Optimum Resource Allocation in HSDPA

Stephan Sigg and Klaus David

University of Kassel, Chair of Communication Technology (ComTec)
Wilhelmshoeher Allee 73, D-34109 Kassel, Gernany
Email: comtec@uni-kassel.de

Abstract - **We propose a channel aware resource allocation algorithm for HSDPA that receives higher overall weighted throughput than the popular proportional fair scheduler, while following the same fairness scheme. Our algorithm combines features of the proportional fair algorithm with a recent optimisation algorithm for the randomised knapsack problem. We demonstrate the high potential of the proposed algorithm in a simulation environment.**

1. Introduction

With the introduction of HSDPA (High Speed Downlink Packet Access) in release 5 of the UMTS standard, downlink data rates beyond 10 Mbps became possible [1]. To optimally explore the available resources, new scheduling algorithms have been introduced that prefer connections with excellent channel conditions [2,3,4,5]. Scheduling schemes that strictly prefer mobiles with better channel conditions optimise the overall throughput of the system [6,7]. It can be argued, that customers with bad channel conditions experience inferior throughput with these simple scheduling schemes. To circumvent this problem, several authors have introduced scheduling schemes that provide some kind of fairness between different users [8,9,10]. Unfortunately, since those schemes are often adopted from scheduling algorithms in wire line networks, the specific characteristics of wireless networks, such as fading or Doppler shift, are not addressed by these implementations. In [5] the proportional fair scheduler is introduced, which takes advantage of the instantaneous fluctuations of the radio channel. This scheduling algorithm has been further analysed in [3,4, 11]. The proportional fair scheduler provides proportional fairness [12] between all users since it guarantees that every single active customer occupies asymptotically the same fraction of time. However, it is not optimal in terms of overall throughput [6,13,14].

We propose a scheduling algorithm that keeps track of the data rate, channel condition and weighting according to the proportional fairness weighting scheme [12] and also considers the available code channels and maximum transmission power as constraints. The proposed scheduling scheme solves the underlying optimisation problem by adopting an algorithm that was introduced in [15].

Similar to our contribution, the authors in [13] model the environment faced by a resource allocation algorithm in UMTS networks as a linear optimisation problem with several constraints but do not include the maximum transmission power of the base station in their formulas. Recently, the authors of [16] presented a similar analysis that also abstracts from the maximum transmission power of the base station. To the best of our knowledge no other publications exist that describe how to optimally solve the resource allocation problem or that even mentions the difficulty of the underlying problem.

The paper is organised as follows. In section 2 we analyse various environments in HSDPA that may be observed by a scheduling algorithm. These various environments are defined by different network implementations. In section 3 we give a short overview on the PF-scheduler and introduce the KP-scheduler in section 4. Section 5 contains simulation results while section 6 draws our conclusion.

2. Analysis of the environment

We will explicitly address issues experienced with HSDPA and for simplicity assume only one carrier frequency. The obtained results may be generalised to several carriers. At the base station packet switched data is constantly arriving and buffered. The aim of the packet scheduler is to assign all these data to the code channels that are available in every timeslot. Due to code multiplexing and multi code operation a maximum of 15 code channels can be assigned to one or more mobiles in one timeslot. Because of the experienced interference this number may be smaller so that in general only $C_{\max}$ code channels with $0 \leq C_{\max} \leq 15$ are available in one Transmission Time Interval (TTI). Depending on the experienced SIR to one mobile the data rate varies with different modulation schemes. Since the transmission of packet switched data is based on a transmission time interval basis, the time axis is divided into discrete sections in each of which the algorithm basically faces the same optimisation problem (see Figure 1).

