Deterring the Manipulation of N50 Assembly Quality Scoring

Hershil Devnani
Queen's University
Kingston, Ontario, Canada

Andrew Ma
Queen's University
Kingston, Ontario, Canada

Rayan Shaikli
Queen's University
Kingston, Ontario, Canada

ABSTRACT

Genome assembly is becoming a routine procedure along with the advances in next-generation sequencing, it is important to evaluate the methods used to assess the quality of the genome assembly. Significant problems associated with N50, a common assembly quality measure, is the susceptibility of misrepresentation in the cases concerning viral or microbial datasets and misassembly. In response to the problem, we propose a novel metric, UA_{50}, that determines the U50 score for a set of aligned contig blocks rather than the initial set of contigs. The algorithm for NA50 performs sequence alignment of the initial contigs to the provided reference genome. Misassemblies are identified and the corresponding contigs are split into aligned blocks which are aligned to the reference genome. UA_{50} score is computed by calculating the U50 score with the set of aligned blocks. Our results highlight the advantage of UA_{50} over N50 where a large proportion of the contig set is prone to misassembly. Further exploration opportunities on UA_{50} include its effectiveness in scoring of largely overlapping contig sets, and a percentage based UA_{50} metric.

CCS CONCEPTS

• **Applied computing** → **Computational genomics**; **Computational genomics**; **Computational biology**; • **Hardware** → *Biology-related information processing*; • **Theory of computation** → *Bio-inspired optimization*; **Genetic programming**; • **Computing methodologies** → **Genetic algorithms**; **Genetic programming**.

KEYWORDS

genome assembly quality, genomics, computational biology

ACM Reference Format:
Hershil Devnani, Andrew Ma, and Rayan Shaikli. 2021. Deterring the Manipulation of N50 Assembly Quality Scoring. In . ACM, New York, NY, USA, 7 pages.

1 INTRODUCTION

N50 statistics is frequently used for assessing genome assembly quality. It is defined as the shortest contig length at the 50% of the total genome length which is equivalent to the total length of the contigs. Although N50 is a useful indicator of assembly quality, it can be easily manipulated to output a higher value. In this study, we attempt to address the problems associated with using the N50 statistics. One of the problems using N50 is the misassembly problem shown in Figure 1. In the genome assembly of the reference genome (Figure 1), three correctly aligning contigs can result from the assembly (Figure 2), each having an equal unit length of $1u$. In this set of three contigs, the N50 is $1u$. However, the three contigs can be joined incorrectly in respect to the reference genome to manipulate the N50 score, resulting in a longer contig with a length of $3u$ (Figure 3). These incorrect joints of contigs are misassemblies which can be a result of poor genome assembly or the use of an incorrect algorithm. This contig set containing multiple misassemblies results in an overinflated N50 score since the contig is three times the size of the correct contigs. The misassembly problem highlights the failure of N50 to evaluate the correctness of the assembly and the susceptibility to overinflation given a contig set containing high proportions of incorrectly assembled contigs.

Figure 1: Hypothetical reference genome.

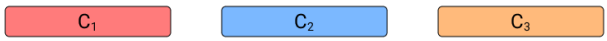


Figure 2: Correctly assembled contigs (C_1, C_2, C_3) with a length of 1u each. The N50 for this contig set is $1u$.

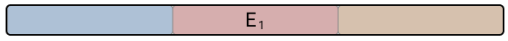


Figure 3: Incorrectly assembled contig (E_1), resulting from erroneous joints between C_1, C_2, C_3. Total length of E_1 is $3u$.

The overlapping contigs problem is the nature of N50 to assign a high score to a set of largely overlapping contigs. For example, given a reference genome (Figure 4), a possible set of non-overlapping contigs is shown in (Figure 5), and another set of contigs with overlap (Figure 6). The contig set with high proportion of contigs with large overlaps have an incomplete coverage of the reference genome. Compared to another set of shorter contigs containing less overlaps and more coverage results in a lower N50 value compared to the contig set with many overlaps between contigs. The overlapping contigs problem highlights the failure of N50 to evaluate the coverage of the assembly and the susceptibility to overinflation given a contig set containing high proportions of overlapping contigs. The problems associated with N50 statistics highlight the need

Queen's University - CISC471 2021, April 23, 2021, Kingston, ON, Canada

for an alternative assembly quality metric that is able to evaluate the correctness and the coverage of the assembly.

