# LCR_Finder: A de novo low copy repeat finder for human genome

Xuan Liu, David Wai-lok Cheung, Hing-Fung Ting,
Tak-Wah Lam[*], and Siu-Ming Yiu[*]

Department of Computer Science, The University of Hong Kong, Hong Kong
{xliu, dcheung, hfting, twlam, smyiu}@cs.hku.hk

**Abstract.** Low copy repeats (LCRs) are reported to trigger and mediate genomic rearrangements and may result in genetic diseases. The detection of LCRs provides help to interrogate the mechanism of genetic diseases. The complex structures of LCRs render existing genomic structural variation (SV) detection and segmental duplication (SD) tools hard to predict LCR copies in full length especially those LCRs with complex SVs involved or in large scale. We developed a de novo computational tool LCR_Finder that can predict large scale (>100Kb) complex LCRs in a human genome. Technical speaking, by exploiting fast read alignment tools, LCR_Finder first generates overlapping reads from the given genome, aligns reads back to the genome to identify potential repeat regions based on multiple mapping locations. By clustering and extending these regions, we predict potential complex LCRs. We evaluated LCR_Finder on human chromosomes, we are able to identify 4 known disease related LCRs, and predict a few more possible novel LCRs. We also showed that existing tools designed for finding repeats in a genome, such RepeatScout and WindowMasker are not able to identify LCRs and tools designed for detecting SDs also cannot report large scale full length complex LCRs.

## 1 Introduction

Complex low copy repeats (LCRs, also termed as segmental duplications), which are composed of multiple repeat elements either in direct or inverse directions, provide the structural basis for diverse genomic variations and combinations of variations (Zhang et al. 2009). Around 5% of sequenced portion of human genome is composed of LCRs. LCRs usually range from 10Kb to several hundred Kb. Among the copies, there are common regions (with sequence similarity as high as 95% (Babcock et al. 2007). However, there may also be big gaps between common regions and not all common regions exist in all copies. This makes the detection process very difficult. Complex LCRs are found to trigger and mediate genomic rearrangement including deletion, duplication etc., by altering gene dosage, and result in human genetic disorders (Stankiewicz et al. 2002; Zhang et al. 2009). Several human genetic disorders caused by genomic recombination are reported to be triggered and mediated by LCRs,

[*] Joint corresponding authors: to whom correspondence should be addressed.

such as Familial Juvenile Nephronophthisis at chromosome band 2q13 caused by 2.9Mb deletion (Saunier et al. 2000; OMIM 607100), William-Beuren syndrome (WBS) at chromosome band 7q11.23 caused by 1.5Mb to 1.8Mb deletion (Valero, M. C. et al. 2000; OMIM 194050), Sotos syndrome (Sos) by 5q35 deletion (Kurotaki, N. et al. 2005; OMIM 117550) and Smith-Magenis syndrome (SMS) by deletion on 17p11.2 ( Claudia M.B. Carvalho et al. 2008 ; OMIM 182290). Genes inside the regions of LCRs have a high chance to develop into diseases. Identifying the locations of LCRs thus is important and the pattern of LCRs can help to unveil the complicated mechanism of genetic diseases.

Several computational methods have been proposed to identify copy number repeats and duplications such as RepeatScout (Price et al. 2005) and WindowMasker (Morgulis et al. 2006). Although detecting repeats on genome sequence is relatively well studied, there are only a limited number of tools for LCRs/SDs such as DupMasker (Jiang et al. 2007). The outcome of existing software for detecting LCRs, especially complex LCRs is not satisfactory.

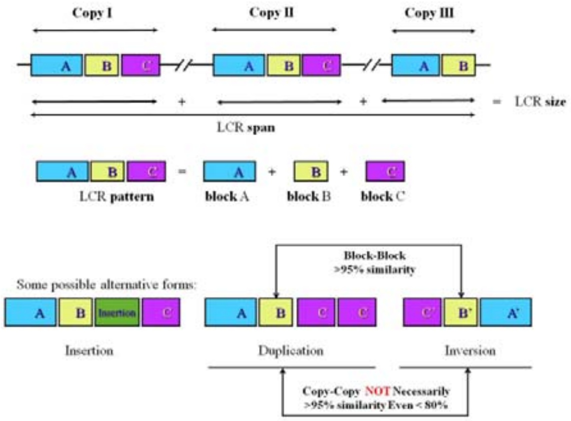

**Fig. 1.** Illustration of LCR structure. An LCR is composed of several copies, and each copy is an alternative form of pattern of this LCR. LCR pattern is a combination of several tandem/disconnected blocks. LCR span is defined as the minimum interval that all copies are included. LCR size is length sum of all copies. There is a high similarity between the blocks with the same label in different copies. Insertion, deletion, inversion, translocation, duplication etc. could be introduced to blocks or gaps between blocks to make an alternative form.

