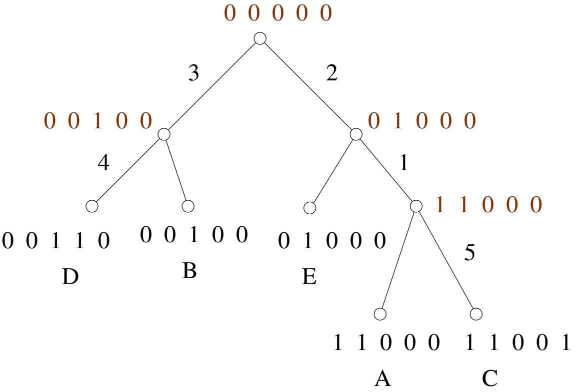
Generalizing the Four Gamete Condition and Splits Equivalence Theorem: Perfect Phylogeny on Three State Characters

Fumei Lam, Dan Gusfield, Srinath Sridhar

12 September 2009

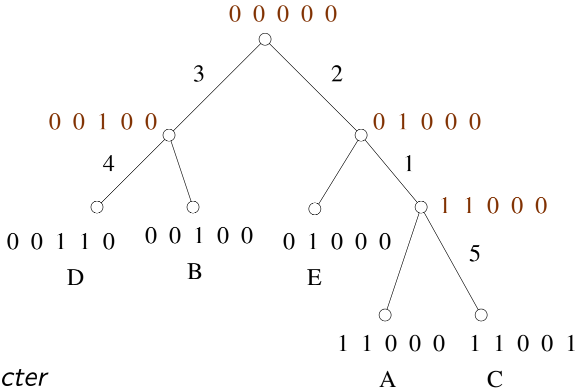
Phylogeny Construction

A	1	1	0	0	0
B	0	0	1	0	0
C	1	1	0	0	1
D	0	0	1	1	0
E	0	1	0	0	0



Phylogeny Construction

A	1	1	0	0	0
B	0	0	1	0	0
C	1	1	0	0	1
D	0	0	1	1	0
E	0	1	0	0	0



Column = *Character*

Row = *Species / Taxa*

Each character takes r *states* (in this example, $r = 2$)

Perfect phylogeny:

Perfect Phylogeny

Perfect phylogeny:

- displays each species on a leaf vertex

Perfect phylogeny:

- displays each species on a leaf vertex
- edges labeled by mutation events

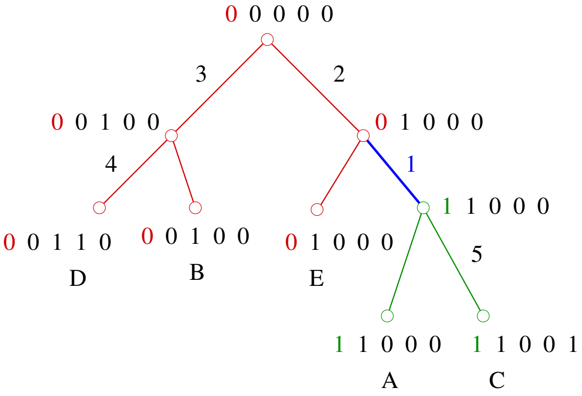
Perfect Phylogeny

Perfect phylogeny:

- displays each species on a leaf vertex
- edges labeled by mutation events
- states in each character appear in a connected subtree

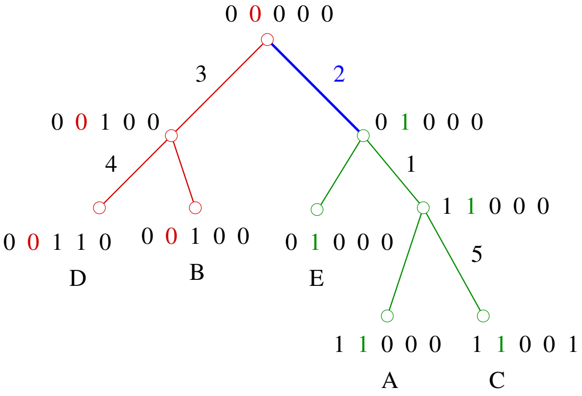
Perfect Phylogeny

A	1	1	0	0	0
B	0	0	1	0	0
C	1	1	0	0	1
D	0	0	1	1	0
E	0	1	0	0	0



Perfect Phylogeny

A	1	1	0	0	0
B	0	0	1	0	0
C	1	1	0	0	1
D	0	0	1	1	0
E	0	1	0	0	0



Perfect Phylogeny Problem

Input: Set S of n species with m characters over r states
Problem: Is there a perfect phylogeny displaying S?

