Wireless Networks 2 (1996) 27–43

Gradual removals in cellular PCS with constrained power control and noise *

Michael Andersin [a], Zvi Rosberg [b] and Jens Zander [a]

[a] *Radio Communication Systems, Department of Signals, Sensors and Systems, Royal Institute of Technology, S-100 44 Stockholm, Sweden*

[b] *Haifa Research Laboratory, Science and Technology, IBM Israel, Matam, 31905 Haifa, Israel*

Abstract. In this paper we study the *mobile removal problem* in a cellular PCS network where transmitter powers are constrained and controlled by a Distributed Constrained Power Control (DCPC) algorithm. Receivers are subject to non-negligible noise, and the DCPC attempts to bring each receiver's CIR above a given target. To evaluate feasibility and computational complexity, we assume a paradigm where radio bandwidth is scarce and inter-base station connection is fast. We show that finding the optimal removal set is an NP-Complete problem, giving rise for heuristic algorithms. We study and compare among three classes of transmitter removal algorithms. Two classes consist of algorithms which are invoked only when reaching a stable power vector under DCPC. The third class consist of algorithms which combine transmitter removals with power control. These are *One-by-one Removals, Multiple Removals*, and *Power Control with Removals Combined.* In the class of power control with removals combined, we also consider a distributed algorithm which uses the same local information as DCPC does. All removal algorithms are compared with respect to their outage probabilities and their time to converge to a stable state. Comparisons are made in a hexagonal macro-cellular system, and in two metropolitan micro-cellular systems. The *Power Control with Removals Combined* algorithm emerges as practically the best approach with respect to both criteria.

1. Introduction

Future PCS cellular networks will mainly be driven by high quality channels, high bandwidth utilization, low power consumption and efficient network management. Constrained power control (up-link and down-link) is one of several major techniques which is being studied to address these goals.

In PCS, cell sizes are small and transmission power is limited, exposing the receiver to more severe noise compared to larger cells where higher transmission power is used. This has been recently incorporated into the model in [12], where a power constrained control problem in a cellular network with cochannel interference and receiver noise, has been studied. (A more detailed version of [12] is found in [13].) The model there, and in this study, assumes a stationary link gain matrix, which is reasonable when the power control converges much faster than the link gain changes. The channel quality is measured by its Carrier to Interference Ratio (CIR). It is well known that there is a monotonically increasing relation between the CIR, and the channel symbol error rate. Thus, driving the CIR to some CIR target value, is the same as driving the channel to some capacity target. In practice, the CIR target is determined by the operator, based on the error rate that the decoder can tolerate.

* This research has been done while the 2nd author was visiting at the Royal Inst. of Tech. under a grant from ISS '90 foundation, Sweden.

As radio bandwidth is a scarce resource, channel allocation schemes may occasionally over-allocate transmitters to the same channel. The main reason for this is the fluctuating number of mobiles and their mobility, which may be hard to predict. This temporary over-allocation is not necessary something bad, as long as there is a Dynamic Channel Allocation (DCA) scheme which efficiently copes with this situation. In fact, efficient DCA will perform frequent re-allocations to achieve high capacity. Temporarily over-allocating a channel with up to 20–30% more transmitters than those that can be supported, may be quite common. A too defensive channel allocation on the other hand (e.g., conventional *fixed* channel allocation), where very few temporary over-allocations occur, usually leads to low resource utilization.

Consequently, in a power control process, when an over-allocation situation occurs in some channel and not all transmitters can be supported, some of them have to be *removed* from the channel. The removed transmitters are either transferred to another channel in the same cell, or to another cell, or being disconnected. Hand-off actions and disconnections are highly undesirable in cellular networks. Therefore, minimizing the number of removals, and the time to identify them, is of utmost importance. This is the problem we focus on in this paper. Note that in traditional systems with rare mobile outage, almost any mobile removal algorithm will perform equally well. However, for bold channel allocation (which requires efficient hand-off), differences among the algorithms will be manifested in practice.

The Distributed Constrained Power Control (DCPC) scheme in [13] (which is used as a building block in one of our removal schemes) aims at reducing the cochannel interference, hence maximizing bandwidth utilization. It has been shown that, it converges to a unique power vector, under synchronously and asynchronously power updates. The latter algorithm is denoted by ADCPC. Also, it has been demonstrated by numerical examples that asynchronous updates converge faster than synchronous updates. These algorithms have the following properties. For every CIR target and from every initial power vector, the updated powers converge to a unique positive power vector, under which some receivers exactly achieve their CIR target (i.e., *supported transmitters*), and the others are below their CIR target (i.e., *non-supported transmitters*). When all transmitters can achieve their CIR target, then the powers converge to a power vector where all CIRs equal their target. That is, all the transmitters are supported at their minimum required transmission power. Under any event, for any *supported* transmitter, DCPC and ADCPC drive its power to the minimum level where its corresponding receiver has a CIR which equals the CIR target. For any *non-supported* transmitter, the algorithm drives its power to the maximum power level (i.e., a power level which brings the transmitter as close as possible to its CIR target). Observe that the latter property may lead to *extremely oversized* non-supported set of transmitters, as will be shown in the examples in section 3. Hence, simply removing the non-supported set is over-doing (see the numerical examples in section 5), and a good removal algorithm is needed.

Other studies of centralized and distributed power control schemes are given in [1,2,4,8,10,11,17–22]. A related problem is mobile admission control, where a newly arrived mobile has to be admitted to, or rejected from the system. Clearly, the admission policy determines the potential number of removals. A defensive admission policy will result in rare removals, whereas an offensive one, will require more frequent removals. Both are appropriate, but offensive ones could better utilize the resources. The mobile admission problem is addressed in a complementary paper [3], where we derive a new Soft and Safe admission algorithm, under which the CIR of the active receivers does not drop below the target, at any moment of time. A similar approach to the admission control problem is also taken in [5].

Previous studies on mobile removals have been confined to noiseless and unconstrained power systems, and have derived centralized algorithms which assume the knowledge of the link gains. In a recent paper, [16], a family of centralized single transmitter removal algorithms has been proposed. Among those, the algorithm with the smallest outage probability is called SMIRA. It computes the largest eigenvalue of the gain matrix, λ^*, which relates to the maximum achievable CIR, γ^* according to $\gamma^* = 1/(\lambda^* - 1)$. Thus, it can determine whether or not all transmitters can be supported. If they can, then the eigenvector (up to a scaling constant), serves as the required powers. Otherwise, it removes one transmitter as follows, and re-iterates without that transmitter. To remove a transmitter, SMIRA associates to each transmitter a value which equals the *maximum between its received interference, and its transmitted interference*. The assumed transmission power vector is the eigenvector. Then, it removes the transmitter with the largest value. The study in [16] has demonstrated by numerical examples, that SMIRA outperforms a Stepwise Removal Algorithm (SRA), which has been earlier proposed in [21]. The specific SRA to which SMIRA has been compared, is the following. Each transmitter i is associated with an index which equals $Max\{$*the sum of link gains in row i; the sum of link gains in column i*$\}$, where the rows and the columns are those from the gain matrix. SRA removes the transmitter with the largest index. Note that the index in SRA does not incorporate the transmission powers.

In practice, removal algorithms must be distributed, and be based mainly on local measurements. Further, they must efficiently cope with transmitters mobility. These requirements are not met by the centralized algorithms above. Our objectives in this study, are therefore the following:

- To show that the removal problem is NP-complete.
- To provide a thorough examination and comparison among sensible removal algorithms.
- To show how to implement the algorithms (SMIRA and SRA, included) in a distributed manner.
- To devise a distributed algorithm which removes mobiles "on-the-fly" during the power control, and uses the same amount of local information as DCPC does.

To demonstrate the performance of the "on-the-fly" removal algorithm (referred to as *Distributed Gradual Removal DCPC*), we compare it to other distributed algorithms, and to a large class of algorithms which use more information. (This will put its performance in a wider perspective.) The comparison is done for a classical hexagonal macro-cellular system, and for a Manhattan-like micro-cellular system.

We confine our framework to the model which has been set in [12]. To evaluate feasibility and computational complexity, we adopt a paradigm where radio bandwidth is scarce and base stations are inter-connected by a *high-speed wire-line network*. Base stations are managing and distributively controlling the PCS network by exchanging and sharing information. Particularly, transmitter control actions are computed by the base stations, which then instruct the transmitters on their actions.

Previous studies have examined only step-wise removal algorithms. However, it is conceivable that other types of algorithms may work well. One way to classify them is as follows. Algorithms which are invoked only when reaching to a stable power vector under a power control scheme (e.g., DCPC). These algorithms can be further classified into those that may remove at most one transmitter, *One-by-one Removals*; and those that may remove multiple transmitters, *Multiple Removals*. After removal(s), both types resume the power control with the remaining set of transmitters. After reaching again to a stable power vector, the removal algorithm is re-invoked. A third and quite attractive class, consists of algorithms that combine transmitter removals with power control. That is, *Power Control with Removals Combined*. Note that combining resource allocation decisions with power control is also used for cell-site selection [15,20], and mobile admission [5,3].

From our case studies it turns out that, there is a trade-off between the number of removal steps until all transmitters are supported, and the outage probability. That is, one-by-one removals tend to have a lower outage probability, but they are clearly slower. It is *most intriguing* to investigate if there is a fast algorithm which also have low outage. Apparently, and we found it quite surprising, the *Power Control with Removals Combined* algorithm is the fastest, and has outage probability which is very close to that of the optimal removal algorithm in a macro-cellular hexagonal system.

Since the problem will be shown to be NP-complete, we search for good heuristic algorithms. The quality of removal decisions which do not rely on fairly good estimators of the link gains, may be limited. Therefore, we propose a computational paradigm which is feasible in the near future, and which facilitates the estimations of the link gains. We show that within this computational paradigm, a special form of the Asynchronous DCPC (ADCPC) algorithm can be used to evaluate the link gains with almost no additional computation. This will be used below, to show how SMIRA can be implemented distributively. It will also serve as a building block for new removal algorithms. (Note that SMIRA has been originally evaluated under noiseless and unconstrained powers, whereas we deal with receiver noise and constrained power.)

In section 2, we introduce the model and general definitions. In section 3, we derive basic results laying out the foundation for the algorithms we are proposing. In section 4, we specify the algorithms, and in section 5 we compare the algorithms with respect to their outage probabilities and their time to converge to a steady state. Comparisons are made in a hexagonal macro-cellular system, and in two metropolitan micro-cellular systems. Finally, we present our conclusions in section 6.

2. System model

We restrict our definitions to the uplink case (from mobile to base). The downlink is modeled in the same way, with the appropriate notational changes. The link propagation and the receiver noise variables are clearly different, but the model is still the same, and the results in this paper hold true for the downlink case as well.

Consider a cellular radio system and focus on a generic channel (a specific frequency or time slot). Assume that channels are orthogonal, so adjacent channel interference is negligible. Let $\mathcal{N} = \{1, 2, ..., N\}$, be the set of transmitters using this generic channel, and let $\boldsymbol{p} = (p_1, p_2, \ldots, p_N)$ denote the transmission powers used by the mobiles communicating with their corresponding base stations. We will add a time index to the powers, whenever necessary. That is, $\boldsymbol{p}(t) = (p_1(t), \ldots, p_N(t))$. The transmission power level is bounded from above by $\bar{p}$.

Denote the link gain matrix (see, e.g., [10] and [21]) by $\boldsymbol{G} = [g_{ij}]$, where g_{ij} is the gain of the radio link from transmitter j to base i, $1 \leqslant i, j \leqslant N$. All link gains assume values in the semi-open interval $(0, 1]$. Let $\boldsymbol{\nu} = (\nu_1, \ldots, \nu_N)$, be the receiver noise vector at the base stations. The noise vector in general is non-negative, and we require that at least one element is positive. As discussed in the Introduction, the link quality is measured by the Carrier to Interference Ratio (CIR). For a given power vector $\boldsymbol{p}$, the CIR at the base station used by transmitter i is given by

$$\gamma_i = \frac{p_i g_{ii}}{\nu_i + \sum_{j: j \neq i} g_{ij} p_j}, \quad 1 \leqslant i \leqslant N .$$

For notational convenience, we represent γ_i by

$$\gamma_i = \frac{p_i}{\eta_i + \sum_j a_{ij} p_j}, \quad 1 \leqslant i \leqslant N, \tag{1}$$

where $\eta_i = \nu_i / g_{ii}$, and

$$a_{ij} = \begin{cases} g_{ij}/g_{ii}, & \text{if } i \neq j, \\ 0, & \text{if } i = j. \end{cases}$$

The matrix and the vector of the normalized gains and noises are denoted by $\boldsymbol{A} = [a_{ij}]$ and by $\boldsymbol{\eta} = (\eta_1, \ldots, \eta_N)$, respectively. Also let $\bar{\boldsymbol{p}} = (\bar{p}, \bar{p}, \ldots, \bar{p})$.

