BIOINFORMATICS ORIGINAL PAPER Vol. 26 no. 20 2010, pages 2509–2516
doi:10.1093/bioinformatics/btq465

Genome analysis Advance Access publication August 24, 2010

# DRIMM-Synteny: decomposing genomes into evolutionary conserved segments

Son K. Pham* and Pavel A. Pevzner

Department of Computer Science and Engineering, University of California San Diego, La Jolla, California, USA

Associate Editor: Alex Bateman

**ABSTRACT**

**Motivation:** The rapidly increasing set of sequenced genomes highlights the importance of identifying the synteny blocks in multiple and/or highly duplicated genomes. Most synteny block reconstruction algorithms use genes shared over *all* genomes to construct the synteny blocks for multiple genomes. However, the number of genes shared among all genomes quickly decreases with the increase in the number of genomes.

**Results:** We propose the Duplications and Rearrangements In Multiple Mammals (DRIMM)-Synteny algorithm to address this bottleneck and apply it to analyzing genomic architectures of yeast, plant and mammalian genomes. We further combine synteny block generation with rearrangement analysis to reconstruct the ancestral preduplicated yeast genome.

**Contact:** kspham@cs.ucsd.edu

**Supplementary information:** Supplementary data are available at *Bioinformatics* online.

Received on June 21, 2010; revised on August 4, 2010; accepted on August 6, 2010

## 1 INTRODUCTION

The evidence in favor of the whole-genome duplication (WGD) in *Saccharomyces cerevisiae* was discovered by Wolfe and Shields (1997) but was heavily contested (e.g. see Koszul *et al.*, 2004). Dietrich *et al.*, 2004 and Kellis *et al.*, 2004 used preduplicated genomes of *Kluyveromyces waltii* and *Ashbya gossypii* to settle this controversy (see Martin *et al.*, 2007, for a recent study contesting the WGD in yeast). Starting from Wolfe and Shields (1997) and Seoighe and Wolfe (1999) all WGD studies essentially amounted to constructing synteny blocks of certain type [e.g. *sister blocks* in Seoighe and Wolfe, 1999 or *doubly conserved synteny (DCS)* blocks in Kellis *et al.*, 2004] and demonstrating that these blocks cover a large portion of the genome. Remarkably, there is still no general purpose tool that would automate such analysis and reduce various WGD studies to simply computing a 'duplicativity coverage' of the genome. For example, software from Kellis *et al.* (2004) is not applicable to finding the synteny blocks from Seoighe and Wolfe (1999) and vice versa. Indeed, most WGD studies (Aury *et al.*, 2006; Cui *et al.*, 2006; Dietrich *et al.*, 2004; Jaillon *et al.*, 2004; Kellis *et al.*, 2004; Machida *et al.*, 2005; Scannell *et al.*, 2007; Van de Peer, 2004) developed new software for WGD analysis instead of using some previously developed tools!

*To whom correspondence should be addressed.

We argue that the lack of tools for automated WGD analysis is the result of the lack of tools for *synteny block* identification in highly duplicated genomes. Many genomes have undergone extensive duplications followed by gene losses and rearrangements, making decoding of genomic architecture (synteny block reconstruction) in such genomes difficult. For example, duplications account for ~70% of the *Arabidopsis thaliana* genome (Blanc *et al.*, 2000) making synteny block reconstruction in this and other plant genomes challenging. Figure 1a shows a highly duplicated 'genome' *G* along with its decomposition into *overlapping* (left) and *non-overlapping* (right) synteny blocks. The non-overlapping decompositions are more desirable since they are required for the follow-up rearrangement and duplication studies (e.g. the existing genome rearrangement algorithms are unable to analyze overlapping decompositions). However, constructing non-overlapping decompositions is more difficult than constructing overlapping decompositions. While it may appear that one can simply subpartition the overlapping blocks into the non-overlapping ones, Jiang *et al.* (2007) and Peng *et al.* (2009) explained that this partitioning does not work for complex genomes.

Sankoff and Blanchette (1997) proposed the first algorithm for synteny block generation, which was aimed at comparative mapping data and did not take into account *micro-rearrangements*. The first algorithms for synteny block reconstruction in sequenced genomes [GRIMM-Synteny (Pevzner and Tesler, 2003) and Chains-and-Nets (Kent *et al.*, 2003)] were developed in 2003 when thousands of micro-rearrangements in mammalian genomes were discovered. These and many other synteny block generation algorithms (Bourque *et al.*, 2005; Brudno *et al.*, 2003; Calabrese *et al.*, 2003; Darling *et al.*, 2004a, b; Dewey *et al.*, 2006; Fujibuchi *et al.*, 2000; Ma *et al.*, 2006; Swidan *et al.*, 2006) proved to be adequate for small sets of genomes but did not address issues that stem from extensive duplications and deletions. Most previous efforts to generate synteny blocks for highly duplicated genomes (Blanc *et al.*, 2003; Bowers *et al.*, 2003; Haas *et al.*, 2004; Hampson *et al.*, 2005; Kellis *et al.*, 2004; Simillion *et al.*, 2008; Soderlund *et al.*, 2006; Vandepoele *et al.*, 2002) generated overlapping rather than non-overlapping blocks. In contrast, some recently developed tools [e.g. Enredo tool (Paten *et al.*, 2008) used in Ensembl (Hubbard *et al.*, 2002)] aim to generate non-overlapping synteny blocks. The non-overlapping representation has advantages over the traditional pairwise (and overlapping) representation of duplications. Indeed, the pairwise representation (that dominated previous studies of human segmental duplications) left the question of finding ancestral *duplicons* in the human genome unanswered (Bailey *et al.*, 2001), while the non-overlapping representation constructed in Jiang *et al.* (2007)



 

