COSC-450 Final Project Writeup
Fairness in Single Resource Environment

December 18, 2019

Lesley Zheng, Juhwan Jeong

Scheduling Policies Acronym

FB	Foreground background
FCFS	First come first serve
JTF	Jump to front (combines FCFS and PLFCS)
LCFS	Last come first serve
LJF	Longest job first
LRPT	Least remaining processing time
PS	Processor sharing
PSJF	Preemptive shortest job first
PLCFS	Preemptive last come first serve
PLJF	Preemptive longest job first
SJF	Shortest job first
SRPT	Shortest remaining processing time

1. Introduction

1.1. Motivation

The initial motivation for this project came from one of the senior computer science thesis talks I [Lesley] attended last semester. The thesis student worked with Professor Gardner to analyze scheduling policies in homogenous and non-homogenous service farms. At one point in the talk, the student explained that one policy is better than another because it had fewer small jobs wait for a large job -- a notion of fairness known as proportional fairness. I remember being perplexed by how a vague and abstract concept of fairness could be defined so simply. I reached out to Professor Gardner and she pointed me to several papers [paper #s in Sources] and I realize there is actually a plethora of research into queueing fairness.

We decided to pursue this topic for our final project because it allows us to use the tools we have learned in class to tackle a familiar queueing setting from a completely different perspective. The setting is a single server queue, which is what we started our study of queueing theory with. The equations for fairness build on an understanding of the performance for queues, such as mean response time, queueing time, and throughput. The criterion of fairness requires a blend of qualitative and quantitative explanations.

In this final project, we seek to explore the multitudes of the notion of fairness and paint a picture of the state of the art of fairness in single server queues.

1.2. Fairness

As a broad and abstract concept, fairness has multiple important attributes in queues. Back in 1987, in *Perspectives on Queues: Social Justice and the Psychology of Queueing*, Richard Larson organizes these thoughts into the following sections:

- Social Justice: adherence or violation of first in first out to correspond to obtaining or not obtaining social justice. Larson uses the example of a woman he met at a local department store who was "on the verge of tears" because she had been waiting for 30 minutes for her merchandise, while seeing "numerous other customers... come and gone" with their purchased items (Larson 895).
- Skips and Slips: more specifically, the source of violations to the first in first out principle are slips and skips. A job A is victim to a slip if another job B that arrives after A enters service before A. B is said to have experienced a skip. Queueing theorists and social scientists have long agreed that "FCFS is the socially just queue discipline and… system discipline" (Larson 896).
- Environment: positive and entertaining environment generally improve the queueing experience, and thus reduces the burden on stress from potential queueing unfairness.
- Eliminating Empty Time: even if there were no specific violations of social justice, the waiting itself can be though as "unfair." Larson quotes a *Times* essayist: "Waiting is a form of imprisonment. One is doing time -- but why? One is being punished for an offense of one's own but one for the inefficiencies of those who impose the wait" (Larson 897).
- Feedback: generally, providing information or feedback on the expected wait time improves the queueing experience.

In 2002, A. Rafaeli, G. Barron and K. Haber investigate how the attributes of queue structure people's psychological perception of fairness in *The Effects of Queue Structure on Attitudes*.

- Queue Structure: They ran an experiment with participants, where each participant interacted with a computerized representation of their progress in the wait while actually waiting to get in the laboratory to participate in an unrelated experiment. Every participant waited for 12 minutes, but their screen displayed their progress in terms of either a single queue or multi-queue. They found that generally, participants waiting in the multi-queue felt more emotions of "unfairness" than those who waited in a single queue, even though both had the same wait time.

When researchers in the computer science community started developing work on fairness in queues in the early 2000s, they generally organized their work under two intuitive notions of fairness:

- Temporal fairness: if A and B are two jobs of the same size and A arrives before B, then it is temporally fair to have A receive service before B.
- Proportional fairness: if A is a really small job and B is a really large job and A arrives slightly after B, it is proportionally fair to have A receive service before B.

Compared to other attributes of fairness, temporal and proportional fairness are easier to quantify, and thus allow further quantitative analysis. Beginning mid-2000s, researchers started to think of creative ways to combine the two.
In section 2 of this write up, we present previous research on fairness grouped under proportional, temporal, then combined notions of fairness.

1.3. Relevance

The issue of fairness in queues is important because it is clearly an issue that we care about. If left unaddressed, any development in the research community of a scheduling policy in terms of traditional performance metrics will find hard to be integrated into use in practice.

In *Job Fairness in Non-Preemptive Job Scheduling*, G. Sabin, G. Kochhar and P. Sadayappan discuss how even though in non-preemptive setting, SJF scheduling policy is known to achieve the best average slowdown than FCFS, but “system administrators generally prefer some variant of a FCFS policy...because of their concerns over issues such as job starvation and fairness” under SJF.

As important as it is to improve traditional performance metrics, it is equally important to investigate how policies do measured by fairness, so the improvements can actually be implemented into practice and improve people’s day to day lives.

2. Summary of Previous Work

2.1. Proportional Fairness

Given the amorphous nature of the concept of fairness -- ranging from considerations of social justice to the queueing design (see Section 1.2) -- the study of fairness in the computer systems community has been rather sporadic until recently.

