Trading Hubs Construction in Electricity Markets Using Evolutionary Algorithms

Pavel A. Borisovsky[1], Anton V. Eremeev[2][**],

Egor B. Grinkevich[3], Sergey A. Klokov[2], and Nikolay A. Kosarev[4]

[1] Omsk F.M. Dostoevsky State University, 55a Prospect Mira, 644077, Omsk, Russia,

[2]Omsk Branch of Sobolev Institute of Mathematics, SB RAS,

13 Pevtsova street, 644043 Omsk, Russia,

[3] Department of Electric Power Industry Development, Ministry of Energy,

42 Schepkin str., Moscow, Russia

[4]Omsk Scientific Center, Siberian Branch of Russian Academy of Sciences,

15/1 Prospect K. Marksa, 644010, Omsk, Russia

borisovski@mail.ru, eremeev@ofim.oscsbras.ru,

grinkevichEB@minenergo.gov.ru, klokov@ofim.oscsbras.ru, nkosarev@mail.ru

[**] Corresponding author. E-mail: eremeev@ofim.oscsbras.ru, tel. +7 3812 236739, fax +7 3812 234584.

Abstract. The trading hubs construction problem for electricity markets under locational marginal prices is considered. Given historical prices for all nodes of the electricity grid and for all market participants over a sufficiently long period of time, the problem is to choose a required number of node clusters (hubs) and to assign market participants to hubs so as to minimize the deviation of hub prices from the prices of participants under certain constraints.

In view of problem complexity, two evolutionary algorithms are proposed: a genetic algorithm and a hybrid local search heuristic. It is proved that the proposed genetic algorithm converges to optimum almost surely. The algorithms are tested and compared on the real-life data. The structure of the fitness landscapes is analyzed using multiple restarts of the local search and the behavior of the evolutionary algorithms is explained on the basis of this analysis.

Key words: Clustering, Genetic algorithm, Local search, Electricity markets, Locational marginal prices, Futures contracts

1 Introduction

In the modern electricity spot markets under locational marginal pricing the electricity price is not unique, it varies from one node of the power grid to another and from one pricing period to another [4, 12]. Here the nodes $i = 1, \ldots, n$ of a power grid correspond to the points of connections of generators or power consumers and network substations to the transmission lines. In some nodes the price may be temporary undefined due to accidents or planned maintenance.

The market participants in this situation are interested in one or several reference prices to hedge the price risks by settling the futures contracts. These reference prices can be calculated by taking the arithmetic average or a weighted average of energy prices over a cluster of nodes with typical price dynamics for some region. A set of cluster nodes $H \subseteq \{1, \ldots, n\}$ with a set of weights α_i, $i \in H$, assigned to them is called a *trading hub* (*hub* for short). An example

of price behavior can be seen in Fig. 1. This figure contains the Real-Time data for two nodes and two hubs of PJM system operator.

If the price is undefined in some nodes of a hub at a certain pricing period, then the hub price is averaged only over the set of the hub nodes where the price is defined (if this set is empty, the hub price is undefined at the problematic time interval). Further details on hedging by means of futures and forward contracts on hub price and some examples of trading hubs in the existing electricity markets can be found e.g. in [2].
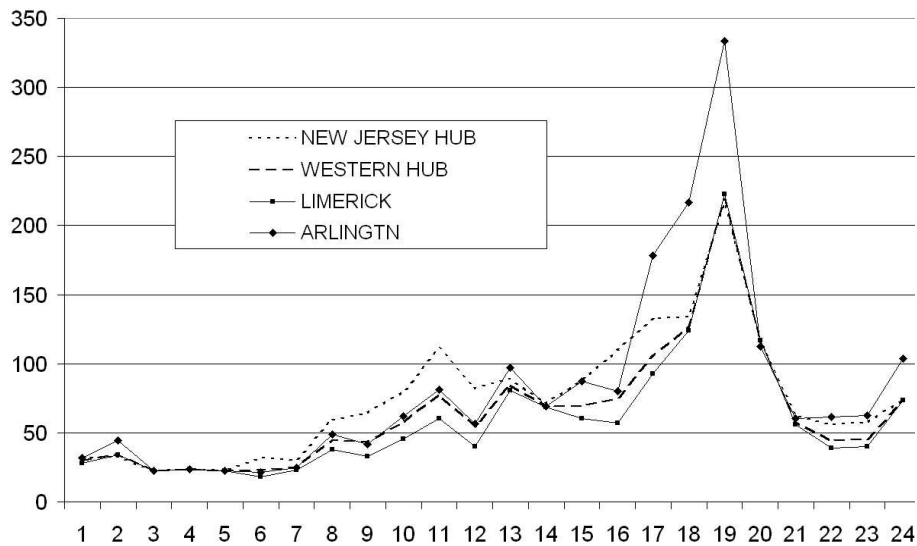
Fig. 1. Real-Time LMP prices at two nodes and two hubs in PJM Interconnection as of 13 Feb 2007. The plot suggests that Western hub is preferable for hedging in node Limerick, while node Arlington might be better hedged in some other hub.

Informally, the Hubs Construction Problem consists in finding a given number of hubs which would approximate as much as possible the price dynamics in connection points of the market participants and respect some additional constraints, depending on variation of

the problem. Since the values of nodal prices in future are unknown, in most practical cases it is acceptable to assume that in future the price dynamics will be similar to that in a preceding sufficiently long historic period (one year or more).

One of the important requirements imposed on the set of hubs in electricity market is the *liquidity* of futures trades in these hubs, which implies that there should be a sufficient number of market participants willing both to sell and to buy futures contracts in each hub. The liquidity requirement is difficult to formalize because this property depends on many non-formalized factors like market structure and trading strategies of market participants, but some well-defined characteristics of hubs may create appropriate conditions for liquid trades. These characteristics include closeness of the hub prices to the prices of many participants, sufficiently small number of hubs, low probability of the event that the hub price is undefined and low market concentration in each of the hubs. The last characteristic can be measured by the Herfindahl-Hirschmann index (see the details in Subsection 2.2). Requirements for these features may be included in the form of constraints in the Hubs Construction Problem.

The electricity grid may contain thousands of nodes, and the problem should be solved on the basis of analysis of historical nodal prices over thousands of pricing periods, which makes it a hard large-scale optimization problem. We suggest further some modifications of the problem which are more adjusted to the practical needs.

In view of complexity of this problem, instead of finding an exact solution we propose and compare two heuristic methods for it: a genetic algorithm and a hybrid local search heuristic combining local search with a simple evolutionary algorithm (1+1)-EA. We have proved that the proposed genetic algorithm converges to optimum almost surely. The algorithms are tested and compared on the real-

life data. The structure of the fitness landscapes is analyzed using multiple restarts of the local search and the behavior of the evolutionary algorithms is explained on the basis of this analysis.

The paper is organized as follows. In Section 2, we formally state the problem and its modifications. The proposed algorithms are described in Section 3 and the computational results are provided in Section 4. Section 5 contains the conclusions.

