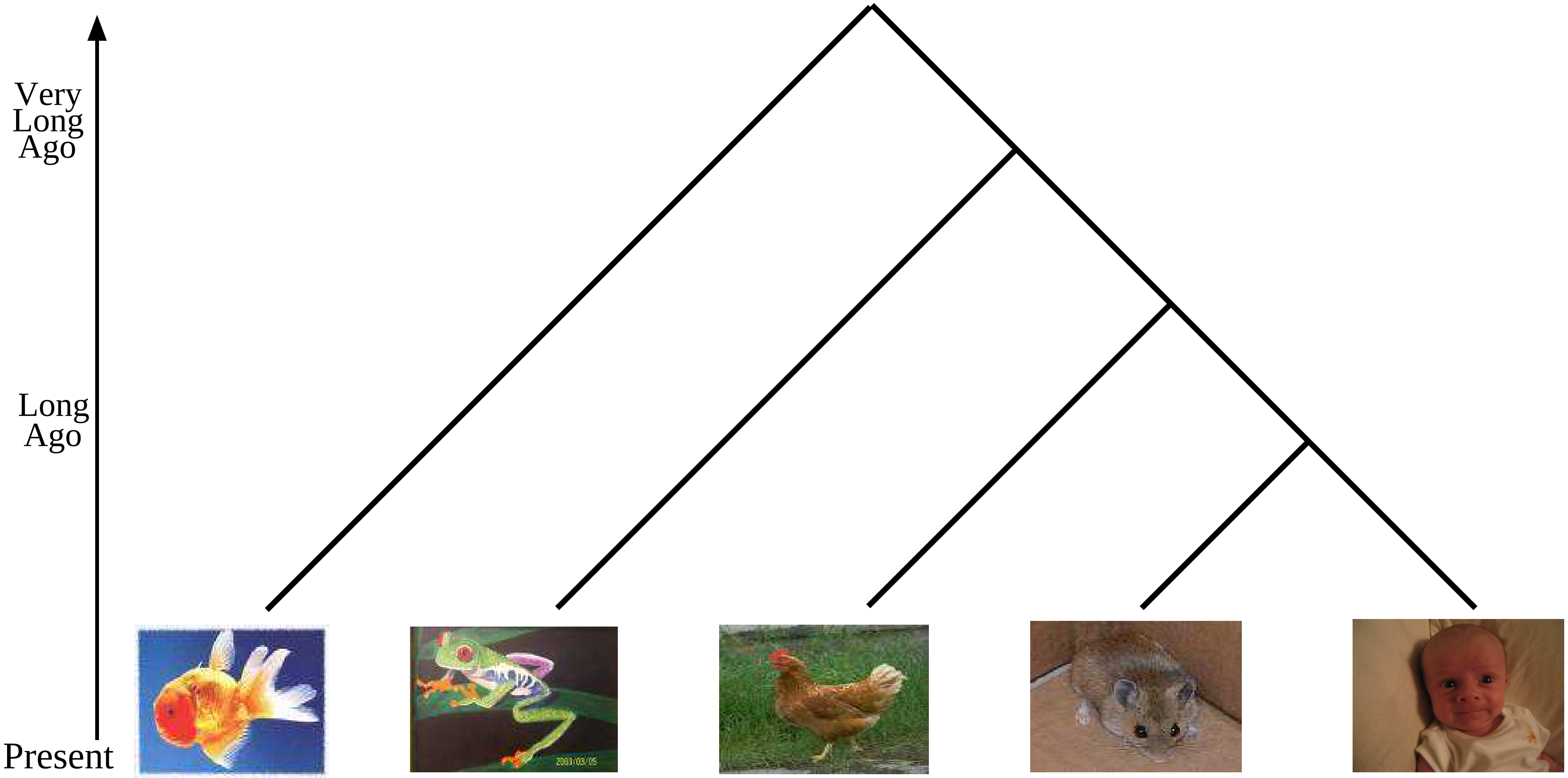
Phylogenetic Trees in ACL2

Warren A. Hunt Jr. and Serita M. Nelesen

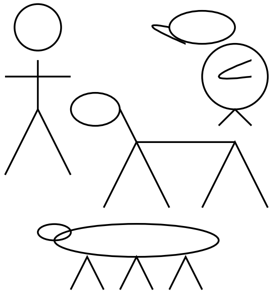
The University of Texas at Austin

Phylogenetic Trees

- Representation of the evolutionary relationship between species



From Organisms to Trees



A Set of Taxa

DNA Sequencing

```
Ape: ACCGTAGCTT
Bear: ATAGTAACT
Dog: CCGTATTT
Emu: CGCATAGC
Frog: CCTAAAC
Goat: GTAATAGAAC
```

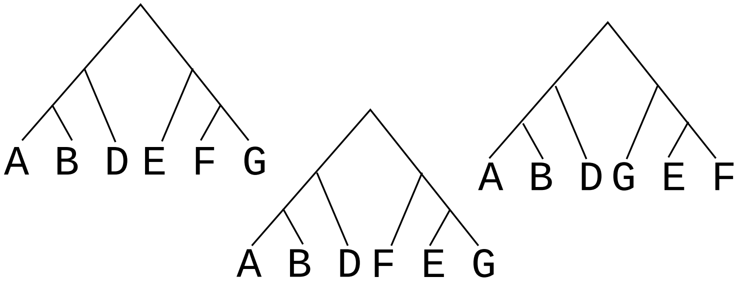
Unaligned Sequences

Multiple Sequence Alignment

```
Ape : ACCGTAGCTT
Bear: ATAGTAACT-
Dog : -CCGTA-TTT
Emu : CGCATAGC--
Frog: C-C-TA-AAC
Goat: GTAATAGAAC
```

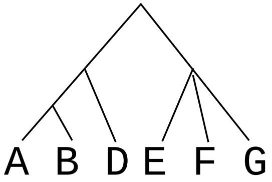
Aligned Sequences

Maximum Parsimony Search



Set of Optimal Trees

Consensus Analysis



Consensus Tree

Lots and lots of trees