In 2001, N. Bansal and M. Harchol-Balter published a paper *Analysis of SRPT Scheduling: Investigating Unfairness* that formally investigated and dismissed notions of proportional unfairness of SRPT as compared to PS. The paper uses slow down as its fairness metric, where for any job of size x with expected response time $E[T(x)]$, the expected slow down $E[S(x)]$ is defined as (Bansal and Harchol-Balter 6):

$$E[S(x)] = \frac{E[T(x)]}{x}$$

Clearly, slowdown is a fairness metric that aligns with the proportional notion of fairness, because it scales appropriately for small and large job sizes. More formally, since $E[T(x)] = \theta(x)$ under all work conserving scheduling policies, $1/x$ is a non-trivial normalizing factor that allows comparisons across different x’s (Wierman 41).

However, it wasn't until their paper in 2003 *Sized-based scheduling to improve web performance* along with B. Schroeder and M. Agrawal that really drew the computer systems community to the issue of fairness. In this paper, Harchol-Balter et al. specifically discussed an SRPT-based design for web servers and continued to dispel worries of starvation under SRPT (Wierman 41). This began the recent focus on fairness and the growing literature of proportional fairness in terms of slow down for all different scheduling policies.

More formally, proportional fairness in terms of slow down is generalized as follows in *Classifying scheduling policies with respect to unfairness in an M/GI/1*:

2.1.1. Mean Conditional Slowdown

$$\frac{E[T(x)]^P}{x} \leq \frac{1}{1-\rho}$$

Where ρ is the load and $0 < \rho < 1$, P is the scheduling policy, and the system is any M/GI/1. A scheduling policy P is fair if every job size obeys the inequality. If any job size x violates the inequality, it is considered to be treated unfairly, and so the scheduling policy P is unfair (Wierman 41).

Notice that the right-hand side of the inequality is the slow down for all job sizes under PS. PS is known to be the proportionally fairest scheduling policy because all jobs have the same slow down regardless of their sizes (Bansal and Harchol-Balter 6). So it is a natural choice to use that as the criterion to determine proportional fairness.

Under this definition of fairness, researchers found that contrary to popular belief, SRPT is not as bad as a scheduling policy. In *Analysis of SRPT Scheduling: Investigating Unfairness*, Bansal and Harchol-Balter finds that specifically,

- There exist job size distributions such that every job does better under SRPT than under PS (Section 5, Claim 1) (Bansal and Harchol-Balter 8).

For example, for the job size distribution $BP(k,\ p,\ \alpha = 1.1)$ at load 0.9 in Figure 1, every job does better under SRPT than under PS.

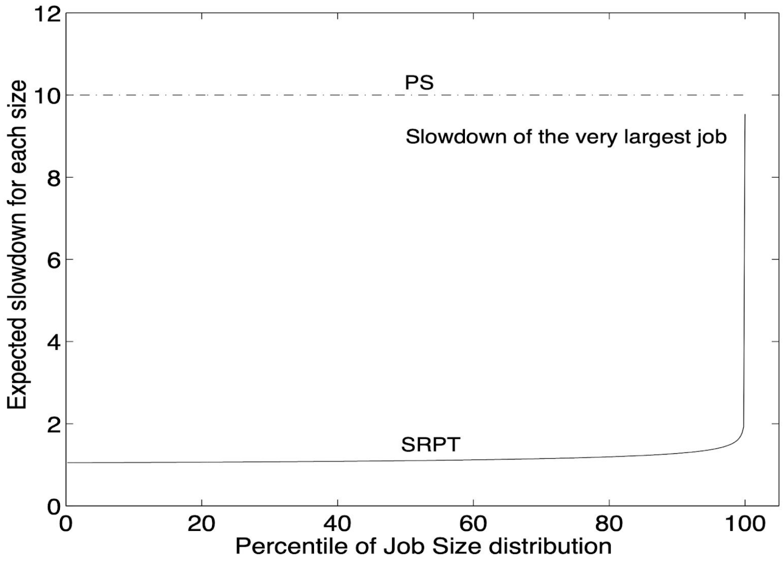

Figure 1. Expected Slowdown versus Job Size Percentile of PS and SRPT.

In Figure 1, job size expressed in terms of its percentile of the job size distribution is plotted against its expected slowdown. Every job size under SRPT has a slowdown below 10 units, which is the slowdown for all job sizes under PS. In fact, most job sizes under SRPT are much smaller -- below 2 units of expected slowdown -- and even the largest job has only 9.54 units, which is below the 10 units of PS.

Unless load is close to 1, most heavy-tail distributions have every job do better under SRPT than under PS in terms of slow down (Bansal and Harchol-Balter 8). So under these specific setting, SRPT is proportionally fair.

2.1.2. Classifying Policies by Fairness: Always Fair, Sometimes Unfair, Always Unfair

More specifically, we can qualify how fair a policy is by assessing how the policy adheres to the mean conditional fairness definition under different settings. In the 2003 paper *Classifying Scheduling Policies with Respect to Unfairness in an M/GI/1,* A. Wierman and M. Harchol-Balter apply this definition while varying the load and service distribution to classify policies into three types: Always Fair, Sometimes Unfair, and Always Unfair. A policy is Always Fair if it is fair under all loads and service distributions. A policy is Sometimes Unfair if it is fair under some loads and service distributions and unfair in others. A policy is Always Unfair if it is unfair under all loads and service distributions. The authors classified many of the well-known policies accordingly.