2   Background and Problem Formulation

A market participant may possess one or several nodes of the grid and trade electricity at spot market in all of these nodes. In fact, the set of nodes of a participant may consist of several *groups of connection points*, where each group contains the nodes of a separate industrial enterprise or power plant or a populated area. When a participant trades electricity at the wholesale market, any contract of this participant applies to a certain group of connection points rather than separate nodes independently.

The electricity price of a group of connection points is calculated on the basis of the nodal prices as a weighted average of the nodal prices. Let R be the set of market participants, G be the set of all groups of connection points and let G^r denote the set of groups of connection points belonging to a participant r. For any $g \in G$, a vector $(p_{g1}, \ldots, p_{gT})$ of prices of this group of connection points over the historical period $1, \ldots, T$ is given. Besides this, a vector of nodal prices $(c_{i1}, \ldots, c_{iT})$ is provided for each of the nodes $i = 1, \ldots, n$. The pricing periods $1, \ldots, T$ are called *hours* in what follows (usually in practice they equal to 1 hour).

The number of required hubs m is given (usually m is not large, up to 10). Parameter m should be chosen, keeping in mind that on one hand, too many hubs will disperse the market participants into small

groups attracted to different hubs which will be of little liquidity therefore; on the other hand, too few hubs will poorly approximate the prices of some participants and they may loose interest in trading futures in hubs.

To indicate different significance of different groups of connection points, the weighting factors w_g, $g \in G$, are introduced. In this paper, we assume that the weights are equal to the average traded volumes $w_g = W_g^{\text{gen}} + W_g^{\text{con}}$. Here and below W_g^{gen} is the average amount of power sold and W_g^{con} is the average amount of power bought in a group of connection points g at the spot market in the given historical period. Alternatively, one can assign a set of weights e.g. equal to the installed capacity in the group of connection points. Note that the algorithms proposed in this paper can be extended to a more general setting, where the weighting factors are defined variable in time.

For simplicity of the models in this paper we assume that the nodal prices are defined for all hours.

Let us assume that the measure of closeness between the price C_{jt} of a hub j, $j = 1, \ldots, m$, and the price $p_{gt}, g \in G$ is

$$d(g, j) = \sum_{t=1}^{T} (p_{gt} - C_{jt})^2 w_g.$$

2.1 Groups of Connection Points Coincide with Nodes

Originally the Hubs Construction Problem was formulated by W. Hogan, A. Hartshorn and S. Chang [7] for the special case where each group of connection points consists of a single node, i.e. $G = \{1, \ldots, n\}$ and $p_{it} = c_{it}$, $i = 1, \ldots, n$:

$$\text{Min} \quad D = \sum_{i=1}^{n} \sum_{j=1}^{m} \delta_i^j \sum_{t=1}^{T} (p_{it} - C_{jt})^2 w_i \tag{1}$$

s.t.

$$C_{jt} = \sum_{i=1}^{n} \alpha_i^j c_{it}, \ t = 1, \dots, T, \ j = 1, \dots, m, \tag{2}$$

$$\sum_{i=1}^{n} \alpha_i^j = 1, \ j = 1, \dots, m, \tag{3}$$

$$\sum_{j=1}^{m} \delta_i^j = 1, \ i = 1, \dots, n, \tag{4}$$

$$\delta_i^j \in \{0, 1\}, \quad \alpha_i^j \geq 0, \ i = 1, \dots, n, \ j = 1, \dots, m. \tag{5}$$

Here the *assignment variables* δ_i^j, $i = 1, \dots, n$, $j = 1, \dots, m$, relate the nodes to hubs. The variables α_i^j, $i = 1, \dots, n$, define the set of weights within the hub j, while the variables C_{jt}, $t = 1, \dots, T$, give the price of hub j, $j = 1, \dots, m$, at each hour t. Equation (2) defines the hub price calculation. Equation (4) ensures that each node is assigned to exactly one of the hubs. The constraint (3) is not an indispensable part of the model, but it plays the role of a valid cut, as it can be seen in the proof of Proposition 1 below. The optimization criterion (1) is based on the assumption that the measure of closeness between the hub price C_{jt} and the price $p_{it}, i = 1, \dots, n$, is the sum of squared differences between p_{it} and C_{jt}.

As it is noted by W. Hogan, once the set of all variables δ_i^j is fixed, the remaining variables may be found by solving a convex optimization problem; sometimes they may be assigned explicitly as we will see in the proof of Proposition 1 below.

Naturally, in an optimal solution any node i should be assigned to a hub with minimal value $d(i, j) = \sum_{t=1}^{T} (p_{it} - C_{jt})^2 w_i$ among the available m hubs, so $\delta_i^j = 1$ will hold for a hub chosen this way. Note that in a feasible solution, for one or several hubs j' may hold $\sum_{i=1}^{n} \delta_i^{j'} = 0$. Such assignments are called *degenerate*.

The following proposition is aimed at finding the best-possible set of real-valued variables $\{\alpha_i^j\}$, complementing a non-degenerate assignment of nodes.

Proposition 1 (see [2]). *Suppose, a feasible non-degenerate assignment $\{\delta_i^j\}$ is given. Then the optimal price in hub j, $j = 1, \ldots, m$, can be calculated as the weighted average of prices in the assigned nodes:*

$$C_{jt} = \sum_{i:\ \delta_i^j = 1} w_i c_{it} \bigg/ \sum_{i:\ \delta_i^j = 1} w_i. \tag{6}$$

Proof. Denote by F_{jt} the weighted total deviation of nodal prices within hub j from the hub index at hour t.

$$F_{jt} = \sum_{i=1}^{n} \delta_i^j (c_{it} - C_{jt})^2 w_i.$$

To find the hub price C_{jt} minimizing F_{jt} we differentiate it over C_{jt} and set to zero:

$$\frac{\partial F_{jt}}{\partial C_{jt}} = -\sum_{i=1}^{n} \delta_i^j 2(c_{it} - C_{jt}) w_i = 0. \tag{7}$$

Solving this equation, we obtain

$$C_{jt} = \sum_{i=1}^{n} \delta_i^j w_i c_{it} \bigg/ \sum_{i=1}^{n} \delta_i^j w_i.$$

Define $\alpha_i^j = w_i / \sum_{k=1}^{n} \delta_k^j w_k$ if $\delta_i^j = 1$ and $\alpha_i^j = 0$ otherwise. It is easy to check that $\sum_{i=1}^{n} \alpha_i^j = 1$, so the obtained solution is feasible and hence it is optimal due to (7). Q.E.D.

If $n \geq m$, then it is always possible to eliminate degeneracy of an assignment, not increasing the objective function value. This can be done iteratively by finding a hub j' for which more than one node is assigned – let i' be one of these nodes – and assigning i to a

degenerate hub j''. The variables $\{\alpha_i^{j'}\}$ are chosen according to (6) so as to minimize $d(i, j')$ for all nodes i assigned to hub j'. To see that the objective function value does not increase, note that the term of objective function associated with the node i' will now turn to zero, and the sum of terms associated with nodes assigned to hub j' is now minimal for this set of nodes according to Proposition 1. In view of this observation, in what follows we can consider only non-degenerate solutions.

