Efficient energy dispatching in smart microgrids using an integration of fuzzy AHP and TOPSIS assisted by linear programming

Beatrice Lazzerini, Francesco Pistolesi

Dipartimento di Ingegneria dell'Informazione, University of Pisa, Largo Lucio Lazzarino 1, 56122 Pisa - Italy

Abstract

Energy dispatching in smart (micro)grids must take into account more conflicting objectives (or criteria), such as power reliability and quality, proper handling of the electricity demand, and cost decrease. The choice of the best alternative in energy dispatching decisions can be dealt with as a multi-criteria optimization and decision making problem. To this aim, we propose the use of linear programming to generate the possible alternatives, and the integration of fuzzy AHP and TOPSIS to select the best alternative. In particular, fuzzy AHP and TOPSIS are used, respectively, to prioritize the criteria and to evaluate the alternatives with respect to four conflicting criteria, namely, environmental impact, cost of the energy, distance of supply, and load level of power lines.

1. Introduction

Contrary to the traditional electricity network, which passively carries energy from few large power plants to several small and medium consumers, a new electric grid, called *smart grid*, is getting more and more important. A smart grid allows intelligent integration of all connected users (namely, producers, consumers or prosumers) in order to distribute energy in an efficient, sustainable and secure way. The following major aspects characterize a smart grid:

- power generation sources of any technology and dimension can be connected to the grid. The distributed generation of electricity from many small energy sources provides increased efficiency of energy generation and higher security of power supply. Furthermore, the greater and greater use of renewable energy resources, such as wind, sunlight, biomass and geothermal heat, helps minimize carbon emission;
- the consumers have more information and tools that help them participate in the electricity enterprise; in practice, being energy aware, customers can consume as well as produce energy (so-called prosumers).

Nowadays the idea of distributed energy generation is stressed by an emerging concept: the so-called *microgrid* [6]. A microgrid is a small scale energy system consisting of electricity sources, energy storage, and loads. A microgrid can work in two different ways: in grid connected mode (i.e., connected to the traditional electricity network) or in islanded mode (i.e., isolated from the larger power network so that it functions autonomously) [3]. In a microgrid, generation and loads are typically interconnected at low or medium voltage. The capability to island distributed generators and loads together has the potential to improve local supply reliability and demand stability. Yet a microgrid can still be seen as a single entity connected to the main grid through the transmission and distribution system: at any given time, from the perspective of the main grid, the microgrid will be either a consumer or a producer. This implies that microgrids can be considered as the building blocks of a wider power grid [5][8][13]. In other words, the smart grid can be modeled as a hierarchical structure in which two-way flows of electricity and information travel between the high-voltage network and smart microgrids at different hierarchical levels.

It follows that the main goal of a smart grid (or smart microgrid) is to optimize the integration of all the users connected to the grid (or microgrid) from several points of view, such as enhancement of power reliability, security and quality, proper handling of the electricity demand, reduction of greenhouse gas emissions, improvement of the use of renewable energy sources, and cost decrease. This can be regarded as a multi-criteria optimization and decision making problem, whose solution requires the following two steps:

1. *solution generation*: in this step, as there are more conflicting objectives to be optimized simultaneously, more possible solutions (or alternatives) will be generated;
2. *solution selection*: in this step, the possible solutions must be compared with respect to different, typically conflicting, goals (or criteria).

In this paper we propose an approach to optimize energy dispatching in smart (micro)grids based on linear programming for the solution generation phase, and the integration of fuzzy AHP and TOPSIS for the solution selection phase. In particular, fuzzy AHP and TOPSIS are used, respectively, to prioritize the criteria and to evaluate the alternatives.

As a preliminary step, we model the smart (micro)grid as a directed graph (which is called *graph-grid*) consisting of a set of nodes and arcs. For the sake of simplicity, we consider a reference (micro)grid architecture consisting of a radial electrical system with one or more

$$
P = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_n \end{array}
\begin{pmatrix}
\begin{array}{cccc} C_1 & C_2 & \ldots & C_m \end{array} \\
\begin{pmatrix}
p_{11} & p_{12} & \ldots & p_{1m} \\
p_{21} & p_{22} & \ldots & p_{2m} \\
\vdots & \vdots & \ddots & \vdots \\
p_{n1} & p_{n2} & \ldots & p_{nm}
\end{pmatrix}
\end{pmatrix}
$$

Figure 1: A decision matrix with n alternatives and m criteria.

feeders and a set of loads. The system is connected to the higher-level distribution system through a separation device. A node can be either a producer, a consumer, a prosumer, or a dispatcher. Each type of node is characterized by given attributes; in particular, producers and consumers, respectively, produce and require electricity, while prosumers may either produce or consume electricity; finally, a dispatcher aims to dispatch the power produced by producers to consumers based on appropriate efficiency criteria. All nodes in a graph-grid are connected to a dispatcher node. In this way, a lowest-level microgrid is modeled as a graph-grid including one producer node for each electricity source, one consumer node for each load, one prosumer node for each prosumer or energy storage, and one dispatcher node. On the other hand, a higher level (micro)grid is represented by a graph-grid that may include, besides the previous nodes, also prosumer nodes modeling lower-level microgrids connected to the (micro)grid under analysis.