Perfect Phylogeny Problem

Two problems:

1. If there is a perfect phylogeny, construct it
    - Phylogeny is witness for YES answer

Perfect Phylogeny Problem

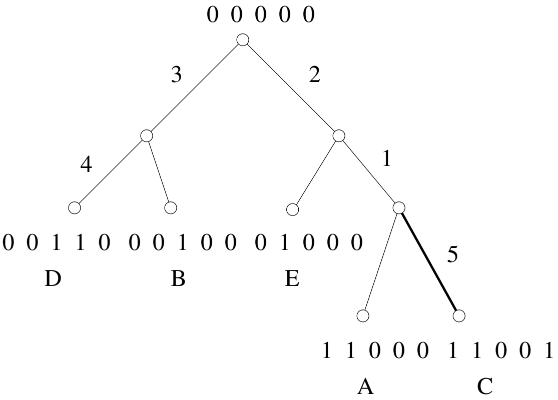
Two problems:

1. If there is a perfect phylogeny, construct it
   - Phylogeny is witness for YES answer
2. If there is no perfect phylogeny, give a certificate of nonexistence
   - What is the witness for NO answer?

Perfect Phylogeny Problem

If perfect phylogeny exists on the entire set of characters, then it exists for any subset of characters

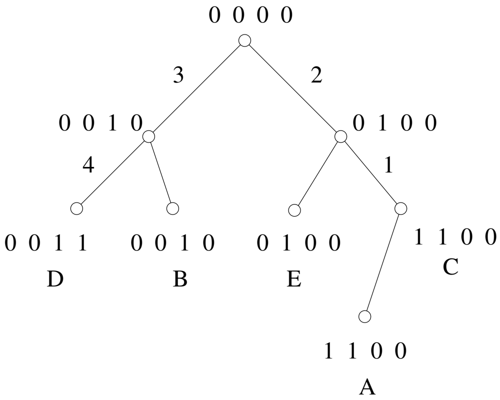
A	1	1	0	0	0
B	0	0	1	0	0
C	1	1	0	0	1
D	0	0	1	1	0
E	0	1	0	0	0



Perfect Phylogeny Problem

If perfect phylogeny exists on the entire set of characters, then it exists for any subset of characters

A	1	1	0	0	0
B	0	0	1	0	0
C	1	1	0	0	1
D	0	0	1	1	0
E	0	1	0	0	0



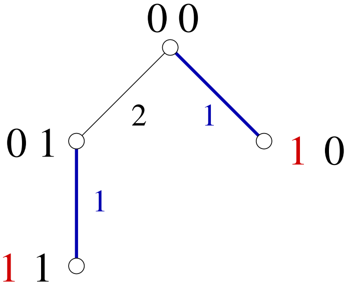
Binary Input: Four Gamete Test

0 0

0 1

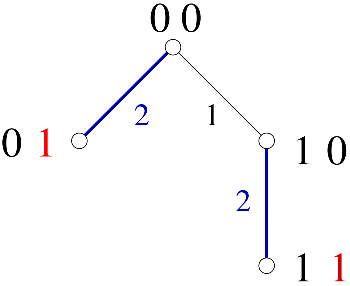
1 0

1 1



Binary Input: Four Gamete Test

0	0
0	1
1	0
1	1



Binary Input: Four Gamete Test

Two problems:

1. If there is a perfect phylogeny, construct it
   - Phylogeny is witness for YES answer
2. If there is no perfect phylogeny, give a certificate of nonexistence
   - What is the witness for NO answer?

Necessary condition for perfect phylogeny on binary input: Each pair of columns must contain at most three out of the four gametes

Binary Input: Four Gamete Test

This condition is both necessary and sufficient.