- Number of possible trees grows exponentially with the number of leaves in the tree
- Two main methods used to determine the correct tree
  - A heuristic search through tree space
  - A Bayesian estimation of phylogeny using Markov chain Monte Carlo
- Both of these methods may produce hundreds, or thousands of trees which are then the input to further processing

Lots and lots of trees

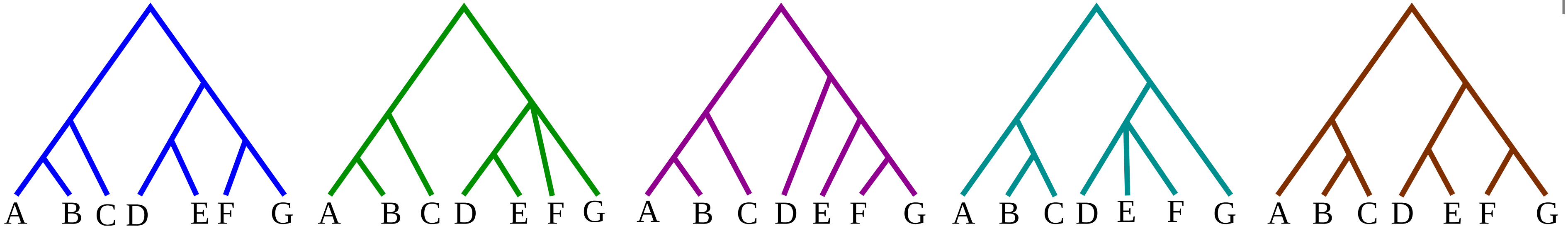
- Number of possible trees grows exponentially with the number of leaves in the tree
- Two main methods used to determine the correct tree
  - A heuristic search through tree space
  - A Bayesian estimation of phylogeny using Markov chain Monte Carlo
- Both of these methods may produce hundreds, or thousands of trees which are then the input to further processing

Need a system to store these trees efficiently, and perform post-tree analysis.