2. Multi-criteria decision making

A multi-criteria decision making (MCDM) problem is characterized by a goal, a set of criteria and a set of alternatives. Criteria and alternatives are called *elements*. The goal consists in finding the best alternative with respect to all the criteria.

Most MCDM methods associate a weight with each criterion: usually weights are normalized to add up to one. Weights can be chosen by the expert or can result from a specific ranking technique.

An MCDM problem is usually described using a decision matrix. A decision matrix P is an $n \times m$ matrix, where n is the number of alternatives and m is the number of criteria, as shown in figure 1. Each element p_{ij} of P evaluates the performance of alternative A_i with respect to criterion C_j, where $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$.

2.1. Analytic Hierarchy Process (AHP)

Analythic hierarchy process (AHP) [9][10] decomposes a complex decision making problem into sub-problems which are simpler to solve. Since criteria might be divided into sub-criteria and sub-criteria into sub-subcriteria and so on, we will refer to a lowest-level sub-criterion as a *lowest sub-criterion*. AHP organizes the problem as a hierarchy whose uppermost level contains the goal, intermediate levels contain criteria, sub-

criteria, etc., and the lowest level contains the alternatives.

AHP ranks criteria with respect to each other, with reference to their parent in the hierarchy. Alternatives are ranked according to each lowest sub-criterion. AHP generates a scale of priorities derived from pairwise comparisons. In practice, AHP firstly gives a structure to a complex problem and then let the decision maker understand which criteria are more important than others and which is the best alternative.

2.1.1. Description of the method

AHP requires to build a pairwise comparison matrix for each level of the hierarchy, by comparing elements sharing the same parent. Given two elements i and j the

Preference weight a_{ij}	Explaination
1	Equally preferred
3	Moderately preferred
5	Strongly preferred
7	Very Strongly preferred
9	Extremely preferred
2, 4, 6, 8	Intermediate values (compromises)

Table 1: Saaty's scale of preference.

result of a pairwise comparison is a coefficient m_{ij} estimating the preference of i over j. Coefficient values, called *preference weights*, are usually expressed by using Saaty's scale of preference which connects qualitative judgements to the first nine integer numbers, as shown in Table 1. If m_{ij} is the generic element of a comparison matrix M, then $m_{ii} = 1$ and $m_{ij} = 1/m_{ji}$ for each $i, j = 1, 2, \ldots, n$. A local weight w_i^{local} expresses the importance of an element i (either a criterion or an alternative) with respect to the others sharing the same parent in the hierarchy.

A matrix A is said to be *consistent* if $a_{ij} = a_{ik}a_{kj}$, where a_{ij} is the generic element of A and $i, j, k = 1, 2, \ldots n$. It has been proved [11] that the principal eigenvector is a representation of the priorities derived from a positive reciprocal pairwise comparison judgement matrix.

In order to obtain *global weights*, i.e., the weight of each element with respect to its uppermost ancestor in the hierarchy, the local weight of each element is multiplied by the ones related to its ancestors in the hierarchy, until the uppermost level is reached. The results of these products are subsequently summed. Global weights of the elements in the lowest level of the hierarchy, i.e, the alternatives, represent the result of the decision making process. The decisional problem is solved by choosing the alternative having the greatest global weight.

2.1.2. Consistency check

Once local weights have been determined, we have to check if they reflect the expert's judgements, obtained by pairwise comparisons. In other words, given an $n \times n$ pairwise comparison matrix A we need to verify how

much the ratios $\frac{w_i^{local}}{w_j^{local}}$ derived from its principal eigenvector differ from the estimates a_{ij} with which the expert has filled the matrix A. In order to perform this check, AHP computes a *consistency index* (CI) which expresses an overall difference between the values a_{ij} and $\frac{w_i^{local}}{w_j^{local}}$, for each $i, j = 1, 2, \ldots, n$.

