Optimal probing rates in packet networks

Ben M Parker

Queen Mary, University of London

June 26, 2008

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.
Increasing the number of probes will

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.

Increasing the number of probes will

- give us more data

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.
Increasing the number of probes will

- give us more data
- therefore increase the precision (lower the variance) of the estimate

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.
Increasing the number of probes will

- give us more data
- therefore increase the precision (lower the variance) of the estimate

BUT

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.

Increasing the number of probes will

- give us more data
- therefore increase the precision (lower the variance) of the estimate

BUT

- there are more probe packets to interfere with themselves and data packets

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.
Increasing the number of probes will

- give us more data
- therefore increase the precision (lower the variance) of the estimate

BUT

- there are more probe packets to interfere with themselves and data packets
- thus potentially lowering the accuracy (increasing the bias) of the estimate.

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.

Increasing the number of probes will

- give us more data
- therefore increase the precision (lower the variance) of the estimate

BUT

- there are more probe packets to interfere with themselves and data packets
- thus potentially lowering the accuracy (increasing the bias) of the estimate.

In practice, uniform probing at a rate of one per second, or one per minute, is often adopted.

Aims

We consider the problem of determining an appropriate rate at which to probe in packet data networks.
Increasing the number of probes will

- give us more data
- therefore increase the precision (lower the variance) of the estimate

BUT

- there are more probe packets to interfere with themselves and data packets
- thus potentially lowering the accuracy (increasing the bias) of the estimate.

In practice, uniform probing at a rate of one per second, or one per minute, is often adopted. But which rate is best?

Idea

- In this work the we regard all network measurements as numerical experiments.

Idea

- In this work the we regard all network measurements as numerical experiments.
- Effectiveness of probing is measured by a utility function that we place on bias and variance in our estimator for λ, and delay to the real data.

Idea

- In this work the we regard all network measurements as numerical experiments.
- Effectiveness of probing is measured by a utility function that we place on bias and variance in our estimator for λ, and delay to the real data.
- We define an optimality criterion,

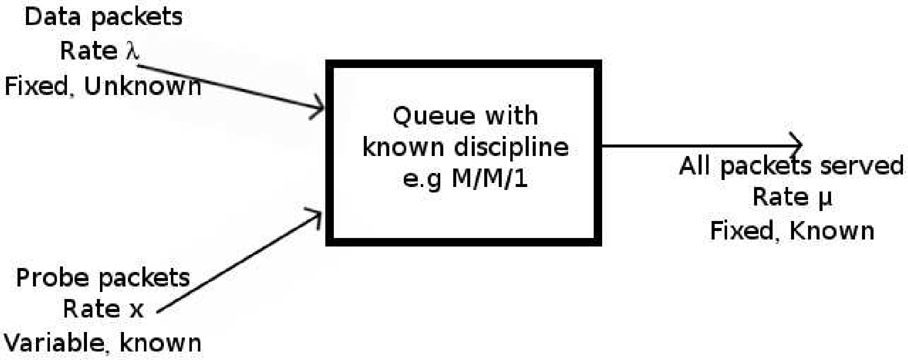
Idea

- In this work the we regard all network measurements as numerical experiments.
- Effectiveness of probing is measured by a utility function that we place on bias and variance in our estimator for λ, and delay to the real data.
- We define an optimality criterion,