Figure 4: Hypothetical reference genome.

Figure 5: Non-overlapping contigs (C_1, C_2, C_3, C_4), with lengths of $1u$ with complete coverage of reference genome.

Figure 6: Overlapping Contigs (O_1, O_2) with a length of $2u$ with incomplete coverage of reference genome.

In response to the problem, we developed a novel metric called UA_{50}. The UA_{50} performs sequence alignment of the initial set of contigs to the reference genome. When erroneous joints are identified within a contig, it is separated into alignment fragments where each fragment is aligned to the reference genome. This process corrects any misassemblies in the initial contig set and produces a corrected set of contigs. Only the aligned contigs are stored and sorted by their lengths from the longest to the shortest. From the aligned contig set, unique regions of the contigs are identified using their coordinates in respect to the reference genome. Only the unique segments of the contigs are stored, removing any existing overlaps between the contigs. The unique contig set is sorted in the same manner as the aligned contig set. The cutoff for UA_{50} is the summation of contig lengths multiplied by the threshold percentage. For example, the percentage of UA_{50} is 50%. The UA_{50} score of the assembly is the length of the shortest unique contig at the first instance where the running sum becomes greater than or equal to the UA_{50} cutoff.

The two problems associated with N50 the overinflation on misassemblies and the large overlappings between the contigs. Our UA_{50} metric measures the length of the contig which covers 50% of a set of aligned, unique lengths of the contigs. The sequence alignment of the initial contigs to the reference genome identifies the misassemblies present in the contigs and splits any erroneous joints into distinct contig blocks, producing an aligned set of contigs. The alignment processing of initial contigs removes the possibility of overinflated score on misassembled contigs. The aligned contig set is mapped to the reference genome to generate a unique set of contigs where the overlapping portions of each contig is removed. Using a unique contig set removes the possibility of the overinflation on a contig set with high proportion of large overlaps.

2 RELATED WORK

N50 was first introduced as an alternative measure for contiguity by Lander et. al [2] in the paper regarding the initial sequencing and analysis of the human genome as a part of the Human Genome Project. N50 was defined as the largest length L such that 50% of all nucleotides are contained in contigs of size at least L. The N50 statistic was developed in response to the problems of using the average length of a contig and the length-weighted average length. Average length metric was prone to deflation by a small proportion of the genome consisting of multiple short contigs whereas the length-weighted average length was prone to inflation by large segments of finished sequence. NG50 is based on the N50 statistics that computes the smallest contig length where 50% of the reference genome is contained in contigs of size NG50 or larger [1]. Despite its advantages, N50 is susceptible to inaccurate results [4]. A poor assembly can lead to erroneous joints between unrelated contigs and produce large missassembled contigs, resulting in an inflated N50 score. Another disadvantage of N50 is its inaccuracy when measuring microbial or viral datasets since most of the reads are noise, thus skewing the N50 score [3]. Lastly, N50 does not factor the uniqueness of the contigs which can lead to inflation of N50 given a largely overlapping contig set.

U50 and related metrics were developed to address the problems associated with the N50 statistic. U50 is the length of the smallest contig such that 50% of the sum of all unique, target-specific contigs is contained in contigs of size U50 or larger. It is calculated by sorting the contigs by their lengths in decreasing order. All contigs are mapped to the reference genome, preserving the unique portions of each contig and removing non-unique regions of each contig. The unique portions of contigs are sorted in decreasing lengths. From the set of unique contigs, the cutoff is defined as the summation of all contigs multiplied by the threshold percentage. A cumulative sum of unique contigs is compared against the threshold percentage. The contig length at the first instance where the cumulative sum is greater or equal to the cutoff is the U50 score. UL50 describes the number of contigs at the first instance where cumulative sum is greater or equal to the cutoff. UG50% is the percentage score of U50. Compared to N50, U50 metric produced a more accurate measure of assembly given sequences containing high noise, such as viral and microbial sequencing. Limitations of U50 include the requirement of a reference genome and lack of consideration for genome coverage due to the removal of overlapping regions.

