# Collaborative Transmission in Wireless Sensor Networks by a $(1+1)$-EA

Stephan Sigg, Michael Beigl
Distributed and Ubiquitous Systems
Technical University of Braunschweig
38106 Brunswick, Germany
{sigg,beigl}@ibr.cs.tu-bs.de

## Abstract

*With collaborative transmission we propose a novel transmission scheme that utilises constructive interference between transmitted signals of wireless sensor nodes. Similar to cooperative transmission approaches we are able to drastically extend the transmission range of a wireless sensor network. We show that synchronisation of received signal components is feasible without inter-node communication. Our approach is capable of synchronising a virtually arbitrary number of signal components. The underlying scenario is modelled as a black-box optimisation problem and is solved by a $(1+1)$ evolution strategy. The method is optimal in the sense that any other evolutionary search approach with equal mutation probability has an expected asymptotic optimisation time that is at least of the same order.*

## 1. Introduction

Networks constructed from wirelessly communicating nodes that are equipped with sensing hardware are able to measure environmental stimuli in hardly accessible areas. Possible use cases for these wireless sensor networks (WSNs) are, for example, the measurement of seismic activities in earthquake regions or the proactive detection of new or recently reignited fire sources in forest fire zones. Furthermore, the observation of ecosystems states an application scenario where sensor nets are advantageous due to their passiveness and the low installation cost, for example, by discarding the sensors from an aeroplane. The ecosystem is then only negligibly disturbed by the installation of the sensor net. Recent examples for this application case in the literature are the observation of bird-races or travel habits of fish swarms [37, 36].

Environmental information like audio, temperature, humidity or pressure is captured by sensors. It is, however, challenging to obtain information from a sensor network at a remote location since the sensors itself can hardly be accessed directly due to an inaccessible environment they are brought out in. The information can, however be sampled by receivers mounted on helicopters, aeroplanes or on other exposed objects as, for example, on mountains or tall buildings. Due to restrictions in hardware, a single sensor is typically not able to reach a destination that is reasonably far off. A typical transmission range of one single sensor node is about twenty meters.

Frequently, the use of specific nodes with an extended transmission range is therefore proposed [3]. Sensed information is routed to these nodes which then transmit the received data to a remote receiver.

The obvious drawback of this solution is the increased probability to loose connection to parts of the network. The reasons for this are that one of the transmission-range-enhanced nodes themself might fail, that they might be irregularly distributed among the network or that the connection of some nodes to one of these enhanced nodes gets lost. Furthermore, these special nodes increase the total cost for the installation of a sensor network.

Alternatively, sensor nodes might co-operate in transmitting sensed information. The sensor network basically constitutes an array of antennas so that in analogy to MISO or MIMO systems the constructive interference between the singular transmitted signals might be explored to increase the transmission range. Several approaches on co-operatively transmitting sensor nodes have been proposed [24, 26, 25] that utilise the constructive interference of transmitted signals to increase connectivity between several parts of a sensor network.

In these approaches, neighbouring nodes are utilised as relays that retransmit a received signal several times. Since signals are transmitted omnidirectional, this transmission scheme might actually lead to some constructive interference somewhere in the transmission range. The actual location where constructive interference occurs is, however, dependent on the location of the transmitting nodes and the phases of the distinct signals.

In order to achieve a reasonable transmission range, however, it is necessary to combine the signals of several hundred sending nodes. The exact time synchronisation between such a great number of sensor nodes is not possible with current network protocols.

We propose a black-box-optimisation approach to cooperative transmission that is guided by receiver feedback instead of direct communication between nodes. Due to the receiver feedback, the location where the maximum constructive interference is observed can be centred on the receiver location.

Solutions to providing receiver-based channel feedback are presented in [45, 28, 44]. The authors utilise long-term prediction at the transmitter in order to estimate the channel state evolution. While this approach might surcharge the node's capabilities, we have demonstrated in [39] that simple comparison to a predefined signal already provides all means for sufficient feedback. We apply a $(\mu + \lambda)$ evolutionary strategy [35] with $\mu = \lambda = 1$ that is implemented on a set of autonomously operating sensor nodes. The expected performance of this approach is derived analytically and is confirmed by a simulation in a Matlab-based simulation environment.

## 2 Related Work

A decent general overview on wireless sensor networks is provided in [27, 3, 46]. In the scope of this work we are interested in cooperative transmission in wireless sensor networks. This is accomplished in the literature by three distinct approaches: Multi-hop, Data flooding and cluster based.

Multi-hop relaying relies on the physical channel. The multi-hop scenario is interpreted as multi-dimensional relay channel, where communication between nodes is allowed [22]. It has been shown that this approach optimally divides the network resources in terms of information theoretic metrics [4]. With increasing scenario sizes, this approach is, however, not well suited since the number of transmitted bits per square meter decreases quadratically with the size of the network [33, 16].

An alternative approach that bases on flooding the network with a message that shall be transmitted is presented in [29, 40]. In the opportunistic large array (OLA) method neighbouring nodes function as relay nodes that retransmit a received signal various times. In this approach the network is flooded with nodes that transmit the desired signal whereby constructive interference is created. A related approach is presented in [26, 24, 23], where the signal is overlaid with white noise to increase the probability of constructive interference. However, for all these strategies the maximum constructive interference occurs at a random point in the transmission range since nodes are not synchronised and also no receiver feedback is utilised.

The third, cluster based approach was first proposed in [38]. The basic idea of this approach is to build up clusters of jointly transmitting nodes that cooperate when sending or receiving messages [4]. In [30] the optimal cluster design is derived. This approach has the benefit that standard routing algorithms as well as multi-hop theory can be applied with little modification. However, the capacity of a network that follows this topology is lower than for the previously detailed approaches [16, 12].