Each coloured Block in the figure illustrates the amount of data scheduled to one of the mobiles with respect to the transmission power. Let $i \in N$ and $r_{\max}$ and $P_{\max}$ represent the maximum data rate for one connection and the maximum transmission power available at the base station. For a connection to mobile i, the corresponding block therefore describes the

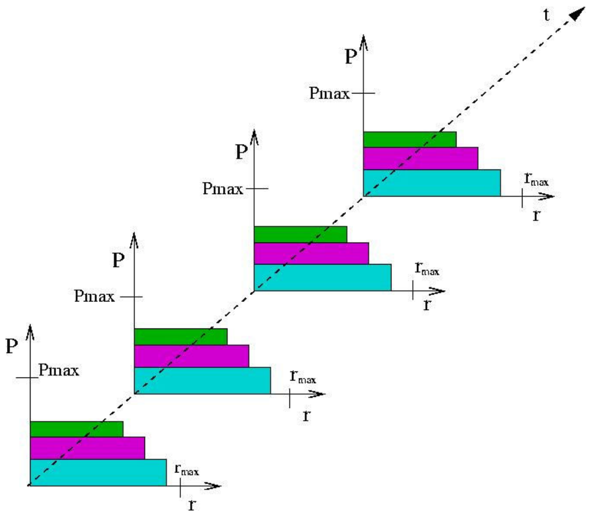

Figure 1 The scheduling algorithm experiences the same optimisation problem in discrete timeslots.

necessary transmission power $p_i \in R^+$ and the required data rate $r_i \in N$. For every block it is $r_i \le r_{\max}$ and $p_i \le P_{\max}$. Note that we assume the data rate to be integer valued. This is feasible since the number of possible data rates is finite (compare Table 1).

At the beginning of a transmission time interval, the resource allocation algorithm considers the link quality to each mobile. Based on this knowledge, the scheduling algorithm assigns appropriate data rates and timeslots to every single mobile. In general, it is possible that the buffer in the Node B is filled to such extend, that the available $C_{\max}$ code channels do not suffice to serve all data. To reduce the delay for all calls it is therefore essential that the scheduler at least manages to assign as much data as possible in every TTI. Fairness constraints that have additional influence on the importance or the urgency of a connection i are modelled by weights $w_i \in R$ that are multiplied to the data rate r_i.

We will in the following discussion consider only one discrete transmission time interval at a time. Since a maximum (weighted) throughput in every TTI also maximises the overall (weighted) throughput, the obtained results may be carried over to continuous environments with consecutive transmission intervals.

The environment that contains the optimisation goals and constraints discussed above is described by the following optimisation problem.

$$\begin{aligned} &\max imise \sum_{i=1}^{n} w_i a_i c_i r_i \\ &subject\ to \sum_{i=1}^{n} a_i c_i p_i \le P_{\max} \\ &\qquad \sum_{i=1}^{n} a_i c_i \le C_{\max} \\ &\qquad c_i \in \{1,\dots,C_{\max}\} \\ &\qquad a_i \in \{0,1\}. \end{aligned} \tag{1}$$

In this optimisation problem the binary variable a_i specifies, if a call i is actually scheduled or not. Furthermore, c_i is the number of code channels occupied by the connection to mobile i. Relying on the actual implementation of the network operator, several different optimisation problems evolve from the general problem (1). For example, the weights w_i that are multiplied to the data rate r_i may be defined to be integer valued, or the transmission power p_i may follow a special distribution function.

For a moment let us assume that every mobile is only allowed to use one single code channel at a time. To improve the comprehensibility we further abstract from the $C_{\max}$ constraint. Instead of considering the maximum number of code channels that may be occupied by one connection, we model a connection to a mobile i on $c_i \in \{1,\dots,C_{\max}\}$ code channels as c_i single connections to this mobile. This is feasible, since the use of multiple code channels in parallel linearly increases the data rate to one mobile (compare Table 1). If we define the weighted data rate with $r'_i = w_i r_i$, the resulting optimisation problem is

$$\begin{aligned} &\max imise \sum_{i=1}^{nC_{\max}} a_i r_i' \\ &subject\ to \sum_{i=1}^{nC_{\max}} a_i p_i \le P_{\max} \\ &\qquad a_i \in \{0,1\}. \end{aligned} \tag{2}$$

Observe that this is the knapsack problem which is known to be NP-complete [17]. With integer weights $w_i \in N$ the problem is easily solvable. In fact, it suffices to choose w_i such that $w_i r_i \in N$ for every r_i (compare section 4.1).