In conditions of Proposition 1, hub j, $j = 1, \ldots, m$, is completely defined by the set of nodes assigned to it:

$$H_j = \{i : \delta_i^j = 1, \ i = 1, \ldots, n\},$$

since the coefficients α_i^j are given by

$$\alpha_i^j = w_i \Big/ \sum_{i:\ \delta_i^j = 1} w_i.$$

If the weights w_i are all identical, then the problem turns into the Minimum Sum-of-Squares Clustering Problem: find a partition of a given finite set of vectors in Euclidean space into several disjoint sets (clusters), minimizing sum of squared distances from each element to the centroid of its cluster. Here centroid means the simple average of vectors in a cluster. The Minimum Sum-of-Squares Clustering Problem is NP-hard [1] for the number of clusters $m = 2$, so the same applies to the Hubs Construction Problem (1)–(5).

Application of model (1)–(5) to the real-life data of Midwest Independent System Operator MWISO (USA) indicated that a modern commercial nonlinear continuous optimization package is unable to obtain a practically suitable solution using this model [7] due to high dimensionality of the problem and multiplicity of local optima. Therefore the hubs for MWISO were chosen using the well-known h-means clustering heuristic [11] neglecting the weights $\{w_i\}$.

In [2] we proposed a two-stage approach to the hubs construction, where at the first stage a given number of hubs are constructed using clustering heuristics and at the second stage the obtained clusters are refined using exact mathematical programming methods or evolutionary algorithms. These methods proved to be applicable to the large scale real-life instances but they do not respect some important practical constraints discussed in the next subsection.

The special case when groups of connection points coincide with nodes considered above is not quite applicable to the energy market of Russia. In this market, the participants trade electricity trough a simultaneous auction where each bid is associated with purchase or selling electricity in a group of connection points consisting of one or several nodes. This is why we proceed to the general case below.

2.2 Proposed Problem Formulation

Original formulation of the Hubs Construction Problem (1)–(5) lacks some practical constraints which are usually taken into account when the hubs are being constructed. These requirements are listed below.

The size of a hub. The nodal prices are defined only for the nodes which are not switched off in the electricity system. However many nodes require maintenance from time to time, or may be switched off due to an accident. In this situation, the more nodes constitute a set of hub nodes H_j, the more stable and predictable the hub price C_{jt} can be made, and the smaller is the chance that C_{jt} will become undefined in some hour. A simple assumption that all nodes of the system have equal probability of outage implies that a minimal admissible hub size $n_{\min}$ should be given. In practice the hubs usually consist of tens or hundreds of nodes (see e.g. [8]).

If the hub price is computed as a weighted average over the hub nodes, then outage of a node with high weight may have a significant

impact on the hub price. Therefore, the definition of a hub price as the arithmetic average over the hub nodes is more acceptable than the weighted hub price definition (2). The arithmetic average is also the simplest and the most understandable for the market participants.

Competition level at a hub. An important parameter of market competitiveness is the market concentration, measured by the Herfindahl-Hirschmann Index (HHI), which is the sum of squared market shares in percents. High values indicate that the market is divided amongst few market participants. HHI values greater than 1800 are considered highly concentrated [9].

Given a family of hubs, we will say that a group of connection points g belongs to the *area of preference* $G_{j'} \subseteq G$ of a hub j' if $d(g, j')$ is minimal over all $j = 1, \ldots, m$. In what follows, it is assumed that $G_1, \ldots, G_m$ constitute a partition of set G.

The HHI indices are defined both for generation and consumption as follows:

$$I_j^{\text{gen}} = \sum_{r \in R} \left(\frac{100 \cdot \sum_{g \in G_j \cap G^r} W_g^{\text{gen}}}{\sum_{g \in G_j} W_g^{\text{gen}}} \right)^2 , \ j = 1, \ldots, m, \qquad (8)$$

$$I_j^{\text{con}} = \sum_{r \in R} \left(\frac{100 \cdot \sum_{g \in G_j \cap G^r} W_g^{\text{con}}}{\sum_{g \in G_j} W_g^{\text{con}}} \right)^2 , \ j = 1, \ldots, m. \qquad (9)$$

Problem formulation. The Hubs Construction Problem proposed in this paper takes into account the practical requirements mentioned above. In particular, in what follows we will always assume that the hub price C_{jt} is calculated as an arithmetic average of the nodal prices, i.e. it is completely defined by the set of nodes H_j and the set of nodal prices $c_{it}, \ i \in H$. The mathematical programming model of the problem is as follows:

$$\text{Min} \sum_{j=1}^{m} \sum_{g \in G} \delta_g^j \sum_{t=1}^{T} (p_{gt} - C_{jt})^2 w_g \tag{10}$$

s.t.

$$C_{jt} = \sum_{i=1}^{n} x_i^j c_{it} \Big/ \sum_{i=1}^{n} x_i^j, \ t = 1, \dots, T, \ j = 1, \dots, m, \tag{11}$$

$$\sum_{j=1}^{m} \delta_g^j = 1, \ g \in G, \tag{12}$$

$$\sum_{j=1}^{m} x_i^j \le 1, \ i = 1, \dots, n, \tag{13}$$

$$\sum_{i=1}^{n} x_i^j \ge n_{\min}, \ j = 1, \dots, m, \tag{14}$$

$$\sum_{r \in R} \left(\frac{100 \cdot \sum_{g \in G^r} \delta_g^j W_g^{\text{gen}}}{\sum_{g \in G} \delta_g^j W_g^{\text{gen}}} \right)^2 \le I_{\max}, \ j = 1, \dots, m, \tag{15}$$

$$\sum_{r \in R} \left(\frac{100 \cdot \sum_{g \in G^r} \delta_g^j W_g^{\text{con}}}{\sum_{g \in G} \delta_g^j W_g^{\text{con}}} \right)^2 \le I_{\max}, \ j = 1, \dots, m, \tag{16}$$

$$\sum_{j=1}^{m} \delta_g^j \sum_{t=1}^{T} (p_{gt} - C_{jt})^2 \le \sum_{t=1}^{T} (p_{gt} - C_{kt})^2, \ g \in G, \ k = 1, \dots, m, \tag{17}$$

$$\delta_g^j \in \{0, 1\}, \ g \in G, \ j = 1, \dots, m, \tag{18}$$

$$x_i^j \in \{0, 1\}, \ i = 1, \dots, n, \ j = 1, \dots, m. \tag{19}$$

Here the variables δ_g^j and C_{jt} have a similar meaning as in problem (1)–(5), with the only difference that now δ_j^i assigns a group of connection points g rather than a separate node. The indicator variables x_i^j, $i = 1, \dots, n$ define the set of vertices of hub j, i.e.

$H_j = \{i : x_i^j = 1\}$. Equation (11) defines the hub price as an arithmetic average over nodal prices in H_j. Equation (12) ensures each group of connection points is assigned to exactly one of the hubs. Inequality (13) implies that each node belongs at most to one hub. Condition (14) imposes a lower bound on a hub size, while (15) and (16) impose the bounds on market concentration index HHI. Practitioners propose to set $I_{\max} = 1800$ [9]. The constraint (17) forces the variables δ_g^j to assign each group of connection points g to the closest hub.

