Genome Assembly and De Novo RNA-seq

BMI 7830

Kun Huang

Department of Biomedical Informatics
The Ohio State University
Outline

• Problem formulation
• Hamiltonian path formulation
• Euler path and de Bruijin graph
• Tools
Genome assembly application

- De novo genome sequencing
- Whole genome re-sequencing
- RNA-seq
- Targeted sequencing
- ...
Human Genome Project

Cold Spring Harbor Laboratory
Long Island, New York

June 26, 2000 at the Whitehouse
NGS vs Moore’s Law

Cost per Raw Megabase of DNA Sequence

Moore’s Law

National Human Genome Research Institute

genome.gov/sequencingcosts
Human Genome Sequencing

STRATEGIES FOR SEQUENCING THE HUMAN GENOME

BY MAPPED CLONES

1. Construction of maps of ordered landmarks (genetic markers, genes): provides long-range map and organisation into individual chromosomes.
2. Physical maps of overlapping clones anchored to the landmark maps.
3. Selection of tile path (clones in red)
4. Shotgun sequencing and assembly (for working draft); subsequent directed finishing (for reference sequence).

BY WHOLE GENOME SHOTGUN

1. Shotgun sequencing of short-insert clones
2. Paired end sequencing of large-insert clones
3. Assembly of seed contigs (unifigs)
4. Incorporation of other sequences, and integration of long-range data.

http://www.sanger.ac.uk/HGP/draft2000/gfx/fig2.gif
Genome Mapping

- **STS** – *sequence-tagged sites* (short segments of unique DNA on every chromosome – defined by a pair of PCR primers that amplified only one segment of the genome)
- **BAC** – Bacterial artificial chromosome, 100-400kb
- **YAC** – Yeast artificial chromosome, 150kb-1.5Mb
- **Contig** – assembled *contiguous* overlapping segments of DNA from BACs and YACs
- **ESTs** – Expressed Sequence Tags
- **UniGene Database** – a database for ESTs
Genome Mapping - Evaluation

- Genome coverage
- Contig size
- Quality – error
- N50
Shotgun Sequencing

- Segments are short ~2kb
- Problem with repeated segments or genes

Concepts in Biochemistry, 2nd Ed., R. Boyer
Overlap graph formulation

- Treat each sequence as a “node”
- Draw an edge between two nodes if there is significant overlap between the two sequences
- This is called an Overlap Graph
- Hopefully the contig covers all or large number of sequences, once for each sequence
Instead of traversing edges, how about nodes?

- Hamiltonian path/cycle problem
- NP-complete – current no efficient accurate algorithm, only heuristic ones
History of graph theory

- Seven bridge in Konigsberg
- Is there a path that goes through each bridge only once (and come back to the starting point)?
- Euler first solved this problem and founded Graph Theory
History of graph theory

- Seven bridge in Konigsberg
- Is there a path that goes through each bridge only once (and come back to the starting point)?
- Eulerian path (cycle)
- For path – at most two nodes can have odd degrees (number of edges), the rest all need to have even degrees (please note that the number of odd degree node is either 0 or 2)
- For cycle – all nodes have even degrees
Overlap graph formulation

- Treat each sequence as a “node”
- Draw an edge between two nodes if there is significant overlap between the two sequences
- This is called an Overlap Graph
- Hopefully the contig covers all or large number of sequences, once for each sequence
- In other words, we are looking for Hamiltonian path in the overlap graph
- Pros: straightforward formulation
- Cons: no efficient accurate algorithm; repeats
Overlap – Layout – Consensus approach

- **Overlap** – find potentially overlapping reads
- **Layout** – (use the overlap graph to) generate small contigs, merge to super-contigs
- **Consensus** – generating the “most common” call for each base and correct sequencing errors
- **Examples**: ARACHNE, PHRAP, CAP, TIGR, CELERA
Hamiltonian path problem - Challenges

• Repeat problem
• It was the best of age of wisdom, it was
  best of times, it was
  it was the age of
  it was the worst of
  of times, it was the
  of times, it was the
  of wisdom, it was the
  the age of wisdom, it
  the best of times, it
  the worst of times, it
  times, it was the age
  times, it was the worst
  was the age of wisdom,
  was the age of foolishness,
  was the best of times,
  was the worst of times,
  wisdom, it was the age
  worst of times, it was

It was the best of times, it was the age of wisdom, it was
the age of foolishness, it was the epoch of belief, it was
the epoch of incredulity, it was the season of Light, it was
the season of Darkness, it was the spring of hope, it was
the winter of despair, we had

– Charles Dickens, A Tale of Two Cities
Hamiltonian path problem - Challenges

• Repeat problem

A Read Layout

R_1 : GACCTACA
R_2 : ACCTACAA
R_3 : CCTACAAG
R_4 : CTACAAGT
A : TACAAGTT
B : ACAAGTGA
C : CAAGTTGA
X : TACAAGTC
Y : ACAAGTCC
Z : CAAGTCGG

