Euler paths and circuits on digraphs and genome sequencing

Nilay Bhatt

Central Connecticut State University

nilaybhatt@my.ccsu.edu

June 2, 2017

Overview

Definitions

Definition (Graph)

A **graph** G consists of a non-empty finite set $V(G)$ of elements called **vertices**, and a finite 'family' $E(G)$ of unordered pairs of (not necessarily distinct) elements of $V(G)$ called **edges**.

Definitions

Definition (Graph)

A **graph** G consists of a non-empty finite set $V(G)$ of elements called **vertices**, and a finite 'family' $E(G)$ of unordered pairs of (not necessarily distinct) elements of $V(G)$ called **edges**.

- Too formal? Think of a graph as a bunch of tennis balls connected with a thread.

Definitions

Definition (Graph)

A **graph** G consists of a non-empty finite set $V(G)$ of elements called **vertices**, and a finite 'family' $E(G)$ of unordered pairs of (not necessarily distinct) elements of $V(G)$ called **edges**.

- Too formal? Think of a graph as a bunch of tennis balls connected with a thread.

Definition (Digraph)

A **digraph** G consists of a non-empty finite set $V(G)$ of elements called **vertices**, and a finite 'family' $E(G)$ of ordered pairs of distinct elements of $V(G)$ called **directed edges**.

Definitions

Definition (Graph)

A **graph** G consists of a non-empty finite set $V(G)$ of elements called **vertices**, and a finite 'family' $E(G)$ of unordered pairs of (not necessarily distinct) elements of $V(G)$ called **edges**.

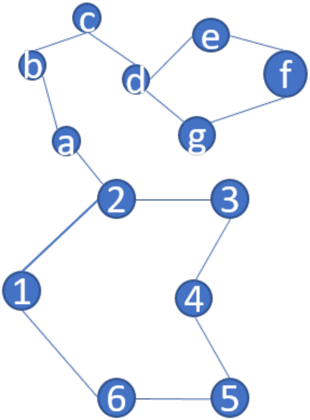
- Too formal? Think of a graph as a bunch of tennis balls connected with a thread.

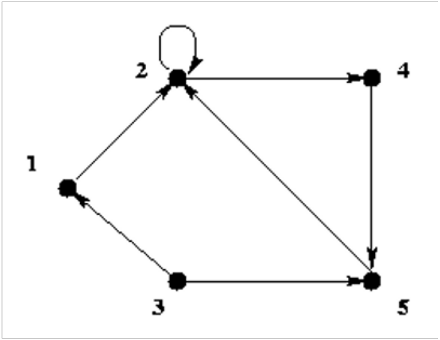
Definition (Digraph)

A **digraph** G consists of a non-empty finite set $V(G)$ of elements called **vertices**, and a finite 'family' $E(G)$ of ordered pairs of distinct elements of $V(G)$ called **directed edges**.

- NYC grid with all the streets being one-way, which then restricts our movement on the graph.

Graphs and Digraphs





Euler Paths vs. Circuits

Definition (Euler Path)

An **Euler path** on a graph G is a special walk that uses each edge exactly once, and it starts and ends at **different** vertices.

Definition (Euler Circuit)

An **Euler circuit** on a graph G is a walk that uses each edge exactly once, and it starts and ends at the **same** vertex.

Criterion for an Euler path or circuit on a graph

Euler Path

A given graph G has an Euler path if and only if the graph is connected and **all but 2 vertices in the graph are of odd degree.**

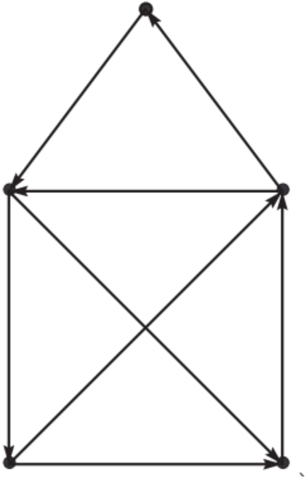
Criterion for an Euler path or circuit on a graph