Modulation\Codes	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
QPSK – ¼	0.12	0.24	0.36	0.48	0.6	0.72	0.84	0.96	1.08	1.2	1.32	1.44	1.56	1.68	1.8
QPSK – ½	0.24	0.48	0.72	0.96	1.2	1.44	1.68	1.92	2.16	2.2	2.64	3.88	3.12	3.36	3.6
QPSK – ¾	0.36	0.72	1.08	1.44	1.8	2.16	2.52	2.88	3.24	3.6	2.96	4.32	4.68	5.04	5.4
16 QAM – ½	0.48	0.96	1.44	1.92	2.4	2.88	3.36	3.84	4.32	4.8	5.28	5.76	6.24	6.72	7.2
16 QAM – ¾	0.72	1.44	2.16	2.88	3.6	4.32	5.04	5.76	6.48	7.2	7.92	8.64	9.36	10.1	10.8

Table 1 Different data rates (Mbps) due to various modulation schemes and code channels [1].

3. The proportional fair scheduler

Channel-aware scheduling algorithms as the proportional fair scheduler are believed to be most promising in achieving maximum data rates in current CDMA-TDD systems [3,4,14]. The proportional fair scheduler also guarantees proportional fairness [12] between all users. It was first described in [5] and further analysed in [3,4]. The scheduler sorts all customers by means of $w_i r_i = r'_i$, where w_i^{-1} is the average throughput of connection i. Mobiles that experience an instantaneous rise in channel conditions will receive higher r'_i values. The PF-scheduler greedily serves those customers with maximum r'_i first until no further mobiles can be served in this timeslot because of power or code restrictions. The running time of this algorithm with n combating users is $O(n + n\log n)$ since the data rates are weighted and then sorted before scheduling the mobiles with the highest r'_i one after another. Due to the NP-completeness of the underlying KP-problem, we cannot hope to always solve it to optimality with the proportional fair scheduler. It is easy to construct an input distribution in which the greedy assignment of weighted calls does not lead to an optimal solution with respect to the maximum weighted data rate.

4. The knapsack scheduler

In [15] an algorithm is described that solves the knapsack problem to optimality with an average case polynomial running time provided that several assumptions on the input distribution hold. The algorithm iteratively considers all items of the knapsack problem one after another in arbitrary order. In every iterative step it calculates the profits and weights of all pareto optimal solutions that can be constructed with all items considered so far. In the resource allocation problem, the profits considered by the algorithm are the weighted data rates and the weights are the transmission powers. Let Sol_i be the set of all weighted calls in the pareto optimal solution i. The profit of the pareto optimal solution i is

$$f_i = \sum_{k \in Sol_i} r'_k \,. \tag{3}$$

The weight of the pareto optimal solution i is

$$W_i = \sum_{k \in Sol_i} p_k \tag{4}$$

Let S_j be the set of all pareto optimal solutions after j items have been considered. S_{j+1} is constructed from S_j by first duplicating each solution in S_j, adding item j+1 to every solution in the duplicated set and eliminating all non pareto optimal solutions in both sets. This leads to two sets of solutions S_{j_1} and S_{j_2}. We obtain $S_{j+1} = S_{j_1} \cup S_{j_2}$. Let $|S_j|$ be the cardinality of the Set S_j and $|S_{\max}| = \max\{|S_j| \mid j \in \{1,...,n\}\}$ be the maximum cardinality of all sets of pareto optimal solutions. If we consider the cost of the operation $S_{j_1} \cup S_{j_2}$ to be $O(|S_j|)$, the cost of finding a solution to the knapsack problem with n different active calls is then $O(|S_{\max}| n)$. The KP-scheduler uses the same weighting scheme as the PF-scheduler. The resulting KP-problem on the weighted data rates is solved according to the description above.