Why Use ACL2?

- Standard answer: Accuracy
  - Explicit specification of input and output for all functions together with proof that the specification is met within the code (guards)
  - Two representations of trees, with proof that we can accurately move from one representation to the other and back
- Additional answers: Storage space and performance speed
  - Hash-consing gives greatly reduced storage space
  - Memoization gives improved performance speed
- Overall: Medical systems of the future

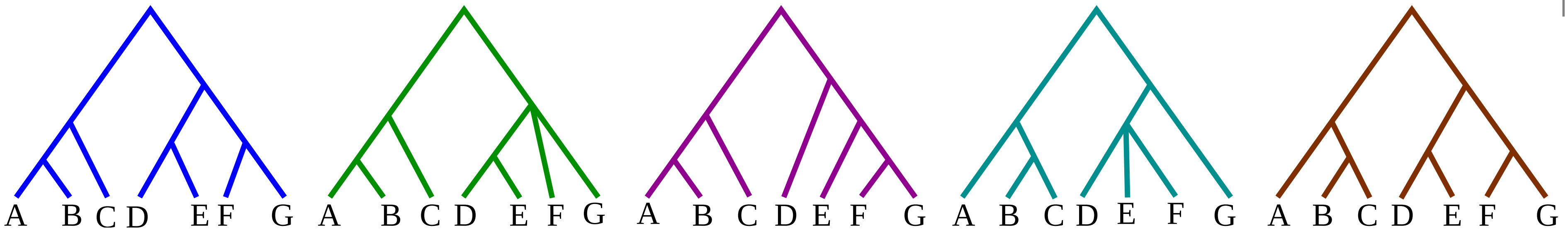
Representation



TASPI High-Level Representation:

```
(((A B) C) ((D E) (F G)))
(((A B) C) ((D E) F G))
(((A B) C) (D (E (F G))))
((A (B C)) ((D E F) G))
((A (B C)) ((D E) (F G)))
```

