AUTOREFERAT

1. Name and Surname

Adam Piotrowski

2. Scientific titles and degrees, including names, place, date and the title of PhD thesis

2001 – Master: Physical Geography, Department of Geography and Regional Studies, Warsaw University

2006 – PhD: Earth Sciences, Institute of Geophysics, Polish Academy of Sciences, the title of PhD thesis: Intelligent Data Analysis in Hydrology

3. Information about the scientific/ artistic employment

2001-2006 – PhD studies in Institute of Geophysics, Polish Academy of Sciences

2006-2006 – assistant in Water Resources Department, Institute of Geophysics, Polish Academy of Sciences

2006-till now – assistant professor in Water Resources Department, Institute of Geophysics, Polish Academy of Sciences, then, after renaming the Department, in the Department of Hydrology and Hydrodynamics, Institute of Geophysics, Polish Academy of Sciences

4. Scientific achievement, according to the art. 16 par. 2 of the act (Dz. U. nr. 65, poz. 595 with later changes):

a) the title of scientific/ artistic achievement

Evolutionary Algorithms – development and application to hydrological variables forecasting

b) author/ authors, title/ titles of the publication, year, Journal name

[1] Piotrowski, A.P. (2013) Adaptive Memetic Differential Evolution with Global and Local neighborhood-based mutation operators. Information Sciences 241, 164-194.

[2] Piotrowski, A.P., Napiorkowski, J.J. (2013) A comparison of methods to avoid overfitting in neural networks training in the case of catchment runoff modeling. Journal of Hydrology 476, 97-111.

[3] Piotrowski, A.P., Napiorkowski, J.J. (2012) Product-Units neural networks for catchment runoff forecasting. Advances in Water Resources 49, 97-113.

[4] Piotrowski, A.P., Rowinski, P.M., Napiorkowski, J.J. (2012) Comparison of evolutionary computation techniques for noise injected neural network training to estimate longitudinal dispersion coefficients in rivers. Expert Systems with Applications 39, 1354-1361.

[5] Piotrowski, A.P., Napiorkowski, J.J., Kiczko, A. (2012) Differential Evolution algorithm with separated groups for multi-dimensional optimization problems. European Journal of Operational Research 216, 33-46.

unfortunately, the paper needed correction, which has been published in the same Journal:

Piotrowski, A.P., Napiorkowski, J.J., Kiczko, A. (2012) Corrigendum to: ‘‘Differential evolution algorithm with separated groups for multi-dimensional optimization problems’’ [Eur. J. Oper. Res. 216 (2012) 33–46]. European Journal of Operational Research 219(2), 488.

[6] Piotrowski, A.P., Napiorkowski, J.J. (2011) Optimizing neural networks for river flow forecasting – Evolutionary Computation methods versus Levenberg–Marquardt approach. Journal of Hydrology 407, 12-27.

[7] Piotrowski, A.P., Napiorkowski, J.J. (2010) Grouping Differential Evolution algorithm for multi-dimensional optimization problems. Control and Cybernetics 39(2), 527-550.

[8] Rowinski, P.M., Piotrowski, A. (2008) Estimation of parameters of transient storage model by means of multi-layer perceptron neural networks. Hydrological Sciences Journal 53(1), 165-178.

c) description of the scientific goal of the above papers and achieved results, including discussion of their possible applications

Scientific disciplines studying the water related issues, like hydrology, hydraulics, water resources, glaciology or oceanography are sometimes considered on the edge of geophysics, nonetheless hydrosphere is without doubt of great importance to the planet Earth. Among disciplines studying the water on Earth, hydrology is probably the most frequently mentioned, as it aims at the most practical and widely sound issues, like floods, droughts, or water quality. In hydrology the main goal is not only to clarify, but also to predict, evaluate or project, what requires application of various kinds of semi-physical, conceptual or data-based models. However, models usually require calibration of a number of parameters, which values due to various reasons cannot be measured or assessed by the experts. This means that the performance of a semi-physical or conceptual models rely to large extent not only on how well they represent or generalize the processes they mimics, but also on the proper choice of the optimization procedure, even if such issue is by some considered purely technical. In case of data-based models, which during the last 20 years become so popular in water-related sciences, the importance of optimization methods is even larger.

My main scientific interest, and the main goal of the papers that constitute the scientific achievement, is to find or propose the most suitable optimization methods for data-based models that are frequently used to help solving specific hydrological problems. This aims at linking hydrology, data-based models and optimization procedures. The eight papers may be divided into two groups: five of them are devoted to comparison of the performance of optimization methods used to solve hydrological problems [2,3,4,6,8], three others [1,5,7] are methodological ones in which the novel optimization methods (belonging to Evolutionary Algorithms that are so popular today) are proposed. Obviously, due to the outstanding number of hydrological problems that may require data-based models which needs calibration, and due to huge number of models and optimization methods that have been proposed so far, the topics of my research had to be restricted. In the papers that constitute the scientific achievement two hydrological problems were considered:

1. Rainfall-runoff forecasting [2,3,6] based on the case study from moderately cold climate zone, namely the Upper Annapolis River catchment, Nova Scotia, Canada;

2. Modelling of the pollutant transport in rivers [4,8].

Both problems will be motivated and described in more details in the further parts of the manuscript where the particular papers are described.

The models applied for both tasks are primarily artificial neural networks (ANN): probably the most popular in various fields of science multi-layer perceptrons (MLP) (in [2,3,4,6,8]) and the little known product-units (PU) (in [3]). However, other model types are also tested, in [8] the performance of MLP is compared with classical regression approaches, in [3] the performance of PU and MLP is compared with HBV conceptual model designed for catchment runoff forecasting.

