Bioinformatics, 34(4), 2018, 681–683
doi: 10.1093/bioinformatics/btx665
Advance Access Publication Date: 18 October 2017
Applications Note

Genome analysis

Krait: an ultrafast tool for genome-wide survey of microsatellites and primer design

Lianming Du[1], Chi Zhang[2,*], Qin Liu[3], Xiuyue Zhang[4] and Bisong Yue[4]

[1]Institute for Advanced Study, Chengdu University, Chengdu, Sichuan 610106, China, [2]Rehabilitation Medicine Department, The Affiliated Hospital of Southwest Medical University, Luzhou, Sichuan 646099, China, [3]College of Life Sciences and Food Engineering, Yibin University, Yibin, Sichuan 644000, China and [4]Key Laboratory of Bio-Resources and Eco-Environment, Ministry of Education, College of Life Science, Sichuan University, Chengdu, Sichuan 610065, China

*To whom correspondence should be addressed.
Associate Editor: John Hancock

Received on August 22, 2017; revised on October 2, 2017; editorial decision on October 16, 2017; accepted on October 17, 2017

Abstract

Summary: Microsatellites are found to be related with various diseases and widely used in population genetics as genetic markers. However, it remains a challenge to identify microsatellite from large genome and screen microsatellites for primer design from a huge result dataset. Here, we present Krait, a robust and flexible tool for fast investigation of microsatellites in DNA sequences. Krait is designed to identify all types of perfect or imperfect microsatellites on a whole genomic sequence, and is also applicable to identification of compound microsatellites. Primer3 was seamlessly integrated into Krait so that users can design primer for microsatellite amplification in an efficient way. Additionally, Krait can export microsatellite results in FASTA or GFF3 format for further analysis and generate statistical report as well as plotting.

Availability and implementation: Krait is freely available at https://github.com/lmdu/krait under GPL2 License, implemented in C and Python, and supported on Windows, Linux and Mac operating systems.

Contact: chizhang_swmu@126.com

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

1 Introduction

Microsatellites, also commonly referred to as simple sequence repeats (SSRs) or simple tandem repeats (STRs), are short tandemly repeated DNA sequences of 1–6 bp unit length. Owing to their ubiquitous occurrence and high polymorphism, they have been extensively utilized as powerful molecular markers in genetic mapping, population genetics, DNA forensics and phylogenetics (Ellegren 2004; Vieira *et al.*, 2016). Microsatellite mutations are dynamic processes within tissues and across generations that are implicated in numerous human genetic diseases and play pivotal roles in various regulatory mechanisms and evolution (Kelkar *et al.*, 2011). For example, it has been reported that microsatellites are mainly responsible for morphological evolution in domestic dogs and distinctive social behavior in voles (Merkel and Gemmell, 2008).

With the rapidly increased availability of genomic data, a new in silico approach based on computational tools is widely used to extract microsatellite markers instead of using the cost- and labor-intensive traditional approaches. To date, large numbers of algorithms and bioinformatic software have been developed for microsatellite search and investigation, such as MISA (Thiel *et al.*, 2003), SciRoKo (Kofler *et al.*, 2007), msatcommander (Faircloth, 2008), MSDB (Du *et al.*, 2013) and GMATA (Wang and Wang, 2016). Most of them exhibit excellent performance in genome-wide microsatellite searches, except the web-based tools that only able to deal with a limited range of genomic sequences. Generally, searching for microsatellites within genomes is not only used to study their distribution and abundance, but more importantly, to obtain useful and stable markers for subsequent biological analysis. Unfortunately, animal and plant genomes are so large and contain vast numbers of microsatellites that it remains a challenge to screen for microsatellite markers from huge datasets that meet our requirements for primer design. For example, the human genome contains two million

microsatellite loci (Subramanian *et al.*, 2003). Additionally, all the previously developed search tools have their own proprietary output formats that restrict biologists to carry out the downstream analysis and provide no visualization for microsatellite search results (see Supplementary Table S1 for full comparison).

Here we present Krait, a robust and ultrafast tool with a user-friendly graphic interface for genome-wide investigation of microsatellites, which attempts to overcome the limitations of the currently available tools. Krait allows searches for perfect, imperfect and compound microsatellites and can be easily used to screen microsatellite markers for subsequent primer design.

2 Implementation

Krait is written in Python and can be run as a standalone desktop application on Windows, Linux or Mac systems without dependencies. The microsatellite search engine is written in C and compiled as Python modules for import into Krait.

2.1 Microsatellite search algorithm

We employed brute force search algorithm 2 described by Sokol *et al.* (http://dimacs.rutgers.edu/Publications/Modules/Module09-2/dimacs09-2.pdf) to search perfect microsatellites. Let S be a DNA sequence, S[i] denotes a nucleotide at position i and p denotes the period of the repeat. Thus, for each position i in the sequence, the algorithm first looks for a mono-nucleotide repeat tract ($p=1$) by comparing S[i] with S[$i+p$] and increments of i if they are equal, until no further equation can be found (Supplementary Fig. S1A). If no mono-nucleotide repeat tract is detected, then it looks for a di-nucleotide repeat tract ($p=2$) and so on until searching for a hexa-nucleotide ($p=6$). If an identified repeat tract meets the user-specified minimum repeats, it will be reported in a result table.