3 APPROACH

The approach taken to develop our Unique Assembly metric involved taking a look at the limitation of the current NXX scoring metric and UXX scoring metric.

We found that the NXX algorithm is one of the most widely used and regarded algorithm used to base contig assembly quality against. A greater NXX score relative to the length of the genome generally indicates a better assembly and returns the length of the contig at the XX percent of the total genome assembly length. However, some limitations we observed with NXX assembly scoring algorithm included the misleading nature of the metric in terms of how it can be very easily, and is often times, manipulated to inflate

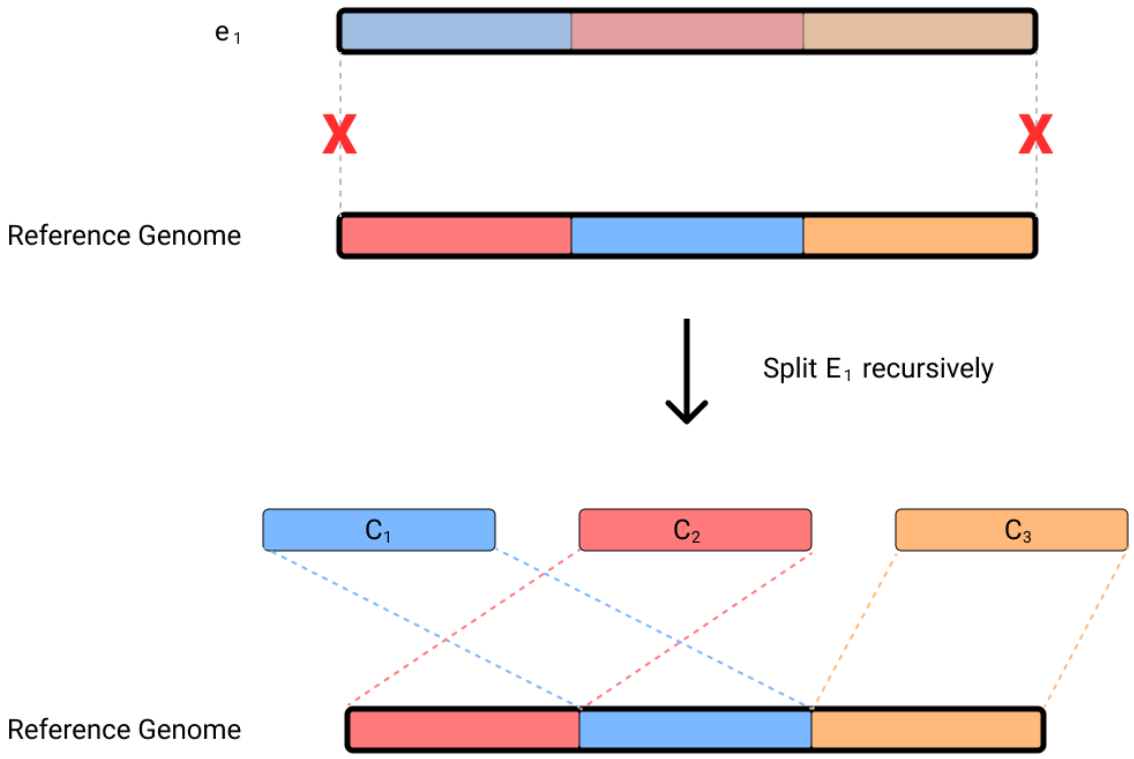


Figure 7: E_1 does not completely align with the reference genome. e_1 is recursively split into three new contigs (C_1, C_2, C_3, C_4) that completely align to the reference genome.

assembly scores by means of filtering out short contigs and with the introduction of misassembly biases.

The most prominent factor that causes the NXX scoring algorithm to be rendered misleading is the use of filters. Many filtering techniques can be used to filter out small length contigs in the assembly data set, which inherently skews the NXX results by increasing the likelihood of smallest contig size making up 50% of the genome assembly to be larger in length than would be without the filtering. Secondly, using a scoring algorithm that incorrectly joins contigs together during assembly, by unjustifiably algorithmically joining certain contigs together to increase the average length of contigs in the data set, which inadvertently construes a greater NXX score. This leads to a positive feedback generated by manipulating the underlying data sets and the resulting inflation in scores due to skewness in proportionality of contig lengths.