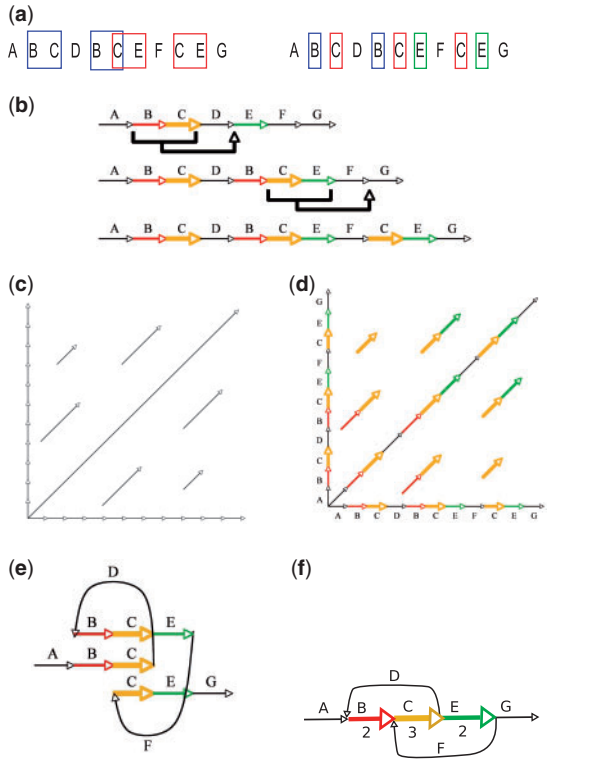


**Fig. 1.** (**a**) Decomposition of a 'genome' into overlapping (left) and non-overlapping (right) synteny blocks. A highly duplicated 'genome' (**b**) and its genomic dot-plot (**c**). (**d**) Since the diagonals in 2D representations overlap in 1D representation, one has to subpartition them into red, yellow and green subdiagonals to avoid overlaps. (**e**) Generating A-Bruijn graph. (**f**) A-Bruijn graph reveals synteny blocks B, E (each with two copies) and C with three copies. (While this represents anchors as directed edges, all other figures in this article represent anchors as vertices. We found that the vertex representation of anchors does not significantly affect our results, while significantly simplifying the presentation of DRIMM-Synteny.)

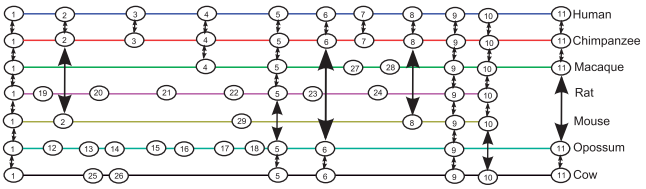


**Fig. 2.** A Human-Chimpanzee-Macaque-Rat-Mouse-Opossum-Cow synteny block (in Human chromosome 1) contains 29 genes with only 2 of them shared between all 7 species. While 17 of these 29 genes appear to be present only in a single genome (like gene 27 in macaque), most of these 17 genes have orthologs in other species (these orthologs are not shown since they are located within other synteny blocks in other species).

resolved it. Also, an overlapping representation can be easily obtained from a non-overlapping representation but not vice versa.

Figure 2 shows an Human–Chimpanzee–Macaque–Rat–Mouse–Opossum–Cow synteny block and illustrates the challenge of constructing synteny blocks in multiple genomes. As the number of analyzed genomes increases, the number of shared genes may substantially decrease. While this block contains 29 genes, only 2 of them are shared between all 7 species. The existing synteny block generation algorithms (such as GRIMM-Synteny) are likely to miss such a block with only two shared genes or discard it as statistically insignificant.

Peng *et al.* (2009) noticed that the problem of constructing non-overlapping synteny blocks is similar to the difficult problem of *de novo* repeat classification (Bao and Eddy, 2002). Pevzner *et al.* (2004) introduced the *A-Bruijn graph* approach to repeat classification, representing all repeats as a mosaic of non-overlapping subrepeats. Later, the A-Bruijn graphs were found to be useful in diverse applications such as multiple alignment (Raphael *et al.*, 2004), *de novo* protein sequencing (Bandeira *et al.*, 2008), analysis of segmental duplications (Jiang *et al.*, 2007) and next generation DNA sequencing (Butler *et al.*, 2008; Chaisson and Pevzner, 2008; Zerbino and Birney, 2008).

While diverse applications of A-Bruijn graphs use the same algorithmic idea, each application has unique features that need to be addressed for a new research domain. The original A-Bruijn graph approach (Pevzner *et al.*, 2004) involves some heuristics that may or may not work for a particular application. For example, the *bulge removal* heuristic was originally designed for fragment assembly of Sanger reads but turned out to work well in various tools for next-generation DNA sequencing (Butler *et al.*, 2008; Chaisson and Pevzner, 2008; Zerbino and Birney, 2008), mass spectrometry (Bandeira *et al.*, 2008) and synteny block reconstruction (Paten *et al.*, 2008; Peng *et al.*, 2009). Another important heuristic that is application specific is the *threading* procedure from Pevzner *et al.* (2004) that reconstructs how the genome traverses the transformed A-Bruijn graph. While threading was never problematic in sequencing applications, Peng *et al.* (2009) came to the conclusion that it is a major bottleneck in synteny block reconstruction and wrote: 'Optimizing the A-Bruijn graph approach for synteny block generation represents the next challenge in analyzing the genomic architectures.' Our article addresses this problem by devising the first A-Bruijn graph approach that does not require a threading step and substitutes it with an alternative *genome modification* step implemented in the Duplications and Rearrangements In Multiple Mammals (DRIMM)-Synteny software ( http://bix.ucsd.edu/projects/drimm/). We illustrate applications of DRIMM-Synteny to analyzing yeast, plant and mammalian genomes and further combine it with rearrangement analysis to reconstruct the ancestral preduplicated yeast genome (see section 5 of the Supplementary Material).

## 2 METHODS

*Preliminaries*: a typical synteny block generation algorithm takes as an input a set of *anchors* (e.g. local alignments or pairs of similar genes) between two genomes and constructs a set of synteny blocks that cover (without overlaps) most of each genome. As a result, each genome is represented as a shuffled sequence of the synteny blocks. For two genomes, most synteny blocks generation algorithms employ a 2D genomic dot-plot where two genomes are placed along the axes on the plane and their anchors are represented as dots (Supplementary Fig. S1a). These algorithms further decompose the dot-plot into long diagonal-like segments constituting 2D synteny blocks. The conventional (1D) synteny blocks for each genome can be obtained as projections of the 2D synteny blocks onto a corresponding axis (Supplementary Fig. S1b).

Figure 1b and c shows a highly duplicated 'genome' and its genomic dot-plot. The diagonals in Figure 1c are what conventional synteny block reconstruction methods would produce as synteny blocks from the genomic dot-plot of a genome against itself. Since these 2D blocks overlap along the sequence (in 1D), the duplication structure is unclear. Ideally, we would like to see diagonal segments that do not overlap along the sequence (Fig. 1d). The non-overlapping segments are revealed by the A-Bruijn graph (Fig. 1e and f) approach described in Pevzner *et al.* (2004).