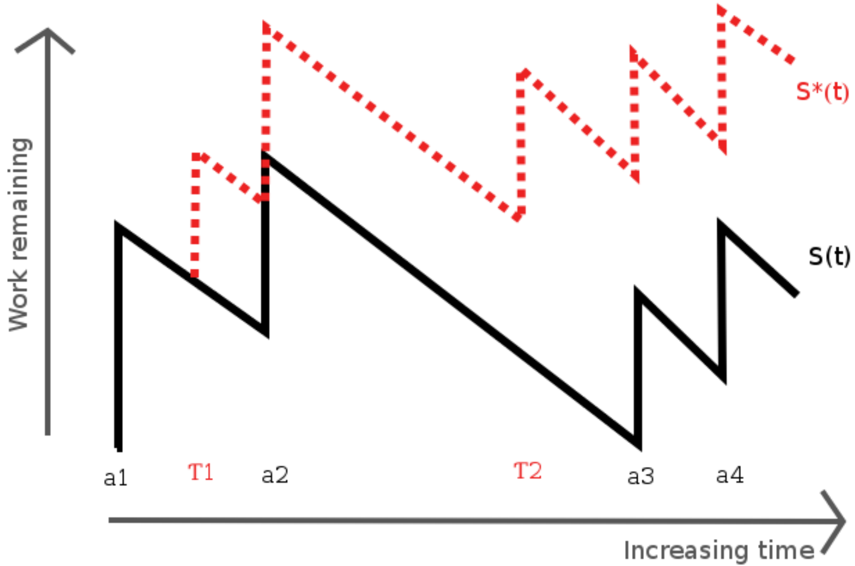
Idea

- In this work the we regard all network measurements as numerical experiments.
- Effectiveness of probing is measured by a utility function that we place on bias and variance in our estimator for λ, and delay to the real data.
- We define an optimality criterion, and apply the statistical principles of Design of Experiments to network measurement experiments to develop a methodology to find an optimal probing rate.

Black box view of the network



Virtual waiting time and effect of probing



Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.

Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.
- Arrival times are $a_0, a_1, a_2, \ldots$.

Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.
- Arrival times are a_0, a_1, a_2, ...
- Probe packets introduced as point process with rate x at times $\tau_1, \tau_2, \ldots \tau_N$.

Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.
- Arrival times are a_0, a_1, a_2, $\ldots$.
- Probe packets introduced as point process with rate x at times $\tau_1, \tau_2, \ldots \tau_N$.
- The new process we form by introducing probe packets is $S^*(t)$.

Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.
- Arrival times are a_0, a_1, a_2, $\ldots$.
- Probe packets introduced as point process with rate x at times $\tau_1, \tau_2, \ldots \tau_N$.
- The new process we form by introducing probe packets is $S^*(t)$.

Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.
- Arrival times are a_0, a_1, a_2, $\ldots$.
- Probe packets introduced as point process with rate x at times $\tau_1, \tau_2, \ldots \tau_N$.
- The new process we form by introducing probe packets is $S^*(t)$.

Outputs

Given $N = n$ packets:

- We find $\tau_1 < \tau_2 < \ldots < \tau_n$.

Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.
- Arrival times are a_0, a_1, a_2, ...
- Probe packets introduced as point process with rate x at times $\tau_1, \tau_2, \ldots \tau_N$.
- The new process we form by introducing probe packets is $S^*(t)$.

Outputs

Given $N = n$ packets:

- We find $\tau_1 < \tau_2 < \ldots < \tau_n$.
- We observe $S^*(\tau_1), S^*(\tau_1), \ldots, S^*(\tau_n)$.

Definitions

Inputs

We define:

- $S(t)$ as the Virtual Waiting time, the amount of time required for all packets in system at time t to complete service and exit the system.
- Arrival times are a_0, a_1, a_2, $\ldots$.
- Probe packets introduced as point process with rate x at times $\tau_1, \tau_2, \ldots \tau_N$.
- The new process we form by introducing probe packets is $S^*(t)$.

Outputs

Given $N = n$ packets:

- We find $\tau_1 < \tau_2 < \ldots < \tau_n$.
- We observe $S^*(\tau_1), S^*(\tau_1), \ldots, S^*(\tau_n)$.
- We let $y_i = S^*(\tau_i)$, and our data are thus $y_1, \ldots, y_n$.

Specifying a utility function

We wish to minimise the bias and variance of some $\hat{\lambda}$, an estimator formed from some function of $\mathbf{Y}$, x, and μ.

Specifying a utility function

We wish to minimise the bias and variance of some $\hat{\lambda}$, an estimator formed from some function of $\mathbf{Y}$, x, and μ.
We also wish to minimise the disruption of the data packets on the network due to increased probe traffic.

We wish to minimise the bias and variance of some $\hat{\lambda}$, an estimator formed from some function of $\mathbf{Y}$, x, and μ.
We also wish to minimise the disruption of the data packets on the network due to increased probe traffic.