In contrast to these cooperative approaches we propose to utilise the receiver feedback in order to guide the synchronisation of transmitted signals. By doing this we are able to synchronise a virtually arbitrary number of received signals at any specific target location of a receiver without the need of direct communication between nodes and at a very fast pace. With respect to the fact that no direct communication between nodes is required we refer to this approach as collaborative transmission. In the following sections we demonstrate that it is possible with collaborative transmission to increase the strength of a received signal by a factor that is easily larger than 100 in the fraction of a second.

In order to synchronise transmitted signals, sensors might adapt their baseband frequency either voltage controlled (VCO-elements) or by even simpler L/RC-transmitter types. Both transmitter types enable the phase shift of a signal by instantaneous alteration of the baseband frequency. The latter L/RC-based transmitters are highly error prone so that frequency as well as phase are subject to errors. Since the more exact VCO-elements are, however, way more expensive and are not likely to decrease in cost due to manufacturing conditions, we believe that L/RC-transmitters are the only feasible approach to implementing sensor nodes that collaboratively transmit their data. Most of the sensor network nodes available use simple COTS type of transceivers, among them the popular MOTES type of sensors or modern IEEE 802.15.4 based sensor nodes. These type of devices are less suitable for use in the proposed approach. Other sensor node types provide more interfaces to internal functionality, among them Particle Computers $\pi$-Part type of devices and the first generation of MOTE/MICA nodes.

Evolutionary algorithms (EAs) operate on a search space $S$ with search points that are weighted by a fitness function $f_{\text{fitness}} \rightarrow \mathbb{R}$. The aim of the algorithm is to compute a point $c$ in the search space with preferably high (or low depending on the fitness function and the optimisation aim) fitness value $f_{\text{fitness}}(c)$. The distinguishing factor between evolutionary algorithms is the optimisation time the algorithm takes in order to compute a reasonable fitness value. This is typically determined by a stop-criteria as, for example, a maximum number of iterations or a threshold fitness value that suffices for the optimisation. Evolutionary algorithms provide no guaranty on the optimisation time or even if a sufficiently high rated point in the search space is actually computed in reasonable time.

An evolutionary algorithm holds a population of search points or individuals in the search space $S$. In every iteration or generation, the algorithm first rates all individuals in the population. Afterwards a set of individuals is chosen for mutation or crossover to create a new population. If the overall fitness value of newly created individuals is better than the fitness value of the existent individuals, the old ones are exchanged for the new individuals. We distinguish between $(\mu+\lambda)$ and $(\mu, \lambda)$ strategies depending on whether the $\mu$ best rated members out of both, the $\lambda$ members of the offspring population and the $\mu$ members of the current population, or only the $\mu$ best individuals out of the offspring population constitute the next population.

Evolutionary algorithms have been developed in the 1960s. We can distinguish between four general classes of algorithms that have been subsequently merged together in the pace of research.

John Holland invented the genetic algorithms [17] that typically utilise the search space $\mathbb{B}^n$ and $k$-point crossover in most cases. Hans-Paul Schwefel and Ingo Rechenberg on the other hand developed the so-called evolution strategies [35] that at first mostly operated in the search space $\mathbb{R}^n$ and utilised the mutation operator. Furthermore, Lawrence Fogel developed the evolutionary programming [11] that takes the space of all finite automata as search space and utilises the mutation operator at most. Finally, John Koza developed the genetic programming [21] that utilises a set of Graphs that follow logic formulae or programs as search space.

During the last years, evolutionary algorithms have also been considered by theoretical computer scientists [2, 5, 6, 19]. In the scope of this work especially findings related to $(1 + 1)$-strategies are of major interest for our studies [43, 42, 7, 8, 9]. Typically, complexity theoretic issues related to the $(1 + 1)$-EA on various binary search spaces and on typical classes of functions are studied [13, 41, 34, 20].

Further work considers the optimal mutation probability [1], the choice of the offspring population size [18] or general boundaries for black-box optimisation [10].

## 3 Problem statement

In this section we detail the underlying optimisation problem of finding an optimal phase shift for a great number of transmitted signals from a set of sensor nodes without allowing communication between the nodes. The setting is constituted by a set of sensor nodes that are arbitrarily distributed. The nodes aim to transmit their sensed information to a remote receiver. The distance between the nodes and the receiver exceeds the maximum transmission range of every single sensor.

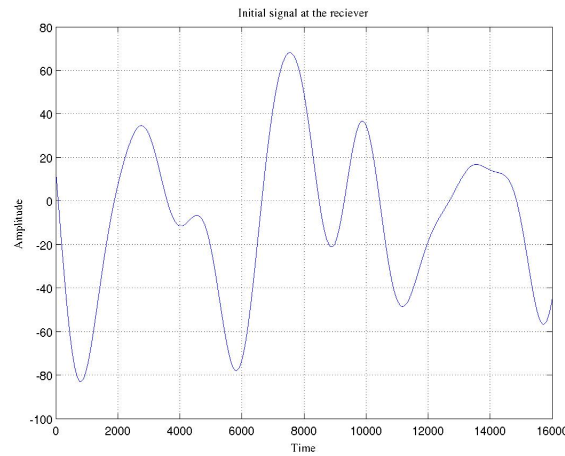


**Figure 1. Illustration of a received superimposed signal from $n = 1000$ signal components.**

At the receiver, the sum signal of all distinct signal components is received but initially distorted due to constructive and destructive interference between the not synchronised signal components. Figure 1 depicts the superimposed sum signals of $n = 1000$ nodes. All signal components have an amplitude of 1 and phase shifts that are drawn uniformly at random from $[-\pi, \pi]$ while frequency is Gaussian distributed with mean at 2.4 GHz and standard derivation $\sigma$ of 10 MHz.