Binary Input: Four Gamete Test

This condition is both necessary and sufficient.

Four Gamete Test, Splits-Equivalence Theorem (Buneman 1971)

A set of binary sequences allows a perfect phylogeny if and only if no two columns contain all four pairs

$$\begin{matrix} 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \end{matrix}$$

Binary Input: Four Gamete Test

- Fitch (1975), Estabrook and Landrum (1975)
- McMorris (1977)

Binary Input: Four Gamete Test

- Fitch (1975), Estabrook and Landrum (1975)
- McMorris (1977)

Input: Set S of n species and m *binary* characters
Question: Is there a perfect phylogeny displaying S?

- **YES:** If there is a perfect phylogeny, construct it
- **NO:** If there is no perfect phylogeny, output a pair of columns containing all four gametes

Applications

Theoretical results and practical algorithms:

- block partitioning algorithm of HaploBlockFinder
- faster near-perfect phylogeny reconstruction algorithm (Sridhar et. al.)
- phase inference (Gusfield)
- obtaining phylogenies from genotypes (Sridhar et. al.)

Extension to multi-state characters: Fitch Examples

Fitch (1975, 1977) showed an example S on characters over *three* states such that

- every pair of characters in S is compatible
- S does not allow a perfect phylogeny

Extension to multi-state characters: Fitch Examples

Fitch (1975, 1977) showed an example S on characters over *three* states such that

- every pair of characters in S is compatible
- S does not allow a perfect phylogeny

Meacham (1983):

> *"The Fitch example shows that any algorithm to determine whether a set of characters is compatible must consider the set as a whole and cannot take the shortcut of only checking pairs of characters."*
>
> *(Theoretical and Computational Considerations of the Compatibility of Taxonomic Characters)*

Multi-State Perfect Phylogeny

Bounded r (number of states):

- $r = 3$: $O(nm^2)$ algorithm for testing the compatibility of ternary characters (Dress and Steel 1992)

Multi-State Perfect Phylogeny

Bounded r (number of states):

- $r = 3$: $O(nm^2)$ algorithm for testing the compatibility of ternary characters (Dress and Steel 1992)
- $r = 4$: $O(n^2m)$ algorithm for quaternary characters (Kannan and Warnow 1990)

Multi-State Perfect Phylogeny

Bounded r (number of states):

- $r = 3$: $O(nm^2)$ algorithm for testing the compatibility of ternary characters (Dress and Steel 1992)
- $r = 4$: $O(n^2 m)$ algorithm for quaternary characters (Kannan and Warnow 1990)
- Polynomially solvable for all *fixed* r:
  - $O(2^{3r}(nm^3 + m^4))$ algorithm (Agarwala and Fernandez-Baca 1994)
  - $O(2^{2r} nm^2)$ algorithm (Kannan and Warnow 1997)

Generalizing the four gamete test

- If no perfect phylogeny exists for a set of sequences on $r \geq 3$ states, what is the size of the smallest witnessing obstruction set?
- Is it possible to find small obstruction sets (analogous to four gamete test)?

Generalizing the four gamete test

- If no perfect phylogeny exists for a set of sequences on $r \geq 3$ states, what is the size of the smallest witnessing obstruction set?
- Is it possible to find small obstruction sets (analogous to four gamete test)?

We answer these question for $r = 3$.

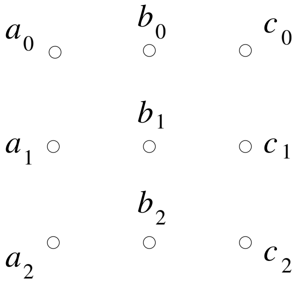
Generalizing the four gamete test

Main Theorem (L., Gusfield, Sridhar, 2008)

Given an input set S with at most three states per character, S admits a perfect phylogeny if and only if every subset of three characters of S admits a perfect phylogeny.