Note that due to the form of the objective function (10), constraint (17) would be redundant in the model if there were no relationships (15) and (16), which also involve the variables δ_g^j. Given a family of hubs $H_1, \ldots, H_m$, in view of (17), the objective (10) may be re-written as

$$f(H_1, \ldots, H_m) = \sum_{g \in G} \left(w_g \cdot \min_{j=1,\ldots,m} \sum_{t=1}^{T} \left(p_{gt} - \frac{\sum_{i \in H_j} c_{it}}{|H_j|} \right)^2 \right).$$

The Hubs Construction Problem (10)–(19) is NP-hard when the number of hubs is equal to 2 even if we exclude the constraints on hub size and on HHI indices by setting $n_{\min} = 1$ and $I_{\max} = 10^4$ and suppose that all groups of connection points are nodes, $G = \{1, \ldots, n\}$. Indeed, in this case the problem becomes equivalent to the minimum Sum-of-Squares Clustering Problem with 2 clusters, which is NP-hard [1].

In the special case where the number of hubs equals one it was shown in [2] that if the weights w_g are substituted with time-dependent weights w_{gt}, this problem is NP-hard.

2.3 Alternative Distance Measure Based on Linear Regression

The value $d(i,j) = \sum_{t=1}^{T}(p_{it} - C_{jt})^2$ measures the squared distance from node i to hub j. However, there is an alternative way to compare node prices (p_{it}) and hub prices (C_{jt}). In some electricity markets the participants are interested in high correlation between the nodal prices and the hub prices rather than in close values of p_{it} and C_{jt}. In presence of significant correlation, one can forecast the nodal prices with the help of a linear model.

In the simplest case, the linear model predicts the node price p_{it} by $a + bC_{jt}$, where a and b are real numbers fixed for all hours t. Thus, one may use the value

$$\tilde{d}(i,j) = \min_{a,b} \sum_{t=1}^{T}(p_{it} - a - bC_{jt})^2 \tag{20}$$

instead of $d(i,j)$.

Taking partial derivatives with respect to a and b and setting them to zero, we deduce the formulas for computing the values

$$a^* = \frac{\sum_{t=1}^{T} p_{it} - b^* \sum_{t=1}^{T} C_{jt}}{T},$$

$$b^* = \frac{T \sum_{t=1}^{T} p_{it}C_{jt} - \sum_{t=1}^{T} p_{it} \sum_{t=1}^{T} C_{jt}}{T \sum_{t=1}^{T} C_{jt}^2 - \left(\sum_{t=1}^{T} C_{jt}\right)^2}.$$

Note that substituting the obtained expression of a^* into (20) we get

$$\tilde{d}(i,j) = \sum_{t=1}^{T}\left(p_{it} - \frac{1}{T}\sum_{\tau=1}^{T} p_{i\tau} + b^* \cdot \frac{1}{T}\sum_{\tau=1}^{T} C_{j\tau} - b^*C_{jt}\right)^2$$

$$= \sum_{t=1}^{T}\left(p_{it} - b^*C_{jt} - \frac{1}{T}\sum_{\tau=1}^{T}(p_{i\tau} - b^*C_{j\tau})\right)^2 = (T-1)\cdot\hat{\sigma}^2(p_{it} - b^*C_{jt}),$$

where $\hat{\sigma}(\cdot)$ denotes the maximum likelihood estimate of a standard deviation. It is known (see e.g. [2, 18]) that a market participant i, that uses the so-called *optimal hedge ratio* strategy for hedging in hub j and producing or consuming one unit of electricity power, has the standard deviation of the financial result (i.e. the amount received minus amount payed at spot and futures markets) equal to $\min_h \sigma(p_{it} - hC_{jt})$. Therefore, the above choice of parameters a^* and b^* is equivalent to the optimal hedge ratio assumption. The value $\tilde{d}(i, j)$ in this case is the maximum likelihood estimate for standard deviation of financial result of participant i using hub j, multiplied by a constant factor $(T - 1)$.

Conditions (10) and (17) in the problem's formulation should be rewritten accordingly,

$$\text{Min} \sum_{j=1}^{m} \sum_{g \in G} \delta_g^j \sum_{t=1}^{T} (p_{gt} - a_{gj} - b_{gj}C_{jt})^2 w_g, \tag{21}$$

$$\sum_{j=1}^{m} \delta_g^j \sum_{t=1}^{T} (p_{gt} - a_{gj} - b_{gj}C_{jt})^2 \leq$$

$$\sum_{t=1}^{T} (p_{gt} - a_{gk} - b_{gk}C_{kt})^2, \quad g \in G, \ k = 1, \ldots, m, \tag{22}$$

where the variables a_{gj}, b_{gj} are the coefficients of linear regression for the price at a group of connection points g, $g \in G$ being a regressand and the price of hub j, $j \in 1, \ldots, m$ being a regressor.

3   Algorithms Description

3.1   Genetic Algorithm

The genetic algorithm (GA) is a random search method that models a process of evolving a population of encoded tentative solutions [10].

Encodings of solutions in the GA implementation are usually called *genotypes*. Each genotype ξ corresponds to some solution $\mathbf{x}(\xi)$ to the problem (feasible or maybe infeasible) and it is characterized by the *fitness* $F(\xi)$, reflecting the objective function value and satisfaction of problem constraints. Each genotype in a population may be selected as a parent, and its chances to be selected depend on the fitness value.

New genotypes are built by means of *crossover* and *mutation* procedures Cross and Mut. Cross produces the offspring from two parent genotypes by combining their elements. Mut adds small random changes to a genotype.

Let a family Π of N solutions be the current population of the GA. The proposed GA is based on the steady state replacement scheme [3, 15]:

Steady state GA
1. Generate the initial population Π at random and set $\theta := 1$.
2. While a termination condition is not satisfied, do:
    2.1 Selection: choose parent genotypes ξ, η from Π.
    2.2 Produce a child ζ applying the crossover: $\zeta := \mathrm{Cross}(\xi, \eta)$.
    2.3 Apply mutation: $\zeta' := \mathrm{Mut}(\zeta)$.
    2.4 Choose one genotype in Π and replace it by ζ'.
    2.5 Increment the iterations counter $\theta := \theta + 1$.
3. Output the best found solution.

In our implementation of the GA, the choice of each parent on Step 2.1 is done by the *s*-tournament selection: choose s genotypes from Π at random with uniform distribution and select a one with minimal fitness among them.

Let the genotype ξ be a vector $(\xi_1, \ldots, \xi_n)$, where $\xi_i, i = 1, \ldots, n$, is the number of the hub to which the node i is assigned, i.e. $i \in H_j(\xi)$ iff $\xi_i = j$. Some nodes may not belong to any hub at all, which is expressed by setting ξ_i to zero. The family of hubs $(H_1(\xi), \ldots, H_m(\xi))$ defined by ξ is denoted by $\mathbf{x}(\xi)$.