In order to synchronise signals at the receiver, we propose the use of channel feedback as, for example, detailed in [45, 28, 44] to guide this synchronisation. The received sum-signal is analysed at the receiver and a feedback describing the state of the optimisation is created and transmitted back to the sensor nodes. On receiving the feedback, sensor nodes autonomously decide if their transmission signal shall be adopted regarding the phase shift applied. This step constitutes the mutation of the evolutionary strategy. Since we utilise a random search mechanism, the decision, if and by which amount a phase modulation shall be applied to the transmission signal is guided by a random variable. When the feedback function indicates a decrease in signal quality, all nodes that recently changed the phase of their transmission signal reverse their decision. Otherwise the change in phase is retained. The following sections detail aspects of our approach.

### 3.1 Voting on a transmission sequence

In order for collaborative transmission to succeed, all nodes have to agree on the sequence to transmit. We assume

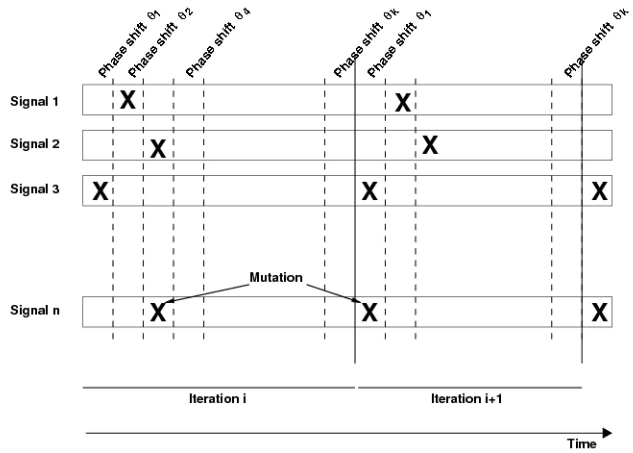


**Figure 2. Illustration of the basic scenario. From iteration $i$ to iteration $i + 1$ a mutation is applied to sensor node $n$.**

that the nodes did agree on such a sequence in advance. For the negotiation of a transmit sequence between nodes, in [24] an energy and time efficient protocol for sensor networks is described, but also other protocols are feasible.

### 3.2 Search space

Since we apply an optimisation algorithm on a search problem the search space $S$ has to be defined. One search point in $S$ corresponds to a unique configuration for the transmit signals from all sensor nodes in the network. Given $n$ nodes, each one is for practical reasons restricted to a finite number of possible phases that can be applied. Consequently, when $k$ describes the count of distinct phase modulations for each sensor node, the search space size is $|S| = k^n$. We model single searchpoints $c \in S$ as the set of phases $\Omega_1, \Omega_2, \cdots, \Omega_n$of all signals where $\Omega_i$ represents the phase of a signal $c_i$. W.l.o.g. we assume a maximum of $k$ distinct phases that can possibly be applied to a signal. Figure 2 depicts this basic scenario.

### 3.3 Optimisation aim and fitness function

The fitness function is guided by the degree by which the transmitting nodes are synchronised. In a realistic setting the fitness function is given by various factors as, for example, multipath fading and the SINR of the signal at the receiver. For our theoretical consideration we abstract from some of these details to keep the model analytically manageable. W.l.o.g. we assume that all nodes have to apply exactly the same phase shift in order to reach optimality. By permutating phases, however, any other configuration is achieved.

### 3.4 Technical considerations

Technically, our approach is similar to that of a phase lock loop (PLL). This feedback loop would be open and shared among receiver and senders. In our case, the feedback provided should adjust the baseband phase of the transmitter. This can be implemented by tuning a VCO in frequency synthesis based transmitter circuits (MICA MOTES) or by changing the modulation parameters in amplifier sequenced hybrid type of transmitters (e.g. $\pi$-Part sensor nodes).

### 3.5 Design of the evolutionary algorithm

We apply a $(1 + 1)$-EA on this optimisation problem. That is, an evolutionary algorithm with population size one and offspring population size one. This solution is straightforward since one configuration of all transmission sequences corresponds to exactly one individual in the population. Without advanced configuration of sensor nodes, a population size greater than one and consequently crossover is not possible.

We apply a mutation probability of $\frac{1}{n}$ for each sensor node while the phase modulation applied is drawn uniformly distributed from the $k$ possible phases. In case the fitness value for the new population is lower than for the old population, the mutation is retained. Otherwise all sensors that changed their signal phases reverse this modification.

Note that no communication between nodes is required for this mechanism.

## 4 Results

In this section we first analytically discuss the expected optimisation time and compare the result to a random search approach. Afterwards, simulation results from our implementation are presented.

### 4.1 Analytical consideration

For the analytic consideration we first calculate the expected optimisation time for a random search method. We consider this method as upper bound for the runtime of the optimisation problem. Finally, we are able to also provide a lower bound on the expected optimisation time of an evolution strategy with mutation probability $\frac{1}{n}$. This lower bound is of the same asymptotic order as the expected runtime of the proposed $(1 + 1)$-EA For these considerations we assume $n$ nodes in the sensor net, $k$ distinct signal phases and a minimum of $m$ signals that are required to be received synchronously in order for the signal strength to suffice at a remote receiver.

Observe that the search problem has $k$ global maxima since a global maximum remains a global maximum when the signal phases of all transmitted signals are shifted simultaneously by the same value $\Omega_\Delta$. Since the transmitted signals are not synchronised for any configuration that differs in less than $n$ phases from an optimal configuration, no further global maximum exists.

We calculate the fitness of one individual as the number of different signal phases utilised in the received sum signal. When all received sequences are synchronous, the fitness is consequently 1 while it is $k$ when every phase is utilised by at least one signal component.