4.1. Integer weights

We first consider an optimisation problem of type (2) with $\forall i : w_i \in N$. Remember that no connection is served on more than one code channel. Since the data rates specified in Table 1 can be mapped unambiguously to integer values we receive integer weighted r'_i values in the range of $\{1,..., R_{\max}\}$ where $R_{\max}$ is the maximum weighted data rate. It is not beneficial for the algorithm to keep more than one equally valued pareto optimal solution in any step. Therefore, our proposed algorithm will at any time only keep the first pareto optimal solution that yields a specific profit value. Observe that the running time of the algorithm can be bounded pseudo polynomial in this case since the step function of a solution S_j can have at most $jR_{\max}$ steps. After considering all n items, the algorithm will choose the solution with maximum profit that contains equal or less than $C_{\max}$ items. The observed maximum running time is therefore $O(nR_{\max} n) = O(n^2)$.

4.2. Real valued weights

In this section we consider an optimisation problem with real valued weights. As usual we consider the interference to the mobiles to be Gaussian distributed real values. As can be obtained from Table 1, the mapping of the interference to the data rate is some kind of step function with a finite number of steps. Putting these two together we obtain a discrete probability distribution on the data rate values which is controlled by the step function and the Gaussian distributed interference. This problem has been addressed in [15]. The authors obtain an average case running time of $O(R_{\max} P_{\max} n^5)$ for the KP-algorithm if the transmission powers p_i are real valued. This is no constraint in our case since only a finite number of transmission powers p_i are applied in a realistic environment.

4.3. Correlated instances

Until now we have only considered that a call with favourable SIR is transmitted at the maximum bit rate possible in any case. However, if the remaining transmission power at the base station does not suffice to transmit a call at the maximum bit rate, a transmission at lower bit rates might nevertheless be possible. It is clear that an additionally scheduled call, regardless of the actual data rate improves the overall throughput.

One approach to realise this adaptive scheduling is to provide the scheduler with all possible combinations of transmission powers and bit rates to every call. Since all possible power to bit rate combinations of every single call would then have to be modelled, the problem became much harder to solve. If we would for example consider only two entities of every call the corresponding optimisation problem is

$$\begin{aligned} &\max imise \ \sum_{i=1}^{nC_{\max}} a_i r_i^{(1)} + \sum_{i=1}^{nC_{\max}} (1-a_i) r_i^{(2)} \\ &subject\ to \ \sum_{i=1}^{nC_{\max}} a_i p_i^{(1)} + (1-a_i) p_i^{(2)} \le P_{\max} \\ &\qquad a_i \in \{0,1\}. \end{aligned} \tag{5}$$

In this optimisation problem $r_i^{(1)}$ and $r_i^{(2)}$ ($p_i^{(1)}$ and $p_i^{(2)}$) represent different data rates (transmission powers) for a connection to mobile i.

To avoid these complications, we employ the algorithm with additional knowledge regarding dependencies between power, interference and data rate. Clearly, the optimisation problem does not become more difficult, since additional information can be utilised by the algorithm. The optimisation of knapsack problems with dependencies between weight and profit have also been discussed in [18,19,20]. The authors present simulation results of various heuristic algorithms for the KP–problem. It can be observed that these algorithms solve correlated instances in a short amount of time, where time is measured in cpu time.

At the base station a measured SIR of a connection to one mobile is related to a unique data rate. The exact mapping from SIR to data rate depends on the implementation of the network operator. The function that describes this mapping may for example depend on the speed, position and type of the mobile. In the following discussion, we assume that this mapping is known to the algorithm. Let $D_i(x)$ be the function describing the mapping from SIR to data rate for a specific connection to mobile i. The SIR to this mobile i is known at the base station in advance of scheduling. It can be described by a function $SIR_I(p_i)$ that is dependent on the transmission power to mobile i and on the corresponding interference I.

$$SIR_I(p_i) = \frac{p_i}{\alpha \cdot I_{Intra} + I_{Extra} + N_0} = \frac{p_i}{I}. \tag{6}$$

In this formula, I_{Intra}, I_{Extra} and N_0 denote the intracell interference, extracell interference and noise. The factor α that is multiplied to I_{Intra} is the orthogonality factor that describes the strength of the intracell interference because of multi path propagation (compare [15]).