The ANNs are usually optimized, or trained – a term frequently used in a specific ANN "jargon", by means of the classical gradient-based algorithms. The reason is that the algorithms that use information about derivatives, especially such like outstandingly popular Levenberg-Marquardt algorithm (LM) (Hagan and Menhaj, 1994, Press et al., 2006) are quick, easy to implement and frequently lead to good enough solutions. However, it is commonly claimed that other optimization methods may be needed to train ANNs as (1) the gradient-based methods are not global search approaches and are prone to stick in a local optimum, (2) in some cases the objective function used to train ANNs may be non-differentiable (such a case is considered in the paper [4]). Hence, the temptation to train ANN models by means of so called metaheuristics, among which the most popular today are Evolutionary Algorithms, is easily understandable and not new. The ANN training by means of Evolutionary Algorithms have been discussed in numerous studies during the last two decades (Whitley et al., 1990; Yao, 1993; Branke, 1995, Yao and Liu, 1997; Yao, 1999; Cantu-Paz and Kamath, 2005; Ding et al., 2013 and many more). In addition, Evolutionary Algorithms are sometimes used not only to optimize ANN parameters, but also to find optimal ANN architecture (Yao, 1999; Stanley and Miikkulainen 2002; Islam et al., 2009; Hunter et al., 2012), but this goal was beyond my interest so far, as it is of very limited significance for hydrological problems. As plenty of metaheuristics exist and their wide applicability is frequently claimed (e.g. Fogel, 2000), the problem of ANN training is even sometimes suggested as a good test of their performance (He et al., 2009). However, although the search of proper metaheuristics for ANN training started in the 1980's, it is still far from being finished, as (1) in the vast majority of papers the number of compared optimization methods is very low (rarely more than a few), what prevents the possibility to draw a more general conclusions, (2) different studies frequently lead to contradictory findings (what is an effect of both possible errors and different assumptions made by various researchers).

Evolutionary Computation, i.e. methods inspired by biological evolution, becomes extremely popular during last two decades. Plenty of optimization approaches pertain to this concept, including Evolution Strategies (Bäck and Schwefel, 1993; Hansen and Ostermaier, 1996), Genetic Algorithms (Holland, 1975), Genetic Programming (Koza, 1992), Differential Evolution (Storn and Price, 1995) and many more (see for example Onwobulu and Babu, 2004). There is also a fast growing community of optimization algorithms based on the collective behavior of the swarms of animals, called Swarm Intelligence, which includes approaches like Particle Swarm Optimization (Eberhart and Kennedy, 1995), Ant Colony Optimization (Dorigo et al., 1996), Cat Swarm (Chu et al., 2006) or Group Search Optimizers (He et al., 2009). Methods based on other biological inspirations are also investigated, examples include biogeography-based algorithm (Simon et al., 2008) and cuckoo search approach (Yang and Deb, 2009). However, not only biologically-inspired metaheuristics exist. Following the success of simulated annealing (Kirkpatrick et al., 1983) recently a number of optimization algorithms have been proposed that are said to be inspired by non-biological objects, phenomena, laws or even philosophies, like gravitation (Rashedi et al., 2009), chemical reactions (Lam and Li, 2010), cooperation (Masegosa et al., 2013, Civicioglu, 2013), the concept of Ockham's razor (Iacca et al., 2012; Caraffini et al., 2013) or the beauty of music (Geem et al., 2001). However, not all novel metaheuristics turn out successful and worth application to the practical problems (Crepisek et al., 2012) and the attention must be paid to a number of possible pitfalls when using them to new applications (Weise et al., 2012). It should also be noted that large number of algorithms developed during last 20 years in fact share similarities with the well known direct search methods (Kolda et al., 2003), like Nelder-Mead simplex (NMA) (Nelder and Mead, 1965), the Rosenbrock algorithm (RA) (Rosenbrock, 1960) or the Controlled Random Search (Price, 1977). Putting it together, today one is equipped with swarm of methods that are frequently poorly motivated, rarely their convergence is proven, their behavior is often not well understood and some of them share large similarities with the others without precisely stated reasons. As a result the choice of the optimization metaheuristics that allow solving the particular problem successfully and efficiently is usually not an easy task.

Why so many metaheuristics have been proposed? In general, metaheuristics are needed when the problem to be solved is multi-modal, i.e. has a number of local optima. In case of uni-modal problems, one should refer to gradient-based algorithms (when objective function is differentiable) or some quick and efficient direct search methods, which basic variants are usually known for many years. However, when number of local optima is large, the problem of proper balance between exploitation (finding the location of the nearest local optimum) and exploration (finding the basins of attraction of other, possibly "better" optima) properties of the algorithm becomes crucial. As such "proper balance" may depend on the problem and is very tricky to be determined, plenty of metaheuristics were developed – some of the more recent ones have self-adapting features, which should allow them to be more flexible and successfully tackle different kinds of practical problems.

Obviously in my research I also had to make some initial selection of optimization algorithms. I pay the major attention to the very popular in recent years Differential Evolution family of methods. The choice was motivated by their popularity and the specific behavior during the search, which helps finding the proper balance between global and local search properties. However, variants of selected swarm intelligence approaches, evolutionary strategies, multialgorithms, as well as more classical direct search and gradient-based methods are also applied in the papers that constitute the scientific achievement. In addition, I also invented three novel Evolutionary Algorithms, two of them published in highly esteemed Journals in the fields of Computer Science [1,5]. Some of these algorithms have been successfully applied to solve hydrological problems.

The discussion of particular papers is divided into three thematic groups: papers devoted to the pollutant transport in rivers [8,4], papers aimed at rainfall-runoff forecasting [6,3,2] and papers introducing novel Evolutionary Algorithms [7,5,1]. In most papers discussed below, apart from the search for the most proper Evolutionary Computation methods for the selected task, other specific hydrological and methodological goals are also addressed and will be shortly discussed below.

The description of papers [8,4]

In these papers the problem of finding the values of parameters required to apply two different one-dimensional pollutant transport models in natural rivers is addressed. The paper [8] reports the results of my first research devoted to application of various optimization methods to solve specific hydrological problem.

Most pollutant transport models require a priori knowledge of a few parameters, which represent some morphological or hydraulic properties of particular river reach. Usually their on-site measurement in natural rivers is expensive and time consuming, as it frequently requires performing tracer experiments. Because of that a number of empirical formulae have been devised in the past (Cheong et al., 2007, Wallis and Manson, 2004, Deng et al., 2002 for some review) and in more recent papers the data-driven methods, including ANNs, are frequently proposed to evaluate the values of such parameters based on some easily measurable properties of the river. However, the selection of such river properties, i.e. model input variables, is not trivial, especially that the available data samples are very limited.

In papers [8,4] MLP neural networks are used to estimate parameters of two different pollutant transport models, and their performance is compared with the performance of classical empirical formulae frequently used in real-world applications. Paper [4] is devoted to the evaluation of a single parameter, namely longitudinal dispersion coefficient, which is needed to apply the simplest one-dimensional advection-dispersion equation (Taylor, 1953) at particular river reach. In paper [8] the three parameters of transient storage model (Bencala and Walters, 1983), which is also commonly used to describe the pollutant transport in rivers, are evaluated for each considered river reach.