As mentioned in the above, LCR structure is complicated (Fig 1). Unlike repeats, complex LCRs are higher level combination of different but associated repeats transposed to specific genomic regions, creating duplication blocks with mosaic architecture of juxtaposed duplicated segments (Jiang et al. 2007). Each copy of LCR pattern is an alternative form of block patterns and not necessarily the same as others and may even quite different in structure. The similarity between block patterns in different LCR copies varies and can range from 30% to 95%. Due to the gaps between blocks, the similarity between the whole LCR copies is even lower. Note that not all block patterns appear in all copies of the LCR and duplication, translocation, and inversion of patterns are not uncommon. Copy number of LCR is usually below 5, while the repeats it contains may either highly or lowly repetitive.

Despite the fact that LCRs could be considered as repeats combination with SVs introduced, Repeat/SV tools are not able to detect them. SV-targeting tools reveal variance at the same loci between two input sequences, instead of different loci of one single sequence. There are Repeat-targeting methods designed to identify repetitive sequences in a single input sequence, such as RepeatMasker (Smit et al. 2011), RepeatScout (Price et al. 2005) and WindowMasker (Morgulis et al. 2006). However, most of these tools were not designed for large repeat patterns, thus many small-scale highly repetitive patterns are reported which cannot be easily grouped to locate the LCRs. For example, we ran RepeatMasker on human chromosome 17 (GRCh37.p9), 165,382 repetitive sequences, covering 37,757,301/81,195,210 bp of chromosome 17 with an average length of repeat as 228.30 are reported. These 165,382 repeats were mapped to only 1268 repeat patterns (55 repeat families, including SINE, LINE, small RNAs, low complex sequences and so on). It is not trivial how to "glue" these short patterns together to locate LCRs.

On the other hand, there are database dependent tools that limit the search scope to existing human segmental duplication libraries and are not applicable to other species. DupMasker reported 47, 62 and 82 duplicons for each SMS-LCRs copy respectively instead of 3 full length LCR copies. Also, DupMasker uses the result of RepeatMasker as part of the input, it is very time consuming.

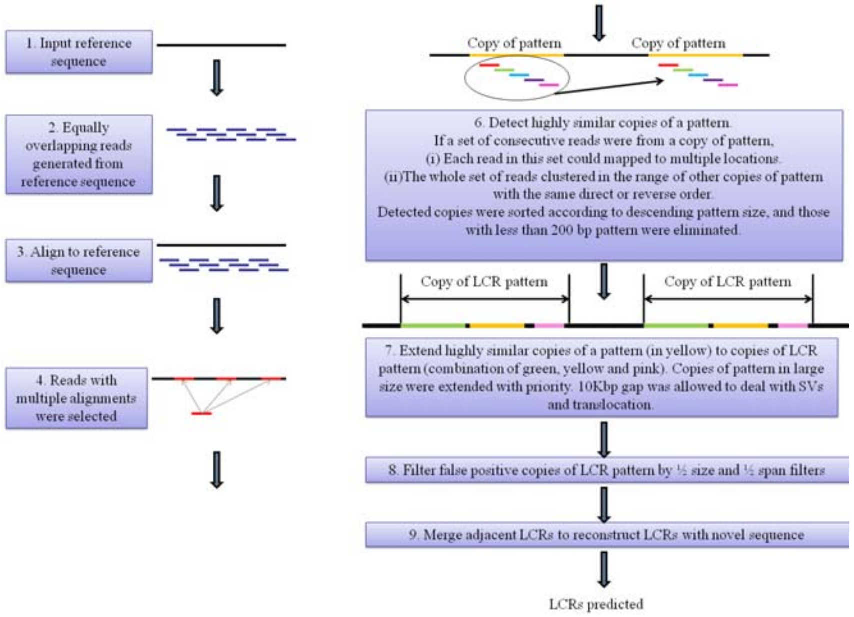
1. Input reference sequence
2. Equally overlapping reads generated from reference sequence
3. Align to reference sequence
4. Reads with multiple alignments were selected
Copy of pattern
Copy of pattern
6. Detect highly similar copies of a pattern.
If a set of consecutive reads were from a copy of pattern,
(i) Each read in this set could mapped to multiple locations.
(ii)The whole set of reads clustered in the range of other copies of pattern with the same direct or reverse order.
Detected copies were sorted according to descending pattern size, and those with less than 200 bp pattern were eliminated.
Copy of LCR pattern
Copy of LCR pattern
7. Extend highly similar copies of a pattern (in yellow) to copies of LCR pattern (combination of green, yellow and pink). Copies of pattern in large size were extended with priority. 10Kbp gap was allowed to deal with SVs and translocation.
8. Filter false positive copies of LCR pattern by ½ size and ½ span filters
9. Merge adjacent LCRs to reconstruct LCRs with novel sequence
LCRs predicted