These issues regarding inflated assembly quality scores defeat the purpose of the metric in the first place, to assess the quality of the genome assembly given a set of contigs. If the NXX metric can be manipulated easily, it takes away from the efficacy of using the metric as a basis to conduct further analysis and quantify the integrity of an underlying data set. In order to combat these issues, we decided to take a look at these limitations. First, we took a look at properly assembly, since NXX lines up contigs sequentially, not considering unique contig matches to the genome. Second, we wanted to develop a method that discourages filters or manipulation of joining contigs together to inflate assembly quality scores. While the U50 metric addresses the uniqueness of contig assembly against a reference genome, it still does not deter from manipulating the data set to inflate the score since filtering and joining contigs will still inflate this score.

And so, to combat this, out approach was to mitigate these artificial inflations by introducing a new metric based on the both the NXX and UXX metric, that will enhance the raw estimated assembly score given a set of contigs. The idea was to create a new metric that can be used to compare the assembly of different sets of contigs against a reference genome that is deterrent to inflated scoring caused by filtering, and being a subsequent extension of the UXX metric, acts a baseline for sequence assembly for related organisms.

Our new Unique Assembly, UA_{50}, metric provides a metric to understand the assembly quality of a raw set of data. Our approach was to enable an accurate quantifiable score that determines assembly quality based on the unique matches to contigs against a reference genome. Our algorithm matches the unique contigs matches against a genome using a masking array, in which once a contigs matches a portion of the genome, we do not double count the length of subsequent contigs that may overlap in the same portion of the genome. This provides a deterrent to filtering our short

contigs or joining contigs together since they will neither promote nor detract from the metrics scoring analysis. We chose to use a reference genome in our scoring metric since we found the NXX score to arbitrary, so a basis of reference genome will provide a better indicator genome assembly of related organisms.

Overall, the key approach to our metric as to develop something that discourages skewing or filtering data for falsely better N50 scores when there is no basis for the score to increase [5].

3.1 Misassembly Correction

The process of identifying and correcting misassemblies in the initial contig set involves the alignment of the contig set to the given reference genome. Here, we assume that all of the correct contigs are completely aligned to the reference genome, meaning the contig is a subsequence of the reference genome. Each contig is checked and added to a list of aligned contigs if it is completely aligned. A misassembly is defined as the incorrect joint between contigs, in other words, the misassembled contig is not a subsequence of the reference genome. When a misassembled contig is identified, it is recursively split into individual contigs until each of the distinct contigs align to the reference genome (Figure 7). The resulting contigs and their coordinates in respect to the reference genome are then added to the list of aligned contigs. Using this process removes all the erroneous joints in the misassembled contigs and produces a set of correct contigs from the misassembled contigs.

3.2 UA_{50} Calculation

Step 1: The first step for calculating UA_{50} involves performing sequence alignment on the initial set of contigs (c) to the reference genome. Contigs that do not completely align with the reference genome are considered misassemblies (e), where misassemblies are composed of one or more incorrect joint between contigs that are aligned with the reference genome. Any contig with misalignment in the initial contig set is recursively split into equal length contig blocks which are aligned with the reference genome. The new contig blocks and the aligned contigs (a) are added to the aligned contig set (A).

$$e = c_1 + ... + c_n \tag{1}$$

$$A = a_1 + a_2 + ... + a_n \tag{2}$$

Step 2: The aligned contigs are sorted by their lengths from the longest ($a1$) to the shortest (an).

$$sort(A) = a_1 + a_2 + ... + a_n \tag{3}$$

Step 3: A mask array is created which has the same length as the reference genome. Initially, the mask array consists of only 0 which indicates the unmapped region of the reference genome. The values are replaced by 1 when a region of a contig maps to the reference genome. The mapping of the contigs to the reference genome is accomplished by comparing the contig coordinates in respect to the reference genome.