In the paper [8] three optimization methods were tested to train neural networks, namely the classical gradient-based Levenberg-Marquardt algorithm which is today probably the most popular ANN training method, and two basic variants of the methods that rapidly gain popularity in very different fields of science, namely Differential Evolution and Particle Swarm Optimization. The main outcomes of the paper were as follows: 1. MLP neural networks highly outperform older empirical formulae and linear regression approach for estimation of all three parameters of transient storage model. However, such performance, especially when the parameter describing the residence time in storage zones is considered, is still not good enough to allow recommendation of the method for practical applications. Although the evaluations of the dispersion coefficient and storage zone area parameters were better, the proper values of all three parameters are needed to apply transient storage zone model. Unfortunately, it seems that reaching sufficient improvement may not be possible until much more data from tracer experiments were available. 2. In spite of the obtained results, the basic Differential Evolution algorithm seems to be a competing method to the widely-used Levenberg-Marquardt algorithm. However, the experiment was performed on small data sample, what may have a significant impact on the conclusions. Indeed, the success of basic Differential Evolution variant was not confirmed in my further works, and is also not confirmed in the majority of papers published by other authors.

The study [4] aimed at estimation of longitudinal dispersion coefficients and comparison of different training algorithms. An additional goal was to test the noise injection method to avoid neural network's overfitting. As neural networks are considered to be universal approximators (Hornik, 1989) they are prone to overfitting, i.e. may fit not only to the signal, but also to the noise present in the training sample. There are various ways to limit the danger of overfitting, one of such method, called noise injection (Holmstrom and Koistinnen, 1992) is said to be very useful in case of small data sample and hence has been tested in the study. In addition, as the fitness function (mean absolute error) used was not differentiable, this paper examines the applicability of Evolutionary Algorithms in the case when classical ANN training methods, like Levenberg-Marquardt algorithm, indeed cannot be used. The problem of longitudinal dispersion coefficient estimation by means of ANNs was also addressed in my first Journal paper published before PhD (Rowinski et al., 2005), and in PhD itself, however neither Evolutionary Algorithms nor specific methods to avoid overfitting were not tested there. In the paper [4] nine different global optimization methods, including six variants of Differential Evolution, two variants of Particle Swarm Optimization and a variant of Evolutionary Strategy (CMA-ES) were compared for MLP neural network's training with noise injection. To allow a fair comparison, each algorithm was applied 50 times for the same task. It was found that the difference between five relatively new Differential Evolution methods is low. However, the GDE method which I proposed in paper [7] was according to the results the most successful one. The paper also showed that even if the difference between the performances of the best methods is small, the choice of optimization method is of large importance, as about half of tested global search algorithms perform noticeably poorer. The performances of the results obtained when two the poorest algorithms are used were even inferior to the predictions obtained from an older empirical formulae. The use of noise injection allowed in most cases the reduction of mean absolute error by 2-20% for independent testing data, depending on the variant considered.

Description of papers [6,3,2]

The rainfall-runoff forecasting seems to be one of the most popular and important topic in hydrology. Since the paper by Hsu et al. (1995) ANNs have also been widely used to this task. However, as suggested by Abrahart et al. (2012) the use of neural networks for river flow forecasting during last two decades was somehow chaotic. Little attention has been put to methodological details, which may however affect the model performance. The three papers [6,3,2], although aim at solving practical hydrological problem, also pay special attention to particular methodological details, that will be given below.

From the hydrological point of view all three papers [6,3,2] were devoted to the one day ahead runoff forecasting at Upper Annapolis River catchment located in Nova Scotia, Canada, based on hydrometeorological data. The catchment is hilly (the highest point do not reach 300 m high) and mostly covered by forests. The Annapolis River is located within the Humid Continental Climate Zone according to the Köppen Climate Classification, with common snowfall from November to April and significant temperature variations during the winter months that results in frequent freezing and melting events. During summer season rainfalls do occur, however due to the high temperature and evapotranspiration, the average runoff in Annapolis River is the lowest from July to September. Although such climate conditions do clearly differ from the ones observed in Poland, there are also similarities between Poland and Nova Scotia – in both places the large seasonal variations occur and similar processes, connected with cold and snow, affect the runoff in winter and spring. Both Poland and Nova Scotia are included in the same climate zone by Koppen Climate Classification. Such similarities and data availability were important motivation of the choice of the research location.

The 30 years long daily hydro-meteorological data were collected from the Water Survey of Canada and the Canada's National Climate Data and Information Archive, for the gauge station situated in Wilmot settlement (catchment area 546 km^2) and the meteorological station located at the Greenwood Airfield, 9 km to the east. However, in the first paper [6], only 10-year long data series were used. Meteorological data include the maximum and minimum daily air temperature, rainfall, snowfall and snow cover. The concentration time of the river is estimated to be, depending on the method, roughly below 1 day. This, together with the availability of long hydrological and meteorological data series with very few gaps and significant daily runoff changes that occur in the Annapolis catchment in winter and spring periods, makes it a good place for daily rainfall-runoff modelling at moderately cold climate zones. Although this is beyond the scientific achievement given in this report, I feel free to add that presently I am working on the comparison of different neural networks and conceptual models applied for rainfall-runoff forecasting in Nova Scotia and catchments located in southern Poland. The mentioned study includes also application of various optimization algorithms to each tested model.

From the methodological point of view apart from the comparison of optimization algorithms, the additional specific goals were addressed in the papers [3] and [2]. Paper [3] aimed to introduce very simple type of higher order ANNs called Product-Units and show their application to rainfall-runoff forecasting. Paper [2] verifies the significance of application of various ANN overfitting methods in rainfall-runoff modelling, when data samples are large.