Let λ_{max} be the principal eigenvalue of A. The consistency index is defined as: $CI \triangleq \frac{\lambda_{max} - n}{n - 1}$. Whenever the matrix A is consistent its principal eigenvalue λ_{max} is equal to n. Therefore, in the case of perfect consis-

n	3	4	5	6	7	8	9
RI	0.5245	0.8815	1.1086	1.2479	1.3417	1.4056	1.4499

Table 2: Random Index values obtained from 100000 n-order matrices.

tency CI is equal to zero. As inconsistency increases, CI becomes greater. AHP compares CI with a *random index* (RI) which is obtained by calculating the mean of the consistency indexes of many reciprocal pairwise comparison matrices of the same order of A, whose elements are randomly generated according to a uniform probability distribution. This comparison is performed by computing the *consistency ratio* $CR = \frac{CI}{RI}$. AHP considers a CR greater than 0.1 as unacceptable. If $CR > 0.1$, the expert has to increase the coherence of his judgements by changing the entire set of estimates a_{ij} (or a part of it) until CR becomes lower than or equal to 0.1. Table 2 shows the random indexes obtained from 100000 n-order matrices.

2.2. Fuzzy AHP

2.2.1. Fuzzy sets and fuzzy numbers

Human judgements are usually affected by imprecision. Fuzzy sets deal with these situations [12]. The main concept of fuzzy set theory is that an element x has a *degree of membership* $\mu(x) \in [0, 1]$ in a fuzzy set [7][14]. A fuzzy number is a fuzzy set. We will use triangular fuzzy numbers. Given $l, m, u \in \mathbb{R}$ such that $l \leq m \leq u$, a triangular fuzzy number is a fuzzy set $\mathsf{A} = \{(x, \mu(x))\}$, where $x \in \mathbb{R}$ and $\mu : \mathbb{R} \to [0, 1]$. The membership function $\mu(x)$ is:

$$\mu(x) = \begin{cases} 0 & \text{if} \quad x < l \vee x > u \\ \frac{x-l}{m-l} & \text{if} \quad l \leq x \leq m \\ \frac{u-x}{u-m} & \text{if} \quad m \leq x \leq u \end{cases}.$$

A triangular fuzzy number is indicated with $\tilde{\mathsf{A}} = \{l, m, u\}$. A further representation is based on the confidence level, i.e.,

$$\tilde{\mathsf{A}}_\alpha = [l^\alpha, u^\alpha] = [(m - l)\alpha + l, -(u - m)\alpha + u], \quad (1)$$

for each $\alpha \in [0, 1]$. Given a fuzzy number (or, in general, a fuzzy set) the set $\{x \mid \mu(x) > 0\}$ is called *support*.

2.2.2. Fuzzy AHP procedure

Fuzzy AHP substitutes the Saaty's scale of preference with a fuzzified version. Judgements are expressed by

triangular fuzzy numbers from $\tilde{1}$ to $\tilde{9}$ and the interval arithmetic is used to solve the fuzzy eigenvector [1].

More rigorously, let $\alpha \in (0, 1]$, let $\tilde{A}$ be an $n \times n$ matrix containing triangular fuzzy numbers $\tilde{a}_{ij}^\alpha = [a_{ij_l}^\alpha, a_{ij_u}^\alpha]$ and let $\tilde{x}$ be a non-zero $n \times 1$ vector containing fuzzy numbers $\tilde{x}_i = [x_{i_l}^\alpha, x_{i_u}^\alpha]$. A fuzzy eigenvalue $\tilde{\lambda}$ is a fuzzy number solution to $\tilde{A}\tilde{x} = \tilde{\lambda}\tilde{x}$. In order to compute the eigenvector of $\tilde{A}$ connected with the principal fuzzy eigenvalue $\tilde{\lambda}$, fuzzy AHP defuzzifies $\tilde{A}$. Defuzzification is a process which maps a fuzzy set into a number. There are several ways to perform defuzzification. Fuzzy AHP defuzzifies the matrix $\tilde{A}$ by introducing a coefficient $\zeta \in [0, 1]$, called *index of optimism*, with which it performs a convex combination of each element of $\tilde{A}$, obtaining elements

$$\hat{a}_{ij}^\alpha = \zeta a_{ij u}^\alpha + (1 - \zeta) a_{ijl}^\alpha. \quad (2)$$

After defuzzifying the elements of $\tilde{A}$, matrix $\hat{A} = [\hat{a}_{ij}^\alpha]$ is obtained. The eigenvector associated with the principal eigenvalue of $\hat{A}$ is calculated as in classic AHP.