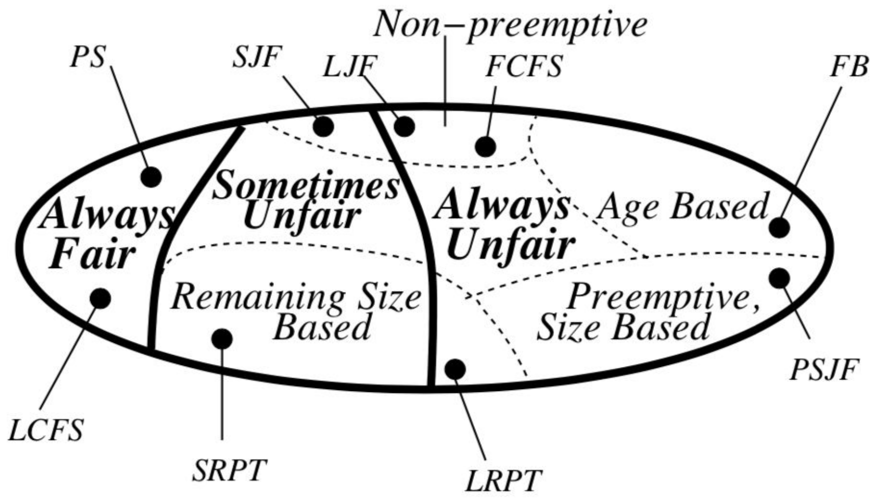

Figure 2. Classification of policies under proportional fairness definition.

Figure 2 classifies some common scheduling policies under the three types. “Always Fair” has the least number of scheduling policies: PS and LCFS. “Sometimes Fair” include SJF, LJF, and SRPT. “Always Unfair” is the most popular type, with LJF, FCFS, FB, PSJF, and LRPT.

This paper is significant for its pioneering attempt to classify all common scheduling policies by a mathematically grounded fairness definition.

2.2. Temporal Fairness

To this point, we have discussed developments of proportional fairness in isolation to temporal fairness. We found that temporal fairness is discussed less often than proportional fairness -- that is probably because the only scheduling policy that strictly obeys temporal fairness is FCFS. Therefore, this section will be rather brief.

In *Fairness and Scheduling Policy in Single Server Queues*, Adam Wierman provides one temporal fairness definition known as politeness. The politeness of a job of size x under policy P is the fraction of its response time which the seniority of the job is respected (Wierman 45). A scheduling policy is impolite if under the job size distribution X and load ρ, $\lim_{x\to\infty} E[Pol(x)]^{P} = 1-\rho$, otherwise P is polite (Wierman 45). As one might guess, FCFS has politeness of 1. One important note is that unlike other fairness measures, higher politeness denotes higher fairness.

Similar to his prior work of classifying policies by proportional fairness, Wierman assigns politeness levels to some common scheduling policies.

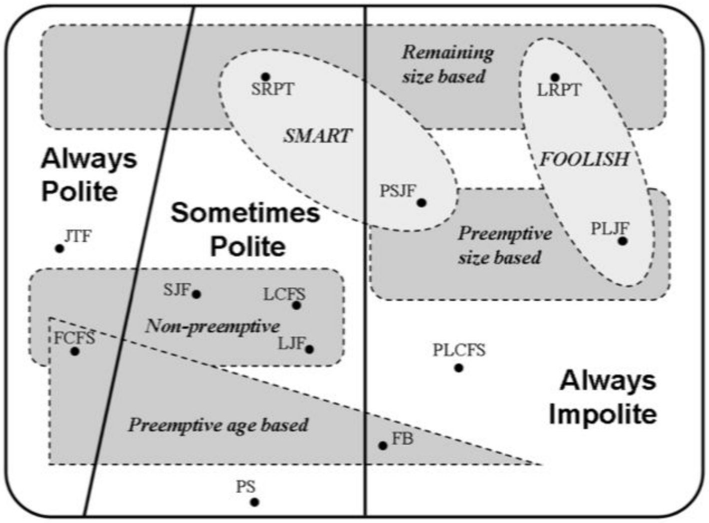

Figure 3. Classification of policies under temporal fairness definition.

Surprisingly, Figures 2 and 3 have a lot in common. For example, SRPT and SJF are classified as Sometimes Fair/Polite and FB and LRPT are classified as Always Unfair/Impolite. There are also disparities. For example, FCFS is classified as Always Unfair under proportional fairness, but under this temporal fairness definition, it is the most polite policy. We believe the disagreement in classifications highlight the necessity for more comprehensive measures - measures that take both proportional and temporal fairness into account.