Given a genotype ξ, the set of vales of all variables δ_g^j is defined by choosing a closest hub for each $g \in G$. The rare cases, when ties are present, may be resolved by enumerating all admissible assignments and choosing the most appropriate one w.r.t. conditions (15) and (16).

If a genotype ξ encodes a feasible solution, then its fitness $F(\xi)$ is set equal to the objective function value $f(\mathbf{x}(\xi))$. If a genotype ξ encodes an infeasible solution, then the objective value is given an additive penalty term as described below. Let us first define the following components for each hub $H_j(\xi)$:

- The number of "missing" nodes is $d_j = \max\{0, n_{\min} - |H_j(\xi)|\}$.
- HHI violation for generators is measured by $D_j^{\text{gen}} = \max\{0, I_j^{\text{gen}} - I_{\max}\}$, where I_j^{gen} is defined as in (8).
- HHI violation for consumers is $D_j^{\text{con}} = \max\{0, I_j^{\text{con}} - I_{\max}\}$, where I_j^{con} is defined as in (9).

Then the fitness in the general case is defined by

$$F(\xi) = f(\mathbf{x}(\xi)) +$$

$$C \cdot \sum_{j=1}^{m} \left(\text{sgn}(d_j) + \text{sgn}(D_j^{\text{gen}}) + \text{sgn}(D_j^{\text{con}}) + \frac{d_j}{n_{\max}} + \frac{D_j^{\text{gen}} + D_j^{\text{con}}}{D_{\max}} \right).$$

Here we use $\text{sgn}(\cdot)$ for the signum function, which equals 1 for positive numbers, -1 for negative numbers and 0 for zero argument; the constant C is the maximum squared nodal price over the whole system through the historic period, multiplied by $T|G|$, i.e.

$$C = T|G| \max\{c_{it}^2 : t = 1, \ldots, T, \ i = 1, \ldots, n\};$$

$D_{\max} = 10^4 - I_{\max}$ is the maximum possible value of D_j^{gen} and D_j^{con}. It is easy to see that for any genotype, which is encoding an infeasible solution, the fitness function $F(\cdot)$ gives a greater value than the objective value of any feasible solution.

The algorithm starts its work with a randomly constructed population of genotypes, where each node is assigned to one of the clusters with uniform probability distribution. In Step 2.4, the genotype with maximal fitness is chosen for deletion. The final solution returned by the GA is decoded from the best found genotype in terms of the fitness.

In view of the fact that the same set of hubs can be encoded differently just by relabelling the hubs, first of all in the crossover operator the two parent genotypes are *aligned* so that the hubs with the same numbers have the least symmetric difference (as sets of nodes). Formally, here a permutation $\{\pi_1, \ldots, \pi_m\}$ is sought, minimizing the function $\sum_{j=1}^{m} H_j^{\xi} \, \Delta \, H_{\pi_j}^{\eta}$, where H_j^{ξ} and H_k^{η} are the sets of nodes of hubs number j and k in parent genotypes ξ and η, respectively. The alignment step in the crossover is similar to that in [14], but since the number of hubs in our problem instances is small, we solve the assignment problems using the complete enumeration.

The crossover operator is implemented using the standard uniform crossover: the hub number for each node is chosen randomly from the two alternatives provided for this node in the aligned parent genotypes. The obtained genotype undergoes mutation, where each node with mutation probability p_{mut} is moved into another hub, chosen randomly with uniform distribution.

The fitness function evaluations are costly for large scale instances of the problem. To speed up the GA, instead of the exact fitness function $F(\cdot)$ we use its approximation $F'(\cdot)$, which is computed using only a random subset $S \subseteq \{1, \ldots, T\}$ of pricing periods. The

subset S is chosen with uniform distribution among all $K(\theta)$-element subsets of $\{1, \ldots, T\}$, where the subset size $K(\theta)$ approaches T in the process of GA computation, i.e. $\lim_{\theta \to \infty} K(\theta) = T$. So, at the beginning, the GA works with a rough approximation of the true fitness function, which nevertheless allows to significantly improve a population quality without a heavy computational burden. As the search continues, the approximation becomes more accurate and the GA becomes able to distinguish solutions of higher quality.

Convergence of the GA. Let $\hat{\Pi}_\theta$ denote the set of genotypes of population Π at iteration θ of the GA. We will say that the GA converges to optimum almost surely (a.s.), if

$$\min_{\xi \in \hat{\Pi}_\theta} F(\xi) \overset{\theta \to \infty}{\longrightarrow} f^* \quad \text{with probability 1,}$$

where f^* is the optimal value of objective (10).

Proposition 2. *Suppose $0 < p_{\mathrm{mut}} < 1$ and the termination condition is never met, then the GA defined above converges to optimum a.s.*

   Proof. Note that the optimal solution can always be represented by some genotype ξ^*. Besides that, application of mutation and crossover operators to any pair of parent genotypes produces the genotype ξ^* with probability at least $(p_{\mathrm{mut}}/m)^n$ if $p_{\mathrm{mut}} \le 0.5$, or at least $((1 - p_{\mathrm{mut}})/m)^n$ if $p_{\mathrm{mut}} \ge 0.5$. Therefore, if $0 < p_{\mathrm{mut}} < 1$ then the probability of producing the genotype ξ^* at any iteration is bounded below by some $\varepsilon > 0$, which does not depend on the iteration number θ.

   Since $\lim_{\theta \to \infty} K(\theta) = T$, after a finite number of iterations θ_0, the evaluations of $F'(\cdot)$ will always give the true fitness $F(\cdot)$. A finite number of iterations does not influence on convergence of the GA, so in what follows, w.l.o.g. we will assume that $\theta_0 = 0$.

Now one can derive the required statement by Corollary 2.1 [17], but for the sake of completeness we provide a straightforward proof. Let $A(\theta)$ denote the event that no optimum is visited on iterations $0, 1, \ldots, \theta$. Then $P\{A(\theta)\} \leq (1 - \varepsilon)^\theta$ and by properties of probability measure,

$$P\left\{\bigcap_{\vartheta=0}^{\infty} A(\theta)\right\} = \lim_{\theta \to \infty} P\left\{A(\theta)\right\} \leq \lim_{\theta \to \infty} (1 - \varepsilon)^\theta = 0.$$

Therefore, with probability 1 within a finite number of iterations an optimal genotype ξ^* will be found at least once. In view of assumption $\theta_0 = 0$, the outline of the steady state GA implies that, after visiting ξ^* once, the GA will always keep a genotype with optimal fitness value. Q.E.D.

Parallel implementation. The proposed GA was adjusted for multiprocessor workstations by running several independent threads (one per processor), each one executing Steps 2.1 – 2.5 of the GA scheme. The only synchronization bottleneck is the population update, however the idle processor time becomes significant only if the number of threads is large. One more thread (the master) manages the algorithm parameters, regenerates subsets of pricing periods and finally terminates the GA.

3.2 The Hybrid Local Search Heuristic

The hybrid local search heuristic (HLS) proposed here is a combination of a local search algorithm and a (1+1)-EA evolutionary algorithm.