B Overlap Graph

Schatz M C et al. Genome Res. 2010;20:1165-1173
Overlap – Layout – Consensus approach

- Looking for Hamiltonian path in the overlap graph
- Pros: straightforward formulation
- Cons: no efficient accurate algorithm; repeats, tangled graph at high coverage;

(a)

(b)
Eulerian path in de Bruijin graph

- Seven bridge in Konigsberg
- Is there a path the go through each bridge only once (and come back to the starting point)?
- Eulerian path (cycle)
- For path – at most two nodes can have odd degrees (number of edges), the rest all need to have even degrees (please note that the number of odd degree node is either 0 or 2)
- For cycle – all nodes have even degrees
Eulerian path discovery

- Algorithmic advantages
- Efficient time
  - Fleury’s algorithm – linear in traversal ($O(|E|)$), bridge detection takes time – naïve implementation takes $O(|E^2|)$, more efficient algorithm can reach to $O(|E|\log^3|E| \log\log|E|)$
  - Hierholzer algorithm – linear
A different formulation – de Bruijin graph

• Instead of being the “nodes”, sequence reads can be “edges” linking fixed size “words”.

A Read Layout

R₁: GACCTACA
R₂: ACCTACAA
R₃: CCTACAAG
R₄: CTACAAGT
A: TACAAGTT
B: ACAAGTTA
C: CAAGTTAG
X: TACAAGTC
Y: ACAAGTCC
Z: CAAGTCCG

B Overlap Graph

C de Bruijin Graph

Schatz M C et al. Genome Res. 2010;20:1165-1173
de Bruijin graph

- Instead of being the “nodes”, sequence reads can be “edges” linking fixed size “words”.
- Now it is ok to have the path come to the same point

After graph construction, try to simplify the graph as much as possible
de Bruijin graph

• Instead of being the “nodes”, sequence reads can be “edges” linking fixed size “words”.

• Now it is ok to have the path come to the same point

• We hope to have path passing all edges (instead of nodes) only once.

• The problem changed from a Hamiltonian path problem to an Eulerian path problem.

• Remember – Eulerian path can be found in polynomial time.

After graph construction, try to simplify the graph as much as possible.
de Bruijin graph

- Instead of being the “nodes”, sequence reads can be “edges” linking fixed size “words”.
- Now it is ok to have the path come to the same point
- We hope to have path passing all edges (instead of nodes) only once.
- The problem changed from a *Hamiltonian* path problem to an *Eulerian* path problem

After graph construction, try to simplify the graph as much as possible.
Devils are in details

- Errors – e.g., clip tips, bubble loops, etc
- Error correction
- Unitig, gaps
- Unitig linking, scaffolding, mate pairs
- Repetitive region classification and statistics
- ...
Current algorithms

- ABYSS
- Velvet
- SOAPdenovo
- EULER-
- ALLPATHS-LG
- Trinity
- Celera (using overlap graph)
- ...

Wexner Medical Center
Current algorithms

- ABBySS – MPI (parallel computing) 168 CPU cores x 96 hours
- Velvet – 2TB memory
- SOAPdenovo – 40 cores x 40 hours, >140GB memory
- EULER-
- ALLPATHS-LG
- Trinity
- Celera (using overlap graph)
- ...
Current algorithms

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>BGI</td>
<td>36</td>
<td>⭐️</td>
<td>⭐️</td>
<td>⭐️</td>
<td>⭐️</td>
<td></td>
<td>⭐️</td>
<td>⭐️</td>
<td>⭐️</td>
</tr>
<tr>
<td>Broad</td>
<td>37</td>
<td>⭐️</td>
<td>⭐️</td>
<td>⭐️</td>
<td>⭐️</td>
<td></td>
<td></td>
<td>⭐️</td>
<td></td>
</tr>
<tr>
<td>WTSI-S</td>
<td>46</td>
<td>⭐️</td>
<td></td>
<td>⭐️</td>
<td></td>
<td>⭐️</td>
<td></td>
<td>⭐️</td>
<td></td>
</tr>
<tr>
<td>CSHL</td>
<td>52</td>
<td>⭐️</td>
<td></td>
<td></td>
<td>⭐️</td>
<td></td>
<td></td>
<td></td>
<td>⭐️</td>
</tr>
<tr>
<td>BCCGSC</td>
<td>53</td>
<td>⭐️</td>
<td></td>
<td>⭐️</td>
<td></td>
<td></td>
<td></td>
<td>⭐️</td>
<td>⭐️</td>
</tr>
<tr>
<td>DOEJGI</td>
<td>56</td>
<td>⭐️</td>
<td>⭐️</td>
<td>⭐️</td>
<td>⭐️</td>
<td></td>
<td></td>
<td>⭐️</td>
<td></td>
</tr>
<tr>
<td>RHUL</td>
<td>58</td>
<td>⭐️</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>WTSI-P</td>
<td>64</td>
<td>⭐️</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>⭐️</td>
<td></td>
</tr>
<tr>
<td>EBI</td>
<td>64</td>
<td>⭐️</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>⭐️</td>
<td></td>
</tr>
<tr>
<td>CRACS</td>
<td>64</td>
<td>⭐️</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>⭐️</td>
<td></td>
</tr>
</tbody>
</table>
Summary