2.3. Combined Fairness

In many scenarios, the proportional and temporal fairness definitions conflict. Imagine a scenario where job A with size 10 enters the system, followed shortly by Job B with size 1. Proportional fairness would rule that job B should be processed first, whereas temporal fairness would rule that job A should be processed first. Recognizing such conflicts and that focusing on a singular definition of fairness is insufficient, pioneering computer scientists began to develop combined fairness measures. We examine DF, SQF, and RAQFM in depth.

2.3.1. DF

In the 2005 paper *A Discrimination Frequency Based Queueing Fairness Measure with Regard to Job Seniority and Service Requirement,* W. Sandmann first introduces a combined fairness measure called discrimination frequency, or DF. In developing this measure, Sandmann states two principles that combined fairness measures should aim to meet: the strong service-requirement preference principle, which states that for every two jobs arriving at the same time the smaller one should be completed first, and the strong seniority preference principle, which states that for every two jobs in the system requiring the same service time the older one should be completed first (Sandmann 107). These two principles correspond to proportional and temporal fairness.

Based on these two principles, Sandmann introduces two components of DF: temporal discriminations and proportional discriminations. The temporal discrimination frequency (n) that a job J_i suffers from is the number of jobs that arrived not earlier than J_i and complete service not later than J_i. The proportional discrimination frequency (m) that a job J_i suffers from is the number of jobs not completely served upon arrival of J_i that have at least as much remaining service requirements and complete service not later than J_i (Sandmann 108). Discrimination Frequency is simply calculated as a sum of the two.

Due to its direct and intuitive nature, DF is used to analyze fairness of scenarios as exemplified by the paper. However, DF has a downside; all discriminations are treated equal. For example, consider two different proportional discriminations where job A with size 1 are preceded by job B with size 10 and job C with size 100. Clearly, job C preceding job A is a more severe discrimination than the job B preceding job A. DF fails to recognize the different severities of discriminations - as such, it limits DF's capabilities as a fairness measure to be used for mathematical analyses, which is why other measures like SQF and RAQFM were introduced.

2.3.2. SQF

In the 2007 paper *SQF: A Slowdown Queueing Fairness Measure*, B. Avi-Itzhak et al. introduce a combined fairness measure called Slowdown Queueing Fairness Measure, or SQF. Similar to RAQFM, it is developed on the notion that in a perfectly fair system all jobs should have the same slowdown. The authors calculate individual discriminations as the difference between the ideal slowdown and each job's actual slowdown, aggregate the discriminations, and arrive at one value for the fairness of the system.

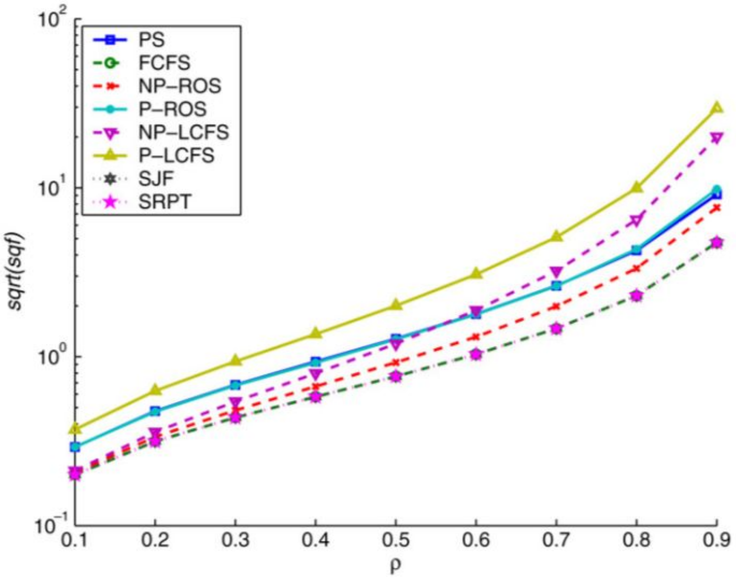

Figure 4. Graph of SQF versus system load for different policies.

Unlike DF, SQF scales individual discriminations by their respective severities, yielding a measure that can more accurately assess system fairness. Using SQF, the authors rigorously prove properties and bounds of systems under different policies. In Figure 4, the authors analyze relative fairness of different policies under varying loads. Interestingly, Figure 4 disagrees with Figure 2 in multiple aspects. For example, while Figure 2 classifies P-LCFS as Always Fair, Figure 4 shows that P-LCFS is actually the most unfair of the given policies under all loads. Such disparities come from the fact that SQF is a more comprehensive fairness measure than the proportional fairness definition used in Figure 2.

2.3.3. RAQFM

Resource Allocation Queueing Fairness Measure, also known as RAQFM, is a combined fairness measure discussed in the 2007 paper *A Resource Allocation Queueing Fairness Measure: Properties and Bounds* by B. Avi-Itzhak et al. It is also developed on the notion that at any point in a perfectly fair system, all jobs in the system should have a common property - in this case, equal amounts of the servicing power. This share of the server is called the warranted service rate. In a real system, however, many jobs do not have the warranted service rate. Hence, the authors again quantify the individual discriminations by calculating the deviation of each job's service rate from the warranted service rate and aggregate the deviations to output one fairness measure - RAQFM.