Utility function

We need to specify a utility function

$$\psi(\mathrm{Bias}(\hat{\lambda}|x), \mathrm{Var}(\hat{(\lambda)}|x), D(x)),$$

Specifying a utility function

We wish to minimise the bias and variance of some $\hat{\lambda}$, an estimator formed from some function of $\mathbf{Y}$, x, and μ.
We also wish to minimise the disruption of the data packets on the network due to increased probe traffic.

Utility function

We need to specify a utility function

$$\psi(\mathrm{Bias}(\hat{\lambda}|x), \mathrm{Var}(\hat{(\lambda)}|x), D(x)),$$

and particularly to find $x_\lambda = \arg\min_x \psi(x)$, our optimal probing rate.

Trade off

- We wish to trade accuracy and precision when estimating λ compared with the disruption caused to date packets in measuring it.

Trade off

- We wish to trade accuracy and precision when estimating λ compared with the disruption caused to date packets in measuring it.
- When comparing bias and variance, natural metric is mean square error

$$\mathrm{MSE}(\hat{\lambda}) = [\mathrm{Bias}(\hat{\lambda})]^2 + \mathrm{Var}(\hat{\lambda}).$$

Trade off

- We wish to trade accuracy and precision when estimating λ compared with the disruption caused to date packets in measuring it.
- When comparing bias and variance, natural metric is mean square error

$$\mathrm{MSE}(\hat{\lambda}) = [\mathrm{Bias}(\hat{\lambda})]^2 + \mathrm{Var}(\hat{\lambda}).$$

Trade off

- We wish to trade accuracy and precision when estimating λ compared with the disruption caused to date packets in measuring it.
- When comparing bias and variance, natural metric is mean square error

$$\mathrm{MSE}(\hat{\lambda}) = [\mathrm{Bias}(\hat{\lambda})]^2 + \mathrm{Var}(\hat{\lambda}).$$

- How much to penalise disruption is more subjective- depends on network under study.

Trade off

- We wish to trade accuracy and precision when estimating λ compared with the disruption caused to date packets in measuring it.
- When comparing bias and variance, natural metric is mean square error

$$\text{MSE}(\hat{\lambda}) = [\text{Bias}(\hat{\lambda})]^2 + \text{Var}(\hat{\lambda}).$$

- How much to penalise disruption is more subjective- depends on network under study.

Trade off

- We wish to trade accuracy and precision when estimating λ compared with the disruption caused to date packets in measuring it.
- When comparing bias and variance, natural metric is mean square error

$$\mathrm{MSE}(\hat{\lambda}) = [\mathrm{Bias}(\hat{\lambda})]^2 + \mathrm{Var}(\hat{\lambda}).$$

- How much to penalise disruption is more subjective- depends on network under study.

General utility function

We propose a general form of the utility function

$$\psi(x) = k\,\mathrm{MSE}(\hat{\lambda}) + (1-k)D(x),$$

where $0 < k < 1$ and $D(x)$ is a function which measures disruption.

Measuring disruption

Disruption function

We measure this disruption per packet at probe rate x as

$$D(x) = \frac{1}{N} \sum_{i=1}^{N} [S^*(a_i) - S(a_i)]$$

Measuring disruption

Disruption function

We measure this disruption per packet at probe rate x as

$$D(x) = \frac{1}{N}(\sum_{i=1}^{N}[S^*(a_i) - S(a_i)]^r)^{1/r}$$

Measuring disruption

Disruption function

We measure this disruption per packet at probe rate x as

$$D(x) = \frac{1}{N}(\sum_{i=1}^{N}[S^*(a_i) - S(a_i)]^r)^{1/r}$$

In general the underlying $S(a_i)$ will be impossible to observe, and we can only estimate this by simulation.

Measuring disruption

Disruption function

We measure this disruption per packet at probe rate x as

$$D(x) = \frac{1}{N}(\sum_{i=1}^{N}[S^*(a_i) - S(a_i)]^r)^{1/r}$$

In general the underlying $S(a_i)$ will be impossible to observe, and we can only estimate this by simulation.

The index r allows us to penalise deviations from the mean delay time. r allows us to penalise jitter.

