CS433 Assignment 2

Parth Sastry 180260026
Sankalp Gambhir 180260032

September 25, 2020

As we see in Figure 1, the runs with `score = false` come out on top at the start, but begin to dwindle with more complex problems. This suggests that the score calculations add unnecessary complexity to smaller problems and removing it adds to the performance, while in large problems with more conflicts it would help settle in on a result quicker.

We are not particularly sure about the `scorefactor` parameter based results. We got different results for the times while changing `scorefactor` even though we had asked to not use EVSIDS scores, consistently across tests we performed. The lower `scorefactor` performed better here. Based on these and the paper on techniques used here by Biere et al, creators of CaDiCaL that this infact corresponds to the damping factor f used in the paper.

Across several runs to verify our results, we did not observe much change in time compared to these results. Here, we see that the reds, 600 and 1000 both perform about the same, with the blues 500 and 900 being a bit better, and black/browns 700 and 800 performing better than the rest. While our data is limited by the scope of available computing power, this does suggest that values of `scorefactor` on either ends of the spectrum (allowed values 500 to 1000 per documentation) seem to degrade performance, perhaps by causing too much decay and flattening the scores, or causing too little, and letting scores build up.

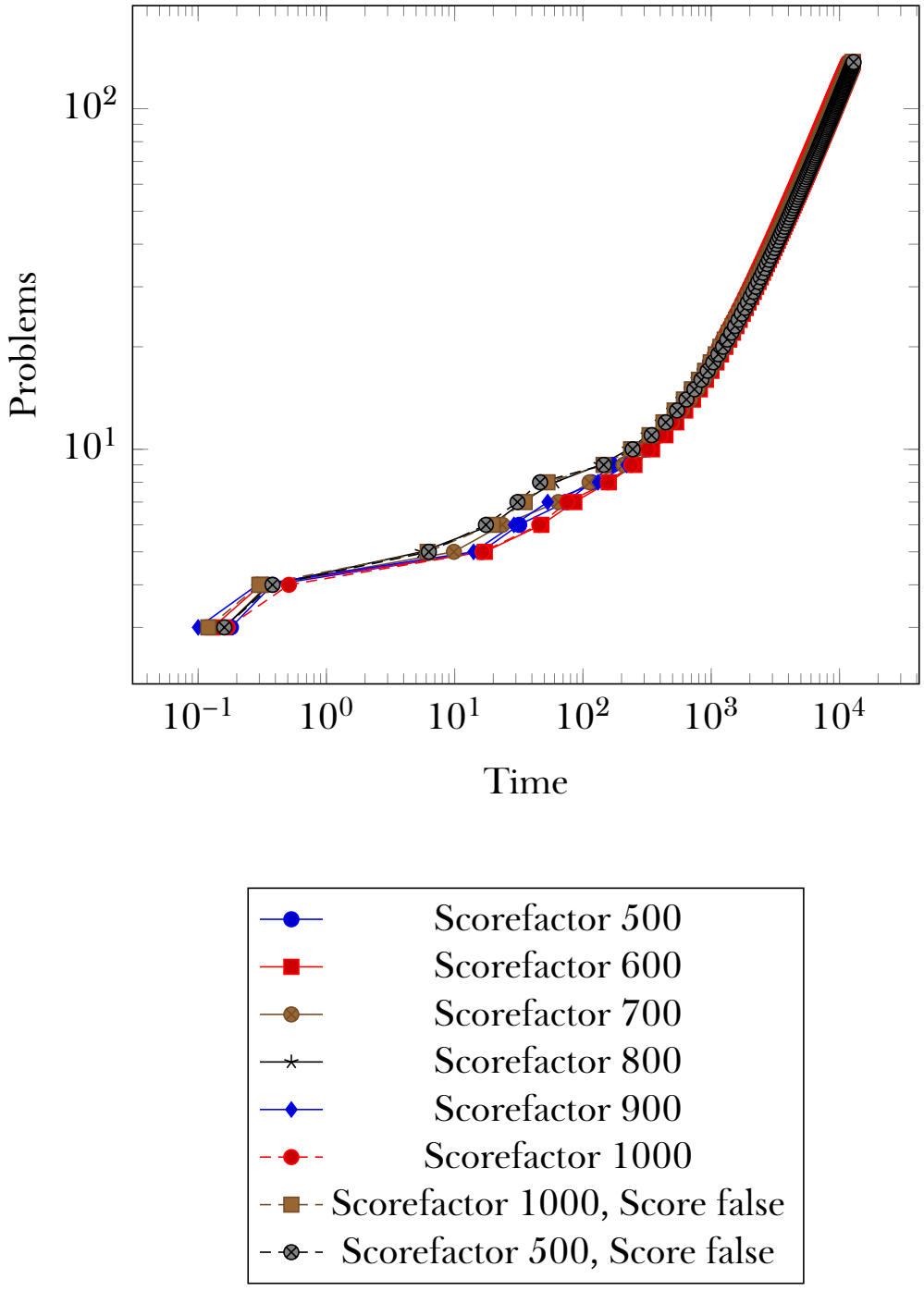

Figure 1: Time taken versus problems solved