The paper draws our attention to important properties of RAQFM. The first property is that RAQFM is based on first accounting for the individual discriminations, allowing users of the measure to get a feel of the fairness encountered in system. The second is that the discrimination function of RAQFM has a locality of reference property; in other words, the discriminations of a job are relative to the jobs around it. Together these properties yield a combined fairness measure useful for mathematical analyses of systems and policies.

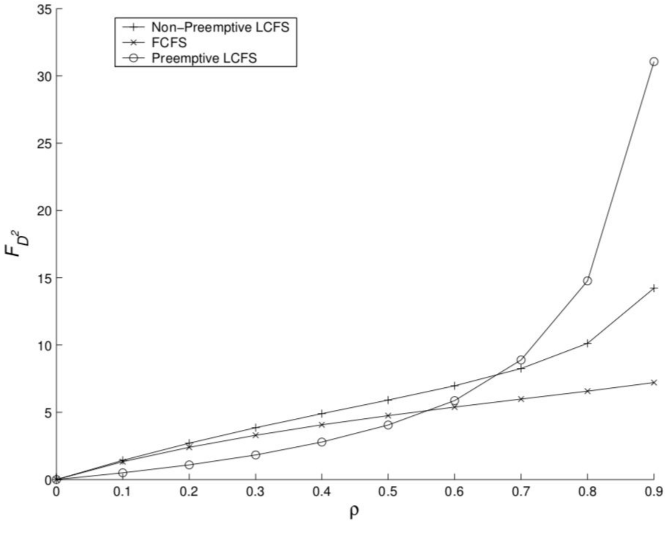

Figure 5. Graph of RAQFM versus load for multiple policies.

In Figure 5, we observe that the results agree with Figure 4 under high loads, where the P-LCFS is the most unfair, then NP-LCFS, and FCFS. This suggests that the two comprehensive measures agree even coming from different points of approach (constant slowdown and equal service rate), hinting that the ideas of perfectly fair system in each definition are mutual to an extent. However, under lower loads in Figure 5, the order of fairness changes whereas in Figure 4 the said observation holds under all loads. This disparity can also be attributed to the different points of approach. Coming from different assumptions about the perfectly fair system, the two measures provide different insights - contributing to a richer pool of combined fairness measures computer scientists can use.

3. Simulation

3.1. Motivation

There are three combined measures introduced so far: RAQFM, SQF, and DF. Of the three, RAQFM and SQF are used to analyze systems -- to draw mathematical bounds, properties, and generalizations about different settings. DF is used to analyze specific scenarios -- to determine which policy is the most adequate given the job sizes and arrival times. We wonder, what fairness results would we get if we analyze steady state systems with DF and how can we extend DF to further capture the notion of fairness in different settings? To study how we might use the measure to achieve fairness in different systems, we run a simulation.

3.2. Setup

In this simulation, we aim to study the fairness behaviors of multiple policies in a steady state system. To do so, we generate 1,000,000 jobs with a Poisson arrival process and an exponential service distribution. We also want a queue to build up and run the simulation in feasible amount of time, so all the simulations are conducted with a load of 0.6. The policies we simulate are: Random, FCFS, SRPT, and SJF.

We first simulate Random to observe trends if we made no use of any information (job sizes and seniority) -- the results are to serve as a basis point for comparison. Then, we implement FCFS, SRPT, and SJF to each represent classes of scheduling policies: FCFS represents non-size-based, non-preemptive policies, SRPT represents size-based, preemptive policies, and SJF represents size-based, non-preemptive policies. During each policy implementation, we calculate temporal discriminations (n) and proportional discriminations (m) for every job along with the job's service time.

3.3. Results and Discussion

To learn the relationship between job size and discrimination frequency, we graph discrimination frequency by job size for each policy.

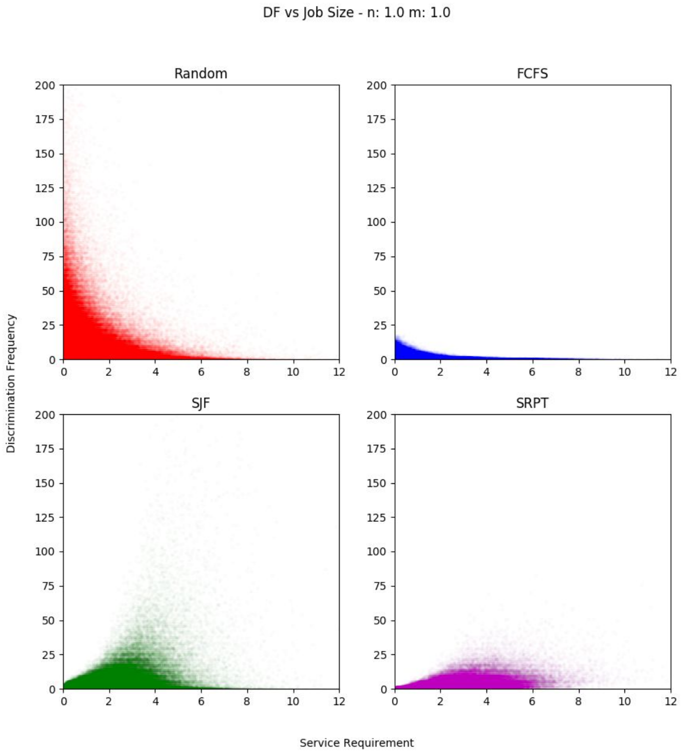

Figure 6. The graph of discrimination frequency by service requirement with equal weights on both types of discriminations

In Figure 6, Random has high discrimination frequencies that exponentially decrease as job size increases. We attribute this to the fact that when the next job is chosen arbitrarily, shorter jobs in the system are more likely to experience a proportional discrimination. Not only that, Random has temporal discriminations that result in the high discrimination frequencies.