Euler Path

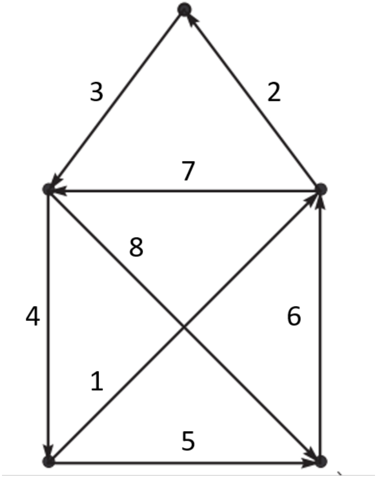
A given graph G has an Euler path if and only if the graph is connected and **all but 2 vertices in the graph are of odd degree.**

- What extra condition do you think we need to add for a digraph to have an Euler path?

Can we find an Euler Path?



YES WE CAN!



Criterion for an Euler path on a digraph

Euler Path

A given graph G has an Euler path if and only if the graph is connected and **all but 2 vertices in the graph are of odd degree.**

Euler Path on a digraph

A digraph D_g has an Euler path iff

- The graph is connected
- All but 2 vertices in the graph are of odd degree
- **the $|\text{indegree} - \text{outdegree}| = 1$ for those two odd vertices in D_g.**

Criterion for an Euler circuit on a digraph

Euler Circuit

A graph G has an Euler circuit if and only if G is connected and all the vertices are of **even degree** .

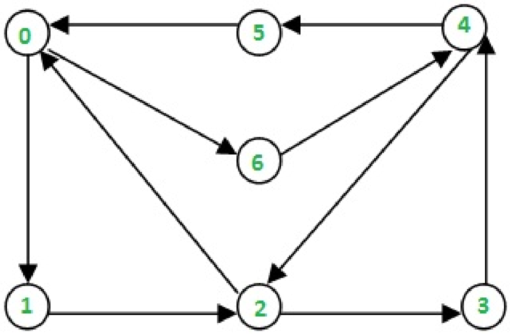
Criterion for an Euler circuit on a digraph

Euler Circuit

A graph G has an Euler circuit if and only if G is connected and all the vertices are of **even degree** .

- What extra condition do you think will be needed on a digraph for it to have an Euler circuit?

Euler Circuit on digraph



Euler Circuit : 0 -> 6 -> 4 -> 5 -> 0 -> 1 -> 2 -> 3 -> 4 -> 2 -> 0

Criterion for an Euler circuit on a digraph

Euler Circuit

A graph G has an Euler circuit if and only if G is connected and all the vertices are of **even degree** .

Euler circuit on a digraph

A graph D_g has an Euler circuit if and only if D_g is connected and all the vertices are of even degree and the **indegree** $=$ **outdegree** for all vertices.

- A formal proof done as part of the final project in Senior Seminar (Spring 2017)

How to find an Euler path and/or circuit on a graph?

Recall: For an Euler Path all but 2 vertices must be of odd degree and to have an Euler circuit, all vertices must be of even degree.

How to find an Euler path and/or circuit on a graph?

Recall: For an Euler Path all but 2 vertices must be of odd degree and to have an Euler circuit, all vertices must be of even degree. While traversing (removing edges once traversed), if removal of a single edge from the remaining connected graph makes it disconnected then such an edge is called a **bridge**.

- To create an Euler path, start at either of the odd vertices.

How to find an Euler path and/or circuit on a graph?

Recall: For an Euler Path all but 2 vertices must be of odd degree and to have an Euler circuit, all vertices must be of even degree. While traversing (removing edges once traversed), if removal of a single edge from the remaining connected graph makes it disconnected then such an edge is called a **bridge**.

- To create an Euler path, start at either of the odd vertices.
- Follow edges one at a time and if you come across a bridge and a non-bridge: Always choose non-bridge

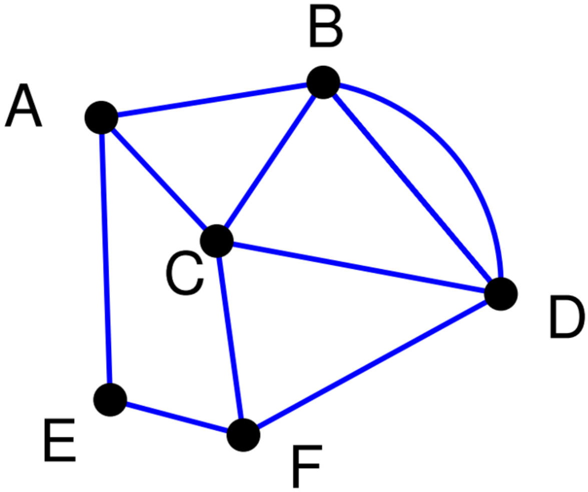
How to find an Euler path and/or circuit on a graph?