Let $S=(s_1,s_2,\ldots,s_n)$ be a sequence of genes in a genome represented as an undirected path (Fig. 3a) and let $m$ be the number of *unique* genes in $S$ ($S$ may have repeated genes). While this article considers genes as anchors, DRIMM-Synteny is applicable to any anchors representing arbitrary regions of similarity. An *A-Bruijn graph* $AB(S)$ is obtained by 'gluing' identically labeled vertices of the path $S$ as shown in Figure 3c [see Pevzner *et al.* (2004) for the precise definition of 'gluing']. We remark that the A-Bruijn graphs are *Eulerian*, i.e. there exists a path in these graphs visiting every edge exactly once. The A-Bruijn graph can be viewed as both an undirected *multi-graph* (adjacent vertices can be connected by multiple edges) and a *weighted* graph with the multiplicity of an edge $(v,w)$ defined as the number of times genes $v$ and $w$ are consecutive in $S$.

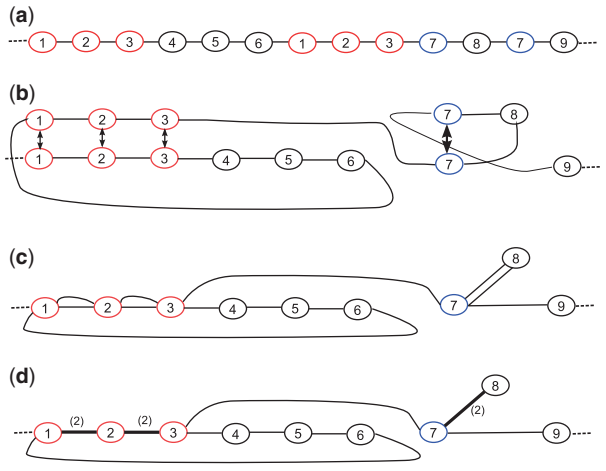


**Fig. 3.** (**a**) A genome $S=(1,2,3,4,5,6,1,2,3,7,8,7,9)$ with 13 genes (9 unique genes) represented as a path. (**b**) Constructing the A-Bruijn graph by gluing vertices with the same labels. (**c**) The A-Bruijn graph of genome $S$. (**d**) The weighted A-Bruijn graph with edge multiplicities shown.

A set of the perfectly repeated regions in $S$ corresponds to a path in the A-Bruijn graph (e.g. the path [1,2,3] in Fig. 3d). The perfectly repeated regions that do not share genes with other regions in $S$ correspond to *non-branching* paths (maximal paths in the graph satisfying the condition that all their internal vertices have only two neighboring vertices), with the multiplicities equal to the number of times these regions appear in the sequence $S$. In the case of the synteny blocks, however, small differences between multiple instances of the same synteny block generate short cycles in the A-Bruijn graphs, while the spurious similarities between different synteny blocks (called *microblocks*) break long non-branching paths into multiple shorter subpaths. Moreover, short palindromic regions within the conserved blocks generate the so-called *thorns* (like path [7,8,7] in Fig. 3c). These short cycles, microblocks and thorns hide the underlying synteny blocks in genomes and make the synteny block generation difficult.

*Synteny blocks in multiple and/or highly duplicated genomes*: from an algorithmic perspective, finding (i) synteny blocks between multiple genomes and (ii) synteny blocks within a single genome are similar problems since (i) can be reduced to (ii) by concatenating (with delimiters) the multiple genomes into a single genome. This illustrates the challenge one faces while reconstructing synteny blocks in *multiple* mammalian genomes that are traditionally viewed as an 'easy target' (compared with plant genomes) for synteny block analysis: while duplications account for <7% of mammalian genomes, the concatenation of mammalian genomes represents a highly duplicated virtual genome that rivals the complexity of plant genomic architectures. Bourque *et al.* (2004) faced this problem while constructing the human–mouse–rat synteny blocks. While their approach (based on anchors shared between *all* genomes) worked for a small number of genomes, it is unsustainable since the number of such anchors decreases with the increase in the number of genomes.

Given a set of chromosomes, one can concatenate them and construct the A-Bruijn graph of the resulting concatenation. Applying this procedure to genomes of *S.cerevisiae* (16 chromosomes with 5616 genes, 5057 are unique) and *K.waltii* (8 chromosomes with 5070 unique genes) results in a complex graph with 6240 vertices and 8976 edges. Figure 4b represents a subgraph of this A-Bruijn graph corresponding to a DCS block (Kellis *et al.*, 2004). This DCS is formed by a pair of regions:
… **1**, 4, **7**, **12**, **15**, 16, **17**, **19**, 21, …
… **1**, 3, 22, 5, **7**, 9, 23, **12**, 13, 24, 25, **15**, **17**, **19**, 26, …
in *S.cerevisiae* and a single region
… **1**, 2, 3, 4, 5, 6, **7**, 8, 9, 10, 11, **12**, 13, 14, **15**, 16, **17**, **19**, 20, …

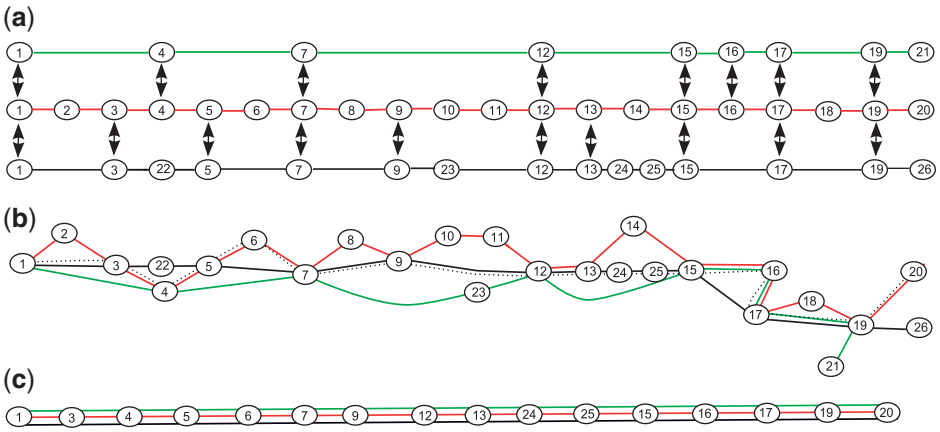


**Fig. 4.** (**a**) A DCB block in *K.waltii* and *S.cerevisiae* where one region in *K.waltii* genome (red) corresponds to two regions in *S.cerevisiae* (green and black). (**b**) An induced subgraph of the A-Bruijn graph corresponding to a DCS block (a) in the genomes of *S.cerevisiae* and *K.waltii* contains many short cycles. (**c**) The sequence modification algorithm reveals the synteny block as a non-branching path.

in *K.waltii* (six genes shared by all three regions are shown in bold). Figure 4b reveals many short cycles that 'hide' these three syntenic regions. Below we propose a *Sequence Modification* algorithm that transforms the original genome (with cryptic) synteny blocks into a slightly different genome with the well-defined synteny blocks. The key idea is to make small changes to the sequence $S$, so that its corresponding A-Bruijn graph is simplified. In contrast, the previous A-Bruijn graph approaches simplify the A-Bruijn graph $AB(S)$ without changing the sequence $S$ and thus faced a difficult challenge of threading $S$ through the simplified graph. The Sequence Modification algorithm transforms the subgraph in Figure 4b into a subgraph in Figure 4c and transforms each of the three (varying) instances of the DCS block into three (non-varying) instances …, **1**, 3, 22, 5, 6, **7**, 9, 23, **12**, 13, 24, 25, **15**, 16, **17**, 18, **19**,….