Measuring disruption

Disruption function

We measure this disruption per packet at probe rate x as

$$D(x) = \frac{1}{N}(\sum_{i=1}^{N}[S^*(a_i) - S(a_i)]^r)^{1/r}$$

In general the underlying $S(a_i)$ will be impossible to observe, and we can only estimate this by simulation.

The index r allows us to penalise deviations from the mean delay time. r allows us to penalise jitter.

- $r = 1$ corresponds to being ambivalent about jitter,
- increasing r penalises high jitter more severely for equal average delays.

Example- an M/M/1 queue

- M/M/1 queue used to demonstrate the method.

Example- an M/M/1 queue

- M/M/1 queue used to demonstrate the method.
- Simulations using $\mu = 5s^{-1}$ throughout.

Example- an M/M/1 queue

- M/M/1 queue used to demonstrate the method.
- Simulations using $\mu = 5s^{-1}$ throughout.
- λ fixed at $2.5s^{-1}$ to allow us to assess bias and variance of our estimator.

Example- an M/M/1 queue

- M/M/1 queue used to demonstrate the method.
- Simulations using $\mu = 5s^{-1}$ throughout.
- λ fixed at $2.5s^{-1}$ to allow us to assess bias and variance of our estimator.
- Estimator picked was

$$\hat{\lambda} = \frac{\frac{1}{N}\sum_{i=1}^{N} Y_i - \frac{1}{\mu}}{\frac{1}{N}\sum_{i=1}^{N} Y_i \frac{1}{\mu}}.$$

Example - continued

- Candidate points x in the range 0.1 to 2.4 seconds, at intervals of 0.1.

Example - continued

- Candidate points x in the range 0.1 to 2.4 seconds, at intervals of 0.1.
- Probing experiment allowed only for 10 seconds. We disregarded any probe packets that had not completed service after 10 seconds.

Example - continued

- Candidate points x in the range 0.1 to 2.4 seconds, at intervals of 0.1.
- Probing experiment allowed only for 10 seconds. We disregarded any probe packets that had not completed service after 10 seconds.
- 1000 simulations (macro-replications) performed for each candidate point. By looking at the 1000 $\hat{\lambda}$ for each candidate point, bias and variance of $\hat{\lambda}$ could be estimated.

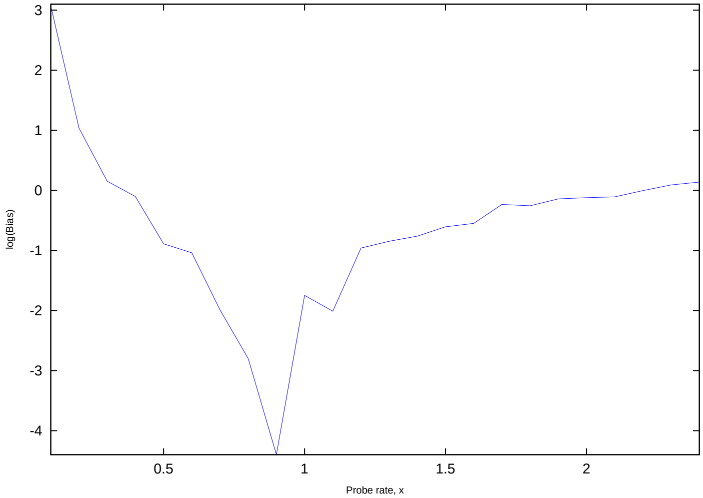
Example - continued

- Candidate points x in the range 0.1 to 2.4 seconds, at intervals of 0.1.
- Probing experiment allowed only for 10 seconds. We disregarded any probe packets that had not completed service after 10 seconds.
- 1000 simulations (macro-replications) performed for each candidate point. By looking at the 1000 $\hat{\lambda}$ for each candidate point, bias and variance of $\hat{\lambda}$ could be estimated.
- Knowing the underlying virtual system time process $S(t)$, and the altered process after probing $S^*(t)$, we were able to estimate the value of our disruption function $D(x)$.