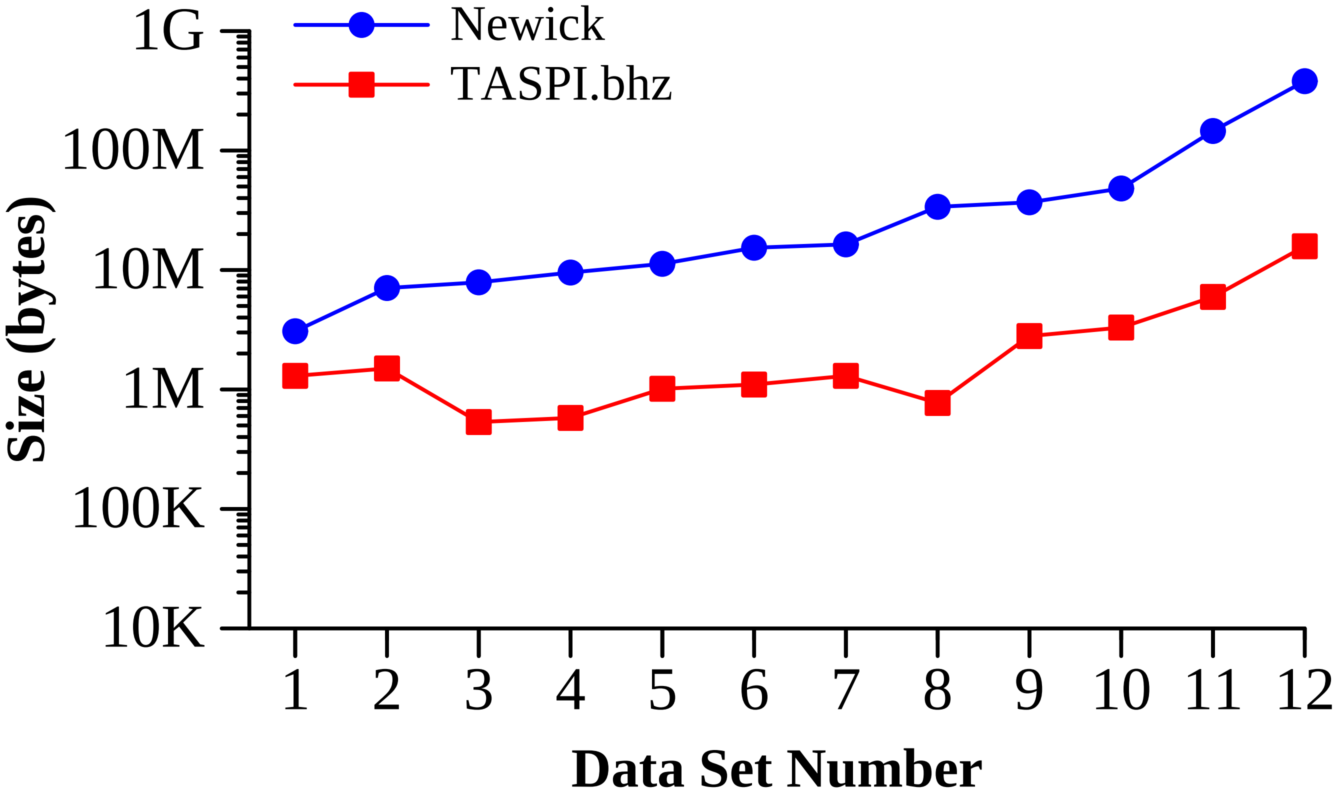
Representation



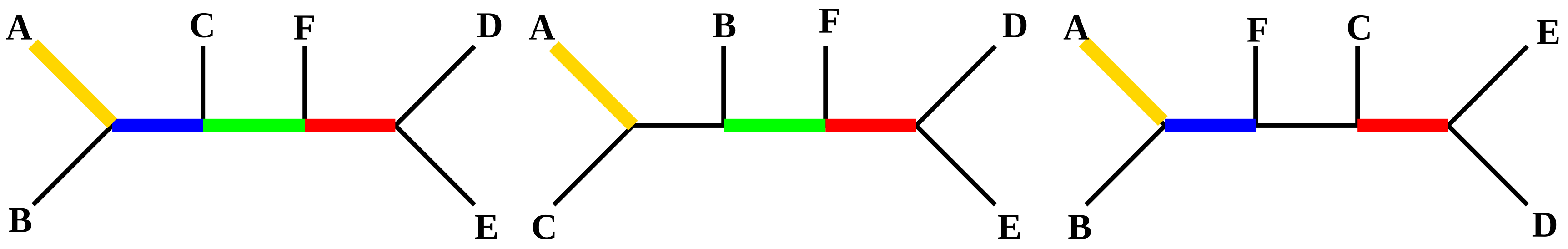
TASPI Low-Level Representation:

```
((#1=((A B) C) #5=(#6=(D E) #9=(F G)))
(#1#(#6# F G))
(#1#(D (E #9#)))
(#12=(A (B C)) ((D E F) G))
(#12##5#))
```