Since the interference is fixed for every single TTI, $SIR_I(p_i)$ is a linear mapping. For proper k_i we therefore obtain

$$SIR_I(p_i) = k_i p_i . \tag{7}$$

Since both, $SIR_I(p_i)$ and $D_i(x)$ are non-decreasing monotonous functions, we obtain

$$D(SIR_I(p_i)) = D(kp_i) = k_i p_i v_i . \tag{8}$$

for proper vi. If we further weight the different calls, we obtain a new optimisation problem as

$$\begin{aligned} &\max imise \ \sum_{i=1}^{nC_{\max}} a_i w_i k_i v_i p_i \\ &subject\ to \sum_{i=1}^{nC_{\max}} a_i p_i \le P_{\max} \\ &\qquad a_i \in \{0,1\} \\ &\qquad p_i \in [0, P_{\max}]. \end{aligned} \tag{9}$$

An optimal solution to this problem can be obtained by greedily scheduling the calls with maximum $w_i k_i v_i$ first while simultaneously keeping p_i as large as possible with respect to $P_{\max}$. If $D_i(x)$ is linear, we obtain for arbitrary two mobiles i and j the equality $v_i = v_j$.

5. Simulation

To illustrate the benefits of the proposed scheduling algorithm over the PF–scheduler, we consider an example simulation scenario in this section. In our simulation environment we model hexagonal sectionised cells (120°). The simulation environment is illustrated in Figure 2. All base stations are located in the corner of a cell with a distance of 2400m to each other. The maximum range of every base station is 1600m (the distance from one corner of a hexagon to the opposite corner).

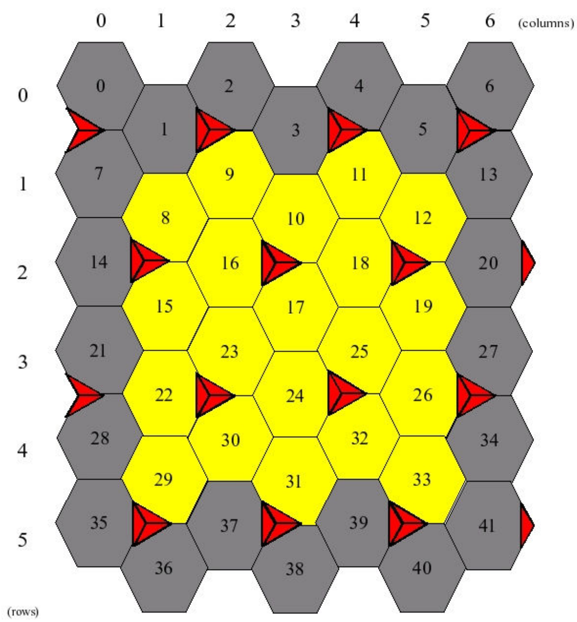

Figure 2 Simulation scenario

The link quality of any specific connection is determined by the distance to the base station, the transmission power of the serving base station and the maxi mum transmission power of all interfering base stations. The maximum transmission power of every base station is 16,596W. Contained in this transmission power is P_{HSDPA}, the power reserved for HSDPA at every base station. The aggregated transmission power of all HSDPA-connections must not exceed P_{HSDPA}.

Let α =0,4 be the orthogonality factor, P_κ the transmission power of base station κ, ζ_i the dumping of the signal due to the distance to receiver i (pathloss) and I_{extra} the interfering influence of the neighbouring cells. Further define N_0 to be the noise and SIR_{target} the quotient of own signal power to interference that is required for a specific data rate. The transmission power $P_{i,\kappa}$ to customer i is then calculated by

$$P_{i,\kappa} = \frac{\alpha P_\kappa + \frac{1}{\zeta_i} \cdot (I_{extra} + N_0)}{\frac{1}{SIR_{target}}}. \qquad (10)$$

For a derivation of this formula we refer to [21].

