Calculating LOD score: experimental comparison

Natalia Flerova

September 17, 2010

1 Introduction

The purpose of this experimental work is to compare the performance of three programs capable of calculating LOD score: Morgan, Superlink and SampleSearch [1]. SampleSearch is a general purpose algorithm for finding probability of evidence in a Bayesian network while MOrgan and Superlink are specialized programs aimed at estimating LOD score. Superlink is an exact scheme while Morgan and SampleSearch are both approximate anytime algorithms that use sampling.

LOD score stands for logarithm of the odds. It is a statistical estimate of whether two loci (the sites of genes) are likely to lie near each other on a chromosome and are therefore likely to be inherited together as a package [2]. By definition, LOD score is:

$$LODscore = \log \frac{P_{sequence\ with\ a\ given\ linkage\ value}}{P_{sequence\ with\ no\ linkage}} \tag{1}$$

Practically, it can be calculated as a difference of log-likelihood assuming certain linkage distance (which will be referred to as a position of the trait locus) and log-likelihood assuming no linkage.

Morgan and Superlink perform this calculation internally and output LOD scores directly, while SampleSearch, being a general purpose algorithm for estimating probability of evidence, returns log-likelihood of a given Bayesian network. Calculating LOD score is done as a post-processing using a separate script. However, the time required for this operation is negligibly small and thus is not taken in consideration.

Experiments were run on instances known as *type 4 pedigrees*, taken from the Superlink official site (`http://bioinfo.cs.technion.ac.il/superlink/ResultTables_Input/Families_Type4Time`). Because of the comparatively large family size those instances are known to be hard, even

though they have few loci. The task is to find the most likely position of the trait locus on the chromosome by changing the position of the disease locus against a fixed map of markers and calculating LOD score corresponding to the given position. The highest LOD score corresponds to the most likely trait locus position. The disease locus is moved within all intervals included in the specified region of the map. The positions of the disease locus throughout the scan are determined by defining the number of equally spaced positions between each two adjacent markers. Original test instances fix the number of positions of trait locus between each two markers to three, however we change this number to one, two and even zero for certain instances, in order to analyze how well the programs scale.

In this report we refer to the instances using notation a_b_c, where a is the number of markers, b is the number of people in the pedigree and c is the number of positions of trait locus between two adjacent markers. The number of distinct LOD scores calculated for each instance is $a \cdot c$ for $c > 0$ and 1 for $c = 0$. In the cases, where the performance on variable number of trait locus positions is analyzed, test instances are notated only by the first two components: a_b. The original files are in Superlink file format. The conversion to Morgan was performed using the script provided by Elizabeth Thompsons group. Conversion to the UAI file format (input format for SampleSearch) was done by BayesNetGenerator version 1.7 with and without using optimization options. We used the following versions of the programs: Superlink v1.7, Morgan v2.9 release 2, SampleSearch v. uai2010.

Since Superlink is an exact algorithm, we treat its results, where available, as a ground truth, in order to evaluate the accuracy of the approximation schemes.

2 Experimental results

2.1 Accuracy of caluculation as a function of cut off time

In this section we explore the dependencies between the accuracy of maximum LOD score estimation and the cut off time and the impact of optimization during conversion on the performance of SampleSearch.

Some notes on the representation of the results are needed. On the plots portraying the dependence of the results on the time, Superlink results (where available) are presented by a line starting at the time it took Superlink to solve the instance. Throughout report the log-likelihood has natural base. Morgan does not output log-likelihood and thus for any instance LOD score and trait locus position are the only available results for this program. Superlink did not run on any of the instances with a single trait position. For SampleSearch

the plots present agregate cut off time, which is the product of individual cut off times per network and the number of networks, that corresponds to the number of trait positions.

Easy instance 100_5_1 (optimization was always used for conversion) Morgan was not applied to this instance.

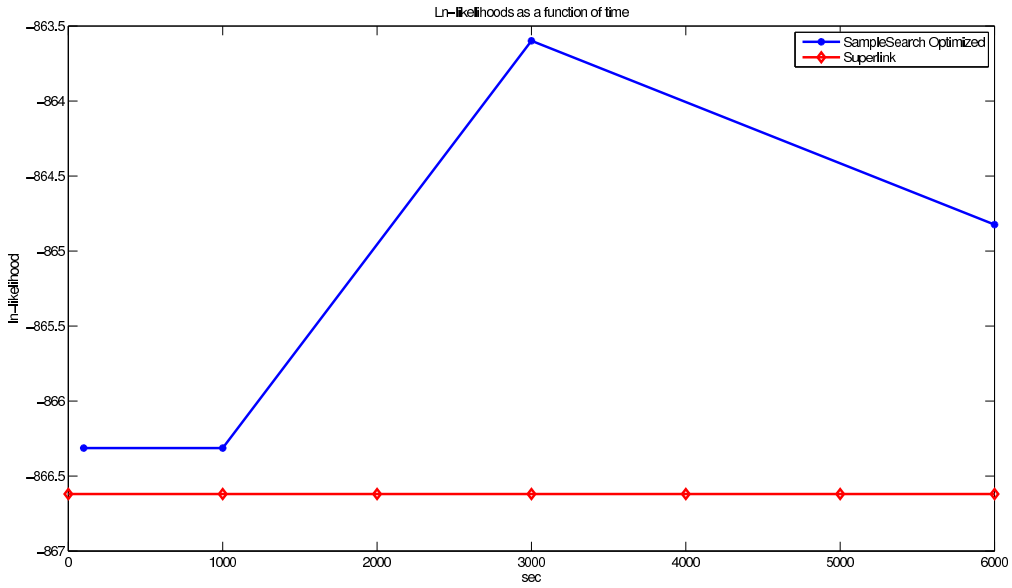

Figure 1: 100_5_1: log-likelihood as a function of the cut off time

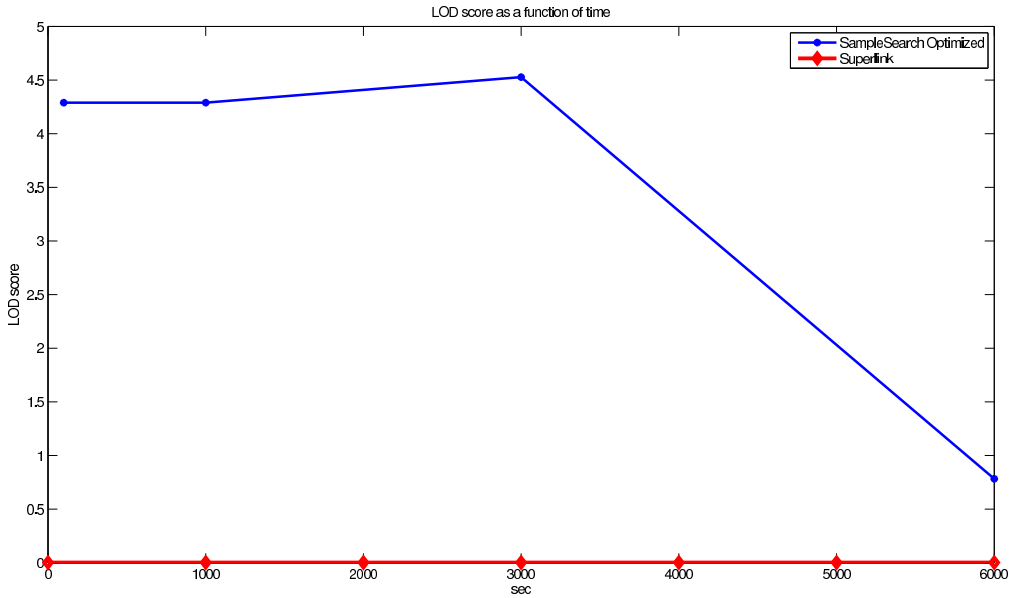

Figure 2: 100_5_1: LOD score as a function of the cut off time

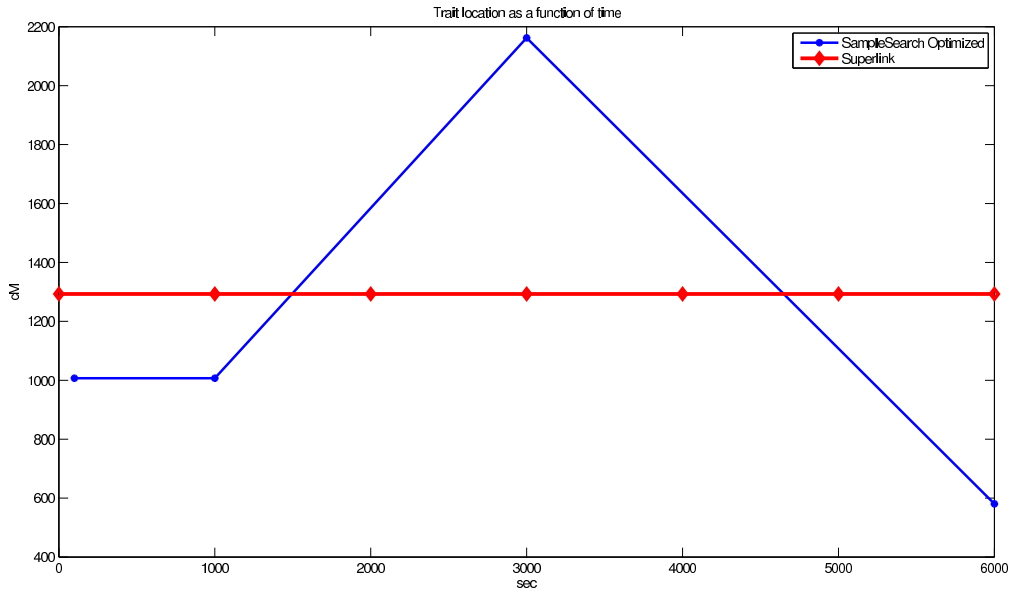

Figure 3: 100_5_1: trait locations as a function of the cut off time

It can be seen that even for an easy instance SampleSearch has troubles determining the correct value of the LOD score. The trait locations found by SampleSearch have very large variance both on this instance and majority of the others.

Instance 110_22_0 (with and without optimization during conversion) As mentioned above, Superlink does not run on the instances that only consider a single trai locus position. Morgan does not output log-likelihood.

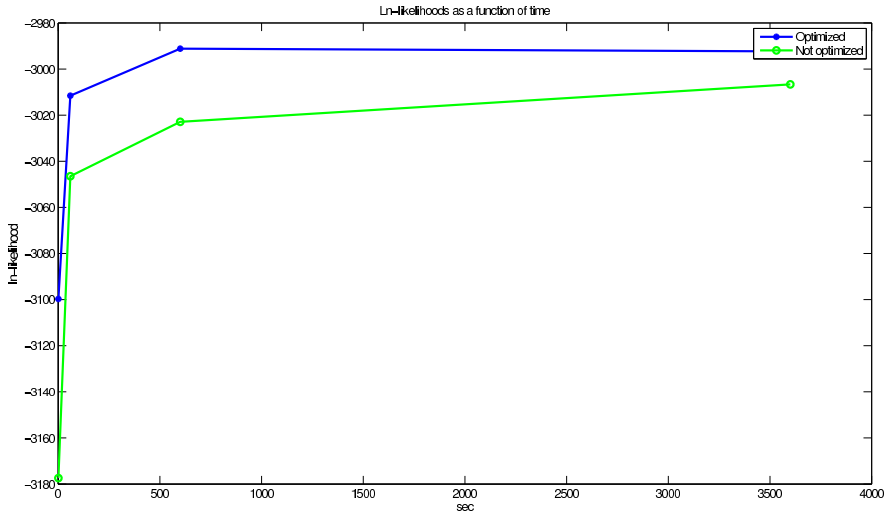

Figure 4: 110_22_0: log-likelihood as a function of the cut off time

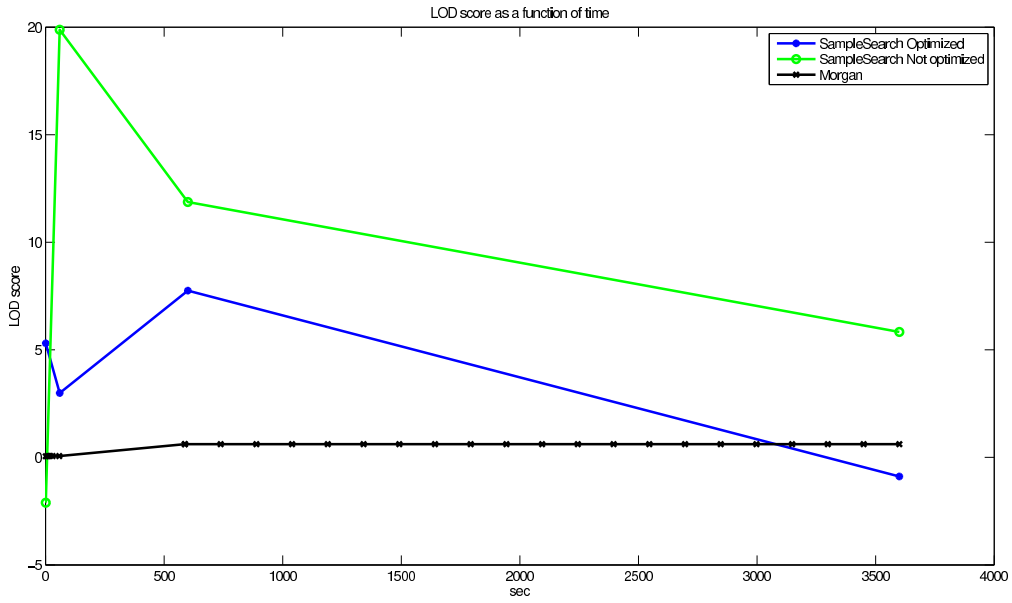

Figure 5: 110_22_0: LOD score as a function of the cut off time

The initial results for SampleSearch are inaccurate, however with the cut off time of 600s the LOD score gets close to the output of Morgan. The optimization during conversion process help increase the accuracy of the results.