The Local Search Algorithm (LS) is a modification of the well-known K-Means clustering heuristic (see e.g. [6, 13]) tailored for the Hubs Construction Problem. The local search starts from an initial genotype, which is encoding a family of hubs $H_1, \ldots, H_m$ and iteratively

moves a node from one hub to another aiming to decrease the value of fitness function $F(\cdot)$. At each iteration, we loop through all nodes, and check if moving the current node to a different hub decreases the fitness function value. The best of such moves is applied to the node. The iterations are performed until a local optimum is reached or the counter of evaluated genotypes reaches a given upper bound $\theta_{\max}^{LS}$. This algorithm may be restarted a number of times from randomly chosen initial genotypes.

The (1+1)-EA evolutionary algorithm is based on a different definition of move from one genotype to another. At each iteration, one current genotype is available. Two nodes are randomly chosen from two different hubs and if exchanging these nodes improves the objective function, then the new genotype becomes the current one. Otherwise, the current genotype remains unchanged. The algorithm continues until the counter of evaluated genotypes reaches the limit $\theta_{\max}^{EA}$.

The Hybrid Local Search Algorithm (HLS). In the hybrid algorithm, both the local search and the (1+1)-EA are applied iteratively to a population of genotypes so that the most promising genotypes receive more computation time than the ones of poor quality. The strategy used here is as follows: initially $N^{(0)}$ genotypes are generated randomly and the local search is applied to each of them. All of the obtained genotypes are used as starting points for the (1+1)-EA. When the (1+1)-EA terminates on each of these genotypes, the best $N^{(1)} = \lfloor N^{(0)}/2 \rfloor$ outcomes among $N^{(0)}$ results are identified and the main iteration is repeated, i.e. the best $N^{(2)} = \lfloor N^{(1)}/2 \rfloor$ genotypes are chosen and improved by the LS and the (1+1)-EA. The process continues until the population size reaches 0. If the termination condition is given in terms of CPU time, the rest of the

time is used for (1+1)-EA iterations working with the best found genotype. The corresponding solution is returned as the final result.
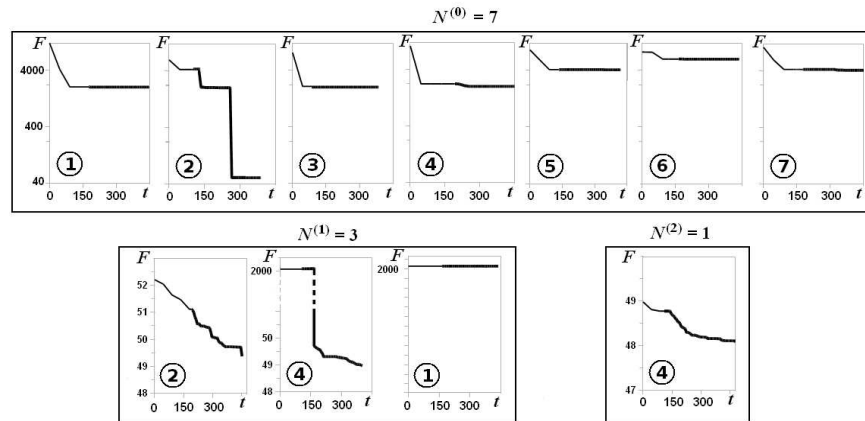
Fig. 2. Search process of the HLS on problem S' with $m = 3$.

The search process of the HLS is illustrated in Fig. 2 (the results were obtained when HLS was applied to problem S' with $m = 3$, see the details in Section 4). The figure shows the dynamics of the fitness function F over the running time (in seconds). At the first stage, $N^{(0)} = 7$ genotypes are generated randomly and both the LS and the $(1 + 1)$-EA are applied. The thin line corresponds to the LS; the $(1 + 1)$-EA is shown by the thick line. The only feasible solution is found in the attempt number 2. The $N^{(1)} = \lfloor N^{(0)}/2 \rfloor = 3$ best genotypes numbered 2, 4, and 1 are chosen for the next stage, where the 4-th genotype is improved. This improved genotype also represents a feasible solution and turns out to be the best one. It is further improved on the final stage with $N^{(2)} = 1$ and the process finishes.

4 Computational Experiments

GA, LS, (1+1)-EA and HLS algorithms were implemented in Java using JDK 6.20 compiler and tested on 8-core Xeon based on E5420 QuadCore 2.5 GHz CPUs. The heuristics were compared on the real-life data from the day-ahead spot market of United Energy System of Russia with about 200 market participants, 1425 groups of connection points in European zone of Russia (*European problems* for short) and 216 groups of connection points in Siberian zone of Russia (*Siberian problems* for short). The pricing period is one hour and the prices are computed for more than 6000 nodes in Europe and more than 500 nodes in Siberia. On the basis of preliminary analysis the set of nodes, that may constitute hubs, was reduced to 1161 in European problems and 325 in the Siberian problems. The historical period of 3 years since year 2007 was considered.

A total of 12 instances were constructed by taking the full 3-year period in each zone (problems E' and S') or taking just year 2008 or 2009 with $T = 8760$ (problems E8, E9, S8 and S9). For Europe $n_{\min} = 100$, $m = 3$ or 4; for Siberia $n_{\min} = 80$, $m = 2$ or 3.

4.1 Implementation Details

The pure local search LS and the (1+1)-EA were tested in multi-processor mode with randomly generated initial genotypes. For each problem instance, 8 independent threads, one per each CPU core, were performed with the same time limit indicated below. Within one run the multi-start approach was used for the LS: when the LS reached a local optimum it was restarted until the given CPU time was used up. The (1+1)-EA simply continued its iterations within the given time limit.

The proposed GA was tested in its multi-processor version. Preliminary computations showed that the quality of obtained solutions

varies significantly in different GA runs so it was decided to use a multistart version of the algorithm. The final experiments were made with 4 independent GA runs, each one using 2 working threads for population update. After one third of the total time elapsed, only two best runs continue their execution occupying 4 threads each.

Every 15 seconds the best genotype of the current population in terms of approximate fitness $F'(\cdot)$ is chosen. Let $\xi_1^*, \xi_2^*, \ldots$ denote the sequence of these genotypes and let $\theta_1, \theta_2, \ldots$ be the corresponding iteration numbers. Only for the genotypes of the sequence $\xi_1^*, \xi_2^*, \ldots$, the exact fitness $F(\cdot)$ is calculated using the full set of pricing periods $\{1, \ldots, T\}$.

The algorithm begins with $K(0) = 250$ pricing periods and increases $K(\theta_i)$ by 250 whenever one of the following conditions is satisfied:

- The approximate fitness has been improved, i.e. $F'(\xi_i^*) < F'(\xi_{i-1}^*)$ but the exact fitness is increasing during the latest five 15-second intervals: $F(\xi_i^*) > F(\xi_{i-1}^*) > \ldots > F(\xi_{i-4}^*)$. This indicates an effect of "overlearning" on the current set of pricing periods.
- The fitness $F(\xi_i^*)$ is penalized (the minimal hub size or the bound on HHI is violated) whereas the approximate fitness at the latest five 15-second intervals $F'(\xi_i^*), F'(\xi_{i-1}^*), \ldots, F'(\xi_{i-4}^*)$ has not been penalized.
- There were no improvements of $F(\xi_i^*)$ at the last 10000 iterations of the GA.