We will analyse the performance of the algorithm at base station 17 (compare Figure 2). To calculate the interfering influence of the neighbouring cells we consider the cells 15,18,23,24 and 25 since the base stations of these cells transmit in direction of cell 17. The influence of the neighbouring cells is calculated by

$$I_{extra}^{cell\kappa} = \sum_{\tau \in \{15,18,23,24,25\}} ((\zeta_\tau + \vartheta_\tau) \cdot P_\kappa) \qquad (11)$$

ϑ_κ specifies the antenna gain to base station κ. The pathloss ζ_κ between base station κ and a mobile station at distance d is calculated by

$$\zeta_\kappa = 15.3 + 37.6 \log_{10} d \qquad [dB] \quad [21] \quad (12)$$

Fluctuations in the channel conditions are modelled by a random function $f_{fluct} : R \to R$ that maps the calculated data rate r_i of a connection to mobile i to the actual obtained data rate for this connection. Every simulation is repeated 20 times for both, the knapsack scheduler and the proportional fair scheduling algorithm. We study the overall throughput of both scheduling algorithms for various maximum data rates $r_{\max}$ and transmission powers P_{HSDPA} that are available for HSDPA. The considered configurations are illustrated in Table 2. In all these simulations, we assume that a minimum count 15 customers is present in cell 17 at all time. From these set of customers, 10 users are transferring data over HSDPA. The amount of this data is for all mobiles fixed at 0.125 MBytes at the start of every simulation.

The data rates and transmission powers specified in Table 2 are chosen such that the following assertions hold in every simulation scenario.

	P_{HSDPA}	$r_{\max}$
Scenario 1	10.62 W	1.50 Mbps
Scenario 2	11.91 W	1.62 Mbps
Scenario 3	12.47 W	1.70 Mbps
Scenario 4	13.36 W	1.79 Mbps
Scenario 5	15.00 W	1.94 Mbps
Scenario 6	16.07 W	2.00 Mbps

Table 2 Various simulation scenarios.

1. Due to the power P_{HSDPA} available for HSDPA, only one mobile at a distance to the base station of 250m or more may be scheduled simultaneously.
2. Mobiles at a distance to the base station of less than 200m may be scheduled together in one TTI with another mobile at this distance.
3. For mobiles that are in between, both cases are possible depending on the actually calculated SIR of the mobile.

The probability to place a mobile at most 200m off the base station is P_A and the probability to place it farther off than 200m is P_{BC}. If we require all mobiles to be iid (independently and identically distributed) in the cell, we obtain $P_A \approx 0.02$ and $P_{BC} \approx 0.975$.

In every scenario we consider 10 active mobiles with 2 mobiles at most 200m away from the base station and 8 mobiles farther away than 200m. The probability that this scenario will actually occur with 15 users present in the considered cell is

$$1 - \left(1 - P_A^2 P_{BC}^8\right)^{\binom{15}{10}} \approx 0.7836. \qquad (13)$$

In all scenarios, both schedulers made use of the weighting of calls according to the proportional fairness scheme as described in section 3. When one of the 10 calls has been finished, we directly start a new one so that the overall load of the system is only slightly fluctuating. The results are presented in Figure 3. We observe that the transmitted data rate of the KP-scheduler is aproximately 20% above the data rate of the PF-scheduler in all cases.

In the top of Figure 3 we observe that the overall transmission time of the proportional fair scheduler is worse than the overall transmission time of the knapsack algorithm. The overall transmission time denotes the time necessary to transmit all requested data to every mobile. Since the knapsack scheduler performs better in this metric, even the connection with the worst link quality is completed earlier for the knapsack scheduler than with the proportional fair scheduler. The overall fairness of the system is therefore enhanced by the knapsack scheduling algorithm.