We adopt the standard convention for matrix and vector inequalities. That is, for every two matrices or vectors $\boldsymbol{A}$ and $\boldsymbol{B}$, $\boldsymbol{A} \leqslant \boldsymbol{B}$, if the inequalities hold element-wise; and $\boldsymbol{A} < \boldsymbol{B}$, if $\boldsymbol{A} \leqslant \boldsymbol{B}$ and strict inequality holds for at least one of the elements.

We say that a power vector $\boldsymbol{p}$ supports all transmitters at a CIR target γ^t, if and only if

$$\boldsymbol{p} \geqslant \gamma^t (\boldsymbol{A}\boldsymbol{p} + \boldsymbol{\eta}) .$$

That is, each receiver i has a CIR $\gamma_i \geqslant \gamma^t$.

Other useful notations are the following:

$$\mathcal{C}(\gamma) = \{\boldsymbol{p} \geqslant \boldsymbol{0} : \boldsymbol{p} \geqslant \gamma(\boldsymbol{A}\boldsymbol{p} + \boldsymbol{\eta})\} .$$

$$\mathcal{C}^{-}(\gamma) = \{\boldsymbol{p} \geqslant \boldsymbol{0} : \boldsymbol{p} < \gamma(\boldsymbol{A}\boldsymbol{p} + \boldsymbol{\eta})\},$$

$$\mathcal{S}(\gamma) = \mathcal{C}(\gamma) \cap \mathcal{B}(\overline{p}),$$

$$\mathcal{S}^{-}(\gamma) = \mathcal{C}^{-}(\gamma) \cap \mathcal{B}(\overline{p}),$$

where $\mathcal{B}(\overline{p}) = \{\boldsymbol{p} : 0 \leqslant \boldsymbol{p} \leqslant \overline{\boldsymbol{p}}\}$.

These sets are graphically depicted in [13], and have the following geometrical interpretations. For $\gamma \leqslant \gamma^*$, the set $\mathcal{C}(\gamma)$ is an infinite polyhedral cone with tip at $\boldsymbol{p}^*$, where all receivers attain $\gamma_i = \gamma$. The set $\mathcal{C}^{-}(\gamma)$, is the inverse polyhedral cone truncated by the non-negative Euclidean subspace. For $\gamma > \gamma^*$, the set $\mathcal{C}(\gamma)$ is empty, and the set $\mathcal{C}^{-}(\gamma)$, is the non-negative part of the polyhedral cone generated by the N hyper-planes $\gamma_i = \gamma$ (where every i, $1 \leqslant i \leqslant N$, generates one hyper-plane). The sets $\mathcal{S}(\gamma)$ and $\mathcal{S}^{-}(\gamma)$, are the $\mathcal{C}(\gamma)$ and $\mathcal{C}^{-}(\gamma)$, truncated by the feasible power vectors box $\mathcal{B}(\overline{p})$, respectively.

Next, we describe the power updates made by the algorithms DCPC and ADCPC, when the target CIR is γ^t. Given the power vector at time t, $\boldsymbol{p}(t)$, and the set of transmitters updating their powers at time $t + dt$, $U(t)$, then

$$p_i(t+dt) = \begin{cases} \min\{\overline{p}, \gamma^t \cdot \frac{p_i(t)}{\gamma_i(t)}\} = \\ \quad \min\{\overline{p}, \gamma^t(\eta_i + \sum_{j\in\mathcal{N}} p_j(t)a_{ij})\}, & \text{if } i \in U(t), \\ p_i(t), & \text{otherwise}. \end{cases} \tag{2}$$

Note that $U(t)$ is an arbitrary set. Thus, any asynchronous power update is allowed (subject to some week conditions which exclude infinitely long intervals where a power is not being updated). If $U(t) = \mathcal{N}$, for every update instance t, then we get the synchronous DCPC algorithm. Otherwise, we get an arbitrary asynchronous version (ADCPC). A special ADCPC version, which we will use below, is the Round Robin ADCPC (RR-ADCPC), where transmitters update their powers one at a time, and in a Round Robin fashion.

Also note, that $p_i(t)g_{ii}/\gamma_i(t)$ is the interfering power (including the background noise) at receiver i. Since the receiver interference power can be measured, and g_{ii} can be detected by the transmitter from the base station pilot signal (assuming a reciprocal system), this algorithm can be implemented in a distributed manner. To exclude non-practical cases where a transmitter cannot overcome its receiver background noise, we assume that $\overline{p} > \gamma^t\eta_i, \forall i$.

It has been shown in [12], that for any given γ^t, DCPC and ADCPC converge to a unique positive power vector determined by the fixed point solution to

$$\boldsymbol{p} = \min\{\overline{\boldsymbol{p}}, \gamma^{\boldsymbol{t}}(\boldsymbol{A}\boldsymbol{p} + \boldsymbol{\eta})\}. \tag{3}$$

A power vector $\boldsymbol{p}$ which satisfies the fixed point equations in (3), will be referred to as the *stationary power vector*. When all transmitters can be supported, the DCPC (and ADCPC) converge to the fixed point solution to

$$\boldsymbol{p} = \gamma^t(\boldsymbol{A}\boldsymbol{p} + \boldsymbol{\eta}). \tag{4}$$

It will be useful to annotate the stationary power vector with its corresponding set of transmitters. That is, for every subset of transmitters $\mathcal{N}_0 \subseteq \mathcal{N}$, $\boldsymbol{p}^{\mathcal{N}_0}$ will denote the stationary power vector of a system which consists only of the set $\mathcal{N}_0$. Also, let $S_{\mathcal{N}_0}$ be the subset of transmitters which are supported (at γ^t) under the stationary power vector $\boldsymbol{p}^{\mathcal{N}_0}$ (i.e., in a system where DCPC runs only with the set of transmitters $\mathcal{N}_0$). Note, that this corresponds to a gain matrix and a noise vector which are obtained from $\boldsymbol{A}$ and $\boldsymbol{\eta}$, respectively, by removing the columns and the rows which do not correspond to $\mathcal{N}_0$. The respective gain matrix is denoted by $\boldsymbol{A}^{\mathcal{N}_0}$. From [12], it follows that

$$p_i = \overline{p}, \quad \forall i \in \overline{S}_{\mathcal{N}_0}, \tag{5}$$

where $\overline{S}$ denotes the complement set of S.

We note here, that all the results which have been obtained in [12] (from which some are cited here) holds in the following more general model. Each receiver i, has its own CIR target γ_i^t, and each transmitter i, has its own upper and lower power constraints, $\overline{p}_i$ and $\underline{p}_i$, respectively.

3. Basic results

In this section we develop the foundation for the heuristic algorithms which we study. The proofs of all the assertions below are given in the Appendix. To justify our heuristic approach we start by showing that the transmitter removal problem is NP-complete.

Proposition 1. The problem of finding the maximum number of transmitters that can be supported at a given CIR target, γ^t, is an NP-complete problem.

Next, we demonstrate the need of transmitter removal algorithms even when the DCPC power control scheme is being used. We will do so by showing that the set of non-supported transmitters under DCPC can be large, while only one removal brings the rest to their CIR target. Thus, simply removing the non-supported set may result in too many unnecessary removals. First consider a case with two transmitters where both of them cannot be supported in the same channel. The power regions indicating where each transmitter is supported, is depicted in Fig. 1. Let $\overline{p}$ be the maximum transmission power, and $\mathcal{C}^{-}(\gamma)$ be the region where both transmitters are not supported, given a target CIR of γ. When $(\overline{p}, \overline{p})$ is in the interior of $\mathcal{C}^{-}(\gamma)$, DCPC converges to $(\overline{p}, \overline{p})$, where both links are not supported. Clearly, by setting off the power of one of the transmitters, or driving the powers

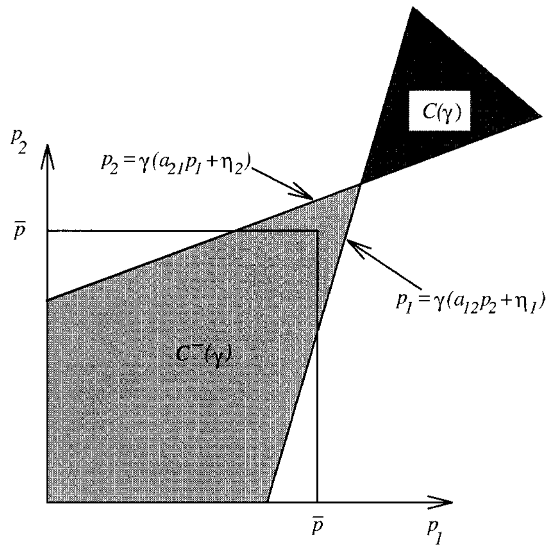


Fig. 1. Two transmitters where both links are not supported under DCPC.

to a point on the apex of the region $\mathcal{C}^-(\gamma)$, result in one supported transmitter.

A similar example can be constructed with any number of transmitters, where DCPC converges to $(\bar{p},\ldots,\bar{p})$, where none are supported, while there are powers where all but one, are supported. Another case with three transmitters, is where the DCPC stationary powers support only transmitter 3, while there are powers that can support transmitters 1, 2. This occurs when $(\bar{p},\bar{p},\bar{p}) \notin \mathcal{C}^-(\gamma)$, but $(\bar{p},\bar{p},x) \in \mathcal{C}^-(\gamma)$, for $0 < x \leqslant \alpha < \bar{p}$. In this case, DCPC converges to $(\bar{p},\bar{p},\alpha)$, where only transmitter 3 is supported. It is not hard to see that there is a power vector $(\bar{p},\bar{p},x)$, $x < \alpha$, where transmitters 1 and 2 are supported.

The rest of this section is sub-divided into three parts. In the first one, we show how to evaluate the matrix $\boldsymbol{A}$ during the power updates made by the RR-ADCPC algorithm. In the second subsection, we derive a sufficient condition to test if a subset of transmitters $\mathcal{N}_0 \subseteq \mathcal{N}$, can be supported under the stationary power vector $\boldsymbol{p}^{\mathcal{N}_0}$ (i.e., the fixed-point solution corresponding to the system with transmitter set $\mathcal{N}_0$). In the third subsection, we propose a generic DCPC algorithm with Gradual Removals (GR-DCPC). We show that a sub-class of these algorithms converge to a unique stationary power vector, under which the set of supported transmitters contains the supported set under the DCPC algorithm.

3.1. Gain matrix derivation

Consider the RR-ADCPC power update algorithm, and agree to identify the power update times with their corresponding sequential numbers. Thus, $t = 1, 2, 3, \ldots$ denote the power update times. By definition, at times $t = k \cdot N + j, k = 0, 1, 2, \ldots$, only transmitter j, $1 \leqslant j \leqslant N$, updates its power.

Thus, from (12.) and (2) it follows that for every update step $(k \cdot N + j), k = 0, 1, 2, \ldots$,

$$\frac{p_i(k \cdot N + j)}{\gamma_i(k \cdot N + j)} - \frac{p_i(k \cdot N + j - 1)}{\gamma_i(k \cdot N + j - 1)} = a_{ij}[p_j(k \cdot N + j) - p_j(k \cdot N + j - 1)], \quad 1 \leqslant i, j \leqslant N. \tag{6}$$

Note that p_i/γ_i is the interference which can be measured by receiver i, and that each base i, knows its corresponding mobile transmission power p_i, at any time. Thus, by distributing the transmitter powers, every base i can compute the gains a_{ij}, $1 \leqslant j \leqslant N$, after every update time $(k \cdot N + j), k = 0, 1, 2, \ldots$. To get the full matrix $\boldsymbol{A}$ in each base, the vectors $\mathrm{a}_{i,.} = (a_{i1}, \ldots a_{iN})$, are distributed among the base stations through the wire-line connecting network.