#### 4.1.1 Optimisation time for random search

The probability that all $n$ transmit signals are synchronously received by a receiver when the signal phases are chosen purely random from $k$ possible phases is

$$\frac{1}{k^n}. \tag{1}$$

The probability that from $n$ signals at minimum $m$ are synchronised when the phases are chosen purely random is

$$\frac{\sum_{i=1}^{n-m} \begin{pmatrix} n \\ i \end{pmatrix} \cdot k \cdot \begin{pmatrix} k-1 \\ i \end{pmatrix}}{k^n} \tag{2}$$

We obtain this result simply by counting all possibilities in the probability space as illustrated in figure 3.

The figure depicts the ordered set of all possible configurations. Observe that this is a fitness based partition of the search space. Each distinct transmit signal in the received sum signal that is not in phase with the magnitude of synchronous transmit signals constitutes an error. Since a maximum of $n-m$ signals that are not in phase are allowed, we can simply count all configurations with less than $m$ synchronous signals in order to obtain the probability that the desired configuration with at most $n-m$ not synchronous signals is found.

The expected optimisation time is therefore

$$\begin{aligned} E[X] = \quad & \sum_{i=1}^{\infty} i \cdot \left( \frac{\sum_{j=0}^{n-m} \begin{pmatrix} n \\ j \end{pmatrix} \cdot k \cdot \begin{pmatrix} k-1 \\ j \end{pmatrix}}{k^n} \right) \\ & \cdot \left( 1 - \left( \frac{\sum_{j=0}^{n-m} \begin{pmatrix} n \\ j \end{pmatrix} \cdot k \cdot \begin{pmatrix} k-1 \\ j \end{pmatrix}}{k^n} \right) \right)^{i-1} \end{aligned} \tag{3}$$

#### 4.1.2 Optimisation time of a (1+1)-EA

We calculate an upper bound for the optimisation time of the $(1+1)$-EA on this search problem. The fitness based partition depicted in figure 3 will again guide the analysis. We label the i-th ($= i$ errors) layer as $L_i$. The optimisation is guided by the fitness function which has the same value for all individuals in $L_i$ but differs for all individuals in $L_j$ with $i \neq j$. Whenever, due to a mutation, the fitness value is increased, the current layer $L_i$ is left by the optimisation algorithm. In layer $n-i$, one of

$$\begin{pmatrix} n-i \\ 1 \end{pmatrix} = n - i \tag{4}$$

distinct phase mutations with probability at least $\frac{1}{n} \cdot \frac{1}{k}$ each are adequate to improve the fitness value. We therefore require that at least one of the non synchronously received signal components is altered in its signal phase so that this signal component is also in phase afterwards while all other $n-1$ signals remain unchanged. This happens with probability

$$\begin{aligned} & \begin{pmatrix} n-i \\ 1 \end{pmatrix} \cdot \frac{1}{n} \cdot \frac{1}{k} \cdot \left(1 - \frac{1}{n}\right)^{n-1} \\ = \; & \left(\frac{n-i}{nk}\right) \cdot \left(1 - \frac{1}{n}\right)^{n-1}. \end{aligned} \tag{5}$$

Since

$$\left(1 - \frac{1}{n}\right)^n < \frac{1}{e} < \left(1 - \frac{1}{n}\right)^{n-1} \tag{6}$$

We obtain the probability $s_i$ that $L_i$ is left and a better layer is reached due to mutation as

$$s_i \geq \frac{n-i}{n \cdot e \cdot k}. \tag{7}$$

The expected number of mutations to change the layer is bounded from above by $s_i^{-1}$. We consequently obtain the overall expected optimisation time as

$$\begin{aligned} E[X] \quad & \leq \sum_{i=0}^{n-1} \frac{e \cdot n \cdot k}{n-i} \\ & = e \cdot n \cdot k \cdot \sum_{i=1}^{n} \frac{1}{i} \\ & < e \cdot n \cdot k \cdot (\ln(n) + 1) \\ & = O(n \cdot k \cdot \log n). \end{aligned} \tag{8}$$

This upper bound for the expected optimisation time is lower than the optimisation time of the random search method. Assume that the sensor nodes in the network transmit a sine signal for initialisation purposes at 2.4GHz. For a total of 100 signals and 100 possible signal phases that are to be synchronised this corresponds to an optimisation time of less than 0.02ms.

#### 4.1.3 A lower bound for the expected optimisation time

We have now derived that the upper bound for the expected optimisation time of a $(1+1)$-EA is $O(k \cdot n \cdot \log(n))$. A

| Errors | Configurations | Count |
|---|---|---|
| $0$ | $t_1 \ldots t_1;\ t_2 \ldots t_2;\quad \cdots;\ t_k \ldots t_k$ | $k$ |
| $1$ | $t_1 \ldots t_1 t_2;\ t_1 \ldots t_1 t_3;\quad \cdots;\ t_1 \ldots t_1 t_k;\ \cdots;\ t_k \ldots t_k t_{k-1}$ | $\binom{n}{1} \cdot k \cdot \binom{k-1}{1}$ |
| $2$ | $t_1 \ldots t_1 t_2 t_2;\ t_1 \ldots t_1 t_2 t_3;\quad \cdots;\ t_1 \ldots t_1 t_{k-1} t_k;\ \cdots;\ t_k \ldots t_k t_{k-1} t_{k-2}$ | $\binom{n}{2} \cdot k \cdot \binom{k-1}{2}$ |
| | $\vdots$ | $\vdots$ |
| $i$ | | $\binom{n}{i} \cdot k \cdot \binom{k-1}{i}$ |
| | $\vdots$ | $\vdots$ |
| $n-1$ | $t_1 t_2 \ldots t_k;\ t_2 t_3 \ldots t_k t_1;\quad \cdots;\ t_k \ldots t_1$ | $\binom{n}{n-1} \cdot k \cdot \binom{k-1}{n-1}$ |

