Closed-Loop Optimization of Scheduling Parameters

Magnus Proebster, Christian M. Mueller, Hajo Bakker
{magnus.proebster; christian.mueller}@ikr.uni-stuttgart.de
hajo.bakker@alcatel-lucent.de
06.10.2010

Universität Stuttgart
Institute of Communication Networks
and Computer Engineering (IKR)
Prof. Dr.-Ing. Andreas Kirstädter

Overview

- Motivation
- Introduction of the applied Scheduler
- Design & Evaluation of a Fairness Controller
- Conclusions

Motivation

Assumption

- Operators want to achieve a high user satisfaction
- This can be achieved by ensuring a certain fairness in a cell

  → Also users with bad channel conditions get an adequate service rate

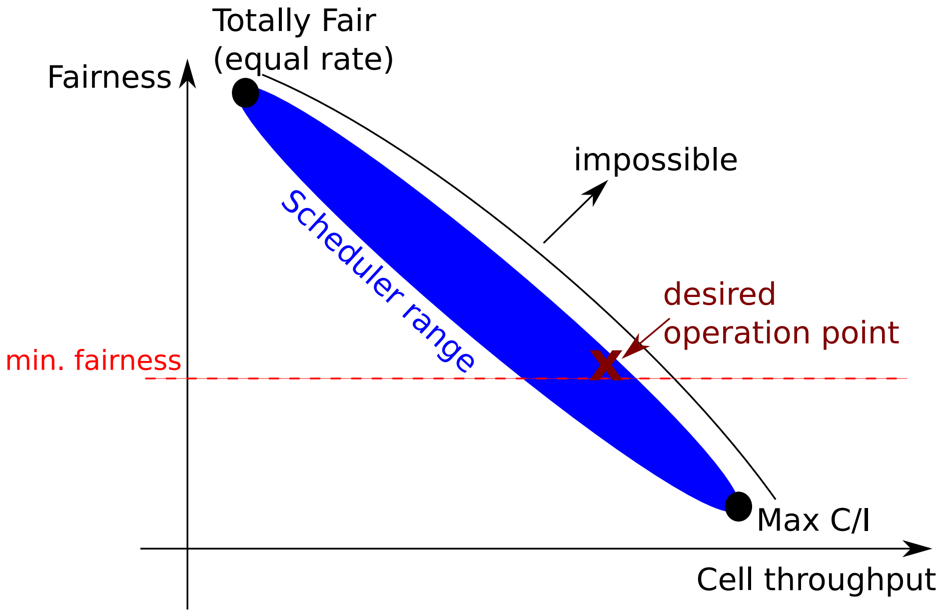
How can fairness be measured?

→ With the NGMN fairness requirement

Why is it difficult to adjust fairness?

- Trade-off: Fairness ↔Total cell throughput
  - Too much fairness: Waste of cell capacity
  - Unfairness: Starvation of cell edge users
- Parameters to achieve a fairness-level are cell-specific and vary dynamically

  (e.g. with cell load and user distribution)



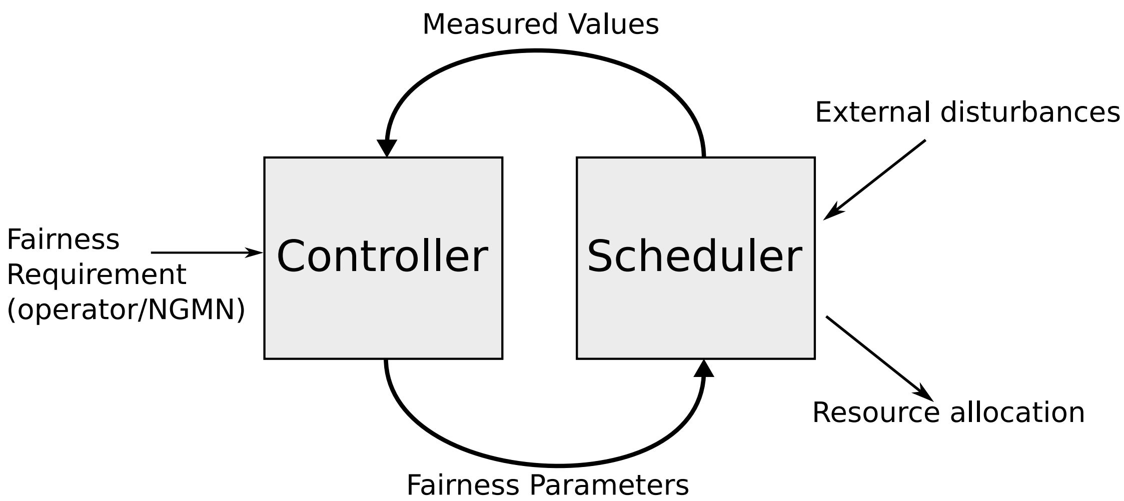
Approach

Goal

Improve the system throughput with an optimal and dynamic fairness adjustment

How do we get there?