Note that RR-ADCPC is not a strict requirement to obtain the link gains. It is only necessary that the powers are being updated one at a time. Also note that this distributed computational procedure of $\boldsymbol{A}$ is enabled by our paradigm, and it is applicable to non-stationary gain matrices. It uses only the fact that each link gain stays stable between two consecutive power updates of the corresponding transmitter.

Hereinafter, we assume that the matrix $\boldsymbol{A}$ is readily available for our algorithms (if needed).

3.2. Bounds on the maximum removals

In this subsection we derive measure functions which facilitate transmitter removal algorithms, and a sufficient condition to test if a subset of transmitters can be supported. The latter will be used to derive an upper bound on the number of required removals.

For every feasible power vector $\boldsymbol{p}$, and every subset $\mathcal{N}_0 \subseteq \mathcal{N}$, define

$$\alpha_j(\boldsymbol{p}) = \gamma^t(\eta_j + p_j \sum_{i \in \mathcal{N}} a_{ij}) - p_j \quad (1 \leqslant j \leqslant N) \ ,$$

$$\beta_j(\boldsymbol{p}) = \gamma^t(\eta_j + \sum_{i \in \mathcal{N}} a_{ji} p_i) - p_j \ ,$$

$$D^{\mathcal{N}_0}(\boldsymbol{p}) = \sum_{j \in \mathcal{N}_0} \beta_j(\boldsymbol{p}) \, . \tag{7}$$

Also, let

$$p_i^{\mathcal{N}/\mathcal{N}_0} = \begin{cases} p_i^{\mathcal{N}}, & \text{if } i \in \mathcal{N}_0 \, , \\ 0, & \text{otherwise} \, . \end{cases} \tag{8}$$

Note that $\alpha_j(\boldsymbol{p})$ measures the total excess interference power resulting from the transmission of transmitter j, and $\beta_j(\boldsymbol{p})$ measures the excess interference power which the receiver of j is experiencing. The sum $D^{\mathcal{N}_0}(\boldsymbol{p})$, simply measures the total excess interference power which all the receivers in subset $\mathcal{N}_0$ are experiencing.

To simplify the notation we agree to use $\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}$ in two ways. One is a N tuple whose elements are defined in (8), and the other is the N_0 tuple, which is the reduction to the elements corresponding to the set $\mathcal{N}_0$. The same notational convention will be used for other vectors.

Some useful properties of these measures, which will be used later on, are given below. By exchanging the summation order, it is easy to verify that

$$D^{\mathcal{N}}(\boldsymbol{p}) = \sum_{j\in\mathcal{N}} \beta_j(\boldsymbol{p}) = \sum_{j\in\mathcal{N}} \alpha_j(\boldsymbol{p}) .$$

Also, from (3) we have

$$\begin{aligned} &\beta_j(\boldsymbol{p}^{\mathcal{N}}) \geqslant 0 , && \text{for } 1 \leqslant j \leqslant N , \\ &D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) \geqslant 0 , && \\ &D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) = 0 , && \text{if and only if all transmitters are supported under the stationary power vector} . \end{aligned} \tag{9}$$

After removing some of the transmitters, one would expect that the new stationary power will not increase. This is indeed shown in the following Lemma, and will be used for subsequent results.

Lemma 2. For every subset $\mathcal{N}_0 \subseteq \mathcal{N}$, we have $\boldsymbol{p}^{\mathcal{N}_0} \leqslant \boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}$.

A direct consequence of Lemma 2 is that all transmitters in $\mathcal{N}$ which are supported under $\boldsymbol{p}^{\mathcal{N}}$, and are not removed from the system, will remain supported under the new stationary power vector $\boldsymbol{p}^{\mathcal{N}_0}$. This is stated in the Corollary below.

Corollary 3. For every subset $\mathcal{N}_0 \subseteq \mathcal{N}$, we have $S_{\mathcal{N}} \cap \mathcal{N}_0 \subseteq S_{\mathcal{N}_0}$.

In the next theorem we derive some relations between the measure functions $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})$, $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}_0})$ and $D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}})$, which will justify their use for transmitter removals. The algorithms which are based on this Theorem are presented in section 4.

Theorem 4. For every subset $\mathcal{N}_0 \subseteq \mathcal{N}$, we have $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}) \leqslant D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}})$, and $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}_0}) \leqslant D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}})$.

From the assertion in Theorem 4 and (9), the functions $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})$ and $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}_0})$ measure in some respect, how likely a subset of $\mathcal{N}_0$ transmitters can be supported under its stationary power vector. Notice however, that given the stationary power $\boldsymbol{p}^{\mathcal{N}}$, the functions $D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}})$ and $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})$ are known, whereas $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}_0})$ is not. Thus, the former ones can be used to evaluate a removal set. In section 4, we show how they facilitate removal algorithms. Note also, that $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})$ may drop below zero.

A useful device in a removal algorithm is a fast computational procedure to test whether or not a subset of transmitters can be supported. It would be nice to have a device for this purpose, which is based just on the function $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})$. However, we were not successful in finding such one. Nevertheless, a simple and fast test function which is based on the stationary power vector $\boldsymbol{p}^{\mathcal{N}}$, is given in the next theorem.

Theorem 5. For every subset $\mathcal{N}_0 \subseteq \mathcal{N}$, if $\gamma^t(\eta_i + \sum_{j\in\mathcal{N}_0} a_{ij} p_j^{\mathcal{N}}) \leqslant \overline{p}$ for every $i \in \mathcal{N}_0$, then $S_{\mathcal{N}_0} = \mathcal{N}_0$ (i.e., all transmitters in $\mathcal{N}_0$ are supported under $\boldsymbol{p}^{\mathcal{N}_0}$).

In section 4, we show how to apply Theorem 5 to derive a non-trivial upper bound on the number of required removals. We will also use it for selecting multiple removals.

3.3. A generic gradual removal DCPC

In this subsection, we propose a removal scheme which combines power control with removal decisions. This approach is quite attractive as it can be implemented as an on-the-fly algorithm. Furthermore, it can be assembled with mobile admission and cell-site selection [15,20,5,3], which also combine their decisions with power control.

The generic combined scheme (*Gradual Removals X DCPC (GRX-DCPC)*), which we propose, consists of two elements. One is a specification of those instances where removal decisions are made before continuing with the power control. The other, is the arbitration rule used to select a single removal. The removal instances which we use, are those where at least one of the receivers experiences an interference power which cannot be over-come by its corresponding transmitter (due to the power level constraint). For the arbitration rule we leave a lot of freedom, and it will mainly depend on the amount of additional measurement and information distribution one is willing to invest. A precise definition of the algorithm is given below. The main idea of the GRX-DCPC algorithm is to remove transmitters that cannot possibly be supported under DCPC, in an early stage of the power update process. Thus, reducing interference and potentially leading to a larger set of supported transmitters. The X in the notation GRX-DCPC stands for either R (Restricted), or N (Non-restricted), which classifies the set from which a mobile is being removed.

We show that under any algorithm in GRR-DCPC, the power vectors converge to a stationary power vector under which all remaining transmitters are supported. Moreover, every transmitter which is supported under DCPC, is also supported under this scheme. The convergence though, holds for every algorithm in GRX-DCPC.

In the following definition we adopt a programming pseudo-code notation $X \leftarrow Y$ to denote that X is substituted by Y. Also, the set $\mathcal{R}$ below denotes the set of removed mobiles.

Generic GRX-DCPC algorithm

Start with the complete set of transmitters $\mathcal{N}$ and set $\boldsymbol{p}(0) \in \mathcal{C}^{-}(\gamma^t)$, $\mathcal{R} = \emptyset$, and $k = 0$. For every step k, update the power vectors as follows, and stop when any pre-determined convergence condition is satisfied:

(1) For every $i \in \mathcal{N} \backslash \mathcal{R}$, Set $p_i(k+1) = \min\{\overline{p}, \gamma^t p_i(k)/\gamma_i(k)\}$ (as DCPC does), and for $i \in \mathcal{R}$, set $p_i(k+1) = 0$.

(2) Let $\Psi = \{j : \overline{p} < \gamma^t \frac{p_j(k)}{\gamma_j(k)}\}$. If $\Psi \neq \emptyset$, then take the following meta-step:

- **(2.1)** If (X=R, i.e., Restricted), set $\Omega = \Psi$. Otherwise (i.e., Non-restricted), set $\Omega = \mathcal{N} \setminus \mathcal{R}$.
- **(2.2)** Remove from Ω a single transmitter j_0, according to any given arbitration rule.
- **(2.3)** Set $p_{j_0}(k+1) \leftarrow 0$, and $\mathcal{R} \leftarrow \mathcal{R} \cup \{j_0\}$.

(3) Set $k \leftarrow (k+1)$ and go to 1.

In the following theorem we prove that the GRR-DCPC algorithm (i.e., for X=R) above, converges to a stationary power vector, which supports at least those transmitters which are supported by the DCPC stationary power vector.

Theorem 6. Under either synchronous or asynchronous power updates, the Generic GRR-DCPC algorithm converges to a stationary power vector with the following properties:

(a) All the remaining transmitters are supported with the target CIR, γ^t.

(b) The set of remaining transmitters contains the set of transmitters which are supported by the stationary power vector of DCPC.

Remark 1. The result that under the Generic GRR-DCPC algorithm, $\mathcal{R} \subseteq \overline{\mathcal{S}}_{\mathcal{N}}$, relies heavily on the fact that the initial power vector is in the set $\mathcal{C}^{-}(\gamma^t)$. In practice, this is a drawback of the algorithm, which will be addressed in subsection 4.3

4. Removal algorithms

In this section we derive one-by-one, multiple, and power control combined removal algorithms. The algorithms will be based on the properties proven in Theorems 4, 5 and 6.

Recall that the one-by-one and multiple removal algorithms are taking the following approach. An underlying power control without removals is assumed. It updates transmitter powers, and converges to a stationary power vector. When reaching a stationary power vector, some of the transmitters may not be supported, in which event a one-by-one or a multiple removal algorithm is invoked. The removal algorithm removes one or more transmitters, and the power control resumes with the remaining ones. After reaching to a new stationary power vector, the removal procedure is repeated. A removal algorithm specifies how to select the set of removals under the stationary power vector.

In the first subsection, we specify an algorithm to compute an upper bound, R^*, on the number of required removals. This upper bound is used to set an upper limit on the number of removals in a multiple removal algorithm. The removal algorithms are specified in the second subsection. In the third subsection we specify a set of power control algorithms which combine power updates with removals. This is a novel approach, which also emerges as practically the best one in our numerical examples.

4.1. Upper bound to the number of removals

To compute an upper bound R^*, to the number of required removals, we assume that the stationary powers and the size of the non-supported set $|\overline{\mathcal{S}}_{\mathcal{N}}|$, are given. This information needs to be passed to the base stations which execute the algorithm. We also assume that the matrix $\boldsymbol{A}$ has been computed by the procedure presented in section 3.1, and is available for any of the base stations which executes the algorithm. The algorithm can be executed by a single, or by all of the base stations. In the latter case, no distribution of the result is needed. Note that a trivial upper bound to R^* is $|\overline{\mathcal{S}}_{\mathcal{N}}|$.

An algorithm to derive an upper bound

For every $k = 0, 1, \ldots, |\overline{\mathcal{S}}_{\mathcal{N}}| - 1$, do:

1. Check the condition in Theorem 5 for all subsets $\mathcal{N}_0 \subset \mathcal{N}$ of cardinality k.
2. If a set which satisfies the condition is encountered, then stop and set $R^* = k$.
3. Otherwise, increment k and continue.
4. If the condition in Theorem 5 is not satisfied for any k, then set $R^* = |\overline{\mathcal{S}}_{\mathcal{N}}|$. (In this case, the optimal subset of removals is $\overline{\mathcal{S}}_{\mathcal{N}}$.)

