

# Combinatorial Problems in Studies of Genetic Variations: Haplotyping and Transcript Analysis

February 3, 2010

Student: Yuri Pirola  
Supervisor: Prof. Paola Bonizzoni  
Tutor: Prof. Lucia Pomello

# Outline

- 1 Aims and Motivations
- 2 Results
  - Haplotype Inference
    - Pure Parsimony Xor Haplotyping
    - Haplotyping on Pedigrees
  - Transcript Analysis
    - Gene Structure Prediction
- 3 Conclusions

# Aims and Motivations

Studies of genetic variations are one of the most important task in the post-genomic era.

Large international projects: HapMap, ENCODE, 1000 genomes ...

But...

- Lot of data are needed
- Cost/technological reasons limit data availability

## Aim

Analysis and design of combinatorial methods that enable large-scale studies of genetic variations.

# Original Contributions

## *Haplotype Inference:*

- Exact and approximate algorithms for two haplotyping problems:
  - Pure Parsimony Xor Haplotyping (PPXH)
  - Haplotyping on Pedigrees with Mutations and Recombinations (MEHC)

## *Transcript Analysis:*

- Efficient algorithm which exploits redundancy to perform gene structure prediction

# Haplotype Inference

## Haplotype Inference Problem

For each individual in a population, distinguish the genome inherited from each parent accordingly to a reference genetic model.

- Well-known problem, studied under different assumptions.  
i.e. (Gusfield, CPM, '03), (Karp, ICALP, '04), (Shamir, IEEE/ACM TCBB, '08)
- Pure Parsimony Xor Haplotyping (PPXH)
- Minimum Events Haplotype Configuration (MEHC)

# Pure Parsimony Xor Haplotyping

*Pure Parsimony Xor Haplotyping:*

pure parsimony

(Gusfield, CPM, '03), (Lancia *et al.*, INFORMS J. Comput., '04)

+

xor-genotype

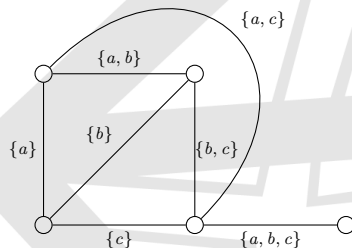
(Shamir *et al.*, IEEE/ACM Trans. Comput. Biology Bioinform., '08)

→ new combinatorial problem!

# Pure Parsimony Xor Haplotyping

## *Pure Parsimony Xor Haplotyping*

Characterization of solutions as graphs  $\rightarrow$  Xor graph



A Xor-Graph

# PPXH - Exact Algorithms

PPXH is *fixed parameter tractable*

- $O(2^{k^2} nm)$  time algorithm  
parameter  $k$  = size of a optimal solution  
 $n$  = population size,  $m$  = genotype length

Polynomial-time algorithms for specific (and motivated) restrictions:

- PPXH(\*,2)
- PPXH(2,\*)

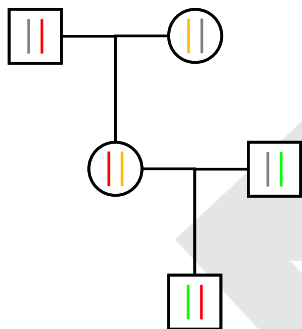
# PPXH - Heuristic Algorithm

## Heuristic Algorithm:

- Based on Xor-graph reconstruction
- Efficient:  $O(\alpha(n, m)n^3m)$  time complexity
- Experimental validation on various kinds of instances
- Experimental observations: **performs well**
  - approximation factor  $\leq 1.57$  (often close to 1)
  - time  $\leq 1h$  (on big instances)

# Haplotyping on Pedigrees

Pedigree  $\rightarrow$  parental relationships

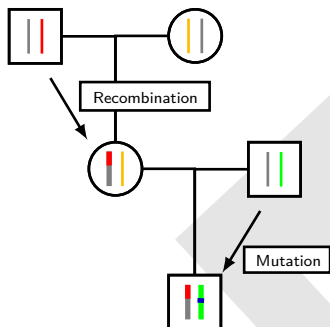


Polynomial if “pure”  
Mendelian Inheritance

(Xiao *et al.*, SIAM J. Comput., '09)