- Determination of relevant scheduler parameters for fairness
- Development of an autonomous controller adjusting the level of fairness



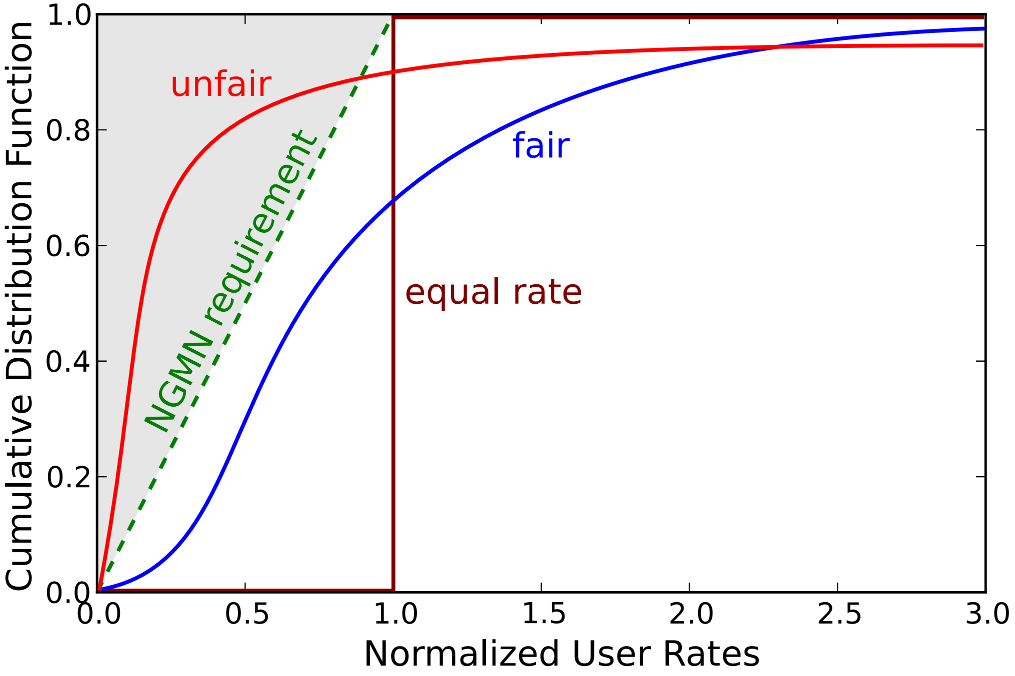
Advantages

- Self-optimizing system that does not require additional expenses (no human interactions needed)
- Operator gets the possibility to easily adjust the fairness level
- Operator does not need to know the underlying scheduling algorithm
- Maximization of the system capacity

NGMN Fairness Requirement

"100-x% of the users should have at least x% of the normalized throughput"

- Corresponds to a straight line in the CDF-plot of normalized user rates
- Shapes of the normalized user throughput CDF
  - Equal rate: step function at 1
  - Fair distribution is completely on the right-hand side of the requirement
  - Differences in system throughput are hidden by normalization



→ Very useful in wireless networks
  - Providing all users with an equal rate is very inefficient
  - Not too restrictive, offers flexibility
  - Distribution-based metric robust against channel fluctuations

→ NGMN requirement adopted by 3GPP, IEEE and others

Overview

- Motivation
- Introduction of the applied Scheduler
- Design & Evaluation of a Fairness Controller
- Conclusions

Scheduling Algorithm

Alpha-Fair Scheduler with Minimum Rate Constraint

Proportional Fair Mechanism

Token Counter Mechanism

$$\text{user} = \arg\max_{j} \left(\frac{R_j}{\overline{R_j}^{\alpha}} \cdot e^{\gamma_j T_j} \right)$$

$$\overline{R_j}(t+1) = (1-\beta)\overline{R_j}(t) + \beta\mu_j(t)$$

$$T_j(t+1) = T_j(t) + MBR - \mu_j(t)$$

PF Fairness Parameter

Minimum Bit Rate

Parameters

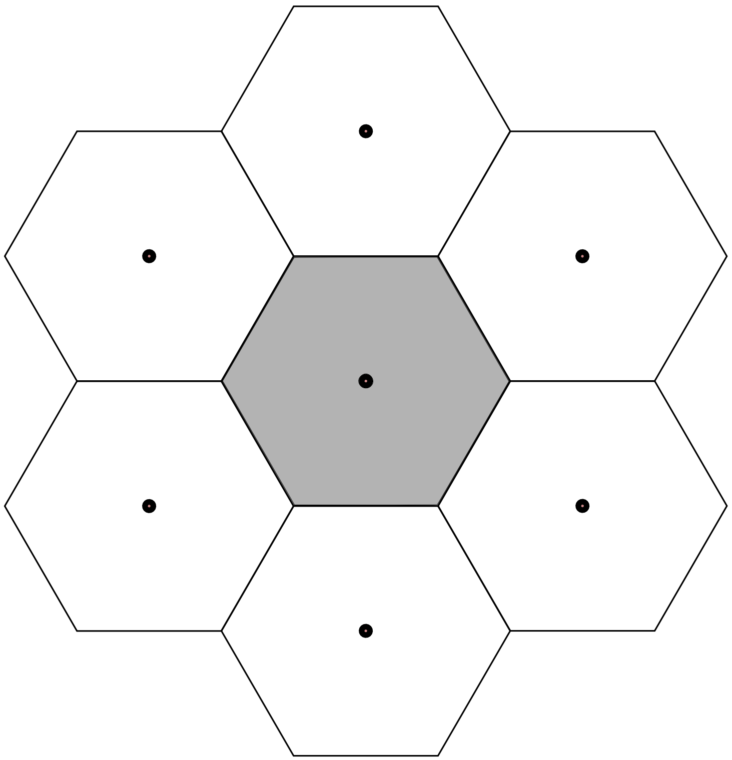
- α controls the fairness achieved by proportional fair
- MBR (Minimum Bit Rate) is the rate ensured by the token counter

→Other schedulers allowing to regulate fairness are also possible!

