Effects of Search Depth and Schedule Placement on WRR Schedules

Sponsor: Sprint

Jason Keimig
David W. Petr

Technical Report TISL-11230-02

Telecommunications and Information Sciences Laboratory
Department of Electrical Engineering and Computer Sciences
University of Kansas

December 1996

Effects of Search Depth and Schedule Placement on WRR Schedules

Jason Keimig
David W. Petr
TiMCAN Project

December 31, 1996

1 Overview

Weighted Round Robin (WRR) scheduling is a means by which resources can be allocated non-uniformly among many requesting agents. The crux of the WRR scheduling algorithm involves parsing a one dimensional "schedule" that dictates the order in which client agents will be serviced. The number of entries a client has in the schedule in turn determines the percentage of access time to the shared resource that the client possesses.

Factors such as client request characteristics, number of active clients, client slot placement in the schedule, and processing capabilities of the scheduler itself can degrade performance from theoretical expectations. This paper explores the effects of all four of theses factors (although not exhaustively) in terms of delay and buffering requirements.

2 Introduction

The application of a WRR scheduler in this analysis relates to bandwidth management control, with clients acting as network sources. The static parameters of the experiments conducted are as follows:

- Network layout is a multi-node to single node configuration. The multi-node consists of 16 source models of VBR (video) traffic. The single node is the WRR scheduler.
- Link rates from all sources to the WRR node are equal at 45 Mbps.
- The output link rate of the WRR scheduler is also 45 Mbps.
- The WRR schedule contains 16 slots - 1 for each source.
- The traffic sources are derived from 4 333-second video traces. The video traces are:
  - Standard JPEG encoding of the movie "BladeRunner".
  - IBBPBB MPEG encoding of the movie "BladeRunner".
  - Standard JPEG encoding of broadcast news from CNN.
  - IBBPBB MPEG encoding of broadcast news from CNN.

The parameters that will be adjusted throughout the course the experiment are as follows:

- WRR Schedule search depth (to simulate limited scheduler processing power).
- Number of *active* sources in the schedule.

- Schedule layout wrt active sources.

Finally, are results were accumulated through the use of the Weighted Round Robin Simulator (WRRS) version 1.0.0. WRRS is a cell-level network analyzer that measures network effects on ATM cells at the switch level. This is a time-based simulator in that each network component works on a common "tick" time interval. This simulator was developed by the author.

3 Parameter Analysis

We will look into the effects of schedule search depth, spacing between active sources in the schedule, and loading effects in the following analysis. Common source types (i.e. video) were used to maintain a *somewhat* consistent injection of cells (i.e. medium-sized bursts separated by long pauses).

The analysis will be quantized on the basis of three network characteristics:

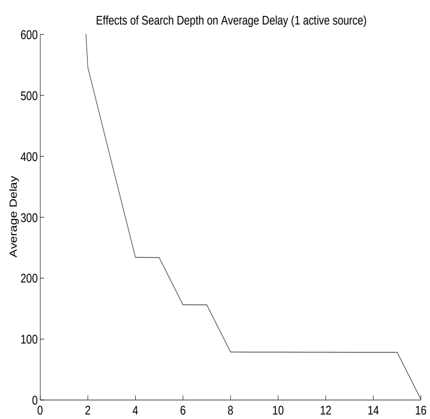
- Average Cell delay in terms of ticks.
- Maximum Cell Delay in terms of ticks.
- Maximum Buffer Size in terms of cell counts.

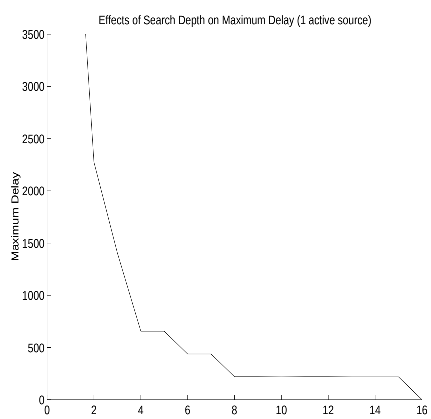
3.1 Search Depth Baseline

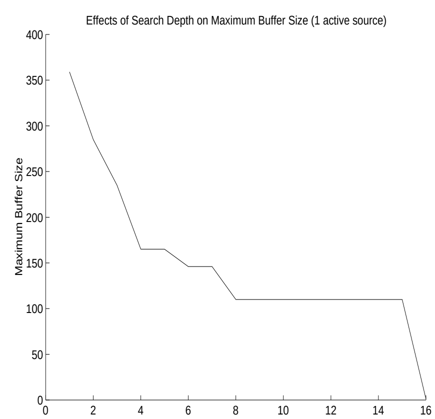
The term "Search Depth" signifies the *number of empty queue schedule slots that the scheduler can parse within a single cell inter-arrival time slot*, where this time is defined from the output link rate of the WRR scheduler.