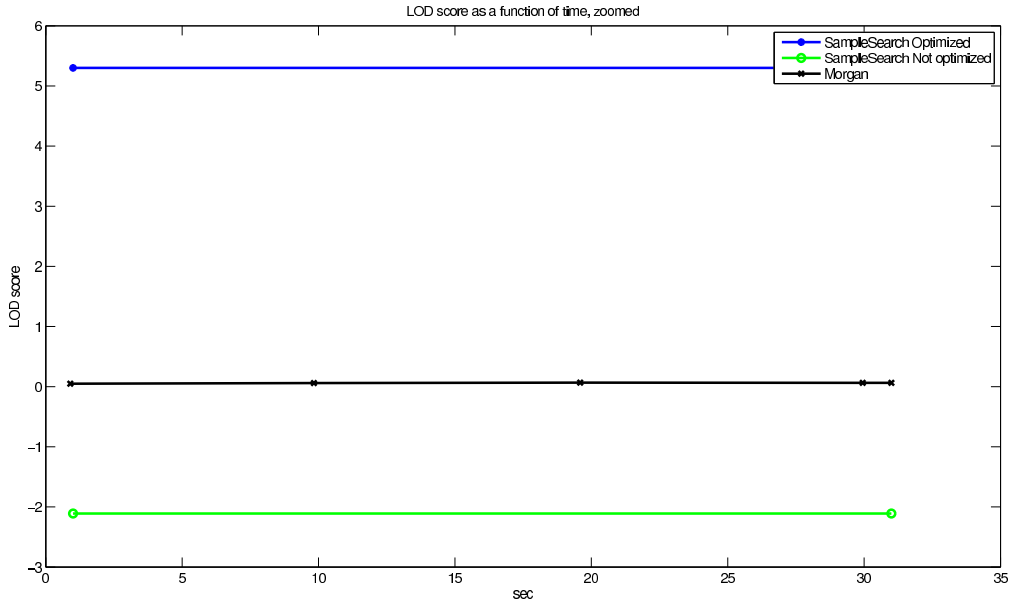

Figure 6: 110_22_0: LOD score as a function of the cut off time, zoomed

On the zoomed plot it can be seen that Morgan and SampleSearch output fisrt results

practially at the same time.

Instance 110_22_1 (with and without optimization suring conversion)

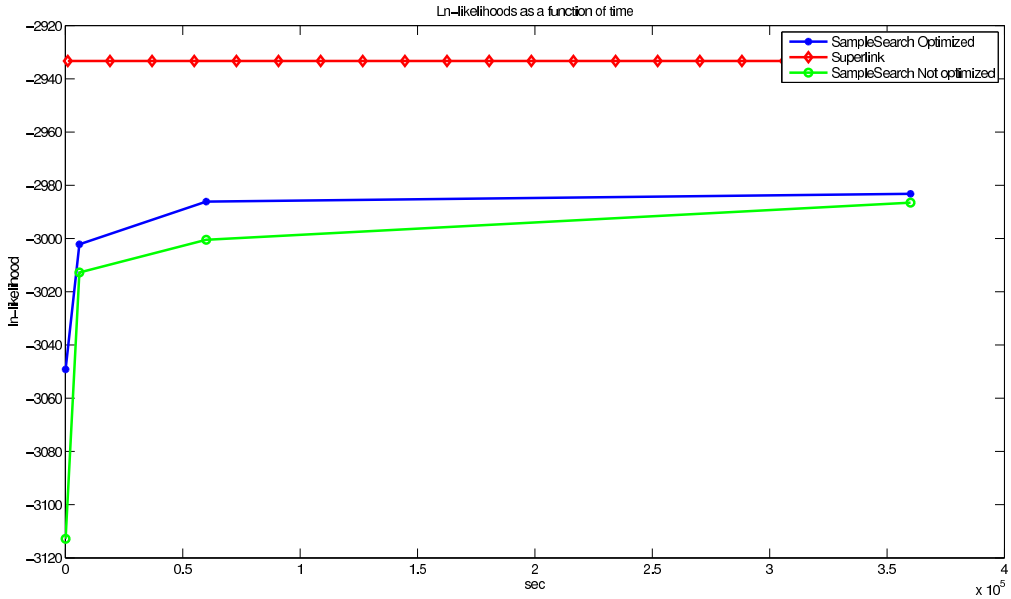

Figure 7: 110_22_1: log-likelihood as a function of the cut off time

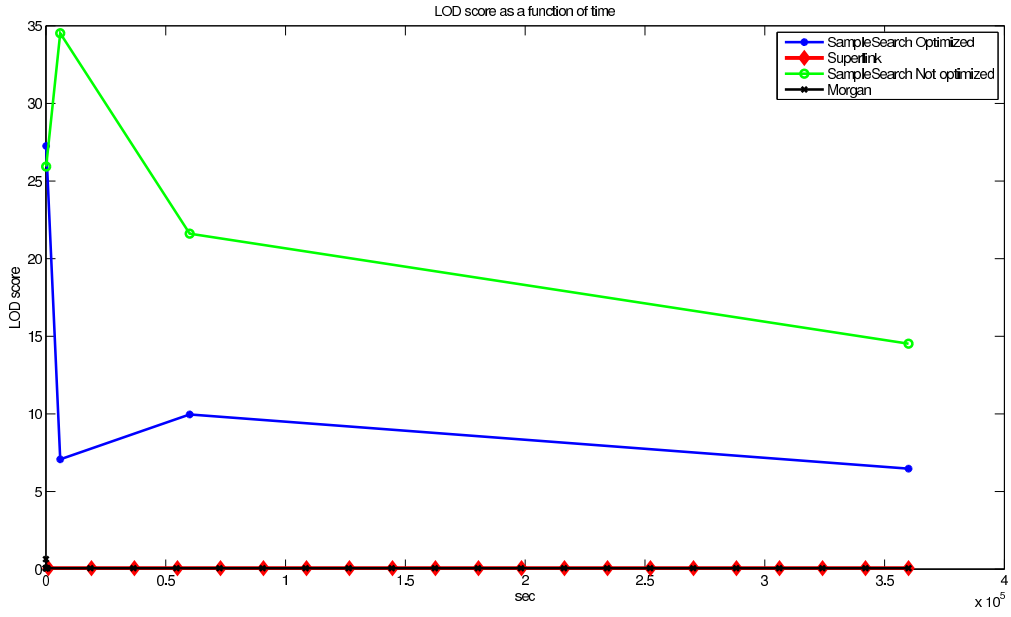

Figure 8: 110_22_1: LOD score as a function of the cut off time

Morgan and Superlink find the same value of LOD scorea and the same position.

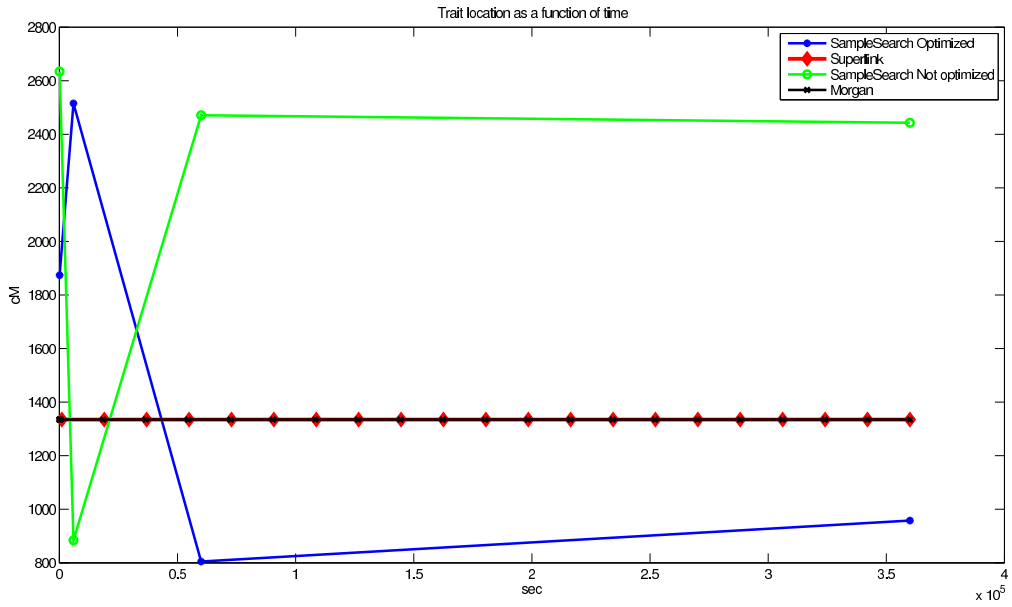

Figure 9: 110_22_1: trait locations as a function of the cut off time

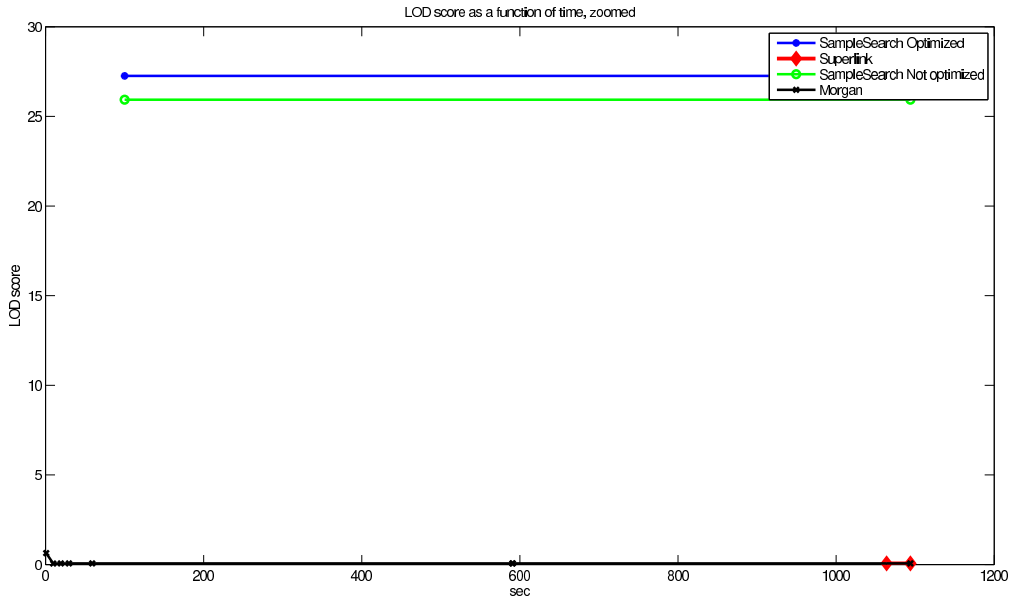

Figure 10: 110_22_1: LOD score as a function of the cut off time, zoomed

For this instance SampleSearch used with and without optimization provides drastically different values of trait locus position, though the log-likelihood and LOD score have a similar trend.

Instance 100_23_0 (with and without optimization)

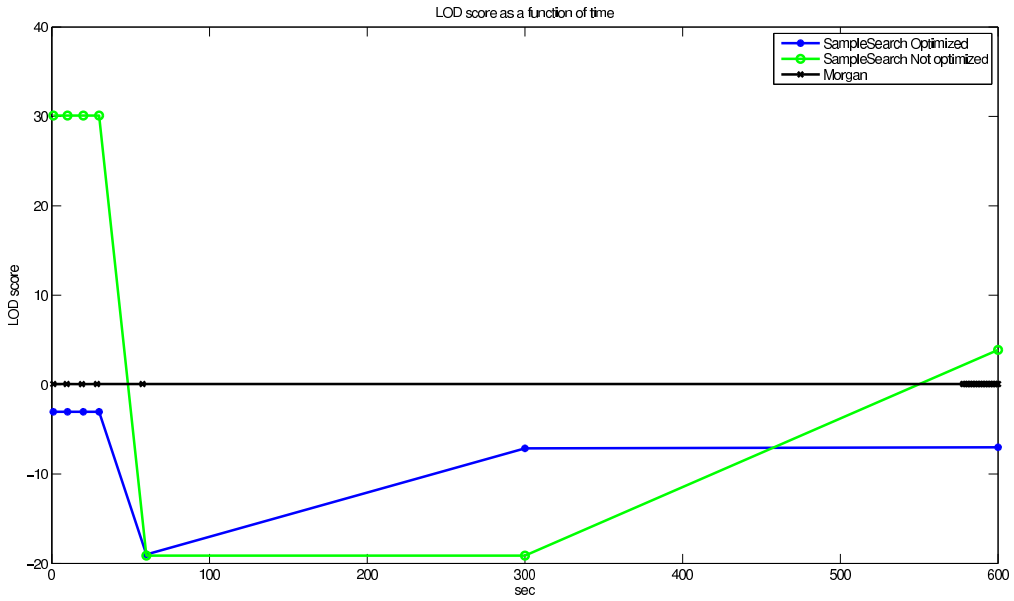

Figure 11: 100_23_0: LOD score as a function of the cut off time

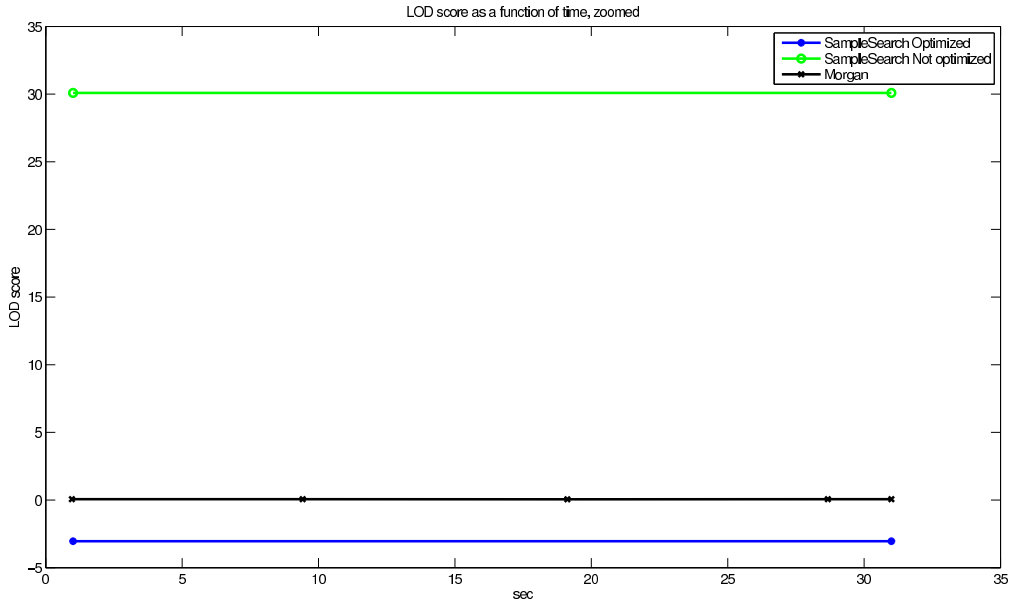

Figure 12: 100_23_0: LOD score as a function of the cut off time, zoomed

Instance 100_23_1 (with and without optimization) On this instance Superlink ran out of memory and didn't finish calculation.

