kovach A, Wegrzyn JL, Parra G, et al. The *Pinus taeda* genome is characterized by diverse and highly diverged repetitive sequences. *BMC Genomics*. 2010;11(1). doi:10.1186/1471-2164-11-420
20. Healey A, Furtado A, Cooper T, Henry RJ. Protocol: A simple method for extracting next-generation sequencing quality genomic DNA from recalcitrant plant species. *Plant Methods*. 2014;10(1). doi:10.1186/1746-4811-10-21
21. Hufnagel DE, Hufford MB, Seetharam AS. SequelQC: Analyzing PacBio Sequel Raw Sequence Quality. *bioRxiv*. Published online 2019.
22. H. L. ligning sequence reads, clone sequences and assembly contigs with BWA-MEM. Published online 2013.

https://github.com/ISUgenomics/SequelQC 13/04/2021

https://canu.readthedocs.io/en/latest/quick-start.html 17/04/2021

https://canu.readthedocs.io/en/latest/tutorial.html#execution 17/04/2021

https://github.com/PacificBiosciences/pb-assembly 1/05/2021

https://dazzlerblog.wordpress.com/command-guides/daligner-command-reference-guide/ 13/05/2021

10 Anexos

Anexo 1: *log* de configuración para el ensamblaje de las lecturas mediante el software *Canu*.

```
## que sólamente use los recursos del ordenador.

useGrid=false

#Comando para listar todas las versiones de java instaladas:
sudo update-alternatives --config java

#Comando para cambiar la version default de java:
export JAVA_HOME="$(jrunscript -e 'java.lang.System.out.println(java.lang.System.getProperty("java.home"));')"

canu -correct \
-p VAD2-PacBio \
-d /media/azken/GENESIS/Elm_genome/Trials/canu/Trial3/ \
java=/usr/lib/jvm/java-8-openjdk/bin/java \
genomeSize=2.1g \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_1/VAD2-PacBio-1_1_subreads.fasta \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_2/VAD2-PacBio-1_2_subreads.fasta \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_3/VAD2-PacBio-1_3_subreads.fasta \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_4/VAD2-PacBio-1_4_subreads.fasta \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_5/VAD2-PacBio-1_5_subreads.fasta \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_6/VAD2-PacBio-1_6_subreads.fasta \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_7/VAD2-PacBio-1_7_subreads.fasta \
-pacbio-raw /media/azken/GENESIS/Elm_genome/PacBio/HN00118617/VAD2-PacBio-1_8/VAD2-PacBio-1_8_subreads.fasta


TRIM

canu -trim \
-p VAD2-PacBio \
-d /media/azken/GENESIS/Elm_genome/Trials/canu/Trial3/ \
java=/usr/lib/jvm/java-8-openjdk/bin/java \
genomeSize=2.1g \
-pacbio-corrected /media/azken/GENESIS/Elm_genome/Trials/canu/Trial3/VAD2-PacBio.correctedReads.fasta.gz


ASSEMBLY

canu -assemble \
-p VAD2-PacBio \
-d /media/azken/GENESIS/Elm_genome/Trials/canu/Trial4/ \
java=/usr/lib/jvm/java-8-openjdk/bin/java \
genomeSize=2.1g \
correctedErrorRate=0.045 \
-pacbio-corrected /media/azken/GENESIS/Elm_genome/Trials/canu/Trial3/VAD2-PacBio.trimmedReads.fasta.gz
```