*Genome Threading Problem*: a cycle in a graph is *short* if it has fewer than *girth* edges, where *girth* is a parameter. Short cycles often aggregate into complex networks and 'hide' the underlying structure of the A-Bruijn graphs. To reveal this hidden structure, Pevzner *et al.* (2004) formulated the *Maximum Subgraph with Large Girth (MSLG)* problem, which aims to find the maximum weight subgraph of the A-Bruijn graph that does not contain short cycles. Pevzner *et al.* (2004) proposed constructing the *Maximum Spanning Tree (MST)* as the first step toward finding MSLG, followed by extending MST into an approximate solution [called the *simplified A-Bruijn graph* and denoted MSLG($S$)] of the MSLG problem, and finally, the genome *threading* procedure. We remark that while the multigraph $AB(S)$ is Eulerian and the genome sequence $S$ represents an Eulerian path in $AB(S)$, MSLG($S$) is typically non-Eulerian. The goal of threading is to find a *Chinese Postman* (Skiena, 1990) path in MSLG($S$) that 'mimics' $S$. While the threading heuristic from Pevzner *et al.* (2004) worked well for fragment assembly, Peng *et al.* (2009) commented that it deteriorates for synteny block construction when missing genes and micro-rearrangements are common.

Therefore, the key complication in the synteny block reconstruction is that, in difference from the (Eulerian) A-Bruijn graph, the simplified A-Bruijn graph is not Eulerian. Since the MSLG algorithm from Pevzner *et al.* (2004) breaks the Eulerian path into multiple segments, threading the original sequence through the simplified graph, in some cases, becomes impossible. This motivates the *Sequence Modification Problem* (*SMP*) defined below.

*SMP*: since the A-Bruijn graphs of real genomes have many short cycles (hiding synteny blocks), the goal of the synteny block reconstruction is to reveal the 'hidden' synteny blocks by removing these short cycles. In an A-Bruijn graph without short cycles, synteny blocks are defined as the non-branching paths in the graph with multiplicity larger than 1. The number of times a synteny block appears in the sequence is the multiplicity of the corresponding non-branching path.

Let $d(S,S')$ be the minimum number of edit operations (e.g. insertions/deletions/substitutions of letters or short substrings) to transform a string $S$ into a string $S'$. We define the SMP as follows:

SMP: given a string $S$ and a parameter *girth*, find a string $S'$ with minimum $d(S,S')$ among all strings such that $AB(S')$ has no cycles shorter than *girth*.

Since the complexity status of SMP is unknown, we propose a greedy algorithm that produces good results in practice. In brief, the algorithm finds short cycles in $AB(S)$ and further changes $S$ into $S'$ with the goal to *eliminate* short cycles from $AB(S')$. Before describing the strategy for eliminating the short cycles, we classify all cycles in the A-Bruijn graphs into *two-*, *one-way* and *composite* cycles.

A cycle $C$ in $AB(S)$ is *formed* by paths $P_1$ and $P_2$ ($P_1$ and $P_2$ are non-overlapping substrings of $S$) if the edge set of $C$ is the union of the edge sets of $P_1$ and $P_2$. A cycle $C$ is called a two-way cycle if it is formed by paths $P_1$ and $P_2$. For example, in Figure 5a, a two-way cycle on vertices (1, 2, 3) in the A-Bruijn graph of $S=(\ldots 1,\ 2,\ 3,\ 4,\ldots,1,\ 3,\ 4,\ldots)$, is formed by paths $P_1=[1,\ 2,\ 3]$ (consisting of two edges) and $P_2=[1,\ 3]$ (consisting of a single edge).

A cycle $C$ in $AB(S)$ is called a one-way cycle if it is formed by a single path (substring) $P$ of sequence $S$ (i.e. the edge sets of $C$ and $P$ are the same).

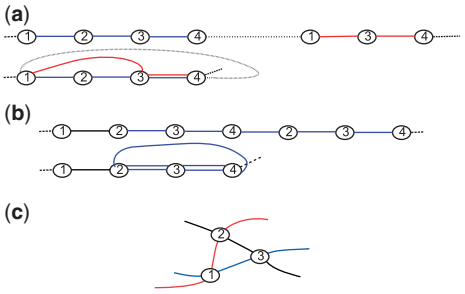


**Fig. 5.** (**a**) A two-way cycle (1, 2, 3) caused by a small difference between the syntenic regions [1, 2, 3, 4] and [1, 3, 4]. (**b**) A one-way cycle (2, 3, 4) formed by a tandem repeat [2, 3, 4, 2, 3, 4]. (**c**) A composite cycle formed by 3 paths: [1, 3], [3, 2] and [2, 1] that share some genes.

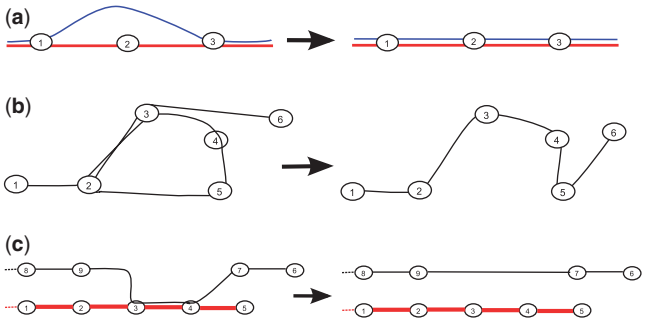


**Fig. 6.** (**a**) Detour defined by a two-way cycle (1, 2, 3) (that is formed by paths [1, 2, 3] and [1, 3]) eliminates the cycle. (**b**) Shortcut of a one-way cycle (2, 3, 4, 5) formed by a path [2, 3, 4, 5, 2, 3] eliminates the cycle. (**c**) Path splitting eliminates spurious similarities.