The GA parameters are: $N = 100, s = 5, p_{mut} = 2/n$.

4.2 Computational Results

The HLS was ran independently on each of the 8 CPU cores and the best outcome was taken. For problems E', E8 and E9 we set

$N^{(0)} = 2$, $\theta^{EA}_{\max} = 3 \cdot 10^3$, $\theta^{LS}_{\max} = 3n$. For problems S', S8 and S9 we have $N^{(0)} = 7$, $\theta^{EA}_{\max} = 4 \cdot 10^4$, $\theta^{LS}_{\max} = 4n$.

All three algorithms were given equal amount of running time: 4 hours for problem E'; 2 hours for S'; 1 hour for problems E8, E9 and 30 min for problems S8, S9. The results obtained by the algorithms are reported in Table 1. Here the values of objective function $f(\mathbf{x})$ are transformed to the cost scale by the function $\sqrt{\frac{f(\mathbf{x})}{T|G|}}$. The best-known results are indicated in bold. On problem S8 the LS and (1+1)-EA did not obtain a feasible solution in any of the runs, this is indicated in Table 1 by the symbol "-".

Table 1. Results of the (1+1)-EA, LS, GA and the HLS

	E'	E8	E9	E'	E8	E9	S'	S8	S9	S'	S8	S9
m	3	3	3	4	4	4	2	2	2	3	3	3
LS	57.88	54.17	55.51	54.66	52.02	**52.8**	44.36	47.16	52.07	50.57	-	56.49
(1+1)-EA	62.99	58.33	62.93	60.74	55.21	60.06	45.61	47.61	53.64	50.07	-	56.52
GA	58.04	54.14	55.45	56.36	52.17	53.21	44.47	**47.04**	**51.94**	**47.12**	**47.65**	**53.45**
HLS	**57.78**	**54.09**	**51.45**	**54.51**	**51.23**	55.29	**44.19**	47.10	51.96	47.52	48.49	54.84

From the tests with the LS algorithm, we observed that for the European problems the given time limit was not enough to reach the local optima. For the largest problem E' the LS made only 13 iterations ($13n$ evaluations of the fitness function) if $m = 3$ and 9 iterations if $m = 4$. For problems E8 and E9, the number of iterations was 25 if $m = 3$ and 15 if $m = 4$. A local optimum in the European problems was obtained only for E9 when $m = 4$. On the contrary, for the Siberian problems, the LS reached local optima many times. For problem S', about 50 local optima were found when $m = 2$ and about 40 when $m = 3$ on each CPU core. For S8 and S9, each CPU core produced about 30 local optima both in the case $m = 2$ and in the case $m = 3$. The number of iterations made by the LS in each run varied from 3 to 7.

The number of iterations made by the (1+1)-EA for the European problems was quite large: approximately 25 000 for problem E', and 50 000 for E8 and E9. For the Siberian problems it was even higher: 60 000 for S', and about 90 000 for S8 and S9. Note that in many cases, the improvements were observed even on the last iterations but they were negligibly small.

The parameters $N^{(0)}$, $\theta_{\max}^{EA}$, and $\theta_{\max}$ for the HLS were chosen so that the running time of the algorithm would fit the given time limit. According to the outline of the HLS (see Section 3), in case the algorithm has finished earlier, the rest of available time is used for iterations of the (1+1)-EA. Otherwise the computation is stopped and the best found solution is returned. For the European problems, the main iterations of the HLS usually took from 85% to 100% of the available time, and so fitted well to the time limits. On the Siberian problems, the main HLS stage was much faster and usually consumed about 70% of the given time.

In most of the cases the improvements of fitness function continued throughout the whole running time of the GA and HLS. The use of fitness function $F'(\cdot)$ based on a reduced set of pricing periods, instead of $F(\cdot)$, improved the performance of the GA. The same applies to alignment of genotypes in the crossover operator.

An additional experiment with the GA was carried out, where instead of the uniform crossover, the classical single-point crossover [10] was used. The other parameters of the algorithm remained unchanged. In all instances, except for two (E', $m = 3$ and E8, $m = 3$), the GA with uniform crossover outperformed the GA with single-point crossover. This comparison supports our original choice of the uniform crossover.

Number of local optima. One of the measures of the difficulty a problem presents for the local search methods is the number of the local

optima ν, induced by the neighborhood in use. The true number of the local optima in our case can only be found by complete enumeration, but the required computation time is prohibitive. Statistical estimates for ν can be obtained using the methods proposed in [5, 16]. Here we evaluate the lower bound on ν using the results of pure local search on problems S', S8 and S9 with $m = 2$ and 3 w.r.t. fitness function F on the domain of all genotypes. In all 150 independent restarts the local search returned different local optima, so a lower bound on ν with confidence level 95% can be computed as $150^2/(-2 \ln 0.05) = 3755$, see Section 2.2 in [5]. The maximum likelihood estimate based on Schnabel census approach [16] gives $\hat{\nu} = 1.074 \cdot 10^9$. The problems E', E8 and E9 were not considered in this respect because the time required to reach a local optimum is prohibitive. These problems have greater dimensions n and m and we conjecture that they have even greater numbers of local optima.

Summary of the experiments. The estimates of the local optima numbers indicate that the Hubs Construction Problem presents significant obstacles for solving it to optimality by multiple restarts of local search method, even in the small problem instances. This might be a reason of relatively weak LS performance on the Siberian problems. The HLS and the GA demonstrate better performance due to their capacity to escape the local optima where the LS is trapped.

The HLS shows advantage on the largest, most difficult problems, while the GA is better on the smaller ones. This complementary behavior might be due to slow convergence rate of the GA on the problems with "deep" local optima. Note that by Proposition 2, the GA eventually finds an optimum after a finite number of iterations with probability 1, even though the CPU time may be practically unacceptable. The European problems are easier for the HLS which suitably combines two types of local moves and gives more CPU time

to those local search processes which have better current solutions. For the Siberian problems it is not so important to move fast towards the closest local optimum (note that the number of steps made in each run of the LS was from 3 to 7), rather it is important to locate the areas of the search space, where the solutions are more promising.

Practical Usage. The proposed algorithms were applied using the distance measure based on linear regression (see Subsection 2.3) to define the current set of hubs "Center", "South", "Ural", "Western Siberia" and "Eastern Siberia" for the market of United Energy System of Russia.

Fig. 3 and Fig. 4 show the correlation coefficients between the hub prices and the weighted average electricity prices in the administrative regions which actively participate in the electricity market. The regional average hourly electricity prices are calculated as a weighted average over all nodes of a region with wight equal to the absolute values of total consumption and/or generation of power in a node. The hue shows the value of correlation coefficient. For each region, the closest hub selection is indicated by the slant of hatching. As it can be seen from the figures, the central region of European Russia and Kranoyarsk region have high correlation with the closest hub price. This is due to relatively large amounts of trading and similar nodal price dynamics within these regions.