The paper [6] was my first work combining rainfall-runoff forecasting, neural networks and Evolutionary Algorithms. Only 10-years long hydrological and meteorological data series were used, of which 4 years were put aside during model's calibration as independent test set. The optimization experiments were performed for several different MLP architectures, including different variants of input variables. The Levenberg-Marquardt algorithm and eight Evolutionary Algorithms were applied to ANN training. Each method was applied 50 times for every MLP architecture to give data sample that may allow seeing the difference in the modeling performance. Most of these algorithms were relatively new methods at the time of writing the paper and many of them have never been compared witch each other before. The 50-run average and median performances were given in the paper, as frequently judging only on the median or on the average may lead to different conclusions. It was found that the performances do differ noticeably when different training methods are used, the best results were obtained by means of two algorithms: Differential Evolution variant called DEGL (Das et al., 2009) and Levenmberg-Marquardt algorithm. Among other tested methods only the performance of EPUS-PSO (Hsieh et al., 2009) seemed promising. Among the two winners, Levenberg-Marquardt algorithm was, as expected, much quicker than DEGL. The performance of Differential Evolution variants deteriorated quicker with the number of parameters to be optimized than the performance of Levenberg-Marquardt algorithm or Particle Swarm Optimization approaches. From the hydrological perspective, two best optimization methods allowed to forecast the daily runoff with the value of Nash-Sutcliffe coefficient being over 0.91 for independent testing data. The importance of the training algorithm may be easily seen when one notice that for the poorest methods the values of Nash-Sutcliffe coefficient dropped below 0.5 and in the worst cases reached around 0. This result was important as a warning that only some Evolutionary Algorithms may be a true competition to the most efficient gradient-based methods, and that not right choice of optimization method may lead to significant underestimation of the model performance due to poor calibration.

The paper [3] introduces Product-Units ANNs to the hydrological problems. This kind of ANNs, although proposed years ago by Durbin and Rumelhart (1989), was surprisingly overseen by the hydrological community. The advantage of PU over MLP and other ANN types is the relatively low number of parameters. However, because PU input variables are raised to exponential weights, as showed in the literature they need a special precaution during data pre-processing and their fitness function is considered extremely bumpy, hence difficult to be optimized. To present a more complete framework for the PU preparation for a specific task, the paper [3] signifies the importance of proper selection of optimization algorithms, methods aimed at improving a neural network's generalization ability, model architectures, and parameter box-constrains. In [3] the proposed method is compared with the MLP neural networks and widely applied in practice HBV conceptual model. In this paper the 30-year long data series were used, of which last 10 years were used as independent testing data. A few PU architectures were tested in the study, but they differ only by the number of hidden nodes, as input variables were kept the same as in the best architecture found for MLP in the paper [6]. Like in previous studies, each algorithm was applied 50 times for each PU architecture. The mean and median values of the performances were given for each considered variant.

Of significant importance in the paper [3] was the idea to introduce box-constraints on PU weights. It was shown that when all PU parameters are restricted within [-1,1] interval, PU optimization problem would be much simpler. The results achieved were up to 10% better (comparing the mean square error), then obtained by means of MLP or HBV variants. Using higher or lower box-constraints lead to deterioration of the results.

Among 11 tested optimization algorithms there were 9 Differential Evolution variants, including DEGL and GDE that performed well in the previous studies [6,4]; the other 2 algorithms were Levenberg-Marquardt and EPUS-PSO method. Along with DEGL, also the variant of DEGL with Proximity-based mutation operator introduced in Epitropakis et al. (2011) was used. As in the previous study on rainfall-runoff modelling [6] at Annapolis catchment, the performance of DEGL (and its variant with Proximity-based mutation operator) and Levenberg-Marquardt algorithms was found to outperform the performance of the other methods. However, additional tests for selected PU architecture were performed and it was found that if sufficiently long training is allowed, the performance of almost all Differential Evolution variants would be much more similar. This means that the main advantage of DEGL for PU training lies in its speed.

It was surprisingly found that when PU with weights limited within [-1,1] interval is used methods to prevent ANN overfitting seems not needed. Such result is in contradiction with findings obtained by MLP neural networks and is either reached “by chance” for the selected problem, or shows the another advantage of PU over MLP for rainfall-runoff modelling. This, however, must be verified in future on the other data.

In the paper it was also found that the one-lead day forecasting performance of HBV conceptual model with updating procedure and the performance of MLP neural networks are very similar. As mentioned before, the performance of PU neural networks is evaluated as up to 10% better.

The paper [2] is devoted to the validation of efficiency of different methods to avoid ANN overfitting applied to rainfall-runoff forecasting. Such studies in hydrology are very rare – probably the only paper devoted to the subject was published by Giustolisi and Laucelli (2005), who studied the impact of such methods on the performance of rainfall–runoff modelling at two very small catchments (up to 5 km^2) in Italy. In their study Evolutionary Algorithms were not considered. The paper [2] presents a deliberate comparison of the performance of Levenberg-Marquardt algorithm and DEGL approach used together with different methods to prevent ANN overfitting. Also the catchment scale is much larger than in the study by Giustolisi and Laucelli (2005), and the climate conditions differ significantly.

In the paper [2] three kinds of methods to avoid ANN overfitting are compared: an early stopping according to Prechlet (1998), the Optimized Approximation Algorithm proposed by Liu et al., (2008), and several variants of noise injection based on Holmstrom and Koistinnen (1992) approach. To see the impact of the number of parameters to be optimized on the performance of the model trained by particular optimization method composed with particular approach to avoid ANN overfitting, different architectures (with various input variables and different number of hidden nodes) were tested. The data used were the same as in the paper [3], optimization performed by means of both algorithms was repeated 50 times for each considered MLP architecture and method to avoid ANN overfitting.

It was found that the noise injection method may be the most successful, but only if the noise injection parameters are chosen properly. This is, however, time consuming not easy. Moreover, in case of noise injection the standard deviation of the results may be high. The performance obtained with much simpler early stopping method is only slightly inferior, but the method is quicker. Hence, the suggestion of the method to avoid overfitting must depend on the amount of work and time one is ready to invest. Optimized Approximation Algorithm, although the most recent among tested methods, perform poorly for the selected problem, probably due to two very technical details, discussed more deeply in the paper [2].

The comparison between Levenberg-Marquardt algorithm and DEGL showed that DEGL is more sensitive to the curse of dimensionality, what could be expected from the experience of the paper [6]. When DEGL optimization algorithm is used smaller MLP architectures lead to relatively better results. Levenberg-Marquardt algorithm may be successfully used also for larger architectures. The performance of Levenberg-Marquardt algorithm achieved for larger ANN architecture turned out better than the performance of DEGL achieved for the simpler one. On the other hand, if the performance reached for two simple architectures is compared, DEGL slightly outperforms Levenberg-Marquardt algorithm.

The description of papers [1,5,7]

Except practical application of various optimization methods to hydrological problems, in my work I also focused on the development and improvement of Evolutionary Algorithms themselves. Three novel Evolutionary Algorithms, all belonging to Differential Evolution family of methods have been proposed. Differential Evolution algorithms gain today a significant popularity and are very rapidly developing in recent years (Das and Suganthan, 2011). My first two algorithms [7,5] are classified as distributed Differential Evolution methods, as their main advantage lies in the distribution of population into sub-populations that most of computational time work independently, but occasionally share information or exchange individuals among themselves. In distributed Evolutionary Algorithms the main difficulty lies in the proper development of rules that govern the process of sharing or exchanging information or individuals. The third algorithm, proposed recently in paper [1] is a kind of adaptive memetic Differential Evolution method.