In Figure 5b, the tandem repeat [2, 3, 4, 2, 3, 4] corresponds to a one-way cycle (2, 3, 4). We also define composite cycles as cycles that are formed by more than two paths/substrings (Fig. 5c).

In practice, the cycles in the A-Bruijn graphs are typically classified in only one of three categories above. However, some cycles are classified into multiple categories, for example, a cycle can be both a one- and a two-way cycle (section 2 of the Supplementary Material).

*Cycle rerouting*: let $C$ be a two-way cycle formed by paths $P_1$ and $P_2$. The string $S$ may contain multiple instances of substrings $P_1$ and $P_2$, with the corresponding multiplicities $n_1 \leq n_2$. $(P_1,P_2)$-*transformation* of $S$ (called DETOUR) is a substitution of all instances of $P_1$ in $S$ by $P_2$. $(P_1,P_2)$-transformation has a simple interpretation: the Eulerian path switches from traversing $P_1$ to traversing $P_2$, thus eliminating an instance of a cycle $C$ from the A-Bruijn graph (Fig. 6a). We choose $n_1$ substitutions of $P_1$ by $P_2$ (rather than $n_2$ substitutions of $P_2$ by $P_1$) to minimize the number of segmental substitutions in the Sequence Modification algorithm.

Let $C$ be a one-way cycle formed by a path $P=(v_{in},\ldots,u,v_{in},\ldots,v_{out})$, where $v_{in}$ and $v_{out}$ are the first and the last vertices of $P$. *P-transformation* (called SHORTCUT) substitutes every instance of path $P$ by a shorter path $(v_{in},\ldots,u)$ (Fig. 6b).

The REROUTE procedure (Supplementary Fig. S4) iterates detours and shortcuts on a cycle $C$ until the cycle is eliminated or is neither a two- nor one-way cycle. DRIMM-Synteny does not have a specific subroutine that removes composite cycle. However, in most cases, composite cycles are removed by the cycle rerouting procedure (on different cycles) or the splitting procedure described below.

*Processing microblocks and thorns*: after REROUTE, the A-Bruijn graph may still be complex. Spurious similarities between different synteny blocks form *microblocks* (non-branching paths shorter than a threshold *pathLength*) and the palindrome-like substrings in the genomic sequences form *thorns* (like path [7, 8, 7] in Fig. 3c). Both microblocks and thorns break long synteny blocks into shorter blocks and need to be processed to avoid unnecessary synteny blocks miniaturization.

GRIMM-Synteny (Pevzner and Tesler, 2003) simply removes microblocks (defined as 'small' synteny blocks) and may occasionally 'destroy' biological synteny blocks formed by multiple microblocks (see Supplementary Fig. S5). DRIMM-Synteny instead *splits* blocks that share a microblock (Fig. 6c). The details of the splitting procedure are given in section 2 of the Supplementary Material.

The palindrome-like substrings in $S$ form non-branching paths called thorns. Long palindromes are valuable synteny blocks, while short ones form thorns that break long synteny blocks into shorter ones. We process short thorns (shorter than *thornLength*) by finding all short palindromes and removing the second halves of these palindromes. Similarly, tandem repeats are reported as synteny blocks (of multiplicity 2) if they form long cycles.

*Identification of syntenic regions: an alternative to genome threading*: DRIMM-Synteny (Fig. 7) is an approximation algorithm for the SMP that first finds a MST $T$ of the graph $AB(S)$ and iteratively analyzes all edges that are not present in $T$ (*outside edges*). We limit our attention to the outside edges forming short cycles, identify a shortest cycle containing an outside edge, and further change $S$ into $S'$ with the goal to *eliminate* this cycle from $AB(S')$. Application of DRIMM-Synteny to the graph in Figure 4b results in a simple graph in Figure 4c that reveals the DCS block. We remark that while any spanning tree (rather than MST) would work for detecting short cycles, DRIMM-Synteny selects MST since it proved to work well in other applications of A-Bruijn graphs. DRIMM-Synteny is fast in practice, taking less than a minute even for the largest dataset we analyzed (∼20 000 genes per each of seven mammalian genomes. See section 2 of the Supplementary Material for the running time analysis).

DRIMM-Synteny transforms the original sequence $S$ (genome) into a new sequence $S'$ with well-defined synteny blocks [each synteny block in $S'$ corresponds to a non-branching path in $AB(S')$]. The only remaining task is to identify the *positions* of all synteny blocks in the original sequence $S$. If we assume that each synteny block in the modified sequence $S'$ is painted with its own color, then the problem is to transform colors from $S'$ back to $S$. While the threading step from Pevzner *et al.* (2004) often results in poor-quality synteny block reconstruction (Peng *et al.*, 2009), our sequence modification approach bypasses the genome threading step as described in the Supplementary section 6.

DRIMM-SYNTENY (sequence $S$, cycle length threshold $girth$, path length threshold $pathLength$, thorn length threshold $thornLength$)

```
Construct the A-Bruijn graph AB(S)
Find the Maximum Spanning Tree MST(S) in AB(S)
for each edge e outside MST(S) (in increasing order of multiplicities)
    Identify the shortest one or two-way cycle C that contains edge e
    if (|C| ≤ girth)
        REROUTE(S, C)
Process palindromes in S that are shorter than thornLength
Process microblocks (nonbranching paths shorter than pathLength)
Return all non-branching paths in AB(S) as synteny blocks
```

**Fig. 7.** DRIMM-Synteny algorithm (the last *color propagation* step is not shown). See Supplementary section 2 for the details of the algorithm.

## 3 RESULTS

*Datasets*[1] *and parameters*: the yeast gene orders were extracted from Kellis *et al.* (2004) and Scannell *et al.* (2007). The mammalian gene orders were generated using MSOAR program (Fu *et al.*, 2007). The gene order of *A.thaliana* was extracted from Bowers *et al.* (2003).

Although every synteny block reconstruction algorithm is parameters independent, we are not aware of tools for automatic derivation of the optimal parameters. The parameters' choice for these tools (and DRIMM-Synteny) relies on an expert analysis (see Supplementary section 3 for parameter choice in DRIMM-Synteny). In this article, we use the default parameters ($girth = 20, pathLength = 3, thornLength = 3$) for all datasets.

*Synteny blocks in seven mammalian genomes*: to benchmark DRIMM-Synteny on multiple (but not highly duplicated) genomes, we analyzed seven mammalian genomes: human (H), chimpanzee (C), macaque (Q), rat (R), mouse (M), opossum (O) and cow (W). As the number of genomes increases, the number of genes that are shared between all genomes decreases and methods relying on the genes shared by all genomes (e.g. GRIMM-Synteny) deteriorate. Figure 2 shows a seven-way synteny block that would most likely be missed by such tools.