**Fig. 2.** Workflow of LCR_Finder.

In this paper, we introduce a novel de novo LCR detection tool, called LCR_Finder, which does not rely on known information on segmental duplications and can large scale LCRs in complex structures, on a given genomic sequence (e.g. chromosome level). Technical speaking, we exploit existing read alignment tools to solve the problem as follows. We (1) computationally generate single end reads from the given genome, (2) align them back to the genome, (3) locate large repetitive patterns using reads with multiple aligned positions, (4) extend copies with large gap allowed and (5) filter out false positives and report potential LCRs (Fig 2). We evaluate LCR_Finder on human chromosomes and show that it can identify four known diseases related LCR loci and report a few more potential novel LCRs. We also compare the results reported by RepeatScout and WindowMasker and found that their results are not as good as given by LCR_Finder.

# 2 Methods

## 2.1 Problem Definition

We define a block as a DNA pattern with size > 5K. A LCR is defined as a set of blocks B:={$b_1$,$b_2$,...,$b_m$}, m≥1. A LCR(C) consists of 2 to 5 repeated regions represented as C:={$c_1$,$c_2$,...,$c_n$} , 2≤n≤5. Each copy is a collection of blocks (with mutations) in set B, $c_i$ :={ $b_{i_1}, b_{i_2}, \dots, b_{i_t}$}. Note that the same block can appear more than once in each copy (probably with different mutations).
For each $i_k$, $1 \le i_k \le i_{t,}$ $b_{i_k}$ is represented by a pair of genomic locations indicating the starting and end positions of that block on reference, bi=($s_i$,$e_i$), $s_i$<$e_i$. $b_{i_1}, b_{i_2}, \dots, b_{i_t}$ are sorted according to increasing starting position order. Note that the same block that appears in each copy is not exactly the same.
The span of $c_i$ is defined as Span($c_i$) =($e_{i_t} - s_{i_1}$), and length of $c_i$ is defined as Len($c_i$) = $\sum_{j=1}^{it}\left(e_{i_j} - s_{i_j}\right)$. Max_span(C):=max{Span($c_i$)} and Max_len(C):= max{ Len($c_i$)}. Each copy $c_i$ := { $b_{i_1}, b_{i_2}, \dots, b_{i_t}$}={($e_{i_1} - s_{i_1}$), ($e_{i_2} - s_{i_2}$), ..., ($e_{i_t} - s_{i_t}$)} satisfies the following properties:

- i) 0< $s_{i_{j+1}} - e_{i_j}$<100K for all j such that $i_t \le j \le i_j$;
- ii) e$i_j$ -s$i_j$>5K;
- iii) Len($c_i$) ≥ 1/2 ×Max_len(C);
- iv) Span($c_i$) ≥1/2 × Max_span(C);
- v) The similarity of each block that appears in $c_i$ and the corresponding block in B should be more than 30%.

The problem is to retrieve all LCRs from a human genome. The problem is in high complexity due to a low similarity requirement and a large number of possible mutations and structural variations, so we designed a heuristic solution to solve the problem.