[1] M. Andrews et al: *"Optimal Utility Based Multi-User Throughput Allocation subject to Throughput Constraints", 2005*

Simulation Scenario

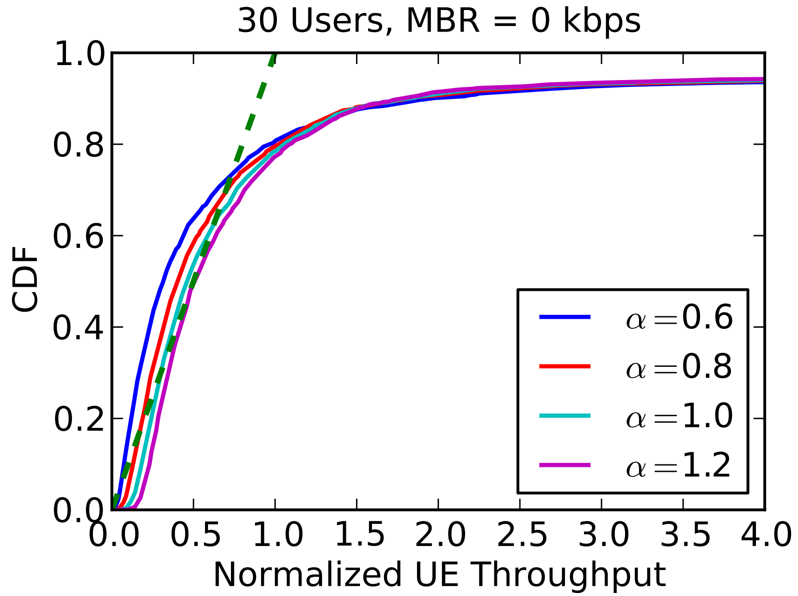
- Downlink only
- Channel model & interference
  - Pathloss, Shadowing & Fast fading
  - Constant interference from neighbour cells
  - SISO transmission
- Seven-site scenario
  - Center cell considered
  - Isotropic antennas at the base stations
- Ideal CQI reporting
- Shannon Capacity; clipped at -5dB and 26dB
- Users distributed equally in cell
  - Fixed user locations
  - Handover at the beginning
- Full buffer simulations