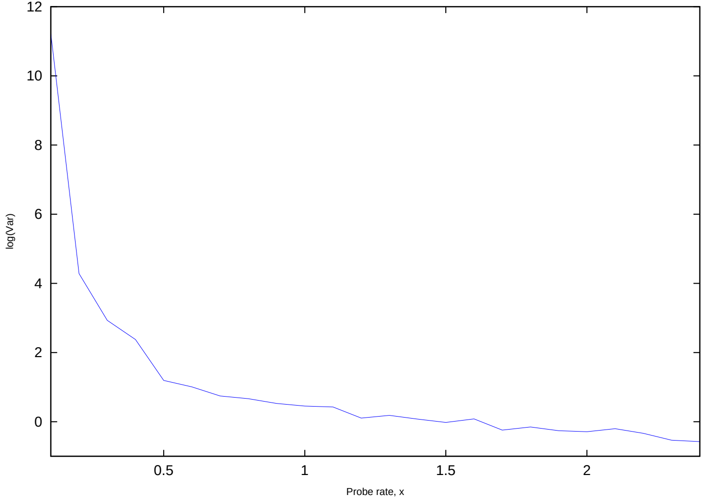
Example - continued

- Candidate points x in the range 0.1 to 2.4 seconds, at intervals of 0.1.
- Probing experiment allowed only for 10 seconds. We disregarded any probe packets that had not completed service after 10 seconds.
- 1000 simulations (macro-replications) performed for each candidate point. By looking at the 1000 $\hat{\lambda}$ for each candidate point, bias and variance of $\hat{\lambda}$ could be estimated.
- Knowing the underlying virtual system time process $S(t)$, and the altered process after probing $S^*(t)$, we were able to estimate the value of our disruption function $D(x)$.
- To illustrate a possible utility function, we set $k = \frac{1}{2}$.

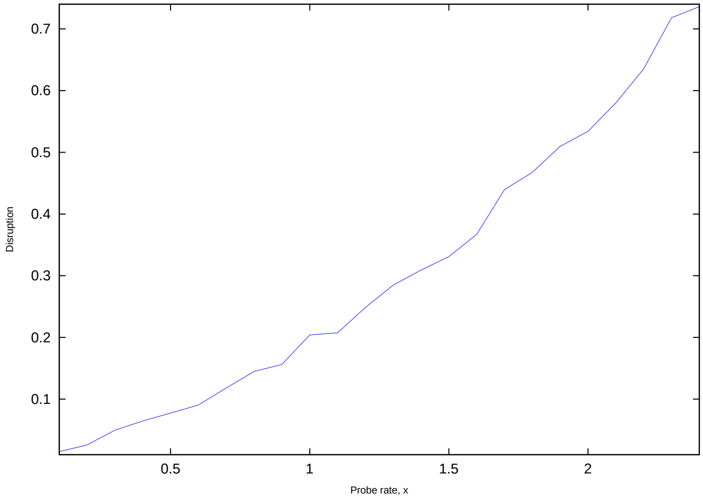
log(Bias)



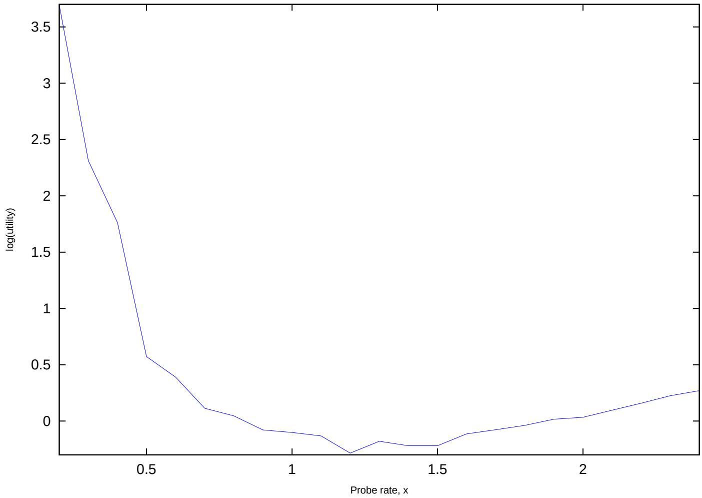
log(Variance)