Observe that during the computation, candidate subsets for removals are also identified. The computational complexity of this algorithm is $O(N^{|\overline{\mathcal{S}}_{\mathcal{N}}|-1})$. Thus, it is tractable only for small sizes of non-supported sets. For cases where the set $\overline{\mathcal{S}}_{\mathcal{N}}$ is too large, one may stop looping when k reaches to a pre-determined threshold. When this occurs, simply set R^* to the trivial upper bound $|\overline{\mathcal{S}}_{\mathcal{N}}|$.

4.2. One-by-one and multiple removals

From (9), Theorem 4 and its subsequent discussion, the functions $D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})$ and $D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}})$ measure in some respect, how likely the subset of transmitters $\mathcal{N}_0$, can be supported if the others are removed. The precise algorithm is specified below, and it assumes the knowledge of the stationary power vector, and the matrix $\boldsymbol{A}$. As for the upper bound, the latter is being computed by the procedure presented in section 3.1. The algorithm can be executed by a single, or by all of the base stations. In the latter case, no distribution of the result is needed.

We start by showing that the one-by-one SMIRA algorithm can be derived based on the measure function $D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}})$. The lower it is being reduced by removing a single transmitter j, the better is the prospect to support the remaining transmitters $\mathcal{N} \setminus \{j\}$. Observe, that it can be reduced either by removing an element $\alpha_j(\boldsymbol{p}^{\mathcal{N}})$, or by removing an element $\beta_j(\boldsymbol{p}^{\mathcal{N}})$. Thus, the transmitter j which attains the

$$\max_{j\in\mathcal{N}}\{\max(\alpha_j(\boldsymbol{p}^{\mathcal{N}});\beta_j(\boldsymbol{p}^{\mathcal{N}}))\}$$

is a natural candidate for removal. This selection criterion is equivalent to the SMIRA algorithm. Note however, that here it is extended to the case with power constraints and receiver noise.

In the sequel, the number of removals, R, are parameterized into a unified algorithm. Thus, one-by-one and multiple removals are special instances of the same algorithm. The parameter R can be determined in several ways. One is to set $R = \min\{R^*, M\}$, where M is a user defined parameter reflecting his personal trade-off between fast and safe. Another is to set $R = \lfloor qR^* \rfloor$, where $0 \leqslant q \leqslant 1$, is a user defined proportion. (Here, $\lfloor x \rfloor$ denotes the largest integer smaller than or equals x.)

From the discussion following Theorem 4, a more promising measure function than $D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}})$, is the difference $D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) - D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})$. Again, the lower it is being reduced by removing a subset of transmitters $\mathcal{N} \setminus \mathcal{N}_0$, the better is the prospect that $\mathcal{N}_0$ will be supported. The new removal algorithm (referred to as, Single or Multiple Accumulative Removals Technique -SMART(R)), is then as follows:

SMART(R) algorithm

Given the number of required removals R, remove the set $\mathcal{N} \setminus \mathcal{N}_0$ with R transmitters which attains

$$\max_{\mathcal{N}_0\subset\mathcal{N}}\{D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) - D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})\} = \max_{\mathcal{N}_0\subset\mathcal{N}}\{\sum_{j\notin\mathcal{N}_0}[\alpha_j(\boldsymbol{p}^{\mathcal{N}}) + \beta_j(\boldsymbol{p}^{\mathcal{N}}) + (p_j^{\mathcal{N}} - \gamma^t\eta_j)]\}\,.$$

The computational complexities of SMART(1) and SMIRA are the same, and are dominated by the computation of $\{\alpha_j(\boldsymbol{p}), \beta_j(\boldsymbol{p}), 1 \leqslant j \leqslant N\}$. This is done in $O(N^2)$ basic operations. Sorting them can be done in $O(NlogN)$ basic operations. Thus, the computational complexity of SMART(1) is $O(N^2)$.

For $R > 1$, the computational complexity is dominated by N^R, an upper bound on the number of subsets of size R. An $O(N^2)$ approximation is to remove the R transmitters with the R largest $[\alpha_j(\boldsymbol{p}^{\mathcal{N}}) + \beta_j(\boldsymbol{p}^{\mathcal{N}}) + (p_j^{\mathcal{N}} - \gamma^t\eta_j)]$.

Remark 2. Note that the sum $\alpha_j(\boldsymbol{p}^{\mathcal{N}}) + \beta_j(\boldsymbol{p}^{\mathcal{N}})$ in the criterion above equals the transmitting interference plus the receiving interference, minus $2p_j^{\mathcal{N}}$. In the criterion above, one $p_j^{\mathcal{N}}$ is canceled out, and the other remains. It would make more sense to factor out also the other $p_j^{\mathcal{N}}$. The reason is that if two subsets result in equal reductions, we would rather remove the one with the higher sum of powers, and not the one with the lower sum of powers as the criterion above suggest. Indeed, it turned out that this results in a slightly better outage probability.

Another removal algorithm can be deduced from Theorem 5. Suppose we remove the set of transmitters $\mathcal{N} \setminus \mathcal{N}_0$. Thus, their power is set to zero, and we resume the power control with the initial power vector $\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}$ (i.e., the previously stationary power vector after removing the elements corresponding to the removed subset).

Let $\boldsymbol{p}(K, \mathcal{N}_0)$ be the powers after K updates, as being computed by the following DCPC algorithm iteration:

$$\boldsymbol{p}(k+1) = \min\{\mathrm{p}, \gamma^t(\boldsymbol{A}^{\mathcal{N}_0}\boldsymbol{p}(k) + \boldsymbol{\eta})\},$$
$$k = 0, 1, \ldots, K-1\,.$$

To derive $\boldsymbol{p}(K, \mathcal{N}_0)$, each base station can emulate the update steps without actually implementing them. What can we learn from the value $\boldsymbol{p}(K, \mathcal{N}_0)$, which may assist us in deciding which subset $\mathcal{N} \setminus \mathcal{N}_0$ to remove? Observe, that from the proof of Theorem 5, the remaining subset $\mathcal{N}_0$ is supported since,

$$\boldsymbol{p}(1, \mathcal{N}_0) \leqslant \boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}\,. \tag{10}$$

This, subsequently results in a non-increasing sequence of power vectors. However, when we restrict ourselves to a limited number of removals, we cannot guarantee the inequality in (10). Alternatively, we can select a set $\mathcal{N}_0$, which results in the largest reduction in a single DCPC update iteration. More general, select the set $\mathcal{N}_0$ which results in the largest reduction after K DCPC update iterations. This is made precise in the following Power Reduction Removal Algorithm in K iterations - PRRA-K(R).

PRRA-K(R) algorithm

Given the number of look-ahead steps K, and the number of required removals R, remove the set $\mathcal{N}_0$ with R transmitters which attains

$$\max_{\mathcal{N}_0\subset\mathcal{N}}\{\sum_{j\notin\mathcal{N}_0} p_j^{\mathcal{N}} + \sum_{j\in\mathcal{N}_0}[p_j^{\mathcal{N}} - p_j(K, \mathcal{N}_0)]\}\,.$$

This algorithm requires the same information as SMART(R) requires, and can also be executed by one, or all the base stations. The computation consists of the DCPC power updates, which is $O(N^2)$; sorting which is $O(NlogN)$; and the construction of all subsets of size R, which is N^R. Thus for $R = 1$, the complexity is $O(N^2)$, and for $R > 1$, the complexity is $O(N^R)$.

To summarize, SMIRA, SMART and PRRA-K, all require the same amount of information which can be evaluated distributively, and be distributed to the base stations which execute the algorithms. For a single removal, all algorithms have computational complexity of $O(N^2)$, and for $R > 1$ removals, the complexity is $O(N^R)$. Thus, in this respect they are equivalent, and the most prefered will be the one which results in the lowest outage probability. Such a comparison will be given in section 5.

4.3. Power control with removals combined

Here, we present a novel approach to the mobile removal problem. Whereas, the one-by-one and the multiple removal algorithms above, remove transmitters only after reaching a stationary power vector, the power control with removals combined, removes them during power updates. In this respect, this algorithm is more efficient than the previous ones. Also, as transmitters are removed earlier, their interference impact is relieved sooner, giving rise to a larger supported set. Moreover, the most important property is that one of its versions *can be implemented **distributively** using the same amount of information that DCPC uses*.

In section 3.3, we introduced the Generic GRX-DCPC class of algorithms. A specific algorithm in this class is determined by selecting $X = R$ or N, and a specific *Arbitration Rule* which is used to remove a transmitter from Ω, in meta-step (2.2). Except for this step, the information and power update are the same as in DCPC. Here, we specify two types of arbitration rules. One uses the current power vector and the matrix $\boldsymbol{A}$. This requires the gain matrix derivation procedure in subsection 3.1, and the distribution of the current power vector when the event $\Psi \neq \emptyset$ occurs (see step (2) of the GRX-DCPC algorithm). The other type of arbitration rule is a distributed one, which uses no further information.

Another issue which needs attention, is how to deal with arbitrary initial powers for the GRR-DCPC class. We show that one may use DCPC in a way which will safely drift any initial power vector to $\mathcal{C}^-(\gamma^t)$, without deteriorating the CIR of the transmitters which are already supported.

A general arbitration rule

An arbitration rule Π selects from several candidates, one transmitter for removal. The selection is based on the current power vector $\boldsymbol{p}$, and the gain matrix $\boldsymbol{A}$. This is precisely what a one-by-one removal algorithm does. Therefore, natural candidates for Π are SMIRA and SMART(1), where the set of candidates is Ω, and the stationary power vector is any relevant $\boldsymbol{p}$.

To avoid coordination when the event $\Psi \neq \emptyset$ occurs, and to rely only on local information, we may use the following distributed rule. Observe that each transmitter knows whether or not it belongs to the set Ψ. In addition, we rather have in each step a single removal than multiple removals. On account of a complete distributed algorithm, this can be obtained only with probability. Let δ be some given success probability of a coin tossing, which is tuned by the system to maximize the probability of exactly one removal, when the event $\Psi \neq \emptyset$ occurs.

A distributed arbitration rule

Each transmitter in the set Ψ removes itself with probability δ, and stays with probability $1 - \delta$. Other transmitters stay with probability one.

In systems where it is most likely that Ψ consists of a single transmitter, δ is taken to be close to 1.

Drifting to $\mathcal{C}^-(\gamma^t)$

For any initial power vector $\boldsymbol{p}$, let $\mathcal{S}$ be the set of transmitters which are supported under the target CIR, γ^t. To drift the power vector into $\mathcal{C}^-(\gamma^t)$, without letting the CIRs of any transmitter in $\mathcal{S}$ drop below γ^t, do:

1. For every $i \notin \mathcal{S}$, temporarily reset its power to zero. That is, $p_i = 0$.
2. Execute DCPC only with the set of transmitters $\mathcal{S}$.
3. Stop when DCPC converges according to a given convergence stopping criterion.

From [12, Lemmas 6 and 9], the power vectors of the transmitters in $\mathcal{S}$, monotonically decrease to a power vector (while keeping all transmitters in $\mathcal{S}$ supported at any time), satisfying

$$p_i = \gamma^t(\eta_i + \sum_{j \in \mathcal{S}} a_{ij} p_j), \quad \forall i \in \mathcal{S}.$$

Since all other powers have been set to zero, the combined power vector after convergence is in $\mathcal{C}^-(\gamma^t)$.

To summarize, we have two basic classes of GRX-DCPC(Π) removal algorithms, GRR-DCPC(Π) and GRN-DCPC(Π). For the former one, we may use a general or a distributed arbitration rule Π. For the latter, only a general arbitration rule can be applied. In the following section we will see how they perform with respect to other algorithms.

5. Numerical examples

The removal algorithms are compared in a macro-cellular system and in a micro-cellular system. The macro-cellular system which we use is a standard hexagonal cell plan, e.g., [16]. This system is selected mainly for comparison with the SMIRA algorithm proposed in [16].

For a micro-cellular system, we use two cell plans in a Manhattan-like metropolitan environment, [14]. The details of the cell plans are given below.

The link gain, g_{ij}, is modeled as a product of two variables, $g_{ij} = l_{ij} \cdot s_{ij}$. The variable l_{ij} is the large scale propagation loss, which depends on the transmitter and receiver locations, and on the type of topology environment (see below). The variable s_{ij} is the variation in the received signal due to shadow fading. We assume that the variables s_{ij}'s are independent, and log-normally distributed with a mean of 0 dB, and a log-variance of σ^2. The value of σ is environment dependent.