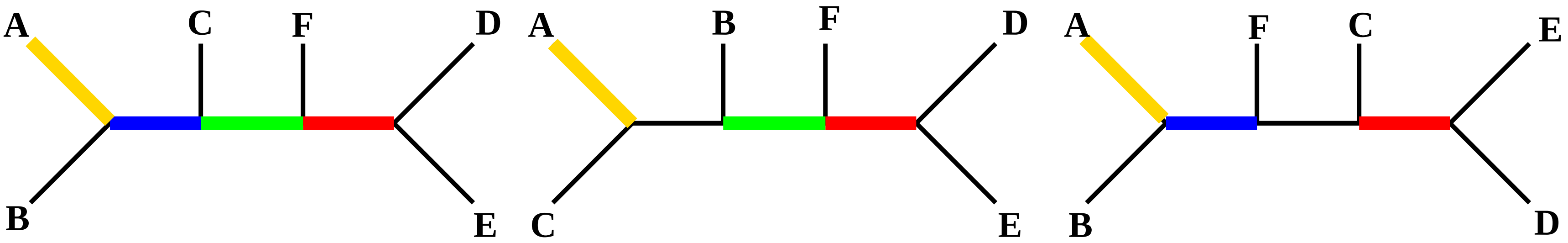
Reduced Storage Space



Bipartition Representation



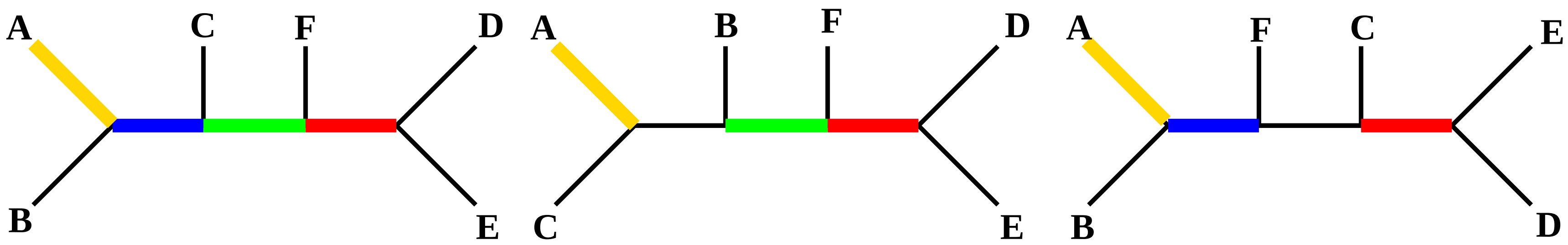
Bipartition Representation



Parenthetical Notation:

(A B (C ((D E) F)))   (A (B ((D E) F)) C)   (A B ((C (D E)) F))

Bipartition Representation



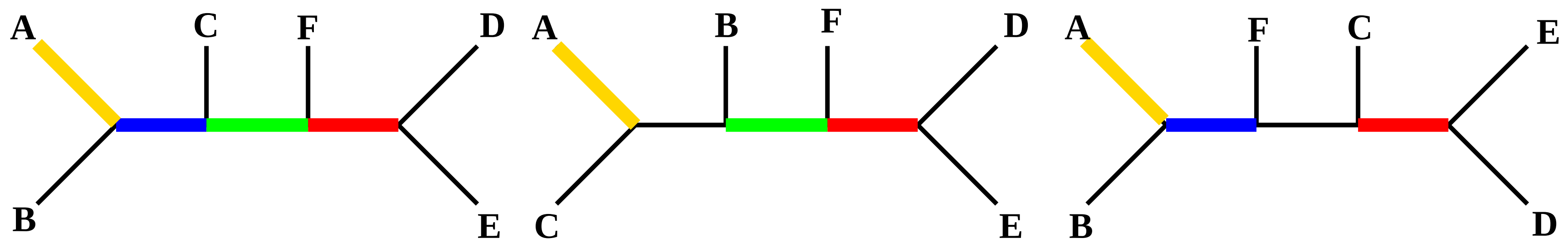
Parenthetical Notation:

(A B (C ((D E) F)))   (A (B ((D E) F)) C)   (A B ((C (D E)) F))

Bipartition Representation:

AB | CDEF	AC | BDEF	AB | CDEF
ABC | DEF	ABC | DEF	ABF | CDE
ABCF | DE	ABCF | DE	ABCF | DE

Bipartition Representation



Parenthetical Notation:

(A B (C ((D E) F)))

(A (B ((D E) F)) C)

(A B ((C (D E)) F))

Bipartition Representation:

AB | CDEF
ABC | DEF
ABCF | DE

AC | BDEF
ABC | DEF
ABCF | DE

AB | CDEF
ABF | CDE
ABCF | DE

Our Bipartitions:

(A B C D E F)
(C D E F)
(D E F)
(D E)

(A B C D E F)
(B D E F)
(D E F)
(D E)

(A B C D E F)
(C D E F)
(C D E)
(D E)

Relationship of Representations

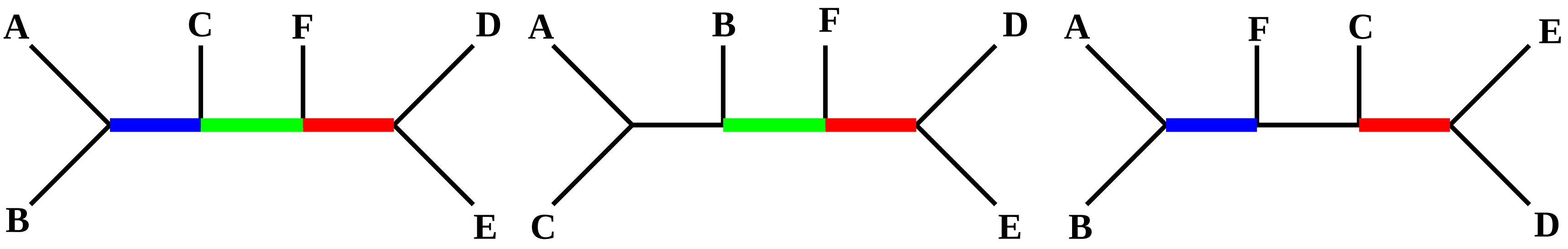
```
(defthm paren-partition-paren
  (implies (and <properties of input tree>
                <properties of ordering>
                <properties of tree and ordering>)
           (equal (tree-from-fringes (get-fringes tree
                                                  ordering)
                                     ordering)

                 tree)))
```

Strict and Majority Consensus

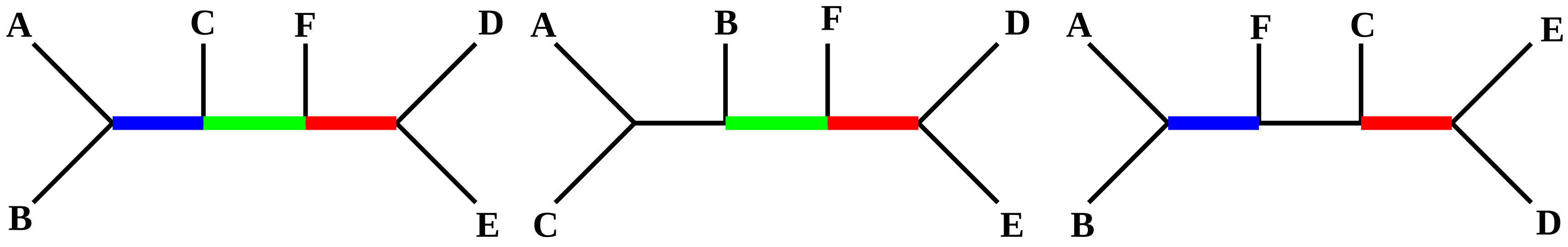
- Strict consensus : Any branch that appears in every input tree is in the consensus tree

- Majority consensus : Any branch that appears in more than half of the input trees is in the consensus tree