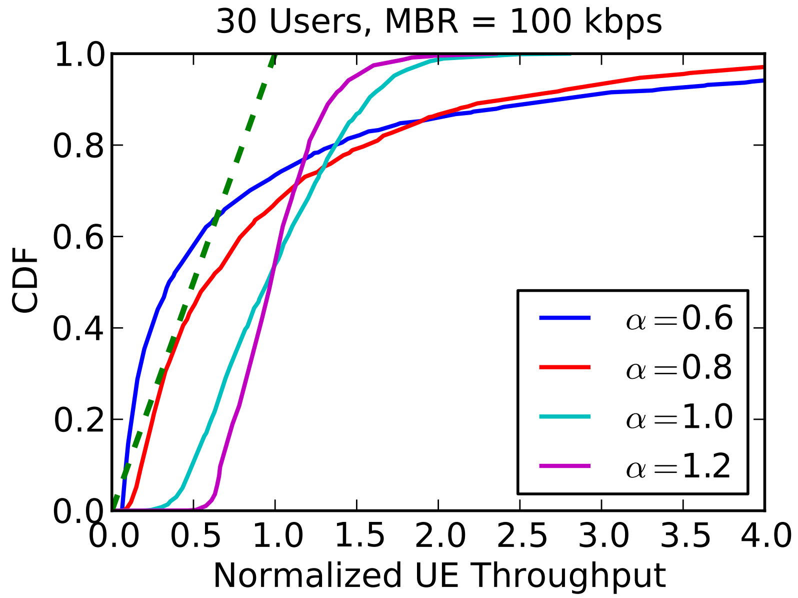


Parameter Influence on Fairness

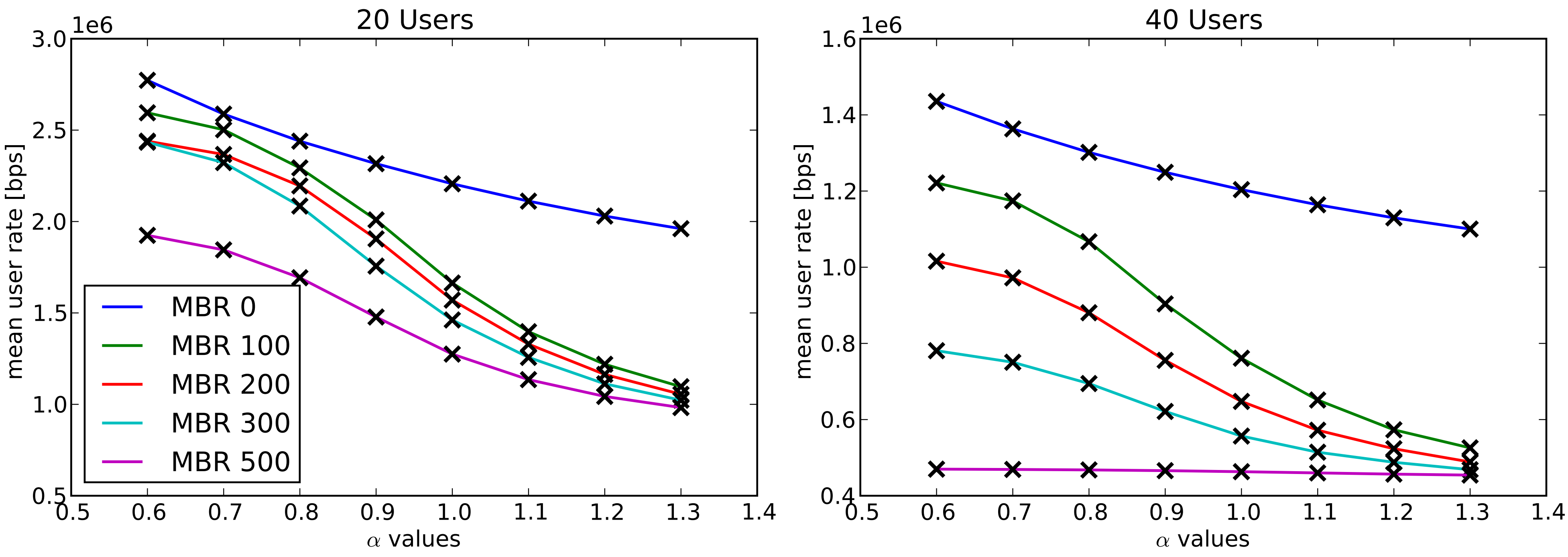
Results

- α and MBR influence the fairness significantly
- Without MBR, a high α value is needed to achieve fairness
- With MBR=100 kbps, α can be reduced





Fairness Parameterization and Cell Throughput



Results

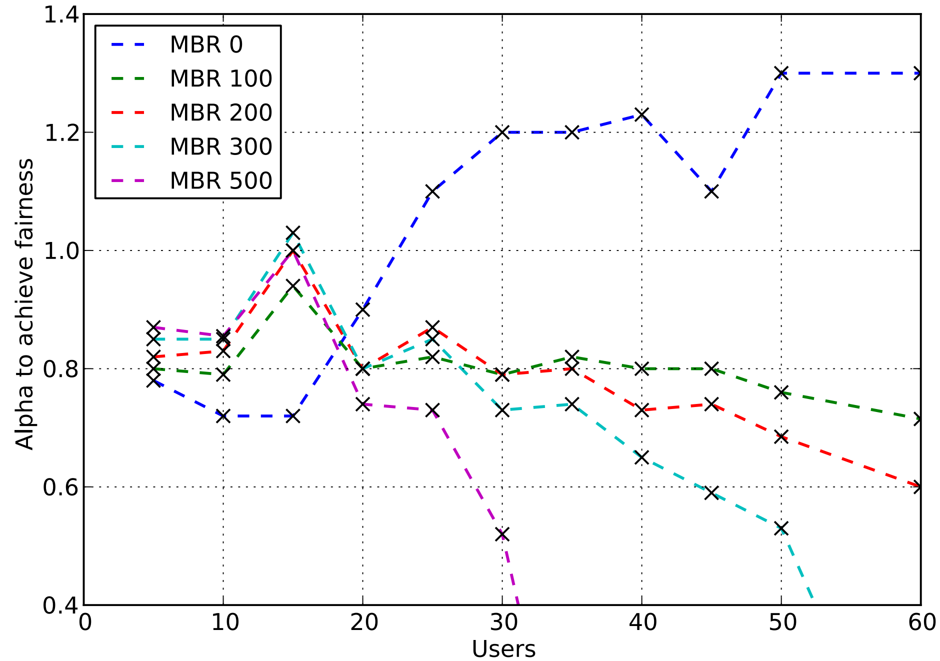
- Increased fairness → reduced cell throughput
- MBR has a higher impact than α

→ Optimization of the fair operating point with α for a given MBR to achieve a higher cell throughput

Throughput Gains from Dynamic α-Variation

α-Values to Achieve Fairness

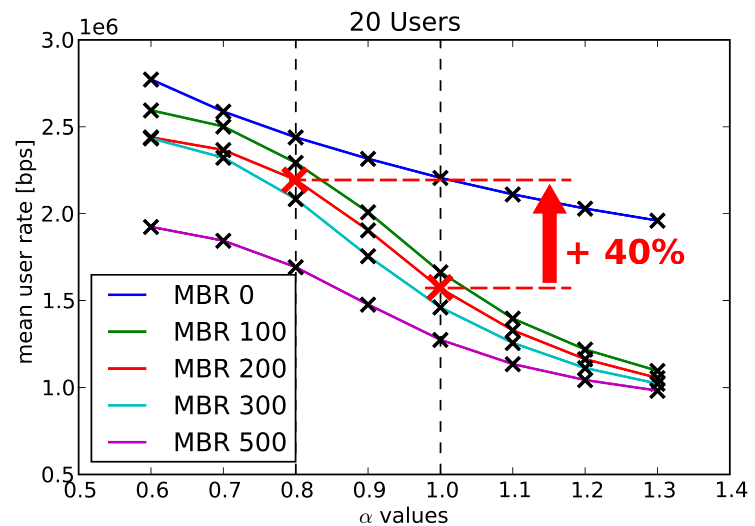
- Depend on MBR-setting (assumed to be fixed)
- Change with the load in the cell
  - Increasing trend without MBR
  - Decreasing trend with MBR