In comparison, FCFS has a similar decreasing exponential shape -- but with less magnitude. This is due to the fact that FCFS also has the property that smaller jobs have a greater probability of being smaller than the next job to be put in service (by seniority). As a result, shorter jobs experience frequent proportional discriminations. In contrast to Random, however, FCFS has no temporal discriminations; by definition, FCFS strictly respects job seniority. This can be observed by the smaller magnitude of discrimination frequencies compared to the Random graph.

SJF, by definition, respects service requirements - this is rooted in the proportional fairness definition that smaller jobs should be processed first. This fact can be observed in the graph, in its increasing shape, where the larger jobs experience more frequent discriminations. In achieving proportional fairness, however, the policy does not account for job seniority in any means, suggesting that most of the discriminations are temporal.

SRPT works very similarly to SJF. One significant difference is the preemption of jobs -- when a new job arrives in the system and the job in server has a greater remaining service time than the new job, the new job is put into the server. This scenario is the only situation in which SJF policy introduces proportional discriminations in the system. In that sense, SRPT has zero proportional discriminations. It does, however, discriminate temporally more often than SJF precisely in the said scenario.

In first discovering the measure of discrimination frequency, we wondered -- why are proportional and temporal discriminations weighed equally? We can easily imagine a situation where one should be treated more severely than the other. For example, imagine the queueing system of an emergency room. We can agree that it is significantly more important to treat a patient with the more severe injuries than to treat a patient that came in a bit earlier. In this sense, we extend the idea of discrimination frequency and calculate the measure as a weighted sum of proportional and temporal discriminations. Note that we normalize the weights such that they always add up to 2 so that we can compare the discrimination frequencies of different weights.

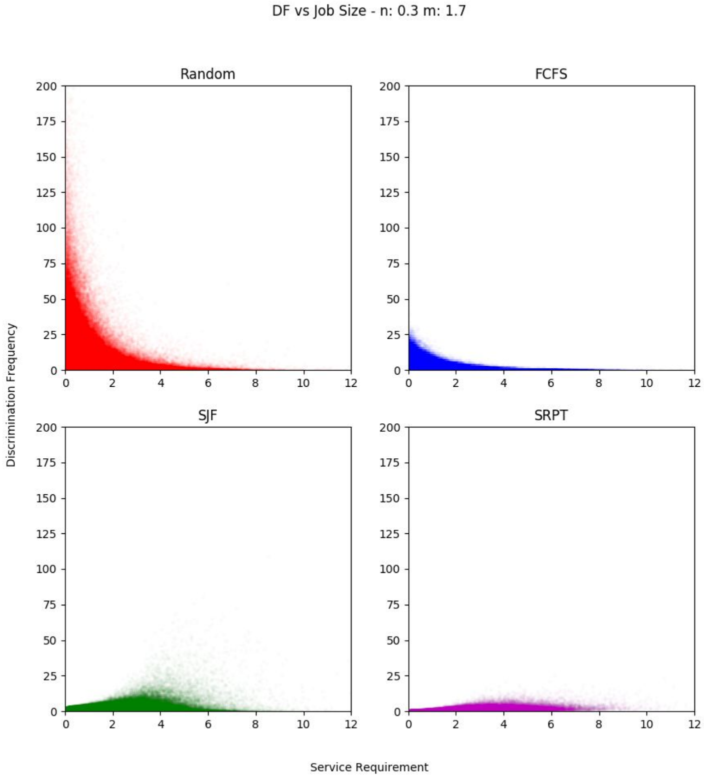

Figure 7. The graph of discrimination frequency by service requirement with greater weight on proportional discrimination

Figure 7 displays the results when proportional discrimination contributes greater to the overall discrimination frequencies than its temporal counterpart does. Compared to the graph in Figure 6 (with equal weights), there are significant changes.

In Random, there isn't a significant change in discrimination frequencies. This is due to the fact that there are approximately equal shares of proportional and temporal discriminations in Random systems so upweighting one and downweighting the other has minimal effect.

In FCFS, we observe that the discrimination frequencies increased slightly. This is due to the fact that all discriminations in FCFS are proportional so we expect the discrimination frequencies of each job to have increased by a factor of 1.3 (how much we upweighted proportional discrminations).

In SJF, we observe that the discrimination frequencies decreased significantly. This is due to the fact that the majority of discriminations in SJF are temporal so the downweighting of temporal discriminations has a greater effect on discrimination frequencies than the upweighting of proportioanl discriminations.

In SRPT, we observe the same effect as we did in SJF -- the only difference is that strictly every SRPT discrimination is temporal so the change in discrimination frequency is greater in SRPT than in SJF.