Step 4: Using the mask array, each of the aligned contigs are mapped to the reference genome. A running sum is used to track the number of unique mapping of the contig. This sum represents the length of the unique segments of each of the aligned contigs. Only the unique segments of each aligned contig are stored (a') in the unique contig set (U), disregarding the overlapping portions of the aligned contigs. The unique contigs are sorted by their lengths from the longest (a'_1) to the shortest (a'_n).

$$sort(U) = a'_1 + a'_2 + ... + a'_n \tag{4}$$

Step 5: The cutoff for UA_{50} is given by the summation of contig lengths multiplied by the threshold percentage. For UA_{50}, the threshold percentage is 50%.

$$UA_{50}cutoff = (\sum_{k=1}^{n} a'_k) \times 50\% \tag{5}$$

Step 5: The running sum is calculated by adding the lengths in the sorted unique contig set in a sequential order. The UA_{50} score is the length of the shortest contig at the first instance where the running sum is greater or equal to the UA_{50} cutoff.

$$UA_{50} = a'_k, where(\sum_{k=1}^{n} a'_k) \geq UA_{50}cutoff \tag{6}$$

3.3 Testing

The first step was to baseline the results by understating how data sets respond to change in N50, L50, U50 and UG50% metrics. Comparing the N50 and U50 data, we see as the data sets are introduced with a greater proportion of larger contigs, the scores increase. Next, we took a look at how simulated filtering and assembly manipulation by joining shorter contigs together would affect the efficacy of the N50 and U50 score.

In order to generate the sample data used to conduct our experiment, we develop a function that would take in the following parameters:

- Proportion of small to medium to large length contigs in the data set
- The size of the data set or the number of contigs present in the data set
- The reference genome to base contigs off in an attempt to simulate the assembly data set for related organisms (introducing 5% error to random contigs)

In order to be able to test our metrics resistance to change when manipulating the data sets, we added an options parameter:

- The amount of simulated filtering and contig joining to a percentage of the entire data set

These control variables in our sample data generator allowed us to build data sets that were used to baseline our understanding of the different scoring algorithms and enable us to test our proposed advantages by introducing the same manipulations usually done to inflate NXX and UXX scores. Our data generator used parameters for the size of contigs based on our experience with de novo assembly the proportions of contig lengths generally considered to be small (36bp-150bp), medium (150bp-250bp) and large (250bp-500bp). These parameters were set as global variables and can be adjusted in the code very easily.

4 RESULTS AND DISCUSSION

We constructed two experiments assessing the performance of the UA50 metric using theoretical contig data sets. One reference

Table 1: Comparative Analysis of Assembly Scoring Metrics based on Contig Data Set Features (Average of 5 Trials)

Metric	Control	SmallSkew	MediumSkew	LargeSkew
N50	285.0	154.0	217.0	372.0
UA50	498.0	499.0	499.0	499.0
U50	498.0	499.0	499.0	499.0
UG50pct	99.6	99.8	99.8	99.8
L50	318.0	277.0	421.0	407.0

Table 2: Comparative Analysis of Composition of Contigs for Trial Data

ContigParam	Control	SmallSkew	MediumSkew	LargeSkew
NUM	1125.0	1126.0	1126.0	1126.0
SMPCT	33.511	75.222	13.144	12.522
MDPCT	33.333	12.345	74.334	12.877
LGPCT	33.156	12.433	12.522	74.6

Table 3: Comparative Analysis of N50 vs UA50 Given Percentage of Induced Error

Metric	5.0	10.0	25.0	50.0
N50	274.0	292.0	329.0	360.0
UA50	499.0	499.0	499.0	499.0

Table 4: Comparative Analysis of Composition of Contigs for N50 vs UA50 Experiment Given Percentage of Induced Error

ErrorPCT	NUM	SmallPCT	MediumPCT	LargePCT
5.0	1180.0	48.051	25.847	25.847
10.0	1236.0	45.874	26.861	26.861
25.0	1405.0	41.21	28.968	28.968
50.0	1686.0	35.409	32.562	32.562

genome was used to generate the data sets per experiment. Total of seven theoretical contig data sets were generated. Three were used in the first comparative analysis concerning the scoring metrics given contig sets with varying proportions of short, medium, and large contig lengths, the proportions are indicated in Table 2. Four contig sets were generated for the comparative analysis between N50 and UA50 with different percentages of induced error, the percentages of error are indicated in Table 4.