To measure the theoretical effects of search depth limitation on a WRR scheduler, we will use a single active source in the multi-node connection, with all other sources "quiet". Since there are no loading effects from other sources, and because the line rates are matched from source ⇒ WRR ⇒ network output, we will have no accumulation of cells in the schedulers input buffers outside of the search depth effects.

The following three graphs show the *theoretical* effects of a limited search depth:







These graphs clearly show that the effects of search depth have well-defined levels of performance in all three categories of measurement. It should be noted, however, these effects are a function of the search depth *and* schedule length. This can be seen by examining the basic structure of the WRR scheduler and the operations that it performs over a given "tick" interval.

Given that the scheduler performs the following sequence of actions, with a defined search depth limit of length n:

1. If current queue in schedule has waiting cells stop, and service queue.
2. If current queue in schedule is empty, move to next item in schedule.
3. If we have moved through n empty schedule slots, stop.
4. Else, goto 1.

From this, we see that the scheduler can look up to n number of *empty* schedule slots before it must terminate. However, if a *single* queue if found to be active (non-empty), then the search must stop at this point. Hence, we can derive a function of the *minimum* number of ticks that a schedule slot can be serviced:

$$T_{min}(n, L) = \lceil \frac{L}{n} \rceil \tag{1}$$

where T is the number of ticks required, n is the imposed search depth, and L is the length of the schedule.

Therefore, we should see steps at points where the whole part of the L versus n ratio changes; and in the case of $L = 16$, we should see 5 such changes. The results agree with this derivation. It is interesting to note that a search depth of 8 is equivalent to a search depth of 15 which is 1 less than the highest attainable search depth.

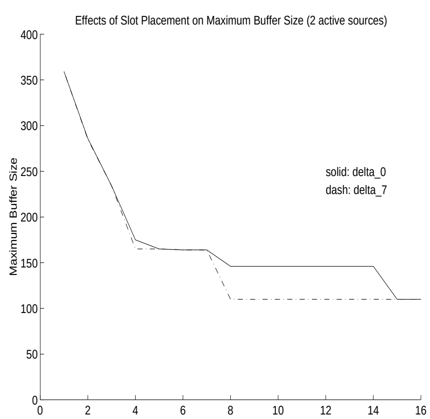
3.2 Effects of Slot Placement

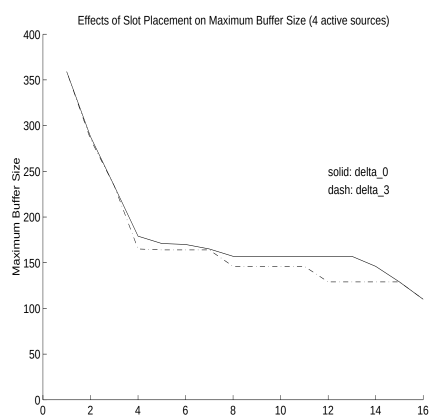
The next parameter under examination will be the effects of slot placement between active sources in the schedule. That is, analysis will now be emphasized with respect on how far apart active sources are located from each other within the context of the WRR schedule.

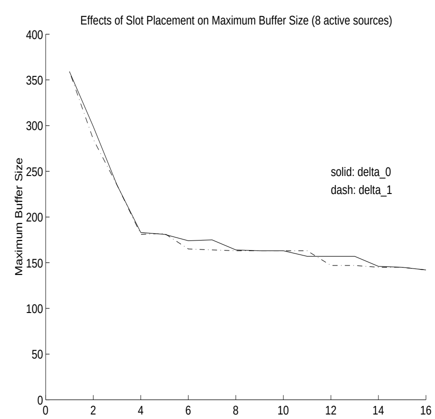
Tests were conducted (using the same sources) with 2, 4, and 8 active sources. For each set of active sources, search depth was varied from 1 schedule slot to L (16) schedule slots over a given schedule. The schedule was changed after each run, moving the active sources farther apart, up to the point that (all) sources were equidistant from each other in the schedule.

It should be noted that due to symmetry in the wraparound feature of the scheduler, there was no need to go beyond the equidistant spacing of the sources. Also, the spacing was performed relative to the first slot in the schedule. As a final note, for the cases where I had 4 and 8 active sources, I used equal spacing between all of the active sources in the schedule. I could have used a binary approach for an exhaustive set of tests, but theses tests would have been bounded by the zero-spacing (all active sources next to each other) and equi-distant spacing of the active sources.

