BMC Bioinformatics

CORRECTION Open Access

Correction to: SLR: a scaffolding algorithm based on long reads and contig classification

Junwei Luo[1*], Mengna Lyu[1], Ranran Chen[1], Xiaohong Zhang[1], Huimin Luo[2] and Chaokun Yan[2]

Correction to: BMC Bioinformatics (2019) 20:539
https://doi.org/10.1186/s12859-019-3114-9

Following publication of the original article [1], the author reported that there is an error in the original article;

1. The figures' order in HTML and PDF are incorrect.
   In the original article incorrect Fig. 1 is the correct Fig. 4.
   In the original article incorrect Fig. 2 is the correct Fig. 5.
   In the original article incorrect Fig. 3 is the correct Fig. 6.
   In the original article incorrect Fig. 4 is the correct Fig. 1.
   In the original article incorrect Fig. 5 is the correct Fig. 2.
   In the original article incorrect Fig. 6 is the correct Fig. 3.

In this correction article the figures are shown correct.

Author details
[1]College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454000, China. [2]School of Computer and Information Engineering, Henan University, Kaifeng 475001, China.

Published online: 10 February 2020

Reference

1. Luo J, et al. SLR: a scaffolding algorithm based on long reads and contig classification. BMC Bioinformatics. 2019;20:539. https://doi.org/10.1186/s12859-019-3114-9.

The original article can be found online at https://doi.org/10.1186/s12859-019-3114-9

* Correspondence: luojunwei@hpu.edu.cn
[1]College of Computer Science and Technology, Henan Polytechnic University, Jiaozuo 454000, China
Full list of author information is available at the end of the article

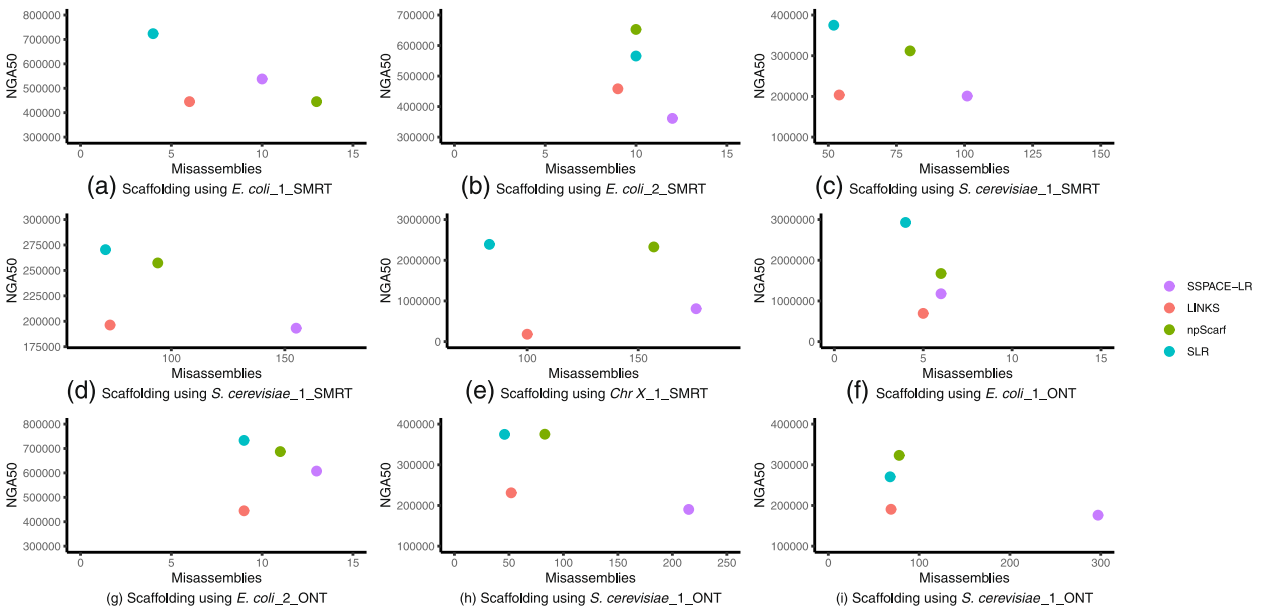

Fig. 1 Nine figures plotting NGA50 vs Misassemblies. The results of SLR usually can be found in the top-left corner, which can illustrate the advantage of SLR