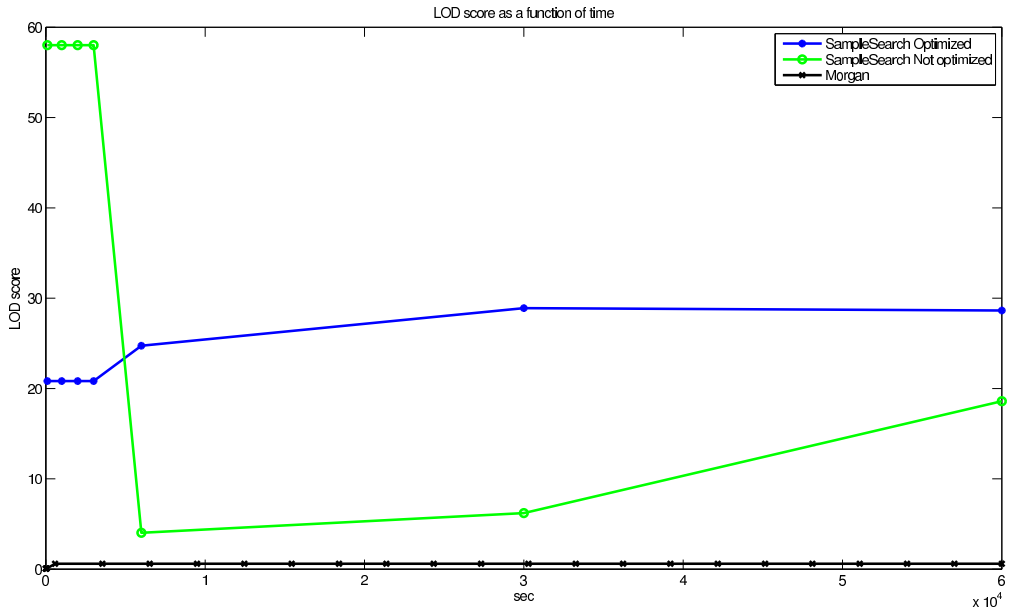

Figure 13: 100_23_1: LOD score as a function of the cut off time

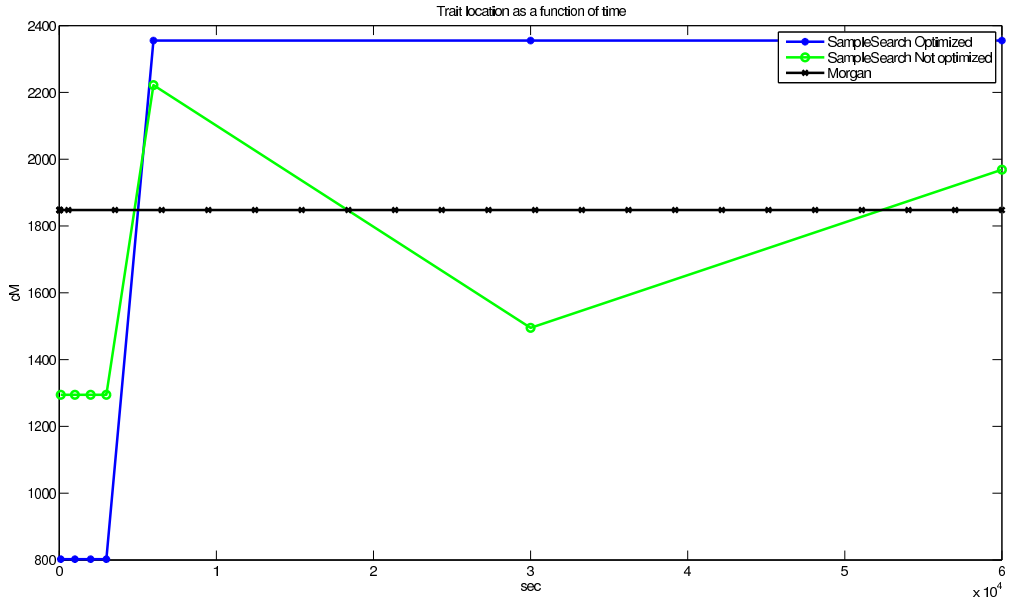

Figure 14: 100_23_1: trait locations as a function of the cut off time

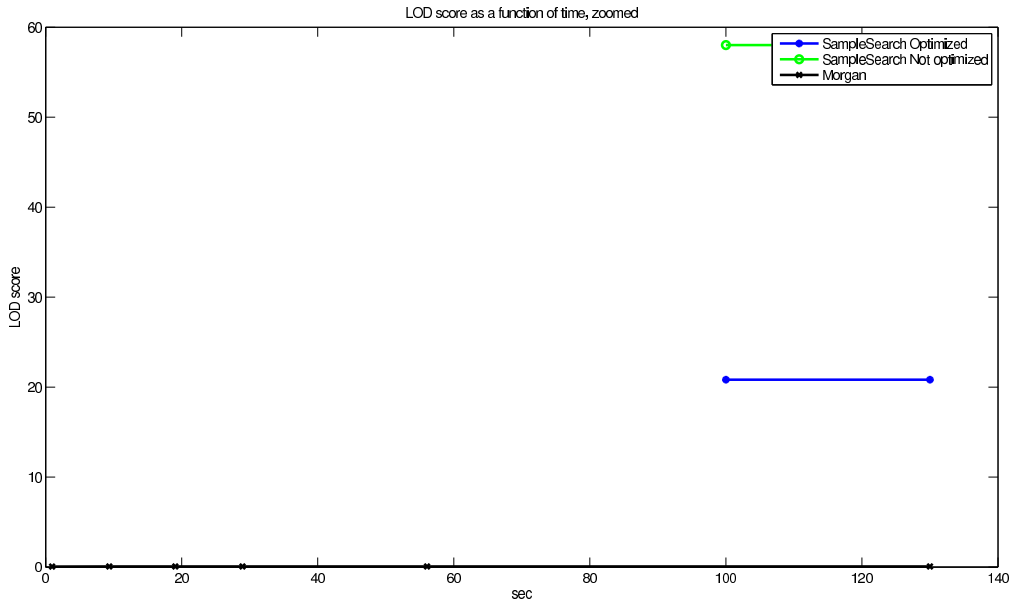

Figure 15: 100_23_1: LOD score as a function of the cut off time, zoomed

The 100_23 instances are hard, even Superlink can solve this instances in over an hour and a half (which is evident from the zoomed plots). SampleSearch provides LOD scores that are far from the ones by Morgan and that dont converge even after running for 5 hours per network. However, with the increase of the cut off time the trait location found by SampleSearch with optimization is coming closer to the Morgan for some period, after which the error rapidly increases again.

Instance 100_23_2 (with and without optimization)

On this instance Superlink ran out of memory and didn't finish calculation.

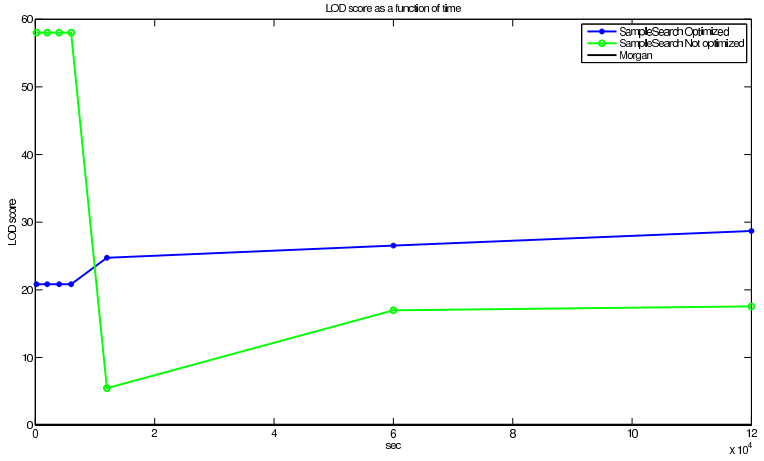

Figure 16: 100_23_2: LOD score as a function of the cut off time

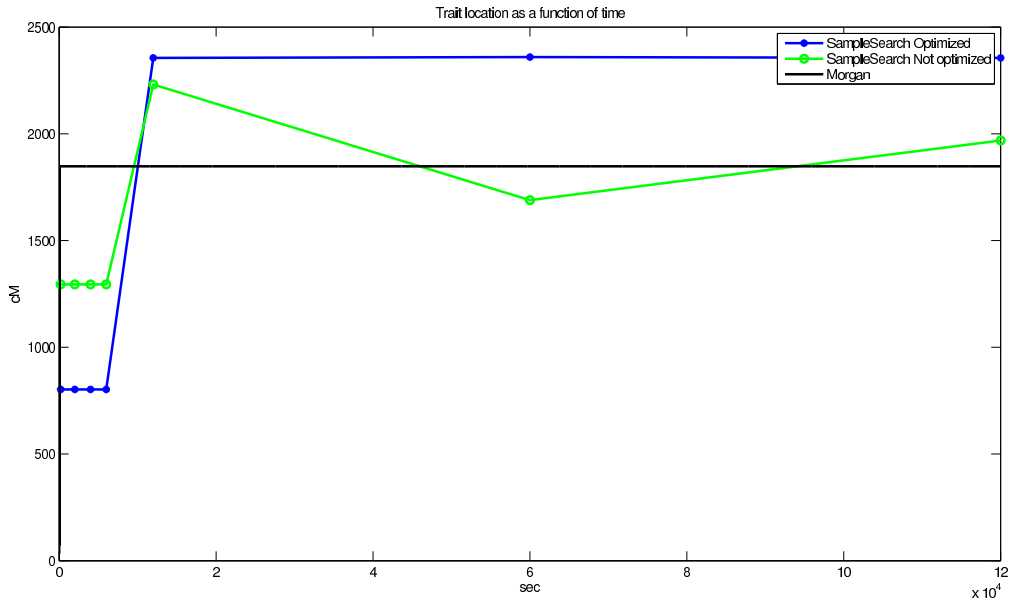

Figure 17: 100_23_2: trait locations as a function of the cut off time

Though, due to the scale of X axis, it is hard to decipher on the graph, for this instance Morgan fails to converge until arounf 60 second, even though the correct location of the trait is found within 20 seconds. However, if the program is run up to 40 seconds, result changes.

Instance 100_23_3 (with optimization)