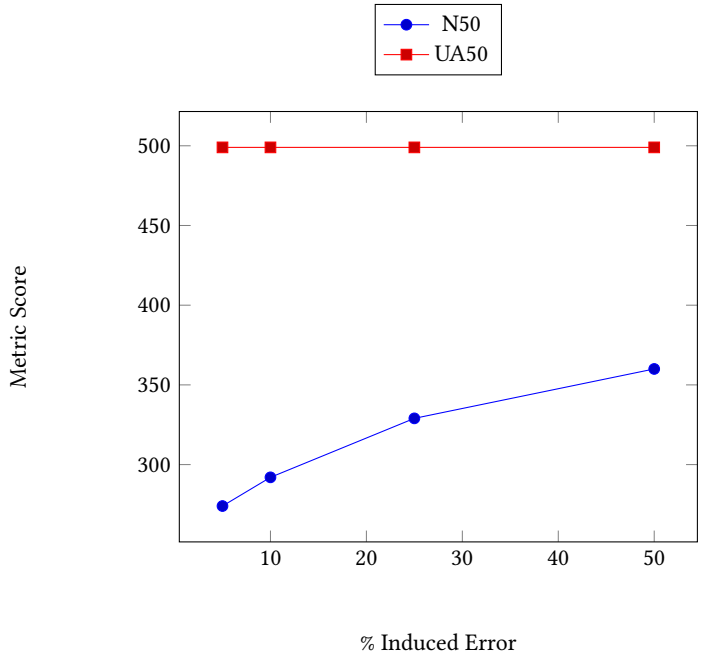
The first experiment, shown 1, involves the comparative analysis of the UA50 metric to pre existing metrics on varying contig length composition in the theoretical data set, without any misassemblies. The UA50 is consistent in all three data sets. This behaviour is equivalent to the U50 results, having the same score for the different proportions of contig lengths. Similarly, UG50% is consistent in the three theoretical contig sets. Conversely, the N50 appears to be increasing as the proportions of large contigs increase in the data sets. The L50 follows the same suit as the N50 where its value increases with increase of the proportions of large contigs in the data sets.

The second comparative analysis, shown in Table 3, involves observing the behaviours of N50 and UA50 when misassemblies are induced as the erroneous joints of contigs. The N50 score increased as the percentage of induced error increased whereas UA50 remained resilient to any amount of induced error and computed a consistent score for all four contig data sets.

5 CONCLUSION

The title of your work should use capital letters appropriately - https://capitalizemytitle.com/ has useful rules for capitalization. Use the `title` command to define the title of your work. If your work has a subtitle, define it with the `subtitle` command. Do not insert line breaks in your title.

Figure 8: Comparative Analysis of Composition of Contigs for N50 vs UA50 Experiment Given Percentage of Induced Error



If your title is lengthy, you must define a short version to be used in the page headers, to prevent overlapping text. The `title` command has a "short title" parameter:

```
\title[short title]{full title}
```

6 CCS CONCEPTS AND USER-DEFINED KEYWORDS

Two elements of the "acmart" document class provide powerful taxonomic tools for you to help readers find your work in an online search.

The ACM Computing Classification System — https://www.acm.org/publications/class-2012 — is a set of classifiers and concepts that describe the computing discipline. Authors can select entries from this classification system, via https://dl.acm.org/ccs/ccs.cfm, and generate the commands to be included in the LaTeX source.

User-defined keywords are a comma-separated list of words and phrases of the authors' choosing, providing a more flexible way of describing the research being presented.

CCS concepts and user-defined keywords are required for for all articles over two pages in length, and are optional for one- and two-page articles (or abstracts).

ACKNOWLEDGMENTS

To Dr. Benjamin T. Cecchetto for the guidance throughout the CISC 471 Computational Biology course at Queen's University and laying the framework for this research project.