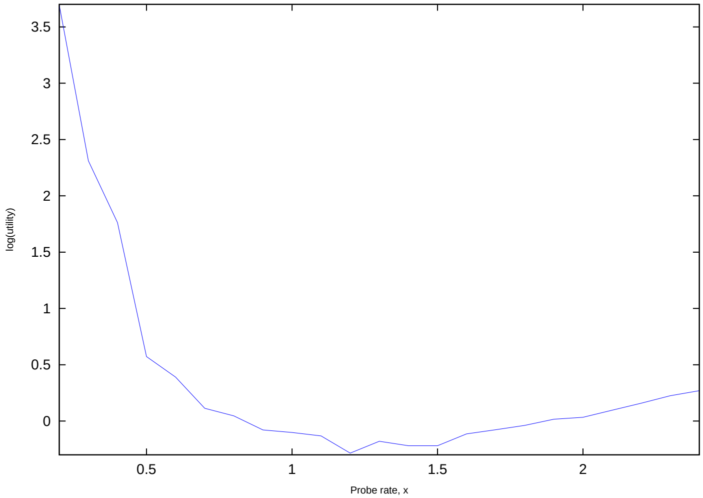
Disruption



log(utility)



log(utility)



Optimal probing rate is minimum on the utility graph, when $x \approx 1.2s^{-1}$.

Discussion

- A low rate ($x < 0.5s^{-1}$) gives significantly worse results (higher utility) than probing at a higher rate $x > 0.5s^{-1}$.

Discussion

- A low rate ($x < 0.5s^{-1}$) gives significantly worse results (higher utility) than probing at a higher rate $x > 0.5s^{-1}$.
- If we had to choose one or the other, it is substantially better to probe slightly too fast than too slow.

Discussion

- A low rate ($x < 0.5s^{-1}$) gives significantly worse results (higher utility) than probing at a higher rate $x > 0.5s^{-1}$.
- If we had to choose one or the other, it is substantially better to probe slightly too fast than too slow.
- This poor behaviour is primarily due to high bias in $\hat{\lambda}$ for low x.

Discussion

- A low rate ($x < 0.5s^{-1}$) gives significantly worse results (higher utility) than probing at a higher rate $x > 0.5s^{-1}$.
- If we had to choose one or the other, it is substantially better to probe slightly too fast than too slow.
- This poor behaviour is primarily due to high bias in $\hat{\lambda}$ for low x.
- N, the number of experimental poisson probes released during 10s of experimental time for low rate x is typically very small: e.g. for $x = 0.1$ we would only expect one probe packet.

Discussion

- A low rate ($x < 0.5s^{-1}$) gives significantly worse results (higher utility) than probing at a higher rate $x > 0.5s^{-1}$.
- If we had to choose one or the other, it is substantially better to probe slightly too fast than too slow.
- This poor behaviour is primarily due to high bias in $\hat{\lambda}$ for low x.
- N, the number of experimental poisson probes released during 10s of experimental time for low rate x is typically very small: e.g. for $x = 0.1$ we would only expect one probe packet.
- As we have such a small number of probes at low x, this leads to very poor estimates.

Discussion

- A low rate ($x < 0.5s^{-1}$) gives significantly worse results (higher utility) than probing at a higher rate $x > 0.5s^{-1}$.
- If we had to choose one or the other, it is substantially better to probe slightly too fast than too slow.
- This poor behaviour is primarily due to high bias in $\hat{\lambda}$ for low x.
- N, the number of experimental poisson probes released during 10s of experimental time for low rate x is typically very small: e.g. for $x = 0.1$ we would only expect one probe packet.
- As we have such a small number of probes at low x, this leads to very poor estimates.
- The estimator $\hat{\lambda}$ is biased for both small and large probe rates x.

Conclusions

- Presented a framework for determining optimal probing rate.

Conclusions

- Presented a framework for determining optimal probing rate.
- Here bandwidth λ is estimated, but can be used for other estimators.

Conclusions

- Presented a framework for determining optimal probing rate.
- Here bandwidth λ is estimated, but can be used for other estimators.
- Not queue or probing method specific, applicable in principle to any queue or network.