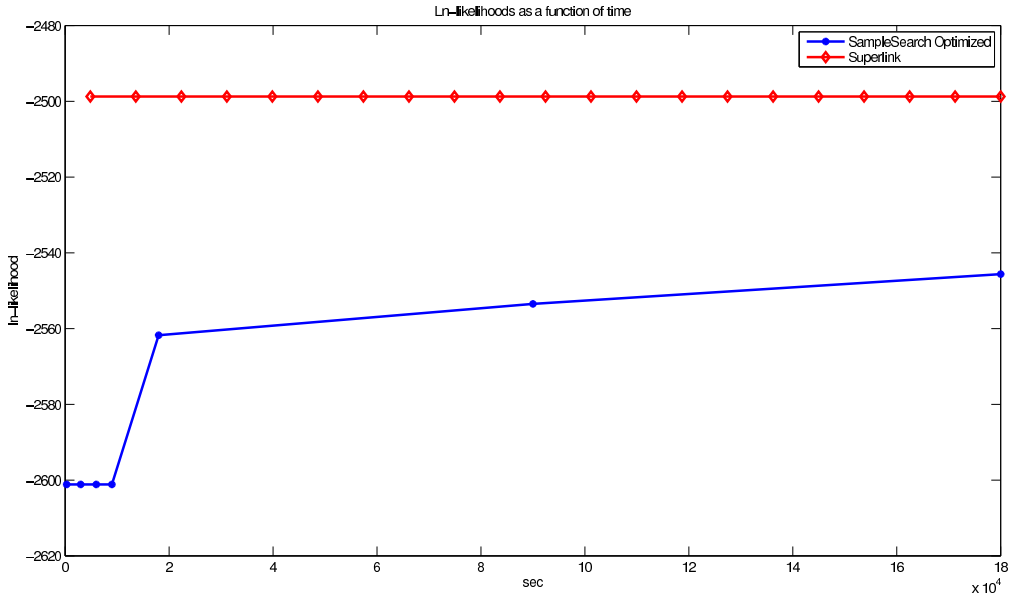

Figure 18: 100_23_3: log-likelihood as a function of the cut off time

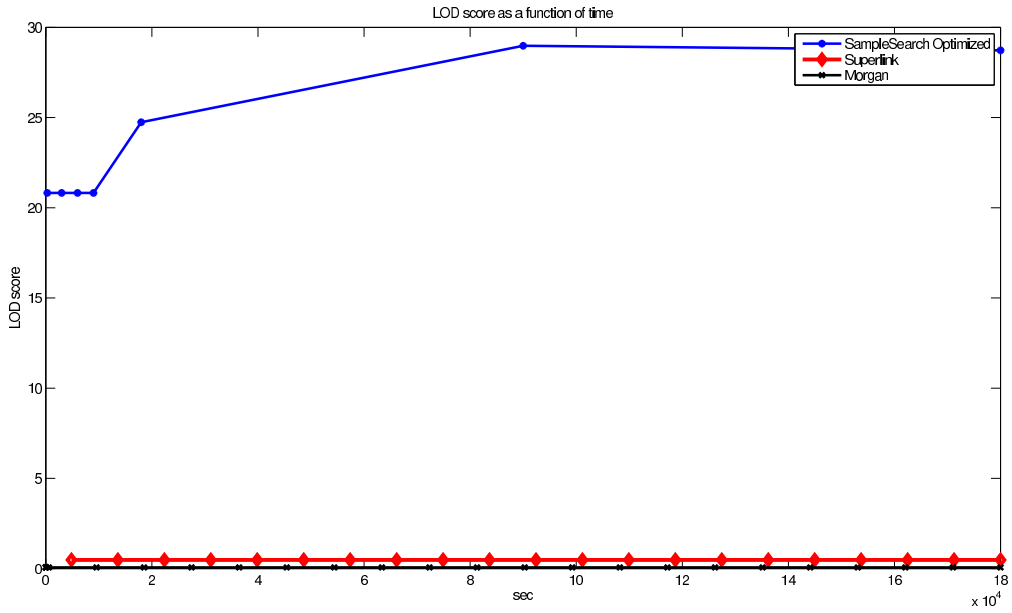

Figure 19: 100_23_3: LOD score as a function of the cut off time

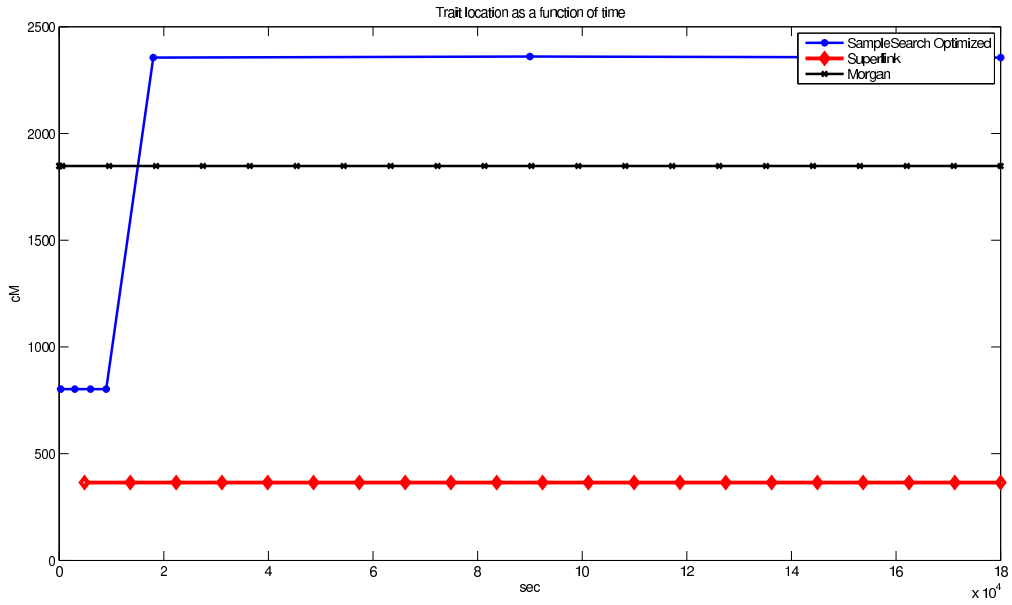

Figure 20: 100_23_3: trait locations as a function of the cut off time

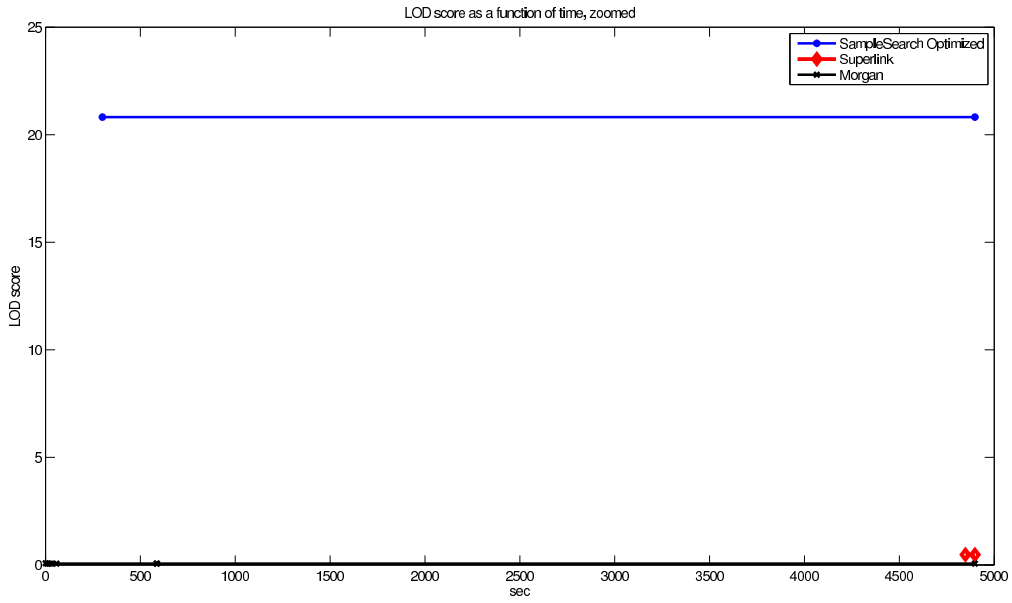

Figure 21: 100_23_3: LOD score as a function of the cut off time, zoom

For this instance, unlike most of others, the accuracy of LOD score calculation by SampleSearch decreases as the cut off time grows. Also, uncharacteristically, Morgan and Superlink completely disagree on the position of the trait locus and output different values of LOD score. On the zoomed plot it is evident. that Morgan outputs result almost instancely, while Superlink takes considerable time.

Instance 110_22_0 (with and without optimization) It can be seen that the LOD score of the SampleSearch with optimization is much closer to the results by Morgan, then the one that is not optimized.

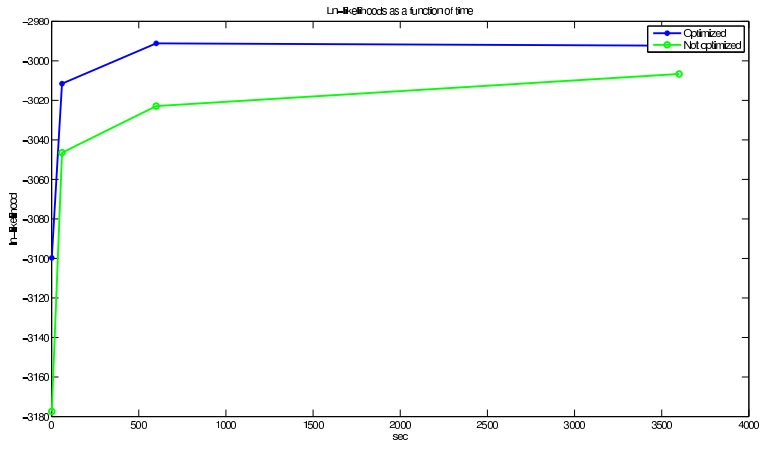

Figure 22: 110_22_0: log-likelihood as a function of the cut off time

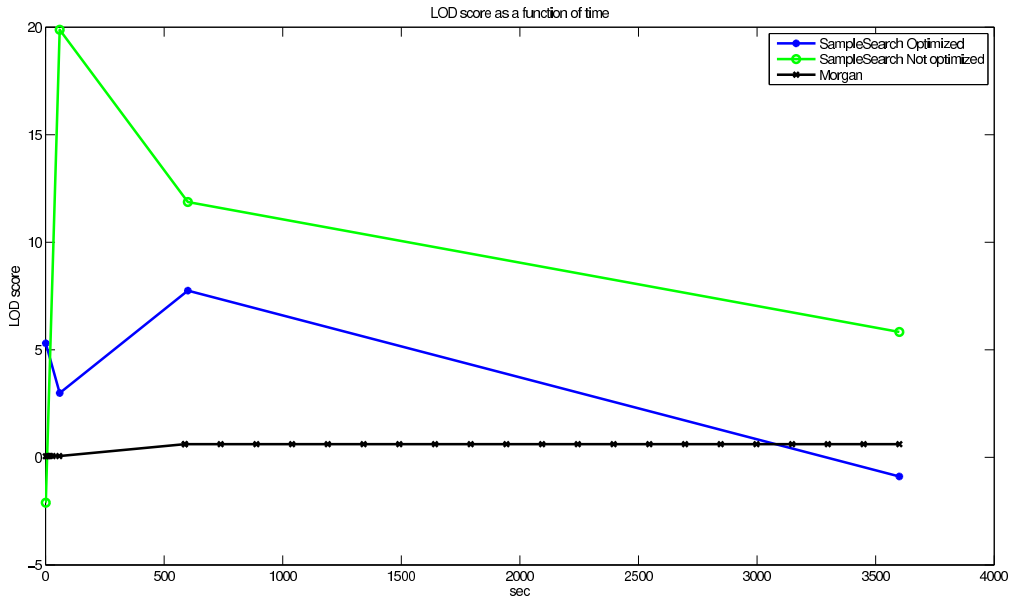

Figure 23: 110_22_0: LOD score as a function of the cut off time

Instance 110_22_1 (with and without optimization)

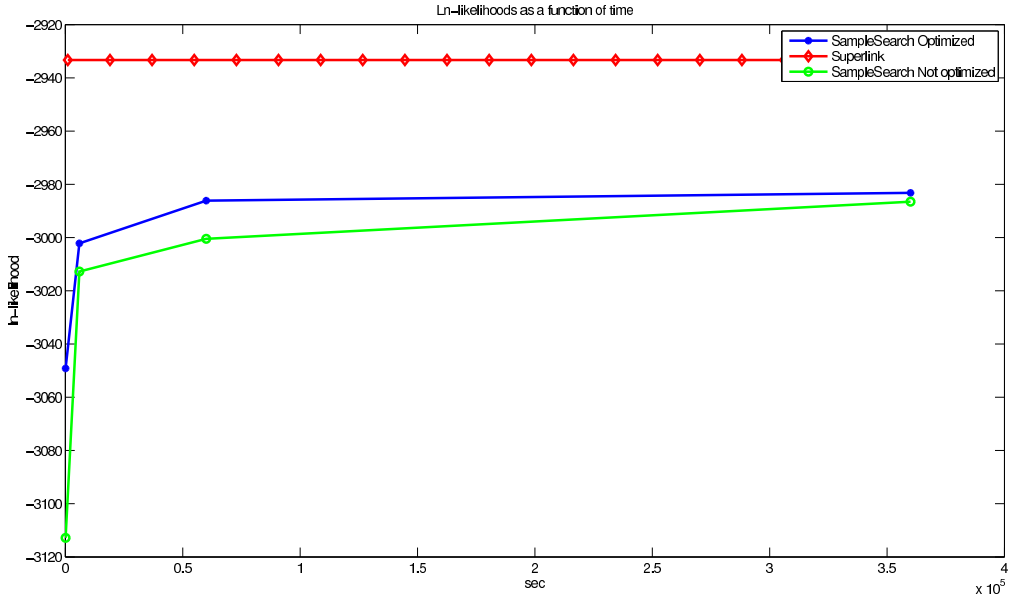

Figure 24: 110_22_1: log-likelihood as a function of the cut off time

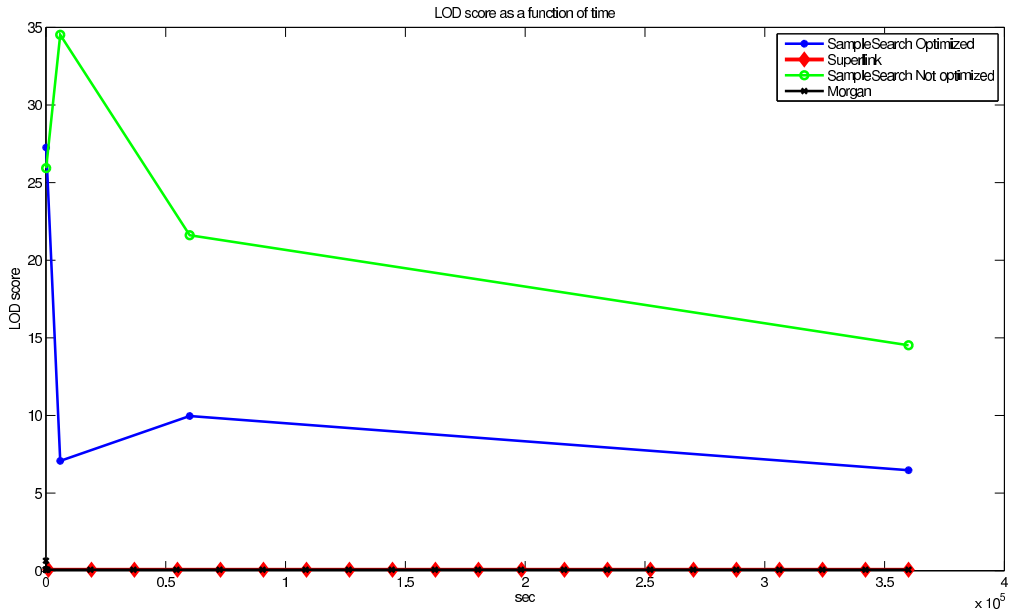

Figure 25: 110_22_1: LOD score as a function of the cut off time

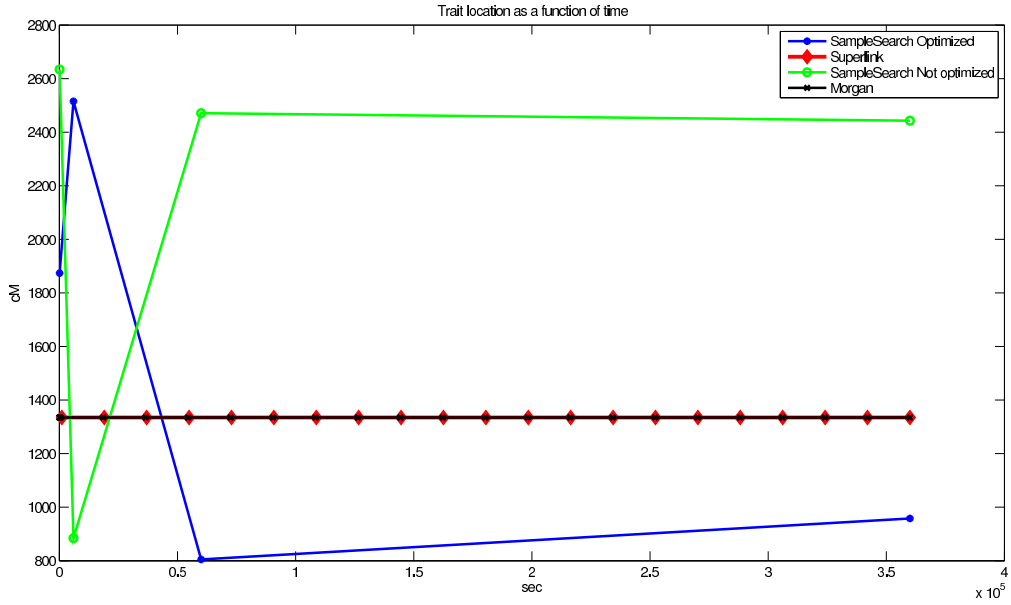

Figure 26: 110_22_1: trait locations as a function of the cut off time

Morgan and Superlink output identical results for this instance.