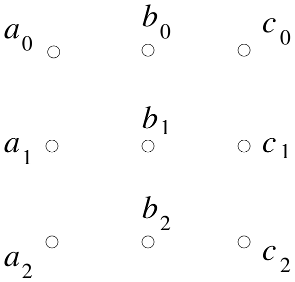
Partition Intersection Graph

a	b	c
0	1	1
0	0	0
1	0	1
1	2	2
2	2	0



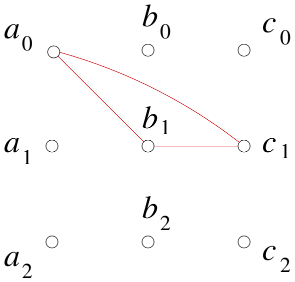
Partition Intersection Graph

a	b	c
a_0	b_1	c_1
a_0	b_0	c_0
a_1	b_0	c_1
a_1	b_2	c_2
a_2	b_2	c_0



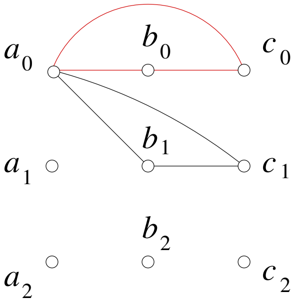
Partition Intersection Graph

a	b	c
a_0	b_1	c_1
a_0	b_0	c_0
a_1	b_0	c_1
a_1	b_2	c_2
a_2	b_2	c_0



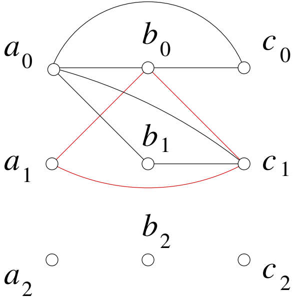
Partition Intersection Graph

a	b	c
a_0	b_1	c_1
a_0	b_0	c_0
a_1	b_0	c_1
a_1	b_2	c_2
a_2	b_2	c_0



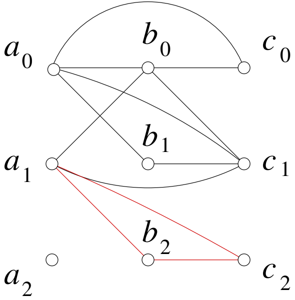
Partition Intersection Graph

a	b	c
a_0	b_1	c_1
a_0	b_0	c_0
a_1	b_0	c_1
a_1	b_2	c_2
a_2	b_2	c_0



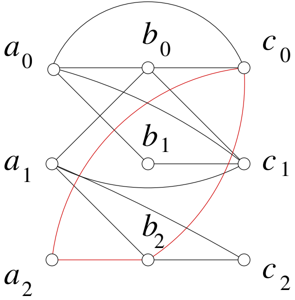
Partition Intersection Graph

a	b	c
a_0	b_1	c_1
a_0	b_0	c_0
a_1	b_0	c_1
a_1	b_2	c_2
a_2	b_2	c_0



Partition Intersection Graph

a	b	c
a_0	b_1	c_1
a_0	b_0	c_0
a_1	b_0	c_1
a_1	b_2	c_2
a_2	b_2	c_0



Partition Intersection Graph

Partition intersection graph $G(S)$:

- vertices correspond to character/state pairs in S
- two character states are adjacent if there exists a row in S containing both

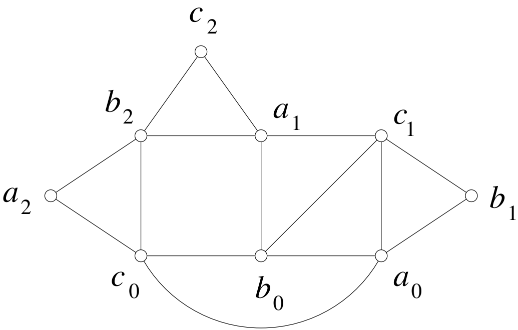
No edge in $G(S)$ between states of the same character.

Chromatic chordal completion problem

A graph H is *chordal*, or *triangulated*, if there are no induced chordless cycles of length four or greater in H.

A graph H is *chordal*, or *triangulated*, if there are no induced chordless cycles of length four or greater in H.

a	b	c
0	1	1
0	0	0
1	0	1
1	2	2
2	2	0



Chromatic chordal completion problem

A graph H is *chordal*, or *triangulated*, if there are no induced chordless cycles of length four or greater in H.

Assign a single color to all the vertices corresponding to the same character in S. A *proper triangulation* of $G(S)$ is a chordal supergraph such that every edge has endpoints with different colors.