Conclusions

- Presented a framework for determining optimal probing rate.
- Here bandwidth λ is estimated, but can be used for other estimators.
- Not queue or probing method specific, applicable in principle to any queue or network.
- Provides example of benefits of using statistical principles of design of experiments in networking. (Not much previous literature.)

Conclusions

- Presented a framework for determining optimal probing rate.
- Here bandwidth λ is estimated, but can be used for other estimators.
- Not queue or probing method specific, applicable in principle to any queue or network.
- Provides example of benefits of using statistical principles of design of experiments in networking. (Not much previous literature.)
- Challenges intuition about optimal probing rates.

Conclusions

- Presented a framework for determining optimal probing rate.
- Here bandwidth λ is estimated, but can be used for other estimators.
- Not queue or probing method specific, applicable in principle to any queue or network.
- Provides example of benefits of using statistical principles of design of experiments in networking. (Not much previous literature.)
- Challenges intuition about optimal probing rates.
- Difficulties in using knowledge to find an appropriate utility function.

Conclusions

- Presented a framework for determining optimal probing rate.
- Here bandwidth λ is estimated, but can be used for other estimators.
- Not queue or probing method specific, applicable in principle to any queue or network.
- Provides example of benefits of using statistical principles of design of experiments in networking. (Not much previous literature.)
- Challenges intuition about optimal probing rates.
- Difficulties in using knowledge to find an appropriate utility function.

Conclusions

- Presented a framework for determining optimal probing rate.
- Here bandwidth λ is estimated, but can be used for other estimators.
- Not queue or probing method specific, applicable in principle to any queue or network.
- Provides example of benefits of using statistical principles of design of experiments in networking. (Not much previous literature.)
- Challenges intuition about optimal probing rates.
- Difficulties in using knowledge to find an appropriate utility function.

Thanks for listening!

Any questions?

Varying the utility function

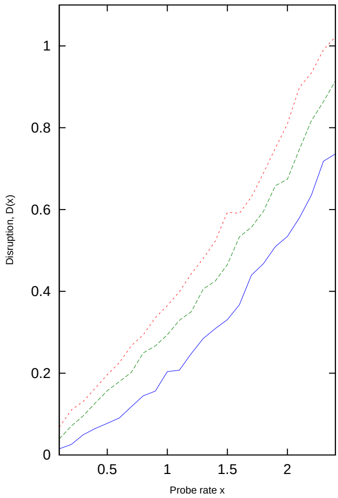
We seek to see how optimal probing rate depends on the utility function chosen.

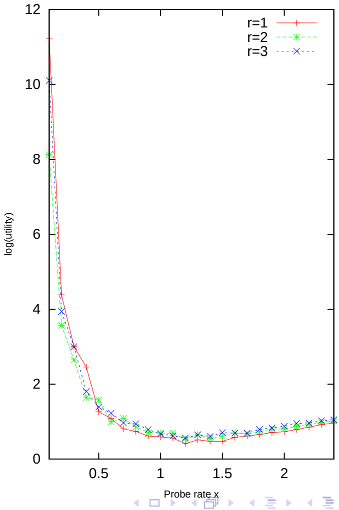
Varying the utility function

We seek to see how optimal probing rate depends on the utility function chosen. We first look at varying the parameter r.

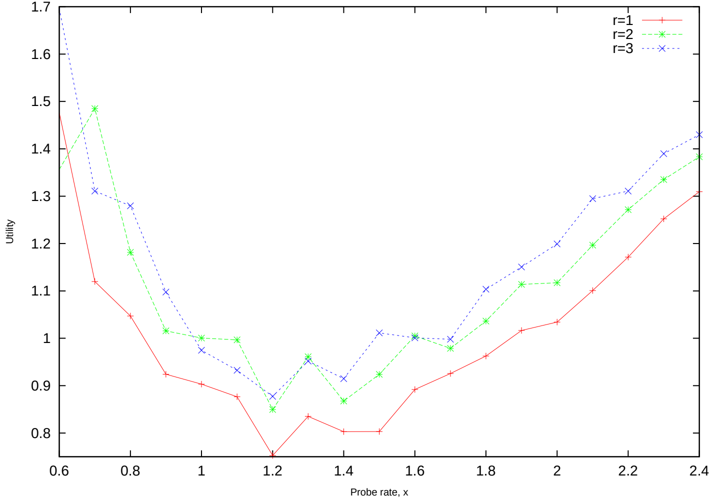
Varying the utility function

We seek to see how optimal probing rate depends on the utility function chosen. We first look at varying the parameter r.