For brevity, I include only the results from the maximum buffer size measurements of each of the configuration types (2,4,8 sources):







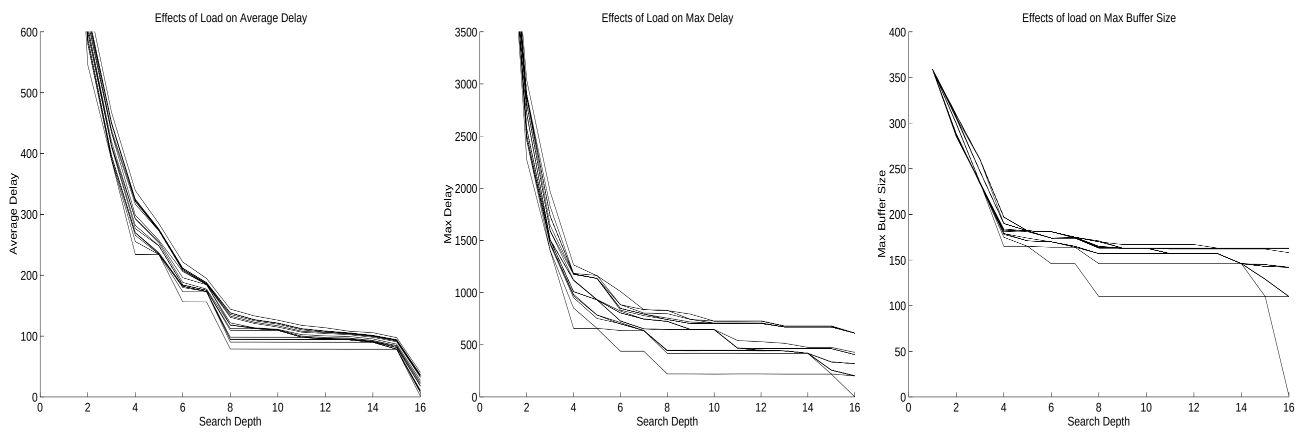
Here again, we see the effects of search depth at defined boundaries, but this time there are more drops and less plateaus (more prevalent in the cases of 4 and 8 active sources). This is mainly due to loading effects.

However, we can point out another interesting characteristic in the differences between zero-spacing and maximal spacing. That is, we see that the measurements are *lower* bounded by maximal spacing and *upper*-bounded by zero-spacing. Without a presented proof, the reader should be lead along the lines that the *average* number of ticks to reach the next active source for the maximal spacing is *lower* than the case for zero-spacing.

3.3 Load effects on Search Depth

The final set of data collected centered around loading effects on the performance costs imposed by search depth. The parameters here involved only "turning on" an additional source for each set of search depth simulations. It should be noted that sources were turned on in the order in which their corresponding queue IDs were represented in the schedule. Hence, a zero-spacing was used for all of the active sources as they were turned on.

The following is an illustration of the loading effects experienced by the first source in the schedule, which was the only source active throughout the entire experiment. The output shown on the graphs should be interpreted as the lower bounding line is the effect of only one source active, while the upper bounding line is the effect of all 16 sources active:



The effects shown in the first graph (average delay) exemplify the averaging of required tick times for zero-spacing as discussed in the previous section. This is noticeable on the plateaus of the graph where the more heavily loaded plots *tend* toward the theoretical plot of the single active source. This is obvious in the range from 8 to 15 search depth allowances where the heaviest loaded plot (16 sources) drops nearly 35terms of average delay.

4 Conclusion

As seen from the conducted experiments, there are several aspects to consider in determining performance of WRR schedulers in physical systems. The following factors must be considered:

1. Length of the Schedule: L
2. Allowable schedule search depth: n
3. Spacing between active sources: d
4. Total number of active sources: s

Something that should always be considered are the types of sources introducing traffic into the network. This parameter is one which is very hard to quantize, and is outside the scope of this paper. The source types chosen in these experiments were of the same format (video), to *try* to keep this parameter on a relative scale, but it is yet to be determined if even video sources exhibit similar properties.

The overall analysis shown here is interesting for the case of a lightly loaded network. With a heavier loaded network, we should see the effects of the search depth and active source spacing drop off to give delay and buffering characteristics analogous to that of normal WRR analysis.

The information here gives insight as to the possible structuring of the WRR schedule in cases where there are fewer than half of the sources in the schedule that are active. More precisely, we should consider the given analysis when there are fewer then half of the total schedule entries that are *active*.