The price indices of these hubs are regularly published on the site of Administrator of Trade System for United Energy System of Russia (http://www.atsenergo.ru) and used for settlement of futures contracts at Moscow Energy Exchange.
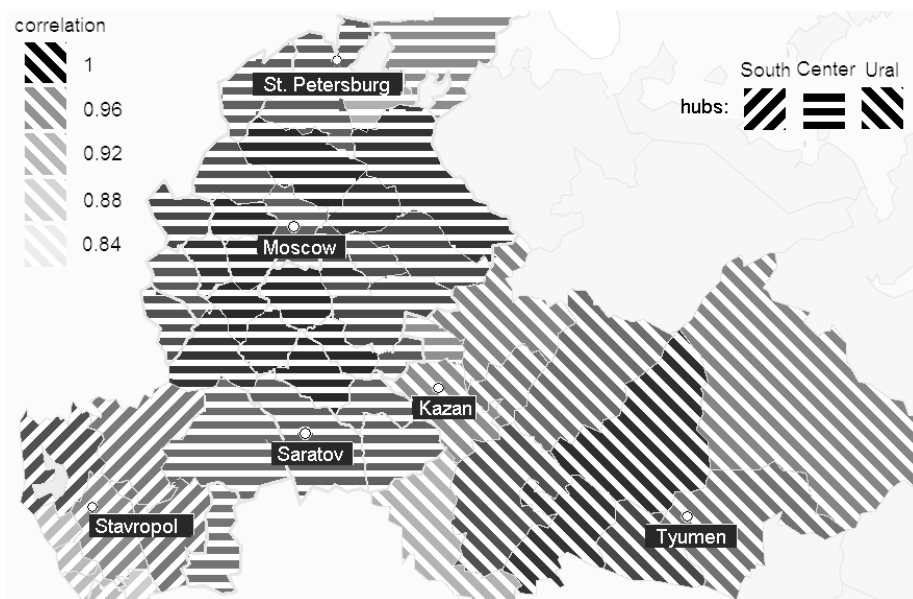
Fig. 3. Correlation coefficients between the average regional prices in the European zone and the closest of the three hub prices: "South", "Center" and "Ural" (left to right).

Fig. 4. Correlation coefficients between the average regional prices in the Siberian zone and the closest of the two hub prices: "Western Siberia" and "Eastern Siberia" (left to right).

5 Conclusions

In this paper, we formulated a new modification of the Hubs Construction Problem and proposed two evolutionary algorithms, the Genetic Algorithm and the Hybrid Local Search using the (1+1)-EA for solving it. Both algorithms were tested on a multi-processor workstation and demonstrated complementary behavior. From the practical point of view, the developed algorithms can serve as a useful decision-support tool in trading hubs construction for the electricity wholesale markets. Further research should be aimed at combining the strong points of the two algorithms in one metaheuristic.

References

1. Aloise D., Deshpande A., Hansen P., and Popat P. NP-hardness of Euclidean sum-of-squares clustering. Machine Learning, Vol. 75, N 2, 2009, pp. 245–248.
2. Borisovsky P.A., Eremeev A.V., Grinkevich E.B., Klokov S.A. and Vinnikov A.V. Trading hubs construction for electricity markets. In: Kallrath, J., Pardalos, P.M., Rebennack, S., Scheidt, M. (eds.) Optimization in the Energy Industry, Berlin Heidelberg, Springer, 2009, pp. 29–58.
3. De Jong K.A. An analysis of the behaviour of a class of genetic adaptive systems. Doctoral dissertation, Ann Arbor, MI, University of Michigan, 1975.
4. Davidson M.R., Dogadushkina Yu.V., Kreines E.M., Novikova N.M., Seleznev A.V., Udaltsov Yu.A. and Shiryaeva L.V. Mathematical model of power system management in conditions of a competitive wholesale electric power (capacity) market in Russia. Journal of Computer and System Sciences International, Vol. 48, N 3, 2009, pp. 243-253.
5. Eremeev A.V., Reeves C.R. On confidence intervals for the number of local optima. In: Raidl, G.R., Meyer, J.-A., Middendorf, M., Cagnoni, S., Cardalda, J.J.R., Corne, D., Gottlieb, J., Guillot, A., Hart, E., Johnson, C.G., Marchiori, E. (eds) Applications of Evolutionary Computing, EvoWorkshop 2003: EvoBIO, EvoCOP,

EvoIASP, EvoMUSART, EvoROB, and EvoSTIM. LNCS, Vol. 2611, Heidelberg, Springer, 2003, pp. 224–235.

6. Hansen P., Mladenović N. J-means: A new local search heuristic for minimum sum-of-squares clustering. Pattern Recognition, Vol. 34, 2001, pp. 405–413.

7. Hartshorn A., Chang S. MWISO hubs development, Technical report. Cambridge, MA, LECG, 2003.

8. Harvey S.M., Hartshorn A.P., Bertagnolli D. and Kowalski R. Derivation of a trading hub in the New England Market, Holyoke, NEPOOL Joint CMS/MSS Group, 1999.

9. Hausman E., Fagan R., White D., Takahashi K., Napoleon A. LMP electricity markets: Market operations, market power, and value for consumers. Synapse Energy Economics. 2006.

10. Holland J. Adaptation in natural and artificial systems, Ann Arbor, MI, University of Michigan Press, 1975.

11. Howard R. Classifying a population into homogeneous groups. In: Lawrence J.R. (ed.) Operational research in the social sciences. London, Tavistock Publ. 1966.

12. Schweppe F.C., Caramanis M.C., Tabors R.D., Bohn R.E. Spot pricing in electricity. Norwell, MA, Kluwer Acad. Pbs., 1988.

13. MacQueen J. Some methods for classification and analysis of multivariate observations. In: Lucien Le Cam, M., Neyman, J. (eds.) 5-th Berkeley Symposium on Mathematical Statistics and Probability, Vol. 2, Berkeley, Calif., University of California Press, 1967, pp. 281–297.

14. Moraglio A., Kim Y.-H., Yoon Y., Moon B.-R. Geometric crossovers for multiway graph partitioning, Evolutionary Computation, Vol. 15, N 4, 2007, pp. 445–474.

15. Reeves C.R. Genetic algorithms for the operations researcher. INFORMS Journal on Computing, Vol. 9, N 3, 1997, pp. 231–250.

16. Reeves C.R., Eremeev A.V. Statistical analysis of local search landscapes, Journal of the Operational Research Society, Vol. 55, N 7, 2004, pp. 687–693.

17. Rudolph G. Finite markov chain results in evolutionary computation: A tour d'horizon, Fundamental Informaticae, Vol. 35, N 1–4, 1998, pp. 67–89.

18. Shawky H.A., Marathe A., Barrett C.L. A first look at the empirical relation between spot and futures electricity prices in the United States, Journal of Futures Markets, Vol. 23, 2003, pp. 931–955.