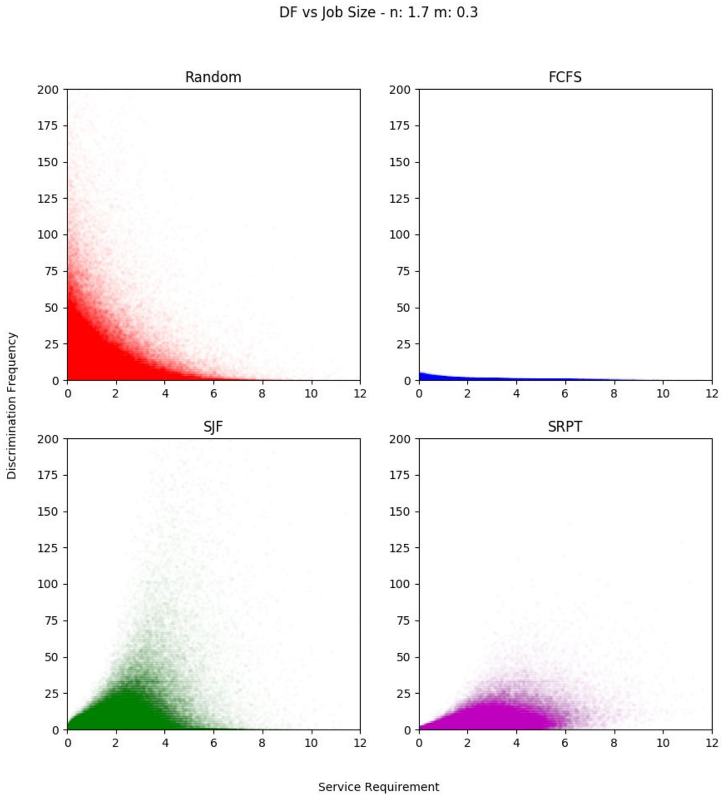

Figure 8. The graph of discrimination frequency by service requirement with greater weight on temporal discrimination

In Figure 8, we explore the opposite situation in which temporal discrimination is weighed heavier. One example of such situation is the queueing system of buying concert tickets. In its simplest form, such a system should respect seniority over size.
In Random, we observe no significant change once again.

In FCFS, discrimination frequencies decrease significantly due to the said fact that all discriminations in FCFS are proportional -- downweighing proportional discriminations reduce the overall discrimination frequencies.

In SJF and SRPT, discrimination frequencies increase significantly. This is also due to the said reason that most (all for SRPT) discriminations are temporal in SJF/SRPT. As a result, upweighing temporal discriminations increase the overall discrimination frequencies.

Having explored the relationship between discrimination frequencies and job sizes by policies and extensions of the measure, we begin to wonder - in systems under each policy, what shares of discriminations are temporal versus proportional?

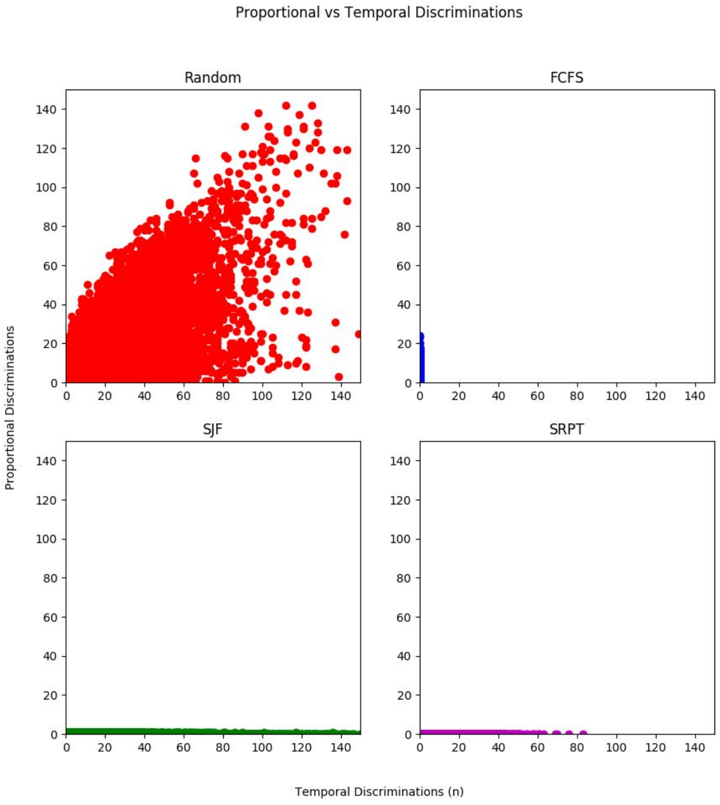

Figure 9. Relationship between temporal and proportional discriminations by policy

In Figure 9, we find unsurprisingly boring results. In Random, we find one significant result: jobs that experience frequent proportional discriminations tend to also experience frequent temporal discriminations. This is due to the fact that the longer a job stays in the system, the more likely it is that the job experiences more temporal and proportional discriminations. In FCFS, we observe, as we conjectured, that all discriminations are proportional. In SJF/SRPT, we again observe, as we conjectured, that most (all for SPRT) discriminations are temporal. Figure 9 is significant in that it provides empirical evidence for our interpretations of Figures 6, 7, and 8.