### 2.2 Overlapping Reads Generation and Alignments

We generate consecutive overlapping reads from the input genomic sequence. The locations of where the reads come from are marked. For highly similar repeats, a set of consecutive reads should form a similar sorted overlapping pattern (either in direct or reverse order) when they were mapped to these repeat regions. Read length, overlapping length and mismatches in alignment can be adjusted (default values: 100 bp read, 20 overlapping length, with 4 mismatches in alignments) according to similarity requirement. Longer read length and fewer mismatches go with higher similarity requirement, otherwise lower.

### 2.3 Small-size Highly Similar Sequences Detection

We mapped those overlapping reads back to reference sequence using Soap3[1](Liu et al. 2012), and selected those reads with multiple alignments. To identify reads from repetitive sequences, we clustered the reads as follows. Let us start with read x. For every position read x could be mapped to, we considered the next consecutive read (x+1). If read (x+1) could be mapped to at least two corresponding locations with ± 10bp, it was chained to x, and we went on with (x+2), otherwise stop and report the chain starting from x. x is the chain head.

We filtered out chains with less than 50 supporting reads, and sorted the rest in descending order. Each chain was supposed to correspond to a repeat pattern. So far, efforts have been put on how to detect repeats using stringent criteria (95%-98% similarity requirement). In LCR structure, those repeats were acted as skeletons, and in the next step, we consider each set of repeats as skeleton of a LCR and extend them one at a time. If a set of repeats were covered by others during extension, this set of repeats would be eliminated.

### 2.4 Basic Extension

Given copies of a repeat pattern, we chose the copy with smallest starting coordinates as model and regarded others as candidates. We applied an extension procedure to model repeat on two directions, one at a time (Fig 3).

Boundary of model was extended by L = 5 Kb region. All valid alignments of reads within this region consecutive to model boundary were clustered. Initially, each alignment was considered as a cluster. Iteratively merge two clusters if there was at least one alignment in each of them, had a distance below 200 bp (10 times read overlapping length), until no more clusters could be merged. The minimum interval on reference sequence that covered all alignments in a cluster was calculated for each cluster. Only intervals larger than of L/2were kept (Fig 3B).

In order to deal with large SVs – insertion, deletion, inversion and translocation with large translocated distance, we applied large extending gap G (default 100Kb) to cover SVs smaller than G. For each candidate, if there was at least one interval whose

[1] Other alignment tools can be used.

gap to this candidate on either side was less than G, then update the corresponding boundary to include the nearest interval. If at least one candidate was updated, then this extension step was considered as successful. Once three consecutive unsuccessful extensions were found, extension to this direction was stopped. Meanwhile, the corresponding boundary of model scrolled 3 steps back to exclude the last three failed extensions at the end (Fig 3C).

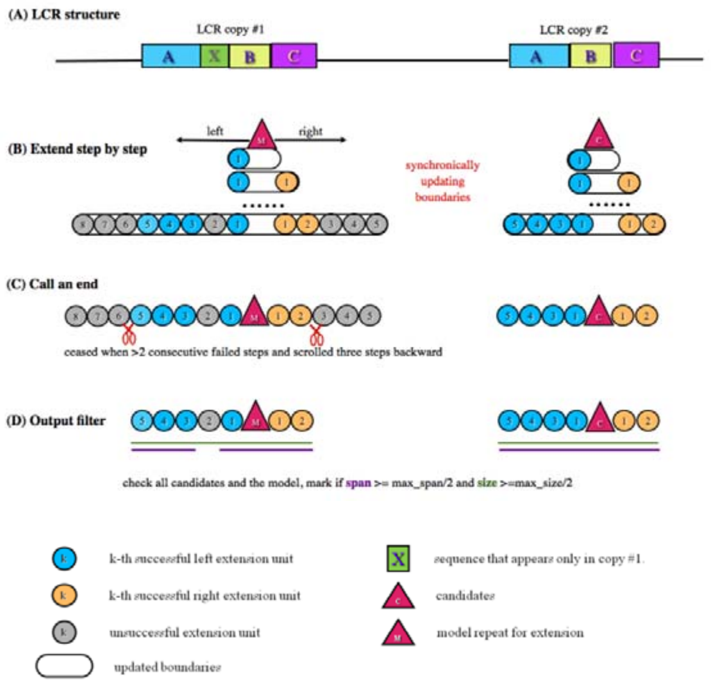
(A) LCR structure
LCR copy #1
A
X
B
C
LCR copy #2
A
B
C
(B) Extend step by step
left
right
M
C
synchronically updating boundaries
(C) Call an end
ceased when >2 consecutive failed steps and scrolled three steps backward
(D) Output filter
check all candidates and the model, mark if span >= max_span/2 and size >=max_size/2
k-th successful left extension unit
k-th successful right extension unit
unsuccessful extension unit
updated boundaries
sequence that appears only in copy #1
candidates
model repeat for extension