**Figure 3. All possible configurations for the transmission times in the sensor network.**

legitimate question is of course if there is not any other evolution strategy that has a reasonably lower optimisation time. We can show that this is not the case for any evolutionary strategy with mutation probability $p_m = \frac{1}{n}$. We will actually derive a lower bound for the optimisation time of $\Omega(k \cdot n \cdot \log(n))$. If the population size is $\mu = \Omega(k \cdot n \cdot \log(n))$, the algorithm calculates already $\Omega(k \cdot n \cdot \log(n))$ fitness values for the initial population without finding one of the $k$ global maxima with probability

$$1 - \frac{k}{k^{\Omega(n)}}. \tag{9}$$

When we assume $\mu = O(k \cdot n \cdot \log(n))$, we can use Chernoff bounds to derive that with probability

$$1 - \frac{k}{k^{\Omega}(n)} \tag{10}$$

any one individual in the start population requires at least $\frac{n}{3}$ phase mutations in order to reach one of the $k$ global optima. The probability for a correct mutation is $\frac{1}{n \cdot k}$ at most. Consequently, the probability to not mutate the phase of one of the $\frac{n}{3}$ signal components in $(n \cdot k - 1) \cdot \ln(n)$ iterations is

$$\left(1 - \frac{1}{nk}\right)^{(nk-1)\cdot\ln(n)} \geq e^{-\frac{(nk-1)\cdot\ln(n)}{(nk-1)}} = \frac{1}{n}. \tag{11}$$

Therefore, the probability, that out of $\frac{n}{3}$ signal components at least one is not mutated in $(nk-1) \cdot \ln(n)$ iterations is

$$1 - \left(1 - \frac{1}{n}\right)^{\frac{n}{3}} \geq 1 - e^{-\frac{1}{3}}. \tag{12}$$

Consequently, the expected number of iterations for an evolutionary algorithm with $p_m = \frac{1}{n}$ is bounded from below by

$$\begin{aligned} E[X] &\geq \left(1 - \frac{k}{k^{\Omega(n)}}\right) \cdot \left(1 - e^{-\frac{1}{3}}\right) \cdot (nk-1) \cdot \ln(n) \\ &= \Omega(k \cdot n \cdot \log(n)). \end{aligned} \tag{13}$$

## 4.2 Simulation

We have implemented a Matlab-based simulator for the described collaborative transmission scenario. The simulator represents the distinct phases of the individual signals by integer values. An individual in the search space is constituted by an array of these node configurations. Mutation is applied with probability $\frac{1}{n}$ for each field in the array and each possible follow-up phase is chosen uniformly at random. The fitness value is calculated as the sum of the phase-differences modulo $k$ from each one signal phase to all other signal phases.

$$f_{fitness} = \sum_{i=1}^{n} \sum_{j=1; j \neq i}^{n} (t_i - t_j) \ mod\ k \tag{14}$$

Parameters are the acceptance threshold that specifies when a simulation is stopped successfully, the number of distinct phases for each signal (number of search points per sensor) and the size of the sensor network. Several simulations have been conducted with various parameter configurations.

### 4.2.1 Simulation results

Figure 4 depicts the simulation results from several simulation runs. The network in all of these simulations consists of 50 nodes with various configurations regarding the acceptance threshold and the possible phase count in distinct simulations. For each single configuration a total of 30 simulation runs are computed. A single simulation run is depicted by a star in the figure. Quite intuitively we observe that the optimisation time increases when either the number of possible phases per received signal component increases or the acceptance threshold decreases. The same is true for the size of the sensor network as depicted in figure 5

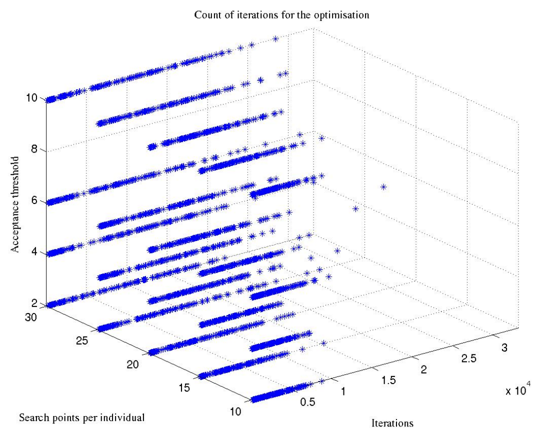


**Figure 4. Simulation results for various acceptance thresholds and number of distinct phases.**

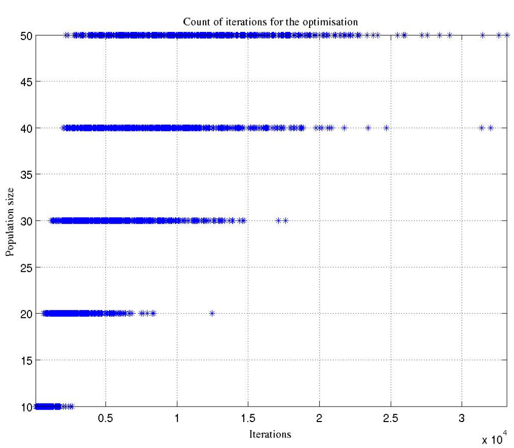


**Figure 5. Number of iterations for various search space sizes**

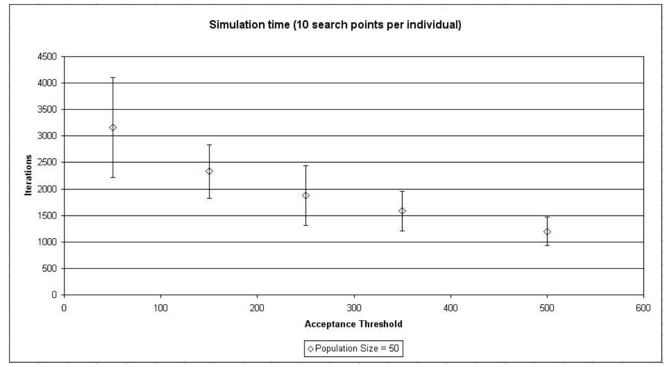


**Figure 6. Simulation time for distinct values of the acceptance threshold.**

With an increasing population of nodes in the network, the optimisation time also increases.