Zoom


1.7
1.6
1.5
1.4
1.3
1.2
1.1
1
0.9
0.8
Utility
0.6
0.8
1
1.2
1.4
1.6
1.8
2
2.2
2.4
Probe rate, x
r=1
r=2
r=3

How does changing r affect optimal probing rate?

- r has the effect of penalising jitter, with higher r penalising large jitter more.

How does changing r affect optimal probing rate?

- r has the effect of penalising jitter, with higher r penalising large jitter more.
- As expected, higher probe rates are now penalised as jitter increases slightly as network load increases.

How does changing r affect optimal probing rate?

- r has the effect of penalising jitter, with higher r penalising large jitter more.
- As expected, higher probe rates are now penalised as jitter increases slightly as network load increases.
- We see that the value of r does not make a substantial difference the relative merits of different probing rates.

How does changing r affect optimal probing rate?

- r has the effect of penalising jitter, with higher r penalising large jitter more.
- As expected, higher probe rates are now penalised as jitter increases slightly as network load increases.
- We see that the value of r does not make a substantial difference the relative merits of different probing rates.
- Optimal probing rate for all $r = 1, 2, 3$ from these simulated data is $x = 1.2s^{-1}$.

Varying k: importance of delay

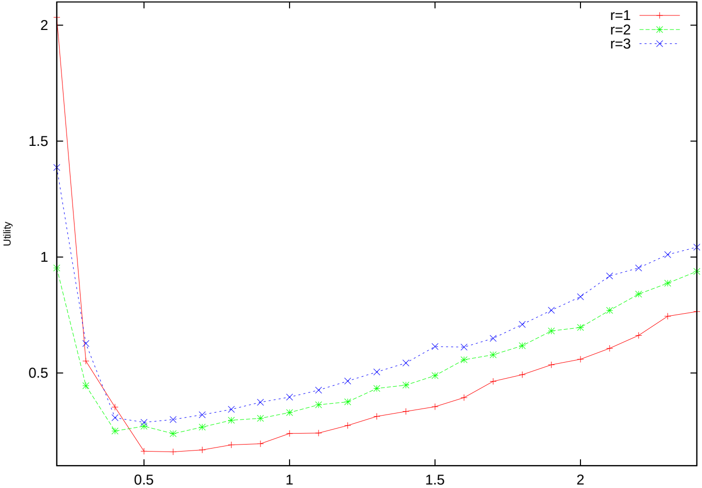
We now set $k = 0.05$.

Varying k: importance of delay

We now set $k = 0.05$. Lower k represents being more concerned with minimising network delays than with minimising the MSE of estimator.

Varying k: importance of delay

We now set $k = 0.05$. Lower k represents being more concerned with minimising network delays than with minimising the MSE of estimator.



Varying k, importance of delay

The optimal probing rate:

- can again be read as the minimal points of the three data sets, $x_\lambda \approx 0.6s^{-1}, r = 1, 2; x\lambda \approx 0.5s^{-1}, r = 3$.

Varying k, importance of delay

The optimal probing rate:

- can again be read as the minimal points of the three data sets, $x_\lambda \approx 0.6s^{-1}, r = 1, 2; x\lambda \approx 0.5s^{-1}, r = 3.$
- is lower now for case $k = 0.05$ than in previous cases for $k = 0.5$

Varying k, importance of delay

The optimal probing rate:

- can again be read as the minimal points of the three data sets, $x_\lambda \approx 0.6s^{-1}, r = 1, 2; x\lambda \approx 0.5s^{-1}, r = 3$.
- is lower now for case $k = 0.05$ than in previous cases for $k = 0.5$
- now varies slightly with r.