The concatenation of seven mammalian genomes results in a virtual genome with 144 149 genes (53 245 unique genes). The simplified A-Bruijn graph of this concatenation (with the default parameters) has 31 282 vertices and 35 773 edges. Substituting non-branching paths in this graph by single edges results in a graph on 2212 vertices and 3514 edges. DRIMM-Synteny still finds many synteny blocks with good coverage (∼70%) in this highly duplicated virtual genome (Supplementary Table S1a and Supplementary Files). Enredo (Paten *et al.*, 2008), an advanced synteny block generation tool used in Ensembl (Hubbard *et al.*, 2002), generated seven-way blocks with a significantly lower coverage (∼32%, Supplementary Table S1b).

To further compare Enredo (Paten *et al.*, 2008) and DRIMM-Synteny, we ran both program on the dataset initially containing only human and chimpanzee genomes where these tools generated nearly identical results. Then at each step, we added one more genome to the dataset, generated $k$-way synteny blocks ($k$ is the number of genomes), computed the genome coverage by these blocks and repeated the process for $k = 3, \ldots, 7$. Figure 8 shows that, as more genomes are added to the dataset, DRIMM-Synteny continues generating synteny blocks with high coverage (∼70% for seven-way blocks), while the seven-way synteny blocks generated by Enredo cover only ∼32% of the genome.

*Synteny blocks in K.waltii and S.cerevisiae*: the concatenation of *S.cerevisiae* (S) and *K.waltii* (K) results in a genome with 10686 genes (6240 unique genes). The simplified A-Bruijn graph of this concatenation has 5844 vertices and 6221 edges. Substituting non-branching paths in this graph by single edges results in a graph on 653 vertices and 997 edges. DRIMM-Synteny finds nearly all DCS blocks identified in Kellis *et al.* (2004) as well as 231 singly conserved synteny blocks (Table 1).

Since most studies of genomic architectures ignore very short synteny blocks, we delete all short synteny blocks (with fewer than $\Delta$ genes in each species) in an iterative fashion as described in Alekseyev and Pevzner (2009). Supplementary Figure S8 presents *S.cerevisiae* and *K.waltii* genomes in the alphabet of 151 large DCS synteny blocks (for $\Delta = 6$).

[1]See the Supplementary section 4 for the benchmark of DRIMM-Synteny on simulated datasets.

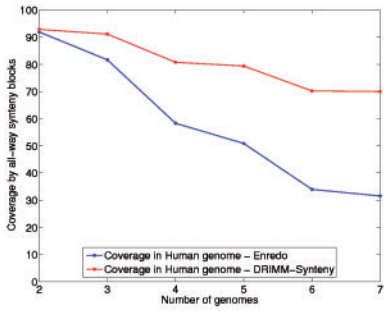


**Fig. 8.** Coverage of human genome by $k$-way synteny blocks for $2 \leq k \leq 7$. While Enredo (Paten *et al.*, 2008) and DRIMM-Synteny produce blocks with similar coverage for small $k$, the $k$-way synteny blocks generated by Enredo have lower coverage for larger $k$.

**Table 1.** Synteny blocks of *K.waltii (K)* and *S.cerevisiae (S)*

| Mult. | Type | No. of blocks | Span on *K* (%) | Span on *S* (%) |
|---|---|---|---|---|
| 3 | K-S-S | 246 | 77 | 78 |
| 2 | K-S | 231 | 13 | 10 |

K-S-S blocks represent blocks of multiplicity 3 that have one instance in *K.waltii* and two instances in *S.cerevisiae* [DCS blocks from (Kellis *et al.*, 2004)]. K-S blocks represent blocks of multiplicity 2 that have one instance in *K.waltii* and one instances in *S.cerevisiae*. The average size of the K-S-S and K-S blocks is 18 and 8 genes, respectively (before removing short blocks).

Figure 9a and b shows the position of DCS blocks (on *S.cerevisiae* genome) generated by DRIMM-Synteny and in Kellis *et al.* (2004) and illustrates that they produced nearly identical results. The statistics of synteny blocks generated by DRIMM-Synteny is given in Table 1. Enredo (Paten *et al.*, 2008), on the other hand, missed many synteny blocks found in Kellis *et al.* (2004) (see section 3 of the Supplementary Material). This raises a question why a rather sophisticated Enredo algorithm failed to reveal synteny blocks constructed using a simple approach from Kellis *et al.* (2004). We emphasize that Enredo is a general synteny block generation tool, while the approach in Kellis *et al.* (2004) has many limitations: it is only applicable to pairs of pre-WGD and after-WGD genomes with small number of additional segmental duplications.

We also ran DRIMM-Synteny on the *S.cerevisiae* genome *alone* with the default parameters.[2] DRIMM-Synteny generated 87 synteny blocks (Fig. 9c), which cover about 51% the genome and reveal a pattern similar to the one shown in Figure 9a and b. This result is consistent with the analysis in Seoighe and Wolfe (1999) (84 blocks with ∼50% coverage). While Seoighe and Wolfe (1999) indeed revealed sister blocks in *S.cerevisiae*, we are not aware of any (general purpose) synteny block generation tool that can automatically construct synteny blocks in highly duplicated genomes. As Figure 9 illustrates, if such a tool was available in 2004 when the paper Kellis *et al.* (2004) was published, it would provide a solid evidence for WGD in *S.cerevisiae* even without additional analysis in Kellis *et al.* (2004). Moreover, the analysis in Kellis *et al.* (2004) would be largely reduced to merely running DRIMM-Synteny.

*How many WGDs have shaped evolution of A.thaliana*? Although the *A.thaliana* genome has been shaped by large duplications, the number and extent of these duplications have been controversial (Sankoff, 2001; Wolfe, 2001). On the one hand, *Arabidopsis'* genomic architecture may be explained by multiple independent segmental duplications. On the other hand, it may originate from a single WGD (or a few rounds of WGDs) followed by genomic rearrangements that split up the original duplicated sequences. The initial *Arabidopsis* studies hypothesized that its ancestor underwent a single WGD (Arabidopsis Initiative, 2000; Blanc *et al.*, 2000). However, Vision *et al.* (2000) argued that *A.thaliana* underwent multiple segmental duplications at different times (rather than WGD). Blanc *et al.* (2003) (see also Bowers *et al.*, 2003) refuted (Vision *et al.*, 2000), confirmed WGD and further found evidence for a second-older WGD that has been partly obscured by other segmental duplications. A good way to resolve this controversy would be to construct synteny blocks and to analyze coverage by blocks of multiplicity larger than 2. However, to the best of our knowledge, the high-coverage non-overlapping decompositions of *A.thaliana* into synteny blocks has not been constructed yet.