REFERENCES

[1] Dent Earl, Keith Bradnam, John St John, Aaron Darling, Dawei Lin, Joseph Fass, Hung On Ken Yu, Vince Buffalo, Daniel R. Zerbino, Mark Diekhans, Ngan Nguyen, Pramila Nuwantha Ariyaratne, Wing-Kin Sung, Zemin Ning, Matthias Haimel, Jared T. Simpson, Nuno A. Fonseca, İnanç Birol, T. Roderick Docking, Isaac Y. Ho, Daniel S. Rokhsar, Rayan Chikhi, Dominique Lavenier, Guillaume Chapuis, Delphine Naquin, Nicolas Maillet, Michael C. Schatz, David R. Kelley, Adam M. Phillippy, Sergey Koren, Shiaw-Pyng Yang, Wei Wu, Wen-Chi Chou, Anuj Srivastava, Timothy I. Shaw, J. Graham Ruby, Peter Skewes-Cox, Miguel Betegon, Michelle T. Dimon, Victor Solovyev, Igor Seledtsov, Petr Kosarev, Denis Vorobyev, Ricardo Ramirez-Gonzalez, Richard Leggett, Dan MacLean, Fangfang Xia, Ruibang Luo, Zhenyu Li, Yinlong Xie, Binghang Liu, Sante Gnerre, Iain MacCallum, Dariusz Przybylski, Filipe J. Ribeiro, Shuangye Yin, Ted Sharpe, Giles Hall, Paul J. Kersey, Richard Durbin, Shaun D. Jackman, Jarrod A. Chapman, Xiaoqiu Huang, Joseph L. DeRisi, Mario Caccamo, Yingrui Li, David B. Jaffe, Richard E. Green, David Haussler, Ian Korf, and Benedict Paten. 2011. Assemblathon 1: a competitive assessment of de novo short read assembly methods. *Genome Research* 21, 12 (Dec. 2011), 2224–2241. https://doi.org/10.1101/gr.126599.111

[2] E. S. Lander, L. M. Linton, B. Birren, C. Nusbaum, M. C. Zody, J. Baldwin, K. Devon, K. Dewar, M. Doyle, W. FitzHugh, R. Funke, D. Gage, K. Harris, A. Heaford, J. Howland, L. Kann, J. Lehoczky, R. LeVine, P. McEwan, K. McKernan, J. Meldrim, J. P. Mesirov, C. Miranda, W. Morris, J. Naylor, C. Raymond, M. Rosetti, R. Santos, A. Sheridan, C. Sougnez, Y. Stange-Thomann, N. Stojanovic, A. Subramanian, D. Wyman, J. Rogers, J. Sulston, R. Ainscough, S. Beck, D. Bentley, J. Burton, C. Clee, N. Carter, A. Coulson, R. Deadman, P. Deloukas, A. Dunham, I. Dunham, R. Durbin, L. French, D. Grafham, S. Gregory, T. Hubbard, S. Humphray, A. Hunt, M. Jones, C. Lloyd, A. McMurray, L. Matthews, S. Mercer, S. Milne, J. C. Mullikin, A. Mungall, R. Plumb, M. Ross, R. Shownkeen, S. Sims, R. H. Waterston, R. K. Wilson, L. W. Hillier, J. D. McPherson, M. A. Marra, E. R. Mardis, L. A. Fulton, A. T. Chinwalla, K. H. Pepin, W. R. Gish, S. L. Chissoe, M. C. Wendl, K. D. Delehaunty, T. L. Miner, A. Delehaunty, J. B. Kramer, L. L. Cook, R. S. Fulton, D. L. Johnson, P. J. Minx, S. W. Clifton, T. Hawkins, E. Branscomb, P. Predki, P. Richardson, S. Wenning, T. Slezak, N. Doggett, J. F. Cheng, A. Olsen, S. Lucas, C. Elkin, E. Uberbacher, M. Frazier, R. A. Gibbs, D. M. Muzny, S. E. Scherer, J. B. Bouck, E. J. Sodergren, K. C. Worley, C. M. Rives, J. H. Gorrell, M. L. Metzker, S. L. Naylor, R. S. Kucherlapati, D. L. Nelson, G. M. Weinstock, Y. Sakaki, A. Fujiyama, M. Hattori, T. Yada, A. Toyoda, T. Itoh, C. Kawagoe, H. Watanabe, Y. Totoki, T. Taylor, J. Weissenbach, R. Heilig, W. Saurin, F. Artiguenave, P. Brottier, T. Bruls, E. Pelletier, C. Robert, P. Wincker, D. R. Smith, L. Doucette-Stamm, M. Rubenfield, K. Weinstock, H. M. Lee, J. Dubois, A. Rosenthal, M. Platzer, G. Nyakatura, S. Taudien, A. Rump, H. Yang, J. Yu, J. Wang, G. Huang, J. Gu, L. Hood, L. Rowen, A. Madan, S. Qin, R. W. Davis, N. A. Federspiel, A. P. Abola, M. J. Proctor, R. M. Myers, J. Schmutz, M. Dickson, J. Grimwood, D. R. Cox, M. V. Olson, R. Kaul, C. Raymond, N. Shimizu, K. Kawasaki, S. Minoshima, G. A. Evans, M. Athanasiou, R. Schultz, B. A. Roe, F. Chen, H. Pan, J. Ramser, H. Lehrach, R. Reinhardt, W. R. McCombie, M. de la Bastide, N. Dedhia, H. Blöcker, K. Hornischer, G. Nordsiek, R. Agarwala, L. Aravind, J. A. Bailey, A. Bateman, S. Batzoglou, E.