**Fig. 3.** Illustration of basic extension. (A) Example of LCR structure. (B) Extension process. Candidate regions marked as triangles were input for extension. Each circle was an extension unit consecutively to current boundaries. If reads from this unit could cluster within certain distance within boundaries of candidates, this extension unit was successful. (C) Call and end and cut the failed extension tails. One side extension was called to an end when three consecutive failed units were witnessed at the boundary. These three failed units were cut and extension to this direction ceased. When both directions reached an end, the whole extension process stopped. (D) Output filter. A filter was applied to eliminate false positives.

When both directions reached to an end, size and span were calculated for model and candidates to eliminate false positives. Size was defined as sum of model/candidate initial size and all extended region for this model/candidate. Span was defined as the minimum interval that covered the initial model/candidate and all other extended regions. For each copy of LCR, calculate the maximum span and size. If more than one copy passed ½ span and ½ size filters, those eligible copies were reported (Fig 3D). For example, suppose one LCR pattern was 50Kb, and there was a 4Kb deletion at the end of one candidate X but the deletion region appeared 80Kb away from this candidate. During extension, this region was incorrectly included as well as 80Kb non-LCR sequence by X (size = 50Kb; span = 130Kb). However, due to ½ span filter, X and other copies were abandoned.

### 2.5 Merge Adjacent LCRs to Deal with Large Size Novel Sequences

We merged adjacent LCRs to deal with large gaps (2.5 L <gap< M) between LCR blocks. If there was novel sequence in model (sequence that didn't appear in other copies) with >2.5 L length made model copy failed to go across the gap during extension. However, we were able to connect LCRs on both sides of the gap and reconstruct the original LCRs (Fig 4). If two LCRs 1 and 2 had the same copy number and for each copy in LCR1 there was a copy in LCR2 that they were either overlapping or within distance < M (default 200Kb), the minimum interval that covered the pair of copies was reported as a reconstructed copy. All reconstructed copies were filtered by ½- span filter to reduce false positives.

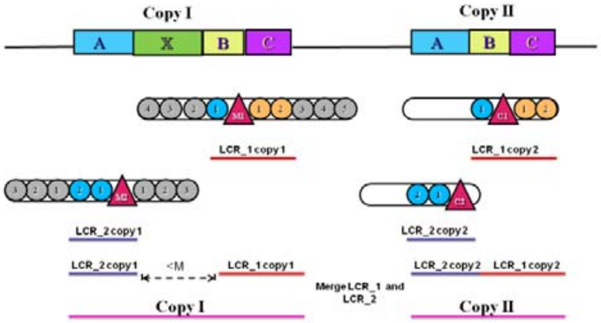

**Fig. 4.** Merge adjacent LCRs to handle novel insertions. Two LCRs were found around novel insertion in Copy I. LCR_1 and LCR_2 were merged to form full-length LCR copies.

## 3 Results

In order to evaluate the performance of LCR_Finder on real data, we tested our tool on 4 human chromosomes, and to each of them there is a known disease-related LCR. We also compared the results of LCR_Finder with RepeatScout, windowMasker and RepeatMasker (chr17 only) on human chromosomes 2, 5, 7, 17.

### 3.1 Performance of LCR_Finder

Human chromosomes (GRCh37.p9) were downloaded from NCBI (http://www.ncbi.nlm.nih.gov). Considering there is at least 95% sequence similarity between repeats, we simulate 100 bp single end reads, and each read overlaps with the next one by 20bp. When they are mapped to reference sequence using Soap3, 4 mismatches are allowed and both orientations are valid.

Our experiments were implemented on Linux86 64 system with 8G memory using Perl. We were able to predict 59, 34, 86, 44 LCRs for chromosome 2, 5, 7, 17 respectively. 15, 9, 16, 9 out of 59, 34, 86, 44 LCRs had LCR copies larger than 100Kb including the 4 known disease-related LCRs. Comparing our results with 4 known disease-related LCRs (one for each chromosome), we successfully identified all 4 known LCRs.

