Faster Brute Force Proofs

by Sven Nilsen, 2020-2021

In this paper I suggest an idea for speeding up brute force propositional logic and at the same time allow easier modeling of path semantics.

By separating propositions[1] into two groups `F` and `X`, one can speed up brute force[2] propositional logic by enumerating only cases that satisfies the core axiom[3] of path semantics. This is possible because when `F(X)`, `X` is uniquely associated with `F`[4].

The core axiom of path semantics might be modeled in propositional logic, for more than two symbols, under the assumption that all propositions in `F` are true or all propositions `X` are equal.

Examples:

F	X	
0100	1001	`F` has a zero, `X`s are not equal (skip case)
0100	0000	`X`s are equal (keep case)
1111	0110	`F` are ones (keep case)

Here is a table with fractions measured by counting cases:

|F| \ |X|	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
2	0.625	0.5625	0.53125	0.51563	0.50781	0.50391	0.50196	0.50098	0.50049
3	0.4375	0.34375	0.29688	0.27344	0.26172	0.25586	0.25293	0.25146	0.25073
4	0.34375	0.23438	0.17969	0.15234	0.13867	0.13184	0.12842	0.12671	0.12585
5	0.29688	0.17969	0.12110	0.09180	0.07715	0.06982	0.06616	0.06433	0.06342
6	0.27344	0.15234	0.09180	0.06152	0.04639	0.03882	0.03503	0.03314	0.03220
7	0.26172	0.13867	0.07715	0.04639	0.03101	0.02332	0.01947	0.01755	0.01659
8	0.25586	0.13184	0.06982	0.03882	0.02332	0.01556	0.01169	0.00975	0.00878
9	0.25293	0.12842	0.06616	0.03503	0.01947	0.01169	0.00780	0.00585	0.00488
10	0.25146	0.12671	0.06433	0.03314	0.01755	0.00975	0.00585	0.00390	0.00293
11	0.25073	0.12585	0.06342	0.03220	0.01659	0.00878	0.00488	0.00293	0.00195
12	0.25037	0.12543	0.06296	0.03172	0.01611	0.00830	0.00439	0.00244	0.00146
13	0.25018	0.12521	0.06273	0.03149	0.01587	0.00805	0.00415	0.00220	0.00122
14	0.25009	0.12511	0.06261	0.03137	0.01575	0.00793	0.00403	0.00207	0.00110
15	0.25005	0.12505	0.06256	0.03131	0.01569	0.00787	0.00397	0.00201	0.00104
16	0.25002	0.12503	0.06253	0.03128	0.01566	0.00784	0.00394	0.00198	0.00101

References:

[1] “Propositional calculus”
Wikipedia
https://en.wikipedia.org/wiki/Propositional_calculus

[2] “Pocket-Prover”
AdvancedResearch
https://github.com/advancedresearch/pocket_prover

[3] “Path Semantics”
Sven Nilsen, 2016-2019
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/path-semantics.pdf

[4] “Philosophy of The Core Axiom”
Sven Nilsen, 2020
https://github.com/advancedresearch/path_semantics/blob/master/papers-wip/philosophy-of-the-core-axiom.pdf