A restart strategy adaptive to the average conflict level

1 Motivation

When a CDCL-based solver searches for a solution, the search is restarted after a certain number of found conflicts. Often a simple geometric function is used for the growth of this number to restart more often in the beginning of the search and less often as the search goes on. This makes deep runs into the search space possible but also effectively eliminates restarts after some time.

The approach that is described in this paper still uses a geometric growth-function but tries to weaken or at least postpone the elimination of restarts by taken into account the average conflict level.

2 Main idea

The average decision level on which conflicts are found ("average conflict level" or "ACL" from now on) is a useful indicator for determining how far the search progresses into the search space. If this number gets lower, the search is stuck in an early part of the search space and doesn't many useful conflict clauses. Since the intervals between restarts are growing with each occured restart, these being-stuck-parts are getting longer as well and thus slowing down the search.

If we can determine effectively this parts, we can keep the intervals smaller to speed-up the search and getting deeper into the search space again.

3 Implementation

If a restart is scheduled, a geometric restart strategy would normally simply multiply the last used interval by a fixed growth factor greater 1. With this approach we have another option - to make the interval smaller instead of bigger - based upon the ACL.

To calculate the ACL, we need to sum up all conflict levels of the current interval as they occur. If the restart strategy schedules a new restart, we determine which action is to be taken by

$$\frac{S_{conflictlevels}}{L_{interval}} < \frac{S'_{conflictlevels}}{L'_{interval}} \times t_{ACL}$$

$S_{conflictlevels}$ ... sum of all conflict levels of current interval
$S'_{conflictlevels}$ ... sum of all conflict levels of previous interval
$L_{interval}$ ... length of the current interval
$L'_{interval}$ ... length of the previous interval
t_{ACL} ... threshold-factor for the ACL-factor

The threshold-factor determines how much the ACL has to shrink between to intervals to effect a different growth for the next interval. Which is also visible at this point is that we only look

at the last two intervals. The reason behind this is the fact that the ACL tends to grow larger in the long run and a comparison of the ACL of the current interval with the ACL of the whole search would almost never fulfill the criteria.

4 Different growth functions

When the criteria is fulfilled, the growth of the restart interval is changed from the "standard" approach. The main question ist now: *How* do we change it?

There are many possible answers to this question and only a few are explored in this paper.

The first tested approach was to "stall" the growth by simply retaking the interval. Another one was to take a step back and take the previous interval, effectively speeding up the restart strategy. Also a precycle-phase, i.e. a number of restarts without any growth in the interval, was added.

Another approach is a more dynamic one which takes the ACL-ratio as growth factor for the new interval. Polynomial and linear growth functions were tested as well.

5 Benchmarks

The different growth functions were tested on the SAT-Competition and ASP-Competition of the year 2009. The used computer has 4 dual-core processors in hyperthreading 3.40 GHz and 32 GB RAM. Unfortunately, the results were lost in head crash but will be restored soon.

5.1 SAT-Competition '09

5.2 ASP-Competition '09

5.3 Results or "magic numbers"

On the given benchmarks, the stalling function worked a little better than the shrinking function, medium ACL-factors worked better than high or low and the precycle-phase helped in general. The combination of this made the stalling funktion with 0.8 as ACL-factor and a precycle-phase of 10 the best combination. All approaches yielded better results than the standard approach of a purely geometric growth. Solely the purely dynamic growth didn't work as well, which might be based on a missing overall growth of the interval (statistically, the ratio should swing around 1.0).