### 3.2 Supporting evidence on novel LCRs

We further investigated the 45 novel LCRs (>100Kb copy size). We were able to see some high similarity blocks (>95%) tandem or interspersed arranged in one LCR copy when it was aligned to other LCR copies using BLASTN (http://blast.ncbi.nlm.nih.gov). The overall similarity between LCR copies was not necessarily to be high, but highly similar blocks should be observed. When we used one copy of LCR as query and Blast other copies (subject) to it, subject sequence were divided into several tandem/dispersed long highly similar sequences to query. In contrast to the case of one LCR copy and a non-LCR sequence, a small number of short subject sequences were sparsely aligned to query (see Fig 5).

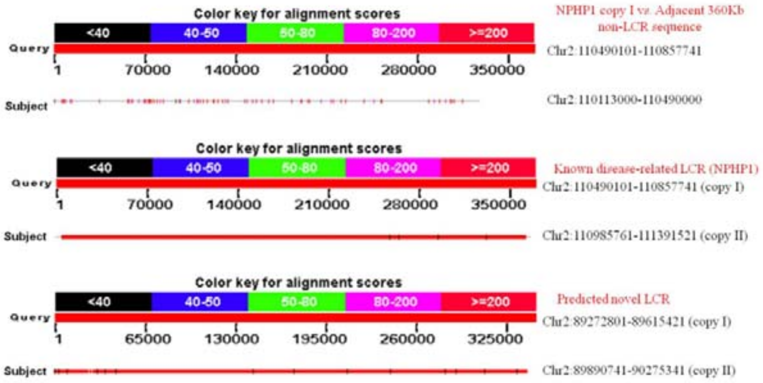

**Fig. 5.** Blast results of non-LCR sequences, known related disease LCR sequences, and predicted LCR sequences. We blast NPHP1 copy I to similar size adjacent sequence (top), two copies of NPHP1 LCR (middle) and two copies of predicted LCR (bottom).

### 3.3 LCR_Finder Limitations

Limitations of LCR_Finder are discussed in this section. Results of LCR_Finder are dependent on how parameters are set, including extending unit length L and gap tolerance G. We compared the results under L = 2Kbp, 5Kbp and 10Kbp and G =50Kbp, 100Kbp, 200Kbp separately. On each (L ,G) pair, we ran LCR_Finder on chromosome 17 and show visualized results on region chr17:15Mb-21Mb using Circos ( Krzywinski et al., 2009) (Fig 6). (i) LCR copy size reported by LCR_Finder is related to extending unit length L. The higher L is, the less the boundary resolution is. Because the extending unit at the boundary is limited to at least L /2, there are at most L /2 non-LCR region being counted as part of LCR copy, which means at most L /2 non-LCR region was incorrectly reported or at most L /2 LCR region was missed. For example, in SMS proximal copy, it could be seen that boundaries found under parameters (10K,100K) were less precise than (5K,100K). (ii) LCR_Finder were not able to identify LCRs with novel insertion in model copy with length larger than 2.5L (2L in some cases depending on extending start position) during extension, and deletion in model copy with deletion size larger than M. For example, LCR_Finder failed to report full length SMS-REPs when (L ,G) =(5K, 50K), because there was ~10Kb region between block A to block C in model copy (distal copy) deleted in proximal and middle copies. Although 1Kb block B existed for all copies, it was too small for minimum successful extension requirement (L /2=2.5Kb). Three consecutive unsuccessful extensions were found after block A, so LCR_Finder reported only block A for all copies. However when (L ,G) =(10K, 50K), ~10K gap was smaller than 2L (20K), thus full-length copies were reported. Besides when (L ,G) =(2K, 50K), block B made an extension successful since it was larger than L /2 (1Kb) and followed by 2 failed extensions, the next extension was successful, so full length SMS-REPs were reported. (iii) Insufficient gap tolerance G caused loss of LCR copy. LCR_Finder could not handle insertion, deletion and inversion larger than G and translocation with transcending distance larger than G. Successful extension happened only when extension unit was found within G of candidate boundaries (Fig 3). Considering the complex structures of LCRs, we recommend users to adopt a set of various parameters to capture LCRs with a wide range of sizes and structures.