# Haplotyping on Pedigrees

Pedigree  $\rightarrow$  parental relationships



Polynomial if “pure”  
Mendelian Inheritance

(Xiao *et al.*, SIAM J. Comput., '09)

Intractable if we admit  
**genetic variation events**

(Liu *et al.*, TCS, '07)

# Minimum Events Haplotype Configuration

		Recombinations	
		NO	YES
Mutations	NO	Polynomial (SIAM J. Comput., '09)	<i>APX-hard</i> (TCS, '07) Randomized algorithm (ESA, '09) ILP formulation (JCB, '05)
	YES	<i>NP-hard</i> Exponential algorithm (CPM, '09)	

# Minimum Events Haplotype Configuration

		Recombinations	
		NO	YES
Mutations	NO	Polynomial (SIAM J. Comput., '09)	<i>APX-hard</i> (TCS, '07) Randomized algorithm (ESA, '09) ILP formulation (JCB, '05)
	YES	<i>NP-hard</i> Exponential algorithm (CPM, '09) <b>APX-hardness</b>	<b>NP-hardness</b> <b>Heuristic</b>

Original contributions

# MEHC - Heuristic Algorithm

## Minimum Events Haplotype Configuration (MEHC):

- connected (via L-reduction) to a well-known Information Theory problem (DECODING OF LINEAR CODES)
- Basic intuition:  
genetic **variation events** = **errors** in a codeword of a binary linear code

# MEHC - Heuristic Algorithm

Heuristic Algorithm (based on the L-reduction):

- Efficient:  $O(n^3 m^3 \cdot k)$  time complexity  
 $n$  = population size,  $m$  = genotype length,  $k$  = no. of events
- Experimentally validated: **extremely good performances**
  - 99.1% success rate (mutations and recombinations)
  - 100% success rate (only mutations on a real pedigree)
  - 99.8% success rate (only recombinations on a real ped.)
  - time per instance  $\leq 16m$

# Transcript Analysis

## Gene Structure Prediction

Locating coding and non-coding regions of DNA

How?

- **Transcripts** = small fragments of coding regions  
→ alignment of transcripts reveals gene structure

But...

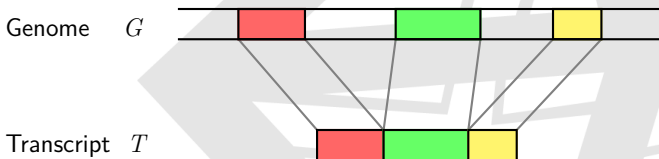
- sequencing errors
- repeated sequences (pseudogenes) and overlapping genes
- data size (genome and transcript libraries)

# Transcript Sequence Factorization

## Sequence Factorization Problem

Given two sequences  $T$  and  $G$ , partition  $T$  into a list of factors such that they occur in  $G$  in the same order.

Example:



Different factorizations can exist!

# Sequence Factorization Problem

Factorizations  $\rightarrow$  obtained from maximal *embeddings*  
(= sequences of common substrings)

New algorithm:

- Idea: find all maximal embeddings
- *CMEG*: compact implicit graph representation of all maximal embeddings
- Efficient *CMEG* construction:  $O(|G| + |T| + k) + O(k^2)$   
 $k = \text{no. of maximal common substrings}$
- Factorizations obtained by a simple visit of the *CMEG*

# Sequence Factorization Problem

How to choose the “right” factorization?

- Idea: exploiting the redundancy of the libraries of transcripts

→ definition of a new optimization problem!

## Factorization Agreement Problem

Given all the factorizations of a set  $S$  of sequences w.r.t. a sequence  $G$ , choose the minimum cardinality set  $F$  of factors of  $G$  such that each sequence of  $S$  can be factorized by using only factors that belong to  $F$ .

# Factorization Agreement Problem

## Results:

- NP-hard (by reduction from `MINSETCOVER`)
- Algorithm:
  - Size-reduction algorithm
  - Enumeration on the reduced instance
- Experimental Evaluation:
  - on 350 genes of Human Chromosome 22
  - fast and quite accurate even w/o advanced biol. criteria (compared with `ASPic` (Castrignanò *et al.*, NAR, '06))

# Conclusions

## *Haplotype Inference:*

- haplotyping under two different models, PPXH and MEHC
- coping computational intractability using different techniques
  - restrictions, FPT, heuristics, ...