The genome of *A.thaliana* contains 28 170 genes (23 129 unique genes). The simplified A-Bruijn graph of this genome has 20 288 vertices and 21 486 edges. Substituting non-branching paths in this graph by single edges results in a graph on 782 vertices and 1224 edges. We further remove short (and potentially spurious) synteny blocks (Supplementary Table S2). While the synteny blocks with multiplicity 2 (supporting one round WGD) span 50% of the genome, the synteny blocks of multiplicity 4 (supporting evidence for two rounds of WGDs) cover only 8% of the genome. If 50% coverage by two-way blocks in *S.cerevisiae* established by Seoighe and Wolfe (1999) was criticized as a proof of WGD in yeast [and required an additional study (Kellis *et al.*, 2004) to establish WGD], why 8% coverage by four-way synteny blocks is a definite proof of two rounds of WGD in *Arabidopsis*. If one counts both three- and four-way synteny blocks, the coverage increases to 16% but in retrospect (see Kellis *et al.*, 2004; Seoighe and Wolfe, 1999) it remains unclear why 16% coverage represents a definite proof of two rounds of WGD.

## 4 DISCUSSION

The rapidly increasing set of sequenced genomes highlights the importance of identifying the synteny blocks in multiple and/or highly duplicated genomes. As the number of analyzed genomes increases, the number of shared genes may decrease substantially. The synteny block generation algorithms based on pairwise comparisons are often limited, since in some cases, the synteny blocks can only be reconstructed by multi-way comparison. We proposed the DRIMM-Synteny algorithm for identifying the non-overlapping synteny blocks and bypassed the difficult threading problem [a bottleneck in Peng *et al.* (2009)] by developing a new A-Bruijn graph approach for solving the SMP.

## ACKNOWLEDGEMENTS

We are indebted to Glenn Tesler for many helpful suggestions that significantly improved the article and to Javier Herrero for help with

[2]See also the Supplementary section 3 for the benchmark of DRIMM-Synteny on eight yeasts genomes (among them, two have undergone WGD).

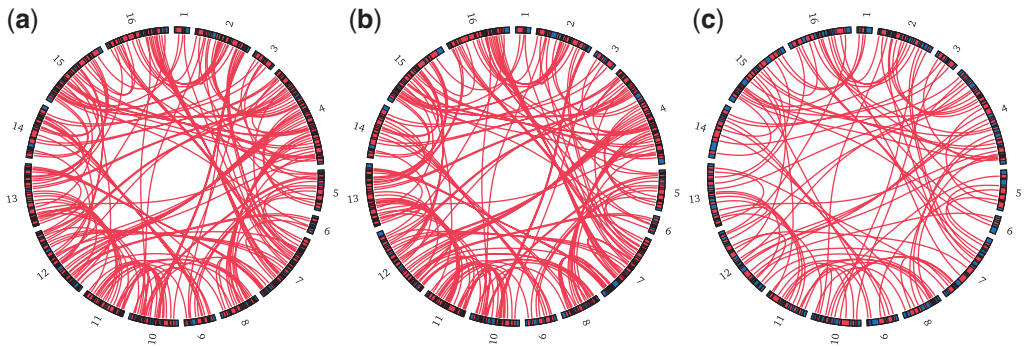


**Fig. 9.** Positions of DCS blocks in *S.cerevisiae*. Sixteen chromosomes of *S.cerevisiae* are presented on the circle. Each arc joins two segments in *S.cerevisiae* forming a DCS block. (**a**) 152 DCS blocks for *K.waltii* and *S.cerevisiae* from Kellis *et al.* (2004), (**b**) 151 DCS blocks generated by DRIMM-Synteny for *K.waltii* and *S.cerevisiae* (with the default parameters). (**c**) 87 DCS blocks generated by DRIMM-Synteny for *S.cerevisiae alone*.

analyzing Enredo's results. We also thank Max Alekseyev, Rustem Aydagulov and Vladimir Dobrynin for their helpful comments. We are grateful to Max Alekseyev, Kevin Byrne, Tao Jiang and Qian Peng for help with generating gene orders of yeast, plant and mammalian genomes.

*Funding*: The project was financially supported by the Vietnam Education Foundation fellowship.

*Conflict of Interest*: none declared.

## REFERENCES

Alekseyev,M. and Pevzner,P. (2009) Breakpoint graphs and ancestral genome reconstructions. *Genome Res.*, **19**, 943.

Arabidopsis Initiative (2000) Analysis of the genome sequence of the flowering plant Arabidopsis thaliana. *Nature*, **408**, 796.

Aury,J. *et al.* (2006) Global trends of whole-genome duplications revealed by the ciliate Paramecium tetraurelia. *Nature*, **444**, 171–178.

Bailey,J. *et al.* (2001) Segmental duplications: organization and impact within the current human genome project assembly. *Genome Res.*, **11**, 1005.

Bandeira,N. *et al.* (2008) Automated de novo protein sequencing of monoclonal antibodies. *Nat. Biotechnol.*, **26**, 1336–1338.

Bao,Z. and Eddy,S. (2002) Automated de novo identification of repeat sequence families in sequenced genomes. *Genome Res.*, **12**, 1269.

Blanc,G. *et al.* (2000) Extensive duplication and reshuffling in the Arabidopsis genome. *Plant Cell Online*, **12**, 1093.

Blanc,G. *et al.* (2003) A recent polyploidy superimposed on older large-scale duplications in the Arabidopsis genome. *Genome Res.*, **13**, 137.

Bourque,G. *et al.* (2004) Reconstructing the genomic architecture of ancestral mammals: lessons from human, mouse, and rat genomes. *Genome Res.*, **14**, 507–516.

Bourque,G. *et al.* (2005) Maximizing synteny blocks to identify ancestral homologs. *Lect. Notes Comput. Sci.*, **3678**, 21.

Bowers,J. *et al.* (2003) Unravelling angiosperm genome evolution by phylogenetic analysis of chromosomal duplication events. *Nature*, **422**, 433–438.

Brudno,M. *et al.* (2003) Glocal alignment: finding rearrangements during alignment. *Bioinformatics*, **19**, 54–62.

Butler,J. *et al.* (2008) ALLPATHS: de novo assembly of whole-genome shotgun microreads. *Genome Res.*, **18**, 810.