Chromatic chordal completion problem

A graph H is *chordal*, or *triangulated*, if there are no induced chordless cycles of length four or greater in H.

Assign a single color to all the vertices corresponding to the same character in S. A *proper triangulation* of $G(S)$ is a chordal supergraph such that every edge has endpoints with different colors.

Theorem (Buneman, 1974)

An input set S admits a perfect phylogeny if and only if the partition intersection graph $G(S)$ has a proper triangulation.

Three-State Perfect Phylogeny: Outline of proof

Idea: Piece together the proper triangulations for each triple of characters to obtain a triangulation for the entire set of characters

Three-State Perfect Phylogeny: Outline of proof

Idea: Piece together the proper triangulations for each triple of characters to obtain a triangulation for the entire set of characters

Check every subset of three characters:

- If some subset of three characters does not allow a perfect phylogeny, output these three characters as the certificate for the nonexistence of a perfect phylogeny

Three-State Perfect Phylogeny: Outline of proof

If every subset of three characters admits a proper triangulation:

$$G(S) \xrightarrow{F-\text{edges}} G'(S) \xrightarrow{F'-\text{edges}} G''(S)$$

Three-State Perfect Phylogeny: Outline of proof

If every subset of three characters admits a proper triangulation:

$$G(S) \xrightarrow{F-\text{edges}} G'(S) \xrightarrow{F'-\text{edges}} G''(S)$$

- Lemma: for every triple of characters, the triangulation is unique

Three-State Perfect Phylogeny: Outline of proof

If every subset of three characters admits a proper triangulation:

$$G(S) \xrightarrow{F-\text{edges}} G'(S) \xrightarrow{F'-\text{edges}} G''(S)$$

- Lemma: for every triple of characters, the triangulation is unique
  - properly triangulate every triple of characters (addition of F-edges)

Three-State Perfect Phylogeny: Outline of proof

If every subset of three characters admits a proper triangulation:

$$G(S) \xrightarrow{F-\text{edges}} G'(S) \xrightarrow{F'-\text{edges}} G''(S)$$

- Lemma: for every triple of characters, the triangulation is unique
  - properly triangulate every triple of characters (addition of F-edges)
- for any chordless cycle in $G(S)$ that remains chordless in $G'(S)$, add chords (F'-edges) of the cycle to obtain $G''(S)$

Three-State Perfect Phylogeny: Outline of proof

If every subset of three characters admits a proper triangulation:

$$G(S) \xrightarrow{F-\text{edges}} G'(S) \xrightarrow{F'-\text{edges}} G''(S)$$

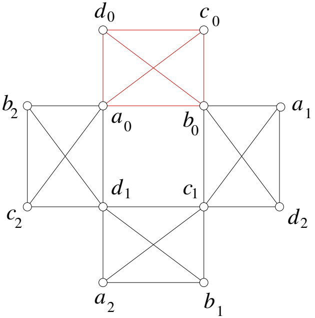
- Lemma: for every triple of characters, the triangulation is unique
  - properly triangulate every triple of characters (addition of F-edges)
- for any chordless cycle in $G(S)$ that remains chordless in $G'(S)$, add chords (F'-edges) of the cycle to obtain $G''(S)$