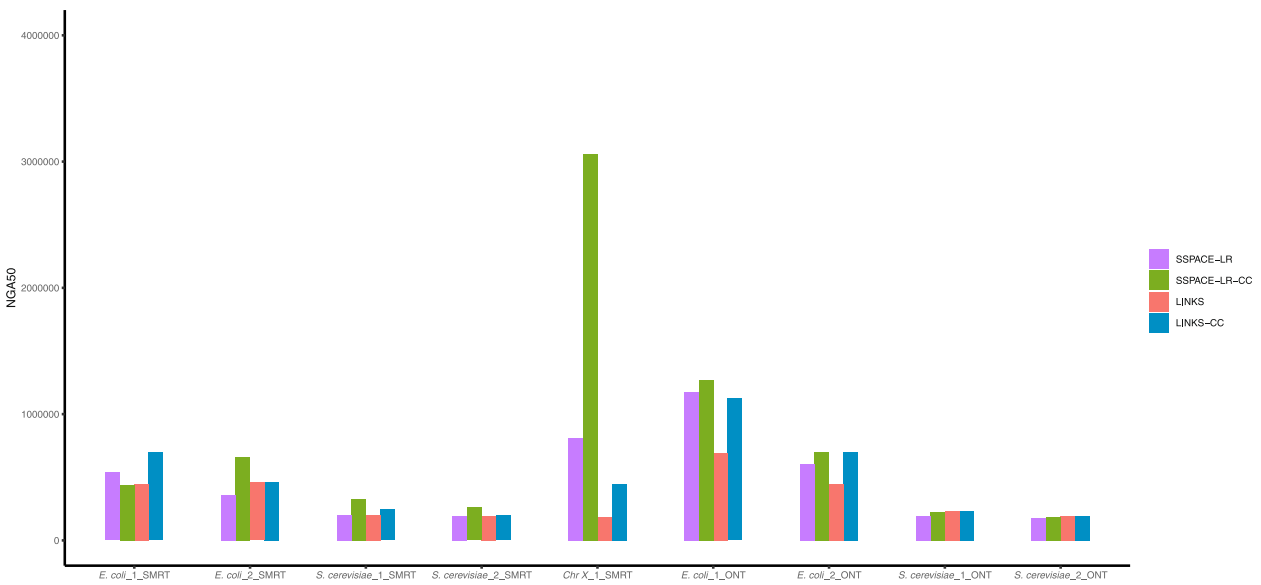

Fig. 2 Contig classification combines with SSPACE-LR and LINKS

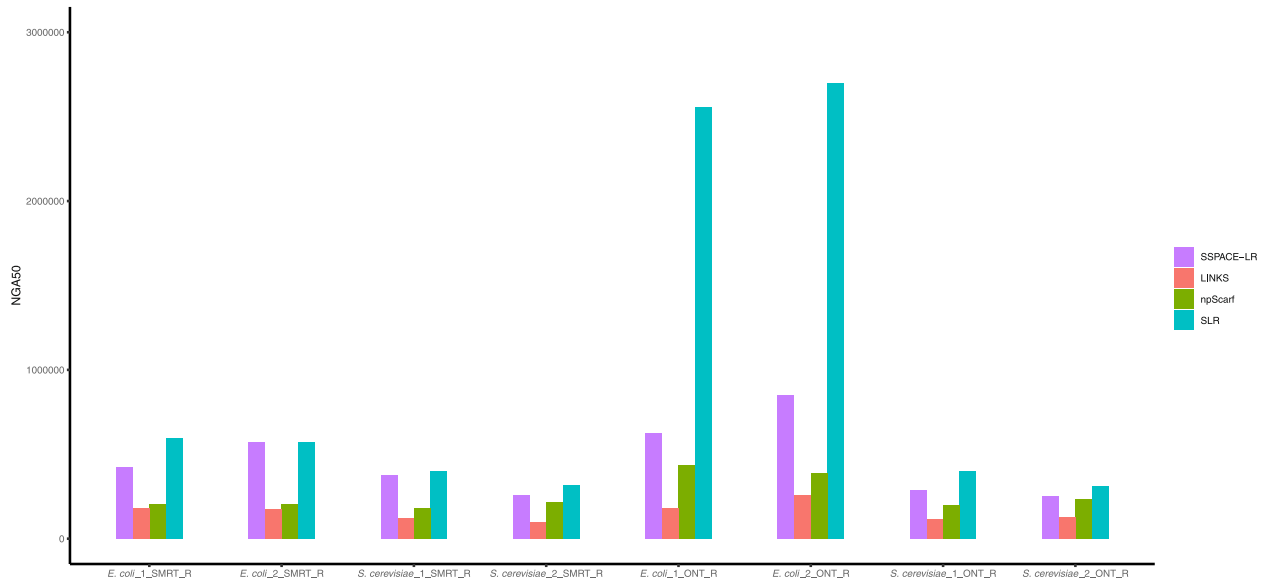

Fig. 3 NGA50 for datasets produced by repeat-aware evaluation framework

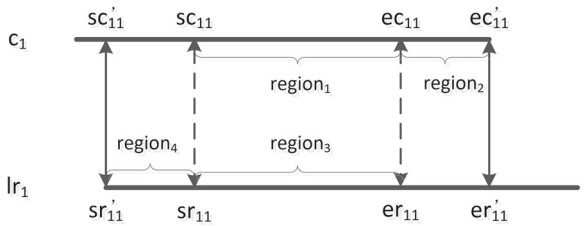

Fig. 4 An example of alignment position revision. For an alignment given by the alignment tool, the region [sr_{11}, er_{11}] ($region_3$) in the long read lr_1 is aligned with the region [sc_{11}, ec_{11}] ($region_1$) in the contig c_1. Because $sr_{11} < sc_{11}$ and $LEN(lr_1) - er_{11} > LEN(c_1) - ec_{11}$, it means the region [0, sr_{11}] ($region_4$) in lr_1 is not aligned with c_1, and the region [ec_{11},$LEN(c_1) - 1$] ($region_2$) is not aligned with lr_1. However, when lr_1 is truely aligned with c_1 and the alignment is reliable, $region_4$ should be aligned with the region [$sc_{11} - sr_{11}$, sc_{11}] in c_1, and $region_2$ should be aligned with the region [er_{11}, $er_{11} + LEN(c_1) - ec_{11}$]. Because of the high sequencing error rate in long reads, the alignment tool usually does not provide accurate alignment start and end positions. Then, SLR sets $sc11' = sc_{11} - sr_{11}$, $sr11' = 0$, $ec11' = LEN(c_1) - 1$ and $er11' = er_{11} + LEN(c_1) - ec_{11}$. When the alignment is reliable, the region [$sc11'$, $ec11'$] in c_1 is aligned with the region [$sr11'$, $er11'$] in lr_1

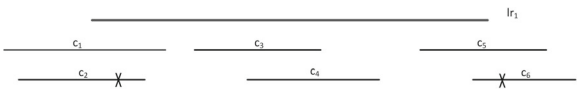

Fig. 5 There are six contigs (c_1,c_2,c_3,c_4,c_5,$andc_6$) that can be aligned with the long read lr_1. Because c_1 and c_2 are simultaneously aligned with the left end of lr_1, SLR retains only contig c_1 which has the greatest alignment length, and deletes the alignment information between c_2 and lr_1. Because c_5 and c_6 have been simultaneously aligned with the right end of lr_1, we keep only c_5, and delete the alignment information between c_6 and lr_1. Finally, SLR determines the orders and orientations of c_1, c_3, c_4 and c_5, which form a local scaffold

