Writing to Read

How to write readable journal papers

Anujan Varma
Acknowledgments

This presentation includes material developed by Dr. Harold Stone at IBM Research, used with his permission. The writing samples are taken from various publications and are the property of the copyright holders. This material may not be reproduced or distributed outside the CPME/CMPS 200 class.
What this talk is about

- How to organize a journal paper
- Applies also to conference papers and other technical publications

What this talk is not about

- How to do good research
- How to typeset your paper
Acceptance Criteria for Journal Papers

- Significance of problem
- Originality
- Technical Contribution
- Presentation

This talk only deals with the presentation.
Outline

- Article Structure
- The Abstract
- The Introduction
- Summary and Conclusions
- Body of a Section
Basics

- Spell correct and use good grammar
- Keep *Strunk and White* handy

Some common flaws
- Using a preposition to end a sentence with.
- “Slower speed with less processors”
- “Between A, B and C”
- Principle Engineers are successful for a reason
The Journal Article

Abstract
1. Introduction
2. Some Results
3. Some More Results
...
N-1. The Last Results
N. Summary and Conclusions
Article Structure
The Butler Did It

Agatha Christie wrote novels, not articles.

Her goal: Suspense

Your goal: Victim, murderer, and plot are clear from the start.
Example

Abstract: The butler did it.

1. **Introduction.** He did it for the inheritance.

2. **Background.** There was a large estate.

3. **Main Result.** The butler did it.

4. **Other Results.** He was caught.

5. **Summary and Conclusions.** Crime does not pay.
Why the difference?

Suspense leads to pleasure,

But not in journal papers.

Readers stop at the first quagmire (or at the first puddle)
Puddle-free writing

Help the reader to create mental boxes to catch ideas.

Jargon-free explanations.

Self-contained: Include definitions and summarize key past results.
The Abstract
The Abstract: Content

- What is the problem?
- What are the results?
- How are they useful? (optional)

Be Specific!
Latency-Rate Servers: A General Model for Analysis of Traffic Scheduling Algorithms

Dimitrios Stiliadis and Anujan Varma, Member, IEEE

Abstract—In this paper, we develop a general model, called Latency-Rate servers (LR servers), for the analysis of traffic scheduling algorithms in broadband packet networks. The behavior of an LR server is determined by two parameters—the latency and the allocated rate. Several well-known scheduling algorithms, such as Weighted Fair Queueing, VirtualClock, Self-Clocked Fair Queueing, Weighted Round Robin, and Deficit Round Robin, belong to the class of LR servers. We derive tight upper bounds on the end-to-end delay, internal burstiness, and buffer requirements of individual sessions in an arbitrary network of LR servers in terms of the latencies of the individual schedulers in the network, when the session traffic is shaped by a token bucket. The theory of LR servers enables computation of tight upper bounds on end-to-end delay and buffer requirements in a heterogeneous network, where individual servers may support different scheduling architectures and under different traffic models.

scheduling algorithms will be a valuable tool in the design and analysis of such networks. In this paper, we develop such a model for studying the worst-case behavior of individual sessions in a network of schedulers where the schedulers in the network may employ a broad range of scheduling algorithms. This approach enables us to calculate tight bounds on the end-to-end delay of individual sessions and the buffer sizes needed to support them in an arbitrary network of schedulers.

Our basic approach consists of defining a general class of schedulers, called Latency-Rate servers, or simply LR servers. The theory of LR servers provides a means to describe the worst-case behavior of a broad range of scheduling algorithms in a simple and elegant manner. This theory is based on the concept of a busy period of a session, a period of time
The Abstract: Style

- For all technical people
- Brief
- Highlight items of importance
The Introduction
The Introduction

- Problem statement and context
- Background, prior art
- Specific results reported
- Structure of the paper
Highlighting Techniques

Nasdf lkj werr afda KDFK aseiv wwqio wwlkdo *kkowyy* isenc.

Xzyljin opwej adie calcgh cmnnki Mnwdeo qqqoid dalwpc cpcpne:
1. Idqwerp pplje dpne.
2. Pqlhdp pjelcp plwaad.
Cdalk dafqe dlqen clcvp.
Specific Results

- All contributions stated
- Amplify the abstract
Structure of paper

- Contents of each section
The rest of this paper is organized as follows. In Section II we define a general methodology for estimating and updating the system potential that will form the basis of the algorithms developed in later sections. In Section III we present FFQ in terms of a hypothetical fluid model, and subsequently extend to a packet server. We also analyze the fairness properties of the algorithm. In Section IV we develop and analyze the SPFQ algorithm. Some concluding remarks are presented in Section V.