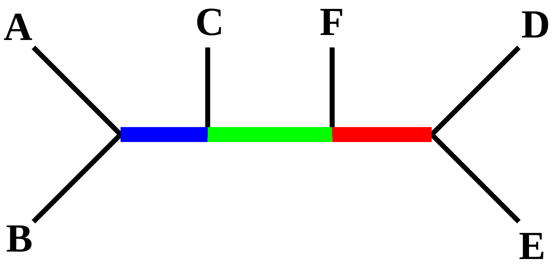
Example



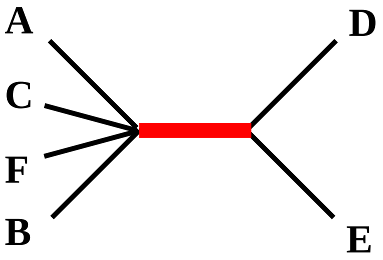
Example



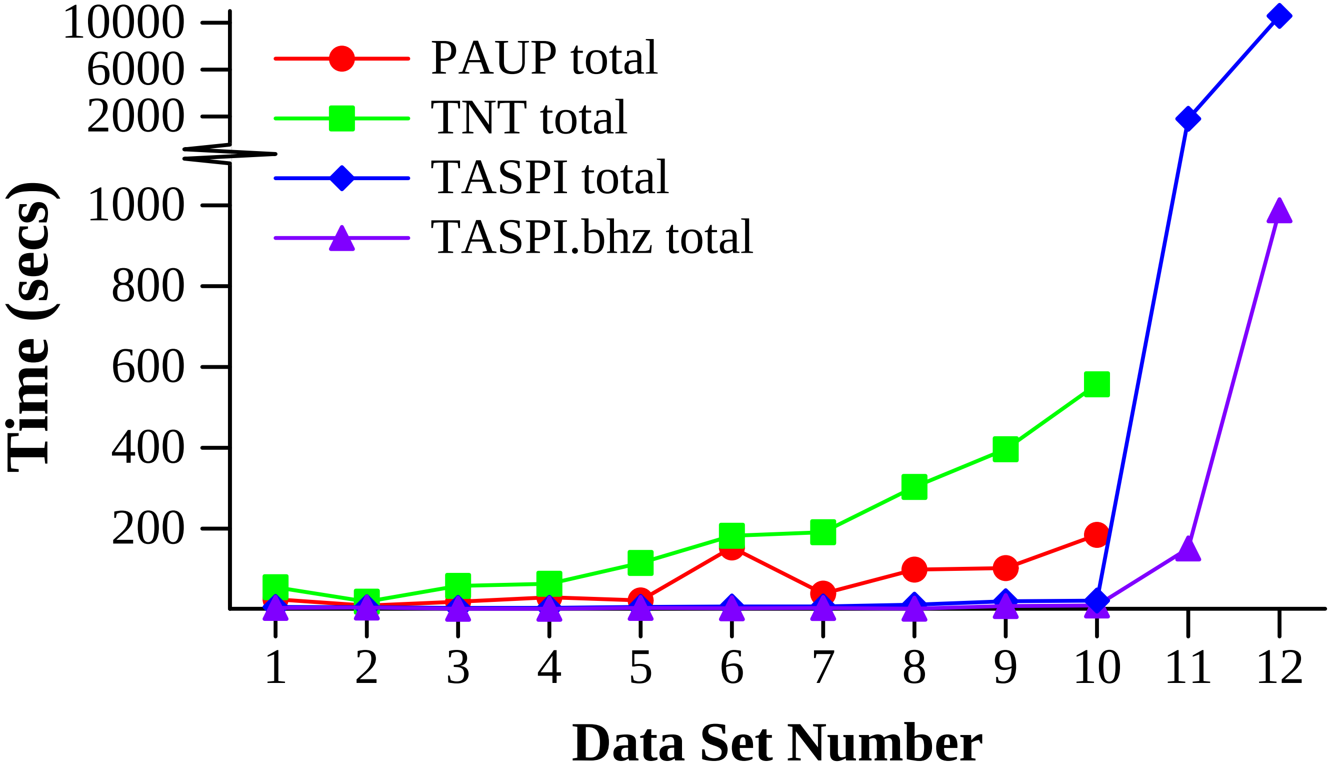
Majority



Strict



Improved Consensus Performance



Conclusion and Future Work

- TASPI provides accuracy guarantees, while providing state of the art performance in terms of size and speed
- TASPI is being extended to perform further post-tree analyses, as well as database operations

Questions?