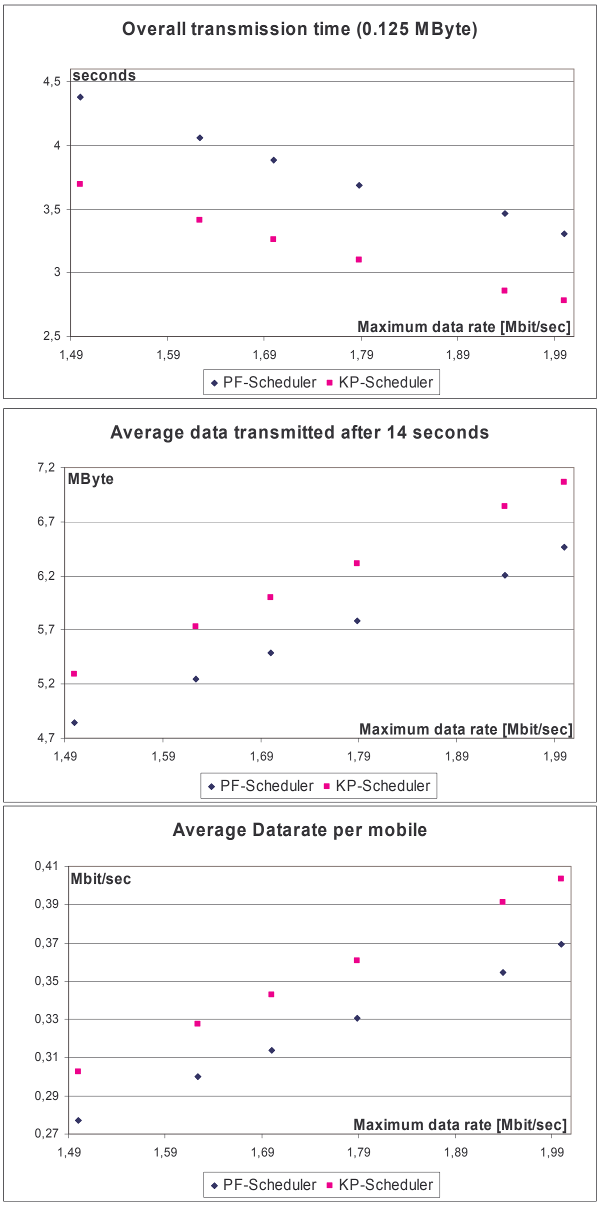

Figure 3 The KP-scheduler achieves a higher overall throughput than the PF-scheduler.

The obtained results support the conclusion achieved in section 4 that was based on the outcome of our analytical analysis. They further strongly suggest additional, more detailed studies of the knapsack scheduling algorithm for the resource allocation problem. To gain additional insight into the benfits of the knapsack scheduler over the proportional fair scheduler, a more realistic simulation scenario has to be constructed that also models additional features of HSDPA as the fast link adaptation mechanism, Chase combining and fast ARQ.

6. Conclusion

With the knapsack scheduler, we have proposed a fast scheduling algorithm, that achieves a higher overall throughput than the popular proportional fair scheduler and even solves the underlying scheduling problem in an optimal way. The average case running time of the proposed algorithm is only slightly higher than that of the proportional fair scheduling scheme. In a simulation we compared both scheduling algorithms and demonstrated the benefits of the KP-scheduling algorithm in a simple concrete scenario.

In addition to the rise in the overall transmitted data rate, the knapsack scheduler also managed to increase the average data rate. This is directly correlated to an improved fairness since the weighting of both algorithms follows the same proportional fairness weighting scheme. We have learned that the knapsack scheduler performs at least as well as the proportional fair scheduler in all possible cases. Since the additional running time of the KP-scheduler becomes less important with every new generation of Node B, network operators could consider implementing the knapsack algorithm for resource allocation.

We further analysed the relation between data rate and power of the active calls. Due to the additional information the problem becomes very straightforward and may be solved to optimality with a simple greedy scheduling scheme.

Another interesting problem not considered so far is the introduction of deadlines or due dates for every call. It is not clear, if the resource allocation algorithm following the knapsack scheduling scheme can be modified to handle such a more sophisticated problem with sufficient results.