## *Transcript Analysis:*

- algorithm to find alternative factorizations
- gene structure prediction via factorization agreement

# Publications

“Pure Parsimony Xor Haplotyping”

with Bonizzoni, Della Vedova, Dondi, and Rizzi.

In *Proceedings ISBRA 2009*, 186–197, 2009.

and *IEEE/ACM Trans. on Comput. Biology and Bioinformatics*, 2010.

“Detecting Alternative Gene Structures from Spliced ESTs: A Computational Approach”

with Bonizzoni, Mauri, Pesole, Picardi, and Rizzi.

In *J. of Computational Biology*, 16(1):43–66, 2009.

“Minimum Factorization Agreement of Spliced ESTs”

with Bonizzoni, Della Vedova, Dondi, and Rizzi.

In *Proceedings WABI 2009*, 1–12, 2009.

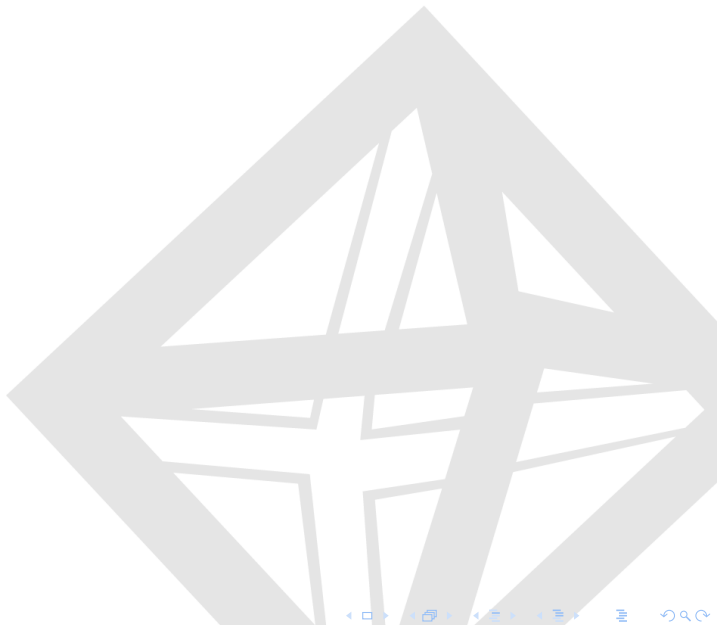
In preparation:

“Haplotype Inference in Pedigrees with Recombinations and Mutations”

with Tao Jiang.

“Faster Spliced Alignment via Maximal Pairings”

with Bonizzoni, Della Vedova, and Rizzi.



# Additional Content

- 4 Pure Parsimony Xor Haplotyping
  - Heuristic Algorithm
  - Experimental Results
  
- 5 Minimum Events Haplotype Configuration
  - Experimental Results - MEHC
  - Experimental Results - MRHC
  
- 6 Factorization Agreement
  - Example

# Heuristic algorithm

## Algorithm **PPXH**( $X$ )

- 1 Find a collection  $C$  of zero-sum subsets of  $X$
- 2  $C' \leftarrow \emptyset$
- 3 **While**  $C \neq \emptyset$  **do**
  - pick a random zero-sum subset  $c$  from  $C$
  - **if**  $C' \cup \{c\}$  admits Graph Realization  $\mathcal{G}'$   
**then** continue  
**else** remove from  $X$  the genotypes which label the last Graph Realization  $\mathcal{G}'$ , and start from step 2
- 4 Terminate when  $X = \emptyset$

# Heuristic algorithm - Example

$X$	$a$	$b$	$c$
$x_1$	1	0	0
$x_2$	0	1	0
$x_3$	0	0	1
$x_4$	1	1	0
$x_5$	0	1	1
$x_6$	1	0	1
$x_7$	1	1	1

# Heuristic algorithm - Example

$X$	$a$	$b$	$c$
$x_1$	1	0	0
$x_2$	0	1	0
$x_3$	0	0	1
$x_4$	1	1	0
$x_5$	0	1	1
$x_6$	1	0	1
$x_7$	1	1	1