For each cell plan we use a fixed channel assignment scheme that divides the cells into N_c different channel groups. The maximum transmitter power is set to 1 W, and the receiver noise is taken to be 10^{-15} W.

A hexagonal macro-cellular system

All cells have an hexagonal shape with a radius of 1 km, and each base station is located at the center of its cell. Base stations use omnidirectional antennas and $N_c = 7$. This cell plan is depicted in Fig. 2, with 19 cochannel cells, which is the number we use in our simulation. The locations of the mobiles are independently sampled from a uniform distribution over each cell area and the mobiles are connected to the closest base station. The large scale propagation loss is modeled as $l_{ij} = 1/d_{ij}^4$, where d_{ij} is the distance between mobile i and base station j. The log-variance of the shadow fading is taken as $\sigma = 6$ dB. This is the same setting which has been used in [16].

A Manhattan-like micro-cellular system

This is a typical metropolitan environment consisting of building blocks of a square shape, Figs. 3, 4. Streets are running between the building blocks in two directions, horizontal and vertical. In our simulation we assume that each block is of length 100 m. We further assume that radio-waves can propagate only along the streets. To model the large scale propagation loss, set

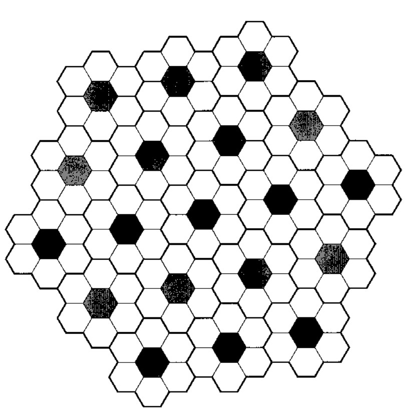


Fig. 2. Hexagonal cell plan with 7 cell cluster and 19 cochannel cells.

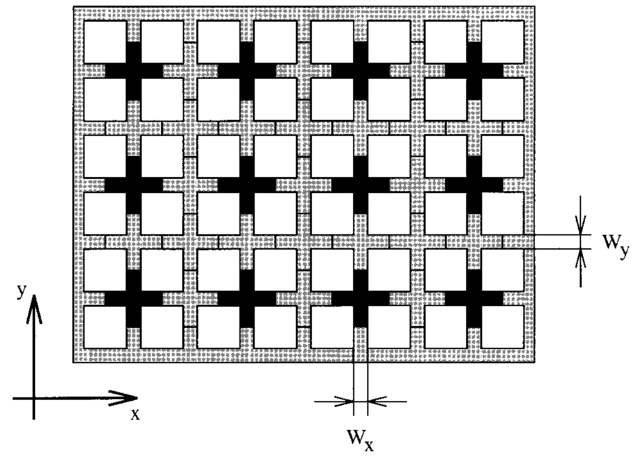


Fig. 3. The symmetric SHS(2,0) cell plan with cluster size $N_c = 4$. The dark crosses are the cochannel cells and the white squares are the buildings seen from above.

(x_i, y_i) and (x_j, y_j) to be mobile i and base station j coordinates, respectively.

Denote by $x = |x_i - x_j|$ and $y = |y_i - y_j|$, the horizontal and the vertical distances, respectively, between the mobile and the base station. From [6], the large scale propagation loss between mobile i and base station j can be modeled by

$$l_{ij} = \left[16 \frac{\pi^2 f^2}{c^2} \left(xye^{-\left(\frac{20 W_x W_y}{xy}\right)} + x + y + 10 \right)^2 \times \left(1 + \sqrt{ \left(\frac{x+y}{L_n} \right)^{(2n-4)} + \left(\frac{x^2+y^2}{L_m{}^2} \right)^{(m-2)} } \right) \right]^{-1},$$

where c is the speed of light, f is the transmission frequency, and W_x and W_y are the street widths in the horizontal and vertical direction, respectively. The parameters n, L_n, m and L_m are all propagation constants, [7].

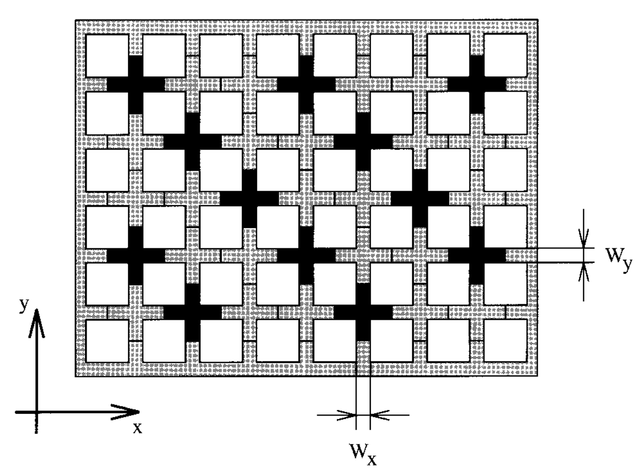


Fig. 4. The asymmetric AHS(1,1) cell plan with cluster size $N_c = 3$. The dark crosses are the cochannel cells and the white squares are the buildings seen from above.

In our simulation we use $f = 900$ MHz, $W_x = W_y = 25$ m, $n = 4$, $m = 25$, $L_n = 200$ m and $L_m = 700$ m. The log-variance of the shadow fading is taken as $\sigma = 4dB$, which is typical for metropolitan areas, [7].

Since in the literature, there are few results on the performance of power control algorithms in metropolitan micro-cellular systems, we study the removal algorithms for two different cell plans. The first one, a *Symmetric Half Square (SHS)* cell plan, is depicted in Fig. 3. The cluster size $N_c = 4$, and the *line-of-sight (LOS)* reuse distance is $D_{LOS} = 2$. This cell plan is denoted by SHS(2,0), in agreement with the notation in [14].

The second cell plan which we consider is an *Asymmetric Half Square (AHS)* cell plan, depicted in Fig. 4. The cluster size $N_c = 3$, and the *line-of-sight (LOS)* reuse distance is $D_{LOS} = 3$. This cell plan is denoted by AHS(1,1), in agreement with the notation in [14].

In both cell plans, one base station is placed in every street corner at lamp-post level. Base stations use omnidirectional antennas and the cell size is assumed to be half a block in all four directions. In the simulation, mobiles locations are independently sampled from a uniform distribution over each cell area, with 36 (6×6) cochannel cells. As in the hexagonal cell plan, the mobiles are connected to the closest base station. From our numerical results, the outage probability curve as a function of the target CIR (which is used as a comparison criterion) for the SHS(2,0) case, is a shift of the curve for the AHS(1,1) case. This can be explained by the fact that the distance between two LOS interferes is smaller in the SHS(2,0) case, which results in a larger interference level. Therefore, we present most of the results only for the AHS(1,1) case.

Method of comparison

The prime criterion by which we compare the removal algorithms is their resulting *outage probability*, P_{outage}, defined as

$$P_{outage} = P(\gamma \leqslant \gamma^t)\,,$$

where γ is the CIR of a random communication link between a mobile and its corresponding base station. The outage probability measures the expected proportion of mobiles which have to be removed by a removal algorithm. The secondary criterion is the time required to converge to a steady state with all mobiles being supported at their target CIR.

To evaluate the outage probabilities, and the time required to remove all non-supported mobiles, we carry the following simulation. Under every removal algorithm we take 1000 independent instances of mobile locations, and shadow fading. For each instance, we remove mobiles according to the algorithm rule. Mobiles are removed until all the remaining mobiles can be supported at their target CIR. The outage probability is estimated by counting the proportion of mobiles which are being removed over the 1000 realizations.

Note that after sampling a realization, mobiles are "frozen" at their position. This is a reasonable model when power control and removal decision are relatively much faster than mobile movements. In addition, taking 1000 independent realizations results in small estimation errors. We confine our examples to the uplink case, and stop the power control update when the maximum relative error between two consecutive power vectors becomes smaller than 10^{-8}.

Results

When comparing between algorithms we have to bear in mind the amount of information and coordination required by each algorithm. We have algorithms which use global information, and algorithms which use only local information. The latter, are easier to implement distributively, and certainly faster. The local-information-based algorithms are the following. The *Distributed GRR-DCPC* defined above; the *Simple DCPC* (where the entire non-supported set under the stationary power vector is removed); the *Random Removal DCPC* (where a single random transmitter is removed from the non-supported set under the stationary power vector); and the *Fixed Transmission Power* (where any non-supported transmitter is immediately removed). The global-information-based algorithms are all the rest.

In Figs. 5–8, we compare the performance of the various one-by-one removal algorithms. The results for the micro-cellular cell plan AHS(1,1) are presented in Figs. 5, 6, and for the hexagonal cell plan in Figs. 7, 8. Fig. 6 and Fig. 8 are drawn in a larger scale covering a smaller target CIR interval, of Figs. 5 and 7, respectively.

For the micro-cellular system we observe that PRRA-3(1) is the best one-by-one removal algorithm.

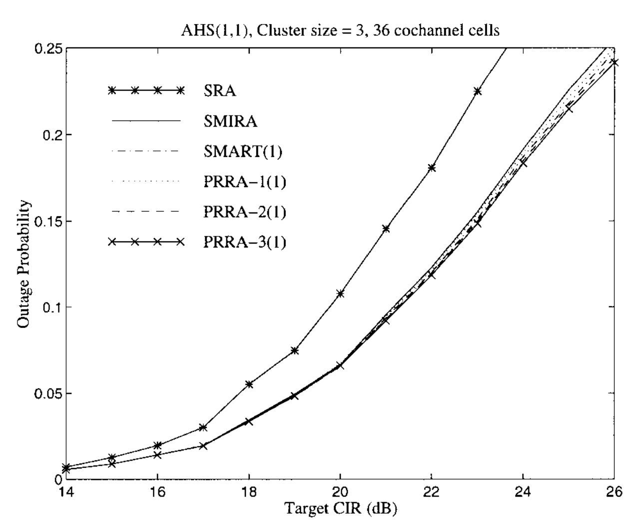


Fig. 5. Outage probabilities for one-by-one removal algorithms.

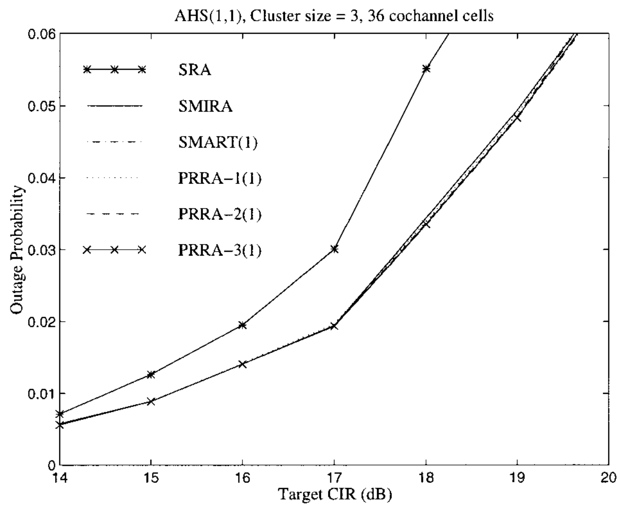


Fig. 6. Outage probabilities for one-by-one removal algorithms magnified for CIR target of 14–20 dB.

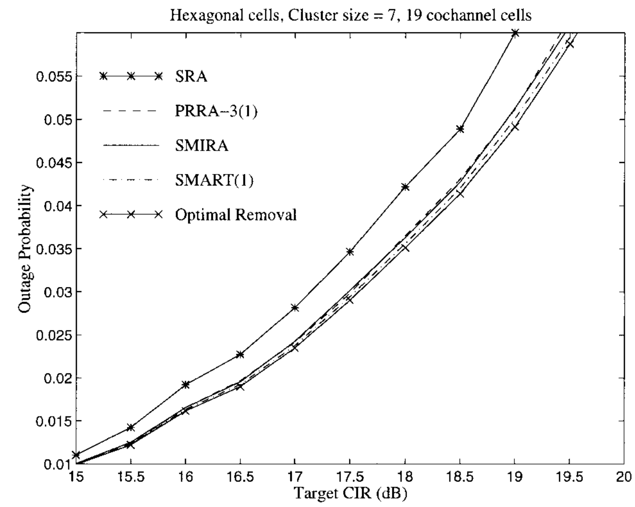


Fig. 8. Outage probabilities for one-by-one and optimal removal algorithms magnified for CIR target of 15–20 dB.