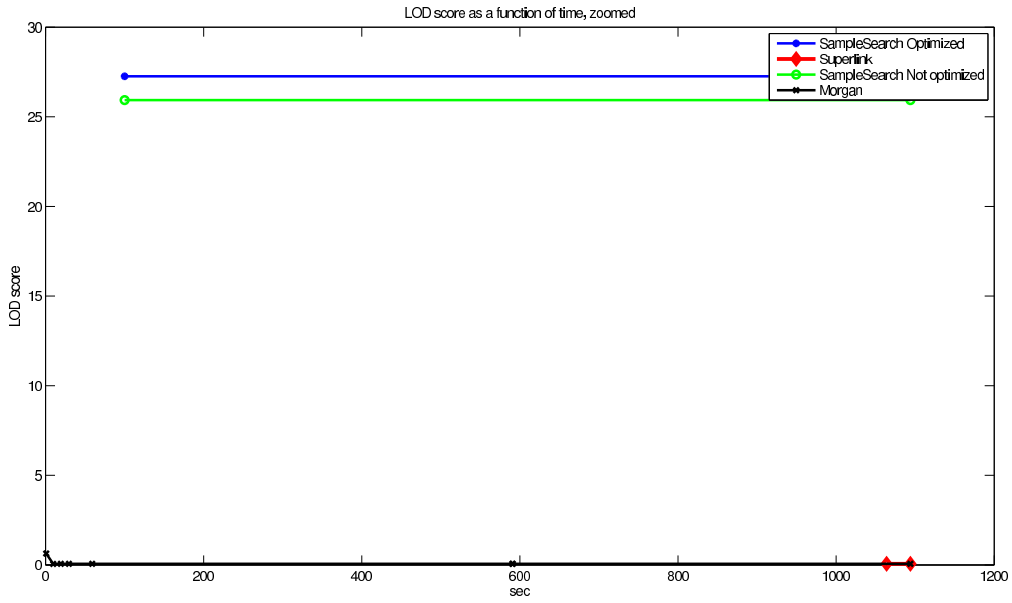

Figure 27: 110_22_1: LOD score as a function of the cut off time, zoomed

Superlink and Morgan found exactly the same position for the trait loci, while SampleSearch is again considerably off the mark. Optimization again helps SampleSearch to find more accurate solution.

Instance 100_19_0 (optimization was used during conversion)

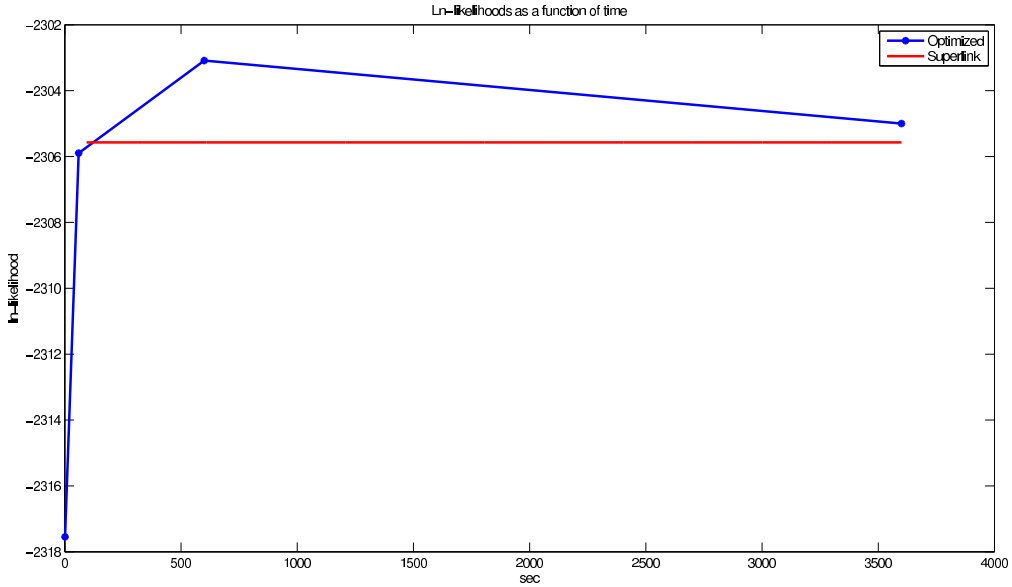

Figure 28: 100_19_0: log-likelihood as a function of the cut off time

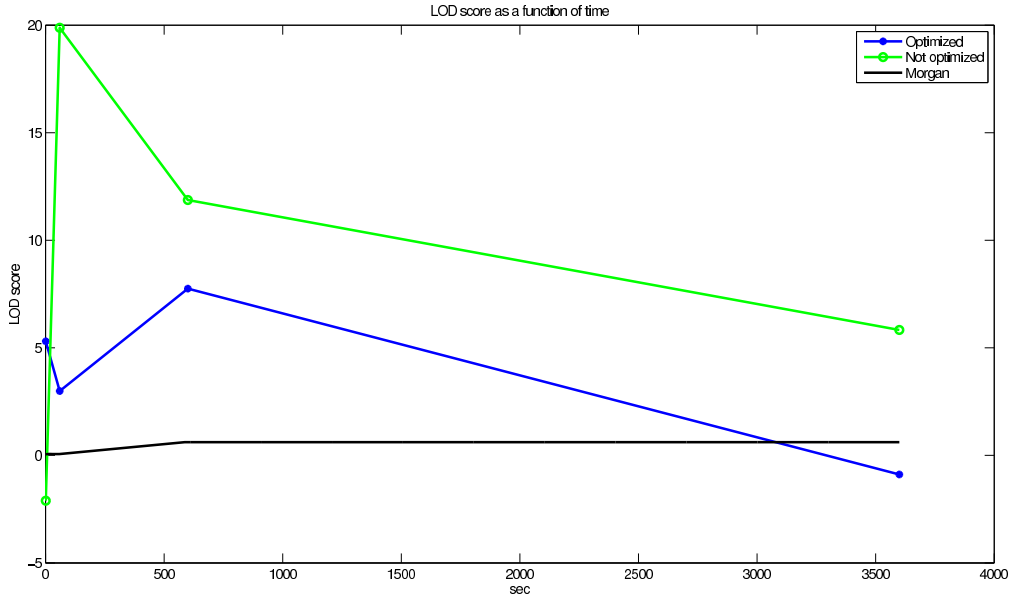

Figure 29: 100_19_0: LOD score as a function of the cut off time

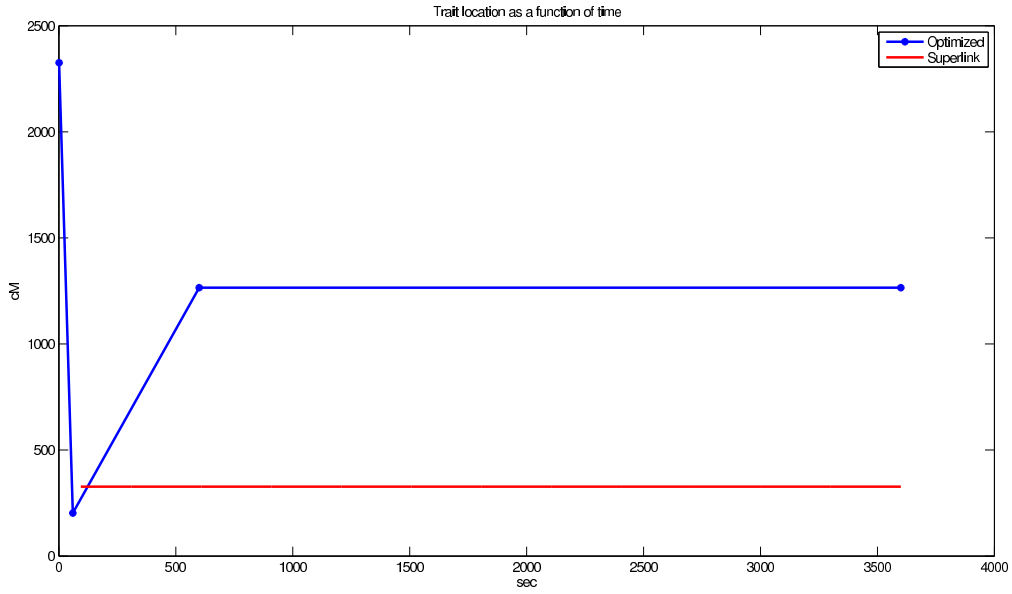

Figure 30: 100_19_0: trait locations as a function of the cut off time

The results for Morgan are not available for this or next instance, because Morgan crashed on both of them. Using optimization during conversion, with time SampleSearch is able to come quite close to the exact value of the log-likelihood. However, the value of LOD score is quite different. That can be explained by SampleSearch not being able to evaluate accurately

the probability of evidence in absence of the linkage. It also can be noted, that the position of the trait locus found by SampleSearch, which does not significantly changes with the increase of cut off time from 600 to 3600 seconds per instance, does not coincide with the one reported by Superlink.

Instance 100_19_1 (optimization was used during conversion)

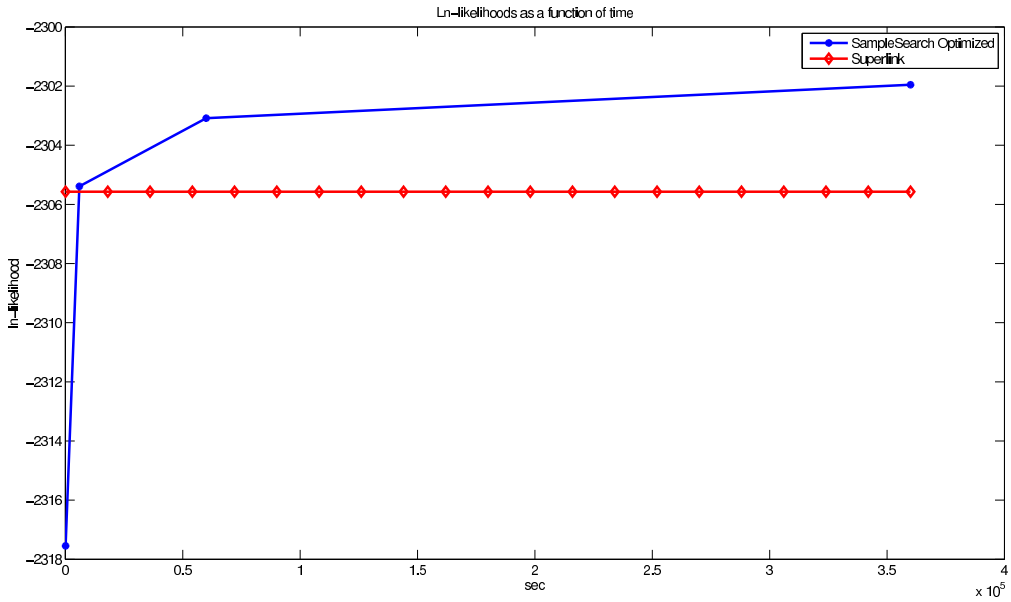

Figure 31: 100_19_1: log-likelihood as a function of the cut off time

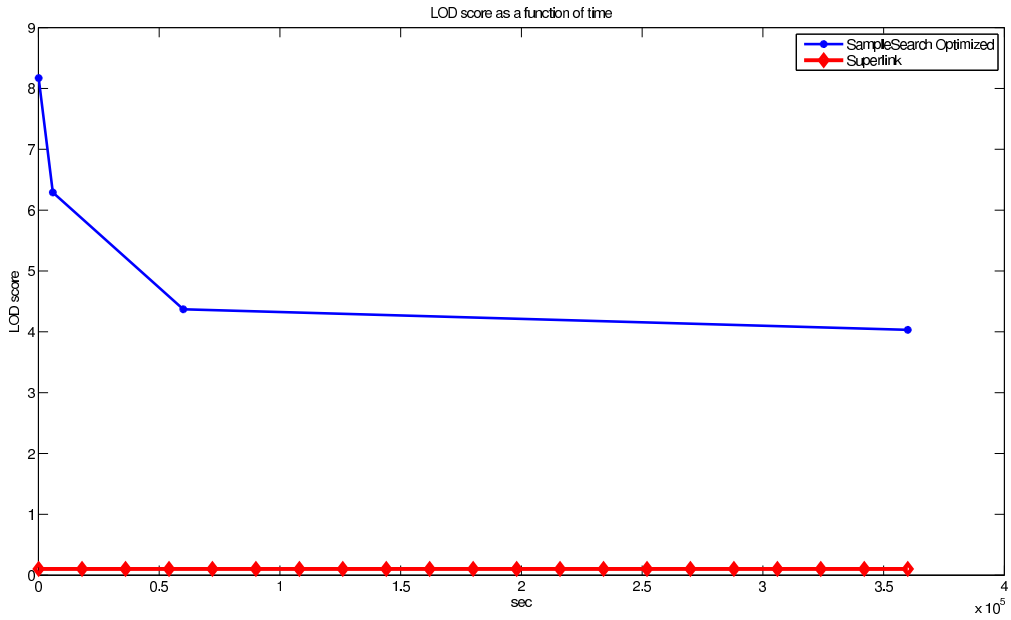

Figure 32: 100_19_1: LOD score as a function of the cut off time

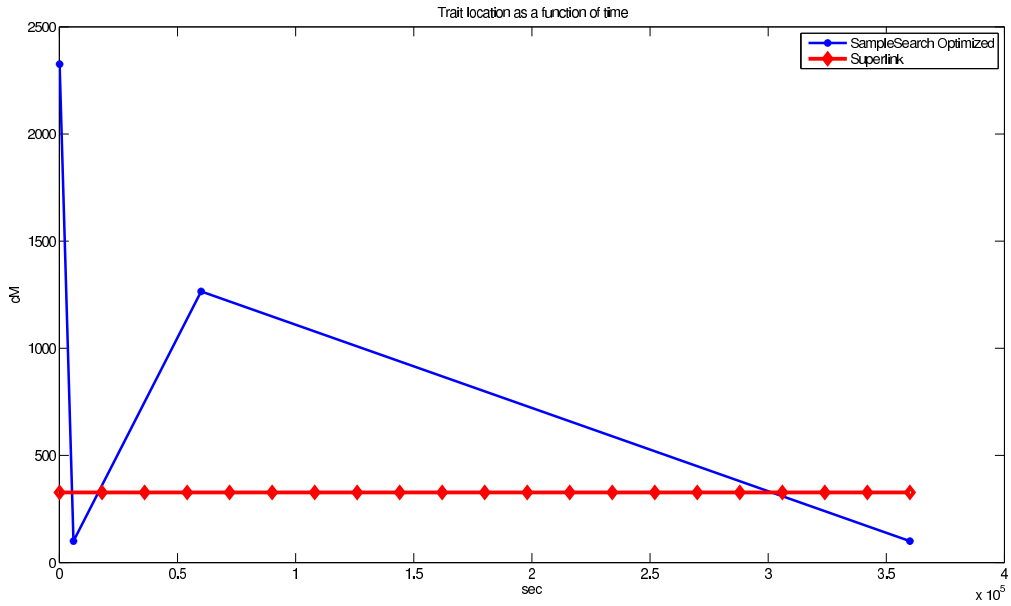

Figure 33: 100_19_1: trait locations as a function of the cut off time

SampleSearch could not solve the instance without optimization.