II. Maintaining the System Potential
Summary and Conclusions
Summary and Conclusions

1. Summarize
   The facts
2. Conclude
   The consequences
3. Open problems
Body of a Section
Body of a Section

1. What happens here
2. How this fits
3. The results
4. Transition
Typical Openings

In this section, ...

This section continues the derivation by ...

Thus far, the discussion has ... . Here, ...

...
The Contribution

Give facts and evidence.

My way is better because:
1. I prove it (Great!)
2. I have much evidence (Good.)
3. It is simpler (Is it?)
4. It looks simpler to me
5. I did it.
The Contribution

Easy evidence:

- Mathematical analysis.
- Proof
- Implementation
- Performance data
The Contribution

Harder:

- A design
- A language
- A representation

Utility: What more does it do?

“Different” may not be enough.
Typical Closings

This completes the derivation of ...

Having described the serial case, we move to ...

Using the framework described here, ...
\[ W_{i,j}^S(\tau, t) \geq \max(0, \rho_i \left(t - \tau - \sum_{j=1}^{k} \Theta_{i}^{(S_j)}\right)) \]

where \( \Theta_{i}^{(S_j)} \) is the latency of the \( j \)th server in the network for session \( i \).

Thus, we have established that the worst-case behavior of a network of \( \mathcal{LR} \) servers can be analyzed simply by studying the behavior of a single equivalent \( \mathcal{LR} \) server. Note that the results so far did not make any assumptions on input traffic. Thus, the \( \mathcal{LR} \) model can be used to derive upper bounds on delay and buffer requirements for any traffic model. We illustrate this in the next section by deriving upper bounds on delays and buffer requirements for two well-known traffic models.

III. DELAY AND BACKLOG ANALYSIS
OF NETWORKS OF \( \mathcal{LR} \) SERVERS

In this section, we will demonstrate the utility of the model of \( \mathcal{LR} \) servers by deriving bounds on end-to-end delays and buffer requirements for input traffic defined by two common shaping schemes—token bucket and dual leaky bucket.

A. Token-Bucket-Shaped Traffic

Since we have already shown that an arbitrary network of
The Body: some tips

- Simple case, then general case
- Make compartments, then fill in
- Twice told, different ways
  - Picture with text
  - Text with equation
Twice Told

“Life is as tedious as a twice-told tale
Vexing the dull ear of a drowsy man.”

- Shakespeare wasn’t referring to journal papers
Twice Told

\[ f(x) = \sum_i w(i) \times B(i) \]

that is, \( f(x) \) is a weighted sum of \( Bs \).
Twice Told

Let $X$ be the child of $Y$, and $Y$ the child of $Z$, as shown in Fig. 2.
Twice Told

**Definition:** A pig is a …

**Definition:** A peg is a …

Note that a pig differs from a peg by …

Fig 2.2 shows a pig and a peg …
The Running Example

1. First concept
2. Example of first concept
3. Next concept
4. Example embellished
5. Next concept
6. More embellishment
The Running Example

(Not as good)

1. First concept
2. Next concept
3. Next concept
4. Example of first concept
5. Example embellished
6. More embellishment
Hindrances

- Jargons and acronyms
- Notation
- Ambiguity
JAA: Jargons And Acronyms

Yes, you can use jargons and acronyms

- Max of two new definitions per section, four max per paper
- Ideally one acronym per paper
- Redundancy required on first usage
- Reuse periodically to refresh reader
Hints on notation

\[ (i) F_{m,n}^{j_k} \]
Hints on notation

\( F_{m,n}^{(i)} j_k \)

- What parameters change?
- If only \( m \) changes in a section, then make \( m \) the only
Hints on notation

- Avoid subscripts on supers or subs, if possible.
- $F^j$ versus $F^{j_k}$
Ambiguity

- Which meaning?
- Redundancy helps
- The reader can’t ask.
Fighting Ambiguity

... performance of two bus systems ...

... crude image analysis ...

... common sense amplifier ...
Hyphenation

Common sense-amplifier
A sense amplifier that is common.

Common-sense amplifiers
An amplifier of common sense.
Which versus That

From an IBM summer internship offer letter:

“You will be paid for any IBM holidays which fall during the summer.”

If “that” changes the meaning, take the job!
Practice: Read

1. Read an article.
2. If you stop, why?
3. Was structure made known?
4. What questions were answered?
5. What was twice told?
6. What would improve it?
Practice: Edit

1. Edit your past work.
2. Edit new work from your colleagues.
3. Insist on what you learned here.
Practice: Write

1. Write the Body.
2. Then the Introduction.
3. Then Summary and Conclusions
4. Then the Abstract