Recall: For an Euler Path all but 2 vertices must be of odd degree and to have an Euler circuit, all vertices must be of even degree. While traversing (removing edges once traversed), if removal of a single edge from the remaining connected graph makes it disconnected then such an edge is called a **bridge**.

- To create an Euler path, start at either of the odd vertices.
- Follow edges one at a time and if you come across a bridge and a non-bridge: Always choose non-bridge

This is called **Fleury's Algorithm**

Find an Euler Path?



What's next?

This is cool, but for all the applied mathematicians in the room and computer engineers like me the question becomes:

How does that help me?

Applications

1. Mathematical modeling
2. DNA fragment reconstruction
3. Postman problem
4. Traveling salesman problem
5. many more ....

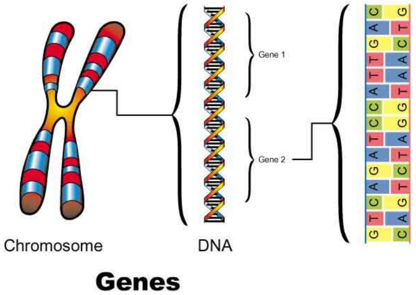
Applications

1. Mathematical modeling
2. **DNA fragment reconstruction**
3. Postman problem
4. Traveling salesman problem
5. many more ....

Genome Sequencing

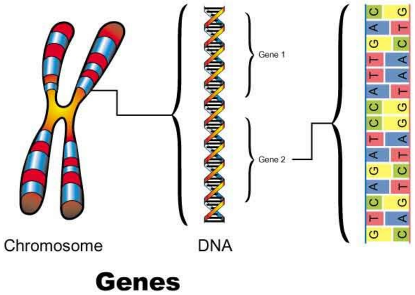
DNA

- Protein that stores the genetic information of a living organism



Genome Sequencing

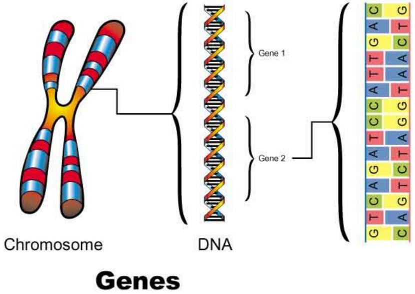
DNA



- Protein that stores the genetic information of a living organism
- Four components to it: A,T,G,C. A connects to T and G connects to C

Genome Sequencing

DNA



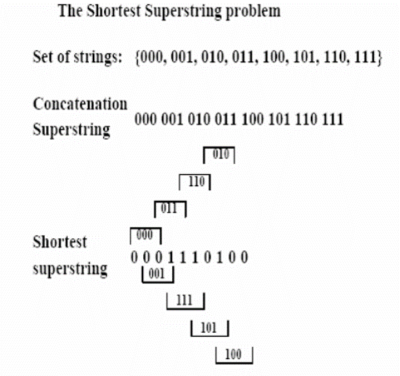
- Protein that stores the genetic information of a living organism
- Four components to it: A,T,G,C. A connects to T and G connects to C
- Fragments of DNA are used in genetic research and discovery
- Strings of such ATGC pairs have this information stored in them

Genome Sequencing

- Reads are taken of a known DNA strand (Sanger, 1977)
- They vary in size from 30-800 nucleotides
- Reads are taken in an overlapping form
- Reconstructing the specific genome with the overlapping to create a superstring is NP-complete

Shortest Super String Problem

Problem: **Given a set of strings, find a shortest string that contains all of them** - NP Complete



De Bruijn Sequence Approach

- Given a string, split it into $k - mers$

De Bruijn Sequence Approach

- Given a string, split it into $k - mers$
- Then split each $k - mers$ into $k - 1 - mer$