Instance 200_12_0 (optimization was always used for conversion)

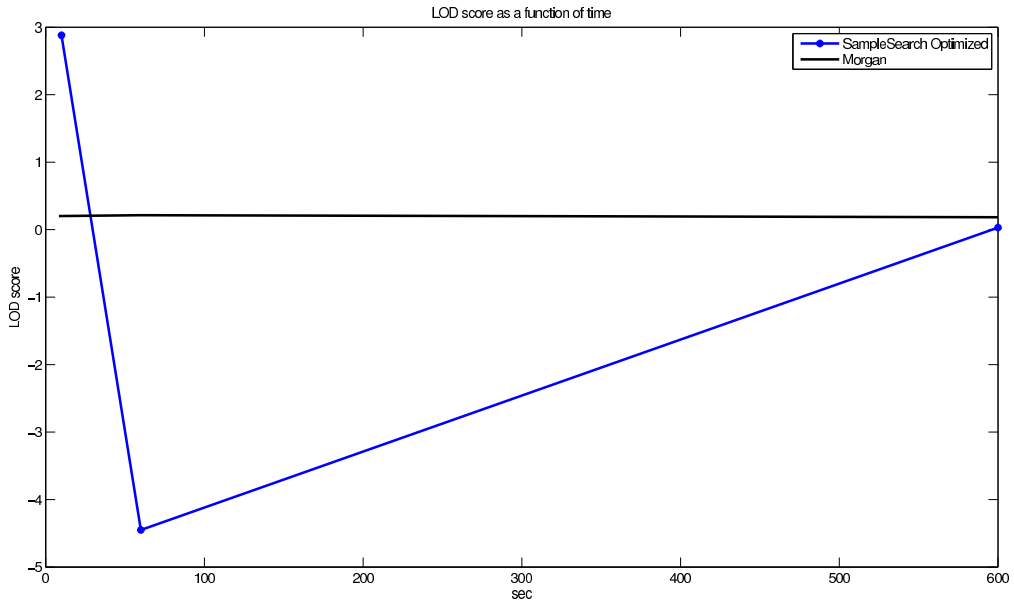

Figure 34: 200_12_0: LOD score as a function of the cut off time

Though for the smaller cut off time the LOD score calculated by SampleSearch is dramatically different from the one by Morgan, with the increase of run time, SampleSearch produces result which is quite close to the correct one.

Instance 200_12_1 (optimization was always used for conversion)

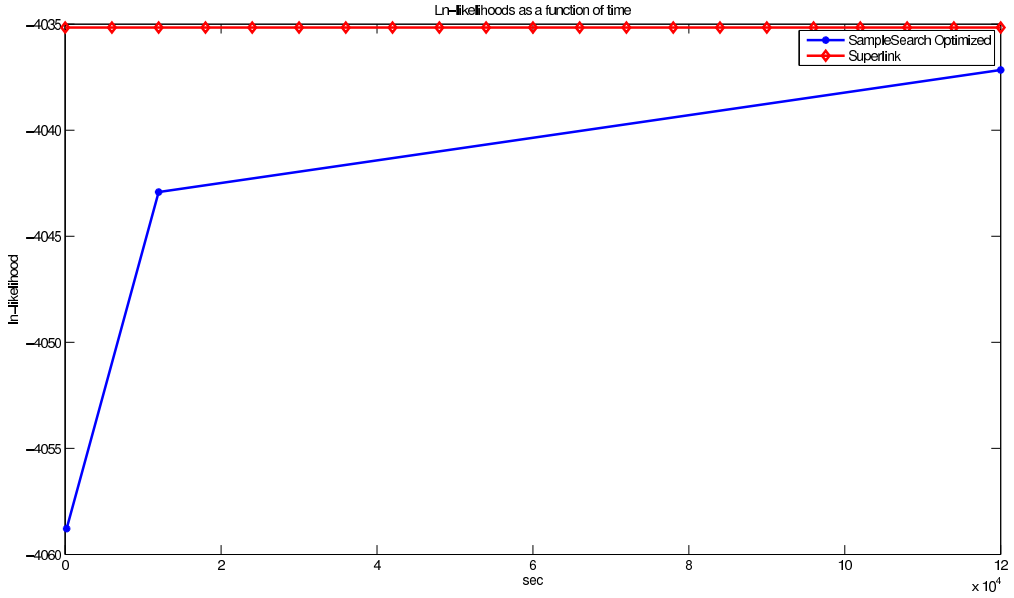

Figure 35: 200_12_1: log-likelihood as a function of the cut off time

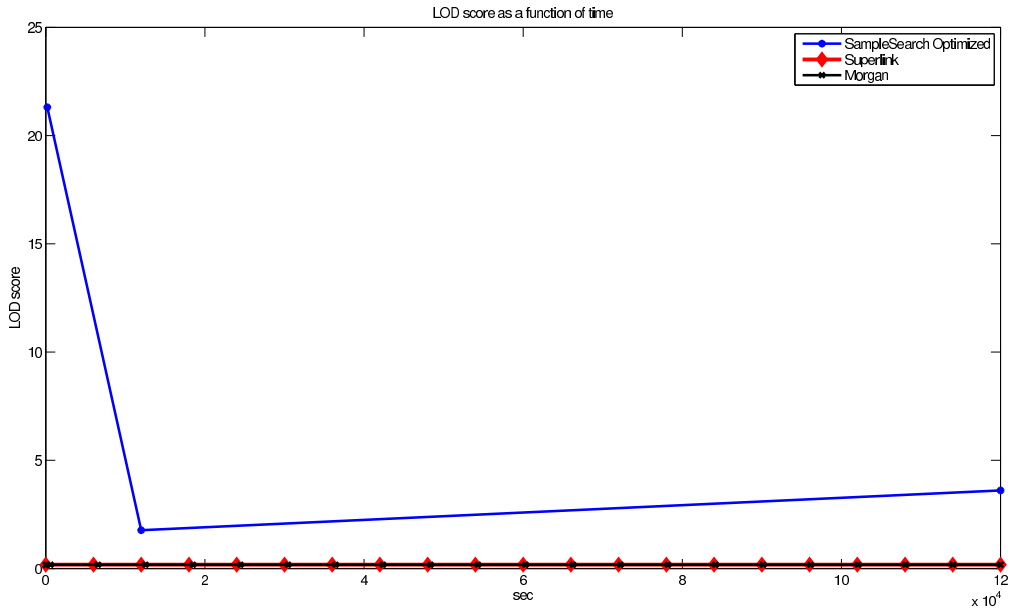

Figure 36: 200_12_1: LOD score as a function of the cut off time

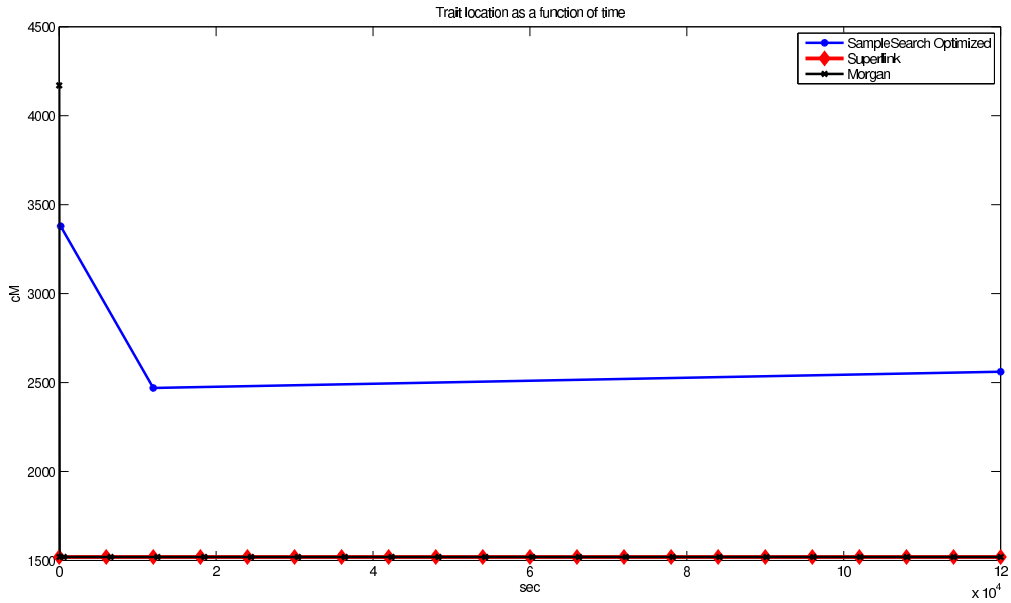

Figure 37: 200_12_1: trait locations as a function of the cut off time

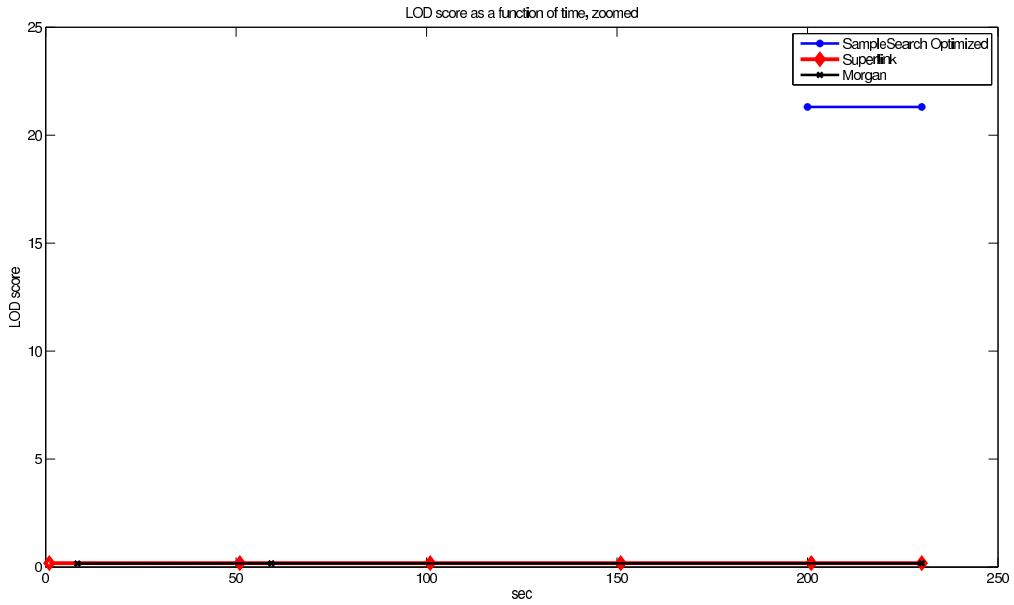

Figure 38: 200_12_1: LOD score as a function of the cut off time, zoomed

For this instance, though SampleSearch does not provide exact values of LOD score and log-likelihood, there can be seen a clear tendency towards convergence with Superlink results as the cut off time increases.

Instance 200_16_0 (optimization was used during conversion)

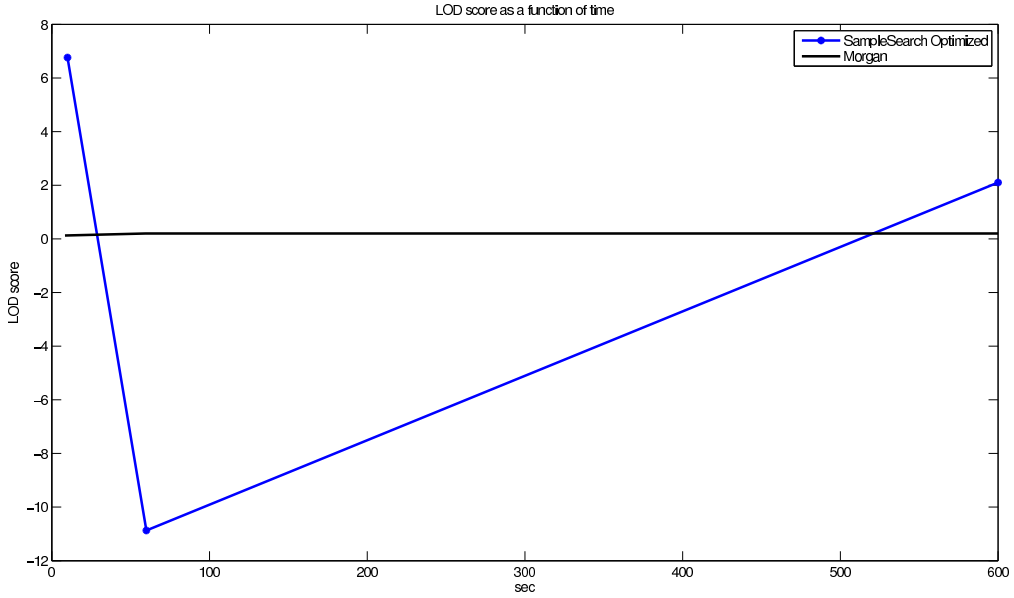

Figure 39: 200_16_0: LOD score as a function of the cut off time

Instance 200_16_1 (optimization was used during conversion)

