Libpll sequential benchmarks

November 23, 2016

1 Benchmark description

The following benchmarks compare several libpll implementations with different modes. They measure the execution time of a full likelihood computation on a fixed tree. To avoid measuring the initialization part, we repeat several times `pll_update_partials` and `pll_compute_edge_likelihood` on the same partitions and tree.

- xflouris means that the implentation used is this one: https://github.com/xflouris/libpll.
- bmorel means that the implementation used is this one: https://github.com/BenoitMorel/libpll. It supports sites repeats, and the data structure used is a bit different from xflouris (even without sites repeats): CLVs are not assumed to be sorted by sites, and an additional lookup table is used to access them in most of the core functions.
- bmorel2 is a (temporary) hacked version of bmorel where I do not use the new lookup structure for the scalers, because I think it slows down the execution, and the speed up with the sites repeats is not great.
- default mode means that the option `PLL_ATTRIB_PATTERN_TIP` and `PLL_ATTRIB_SITES_REPEATS` are unset.
- tip pattern means that the option `PLL_ATTRIB_PATTERN_TIP` is set.
- sites repeats means that the option `PLL_ATTRIB_SITES_REPEATS` is set.
- M is the size of the buffer allocated to compute the sites repeats class identifiers. When it increases, more nodes can benefit from sites repeats.

2 Summary of the results

Without sites repeats, bmorel's implementation with the additional lookup table can be from 0% to 10% slower than xflouris' one, especially with SSE and AVX architectures. bmorel2's hacked implementation (so without using the lookup for the scalers) is better, but can also be slower than xflouris'one.

In all versions, the tip pattern mode is around 1.5 faster than the default mode.

The sites repeats speed up is between 2 and 5 times faster than the tip pattern mode. It depends on :

- the dataset
- the architecture : it performs a bit better with CPU than with AVX and SSE
- the size of the sites repeats matrix buffer (M) : it performs better when the size increases

3 Benchmark

CPU architecture, 500 iterations

	seq59	seq128	seq404
xflouris default mode	9132.06 ms	115254 ms	141975 ms
bmorel default mode	8860.5 ms	120630 ms	145938 ms
bmorel2 default mode	8374.93 ms	113796 ms	144718 ms
xflouris tip pattern	5546.77 ms	72217.7 ms	93699.7 ms
bmorel tip pattern	5751.83 ms	75031.5 ms	93745.3 ms
bmorel2 tip pattern	5596.74 ms	72472.1 ms	92985.9 ms
bmorel sites repeats M=1000	4042.78 ms	44633.5 ms	63776.8 ms
bmorel sites repeats M=10000	3363.87 ms	36214.5 ms	44989 ms
bmorel sites repeats M=100000	2564 ms	29795.9 ms	28200.7 ms
bmorel sites repeats M=1000000	2441.83 ms	26135.1 ms	21914.5 ms

SSE architecture, 500 iterations

	seq59	seq128	seq404
xflouris default mode	3351.18 ms	46966.2 ms	56187.7 ms
bmorel default mode	3581.81 ms	51892.6 ms	60900.9 ms
bmorel2 default mode	3769.5 ms	50549.6 ms	60982.3 ms
xflouris tip pattern	2394.15 ms	33208.1 ms	38812.9 ms
bmorel tip pattern	2704.76 ms	36937.8 ms	41669.1 ms
bmorel2 tip pattern	2632.18 ms	34065.3 ms	40307.7 ms
bmorel sites repeats M=1000	1809.24 ms	23414.6 ms	28248.3 ms
bmorel sites repeats M=10000	1577.6 ms	19630.4 ms	21174.4 ms
bmorel sites repeats M=100000	1379.09 ms	17273.5 ms	15398 ms
bmorel sites repeats M=1000000	1355.33 ms	14931.4 ms	14625.3 ms

AVX architecture, 500 iterations

	seq59	seq128	seq404
xflouris default mode	3068.1 ms	42753.2 ms	51747.3 ms
bmorel default mode	3301.37 ms	47027 ms	55854.2 ms
bmorel2 default mode	3391.64 ms	48264.6 ms	55732.1 ms
xflouris tip pattern	2275.46 ms	32892.7 ms	36336.2 ms
bmorel tip pattern	2279.83 ms	33309.5 ms	38258.6 ms
bmorel2 tip pattern	2313.39 ms	32949.3 ms	36811.5 ms
bmorel sites repeats M=1000	1838.98 ms	21830.1 ms	27634.8 ms
bmorel sites repeats M=10000	1530.13 ms	18752.6 ms	21540.3 ms
bmorel sites repeats M=100000	1314.12 ms	15113.5 ms	16043 ms
bmorel sites repeats M=1000000	1345.76 ms	14301.3 ms	13805.9 ms