Note that the performance of distinct simulation runs differs greatly from each other since the $(1+1)$-EA is guided randomly in its search.

Figure 6 depicts the mean optimisation times for several configurations of the acceptance threshold and 10 distinct phases per received signal component. As the error bars indicate, the standard deviation increases with decreasing acceptance threshold.

Figure 7 details further simulation results for various phase counts and sizes of the sensor net.

We observe that all, the population size, as well as the number of possible phases per signal component and the acceptance threshold impact the optimisation time. However, with no more than 35000 iterations the optimisation time is in the order of 15 milliseconds for a 2,4GHz signal even when nearly perfect synchronisation of 100 signal components with 30 distinct phases is required.

## 5 Research questions

This section discusses several further aspects of collaborative transmission in sensor networks, that have not been considered so far.

### 5.1 Development of a discrete event simulator

The simulator presented in our work constitutes a useful implementation that is able to illustrate the potential of the method. In order to effectively model a realistic setting, a more advanced approach is required. We believe that discrete event simulators as, for example, Ptolemy II or ns2 are sufficient for a start but have to be extended by modelling superimposed waveforms and a 3D environmental ap-

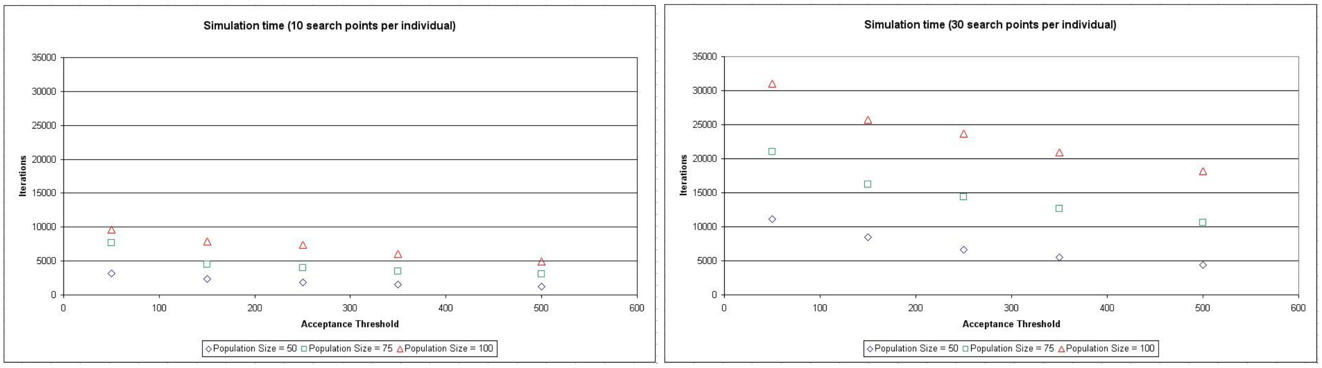


**Figure 7. Simulation times for various search space sizes.**

proach to realistically describe radio propagation. Our team is currently working on the easy integration of real-world-objects into a 3D environment. This covers static objects as well as moving individuals. A simulator that is able to easily adapt and model a realistic environment in 3D should be able to provide a simulation environment that accurately models the radio propagation and multi-path propagation of transmitted signals in this setting. In the long run this leads to a near-realistic simulation environment.

## 5.2 Consideration of other algorithms

Our approach of modelling the optimisation problem with an evolutionary algorithm might differ from the realistic situation in one determining aspect. Mutation for an evolutionary algorithm can at each time reach an arbitrary point in the search space with some probability. While this property is vital to overcome local minima in the search space and might speed up the optimisation process, in a realistic scenario it is likely not possible to reach an arbitrary point in the search space in every step. As explained in section 3.4, the phase of a node can only be gradually shifted in state-of-the-art sensor nodes. Therefore, a neighbourhood-search as, for example, simulated annealing will more adequately describe the realistic situation.

## 5.3 Consideration of movements

Movement of the receiver and the environment has not been considered so far. While the slow fading due to moving obstacles poses a manageable challenge for the collaborative transmission approach, a fast moving receiver as, for example, an aeroplane might easily render the method useless when the adaptation of received signal components is not fast enough. Since the adaptation speed is in the order of milliseconds for a great number of nodes as demonstrated we believe that communication with fast moving objects is possible, although the multipath fading channel might impose a reasonable challenge in this case. In future studies we will regard the ability of collaborative transmission techniques to cope with these difficulties.

## 5.4 Synchronisation between nodes

So far we have not considered nodes to communicate in order to speed up the synchronisation process. Note that nodes that are near to each other will have a similar phase shift of the transmitted signal in an optimal solution. It would therefore be interesting if and how nodes can speed up the initialisation process when communication between nodes is allowed. One possible approach is the clustering of nodes in order to apply divide and conquer methods for iteratively finding an overall solution to the collaborative transmission problem.

By a straightforward estimation of the expected optimisation time we can easily estimate the expected speed up. Assume, for example, that we divide the network of $n$ nodes into $l$ pairwise disjoint clusters of size $\frac{n}{l}$. The overall optimisation time obtained by a divide and conquer approach can then be estimated as

$$O(n \cdot \log(n) + \frac{n}{l} \cdot \log(n) - \left(n + \frac{n}{l}\right) \cdot \log(l)) \tag{15}$$

in analogy to section 4.1.2. This is smaller than $O(n \cdot \log(n))$ for all $l$ with

$$\begin{aligned} & \frac{n}{l} \cdot \log(n) < n + \frac{n}{l} \cdot \log(l). \\ \Rightarrow \quad & l^{l+1} > n \end{aligned} \tag{16}$$

The interesting trade of that is guided by the cluster size $l$ is then the extra power required for a smaller set of nodes to reach the remote receiver location versus the time and consequently transmit power that is saved for solving the optimisation problem.