Throughput Gains

- Dynamic α-adaptation vs. static configuration

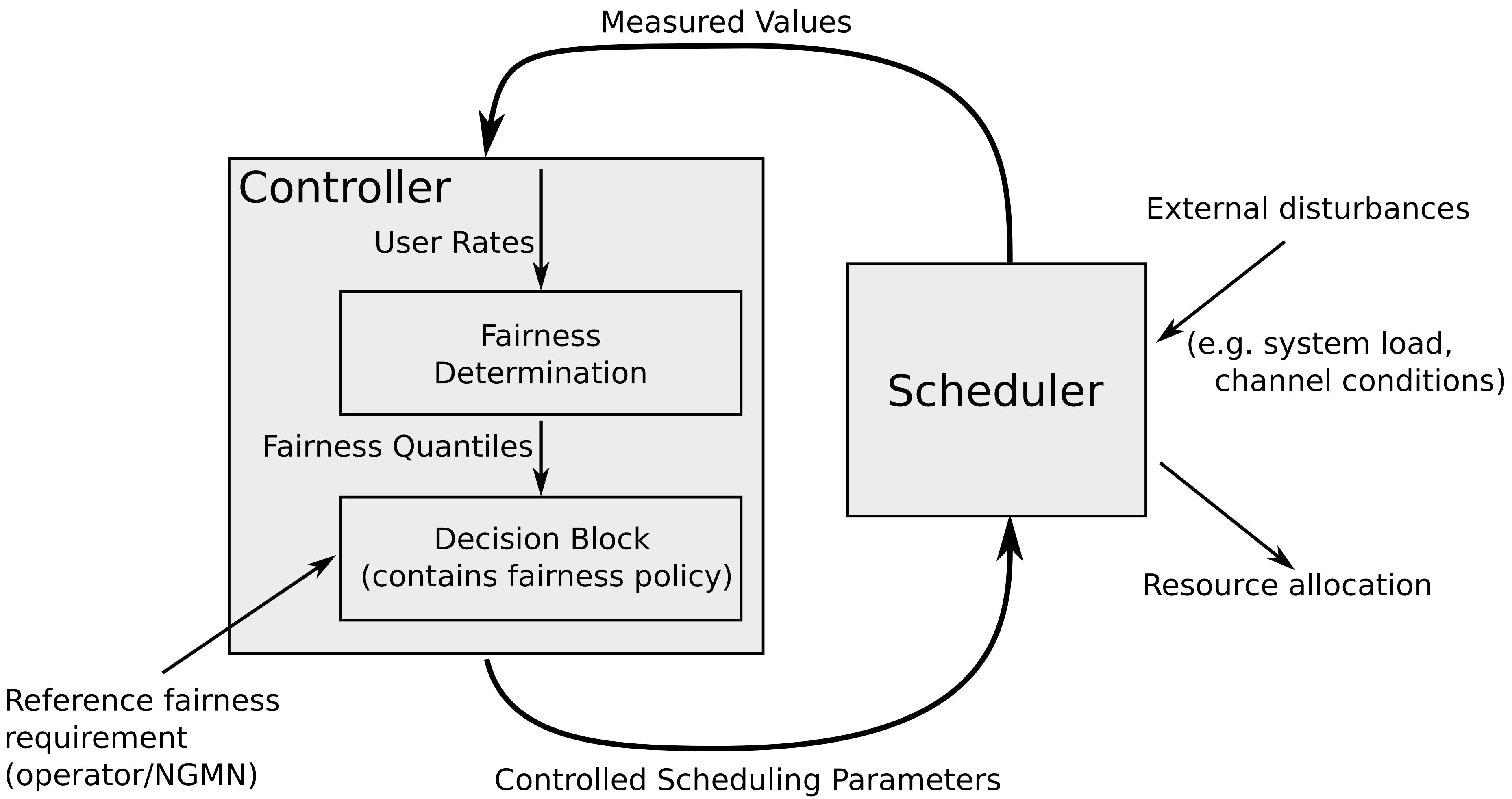
→ For 200 kbps MBR and 20 users:
mean throughput increased by ~40%



Overview

- Motivation
- Introduction of the applied Scheduler
- Design & Evaluation of a Fairness Controller
- Conclusions