The second best is PRRA-2(1), and then SMART(1). Following is PRRA-1(1), SMIRA and SRA. As expected, the outage probability of PRRA-K(1) improves with the number of look-ahead steps K. Except for SRA, all algorithms perform almost equally and have the same computational complexity. Thus, PRRA-3(1) is the preferred choice. It should be noted that all figures present outage probabilities, and the best algorithm in this sense is not necessarily the best realization-wise.

For the macro-cellular hexagonal system we observe that SMART(1) is the best one-by-one removal algorithm. The second best is SMIRA, and then follow the PRRA-3(1) and SRA. In this case we have also computed the outage probability under the optimal removal algorithm, which has been obtained by an exhaustive search in each realization. Notice that SMART(1) closes more than 50% of the remaining gap left between SMIRA and the optimal outage probabilities. Nevertheless, they are quite similar in absolute values. Again, as their computational complexity is the same, SMART(1) is the preferred choice.

By their nature, one-by-one removal algorithms are slower in converging to a state where all mobiles are supported. This is evident in our next comparison where the best one-by-one, and multiple removal algorithms in an AHS micro-cellular system are compared. The outage probabilities are shown in Fig. 9, and the number of removal steps are depicted in Fig. 10. The number of removal steps are counted as follows. A mobile is sampled in every cell, the DCPC power control is applied until convergence, and then one or more mobiles are removed (depending on the algorithm). This is the first removal step. After this, the DCPC power control is re-invoked until convergence, and the second removal step is taken. This process continues until all the remaining mobiles are supported.

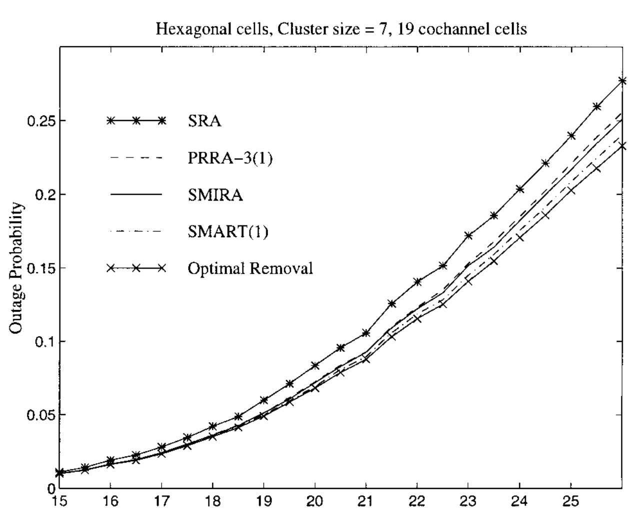


Fig. 7. Outage probabilities for one-by-one and optimal removal algorithms.

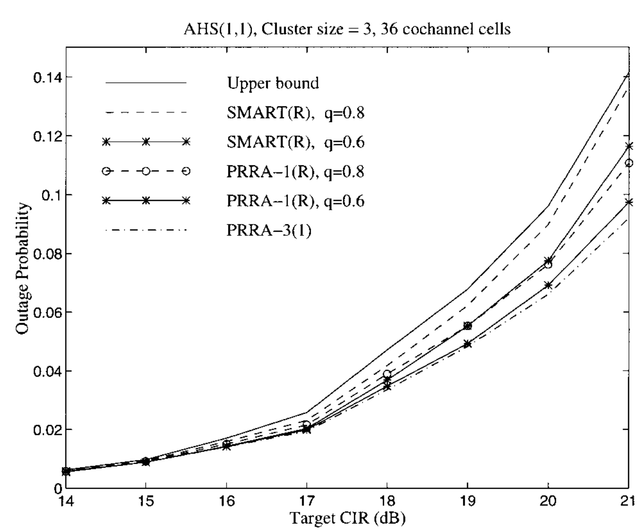


Fig. 9. Outage probabilities for multiple removal algorithms.

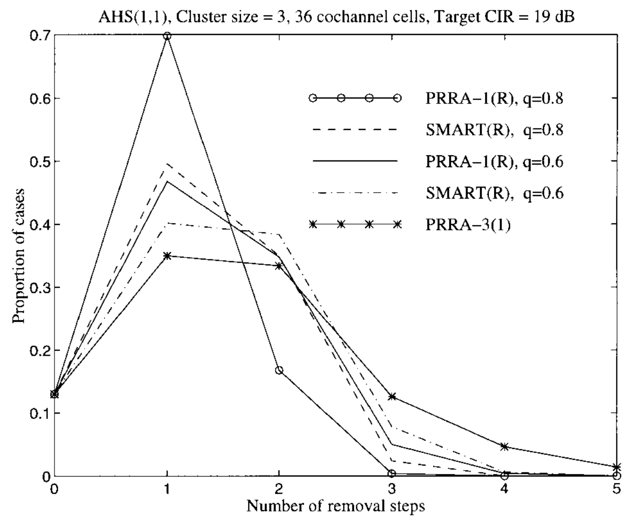


Fig. 10. Distribution of number of removal steps.

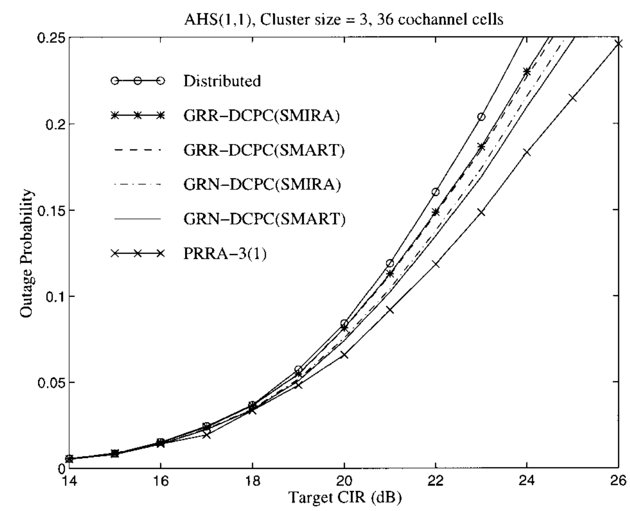


Fig. 11. Outage probabilities for gradual removal algorithms.

In the multiple removal method we determine the number of removals R for every realization, by first calculating the upper bound R^*. Then, we set $R = \lfloor qR^* \rfloor$, where q is either 0.6 or 0.8. In this comparison, we also include the computation of the upper bound. This algorithm is denoted in the figures by *Upper Bound*.

Observe that the best one-by-one removal algorithm outperforms all the others in terms of outage probability. Further, it pays off to cautiously remove mobiles, although it is slower, as seen from Fig. 10. Another observation is that PRRA-1(R) outperforms SMART(R) with respect to both criteria, the outage probability and the number of steps. Although this is not shown in our graphs, we found out that the $O(N^2)$ approximation to SMART(R) in section 4.2, yields almost identical results as SMART(R).

From Figs. 9 and 10 we learn that, there is a trade-off between the number of removal steps and the outage probability. Surprisingly in our mind, is that the GRN-DCPC(SMART) removal algorithm, has a small number of steps and a small outage. Moreover, its outage probability does not differ very much from the best one-by-one removal algorithm.

In Fig. 11, the outage probabilities of several GRX-DCPC algorithms are compared to the best one-by-one algorithm, PRRA-3(1), in a micro-cellular AHS cell plan. We observe that the best results are obtained by using a non-restricted removal set, and the SMART arbitration rule, GRN-DCPC(SMART). The second best is the non-restricted removal with the SMIRA arbitration rule, GRN-DCPC(SMIRA), followed by the restricted GRR-DCPC(SMART), GRR-DCPC (SMIRA), and then the distributed GRR-DCPC. As noted above, the outage probability of GRN-DCPC(SMART) is not significantly different from that of PRRA-3(1).

Finally, in Figs. 12–14 we compare among the best removal algorithms from each class which use global information, and all the algorithms which use local information. Fig. 12 presents the outage probabilities in the hexagonal macro-cellular system, Fig. 13 in the SHS micro-cellular system, and Fig. 14 in the AHS micro-cellular system.

We observe that for every cell plan, the *Distributed GRR-DCPC* is significantly better than any other local information based algorithm. Furthermore, its outage in comparison to the globally best algorithm is reasonably close. At the 10% outage probability level, a CIR gain of about 1.3 dB can be achieved by the Distributed GRR-DCPC compared to the Simple DCPC. At the 20% outage probability level, a CIR gain of about $2dB$ can be achieved.

Not surprisingly, algorithms which use global information achieve lower outage probabilities. Note that in the hexagonal macro-cellular system (Fig. 12), the outage probability curves of GRN-DCPC(SMART),

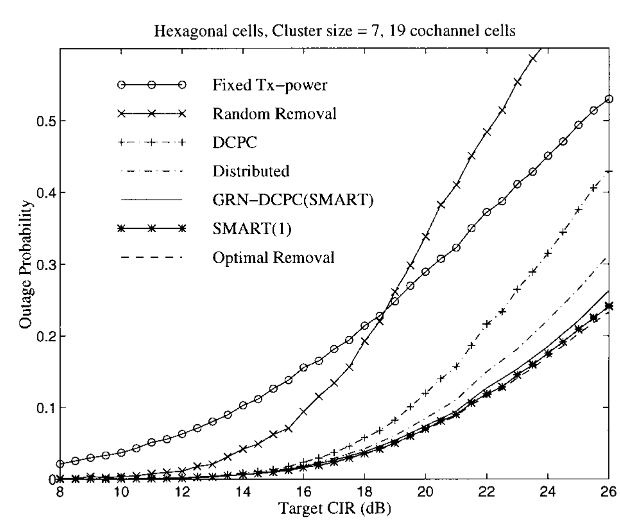


Fig. 12. Outage probability comparison between all removal algorithms.

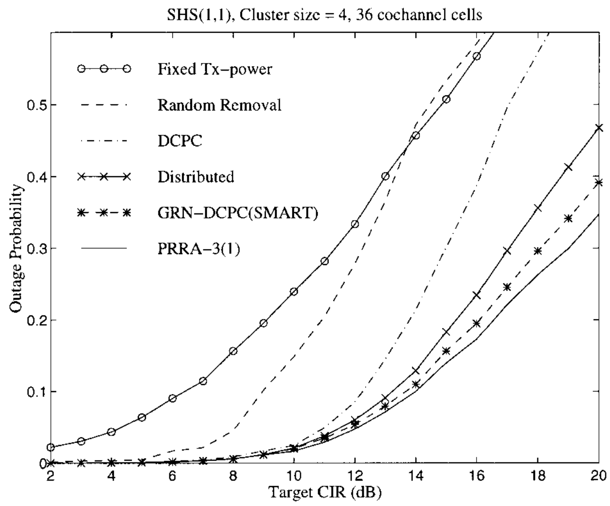


Fig. 13. Outage probability comparison between all removal algorithms.

SMART(1), and the optimal removal algorithms, are extremely close to each other. For the micro-cellular systems (Figs. 13, 14), the best one-by-one algorithm, PRRA-3(1) is still the global best, and the GRN-DCPC(SMART) is very close. In all cases, the one-by-one and Gradual Removal algorithms outperform the reference algorithms for all target CIRs. At the 10% outage probability level, a CIR gain of about 8 dB can be achieved compared to the fixed transmitter power scheme, and about 2 dB compared to the DCPC algorithm.

6. Conclusions

The problem of mobile removals in cellular networks is shown to be NP-complete. Properties of a broad set of removal algorithms which use local and global information are proven, and snapshot simulations are used to compare their performance. The following is a summary of the main comparison results.