Birney, P. Bork, D. G. Brown, C. B. Burge, L. Cerutti, H. C. Chen, D. Church, M. Clamp, R. R. Copley, T. Doerks, S. R. Eddy, E. E. Eichler, T. S. Furey, J. Galagan, J. G. Gilbert, C. Harmon, Y. Hayashizaki, D. Haussler, H. Hermjakob, K. Hokamp, W. Jang, L. S. Johnson, T. A. Jones, S. Kasif, A. Kaspryzk, S. Kennedy, W. J. Kent, P. Kitts, E. V. Koonin, I. Korf, D. Kulp, D. Lancet, T. M. Lowe, A. McLysaght, T. Mikkelsen, J. V. Moran, N. Mulder, V. J. Pollara, C. P. Ponting, G. Schuler, J. Schultz, G. Slater, A. F. Smit, E. Stupka, J. Szustakowki, D. Thierry-Mieg, J. Thierry-Mieg, L. Wagner, J. Wallis, R. Wheeler, A. Williams, Y. I. Wolf, K. H. Wolfe, S. P. Yang, R. F. Yeh, F. Collins, M. S. Guyer, J. Peterson, A. Felsenfeld, K. A. Wetterstrand, A. Patrinos, M. J. Morgan, P. de Jong, J. J. Catanese, K. Osoegawa, H. Shizuya, S. Choi, Y. J. Chen, J. Szustakowki, and International Human Genome Sequencing Consortium. 2001. Initial sequencing and analysis of the human genome. *Nature* 409, 6822 (Feb. 2001), 860–921. https://doi.org/10.1038/35057062

[3] Samia N. Naccache, Scot Federman, Narayanan Veeraraghavan, Matei Zaharia, Deanna Lee, Erik Samayoa, Jerome Bouquet, Alexander L. Greninger, Ka-Cheung Luk, Barryett Enge, Debra A. Wadford, Sharon L. Messenger, Gillian L. Genrich, Kristen Pellegrino, Gilda Grard, Eric Leroy, Bradley S. Schneider, Joseph N. Fair, Miguel A. Martínez, Pavel Isa, John A. Crump, Joseph L. DeRisi, Taylor Sittler, John Hackett, Steve Miller, and Charles Y. Chiu. 2014. A cloud-compatible bioinformatics pipeline for ultrarapid pathogen identification from next-generation sequencing of clinical samples. *Genome Research* 24, 7 (July 2014), 1180–1192. https://doi.org/10.1101/gr.171934.113

[4] Derrick Scott. 2014. Utilizing Next Generation Sequencing to Generate Bacterial Genomic Sequences for Evolutionary Analysis. *Theses and Dissertations* (Aug. 2014). https://scholarcommons.sc.edu/etd/2887

[5] Adam Thrash, Federico Hoffmann, and Andy Perkins. 2020. Toward a more holistic method of genome assembly assessment. *BMC Bioinformatics* 21, 4 (July 2020), 249. https://doi.org/10.1186/s12859-020-3382-4