Claim

$G''(S)$ is a properly triangulated graph

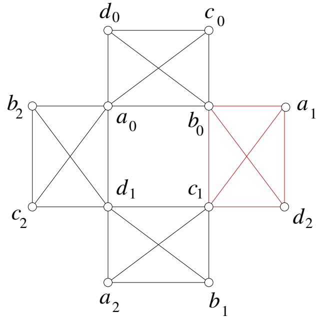
Structure of $G'(S)$

a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1



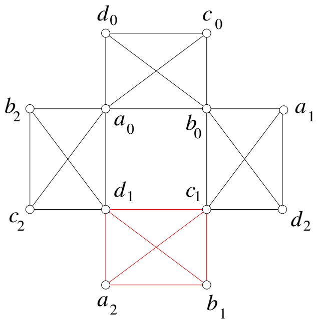
Structure of $G'(S)$

a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1



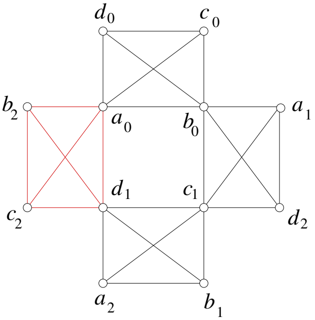
Structure of $G'(S)$

a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1



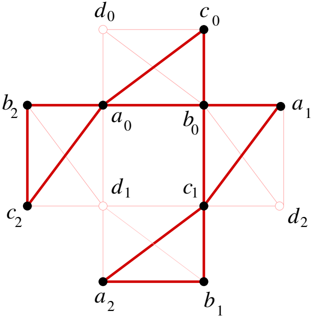
Structure of $G'(S)$

a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1



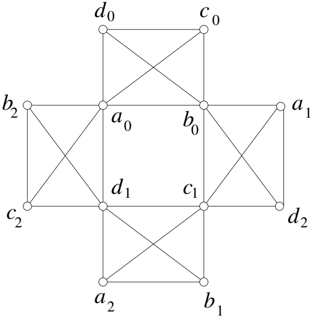
Structure of $G'(S)$

a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1



Structure of $G'(S)$

a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1

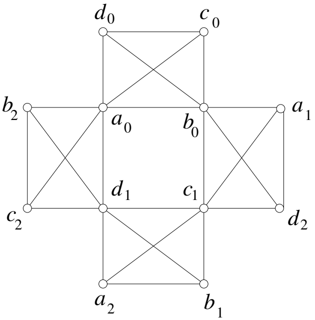


Each triple of characters induces a chordal graph while the entire partition intersection graph $G(S)$ contains a chordless cycle of length four

Structure of $G'(S)$

$$G(S) \xrightarrow{F-\text{edges}} G'(S) \xrightarrow{F'-\text{edges}} G''(S)$$

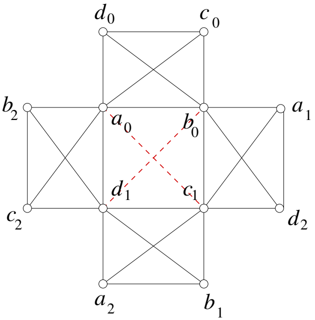
a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1



Structure of $G'(S)$

$$G(S) \xrightarrow{F-\text{edges}} G'(S) \xrightarrow{F'-\text{edges}} G''(S)$$

a	b	c	d
0	0	0	0
1	0	1	2
2	1	1	1
0	2	2	1



Enumerating three character obstruction sets

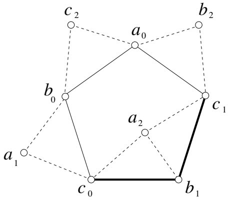
Minimal obstruction sets for trinary input contain three characters.

Enumerate all instances on three characters a, b, and c such that:

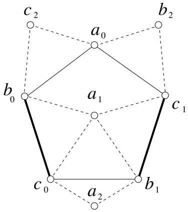
(i) a, b and c are characters on at most three states
(ii) every pair of characters allows a perfect phylogeny
(iii) the three characters a, b, and c together do not allow a perfect phylogeny.

Enumerating three character obstruction sets

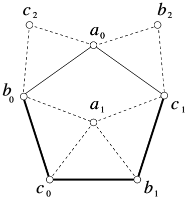
a	b	c
0	0	2
0	2	1
1	0	0
2	1	0
2	1	1



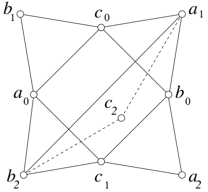
a	b	c
0	0	2
0	2	1
1	0	0
2	1	0
1	1	1



a	b	c
0	0	2
0	2	1
1	0	0
1	1	0
1	1	1



a	b	c
0	1	0
1	0	0
2	0	1
0	2	1
1	2	2



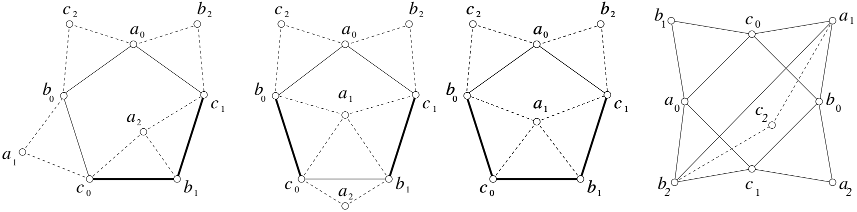
Enumerating three character obstruction sets