4. Conclusion

The goal of this final project is to paint the state of the art on fairness in scheduling policies. We began by exploring the transition of fairness as an abstract and ambiguous concept to the two analytical definitions of proportional and temporal fairness. We highlighted representative papers under each notion of fairness and discuss interesting findings. For example, in *Analysis of SRPT Scheduling: Investigating Unfairness*, N. Bansal and M. Harchol-Balter found contrary to popular belief, SRPT does not always starve large jobs. This result inspires researchers to develop new metrics of fairness that build on the fact that giving priority to shortest remaining processing time jobs can still be fair.

Papers rigorously highlight how proportional and temporal measures each reflect fairness in queueing systems. However, it also becomes obvious that any policy that strictly obeys temporal fairness will perform poorly under proportional fairness, and vice versa. For example, FCFS performs the fairest under temporal fairness, but it disregards the fact that there may be many smaller jobs in the queue that, under proportional fairness, should not have waited for that long. As it becomes clear that one singular definition fails to paint the whole picture, combined fairness definitions began to emerge.

In the plethora of combined fairness measures, we found that there are generally two types: 1) sophisticated, mathematical measures (e.g. RAQFM and SQF), and 2) direct, intuitive measure (e.g. DF). We examine papers under each type. For the first type of combined fairness measure, we studied RAQFM and SQF. We found that this type of fairness measure provides very high levels of detail that make it possible to conduct mathematical analyses of the system. For the second type, we read about DF. We found the second type of fairness measure immensely helpful for understanding how fairness works under specific scenarios.

We investigated one particular gap in the study of DF. We found that the paper limited its analysis of the scheduling policy to scenarios. We were curious about the performance of DF under steady state systems, so we created a simulation to examine DF further. Our simulation involved recording the performance of DF under four different policies: Random, SRPT, FCFS, and SJF. We observed that the policies behaved as predicted -- confirming DF's potential to be used as a system fairness measure. Furthermore, we extended the definition of DF to reflect that proportional and temporal discriminations are not equally unfair.

In discrimination frequency, we identified a potential flaw: all discriminations are viewed as equally unfair. For example, a proportional discrimination where a job of size 2 precedes a job of size 1 is weighed equally as a proportional discrimination where a job of size 100 precedes a job of size 1. This loss of information (severity of discriminations) contributes to the simplicity of DF as a fairness measure. However, we believe that the DF definition allows flexibility for extending it to incorporate discrimination severity without losing its simple nature.

Interesting and important questions also remain outside the scope of discrimination frequency. We observe fewer literature focused on fairness for non sized-based and non-preemptive scheduling policies despite the fact that such policies are the most commonly implemented. The question remains -- has the area been fully researched or is the area of not significant interest? Along the evolution of fairness in scheduling policies, we observed that fairness definitions become more sophisticated and would include many new factors. We wonder, what factors are next? Lastly, while studying combined measures, we observed a tradeoff between accuracy and simplicity of fairness measures. What are the best standards for balancing accuracy and simplicity? Is there a fundamental limit to which one can achieve both?

5. Bibliography

Abi-itzhak, B. et al. Quantifying fairness in queueing systems: principles, approaches, and applicability. *Probab Eng Inf Sci 22 (4).* 2008.

Avi-Itzhak, B. et al. SQF: a slowdown queueing fairness measure, *Performance Evaluation 64*, 1121–1136.

B. Avi-Itzhak, et al. A resource allocation fairness measure: Properties and bounds, *Queueing Systems Theory and Applications 56 (2)* (2007) 65–71.

Bansal, N. and M. Harchol-Balter. Analysis of SRPT scheduling: Investigating unfairness. In *Proceedings of ACM Sigmetrics Conference on Measurement and Modeling of Computer Systems*, 2001.

Gordon, E.S., New problems in queues: social injustice and server production management, *Ph.D. Thesis*, Massachusetts Institute of Technology, 1987.

Harchol-Balter, M. et al. Size-based scheduling to improve web performance, *ACM Transactions on Computer Systems* 21 (2) (2003)

Larson, R. Perspectives on queues: social justice and the psychology of queueing, *Operations Research* 35 (1987) 895–905.

Rafaeli, A. The effects of queue structure on attitudes, *Journal of Service Research* 5 (2) (2002) 125–139.

Rai, I.A. Analysis of LAS scheduling for job size distributions with high variance, in: *Proc. of ACM Sigmetrics*, 2003.

Sabin, G. "Job fairness in non-preemptive job scheduling," *International Conference on Parallel Processing, 2004.*

Sandmann, W. A discrimination frequency based queueing fairness measure with regard to job seniority and service requirement, in: *Proc. of Euro NGI Conf. on Next Generation Int. Nets*, 2005.

Wierman, A. Fairness and scheduling in single server queues. *Surv Oper Res Manage Sci 16 (1)*. 2011

Wierman, A. Classifying Scheduling policies with Respect to Unfairness in an M/Gl/1. *Sigmetrics*. 2003.