Controller Building Blocks



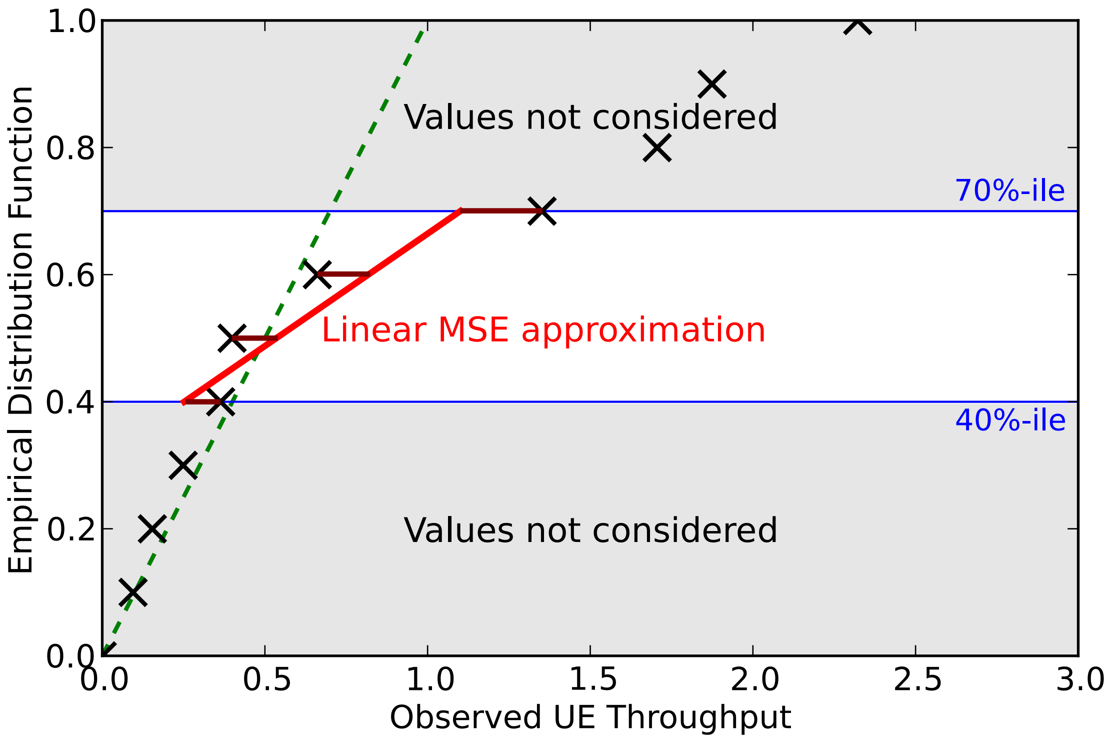
Properties

- Closed-loop feedback system
- Acts on much longer time-scales than the scheduler (in the order of seconds)

Controller Fairness Determination

How does it work?

- User rates are collected during a sampling interval
- From these rates, fairness quantiles can be obtained by sorting and normalizing
- The quantiles have to be matched with the fairness requirement



→ After fairness determination, a control action follows

- Not enough fairness → Increase α
- Too much fairness → Reduce α

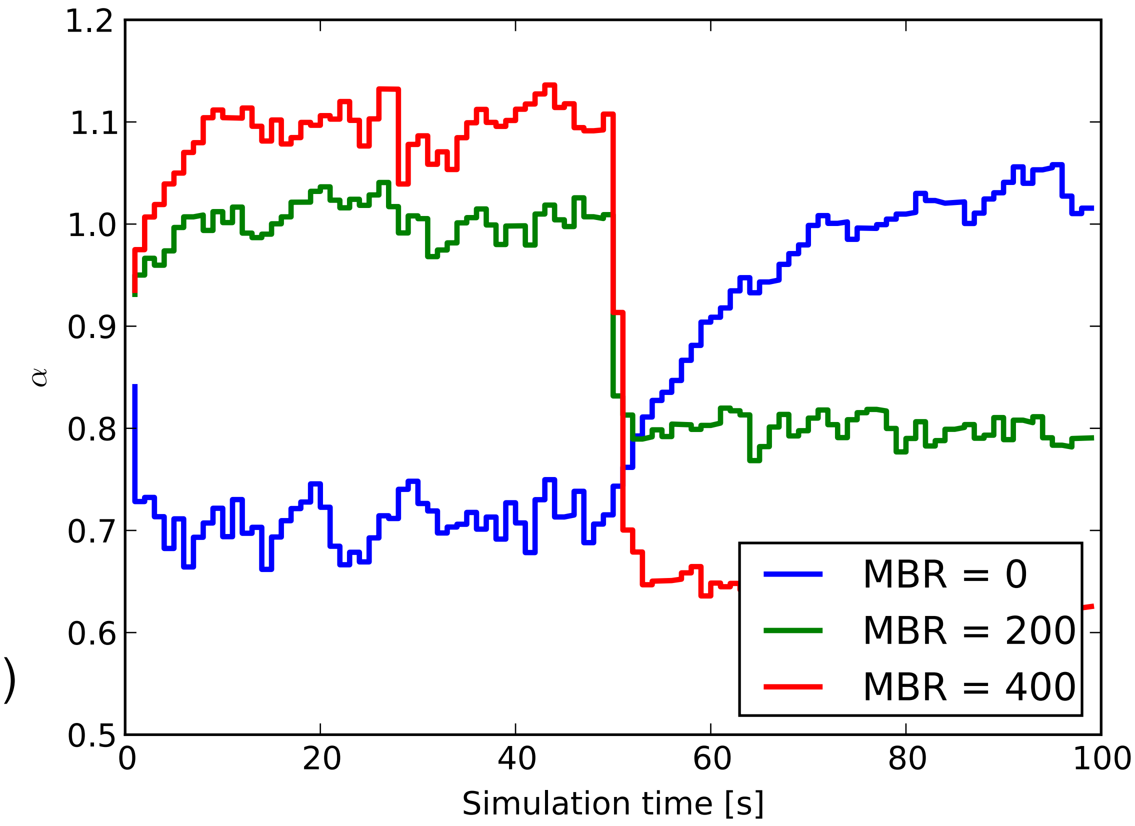
Controller Evaluation - Transient Behaviour