### 5.5 Increasing the population size

Following the building block hypothesis [14], crossover might speed up the optimisation time when the fitness function is composed of various blocks of individuals in a search space [32, 15] as empirically derived by Rechenberg in [31]. We have not yet considered a crossover operator, since the population in the collaborative transmission scenario is naturally of size one. However, the sensor nodes have some memory that easily suffices to store several configurations. It is therefore feasible to switch in consecutive iterations between several configurations. When all nodes switch between configurations in the same succession, populations of greater size than one can be implemented when several iterations in a row represent various individuals in a population.

This enables advanced crossover operators. It is, however, an open question if and by which degree crossover might speed up the optimisation time since it directly slows down the synchronisation process, as the time to complete one iteration is extended.

## 6 Conclusion

We have demonstrated the potential of collaboratively transmitting data to a remote receiver. The synchronisation between nodes can be obtained by applying an $(1+1)$-EA to optimise a search problem in this domain. We have presented analytic estimates on the expected runtime of the optimisation method and have discussed search results obtained by a matlab-based simulation environment.

It was derived that the asymptotic simulation time for any evolutionary strategy with mutation probability $p_m = \frac{1}{n}$ has an optimisation time that is at least of the same order of the asymptotic optimisation time of the proposed $(1+1)$-EA.

The initialisation time of the approach depends on the signal frequency utilised. With 2.4 GHz signal frequency the initialisation time is in the order of milliseconds for the synchronisation of several hundred nodes. The presented approach is the first solution to collaborative transmission in wireless sensor networks that does not require communication between nodes.

Additionally, we presented considerations of how to model population sizes greater than one as well as a crossover operator in conjunction with a divide and conquer technique. Both techniques might further speed up the optimisation time considerably.

However, the simulator utilised lacks several realistic restrictions like the actual superposition of waveforms due to multipath propagation and fading. Furthermore, since other algorithmic approaches as, for example, simulated annealing might be able to more realistically model the neighbourhood search of a real implementation.

## References


[1] T. Bäck. Optimal mutation rates in genetic search. In S. Forrest, editor, *Proceedings of the 5th International Conference on Genetic Algorithms (ICGA 93)*, pages 2–8. Morgan Kaufmann, 1993.

[2] T. Bäck. An overview of parameter control methods by self-adaptation in evolutionary algorithms. *Fundamenta Informaticae*, 35:51–66, 1998.

[3] D. Culler, D. Estrin, and M. Srivastava. Overview of sensor networks. *IEEE Computer*, 37(8):41–49, August 2004.

[4] A. del Coso, U. Sagnolini, and C. Ibars. Cooperative distributed mimo channels in wireless sensor networks. *IEEE Journal on Selected Areas in Communications*, 25(2):402–414, February 2007.

[5] S. Droste. Efficient genetic programming for finding good generalizing boolean functions. In J. R. Koza, K. Deb, M. Dorigo, D. B. Fogel, M. Garzon, H. Iba, and R. L. Riolo, editors, *Proceedings of the second Genetic Programming conference (GP 97)*, pages 82–87. Morgan Kaufmann, 1997.

[6] S. Droste, D. Heutelbeck, and I. Wegener. Distributed hybrid genetic programming for learning boolean functions. In M. Schoenauer, editor, *Proceedings of the 6th Parallel Problem Solving from Nature (PPSN VI)*, volume 1917 of *Lecture Notes in Computer Science (LNCS)*, pages 181–190. Springer, 2000.

[7] S. Droste, T. Jansen, and I. Wegener. A rigorous complexity analysis of the (1+1) evolutionary algorithm for linear functions with boolean inputs. In *Proceedings of the third IEEE International conference on Evolutionary computation (ICEC 98)*, pages 499–504, Piscataway, NJ, 1998. IEEE press.

[8] S. Droste, T. Jansen, and I. Wegener. A rigorous complexity analysis of the (1+1) evolutionary algorithm for separable functions with boolean inputs. *Evolutionary Computation*, 6(2):185–196, 1998.

[9] S. Droste, T. Jansen, and I. Wegener. On the analysis of the (1+1) evolutionary algorithm. *Theoretical Computer Science*, 276(1-2):51 – 81, April 2002.

[10] S. Droste, T. Jansen, and I. Wegener. Upper and lower bounds for randomized search heuristics in black-box optimisation. *Theory of computing systems*, (4):525–544, 2006.

[11] L. J. Fogel and D. B. Fogel. A preliminary investigation on extending evolutionary programming to include self-adaptation on finite state. *Informatica*, 18(4), 1994.

[12] M. Gastpar and M. Vetterli. On the capacity of wireless networks: the relay case. In *Proceedings of the IEEE Infocom*, pages 1577–1586, June 2002.

[13] O. Giel and I. Wegener. Evolutionary algorithms and the maximum matching problem. In *Lecture Notes in Computer Science (LNCS)*, volume 2607 of *Symposium on Theoretical Aspects of Computer Science*, pages 415 – 426, 2003.

[14] D. E. Goldberg. *Genetic Algorithms in Search, Optimisation and machine Learning*. Addison-Wesley, 1989.

[15] J. J. Grefenstette. Deception considered harmful. volume 2, pages 75–91. Morgan Kaufmann, 1993.

[16] P. Gupta and R. P. Kumar. The capacity of wireless networks. *IEEE Transactions on Information Theory*, 46(2):388–404, March 2000.

[17] J. Holland. Genetic algorithms and the optimal allocation of trials. *SIAM, Journal of computing*, 3(4), 1974.