In paper [7] my first optimization method, called Grouped Multi-Strategy Differential Evolution (GDE) algorithm was proposed. The main idea behind this approach was to exploit the knowledge about the local minima already found in different parts of the search space in order to facilitate further search for the global one, using the concept of distributed computing. The population of individuals is distributed into four groups. Three of them very rarely communicate with the others, but one is allowed to gain all available knowledge from the whole population throughout the whole search. The individuals simultaneously use three different crossover and mutation strategies, what makes the algorithm more flexible than the classical Differential Evolution approaches. The proposed algorithm was compared with two other Differential Evolution variants on thirteen 10- to 100-dimensional benchmark functions of varying difficulty. The proposed method achieved very encouraging results; its advantage was especially seen when more difficult among tested 50- and l00-dimensional problems were considered. Distributed Differential Evolution algorithms are frequently sensitive to the population size, as both using too small or too large groups, or sub-populations, may highly disturb their performance. Hence, the impact of the number of individuals on the performance of GDE was also studied in the paper. GDE algorithm, proposed in paper [7], was further applied to neural networks calibration for evaluation of longitudinal dispersion coefficients in rivers (in [4] and Piotrowski et al., 2010) and rainfall-runoff modelling (in [6,3]). However, in my further work with optimization algorithms I found that the GDE algorithm could be noticeably improved. I also realized that the number of competing algorithms used in paper [7] was rather small and the thirteen benchmark functions that were used were probably not the best choice, as they did not include rotated problems. However, since the work of Salomon (1996) it is well known that Evolutionary Algorithms should be tested against rotated problems, as some of them are, due to the nature of most popular crossover operations, very successful in solving various problems as long as they are separable or the local optima are located parallel to the coordinate avis. This motivated the further research that led to the much improved version of the method, called Differential Evolution algorithm with Separated Groups (DE-SG) that was published in [5].

The algorithm proposed in paper [5] was based on GDE, but some its features were inspired by other distributed and self-adaptive Differential Evolution methods, as well as so-called Island Models (Tanese, 1989). However, DE-SG differs in its structure and mechanisms aimed at sorting and exchanging information between sub-populations (or groups). Unlike in the majority of distributed Differential Evolution methods, the population of individuals is divided into halves and rules of migration of individuals are different in each half. Each half is further divided into groups (each contains exactly 10 individuals) that operate independently. Because the exchange of information within a small group is quicker, small groups are able to speed up exploitation. To facilitate exploration, communication between individuals belonging to different sub-populations and exchange of individuals between sub-populations are also allowed under specific circumstances. The truly novel idea proposed for the algorithm was that rules governing the migration of individuals between groups within each half differ. In both halves the groups are ranked; the best individuals are attracted to the group in the one utmost edge, the poorest – to the group in the opposite edge. Within one half, the best individuals migrate relatively quickly to an elite group, while the poorest one migrates slowly. Within the other half the best individuals migrate slowly, hence are distributed more widely among various groups. This gives the algorithm additional flexibility. Following the experiences from outstandingly popular SADE (Qin et al., 2009) and GDE, in DE-SG an offspring may be produced by one of two strategies of different nature. The first one is expected to perform better exploration, the second – exploitation. The proposed method was successfully compared to eight state of the art Evolutionary Computation algorithms, including some of Differential Evolution variants recently proposed in highly praised Journals, based on nineteen rotated 10- to 50-dimensional test problems. However, the very high value of maximum number of function calls set for each method may be a kind of disadvantage of the paper (most researchers set much smaller values).

My third variant of Differential Evolution algorithm, published in paper [1] is not based on the concept of distributed Evolutionary Algorithms. Although plenty of Differential Evolution variants were proposed so far, bringing together different ideas that already led to successful algorithm is rare in the literature. In the novel approach proposed in paper [1] three among the most efficient concepts already applied separately within Differential Evolution framework are gathered together. Firstly, the adaptation of algorithm control parameters and probabilities of using different mutation strategies (5 are used in the algorithm) is introduced, following the concept elaborated in Qin et al. (2009). Secondly, the Nelder–Mead algorithm is used as a local search method hybridized with Differential Evolution, such idea was already successfully applied in Caponio et al. (2009). Thirdly, the mutation is split into Global and Local models, when Local mutation model is based on the concept of neighborhood of individuals organized on a ring topology – such idea was borrowed from Das et al. (2009). The performance of the novel algorithm, called Adaptive

Memetic Differential Evolution with Global and Local neighborhood-based mutation operators (AM-DEGL) is compared with thirteen different Differential Evolution variants, including the most recent ones and DE-SG that I proposed in paper [5], on a set of 25 popular problems which include rotated, shifted and hybrid composition functions. In the paper [1] it was found that, although none among tested Differential Evolution algorithms outperform all the others for the majority of problems, on average the proposed AM-DEGL perform better than all 13 Differential Evolution algorithms selected for comparison. This showed the possible strength of the idea of bringing together different successful ideas into a single algorithm. Also in the paper [1] the importance of different components of novel algorithm is tested, showing that eliminating some of them usually only moderately affect the performance of the algorithm.

In most papers devoted to the novel Evolutionary Algorithms the authors restrict their attention to showing that their approach is, in some sense, “better” than the competing methods. However, in the paper [1] also different point of view was discussed. The question may arise, whether proposing novel Evolutionary Algorithms is useful as No Free Lunch theorems for optimization (Wolpert and Macready, 1997) state that the expected performance of all possible heuristics on all possible problems is equal. This means that none metaheuristic may perform better than random search. Hence in the last section of the paper [1] the limitations and implications of No Free Lunch theorems were discussed based on rich, but unfortunately frequently neglected literature. Without getting into details, important point in the discussion is the meaning of “all problems” – among them the ones that may be of interest to anyone are very rare, and vast majority of fitness landscapes looks like a random blurry. This led some researchers to put aside No Free Lunch theorems. However, Evolutionary Algorithms may fail even on problems that looks relatively simple and seems to be of interest to someone. As an example, a very simple continuous and differentiable 2-dimensional minimization problem with box constraints was proposed in paper [1], for which it was empirically verified that each among 14 Differential Evolution algorithms tested in the paper perform on average poorer than random sampling. It was also empirically shown that when all such Differential Evolution algorithms search for the maximum of the proposed problem, they found lower objective function values than Differential Evolution algorithms searching for the minimum. That such problems exist was, of course, expected based on No Free Lunch theorems. The idea behind such discussion was to give a short review of No Free Lunch theorems and stimulate the debate on the ways the Evolutionary Algorithms are compared and “promoted” in the scientific papers.