Calabrese,P. *et al.* (2003) Fast identification and statistical evaluation of segmental homologies in comparative maps. *Bioinformatics*, **19**, 74–80.

Chaisson,M. and Pevzner,P. (2008) Short read fragment assembly of bacterial genomes. *Genome Res.*, **18**, 324.

Cui,L. *et al.* (2006) Widespread genome duplications throughout the history of flowering plants. *Genome Res.*, **16**, 738.

Darling,A. *et al.* (2004a) GRIL: genome rearrangement and inversion locator. *Bioinformatics*, **20**, 122.

Darling,A. *et al.* (2004b) Mauve: multiple alignment of conserved genomic sequence with rearrangements. *Genome Res.*, **14**, 1394.

Dewey,C. *et al.* (2006) Parametric alignment of Drosophila genomes. *PLoS Comput. Biol.*, **2**, e73.

Dietrich,F. *et al.* (2004) The Ashbya gossypii genome as a tool for mapping the ancient Saccharomyces cerevisiae genome. *Science*, **304**, 304–307.

Fu,Z. *et al.* (2007) MSOAR: a high-throughput ortholog assignment system based on genome rearrangement. *J. Comput. Biol.*, **14**, 1160–1175.

Fujibuchi,W. *et al.* (2000) Automatic detection of conserved gene clusters in multiple genomes by graph comparison and P-quasi grouping. *Nucleic Acids Res.*, **28**, 4029.

Gordon,J.L. *et al.* (2009) Additions, losses, and rearrangements on the evolutionary route from a reconstructed ancestor to the modern Saccharomyces cerevisiae genome. *PLoS Genet.*, **5**, e1000485.

Haas,B. *et al.* (2004) DAGchainer: a tool for mining segmental genome duplications and synteny. *Bioinformatics*, **20**, 3643–3646.

Hampson,S. *et al.* (2005) Statistical detection of chromosomal homology using shared-gene density alone. *Bioinformatics*, **21**, 1339.

Hubbard,T. *et al.* (2002) The Ensembl genome database project. *Nucleic Acids Res.*, **30**, 38.

Jaillon,O. *et al.* (2004) Genome duplication in the teleost fish Tetraodon nigroviridis reveals the early vertebrate proto-karyotype. *Nature*, **431**, 946–957.

Jiang,Z. *et al.* (2007) Ancestral reconstruction of segmental duplications reveals punctuated cores of human genome evolution. *Nature Genet.*, **39**, 1361–1368.

Kellis,M. *et al.* (2004) Proof and evolutionary analysis of ancient genome duplication in the yeast Saccharomyces cerevisiae. *Nature*, **428**, 617–624.

Kent,W. *et al.* (2003) Evolution's cauldron: duplication, deletion, and rearrangement in the mouse and human genomes. *Proc. Natl Acad. Sci. USA*, **100**, 11484–11489.

Koszul,R. *et al.* (2004) Eucaryotic genome evolution through the spontaneous duplication of large chromosomal segments. *EMBO J.*, **23**, 234–243.

Ma,J. *et al.* (2006) Reconstructing contiguous regions of an ancestral genome. *Genome Res.*, **16**, 1557.

Machida,M. *et al.* (2005) Genome sequencing and analysis of Aspergillus oryzae. *Nature*, **438**, 1157–1161.

Martin,N. *et al.* (2007) Gene-interleaving patterns of synteny in the Saccharomyces cerevisiae genome: are they proof of an ancient genome duplication event? *Biol. Direct*, **2**, 23.

Paten,B. *et al.* (2008) Enredo and Pecan: genome-wide mammalian consistency-based multiple alignment with paralogs. *Genome Res.*, **18**, 1814.

Peng,Q. *et al.* (2009) Decoding synteny blocks and large-scale duplications in mammalian and plant genomes. In *Algorithms in Bioinformatics*. Springer, Berlin/Heidelberg, pp. 220—232.

Pevzner,P. *et al.* (2004) De novo repeat classification and fragment assembly. *Genome Res.*, **14**, 1786–1796.

Pevzner,P. and Tesler,G. (2003) Genome rearrangements in mammalian evolution: lessons from human and mouse genomes. *Genome Res.*, **13**, 37–45.

Raphael,B. *et al.* (2004) A novel method for multiple alignment of sequences with repeated and shuffled elements. *Genome Res.*, **14**, 2336.

Sankoff,D. (2001) Gene and genome duplication. *Curr. Opin. Genet. Dev.*, **11**, 681–684.

Sankoff,D. and Blanchette,M. (1997) The median problem for breakpoints in comparative genomics. *Lect. Notes Comput. Sci.*, **1276**, 251–263.

Scannell,D. *et al.* (2007) Independent sorting-out of thousands of duplicated gene pairs in two yeast species descended from a whole-genome duplication. *Proc. Natl Acad. Sci. USA*, **104**, 8397.

Seoighe,C. and Wolfe,K. (1999) Yeast genome evolution in the post-genome era. *Curr. Opin. Microbiol.*, **2**, 548–554.

Simillion,C. *et al.* (2008) i-ADHoRe 2.0: an improved tool to detect degenerated genomic homology using genomic profiles. *Bioinformatics*, **24**, 127.

Skiena,S. (1990) *Combinatorics and Graph Theory with Mathematica*. Addison-Wesley, Redwood City, California.

Soderlund,C. *et al.* (2006) SyMAP: a system for discovering and viewing syntenic regions of FPC maps. *Genome Res.*, **16**, 1159.

Swidan,F. *et al.* (2006) An integrative method for accurate comparative genome mapping. *PLoS Comput. Biol.*, **2**, e75.

Van de Peer,Y. (2004) Tetraodon genome confirms Takifugu findings: most fish are ancient polyploids. *Genome Biol.*, **5**, 250.

Vandepoele,K. *et al.* (2002) The automatic detection of homologous regions (ADHoRe) and its application to microcolinearity between Arabidopsis and rice. *Genome Res.*, **12**, 1792.

Vision,T. *et al.* (2000) The origins of genomic duplications in Arabidopsis. *Science*, **290**, 2114.

Wolfe,K. (2001) Yesterday's polyploids and the mystery of diploidization. *Nature Rev. Genet.*, **2**, 333–341.

Wolfe,K. and Shields,D. (1997) Molecular evidence foran ancient duplication of the entire yeast genome. *Nature*, **387**, 709.

Zerbino,D. and Birney,E. (2008) Velvet: algorithms for de novo short read assembly using de Bruijn graphs. *Genome Res.*, **18**, 821.