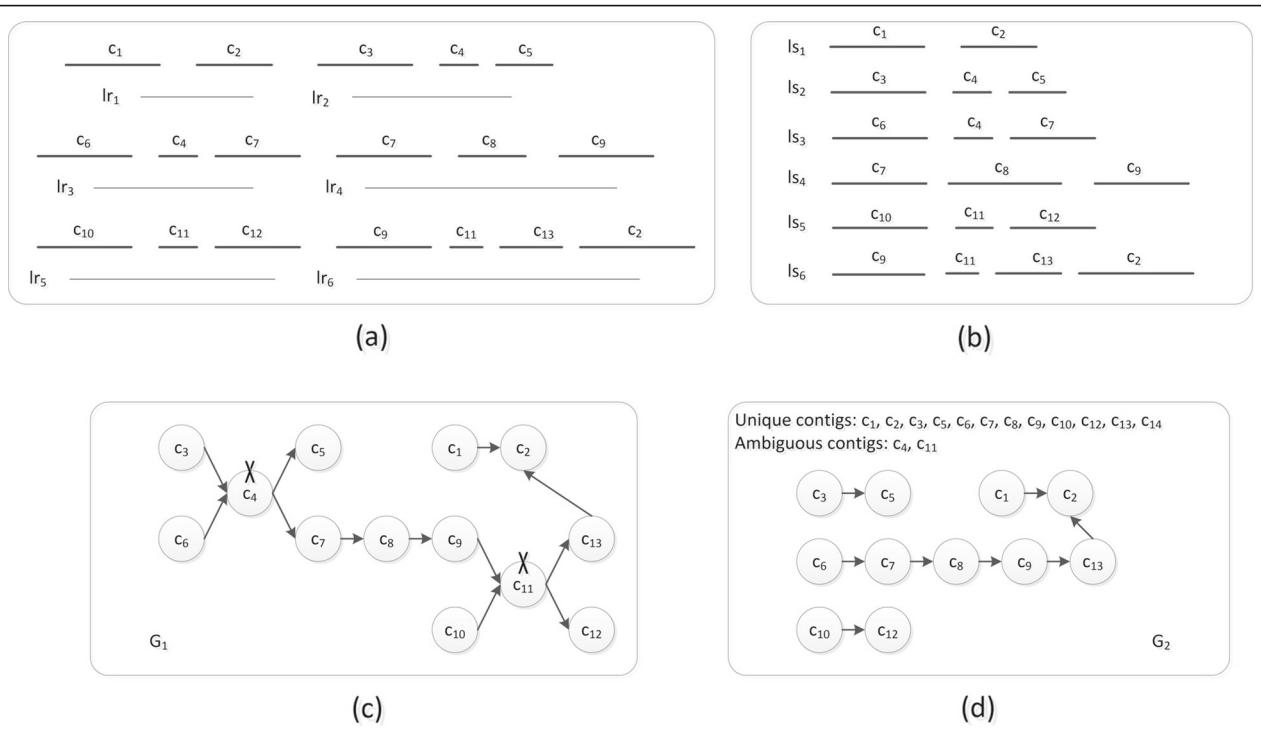

Fig. 6 (a) There are six long reads: lr_1, lr_2, lr_3, lr_4, lr_5, and lr_6. The contigs c_1 and c_2 are aligned with lr_1. c_3, c_4 and c_5 are aligned with lr_2. c_6, c_4 and c_7 are aligned with lr_3. c_7, c_8 and c_9 are aligned with lr_4. c_{10}, c_{11} and c_{12} are aligned with lr_5. c_9, c_{11}, c_{13} and c_2 are aligned with lr_6. We assume that all these alignments are forward, and all contigs are longer than L_{ca}. (b) Based on the alignment result described in (a), SLR obtains six local scaffolds: ls_1, ls_2, ls_3, ls_4, ls_5, and ls_6. (c) The scaffold graph G_1 is built using all contigs. We find that G_1 is complicated. (d) Based on the contig classification method described in Section 2.2, the contigs can be divided into two categories. Because c_4 is located in the middle position of ls_2 and ls_3 and has two distinct 3′-end neighbours and two distinct 5′-end neighbour contigs, it is identified as an ambiguous contig. c_{11} is also an ambiguous contig. The remaining contigs are identified as unique contigs. The scaffold graph G_2 is built based on unique contigs and is thus less complicated than G_1