- The *Distributed GRR-DCPC* removal algorithm is significantly better than any other local-information-based algorithm, and its outage probability is reasonably close to that of the globally best algorithm.
- Among the proposed removal algorithms, SMART(1) and PRRA-3(1) have the lowest outage probability. Although in absolute values, the improvement over SMIRA is marginal, in relative values, with respect to the optimum, the improvement is substantial. As all these algorithms have the same computational complexity, SMART(1) and PRRA-3(1) should be the preferred choices with respect to outage probability.
- With respect to outage probability, it is preferred to remove mobiles in small amounts at a time, rather than in large amounts. With respect to convergence time to a steady state, the reverse holds.
- GRN-DCPC(SMART) obtains outage probabilities which are very close to that of SMART(1) and PRRA-3(1). In the hexagonal macro-cellular system, it is almost indistinguishable from the optimum. With respect to convergence time to a steady state, it is clearly better.

In future PCS environments, user mobility and fast changing propagation conditions will have a greater impact on the system performance. Dynamic and bold channel allocation may efficiently utilize the scarce bandwidth resource. In such systems, distributed and efficient removal algorithms will become of utmost importance. The case studies above suggest that the *Distributed GRR-DCPC* is an excellent candidate for this task.

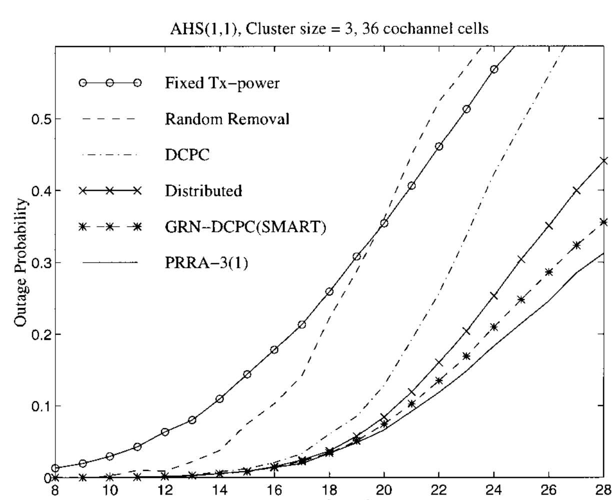


Fig. 14. Outage probability comparison between all removal algorithms.

Appendix

Proof of Proposition 1

We show that this problem is at least as hard as the problem of solving the following *Maximum Induced Subgraph with an Hereditary and Non-trivial Property* Π.

Let $\boldsymbol{A}$ be a given transformed gain matrix, and $\boldsymbol{p} \geqslant \gamma^t \eta$, be any feasible power vector. Define a complete directed graph $G = (V, E)$ as follows.
Set $V = \mathcal{N}$, and $E = \{(j \to i) : \forall i \neq j \in V\}$. To define property Π, we associate to each vertex and edge the following positive weights.
For every $i \in V$ set $w(i) = p_i - \gamma^t \cdot \eta_i$, and for every $(j \to i) \in E$ set $w(j \to i) = \gamma^t a_{ij} p_j$.
We say that a subgraph $G' = (V', E')$ *satisfies property* Π, *if and only if* $w(i) \geqslant \sum_{j \in V'} w(j \to i), \forall i \in V'$.

Note that property Π is nothing but the property that all transmitters are being supported under power vector p. Therefore, property Π is hereditary (i.e., every subgraph of G' satisfies Π whenever G' satisfies Π), and non-trivial (i.e., it is satisfied for infinitely many graphs and false for infinitely many graphs).

Now, finding the maximum number of supported transmitters under power vector $\boldsymbol{p}$, is the same as finding the maximum induced subgraph with property Π. This is known to be NP-complete, [9, GT21]. Note that although the problem is re-stated for complete graphs, it is not less general than an arbitrary graph, as the weights may be arbitrarily small. □

Proof of Lemma 2

Use the vector $\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}$ as the initial powers for the DCPC algorithm with the set $\mathcal{N}_0$. We first prove that

$$\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0} \geqslant \min\{\mathrm{p}, \gamma^t(\boldsymbol{A}^{\mathcal{N}_0}\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0} + \boldsymbol{\eta})\}\,. \tag{11}$$

Consider two cases:
Case (i) for $i \in \mathcal{N}_0 \cap \overline{S}_{\mathcal{N}}$, i.e., transmitters which are not supported under $\boldsymbol{p}^{\mathcal{N}}$.
Case (ii) for $i \in \mathcal{N}_0 \cap S_{\mathcal{N}}$, i.e., supported transmitters.

For every i in case (i) it follows from (5) that, $p_i^{\mathcal{N}/\mathcal{N}_0} = \overline{p}$. Thus, (11) trivially holds for component i.

For every i in case (ii) it follows from (8), and from the fact that the a_{ij}'s are non-negative, that

$$\begin{aligned} p_i^{\mathcal{N}/\mathcal{N}_0} &= p_i^{\mathcal{N}} = \gamma^t(\eta_i + \sum_{j\in\mathcal{N}} a_{ij}p_j^{\mathcal{N}}) \\ &\geqslant \min\{\overline{p}, \gamma^t(\eta_i + \sum_{j\in\mathcal{N}_0} a_{ij}^{\mathcal{N}_0} p_j^{\mathcal{N}/\mathcal{N}_0}\}\,. \end{aligned}$$

Thus, inequality (11) holds for all $i \in \mathcal{N}_0$.

From [12, Lemmas 6 and 9], it follows that using DCPC with an initial power vector $\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}$ satisfying (11), results in a non-increasing powers sequence which converges to $\boldsymbol{p}^{\mathcal{N}_0}$. (This is also a well known result for super-harmonic functions with respect to any operator.) This completes the proof. □

Proof of Corollary 3

First note that the complement set of $(S_{\mathcal{N}} \cap \mathcal{N}_0)$ is $(\overline{S}_{\mathcal{N}} \cup \overline{\mathcal{N}_0})$. Thus, it is sufficient to show that if $i \in \overline{S}_{\mathcal{N}_0}$ then $i \in \overline{S}_{\mathcal{N}}$.

Let $i \in \mathcal{N}_0$ be in the non-supported set $\overline{S}_{\mathcal{N}_0}$. Thus, from (5) and Lemma 2,

$$p_i^{\mathcal{N}} \leqslant \overline{p} = p_i^{\mathcal{N}_0} < \gamma^t(\eta_i + \sum_{j\in\mathcal{N}_0} a_{ij}p_j^{\mathcal{N}_0}) \leqslant \gamma^t(\eta_i + \sum_{j\in\mathcal{N}} a_{ij}p_j^{\mathcal{N}})\,.$$

Thus i, is also in the non-supported set $\overline{S}_{\mathcal{N}}$, which completes the proof. □

Proof of Theorem 4

We start with the first inequality. From (7) and (9) it follows that

$$\begin{aligned} D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) &= \sum_{j\in\mathcal{N}} \beta_j(\boldsymbol{p}^{\mathcal{N}}) \geqslant \sum_{j\in\mathcal{N}_0} \beta_j(\boldsymbol{p}^{\mathcal{N}}) \geqslant \sum_{j\in\mathcal{N}_0} \beta_j(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}) \\ &= D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0})\,. \end{aligned}$$

The last inequality above, results from (8) and the fact that the a_{ij}'s are non-negative.

Next, we prove the second inequality. From (3) and (9) it follows that $\beta_j(\boldsymbol{p}^{\mathcal{N}}) = 0$ for $j \in S_{\mathcal{N}}$, and $\beta_j(\boldsymbol{p}^{\mathcal{N}_0}) = 0$ for $j \in S_{\mathcal{N}_0}$. Thus,

$$D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) - D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}_0}) = \sum_{j\in\overline{S}_{\mathcal{N}}} \beta_j(\boldsymbol{p}^{\mathcal{N}}) - \sum_{j\in\overline{S}_{\mathcal{N}_0}} \beta_j(\boldsymbol{p}^{\mathcal{N}_0})\,. \tag{12}$$

From Corollary 3 we have $\overline{S}_{\mathcal{N}} \supseteq \overline{S}_{\mathcal{N}_0}$. Furthermore, from (9), $\beta_j(\boldsymbol{p}^{\mathcal{N}}) > 0$ for $j \in \overline{S}_{\mathcal{N}}$. Thus, eq. (12) implies

$$D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) - D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}_0}) \geqslant \sum_{j\in\overline{S}_{\mathcal{N}_0}} [\beta_j(\boldsymbol{p}^{\mathcal{N}}) - \beta_j(\boldsymbol{p}^{\mathcal{N}_0})]\,. \tag{13}$$

Since every $j \in \overline{S}_{\mathcal{N}_0}$ is not supported neither with $\mathcal{N}$ nor with $\mathcal{N}_0$, it follows from (5) that their stationary powers in both systems equal $\overline{p}$. Thus, (13) implies

$$\begin{aligned} &D^{\mathcal{N}}(\boldsymbol{p}^{\mathcal{N}}) - D^{\mathcal{N}_0}(\boldsymbol{p}^{\mathcal{N}_0}) \\ &\quad \geqslant \sum_{j\in\overline{S}_{\mathcal{N}_0}} \gamma^t[\sum_{i\in\mathcal{N}} a_{ji}p_i^{\mathcal{N}} - \sum_{i\in\mathcal{N}_0} a_{ji}p_i^{\mathcal{N}_0}] \geqslant 0\,. \end{aligned}$$

The last inequality follows from the fact that $a_{ij} \geqslant 0$ and Lemma 2. □

Proof of Theorem 5

Take $\boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}$ as the initial power vector for the DCPC algorithm for the set $\mathcal{N}_0$. That is, DCPC starts with $\boldsymbol{p}(0) = \boldsymbol{p}^{\mathcal{N}/\mathcal{N}_0}$. We show that under the Theorem condition, the power vector sequence converges to a stationary power vector under which all the transmitters in $\mathcal{N}_0$ are supported. That is,

$$\boldsymbol{p}^{\mathcal{N}_0} = \gamma^t(\boldsymbol{A}^{\mathcal{N}_0}\boldsymbol{p}^{\mathcal{N}_0} + \boldsymbol{\eta}). \tag{14}$$

From the Theorem condition and (11),

$$\boldsymbol{p}(0) \geqslant \min\{\mathrm{p}, \gamma^t(\boldsymbol{A}^{\mathcal{N}_0}\boldsymbol{p}(0) + \boldsymbol{\eta})\} = \gamma^t(\boldsymbol{A}^{\mathcal{N}_0}\boldsymbol{p}(0) + \boldsymbol{\eta})\,.$$

Thus, DCPC yields a subsequent power vector $\boldsymbol{p}(1)$, which satisfies

$$\boldsymbol{p}(1) = \gamma^t(\boldsymbol{A}^{\mathcal{N}_0}\boldsymbol{p}(0) + \boldsymbol{\eta}) \leqslant \boldsymbol{p}(0) \leqslant \overline{p}\,.$$

By induction on the DCPC update steps, it is easy to verify that for every k,

$$\boldsymbol{p}(k+1) = \gamma^t(A^{\mathcal{N}_0}\boldsymbol{p}(k) + \eta) \leqslant \boldsymbol{p}(k) \leqslant \overline{p}\,. \tag{15}$$

Thus, the sequence $\{\boldsymbol{p}(k)\}$ is non-decreasing, and converges to the stationary power vector $\boldsymbol{p}^{\mathcal{N}_0}$ satisfying equation (14). This concludes the proof. □

Proof of Theorem 6

Clearly, the number of removals are bounded by

$N-1$. Let R be the number of removals, and $\mathcal{R} = \{j_1, \ldots, j_R\}$ be the set of removed transmitters. Denote by $0 \leqslant k_1 \leqslant k_2 \leqslant \ldots \leqslant k_R$ the step numbers where the transmitters are removed by meta-step (2.2) of the Generic GRR-DCPC algorithm.

Let $\{\tilde{\boldsymbol{p}}(k), k \geqslant 0\}$, be the power sequence generated by DCPC (as defined in (2)), given that $\tilde{\boldsymbol{p}}(0) = \boldsymbol{p}(0) \in \mathcal{C}^-(\gamma^t)$. Also, denote by $\{\boldsymbol{p}(k), k \geqslant 0\}$, the power sequence generated by Generic GRR-DCPC algorithm, given that $\boldsymbol{p}(0) \in \mathcal{C}^-(\gamma^t)$. Note that

$$\tilde{\boldsymbol{p}}(0) \leqslant \min\{\overline{p}, \gamma^t(\boldsymbol{A}\tilde{\boldsymbol{p}}(0) + \boldsymbol{\eta})\} .$$

Thus, from [12, Lemmas 6 and 9], $\tilde{\boldsymbol{p}}(k), k \geqslant 0\}$, is a non-decreasing sequence which converges to the fixed-point stationary power vector $\boldsymbol{p}^{\mathcal{N}}$. That is,

$$\tilde{\boldsymbol{p}}(k) \leqslant \tilde{\boldsymbol{p}}(k+1) , \quad k \geqslant 0 . \tag{16}$$

To prove the Theorem, we will first show that:

(i) $\boldsymbol{p}(k) \leqslant \tilde{\boldsymbol{p}}(k)$, for every $k \geqslant 0$.