Conclusions

One of the main obstacles in hydrology is frequently number and quality of available data. If number of data is not large enough, like in case of longitudinal dispersion coefficient estimation, the performance of models optimized by means of different algorithms is moderately diversified (but by no means similar). However, when sufficiently long data sets are available, like in case of rainfall-runoff modeling in Annapolis river catchment, the performance of the model significantly depends on the optimization method. What may be a surprise, despite the fact that large number of metaheuristics are used to neural networks training in various papers, and that it is widely known that gradient-based methods may stick in a local optima, according to the achieved results most of tested Evolutionary Algorithms proposed in recent years or widely used today are not very useful in artificial neural networks training. The performance of only a few methods is comparable, or slightly better, than the performance of the most efficient gradient-based approaches. Interestingly, such few methods, like DEGL, EPUS-PSO, in some cases GDE that I proposed myself, are not the ones that are the most efficient according to tests performed on benchmark problems.

References:

Abrahart, R.J., Anctil, F., Coulibaly, P., Dawson, C.W., Mount, N.J., See, L.M., Shamseldin, A.Y., Solomatine, D.P., Toth, E., Wilby, R.L., 2012. Two decades of anarchy? Emerging themes and outstanding challenges for neural network river forecasting. Progress in Physical Geography 36(4), 480-513.

Bäck, T., Schwefel, H. P., 1993. An overview of evolutionary algorithms for parameter optimization. Evolutionary Computation 1(1), 1-23.

Bencala, K.E., Walters, R.A., 1983. Simulation of solute transport in a mountain pool-and-riffle stream: a transient storage model. Water Resources Research 19(3), 718–724.

Branke, J., 1995. Evolutionary algorithms for neural network design and training. Technical Report No. 322, Institute AIFB, University of Karlsruhe.

Cantu-Paz, E., Kamath, C., 2005. An empirical comparison of combinations of evolutionary algorithms and neural networks for classification problems. IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics 35(5), 915-927.

Caponio, A., Neri, F., Tirronen, V., 2009. Super-fit control adaptation in memetic differential evolution frameworks. Soft Com-puting 13(8-9), 811-831.

Caraffini, F., Neri, F., Iacca, G., Mol, A., 2013. Parallel Memetic Structures. Information Sciences 227(1), 60-82.

Cheong, T.S., Younis, B.A., Seo, I.W., 2007. Estimation of key parameters in model for solute transport in rivers and streams. Water Resources Management 21(7), 1165–1186.

Chu, S., Tsai, P., Pan, J., 2006. Cat swarm optimization. Lecture Notes in Computer Science 4099, 854–858.

Civicioglu, P., 2013. Artificial cooperative search algorithm for numerical optimization problems. Information Sciences 229, 58-76.

Crepinsek, M., Liu, S.H., Mernik, L., 2012. A note on teaching-learning-based optimization algorithm. Information Sciences 212, 79-93.

Das, S., Abraham, A., Chakraboty, U.K., Konar, A., 2009. Differential Evolution using a neighborhood-based mutation operator. IEEE Transactions on Evolutionary Computation 13(3), 526-553.

Das, S., Suganthan, P.N., 2011. Differential Evolution: a survey of the state-of-the-art. IEEE Transactions on Evolutionary Computation 15(1), 27-54.

Deng, Z.Q., Bengtsson, L., Singh, V.P., 2002. Longitudinal dispersion coefficient in single-channel streams. Journal of Hydraulic Engineering ASCE 128(10), 901–916.

Ding, S., Li, H., Su, C., Yu, J., Jin, F., 2013. Evolutionary artificial neural networks: a review. Artificial Intelligence Review 39, 251-260.

Dorigo, M., Maniezzo, V., Colorni, A., 1996. Ant system: Optimization by a colony of cooperating agents. IEEE Transactions on Systems Man and Cybernetics – Part B: Cybernetics 26(1), 29-41.

Durbin, R., Rumelhart, D.E., 1989. Product Units: a computationally powerful and biologically plausible extension to backpropagation networks. Neural Computation 1, 133–42.

Eberhart, R.C., Kennedy, J., 1995. A new optimizer using particle swarm theory. In: Proc. 6th Int. Symp. Micromachine Hu-man Sci., Nagoya, Japan, 1995, pp. 39-43.

Epitropakis, M.G., Tasoulis, D.K., Pavlidis, N.G., Plagianakos, V.P., Vrahatis, M.N., 2011. Enhancing differential evolution utilizing proximity-based mutation operations. IEEE Transactions on Evolutionary Computation 15(1), 99-119.

Fogel, D.B., 2000. What is Evolutionary Computation? IEEE Spectrum 37(2), 26-32.

Geem, Z.W., Kim, J.H., Loganathan, G.V., 2001. A new heuristic optimization algorithm: Harmony search. Simulation 76(2), 60-68.

Giustolisi, O., Laucelli, D., 2005. Improving generalization of artificial neural networks in rainfall-runoff modelling. Hydrological Sciences Journal 50(3), 439-457.

Hagan, M.T., Menhaj, M.B., 1994. Training feedforward networks with the Marquardt algorithm. IEEE Transactions on Neural Networks 5(6), 989-993.

Hansen, N., Ostermeier, A., 1996. Adapting arbitrary normal mutation distributions in evolution strategies: The covariance matrix adaptation. In Proc. IEEE Int. Conf. Evol. Comput., Nagoya, Japan, pp. 312–317.

He, S., Wu, Q.H., Saunders, J.R., 2009. Group search optimizer: an optimization algorithm inspired by animal search behaviour. IEEE Transactions on Evolutionary Computation 13(5), 973-990.

Holland, I.H., 1975. Adaptation in Natural and Artificial Systems. University of Michigan Press. Ann Arbor.

Holmstrom, L., Koistinen, P., 1992. Using additive noise in back-propagation training. IEEE Transactions on Neural Networks 3, 24-38.

Hornik, K., Stinchcombe, M., White, H., 1989. Multilayer feed forward networks are universal approximators. Neural Networks 2, 359–366.