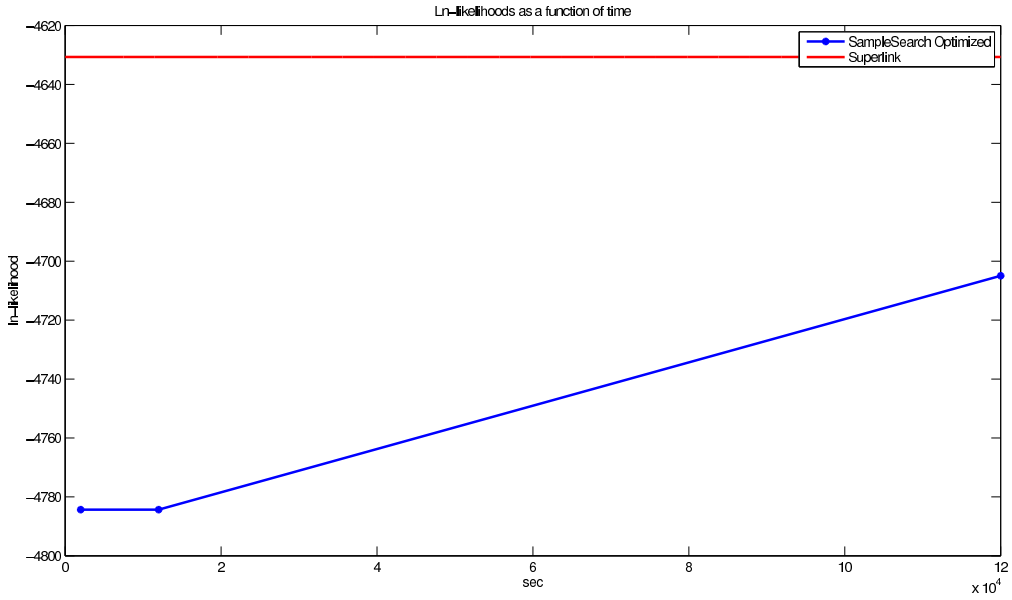

Figure 40: 200_16_1: log-likelihood as a function of the cut off time

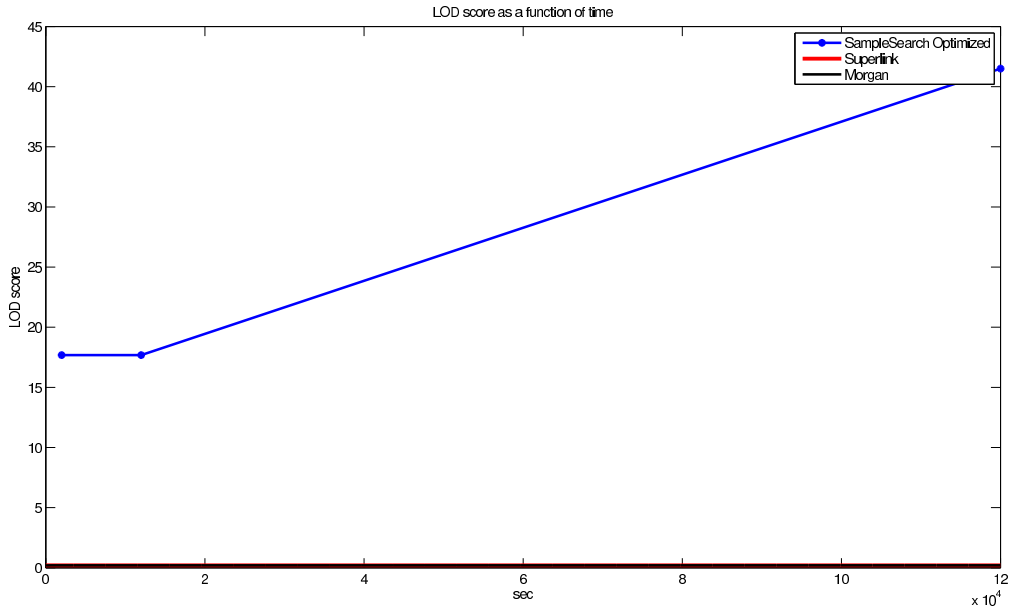

Figure 41: 200_16_1: LOD score as a function of the cut off time

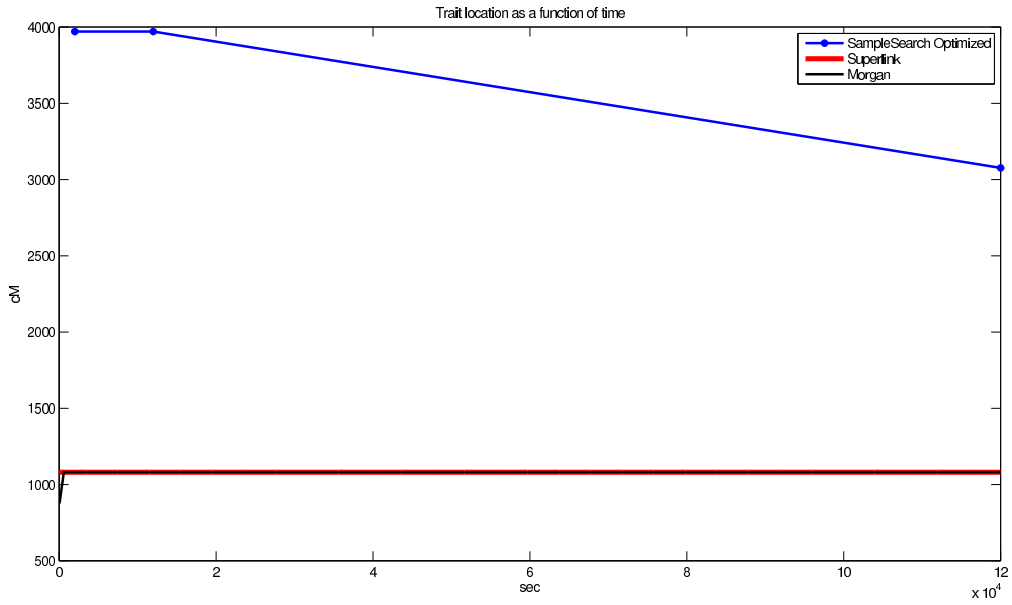

Figure 42: 200_16_1: trait locations as a function of the cut off time

In this instance not only SampleSearch does not provided a correct LOD score initially, it also does not show any move toward convergence with the increase of the running time, even though the log-likelihood comes closer to the exact one in time. This behavior of the LOD score can be explain by the inability of SampleSearch to correctly estimate the probability of unlinked trait.

Instance 100_25 For this instance there is no solution known to us available, except the one provided by SampleSearch. In our experiments both Morgan and Superlink crashed on this instance. According to the test results available on the Supelrink site, none of the programs tested were able to solve this instance. Thus,we can not estimate the quality of solution provided by SampleSearch in the absence of the ground truth.

2.2 Scaling of the running time with the number of trait loci positions

In order to see how well the performances of the programs scale with the number of locations at which LOD score needs to be calculated in the problem, we varied the number of equally spaced positions between each two adjacent markers. For the SampleSearch each new position of the trait locus corresponds to solving from scratch a new Bayesian network and thus the running time is strictly proportional to the number of trait positions. However, for the Superlink it is not true as can be seen in Figure 43, Figure 44 and Figure 45.

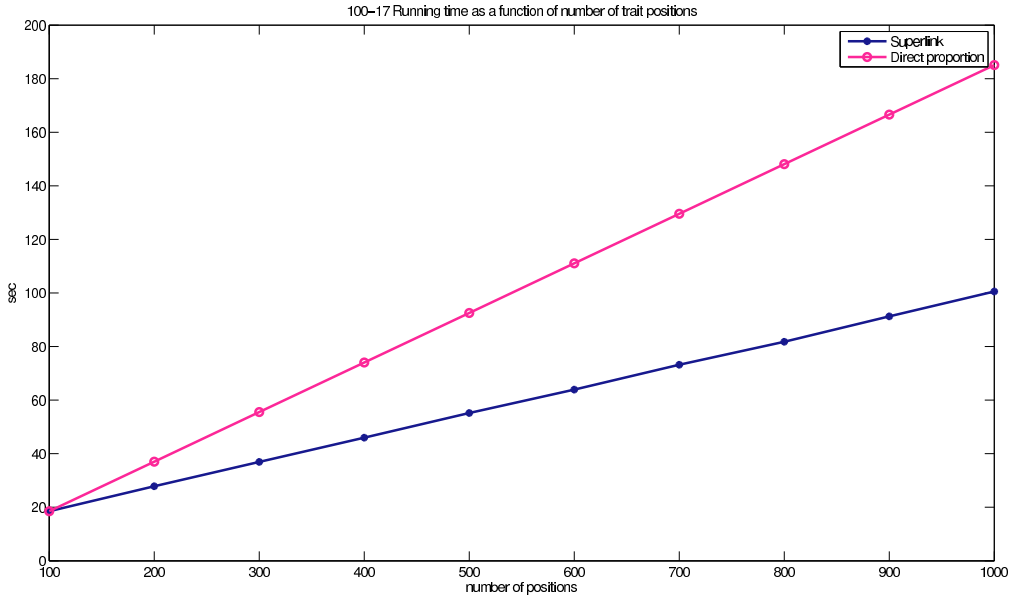

Figure 43: 100_17 Superlink

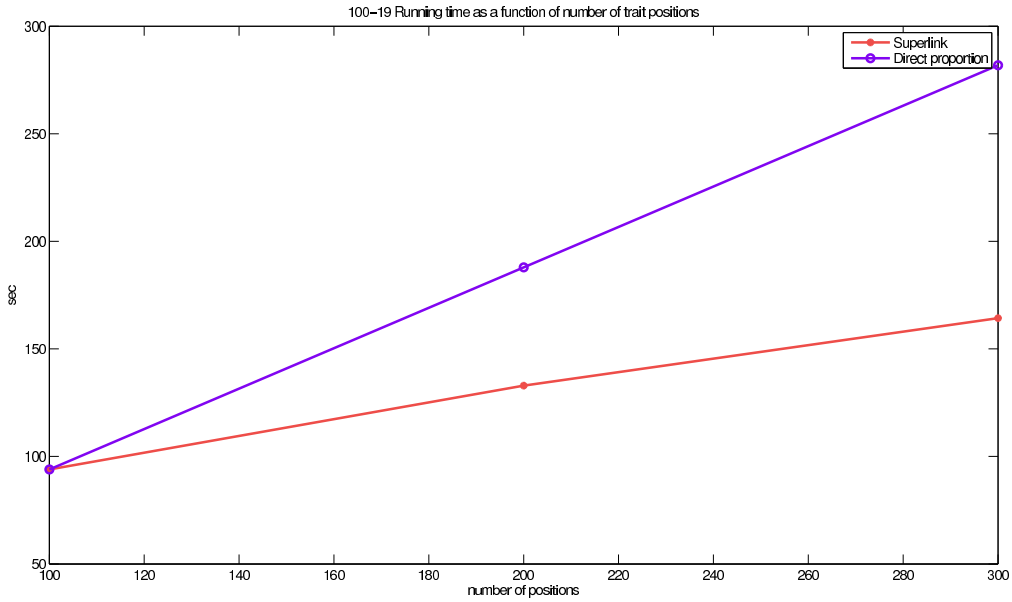

Figure 44: 100_19 Superlink

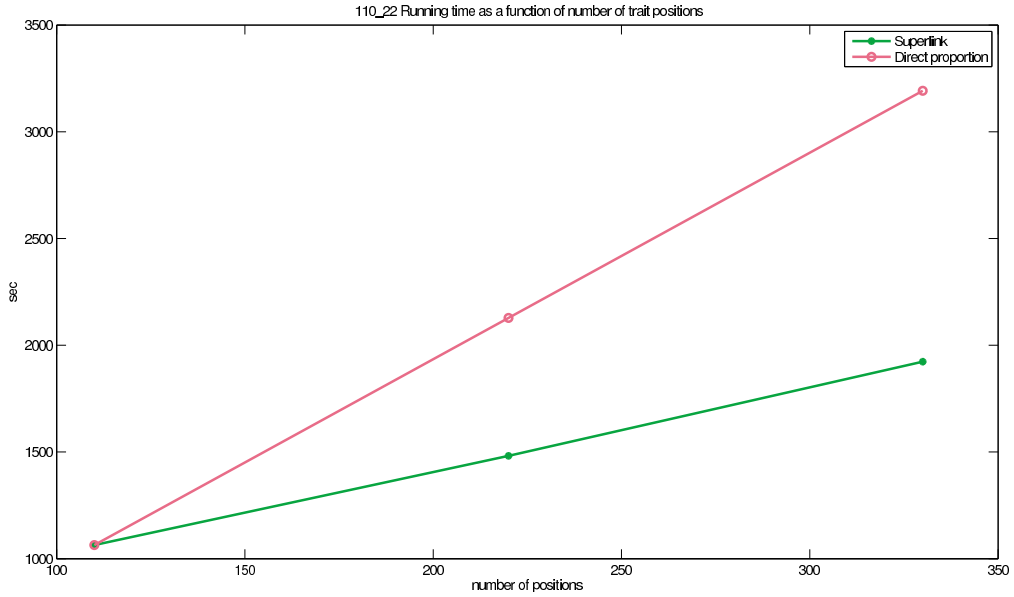

Figure 45: 110_22 Superlink

It can be seen that Superlink scales better the harder is the problem. From Figure 46 it is clear that 100_17 is the easiest problem of the three, however from the Figure 47 we see that the relative increase of the running time of the program with the number of trait positions processed is in fact the greatest for this instance.