For imperfect microsatellite search, the algorithm first looks for a perfect SSR seed, and then the seed is extended in both 5′ and 3′ directions with some edit operations like substitutions and indels. The extension is a process that aligns the original sequence to its expected perfect counterpart by using the dynamic programming algorithm to calculate edit distance (see Supplementary Section S2.2 for details). The extension will be terminated when consecutive edit operations exceed a maximum that is user-specified or extend to the end of the sequence. Once the extension is finished, the optimal alignment will be constructed and the number of substitutions, insertions and deletions are calculated respectively (Supplementary Fig. S1B). Finally, the candidate imperfect microsatellite score is calculated according to a simple equation:

$$\text{Score} = \text{Matches} - \text{Substitutions} * \text{MP} - \text{Indels} * \text{GP}$$

where MP denotes mismatch penalty and GP denotes gap penalty. If the score of the candidate imperfect microsatellite meets the user-specified minimum required score, it will be reported in a result table (see Supplementary Section S2.3 for details).

2.2 Primer design

Krait can design amplification primers for identified microsatellite markers using Primer3 (Untergasser *et al.*, 2012) as its primer design engine. Primer3 was seamlessly integrated into Krait, which is attributed to the primer3-py project (https://github.com/libnano/primer3-py), while currently available tools implement Primer3 from the external environment. In this way, Krait extremely improved the performance of batch primer design for a large number of microsatellites. The parameters used by Primer3 can be specified by the user.

Table 1. Comparison of run time of MISA, MSDB, msatcommander, GMATA, SciRoKo and Krait

Species Genome size	*H. sapiens* (3.25 Gb)	*A. thaliana* (119.67 Mb)	*E. coli* (4.64 Mb)
MISA	30 min 8 s	1 min 36 s	4 s
MSDB	33 min 28 s	1 min 19 s	4 s
msatcommander	31 h 40 min	1 h 31 min	3 min 39 s
GMATA	24 min 33 s	1 min 31 s	3 s
SciRoKo	6 min 1 s	22 s	2 s
Krait	2 min 14 s	8 s	2 s

Note: The genome sequences of *Homo sapiens* (GCF_000001405.37), *Arabidopsis thaliana* (GCF_000001735.3) and *Escherichia coli* (GCF_000005845.2) were downloaded from NCBI database. All tools were run on an Intel(R) Core(TM) i7-6700 CPU @ 3.40 GHz with 16 GB RAM.

The results including primer sequences and detailed information are summarized in a table.

2.3 Input and output

Currently, Krait accepts FASTA formatted files containing any number of sequences as input for microsatellites search. Furthermore, Krait can accept a gzip compressed FASTA file as input that is suitable for large genome analysis. Krait can utilize Gene Transfer Format (GTF) or Generic Feature Format (GFF) formatted annotation file to determine in which region (CDS, UTR, Intron or Exon) the microsatellites are located. Krait supports many output formats. The microsatellite search results can be exported as Comma Separated Values format (CSV), GFF3 or FASTA file. In addition, Krait can generate a PDF formatted statistical report.

3 Run time evaluation

Run time was benchmarked by applying Krait and five other microsatellite detection tools to three genomes for perfect microsatellite identification. The minimum number of repeats were selected as 14, 7, 5, 4, 4, 4 for mono- to hexa-nucleotides, respectively. The benchmark results (Table 1) clearly demonstrate that Krait is the fastest tool for microsatellites detection.

Acknowledgement

We are grateful to Megan Price and Kelei Zhao for their critical reading and English editing on the manuscript.

Funding

This work was supported by the Natural Science Foundation of China [31570534].

Conflict of Interest: none declared.

References

Du,L. *et al.* (2013) MSDB: a user-friendly program for reporting distribution and building databases of microsatellites from genome sequences. *J. Hered.*, **104**, 154–157.

Ellegren,H. (2004) Microsatellites: simple sequences with complex evolution. *Nat. Rev. Genet.*, **5**, 435–445.

Faircloth,B.C. (2008) msatcommander: detection of microsatellite repeat arrays and automated, locus-specific primer design. *Mol. Ecol. Resour.*, **8**, 92–94.

Kelkar,Y.D. *et al*. (2011) A matter of life or death: how microsatellites emerge in and vanish from the human genome. *Genome Res*., **21**, 2038–2048.

Kofler,R. *et al*. (2007) SciRoKo: a new tool for whole genome microsatellite search and investigation. *Bioinformatics*, **23**, 1683–1685.

Merkel,A. and Gemmell,N. (2008) Detecting short tandem repeats from genome data: opening the software black box. *Brief. Bioinform*., **9**, 355–366.

Subramanian,S. *et al*. (2003) Genome-wide analysis of microsatellite repeats in humans: their abundance and density in specific genomic regions. *Genome Res*., **4**, R13.

Thiel,T. *et al*. (2003) Exploiting EST databases for the development and characterization of gene-derived SSR-markers in barley (*Hordeum vulgare* L.). *Theor. Appl. Genet*., **106**, 411–422.

Untergasser,A. *et al*. (2012) Primer3–new capabilities and interfaces. *Nucleic Acids Res*., **40**, e115.

Vieira,M.L. *et al*. (2016) Microsatellite markers: what they mean and why they are so useful. *Genet. Mol. Biol*., **39**, 312–328.

Wang,X. and Wang,L. (2016) GMATA: An Integrated software package for genome-scale SSR mining, marker development and viewing. *Front. Plant Sci*., **7**, 1350.