Anexo 2: *log* de configuración para el ensamblaje de las lecturas mediante el software *Falcon*.

```
#### Input
[General]
input_fofn=subreads.fasta.fofn
input_type=preads
pa_DBdust_option=
pa_fasta_filter_option=streamed-median
target=assembly
skip_checks=False
LA4Falcon_preload=false

#### Data Partitioning
pa_DBsplit_option=-x500 -s400
ovlp_DBsplit_option=-x500 -s400

#### Repeat Masking
pa_HPCTANmask_option=
#no-op repmask param set
pa_REPmask_code=0,300;0,300;0,300

####Pre-assembly
# adjust to your genome size
genome_size=2100000000
seed_coverage=20
length_cutoff=15000
pa_HPCdaligner_option=-v -B128 -M24
pa_daligner_option= -k16 -e0.75 -l2000 -h128 -w8 -s100
falcon_sense_option=--output-multi --min-idt 0.70 --min-cov 4 --max-n-read 200
falcon_sense_greedy=False

####Pread overlapping
ovlp_HPCdaligner_option=-v -B128 -M24
ovlp_daligner_option=-k18 -e0.85 -l2000 -h256 -s100 -M48 -H1000

####Final Assembly
length_cutoff_pr=500
overlap_filtering_setting=--max-diff 100 --max-cov 100 --min-cov 2
fc_ovlp_to_graph_option=

[job.defaults]
job_type=local
pwatcher_type=blocking
JOB_QUEUE=default
MB=196608
NPROC=4
njobs=10
submit = /bin/bash -c ${JOB_SCRIPT} >| ${STDOUT_FILE} 2>| ${STDERR_FILE}
#submit = qsub -S /bin/bash -sync y -V  \
#  -q ${JOB_QUEUE}      \
#  -N ${JOB_NAME}       \
#  -o "${JOB_STDOUT}"  \
#  -e "${JOB_STDERR}"  \
#  -pe smp ${NPROC}    \
#  -l h_vmem=${MB}M    \
#  "${JOB_SCRIPT}"

[job.step.da]
NPROC=4
MB=196608
njobs=10
[job.step.la]
NPROC=4
MB=32768
njobs=10
[job.step.cns]
NPROC=4
MB=196608
njobs=10
[job.step.pda]
NPROC=4
MB=196608
njobs=10
[job.step.pla]
NPROC=4
MB=32768
njobs=10
[job.step.asm]
NPROC=24
MB=196608
njobs=1
```

Anexo 3: combinaciones paramétricas del archivo de configuración del software *Falcon*.

Configuración		***General***		***Data Partitioning***	
Parámetro		***input_fofn***	***pa_fasta_filter_option***	***pa_DBsplit_option***	***ovlp_DBsplit_optio***
Run	0	input.fofn	median	-x500 -s400	-s400
	1	subreads.fasta.fofn	streamed-median	-x500 -s200	-s200
	2	subreads.fasta.fofn	streamed-median	-x500 -s200	-x500 -s200
	3	subreads.fasta.fofn	streamed-median	-x500 -s200	-x500 -s200
	4	subreads.fasta.fofn	streamed-median	-x500 -s400	-x500 -s400
	5	subreads.fasta.fofn	streamed-median	-x500 -s400	-x500 -s400

Configuración		***Repeat Masking***	***Pre-assembly***		
Parámetro		***pa_REPmask_code***	***seed_coverage***	***length_cutoff***	***pa_daligner_option***
Run	0	-	-	1000	-k18 -e0.75 -l1200 -h256 -w8 -s100
	1	0,300;0,300;0,300	40	-1	-k18 -e0.75 -l1200 -h256 -w8 -s100
	2	0,300;0,300;0,300	30	-1	-k18 -e0.80 -l3000 -h256 -w8 -s100
	3	0,300;0,300;0,300	30	-1	-k18 -e0.80 -l3000 -h256 -w8 -s100
	4	0,300;0,300;0,300	20	15000	-k16 -e0.75 -l2000 -h128 -w8 -s100
	5	0,300;0,300;0,300	20	15000	-k16 -e0.75 -l2000 -h128 -w8 -s100

Configuración		***Pread-overlapping***
Parámetro		***ovlp_daligner_option***
Run	0	-k24 -e.92 -l1800 -h600 -s100
	1	-k24 -e.92 -l1800 -h600 -s100
	2	-k24 -e.92 -l1800 -h600 -s100
	3	-k18 -e0.88 -l1800 -h1024 -s100 -M48 -H1000
	4	-k18 -e0.88 -l1800 -h1024 -s100 -M48 -H1000
	5	-k18 -e0.85 -l2000 -h256 -s100 -M48 -H1000