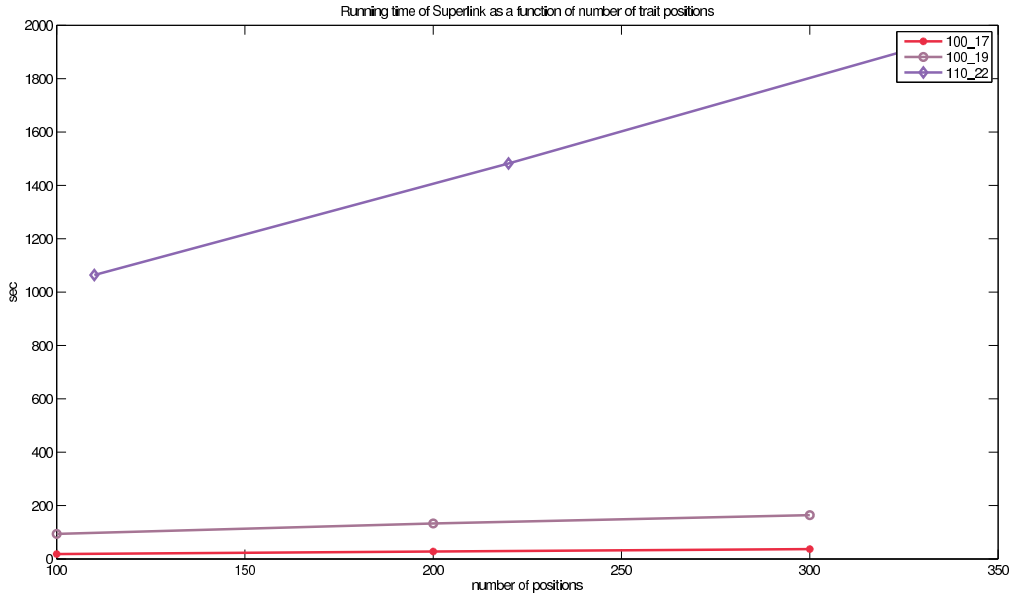

Figure 46: Scaling of the Superlink performance with the number of trait locus positions

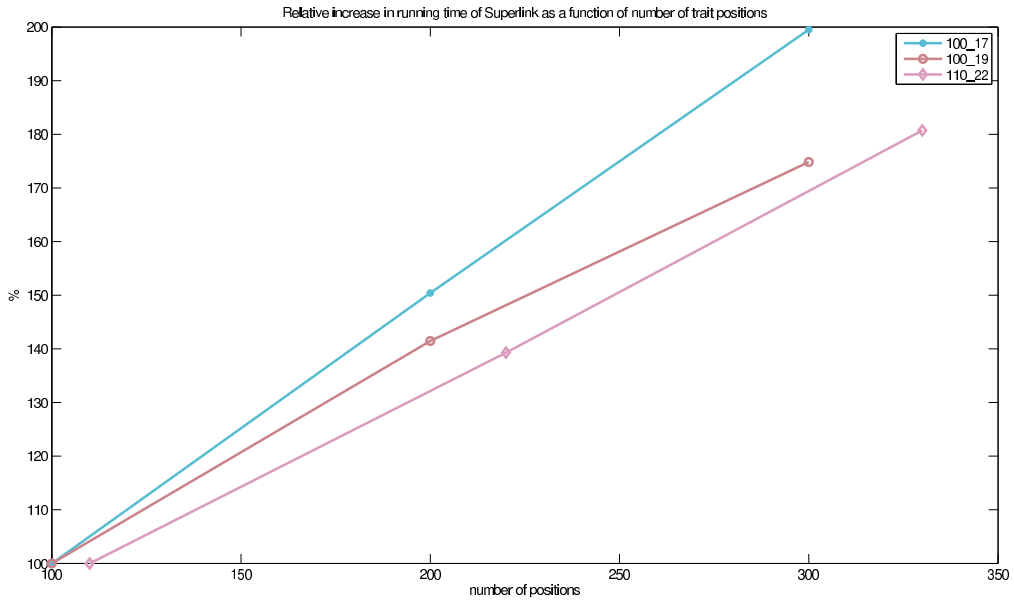

Figure 47: Relative performance of Superlink with the number of trait locus positions

To estimate how well Morgan, which is anytime and approximate, scales with the number of trait positions in the problem, we look at the time and the number of MCMC iteration needed to converge to the most likely trait location, i.e. outputted trait position does not change with the increase of cut off time). Full numberical details of the Morgan performance are presented in the appendix.

Instance 100_23

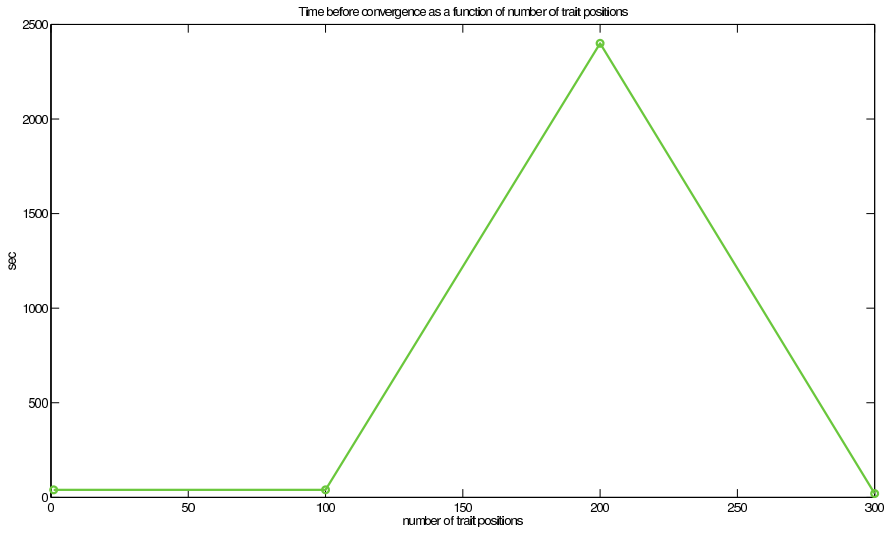

Figure 48: 100_23 Morgan: time before convergence

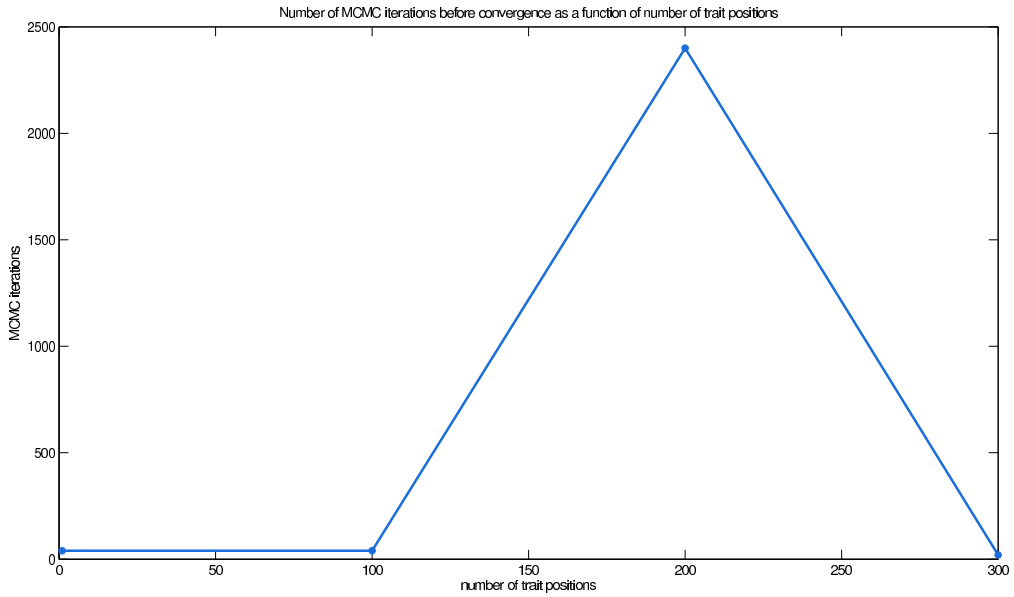

Figure 49: 100_23 Morgan: number of MCMC iterations before convergence

The peak of the run time around 200 positions is surprising and can't be fully explained at the moment.

Instance 110_22

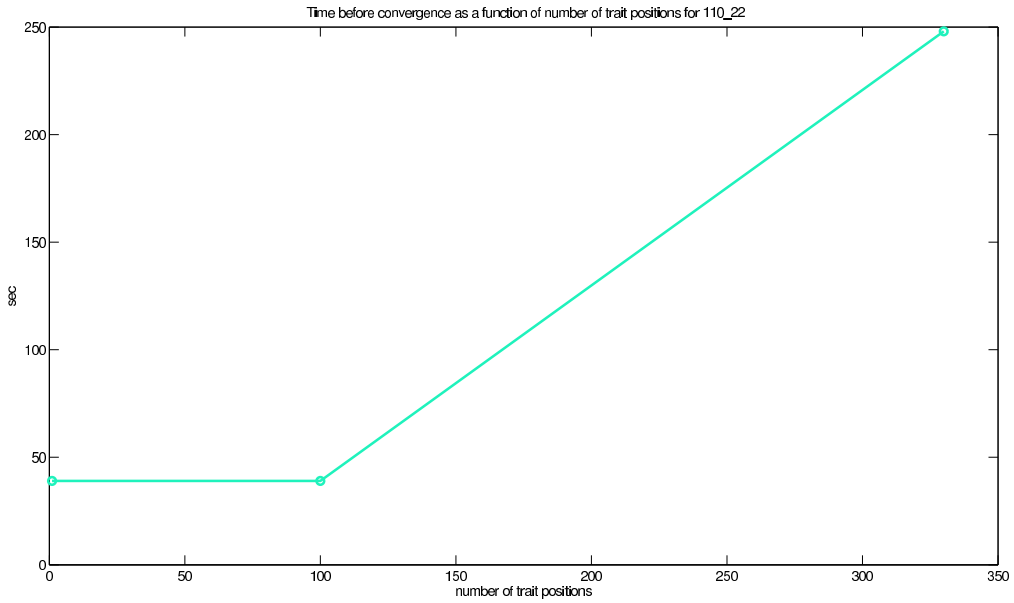

Figure 50: 110_22 Morgan: time before convergence

3 Discussion

The experimental comparison between Superlink, Morgan and SampleSearch revealed that our general purpose scheme, though known to perform well on a variaty of benchmarks, can not compete with the specialized algorithms in solving the task of fining LOD score. We can outline three main issues that make the performance of SampleSearch unsatisfactory.

We can point out three main issues that put SampleSearch at a disadvantage.

First of all, the LOD score (maximum across all posible trait positions in a given problem) found by SampleSearch is usually considerably larger then the exact one found by Superlink, even when the values of maximum log-likelihood found by the two programs are close. In some cases the error is reaches an order of magnitude. We can make a conjecture that the problem is caused by the incorrect assesement of the log-likelihood in the absence of linkage, i.e. the denominator in the expression for the LOD score (Equation 1).

Another problem is the great inaccuracy of the estimation of the most likely trait position on the chromosome by SampleSearch on all instances. The reasons for such behaviour of SampleSearch is unclear at present and requires further investigation.

The third issue, which is inherent to using any general purpose algorithm for purposes of genetic linkage analysis, is the scalability with the number of trait positions at which LOD score is estimated. Each such position corresponds to a separate Bayesian network, which is solved from scratch each time. Thus the running time of SampleSearch is linear in the number of networks. We saw that Superlink and Morgan don't have such problem. We assume that both of the programs reuse the calculations done for the previous trait position in estimating the current one, but we don't know the details of the algorithms. We hope that once we understand the principles behind high efficiency of Superlink and Morgan, we can use similar ideas to improve SampleSearch's performance.

4 Appendix

Morgan: the experimental results.

Name: 100_23_i1
Pedigree size: 23
Number of Markers: 100
Locations per Marker: 0

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
1	0.96	40	0.0656	1847.772
10	9.42	400	0.0604	1847.772
20	19.12	800	0.0575	1847.772
30	28.67	1200	0.0604	1847.772
60	57.56	2400	0.0595	1847.772
600	577.87	25000	0.0599	1847.772

Name: 100_23_i100
Pedigree size: 23
Number of Markers: 100
Locations per Marker: 1

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
1	0.95	40	0.0555	1847.772
10	9.36	400	0.0593	1847.772
20	19.06	800	0.0596	1847.772
30	28.94	1200	0.0597	1847.772
60	56.06	2400	0.0604	1847.772
600	579.64	25000	0.0598	1847.772

Name: 100_23_i200
Pedigree size: 23
Number of Markers: 100
Locations per Marker: 2

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
1	0.96	40	0.0586	32.328
10	9.37	400	0.0504	75.14
20	18.62	800	0.0583	1847.772
30	28.15	1200	0.0516	75.14
60	56.89	2400	0.0602	1847.772
600	581.65	25000	0.0597	1847.772

Name: 100_23
Pedigree size: 23
Number of Markers: 100
Locations per Marker: 3

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
1	0.9	20	0.0786	1847.772
10	9.83	400	0.0606	1847.772
20	19.59	800	0.0600	1847.772
30	29.95	1200	0.0596	1847.772
60	57.83	2400	0.0601	1847.772
600	586.08	25000	0.0600	1847.772

Name: 110_22_i1
Pedigree size: 22
Number of Markers: 110
Locations per Marker: 0

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
1	0.91	39	0.0488	1331.457
10	9.64	350	0.0598	1334.967
20	19.33	700	0.0660	1334.967
30	28.85	1050	0.0631	1334.967
60	59.42	2200	0.0605	1334.967
600	577.40	22000	0.0612	1334.967

Name: 110_22_i100
Pedigree size: 22
Number of Markers: 110
Locations per Marker: 1

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
1	0.92	39	0.0637	1334.967
10	9.75	350	0.0585	1334.967
20	19.44	700	0.0645	1334.967
30	29.62	1050	0.0607	1334.967
60	59.10	2200	0.0635	1334.967
600	590.55	22000	0.0615	1334.967

Name: 200_12_i1
Pedigree size: 12
Number of Markers: 200
Locations per Marker: 0

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
10	8.39	59	0.2009	2843.468
60	59.69	300	0.2132	1519.137
600	598.86	3000	0.1835	1519.137

Name: 200_12_i100
Pedigree size: 12
Number of Markers: 200
Locations per Marker: 1

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
10	8.34	59	0.1623	4169.848
60	59.23	300	0.1693	1519.137
600	592.02	3000	0.1736	1519.137

Name: 200_16_i1
Pedigree size: 16
Number of Markers: 200
Locations per Marker: 0

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
10	8.77	59	0.1263	4566.11
60	59.41	300	0.2020	1080.912
600	593.67	3000	0.2022	1080.912

Name: 200_16_i100
Pedigree size: 16
Number of Markers: 200
Locations per Marker: 1

Time Bound (sec)	Actual Time (sec)	MCMC Iterations	Lod Score	Trait Location (Haladane cM)
10	8.66	59	0.1744	880.429
60	59.86	300	0.1902	880.429
600	598.62	3000	0.1997	1080.912

References

[1] V. Gogate and R. Dechtera. SampleSearch: Importance Sampling in presence of Determinism. Technical report, Tech. Rep., University of California, Irvine (under review at AI Journal), 2009.

[2] J.D. Terwilliger and J. Ott. *Handbook of human genetic linkage*. Johns Hopkins Univ Pr, 1994.