Simulation Parameters

- Number of Active users
  - 15 for t = [0 s; 50 s)
  - 30 for t = [50 s; 100 s]
- Sampling Interval: 1 s

Observation

- Quick adaptation to changed load situation
- Throughput gain: ~10%-17% (compared with static configuration)
- Static configuration is not optimal for half of the time

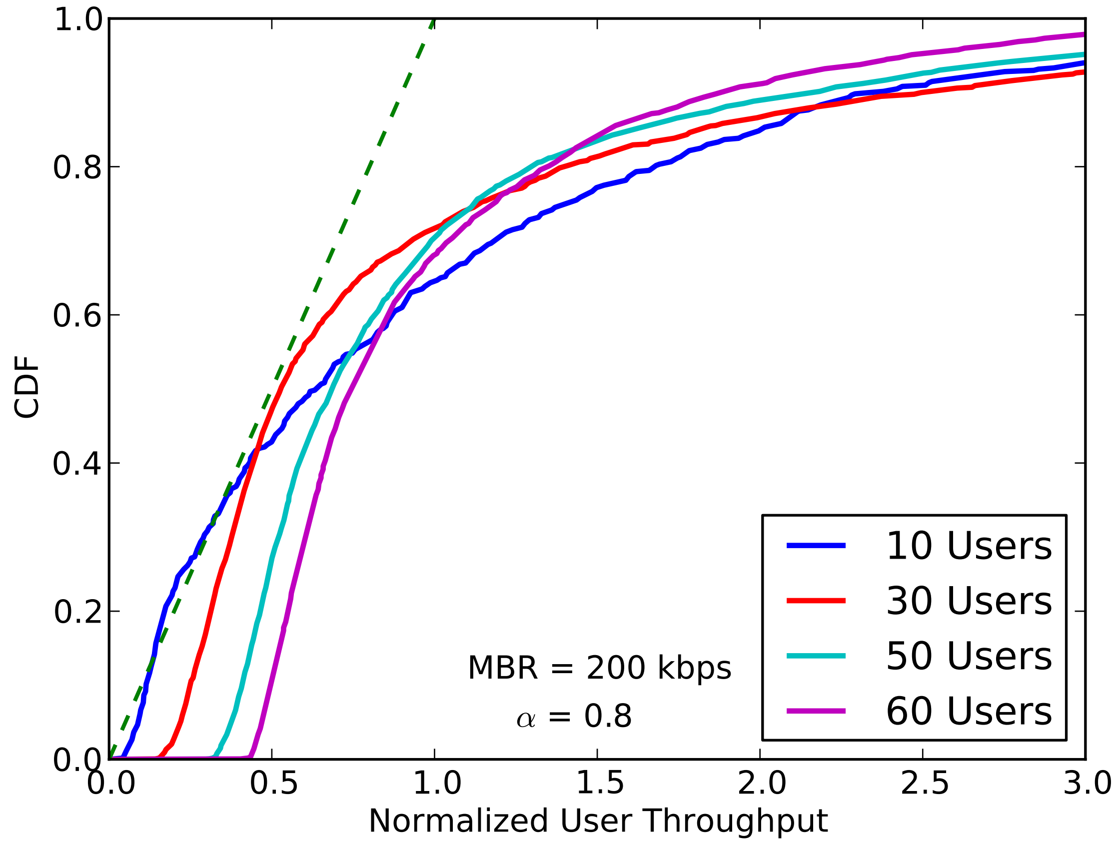


Fairness Adaptation w/o Controller

Without controller

- With increasing number of users, the system gets fairer

→ Waste of cell capacity



Fairness Adaptation with Controller

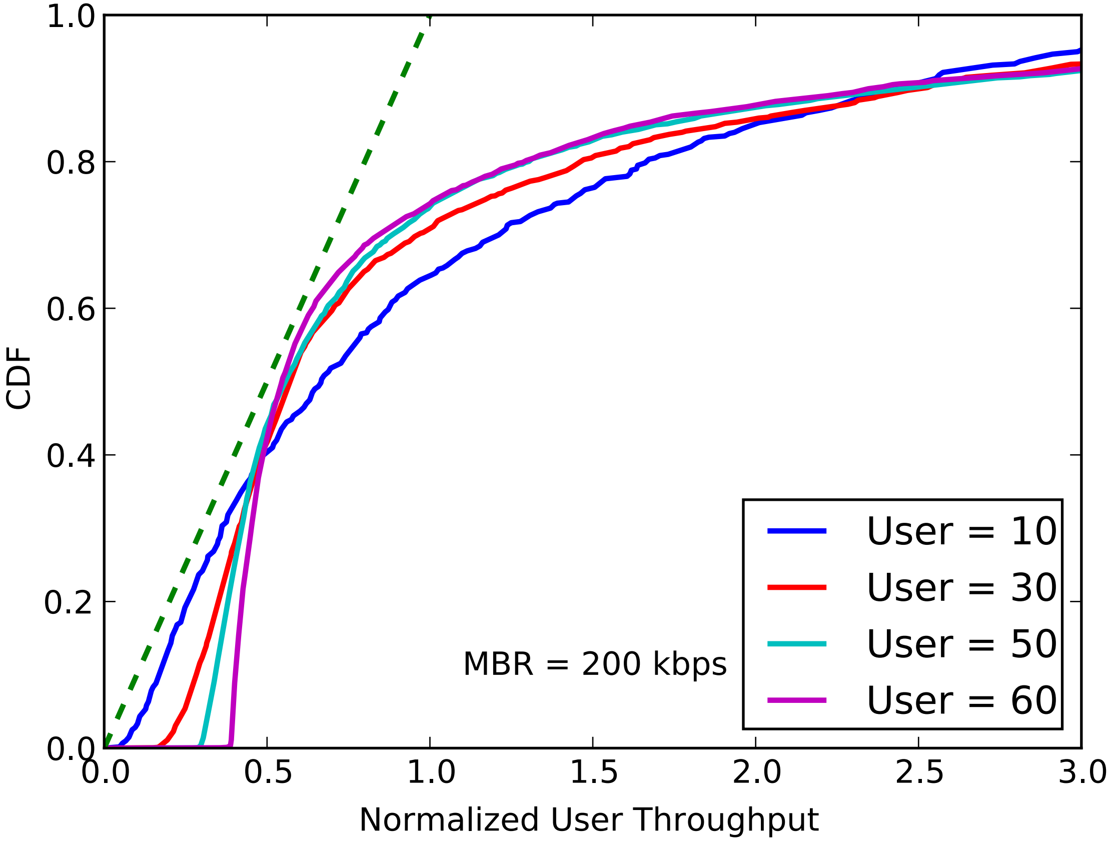
Without controller

- With increasing number of users, the system gets fairer

→ Waste of cell capacity

With controller

- All CDFs lie close to the fairness criterion
- Predicted α-configurations confirmed



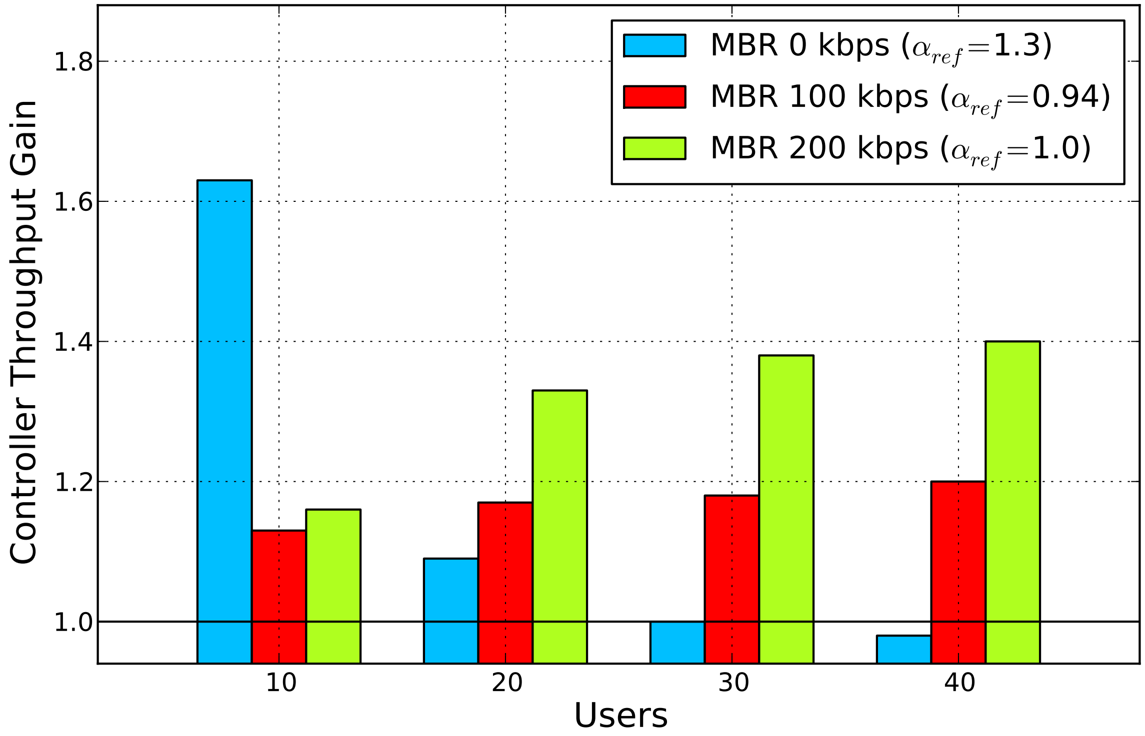
Throughput Gain with Controller

Reference

- MBR is fixed and constant in both cases
- Static reference for α such that system is always fair (conservative assumption)

Observations

- Good accordance to predicted gains
- Slight degradation due to fluctuations
- Acceptable performance for the reference point



Overview

- Motivation
- Introduction of the applied Scheduler
- Design & Evaluation of a Fairness Controller
- Conclusions

Conclusions

- High throughput gains are possible by adjusting fairness adaptively
  - Inherent trade-off between cell throughput & fairness
  - Optimization of scheduler flexibility increases diversity gain
- Design of a self-optimizing controller instance
  - Optimal throughput in the cell achievable while still maintaining fairness
  - No human interaction needed to tune scheduler parameters
  - Automatic adaptation to site-specific constraints (independent of the set of boundary conditions)

  → Increased users satisfaction and reduced costs per bit
- The demonstrated architecture can be applied to any scheduler allowing to parametrize the fairness level