[18] T. Jansen, K. D. Jong, and I. Wegener. On the choice of the offspring population size in evolutionary algorithms. In *Evolutionary Computation*, number 13, pages 413–440, 2005.

[19] T. Jansen and I. Wegener. Evolutionary algorithms – how to cope with plateaus of constant fitness and when to reject strings of the same fitness. *IEEE Transactions on Evolutionary Computation*, 5(6):589–599, December 2001.

[20] T. Jansen and I. Wegener. Real royal road functions - where crossover provably is essential. In *Genetic and Evolutionary Computation Conference (GECCO)*, pages 375–382, 2001.

[21] J. R. Koza. *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. MIT Press, 1992.

[22] G. Kramer, M. Gastpar, and P. Gupta. Cooperative strategies and capacity theorems for relay networks. *IEEE Transactions on Information Theory*, 51(9):3037–3063, September 2005.

[23] A. Krohn. Optimal non-coherent m-ary energy shift keying for cooperative transmission in sensor networks. In *31st IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, May 2006.

[24] A. Krohn. *Superimposed Radio Signals for Wireless Sensor Networks*. PhD thesis, Technical University of Braunschweig, January 2007.

[25] A. Krohn, M. Beigl, C. Decker, and T. Riedel. Syncob: Collaborative time synchronization in wireless sensor networks. In *Proceedings of the 4th International Conference on Networked Sensing Systems (INSS)*, pages 283–290. Braunschweig, Germany, June 2007.

[26] A. Krohn, M. Beigl, C. Decker, and D. G. Varona. Increasing connectivity in wireless sensor network using cooperative transmission. In *3rd International Conference on Networked Sensing Systems (INSS)*, May 2006.

[27] J. Laneman, G. Wornell, and D. Tse. An efficient protocol for realising cooperative diversity in wireless networks. In *Proceedings of the IEEE International Symposium on Information Theory*, page 294, 2001.

[28] W. Liu, L.-L. Yang, and L. Hanzo. Svd assisted joint multiuser transmitter and multiuser detector design for mimo systems. *IEEE ICC'08*, May 2008.

[29] P. Mitran, H. Ochiai, and V. Tarokh. Space-time diversity enhancements using collaborative communications. *IEEE Transactions on Information Theory*, 51(6):2041–2057, June 2005.

[30] L. Pillutla and V. Krishnamurthy. Joint rate and cluster optimisation in cooperative mimo sensor networks. In *Proceedings of the 6th IEEE Workshop on signal Processing Advances in Wireless Communications*, pages 265–269, March 2005.

[31] I. Rechenberg. *Evolutionssstrategie: Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*. Frommann-Holzboog, 1973.

[32] G. Rudolph. *Convergence Properties of Evolutionary Algorithms*. Verlag Dr. Kovac,, 1997.

[33] A. Scaglione and Y.-W. Hong. Cooperative models for synchronization, scheduling and transmission in large scale sensor networks: An overview. In *1st IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing*, pages 60–63, December 2005.

[34] J. Scharnow, K. Tinnefeld, and I. Wegener. The analysis of evolutionary algorithms on sorting and shortest paths problems. *Journal of Mathematical Modelling and Algorithms*, (3):349–366, 2004.

[35] H.-P. Schwefel. *Evolution and optimum seeking*. Wiley-Interscience, 1995.

[36] D. Shin, S. Y. Na, J. Y. Kim, and S.-J. Baek. Fish robots for water pollution monitoring using ubiquitous sensor networks with sonar localization. In *Proceedings of the IEEE International Conference on Convergence Information Technology*, pages 1298–1303, November 2007.

[37] D. Shin, S. Y. Na, J. Y. Kim, and S.-J. Baek. Sonar localization using ubiquitous sensor network for water pollution monitoring fish robots. In *IEEE International Symposium on Signal Processing and Information Technology*, pages 80–85, December 2007.

[38] C. Shuguang and A. Goldsmith. Energy-efficiency of mimo and cooperative mimo techniques in sensor networks. *IEEE Journal on Selected Areas in Communications*, 22(6):1089–1098, August 2004.

[39] S. Sigg and M. Beigl. Randomised collaborative transmission in wireless sensor networks. In *2nd International Workshop on Design and Integration principles for smart objects (DIPSO2008) in conjunction with Ubicomp 2008*, September 2008.

[40] O. Simeone and U. Spagnolini. Capacity region of wireless ad hoc networks using opportunistic collaborative communications. In *Proceedings of the International Conference on Communications (ICC)*, May 2006.

[41] T. Storch and I. Wegener. Real royal road functions for constant population size. In *Generic and Evolutionary Computation Conference (GECCO)*, volume 2724 of *Lecture Notes in Computer Science (LNCS)*, pages 1406–1417, 2003.

[42] I. Wegener and C. Witt. On the analysis of a simple evolutionary algorithm on quadratic pseudo-boolean functions. *Journal of Discrete Algorithms*, (3):61–78, 2005.

[43] I. Wegener and C. Witt. On the optimization of monotone polynomials by the (1+1)ea and randomized local search. In *Genetic and Evolutionary Computation Conference (GECCO)*, number 2723 in Lecture notes in computer sciences (LNCS), pages 622–633, 2005.

[44] C.-Y. Wei, J. Wang, and L. Hanzo. Exit chart aided irregular convolutional code design for iterative downlink sdma systems using imperfect channel state information. *SiPS*, October 2007.

[45] D. Yang, L.-L. Yang, and L. Hanzo. Performance of sdma mimo systems using transmitter preprocessing based on error-infested finite-predcision vector-quantized channel impulse response feedback. *VTC 07 Spring*, 2007.

[46] F. Zhao and L. Guibas. *Wireless Sensor Networks: An Information Processing Approach*. Morgan Kaufmann, Los Altos, CA, 2004.