a	b	c
0	0	2
0	2	1
1	0	0
2	1	0
2	1	1

a	b	c
0	0	2
0	2	1
1	0	0
2	1	0
1	1	1

a	b	c
0	0	2
0	2	1
1	0	0
1	1	0
1	1	1

a	b	c
0	1	0
1	0	0
2	0	1
0	2	1
1	2	2

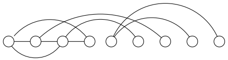


- Conflict or incompatibility between pairs of sites: Meiotic recombination, reticulation and recurrent mutation
- Each site corresponds to a vertex
- (i, j) is an edge if i and j are in conflict.

$$\begin{array}{ccccccccc}
\circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ & \circ \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 1 & 0 & 1 & 1 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 0 & 1 & 1 & 0 \\
0 & 1 & 0 & 1 & 0 & 1 & 1 & 0 & 0
\end{array}$$

Incompatibility/Conflict Graph for binary characters

- Conflict or incompatibility between pairs of sites: Meiotic recombination, reticulation and recurrent mutation
- Each site corresponds to a vertex
- (i, j) is an edge if i and j are in conflict.



0	1	0	0	0	0	0	0	1
0	0	1	0	1	1	0	0	1
1	0	0	0	0	1	1	1	0
1	1	1	1	1	0	1	1	0
0	1	0	1	0	1	1	0	0

Incompatibility (Conflict) Graph

"The non-trivial connected components of the conflict graph are very informative, used both to derive efficient algorithms and to expose combinatorial structure in phylogenetic networks."

– Gusfield, Bansal, Bafna, Song (2006)

Incompatibility Hypergraph for 3-state characters

- Conflict or incompatibility for a set of sites: Meiotic recombination, reticulation and recurrent mutation

Incompatibility Hypergraph for 3-state characters

- Conflict or incompatibility for a set of sites: Meiotic recombination, reticulation and recurrent mutation
- Each site corresponds to a vertex

Incompatibility Hypergraph for 3-state characters

- Conflict or incompatibility for a set of sites: Meiotic recombination, reticulation and recurrent mutation
- Each site corresponds to a vertex
- v_1, v_2, v_3 forms a hyperedge if the corresponding sites do not allow a perfect phylogeny

Incompatibility Hypergraph for 3-state characters

- Conflict or incompatibility for a set of sites: Meiotic recombination, reticulation and recurrent mutation
- Each site corresponds to a vertex
- v_1, v_2, v_3 forms a hyperedge if the corresponding sites do not allow a perfect phylogeny

Character-Removal Problem:

- minimize the number of characters to remove from the data so that the resulting data has a multi-state perfect phylogeny

Incompatibility Hypergraph for 3-state characters

- Conflict or incompatibility for a set of sites: Meiotic recombination, reticulation and recurrent mutation
- Each site corresponds to a vertex
- v_1, v_2, v_3 forms a hyperedge if the corresponding sites do not allow a perfect phylogeny

Character-Removal Problem:

- minimize the number of characters to remove from the data so that the resulting data has a multi-state perfect phylogeny
- hitting set problem

Conclusion

Conclusion

- Improved algorithm for constructing a perfect phylogeny on three states if it exists?

Conclusion

- Improved algorithm for constructing a perfect phylogeny on three states if it exists?
  - $O(2^{3r}(nm^3 + m^4))$ (Agarwala and Fernandez-Baca 1994)
  - $O(2^{2r}nm^2)$ (Kannan and Warnow 1997)

Conclusion

- Improved algorithm for constructing a perfect phylogeny on three states if it exists?
    - $O(2^{3r}(nm^3 + m^4))$ (Agarwala and Fernandez-Baca 1994)
    - $O(2^{2r}nm^2)$ (Kannan and Warnow 1997)
- Correlation between incompatible subsets of characters?

THANKS