(ii) $\mathcal{R} \subseteq \overline{\mathcal{S}}_{\mathcal{N}}$.

(iii) For every $k \geqslant k_R$, the sequence $\{\boldsymbol{p}(k), k \geqslant 0\}$, evolves exactly as the sequence $\{\tilde{\boldsymbol{p}}(k), k \geqslant 0\}$, under DCPC for the set of transmitters $\mathcal{N} \setminus \mathcal{R}$.

Assertions (i) and (ii) will be proven by induction on the removal instances. For $0 \leqslant k \leqslant k_1$ (before the first meta-step in (2.2) of the Generic GRR-DCPC algorithm), we have $\boldsymbol{p}(k) = \tilde{\boldsymbol{p}}(k)$. By the definition of step (2) in the Generic GRR-DCPC,

$$\begin{aligned} \tilde{p}_{j_1}(k_1) = p_{j_1}(k_1) = \overline{p} &< \gamma^t(\eta_{j_1} + \sum_{j \in \mathcal{N}} a_{j_1,j} p_j(k_1 - 1)) \\ &= \gamma^t(\eta_{j_1} + \sum_{j \in \mathcal{N}} a_{j_1,j} \tilde{p}_j(k_1 - 1)) . \end{aligned} \tag{17}$$

Therefore, eq. (16) implies that

$$\tilde{p}_{j_1}^{\mathcal{N}} = \overline{p} < \gamma^t(\eta_{j_1} + \sum_{j \in \mathcal{N}} a_{j_1,j} \tilde{p}_j^{\mathcal{N}}) . \tag{18}$$

Hence, $j_1 \in \overline{\mathcal{S}}_{\mathcal{N}}$. Furthermore, after the meta-step in (2.2) of the generic GRR-DCPC algorithm, we have

$$\boldsymbol{p}(k_1) \leqslant \tilde{\boldsymbol{p}}(k_1) . \tag{19}$$

Assume by induction that, $\boldsymbol{p}(k) \leqslant \tilde{\boldsymbol{p}}(k), \forall 0 \leqslant k \leqslant k_{m-1}$, and $\{j_1, \ldots, j_{m-1}\} \in \overline{\mathcal{S}}_{\mathcal{N}}$.

For every $k_{m-1} < k \leqslant k_m$ (before the m^{th} meta-step in (2.2) of the Generic GRR-DCPC algorithm), $p_i(k) = 0 \leqslant \tilde{p}_i(k)$, $\forall i \in \{j_1, \ldots, j_{m-1}\}$. For $i \notin \{j_1, \ldots, j_{m-1}\}$, we have

$$\begin{aligned} p_i(k_{m-1} + 1) &= \min\{\overline{p}, \gamma^t(\eta_i + \sum_{j \in \mathcal{N}} a_{ij} p_j(k_{m-1}))\} \\ &\leqslant \gamma^t(\eta_i + \sum_{j \in \mathcal{N}} a_{ij} \tilde{p}_j(k_{m-1})) = \tilde{p}_i(k_{m-1} + 1). \end{aligned} \tag{20}$$

Thus, $\boldsymbol{p}(k_{m-1} + 1) \leqslant \tilde{\boldsymbol{p}}(k_{m-1} + 1)$. By a straightforward induction argument, it also follows that, $\boldsymbol{p}(k) \leqslant \tilde{\boldsymbol{p}}(k)$, $\forall k_{m-1} \leqslant k \leqslant k_m$. (If $k_m = k_{m-1} + 1$, the above argument is not required.) Repeating the arguments in (17)–(19) for the mth removal, we obtain the proof of the induction step. Thus, $\boldsymbol{p}(k) \leqslant \tilde{\boldsymbol{p}}(k)$, $\forall\, 0 \leqslant k \leqslant k_R$, and $\mathcal{R} \subseteq \overline{\mathcal{S}}_{\mathcal{N}}$. The latter implies part (b) of the Theorem. To show the convergence properties, note that for $k > k_R$, the power sequence of the transmitters $i \in \mathcal{N} \setminus \mathcal{R}$, under the Generic GRR-DCPC, evolves exactly as under DCPC operating with the set of transmitters $\mathcal{N} \setminus \mathcal{R}$. Thus, having the same convergence properties as DCPC has, under synchronous and asynchronous power updates (see [12]).

Part (a) of the theorem follows from the fact that there are at most $N-1$ removals, and therefore all removals occur after a finite number of steps. This completes the proof. □

References

[1] J.M. Aein, Power balancing in systems employing frequency reuse, Comsat Tech. Rev. 3 (2) (1973).

[2] M. Almgren, H. Andersson and K. Wallstedt, Power control in a cellular system, *Proc. IEEE Veh. Tech. Conf., VTC-94* (1994) pp. 833–837.

[3] M. Andersin, Z. Rosberg and J. Zander, A soft admission in cellular PCS with constrained power control and noise, *Proc. 5th WINLAB Workshop* (1995).

[4] N. Bambos and G.J. Pottie, On power control in high capacity cellular radio networks, *Proc. 3rd WINLAB Workshop* (1992).

[5] N. Bambos, S.C. Chen and G.J. Pottie, Channel access algorithms with active link protection for wireless communication networks with power control, Technical Report, UCLA-ENG-95-114, Department of Electrical Engineering, UCLA (1995).

[6] J.-E. Berg, A simplified street width dependent microcell path loss model, COST 231, TD(94)035 (1994).

[7] J.-E. Berg, R. Bownds and F. Lotse, Path loss and fading models for microcells at 900 MHz, *Proc. IEEE Veh. Tech. Conf., VTC-92* (1994) pp. 666–671.

[8] G.J. Foschini, A simple distributed autonomous power control algorithm and its convergence, IEEE Trans. Veh. Tech. 42(4) (1993).

[9] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-completeness* (W.H. Freeman, San Francisco, 1979).

[10] S.A. Grandhi, R. Vijayan and D.J. Goodman, A distributed algorithm for power control in cellular radio systems, *13th Annual Allerton Conf. on Commun., Contr. and Computing*, Monticello, Illinois (1992).

[11] S.A. Grandhi, R. Vijayan, D.J. Goodman and J. Zander, Centralized power control in cellular radio systems, IEEE Trans. Veh. Tech. 42(4) (1993).

[12] S.A. Grandhi and J. Zander, Constrained power control in cellular radio system, *Proc. IEEE Veh. Tech. Conf., VTC-94* (1994) pp. 824–828.

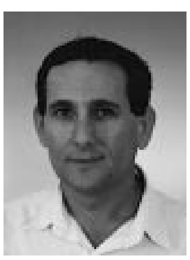
[13] S.A. Grandhi, J. Zander and R. Yates, Constrained power control, Wireless Personal Commun. 1(4) (1995).

[14] M. Gudmundson, Cell planning in Manhattan environments, *Proc. IEEE Veh. Tech. Conf., VTC-92* (1992) pp. 435–438.

[15] S.V. Hanly, An algorithm for combined cell-site selection and power control to maximize cellular spread spectrum capacity, IEEE J. Select. Areas Commun., special issue on Fundamentals of Networking, to appear.

[16] J.C. Lin, T.H. Lee and Y.T. Su, Power control algorithm for cellular radio systems, IEEE Trans. Veh. Tech. 44(1) (1995).
[17] H.J. Meyerhoff, Methods for computing the optimum power balance in multibeam satellite, Comsat Tech. Rev. 4(1) (1974).
[18] D. Mitra, An asynchronous distributed algorithm for power control in cellular radio systems, *Proc. 4th WINLAB Workshop* (1993).
[19] R.W. Nettleton and H. Alavi, Power control for spread-spectrum cellular mobile radio system, *Proc. IEEE Veh. Tech. Conf., VTC-83* (1983) pp. 242–246.
[20] R. Yates and C.Y. Huang, Optimal uplink control and base station assignment, *Proc. 3rd ICUPC*, San Diego, CA (1994) pp. 247–251.
[21] J. Zander, Performance of optimum transmitter power control in cellular radio systems, IEEE Trans. Veh. Tech. 41(1) (1992).
[22] J. Zander, Distributed cochannel interference control in cellular radio systems, IEEE Trans. Veh. Tech. 41(3) (1992).

Michael Andersin received the M.S. degree in physics engineering (applied mathematics) from the Royal Institute of Technology, Stockholm, Sweden, in 1992. Since 1992, he has been a Ph.D. student at the Radio Communication Systems, Royal Institute of Technology. During spring 1995, he was with the Wireless Information Network Laboratory (WINLAB), Rutgers University, USA. His current research interests include radio resource management issues in cellular and personal communication systems.
E-mail: mike@radio.kth.se



Zvi Rosberg received the B.Sc., M.A. and Ph.D. degrees from the Hebrew University of Jerusalem. From 1972 to 1978 he was a senior system analyst in the Central Computing Bureau of the Israeli Government, where he was one of the chief designers of a new on-line Israeli population registration system. From 1978 to 1979 he held a research fellowship at the Center of Operation Research and Econometric (C.O.R.E.), Catholic University of Louvain, Belgium. From 1979 to 1980 he was a visiting assistant professor at the Department of Business Administration, University of Illinois, and during 1980 to 1990, he was with the Department of Computer Science, Technion, Israel.

Since 1990 he is with the Haifa Research Laboratory, Science and Technology, IBM Israel, holding a position of a Program Manager of Communication Networks. From 1985 to 1987 he was a visiting scientist at IBM Thomas J. Watson Research Center, Yorktown Heights, USA. Since 1980 he held summer research positions at IBM Thomas J. Watson Research Center, Yorktown Heights, Department of Electrical and Computer Engineering, University of Massachusetts, Amherst, Department of Electrical Engineering and Computer Science, University of California, Berkeley, and the Radio Communication Systems, Royal Institute of Technology.

He has published numerous papers in the areas of communication networks, queueing theory, control of queueing systems, applied probability, analysis of algorithms in communication and computing systems, power control and resource allocation in cellular radio systems, and performance evaluation. He led a project which developed an IBM licensed product for network topology and routing design, for which he was awarded an IBM Research Division outstanding technical achievement.

His current interest and activity are in the area of resource allocation and power control in cellular radio systems, mobile computing, and high speed networking.
E-mail: rosberg@vnet.ibm.com

Jens Zander received the M.S degree in electrical engineering and the Ph.D Degree from Linköping University, Sweden, in 1979 and 1985, respectively. From 1985 to 1989 he was a partner of SECTRA, an R&D Consultant Firm. His work there was mainly concerned with aerospace and defence applications. In 1989 he was appointed as Professor and Head of the Radio Communication Systems Laboratory at the Royal Institute of Technology, Stockholm, Sweden. Since 1992 he also serves as Senior Scientific Advisor (associate research manager) to the Swedish National Defence Research Establishment(FOA) and on the Board of TERACOM, the Swedish National Broadcasting Operator. In addition, he serves as consultant to government agencies and industry on spectrum allocation techniques and policy.

Dr. Zander has published numerous papers in the field of radio communication, in particular on resource management aspects of personal communication systems. Dr. Zander was the recipient of the IEEE VTS "Jack Neubauer Award" for the best systems paper in 1992. He has co-authored two textbooks on digital radio communication systems, which are now the standard textbooks on this subject in Swedish universities.

Dr. Zander was recently co-chairman of the technical program committe of IEEE Vehicular Technology Conference in Stockholm, June 1994. He is an adjoint member of the Swedish URSI Committee, sections C and G/H and IEEE.

His current research interests include spectrum resource allocation issues in high capacity personal communication systems, advanced HF-communication systems and systems for digital wireless broadcasting.
E-mail: jensz@radio.kth.se