**de Bruijn Graph**

AGA -> GAA -> AAG -> AGT -> GTT -> TTA

ATA -> TAA -> GTC -> TCC

Short read assemblers
- Repeats depends on word length
- Read coherency, placements lost
- Robust to high coverage

**Overlap Graph**

A <-> B <-> X <-> Y

R₁ <-> R₂

W <-> Z <-> C <-> D

Long read assemblers
- Repeats depends on read length
- Read coherency, placements kept
- Tangled by high coverage
Next-generation transcriptome assembly

Jeffrey A. Martin and Zhong Wang

Abstract | Transcriptomics studies often rely on partial reference transcriptomes that fail to capture the full catalogue of transcripts and their variations. Recent advances in sequencing technologies and assembly algorithms have facilitated the reconstruction of the entire transcriptome by deep RNA sequencing (RNA-seq), even without a reference genome. However, transcriptome assembly from billions of RNA-seq reads, which are often very short, poses a significant informatics challenge. This Review summarizes the recent developments in transcriptome assembly approaches — reference-based, de novo and combined strategies — along with some perspectives on transcriptome assembly in the near future.

RNA sequencing (RNA-seq). An experimental protocol that uses next-generation sequencing.

Identifying the full set of transcripts — including large and small RNAs, novel transcripts from unannotated genes, splicing isoforms and gene-fusion transcripts — serves as the foundation for a comprehensive study of the transcriptome.

Reconstructing a comprehensive transcriptome from short reads has many informatics challenges. Similar to short-read genome assembly, transcriptome assembly involves piecing together short, low-quality reads.
a. Generate all substrings of length k from the reads

b. Generate the De Bruijn graph

c. Collapse the De Bruijn graph

d. Traverse the graph

e. Assembled isoforms
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>G-Mo.R-Se</td>
<td>No</td>
<td>None</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td>No</td>
<td><a href="http://www.genoscope.cns.fr/externe/gmorse/">http://www.genoscope.cns.fr/externe/gmorse/</a></td>
<td>17</td>
</tr>
<tr>
<td>Cufflinks</td>
<td>No</td>
<td>MP</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td><a href="http://cufflinks.cbcb.umd.edu/">http://cufflinks.cbcb.umd.edu/</a></td>
<td>20</td>
</tr>
<tr>
<td>Scripture</td>
<td>No</td>
<td>None</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td><a href="http://www.broadinstitute.org/software/scripture/">http://www.broadinstitute.org/software/scripture/</a></td>
<td>16</td>
</tr>
<tr>
<td>ERANGE</td>
<td>No</td>
<td>None</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td><a href="http://woldlab.caltech.edu/rnaseq">http://woldlab.caltech.edu/rnaseq</a></td>
<td>50</td>
</tr>
<tr>
<td>Multiple-k</td>
<td>Yes</td>
<td>None</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td><a href="http://www.surget-groba.ch/downloads/">http://www.surget-groba.ch/downloads/</a></td>
<td>19</td>
</tr>
<tr>
<td>Rnnotator</td>
<td>Yes</td>
<td>MP</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>Contact David Gilbert (<a href="mailto:DEGilbert@lbl.gov">DEGilbert@lbl.gov</a>)</td>
<td>15</td>
</tr>
<tr>
<td>Trans-ABySS</td>
<td>Yes</td>
<td>MPI</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>Yes</td>
<td><a href="http://www.bcgsc.ca/platform/bioinfo/software/trans-abys">http://www.bcgsc.ca/platform/bioinfo/software/trans-abys</a></td>
<td>18</td>
</tr>
<tr>
<td>Oases</td>
<td>Yes</td>
<td>MP</td>
<td>Yes</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td><a href="http://www.ebi.ac.uk/~zerbino/oases/">http://www.ebi.ac.uk/~zerbino/oases/</a></td>
<td>-</td>
</tr>
<tr>
<td>Trinity</td>
<td>Yes</td>
<td>MP</td>
<td>Yes</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td><a href="http://trinityrnaseq.sourceforge.net/">http://trinityrnaseq.sourceforge.net/</a></td>
<td>59</td>
</tr>
</tbody>
</table>

MP, multiple processor support (assembler takes advantage of many cores from a single computer); MPI, message-passing interface support (assembler runs in parallel on multiple computers within a cluster).
De novo analysis

- De novo analysis pipeline
- Assembly for contig
- ABySS, SOAPdenovo, Trinity, etc.
- Annotation (e.g., BLAST)