De Bruijn Sequence Approach

- Given a string, split it into $k - mers$
- Then split each $k - mers$ into $k - 1 - mer$
- Now assign the two $k - 1 - mers$ as vertices and the $k - mer$ as the name of the directed edge

De Bruijn Sequence Approach

- Given a string, split it into $k - mers$
- Then split each $k - mers$ into $k - 1 - mer$
- Now assign the two $k - 1 - mers$ as vertices and the $k - mer$ as the name of the directed edge
- Consolidate the duplicate edges (collapse them, while conserving the directed edges)

De Bruijn Sequence Approach

- Given a string, split it into $k - mers$
- Then split each $k - mers$ into $k - 1 - mer$
- Now assign the two $k - 1 - mers$ as vertices and the $k - mer$ as the name of the directed edge
- Consolidate the duplicate edges (collapse them, while conserving the directed edges)
- The final directed Euler Path is the super string

What is a $k-mer$ composition of a given genome string

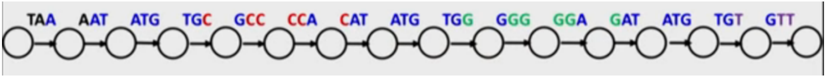
Given a string : $\boldsymbol{TAATGCCATGGGATGTT}$, what is a 3-mer composition?
$= \{TAA, AAT, ATG, TGC, GCC, CCA, CAT,$
$ATG, TGG, GGG, GGA, GAT, ATG, TGT, GTT\}$

- We use these 3-mers as labels of directed edges.

What is a $k - mer$ composition of a given genome string

Given a string : $\boldsymbol{TAATGCCATGGGATGTT}$, what is a 3-mer composition?
$= \{TAA, AAT, ATG, TGC, GCC, CCA, CAT,$
$ATG, TGG, GGG, GGA, GAT, ATG, TGT, GTT\}$

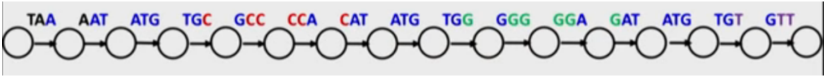
- We use these 3-mers as labels of directed edges.

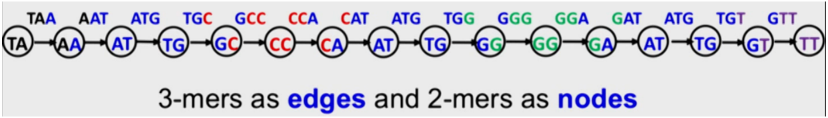


What is a $k-mer$ composition of a given genome string

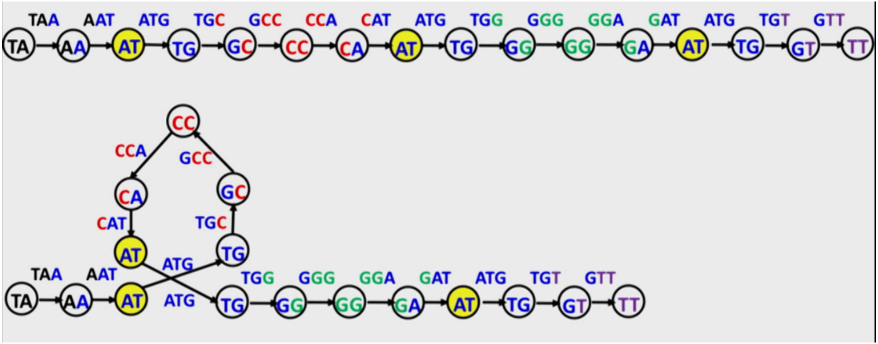
Given a string : ***TAATGCCATGGGATGTT***, what is a 3-mer composition?
$= \{TAA, AAT, ATG, TGC, GCC, CCA, CAT, ATG, TGG, GGG, GGA, GAT, ATG, TGT, GTT\}$

- We use these 3-mers as labels of directed edges.