Zero sum subsets

$x_1, x_2, x_4$

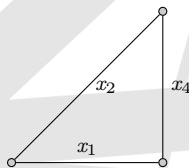
$x_2, x_3, x_5$

$x_1, x_3, x_6$

$x_1, x_2, x_3, x_7$

# Heuristic algorithm - Example

$X$	$a$	$b$	$c$
$x_1$	1	0	0
$x_2$	0	1	0
$x_3$	0	0	1
$x_4$	1	1	0
$x_5$	0	1	1
$x_6$	1	0	1
$x_7$	1	1	1



Zero sum subsets

$x_1, x_2, x_4$

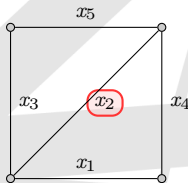
$x_2, x_3, x_5$

$x_1, x_3, x_6$

$x_1, x_2, x_3, x_7$

# Heuristic algorithm - Example

$X$	$a$	$b$	$c$
$x_1$	1	0	0
$x_2$	0	1	0
$x_3$	0	0	1
$x_4$	1	1	0
$x_5$	0	1	1
$x_6$	1	0	1
$x_7$	1	1	1



Zero sum subsets

$x_1, x_2, x_4$

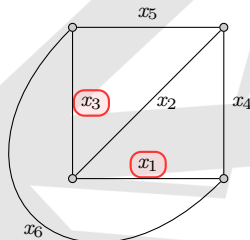
$x_2, x_3, x_5$

$x_1, x_3, x_6$

$x_1, x_2, x_3, x_7$

# Heuristic algorithm - Example

$X$	$a$	$b$	$c$
$x_1$	1	0	0
$x_2$	0	1	0
$x_3$	0	0	1
$x_4$	1	1	0
$x_5$	0	1	1
$x_6$	1	0	1
$x_7$	1	1	1



Zero sum subsets

$x_1, x_2, x_4$

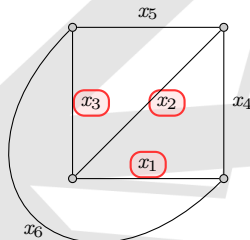
$x_2, x_3, x_5$

$x_1, x_3, x_6$

$x_1, x_2, x_3, x_7$

# Heuristic algorithm - Example

$X$	$a$	$b$	$c$
$x_1$	1	0	0
$x_2$	0	1	0
$x_3$	0	0	1
$x_4$	1	1	0
$x_5$	0	1	1
$x_6$	1	0	1
$x_7$	1	1	1



Zero sum subsets

$x_1, x_2, x_4$

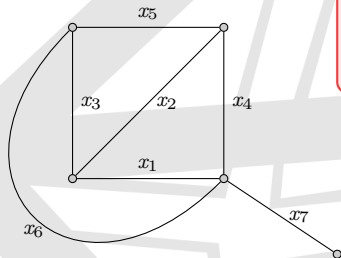
$x_2, x_3, x_5$

$x_1, x_3, x_6$

$x_1, x_2, x_3, x_7$

# Heuristic algorithm - Example

$X$	$a$	$b$	$c$
$x_1$	1	0	0
$x_2$	0	1	0
$x_3$	0	0	1
$x_4$	1	1	0
$x_5$	0	1	1
$x_6$	1	0	1
$x_7$	1	1	1



Zero sum subsets

$x_1, x_2, x_4$

$x_2, x_3, x_5$

$x_1, x_3, x_6$

$x_1, x_2, x_3, x_7$

# PPXH - Experimental results (Synthetic)

no. of genot.	no. of haplot.	no. of char.	avg. result	avg. ratio	no. of genot.	no. of haplot.	no. of char.	avg. result	avg. ratio
100	50	50	50	1	300	86	86	87	1.01
100	50	33	79.2	<b>1.58</b>	300	86	100	86	1
100	50	66	50	1	300	86	200	86	1
100	33	50	33	1	300	100	86	131.2	1.31
100	33	33	33.7	1.02	300	100	100	100.1	1
100	33	66	33	1	300	100	200	100	1
100	66	50	69.7	1.05	300	200	86	283	1.41
100	66	33	87.2	1.32	300	200	100	282.4	1.41
100	66	66	63	0.95	300	200	200	191	0.95
200	70	70	70.4	1	400	100	100	100.4	1
200	70	66	74.2	1.06	400	100	133	100	1
200	70	133	70	1	400	100	266	100	1
200	66	70	66	1	400	133	100	193.7	1.45
200	66	66	66	1	400	133	133	133	1
200	66	133	66	1	400	133	266	133	1
200	133	70	186.4	1.4	400	266	100	383.4	1.44
200	133	66	187.2	1.4	400	266	133	380.7	1.43
200	133	133	126	0.94	400	266	266	250	0.93