Hsieh, S.T., Sun, T.Y., Liu, C.C., Tsai, S.J., 2009. Efficient population utilization strategy for Particle Swarm Optimizer. IEEE Transactions on Systems, Man, and Cybernetics – Part B: Cybernetics 39(2), 444-456.

Hsu, K.L., Gupta, H.V., Sorooshian, S., 1995. Artificial neural network modeling on the rainfall–runoff process. Water Research and Resources 31(10), 2517–2530.

Hunter, D., Yu, H., Pukish, M.S., Kolbusz, J., Wilamowski, B.M., 2012. IEEE Transactions on Industral Informatics 8(2), 228-240.

Iacca, G., Neri, F., Mininno, E., Ong, Y.S., Lim, M.H., 2012. Ockham's razor in memetic computing: three stage optimal memetic exploration. Information Sciences 188, 17-43.

Islam, M.M., Sattar, M.A., Amin, M.F., Yao, X., Murase, K., 2009. A new constructive algorithm for architectural and functional adaptation of artificial neural networks. IEEE Transactions on Systems, Man and Cybernetics – Part B: Cybernetics 39(6), 1590-1605.

Kirkpatrick, S., Gelatt, C.D., Vecchi, M.P., 1983. Optimization by simulated annealing. Science 220, 671-680.

Kolda, T.G., Lewis, R.M., Torczon, V., 2003. Optimization by direct search: new perspectives on some classical and modern methods. SIAM Review 45(3), 385-482.

Koza, J.R., 1992. Genetic Programming: On the programming of computers by means of natural selection. MIT Press.

Lam, A.Y.S., Li, V.O.K., 2010. Chemical reaction inspired metaheuristic for optimization. IEEE Transactions on Evolutionary Computation 14(3), 381-399.

Liu, Y., Starzyk, J.A., Zhu, Z., 2008. Optimized approximation algorithm in neural networks without overfitting. IEEE Transactions on Neural Networks 19(6), 983-995.

Masegosa, A.D., Pelta, D.A., Verdegay, J.L., 2013. A centralised cooperative strategy for continuous optimisation: The influence of cooperation in performance and behaviour. Information Sciences 219, 73-92.

Nelder, A., Mead, R., 1965. A simplex-method for function minimization. Computer Journal 7(4), 308-313.

Onwobulu, G.C., Babu, B.V., 2004. New optimization techniques in engineering. Studies in Fuzziness and Soft Computing, Vol. 141, Springer-Verlag.
Piotrowski, A.P., Rowinski, P.M., Napiorkowski, J.J. (2010) Uncertainty study of data-based models of pollutant transport in rivers. Proceedings of River Flow 2010 Conference, Braunschweig, Germany, 8-10 September.
Prechlet, L., 1998. Automatic early stopping using cross-validation: quantifying the criteria. Neural Networks 11(4), 761-777.
Press, W.H., Flannery, B.P., Teukolsky, S.A., Vetterling, W.T., 2006. Numerical recipes in Fortran 77. The Art of Scientific Computting. Cambridge University Press, Cambridge, UK.
Price, W.L., 1977. A Controlled Random Search procedure for global optimisation. Computer Journal 20(4), 367-370.
Qin, A.K., Huang, V.L., Suganthan, P.N., 2009. Differential Evolution algorithm with strategy adaptation for global numerical optimization. IEEE Transactions on Evolutionary Computation 13(2), 398-417.
Rashedi, E., Nezamabadi-pour, H., Saryazdi, S., 2009. GSA: A gravitational search algorithm. Information Sciences 179, 2232–2248.
Rosenbrock, H.H., 1960. An automated method for finding the greatest or least value of a function. Computer Journal 3(3), 175-184.
Rowinski, P.M., Piotrowski, A., Napiorkowski, J.J., 2005. Are artificial neural networks techniques relevant for the estimates of longitudinal dispersion coefficient in rivers? Hydrological Sciences Journal 50(1), 175-187.
Salomon, R., 1996. Re-evaluating genetic algorithm performance under coordinate rotation on benchmark functions. A survey of some theoretical and practical aspects of genetic algorithms. BioSystems 39, 263–278.
Simon, D., 2008. Biogeography-based optimization. IEEE Transactions on Evolutionary Computation 12(6), 702-713.
Stanley, K.O., Miikkulainen, R., 2002. Evolving neural networks through augmenting topologies. Evolutionary Computation 10(2), 99-127.
Storn, R., Price, K.V., 1995. Differential Evolution – a simple and efficient adaptive scheme for global optimization over continuous spaces. Tech. Report TR-95-012, International Computer Sciences Institute, Berkeley, California, USA.
Tanese, R., 1989. Distributed genetic algorithms. Proceedings of 3rd International Conference on Genetic Algorithms, San Francisco, CA, USA, pp. 434–439.
Taylor, G.I., 1953. Dispersion of soluble matter in solvent flowing slowly through a tube. Proceedings of the Royal Society of London, Series A, 219, 186–203.
Wallis, S.G., Manson, J. R., 2004. Methods for predicting dispersion coefficients in rivers. Water Management, 157(WM3), 131–141.
Weise, T., Chiong, R., Tang, K., 2012. Evolutionary optimization: Pitfalls and booby traps. Journal of Computer Science and Technology 27(5), 907-936.
Whitley, D., Starkweather, T., Bogart, C., 1990. Genetic algorithms and neural networks: optimizing connections and connectivity. Parallel Computing 14(3), 347-361.
Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. IEEE Transactions on Evolutionary Computation 1(1), 67–82.
Yang, X.S., Deb, S., 2009. Cuckoo Search via Levey Flights. 2009 World Congress on Nature & Biologically Inspired Computing, pp. 210-214.
Yao, X., 1993. A review of evolutionary artificial neural networks. International Journal of Intelligent Systems 8(4), 539-567.
Yao X., Liu, Y., 1997. A new evolutionary system for evolving artificial neural networks. IEEE Transactions on neural networks 8(3), 694-713.

Yao, X., 1999. Evolving artificial neural networks. Proceedings of the IEEE 87(9), 1423-1447.

5. Discussion of the other scientific/ artistic achievements

Transport of pollutants in rivers

I was involved in tracer experiments performed on Narew river under the Grant: Procesy transportowe w korytach rzecznych (principal investigator: Pawel Rowinski). I took part in the field experiments and in the evaluation of longitudinal dispersion coefficients for studied river reaches, both by means of artificial intelligence methods and directly from the data collected during the tracer experiments.