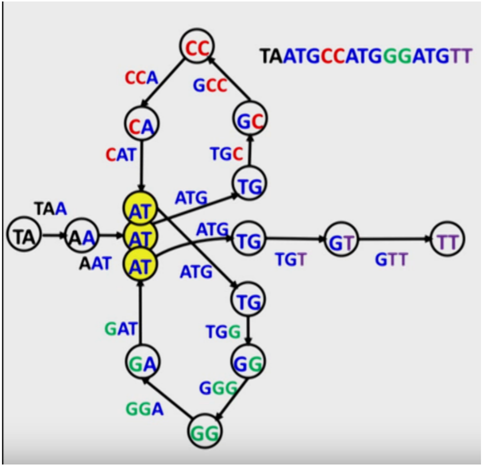




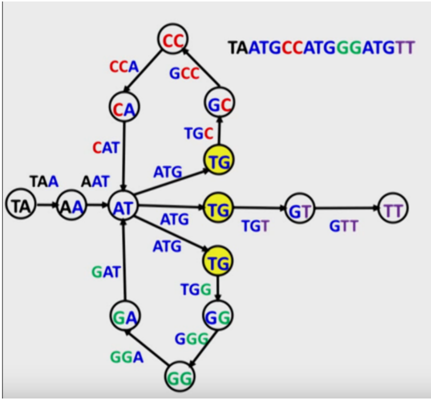
Collapsing like vertices on itself



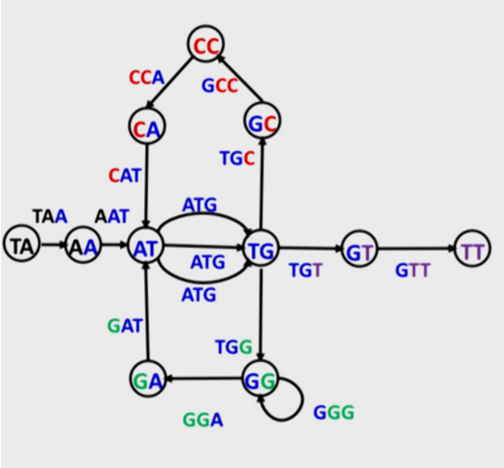
Collapsing like vertices on itself



Collapsing like vertices on itself



DeBruijn Graph of TAATGCCATGGGATGTT



Reconstruction of TAATGCCATGGGATGTT

- An Euler path on this digraph successfully constructs the genome that we are interested in. Of course for the demonstration purposes we started with the genome itself, but in reality all we have are partial reads

Reconstruction of TAATGCCATGGGATGTT

- An Euler path on this digraph successfully constructs the genome that we are interested in. Of course for the demonstration purposes we started with the genome itself, but in reality all we have are partial reads
- Debruijn graph holds the same data as the given genome does!

Reconstruction of TAATGCCATGGGATGTT

- An Euler path on this digraph successfully constructs the genome that we are interested in. Of course for the demonstration purposes we started with the genome itself, but in reality all we have are partial reads
- Debruijn graph holds the same data as the given genome does!
- What if there are multiple Euler paths?

Reconstruction of TAATGCCATGGGATGTT

- An Euler path on this digraph successfully constructs the genome that we are interested in. Of course for the demonstration purposes we started with the genome itself, but in reality all we have are partial reads
- Debruijn graph holds the same data as the given genome does!
- What if there are multiple Euler paths?
- That is where we use paired Debruijn graphs (A special form of 4-mer and 6-mer pairing)

References

- Arratia, Richard, BÃĺla BollobÃąs, Don Coppersmith, and Gregory B. Sorkin. "Euler circuits and DNA sequencing by hybridization." Discrete Applied Mathematics 104, no. 1-3 (2000): 63-96. doi:10.1016/s0166-218x(00)00190-6.
- Pevzner, P. A., H. Tang, and M. S. Waterman. "An Eulerian path approach to DNA fragment assembly." Proceedings of the National Academy of Sciences 98, no. 17 (2001): 9748-753. doi:10.1073/pnas.171285098.
- Bioinfalgorithms. "String Reconstruction as an Eulerian Path Problem." YouTube. May 19, 2014. Accessed June 02, 2017. https://www.youtube.com/watch?v=xYHpuZLvB2s.

Questions?!