*Bioinformatics*, 32(15), 2016, 2380–2381
doi: 10.1093/bioinformatics/btw126
Advance Access Publication Date: 26 March 2016
Applications Note

Sequence analysis

# NSimScan: DNA comparison tool with increased speed, sensitivity and accuracy

**Vladimir Novichkov[1], Anna Kaznadzey[2], Natalia Alexandrova[3,*] and Denis Kaznadzey[4]**

[1]Independent Researcher, Bridgewater, NJ, USA, [2]Institute for Information Transmission Problems, RAS, Moscow, Russia, [3]Genome Designs, Inc, Walnut Creek, CA, USA and [4]Thermo-Fisher, South San Francisco, CA, USA

*To whom correspondence should be addressed.

Associate Editor: John Hancock

Received on December 1, 2015; revised on February 8, 2016; accepted on March 1, 2016

## Abstract

**Summary:** Nucleotide Similarity Scanner (NSimScan) is specialized for searching massive DNA databases for distant similarities. Its targeted applications include phylogenomics, comparative and functional studies of non-coding sequences, contamination detection, etc. NSimScan outperforms industry standard tools in combined sensitivity, accuracy and speed, operating at sensitivity similar to BLAST, accuracy of ssearch and speed of MegaBLAST.
**Availability and implementation:** NSimScan is available at https://github.com/abadona/qsimscan as a part of QSimScan package. It is implemented in C++, distributed under MIT license and supported on Linux, OS X and Windows (with cygwin).
**Contact:** dkaznadzey@yahoo.com
**Supplementary information:** Supplementary data are available at *Bioinformatics* online.

## 1 Introduction

The breakthrough in sequencing technologies has brought the challenges of DNA comparison to a new level. Driven mainly by clinical applications, efficient tools for NGS reads mapping have been developed, however, the search for more distant similarities is still handled by sensitive but slower pre-NGS tools, or by modern algorithms that provide higher speed at the expense of sensitivity.

We present NSimScan, a tool for fast, sensitive and accurate DNA similarity search, and demonstrate its performance comparison to industry-standard alignment search tools. NSimScan was recently successfully employed for ANI computing in a large phylogenetic study (Varghese *et al.*, 2015), and we believe that it will be useful for a variety of applications where efficient detection of distant similarities is desired.

## 2 Implementation

### 2.1 Algorithm

NSimScan uses a pipeline of filters of increasing computational complexity similar to PSimScan (Kaznadzey *et al.*, 2013), a twin tool for searching protein databases. Initial hits are selected from the lookup table addressable directly by binary representation of k-meres. The lookup table is constructed on the fly for query sequences, so no pre-indexing is required. Initial hits are used to update diagonal scores. The update algorithm considers the arrangement of neighbor hits on current and adjacent diagonals and may optionally account for k-mere significance and for local sequence redundancy. Actual alignments are computed only for high scoring diagonals. A heuristic based on bit operations on packed sequences is used for extending the alignments, which are evaluated using a secondary filter based on identity percentage. Optionally, tandem repeat/redundancy filtering can be applied. Aligned segments that pass secondary filter may be further combined into alignments with larger gaps. The speed is achieved by query aggregation, use of optimized bitwise operations in alignment computing, and by avoidance of dynamic programming, which makes the entire hit evaluation a linear operation with respect to the alignment lengths.

Main parameters of the algorithm are: size of k-mere; diagonal score threshold (primary diagonal filter); and parameters of the secondary alignment filter: minimal alignment length, minimal identity at minimal length and minimal identity at infinite length. Signal-to-noise ratio can be further improved by providing k-mere frequency table, and by enabling mismatches in k-meres. Detailed algorithm description and discussion of available parameters is provided in the User Manual.

### 2.2 Benchmarking

We used families of bacterial ribosomal protein genes as a ‘gold standard’ data set for benchmarking. From 1244 bacterial genomes of distinct species from the NCBI collection we chose 53 families, each containing over 600 annotated members. From every family, we randomly selected 200 representatives, resulting in a set of 10 600 sequences.

For the test, we compared the ’gold standard’ set to itself. A match detected between sequences of the same family was counted as True Positive (TP), between sequences of distinct families as False Positive (FP), no match between a pair of sequences from the same family as False Negative (FN), and no match between a pair of sequences from distinct families as True Negative (TN). We calculated arrays of cumulative FN/FP/TN/TP, ‘Errors Per Query’ rate as FP/(match count) and ‘Coverage’ as TP/(TP + FN) for the detected matches ordered by e-value (computed by ssearch), and presented them as Errors versus Coverage plots.

We also tested NSimScan speed on a model task of rough phylogenetic profiling of a large metagenomic dataset, searching representative 16S sequences for 749 taxonomic orders from Silva 16S database version 123 (Quast *et al.*, 2013) against cucumber risosphere metagenome sample SRR908208 from NCBI Short Read Archive (Wheeler *et al.*, 2008) containing 67 million 200-bp paired-end sequences.

All tests were performed on a workstation equipped with Intel Core i7-3820 CPU running at 3.60 GHz, 64 Gb of DDR3 RAM and 2 Tb SATA3 hard drive, running Fedora 21 Linu**x OS.**

## 3 Results

Figure 1 demonstrates six distinct cases of NSimScan performance at different primary filtering parameters, compared to industry-standard tools. Strengthening secondary filtering parameters mostly shifts higher coverage part of the curve towards lower error rates, and relaxing adds a segment to the right leaning towards higher error rates.