# MEHC - Experimental results (Synthetic)

Variable population size ( $n$ )

Population size $n =$	Tree pedigrees			General pedigrees			Mean
	40	60	100	40	60	100	
Avg. no. of heterozygous loci	813	1206	2001	796	1194	1988	1333
Avg. no. of generated events	22.0	30.4	55.2	25.5	35.8	63.6	38.7
Avg. no. of predicted events	21.3	29.5	53.2	24.5	34.9	61.8	37.5
Avg. precision	0.787	0.744	0.768	0.778	0.812	0.809	0.783
Avg. phase error	0.027	0.029	0.028	0.022	0.024	0.024	0.026
Avg. approximation ratio	0.968	0.975	0.965	0.963	0.975	0.972	0.970
Avg. time (s)	36	73	265	62	118	460	169

Fixed parameters:

genotype length  $m = 40$ , recombination rate  $\theta_r = 0.02$ , and mutation rate  $\mu_r = 0.004$

# MEHC - Experimental results (Synthetic)

Variable genotype length ( $m$ )

Genotype length $m =$	Tree pedigrees			General pedigrees			Mean
	40	60	100	40	60	100	
Avg. no. of heterozygous loci	806	1207	2009	800	1196	2032	1342
Avg. no. of generated events	24.0	34.7	53.2	26.4	38.0	61.0	39.6
Avg. no. of predicted events	23.1	33.0	51.2	25.7	36.9	59.6	38.3
Avg. precision	0.732	0.683	0.750	0.797	0.819	0.804	0.764
Avg. phase error	0.035	0.057	0.042	0.026	0.026	0.044	0.039
Avg. approximation ratio	0.964	0.956	0.964	0.975	0.972	0.976	0.968
Avg. time (s)	41	95	247	76	148	485	182

Fixed parameters:

population size  $n = 40$ , recombination rate  $\theta_r = 0.02$ , and mutation rate  $\mu_r = 0.004$

# MEHC - Experimental results (Synthetic)

Variable mutation and recombination rates ( $\theta_r$  and  $\mu_r$ )

	Tree pedigrees			General pedigrees			Mean
	0.02	0.04	0.10	0.02	0.04	0.10	
Recombination prob. $\theta_r =$	0.02	0.04	0.10	0.02	0.04	0.10	
Mutation probability $\mu_r =$	0.004	0.01	0.02	0.004	0.01	0.02	
Avg. no. of heterozygous loci	798	807	799	804	804	796	801
Avg. no. of generated events	24.5	48.8	111.5	24.9	48.9	121.8	63.4
Avg. no. of predicted events	23.8	45.7	94.8	24.0	45.8	105.3	56.6
Avg. precision	0.756	0.707	0.556	0.784	0.746	0.555	0.684
Avg. phase error	0.035	0.061	0.114	0.020	0.053	0.099	0.064
Avg. approximation ratio	0.973	0.937	0.848	0.963	0.939	0.866	0.921
Avg. time (s)	45	74	164	63	86	248	113

Fixed parameters:

population size  $n = 40$ , genotype length  $m = 40$

# MRHC - Experimental results

Comparison with PedPhase (ILP-based, exact, J. Comp. Biol., 2005)

Genotype length	No. of generated recombinations	Avg. recombinations		Avg. time (s)		Success rate (%)
		PedPhase	Heuristic	PedPhase	Heuristic	
50	10	8.87	8.87	6.41	1.21	100
60	10	8.82	8.82	7.69	1.47	100
80	15	13.41	13.41	16.46	3.31	100
80	20	18.02	18.04	17.18	4.2	99
90	20	18.07	18.07	23.61	5.22	100
95	15	13.48	13.48	22.44	4.61	100
Mean		13.45	13.45	15.63	3.34	99.8

Fixed parameters:

real pedigree with 52 members, 100 random genotype assignments per row

# Factorization Agreement - Example