References

[1] H. Holma and A. Toskala. *WCDMA for UMTS*, vol. 3. John Wiley and Sons, 2004.

[2] M. Andrews, K. Kumaran, K. Ramanan, A.L. Stolyar, R. Vijayakumar, and P. Whiting, "Scheduling in a Queueing System with Asynchronously Varying Service Rates," *Probability in the Engineering and Informational Sciences*, vol.18, pp. 191-217, 2004.

[3] R.C. Elliott et al. "Scheduling algorithms for the cdma2000 packet data evolution," in *Proc. Vehicular Technology Conference*, vol. 1, pp. 304–310, 2002.

[4] A. Jalali et al. "Data throughput of cdma-hdr a high efficiency-high data rate personal communication wireless system," in *Proc. Vehicular Technology Conference*, vol. 3, pp. 1854–1858, 2000.

[5] J.M. Holtzman. "Cdma forward link waterfilling power control," in *Proc. Vehicular Technology Conference*, vol. 3, pp. 1663–1667, 2000.

[6] H.v.d. Berg, R. Litjens, and J. Laverman. "Hsdpa flow level performance: The impact of key system and traffic aspects," in *Proc. MSWiM*, pp. 283–292, 2004.

[7] K. Balachandran, S. Nanda, and S. Kumar. „Adaptation techniques in wireless packet data services," *IEEE Communications magazine*, vol. 38, pp. 54-64, Jan. 2000.

[8] Q.L. Ding, P. Lin, B. Bensalou, and K.C. Chua. "A wireless fair scheduling algorithm for error-prone wireless channels," in Proc. *WOWMOM*, pp. 11-20, 2000.

[9] P. Ramanathan, and P. Agrawal. "Adapting packet fair queueing algorithms to wireless networks," in *Proc. MOBICOM*, pp. 1-9, 1998.

[10] T. Nandagopal, S. Lu, and V. Bharghavan. "A wireless fair service algorithm for packet cellular networks," in *Proc. MOBIOM*, pp 10-20, 1998.

[11] P.J.A. Gutierrez. "Packet Scheduling And Quality of Service in HSDPA," PhD thesis, Dep. of Comm. Tech. Inst. of Electronic Systems, Aalborg University, Oct. 2003.

[12] F. Kelly. "Charging and rate control for elastic traffic," in *Proc. European Transactions on Telecommunications*, vol. 8, pp. 33–37, 1997.

[13] S. Oh and K.M. Wasserman. "Optimality of greedy power control and variable spreading gain in multi-class cdma mobile networks," in *Proc. MOBICOM*, pp. 102–112, 1999.

[14] X. Liu, E.K.P. Chong and N.B. Shroff. "Transmission scheduling for efficient wireless utilization," in *Proc. IEEE INFOCOM*, pp. 776–785, April 2001.

[15] R. Beier and B. Vöcking. "Random knapsack in expected polynomial time," in *Proc. STOC*, 2003.

[16] N. Lee, J. Choi and S. Bahk. "Opportunistic scheduling for utility maximization under qos constraints," presented at the *16th Int. Symp. PIMRC*, Berlin, Germany, Sept. 2005.

[17] I. Wegener. *Theoretische Informatik – eine algorithmenorientierte Einführung.* 2nd. ed. B.G.Teubner, 1999. (in german)

[18] D. Pisinger, S. Martello, and P. Toth. "New trends in exact algorithms for the 0/1 knapsack problem," *European Journal of Operations Research*, vol. 123, pp. 325-332, 2000.

[19] S. Martello, and P. Toth. *Knapsack Problems – Algorithms and Computer Implementations.* Wiley, 1990.

[20] D. Pisinger. "Algorithms for Knapsack Problems," PhD thesis, University of Copenhagen, 1995.

[21] M. Hildebrand, G. Piao, K. David, G. Cristache, and F. Fechter. "Efficient estimation of transmission power applied to simulation of cell breathing effect in cdma-based wireless systems," presented at the *Int. Conf. AMS*, Cambridge, USA, 2002.