### 3.4 Tools Comparison

To compare LCR_Finder to RepeatScout (RS) and WindowMasker (WM), we ran RS and WM on human chromosomes 2, 5, 7 and 17, one chromosome at a time. Numbers of detected LCRs are listed in Table 1.

RS and WM had lower time cost than LCR_Finder and predicted a larger number of CNVs. But most of CNVs they predicted were small-scale, and none of them captured LCRs listed in Table A1. The output of RS was CNV patterns while WM reported intervals covered all copies of a pattern. Thus we calculated the total length (sum of intervals) and average length (average of intervals) of LCR_Finder and WM. Besides, average patten length (average of patterns reported by RS and average copy length by LCR_Finder) comparison was conducted for LCR_Finder and RS. We can observe from Fig 7 that WM captured larger total length on chromosome 2 and 5, but the average length was significantly lower than out tool. In addition, average pattern length of LCR_Finder was much high than RS. Although the total number of CNVs/LCRs LCR_Finder predicted was lower than RS and WM, LCR_Finder was more efficient in detecting large-scale LCRs.

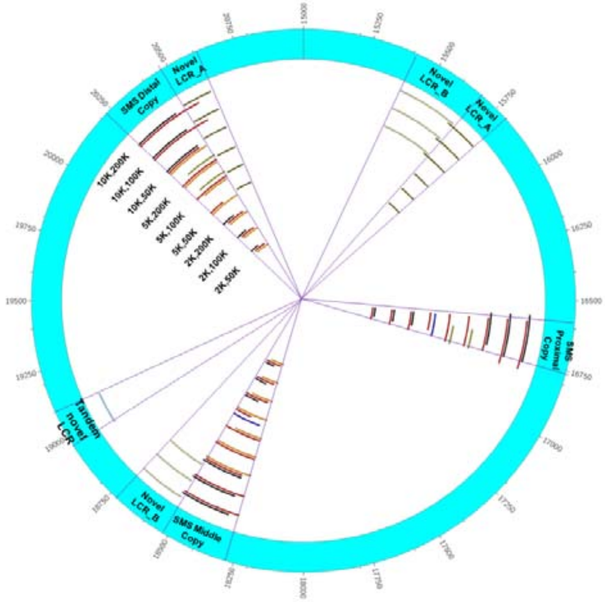

**Fig. 6.** Results of LCR_Finder on various extending parameters. Only >10Kb results were presented. The outermost circle (in blue) with ticks represents region chr17:15Mb-21Mb on the given sequence. Ticks are in Kb scale. Results of LCR_Finder on different parameters (L ,G) were arranged as inner circles. Copies of the same LCR were marked with the same color. SMS-REPs reported by LCR_Finder were colored in red. 2 novel LCRs were predicted.

## 4 Conclusions and Discussions

Although various genomic structural variation detection tools have been developed using the next-generation sequencing data, due to the difficulty in capturing the characteristics of complex low copy repeats, existing methods are not yet satisfactory. In this paper, we presented a novel tool focusing on complex low copy repeats. Besides basic repeats discovery, our tool is capable of combine different sets of repeats according to their genomic locations and report large-scale complex low copy repeats coordinates despite their complex structures. Our tool helps to interrogate genomic coordinates and understand mechanisms of genetic diseases.

Several issues are remained to be better understood and investigated in the future. A more precise formulation and definition of LCR, a more systematical parameter setting up in less ad hoc manner and a more comprehensive evaluation method such as validating putative LCRs with existing LCR database will facilitate the detection of complex low copy repeats.

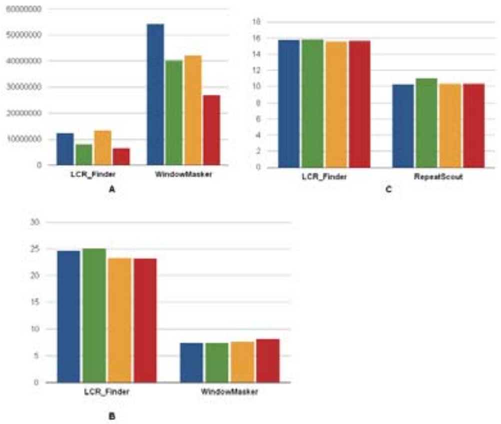

**Fig. 7.** Performance comparison between LCR_Finder and other software. LCR_Finder, WindowMasker (WM) and RepeatScout (RS) were tested using 4 human chromosomes. We calculated (A) Total Length (bp) (B) Average Length (Log2 bp) for LCR_Finder and WindowMasker; and (C)Average Pattern Length (Log2 bp) for LCR_Finder and RepeatScout. 3/4 chromosomes were reported with larger span covered by WM, but the average span and average pattern length of LCR_Finder reported were significantly larger than WM and RS.