In addition I performed a few studies using published data on the application of artificial neural networks for evaluation of pollutant transport in rivers. Apart from the two papers that aimed at application of Evolutionary Algorithms to this topic and were discussed in the main scientific achievement part, the results have been published in three other Journal papers and several conference proceedings.

Published papers:

Piotrowski, A.P., Napiorkowski, J.J., Rowinski, P.M., Wallis, S.G., 2011. Evaluation of temporal concentration profiles for ungauged rivers following pollution incidents. Hydrological Sciences Journal 56(5), 883-894.

Piotrowski, A., Wallis, S.G., Napiorkowski, J.J., Rowinski, P.M., 2007. Evaluation of 1-D tracer concentration profile in a small river by means of multi-layer perceptron neural networks. Hydrology and Earth System Sciences 11, 1883-1896.

Rowinski, P.M., Piotrowski, A., Napiorkowski, J.J., 2005. Are artificial neural networks techniques relevant for the estimates of longitudinal dispersion coefficient in rivers? Hydrological Sciences Journal 50(1), 175-187.

Napiorkowski J.J., Piotrowski A., Rowinski P.M., Wallis S.G., 2012. Product Unit neural networks for estimations of longitudinal dispersion coefficients in rivers. 2nd IAHR Europe Congress, 27-29 June, Germany, Munich.

Piotrowski, A.P., Rowinski, P.M., Napiorkowski, J.J., 2010. Uncertainty study of data-based models of pollutant transport in rivers. Proceedings of River Flow 2010 Conference, Braunschweig, Germany, 8-10 September.

Piotrowski, A.P., Rowinski, P.M., Napiorkowski, J.J., 2009. Estimation of parameters of models of pollutant transport in rivers depending on data availability. 33rd IAHR Congress: Water Engineering for a Sustainable Environment, Vancouver, pp. 1179-1186.

Napiorkowski, J.J., Piotrowski, A., Rowinski, P.M., Wallis, S.G., 2008. Prediction of the fate of pollutants in rivers by means of nonlinear Volterra series. River Flow 2008: Proceedings of the International Conference on Fluvial Hydraulics, Çeşme-İzmir, Turkey, 3-5 September, 2469-2476.

Wallis, S.G., Piotrowski, A., Rowinski, P.M., Napiorkowski, J.J., 2007. Prediction of dispersion coefficients in a small stream using artificial neural networks. Proceedings of 32nd IAHR Congress, Venice.

Rowinski, P.M., Guymer, I., Bielonko, A., Napiorkowski, J.J., Pearson, J., Piotrowski, A., 2007. Large scale tracer study of mixing in a natural lowland river. Proceedings of 32nd IAHR Congress, Venice.

Piotrowski, A., Rowinski, P.M., Napiorkowski, J.J., 2006. Assessment of longitudinal dispersion coefficient by means of different neural networks. Proceedings of the 7th International Conference on Hydroinformatics 2006, Nice, France.

Piotrowski, A., Napiorkowski, J.J., 2005. Dispersion coefficient assessment by means of different neural networks. Materiały VIII Krajowej Konferencji Algorytmy Ewolucyjne i Optymalizacja Globalna, Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa.

River runoff forecasting

A number of my first scientific papers were devoted to runoff forecasting in natural rivers. The work included application and comparison of performance of regression methods, a few types of artificial neural networks, phase-space reconstruction models, nearest neighbors approach and Volterra series for rainfall-runoff forecasting or autoregressive runoff modeling. The data were obtained from Nysa Klodzka River (Poland), a small creek located in southern parts of Illinois (USA) and a few rivers located in western Canada.

Published papers:

Piotrowski, A., Napiorkowski, J.J., Rowinski, P.M., 2006. Flash-flood forecasting by means of neural networks and nearest neighbour approach – a comparative study. Nonlinear Processes in Geophysics 13, 443-448.

Napiorkowski, J.J., Piotrowski, A., 2005. Artificial neural networks as an alternative to the Volterra series in rainfall-runoff modeling. Acta Geophysica Polonica, 53(4), 459-472.

Piotrowski, A., Rowinski, P.M., Napiorkowski, J.J., 2004. River flow forecast by selected black box models. River Flow 2004, Ed: M. Greco, A. Carravetta, R. D. Morte, Leiden, Netherlands.

Piotrowski, A., 2003. Porównanie prognoz przepływów rzecznych otrzymanych z modeli przestrzeni fazowej i sieci neuronowych [ang. Comparison of river runoff forecasting by means of phase-space reconstruction models and neural networks]. Współczesne Problemy Hydrauliki Wód Śródlądowych, Materiały XXIII Ogólnopolskiej Szkoły Hydrauliki, Gdańsk.

Extender abstract:

Piotrowski, A., Napiorkowski, J.J., Rowinski, P.M., 2004. Extended phase-space reconstruction technique for the prediction of river flows. Geophysical Research Abstracts, Vol. 6, 07446, 2004;

Reservoir management

I was involved in studies on management of Siemianowka reservoir (located in the north-eastern Poland) in order to improve the water conditions in Narew National Park. My contribution was mainly the help in the choice of the efficient optimization method for Siemianowka reservoir management.

Published paper:

Kiczko, A., Piotrowski, A., Napiorkowski, J.J., Romanowicz, R.J., 2008. Combined reservoir management and flow routing modelling: Upper Narew case study. River Flow 2008: Proceedings of the International Conference on Fluvial Hydraulics, Çeşme-İzmir, Turkey, 3-5 September, 1921-1928.

Geophysical hazards for nuclear objects

In the period of 2011-2012 I took part as a contractor in the Institute of Geophysics, PAS Grant for the young scientists that aimed at improvement of understanding the geophysical hazards for the nuclear objects by young researchers in the Institute of Geophysics.

Grants and Awards

I currently lead two Grants. The first one, Grant for young scientists financed by Institute of Geophysics, aims at selection of optimization methods for catchment runoff forecasting models in moderate climate zones. The second, Iuventus Plus Grant, is financed by Ministry of Science and Higher Education and aims at water temperature prediction in natural rivers by means of empirical models.

Except the grants discussed above I was also a main contractor in the Grant financed by Ministry of Science and Informatization for preparing my PhD thesis.

I was awarded by National Scholarship for young scientists „Start" financed by Foundation for Polish Science in 2008 and its prolongation in 2009. I was also awarded in 2010 by Scholarship founded by prof. Kacper Rybicki for young researchers in Institute of Geophysics, PAS.