For all shown conditions, NSimScan demonstrates sensitivity and noise levels similar to ssearch, outperforming all other tested tools. With the most relaxed primary filter, NSimScan operates at sensitivity and speed slightly over USEARCH (which appears to have the best coverage among reference tools), and at almost 100× lower error rate. In terms of signal-to-noise ratio, NSimScan outperforms all tested tools by two orders of magnitude. In terms of speed, it is comparable to MegaBLAST at moderately sensitive settings and to BLAST+ at highly sensitive settings. In terms of coverage, it stays on par with BLAST+ (while running 3× faster) and USEARCH (while running over 10× faster; USEARCH was run with usort disabled: enabling it causes 100x increase in speed and 100× reduction in coverage).

Performance metrics of phylogenetic profiling with NSimScan relative to MegaBLAST is presented in Table 1. NSimScan runs over 10× faster and uses 4× less memory, detecting most of the taxa that MegaBLAST does and a substantial amount that it does not.

## 4 Conclusions

NSimScan is: ***fast***: as fast as the fastest of industry standard tools, such as MegaBLAST; ***sensitive***: more sensitive than any tested tool except for ssearch; ***accurate***: its error rates are the lowest among the tested tools, similar to or below ssearch; and ***exhaustive***: it reports all similarities at a given level. It has a particular advantage when searching for moderately distant similarities (60–90% identity) on massive datasets, making it useful for large-scale comparative analyses of DNA sequences.

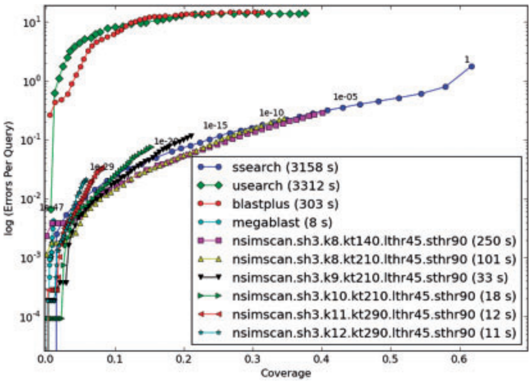

**Fig. 1.** Performance comparison of NSimScan (at different parameter settings) and industry-standard tools. Coverage versus error curves were composed as described in ‘Benchmarking’ for: ssearch (Pearson, 2000), USEARCH (Edgar, 2010), BLAST+ (Camacho *et al.*, 2009), MegaBLAST (Morgulis *et al.*, 2008) and different runs of NSimScan at an increased stringency of primary hit selection. The legend denotes NSimScan parameter settings: *sh* is maximal diagonal shift; *k* is k-mere size; *kt* is diagonal score threshold; *lthr* is the lowest match identity at infinite length; *sthr* is the lowest match identity at minimal length (which was selected as 40 bases). Wall clock times for the runs are given at the end of each legend line. A more complete high resolution version of Figure 1 is provided in Supplementary Materials

**Table 1.** Computing similarities for phylogenetic profiling on a large subject set using NSimScan and MegaBLAST

| Tool | Time | MemUse | Detected | Tx# | MissTx# | ExtraTx# |
|---|---|---|---|---|---|---|
| MegaBLAST | 5 h 18 min | 24.6 Gb | 252956 | 310 | n/a | n/a |
| NSimScan | 26 min | 5.7 Gb | 240934 | 360 | 4.7% | 21.6% |

Parameters used: ‘-k 11, -t 400, -q 4’, where ‘-q’ is the shift between consecutively looked up k-meres. ‘Detected’ is the number of reads identified as 16S fragments; ‘Tx#’ is the number of found distinct taxa; ‘Missing Taxa’ and ‘Extra Taxa’ percentages are with respect to the ones detected by MegaBLAST.

## Acknowledgements

Authors are grateful to Niles Larsen of Danish Genome Institute and Konstantinos Mavrommatis of Celgene for fruitful discussions and feedback.

*Conflict of Interest*: none declared.

## References

Camacho,C. *et al.* (2009) BLAST+: architecture and applications. *BMC Bioinformatics*, **10**, 421.

Edgar,R.C. (2010) Search and clustering orders of magnitude faster than BLAST. *Bioinformatics*, **26**, 2460–2461.

Kaznadzey,A. *et al.* (2013) PSimScan: algorithm and utility for fast protein similarity search. *PLoS One*, **8**, e58505.

Morgulis,A. *et al.* (2008) Database indexing for production MegaBLAST searches. *Bioinformatics*, **24**, 1757–1764.

Pearson,W.R. (2000) Flexible sequence similarity searching with the FASTA3 program package. *Methods Mol. Biol.*, **132**, 185–219.

Quast,C. *et al.* (2013) The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucleic Acids Res.*, **41**, D590–D596.

Varghese,N.J. *et al.* (2015) Microbial species delineation using whole genome sequences. *Nucleic Acids Res.*, **43**, 6761–6771.

Wheeler,D. *et al.* (2008) Database resources of the National Center for Biotechnology Information. *Nucleic Acids Res.*, **36**, D13–D21.