| Software | Running time (min) | | Number of LCRs/CNVs detected |
|---|---|---|---|
| RS | Chr2 | 59 | 1,146 |
| | Chr5 | 44 | 1,331 |
| | Chr7 | 35 | 1,656 |
| | Chr17 | 18 | 1,477 |
| WM | Chr2 | 55 | 323,086 |
| | Chr5 | 41 | 242,639 |
| | Chr7 | 34 | 207,383 |
| | Chr17 | 17 | 94,783 |
| LCR_Finder | Chr2 | 302 | 59 |
| | Chr5 | 205 | 34 |
| | Chr7 | 180 | 86 |
| | Chr17 | 106 | 44 |

**Table 1.** Time cost of RS, WM and LCR_Finder running on chromosome 2, 5, 7 and 17.

## References

1. A.F.A. Smit, R. Hubley & P. Green. (2011) unpublished data. Current Version: open-3.3.0 ( RMLib: 20110920 )
2. Babcock, M., *et al*. (2007) Hominoid lineage specific amplification of low-copy repeats on 22q11.2 (LCR22s) associated with velo-cardio-facial/digeorge syndrome. *Hum. Mol. Genet.* **16,** 2560-2571.
3. Bailey, J.A., *et al*. (2002) Recent segmental duplications in the human genome. *Science* **297**, 1003-1007.
4. Cheung, V.G., *et al*. (2001) Integration of cytogenetic landmarks into the draft sequence of the human genome. *Nature* **409**, 953-958.
5. Jiang Z. *et al*. (2007) Ancestral reconstruction of segmental duplications reveals punctuated cores of human genome evolution. *Nat Genet.* **39**,1361-1368.
6. Kurotaki, N., *et al*. (2005) Sotos syndrome common deletion is mediated by directly oriented subunits within inverted Sos-REP low-copy repeats. *Hum. Molec. Genet.* **14**, 535-542.
7. Krzywinski, M. *et al*. (2009) Circos: an Information Aesthetic for Comparative Genomics. *Genome Res.* **19**, 1639-1645.
8. Li, H., *et al*. (2008) Mapping short DNA sequencing reads and calling variants using mapping quality scores. *Genome Res*. **18**, 1851-1858.
9. Liu, C.M., *et al*. (2012) SOAP3: Ultra-fast GPU-based parallel alignment tool for short reads. *Bioinformatics* **28**, 878-879.
10. Morgulis, A., *et al*. (2006) WindowMasker: window-based masker for sequenced genomes. *Bioinformatics* **22**, 134-141.
11. Park, S.S., *et al*. (2002) Structure and Evolution of the Smith-Magenis Syndrome Repeat Gene Clusters, SMS-REPs. *Genome Res.* **12,** 729-738.
12. Price, A.L., *et al*. (2005) De novo identification of repeat families in large genomes. *Bioinformatics* **21,** i351-i358.
13. Saunier, S., *et al*. (2000) Characterization of the NPHP1 locus: mutational mechanism involved in deletions in familial juvenile nephronophthisis. *Am. J. Hum. Genet.* **66,** 778-789.
14. Stankiewicz, P. and Lupski, J.R. (2002) Genome architecture, rearrangements and genomic disorders. *Trends Genet.* **18**, 74-82.
15. Valero, M. C., *et al*. (2000) Fine-scale comparative mapping of the human 7q11.23 region and the orthologous region on mouse chromosome 5G: the low-copy repeats that flank the Williams-Beuren syndrome arose at breakpoint sites of an evolutionary inversion(s). *Genomics* **69**, 1-13.
16. Zhang, F., *et al*. (2009) Copy number variation in human health, disease, and evolution. *Annu. Rev. Genomics hum. Genet*. **10**, 451-481.