2.3. Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS)

2.3.1. Overview

The technique for order of preference by similarity to ideal solution (TOPSIS) is a multi-criteria decision making approach [2]. Consider a decisional problem characterized by n alternatives and m criteria. Let $\mathbf{A} = \{1, 2, \ldots, n\}$ be the set of indexes of the alternatives and let $\mathbf{C} = \{1, 2, \ldots, m\}$ be the set of indexes of the criteria. TOPSIS needs an $n \times m$ decisional matrix whose generic row i contains the performance the expert has estimated for the alternatives A_i with respect to each criterion C_j, where $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$. In addition, TOPSIS requires a vector containing the weights the expert has assigned to the criteria. Differently from AHP, in TOPSIS the weights of the criteria and the weights of the alternatives with respect to each criterion are directly expressed by the expert. In order to select the best alternative, TOPSIS computes the ideal best and worst solutions to the problem: we will refer to them as the *ideal solution* and the *negative-ideal solution*, respectively. The ideal solution and the negative-ideal solution are characterized, respectively, by the best and the worst performance attainable by all the alternatives with respect to each criterion. TOPSIS compares the alternatives with each other by computing the euclidean distance of each alternative from the ideal and negative-ideal solutions. The alternative to be chosen must have the shortest distance from the ideal solution and the farthest distance from the negative-ideal solution.

2.3.2. Description of the method

The expert builds an $n \times m$ decision matrix P such that n is the number of alternatives and m is the number of criteria, as shown in figure 1, where A_i is the i-th alternative, C_j is the j-th criterion, and p_{ij} is the expert's

judgement of the performance of the i-th alternative with respect to the j-th criterion.

Criteria are divided into *cost* criteria and *benefit* criteria. With reference to the benefit criterion C_j the higher the element p_{ij}, the greater the preference the expert assigns to alternative A_i over the others. On the other hand, considering the cost criterion C_k, the lower the element p_{ik} the higher the preference the expert expresses for A_i. Criteria may not have the same importance, so TOPSIS considers a further vector $w^T = (w_1, w_2, \ldots, w_m)$ containing the weights of the criteria such that $\sum_{j=1}^{m} w_j = 1$.

By using the previous arguments, TOPSIS performs the following steps:

Step 1: construct the normalized decision matrix R.
The generic element r_{ij} of R is: $r_{ij} = \frac{p_{ij}}{\sqrt{\sum_{i=1}^{n} p_{ij}^2}}$.

Step 2: construct the weighted normalized decision matrix V.
The generic element v_{ij} of V is: $v_{ij} = w_j r_{ij}$.

Step 3: determine the *ideal* and the *negative-ideal* solutions.
The two ideal alternatives (i.e., best and worst) A^+ and A^- are determined. Let $\mathcal{J}$ be the set of indexes of the benefit criteria, and let $\mathcal{J}'$ be the set of indexes of the cost criteria. A^+ and A^- are composed by the elements:

$$v_j^+ = \begin{cases} \max_{i \in \mathbf{A}, j \in \mathcal{J}} v_{ij} \\ \min_{i \in \mathbf{A}, j \in \mathcal{J}'} v_{ij} \end{cases} \quad v_j^- = \begin{cases} \min_{i \in \mathbf{A}, j \in \mathcal{J}} v_{ij} \\ \max_{i \in \mathbf{A}, j \in \mathcal{J}'} v_{ij}. \end{cases}$$

Step 4: calculate the separation measure.
Given an alternative A_i, $i = 1, 2, \ldots, n$, TOPSIS measures the separations of A_i from A^+ and A^-, indicated respectively with δ_i^+ and δ_i^- by computing the euclidean distance, as follows: $\delta_i^+ = \sqrt{\sum_{j=1}^{m} (v_{ij} - v_j^+)^2}$, $\delta_i^- = \sqrt{\sum_{j=1}^{m} (v_{ij} - v_j^-)^2}$

Step 5: calculate the relative closeness to the ideal solution. For each alternative A_i, TOPSIS computes the so-called closeness coefficient χ_i^+, defined as: $\chi_i^+ = \frac{\delta_i^-}{\delta_i^+ + \delta_i^-}$. An alternative A_i is closer to A^+ as χ_i^+ approaches to 1.

Step 6: give a rank to the alternatives.
In this last step alternatives are ranked, according to the descending order of χ_i^+.

The alternative A_k such that $k = \arg\max_i \chi_i^+$ is the best.

3. Fundamentals of linear programming

A linear programming (LP) problem consists in minimizing or maximizing a linear function, called *objective function*, subject to linear constraints. A point that satisfies all the constraints is called a *feasible point* and the set of all feasible points forms the *feasible region*,

which geometrically represents a polytope described by the constraints of the problem. The objective function $z : \mathbb{R}^n \to \mathbb{R}$ of an LP problem is a linear combination of the decision variables x_j whose coefficients are c_j, where $j = 1, 2, \ldots n$. Therefore, the objective function is $z = \sum_{j=1}^{n} c_j x_j$.

Since mimimizing the objective function z is equivalent to maximizing $-z$, from now on we will discuss minimization problems, without loss of generality.

Constraints are equations or inequalities of the form $\sum_{j=1}^{n} a_{ij} x_j \lesseqgtr b_i$, generated by linear combinations of the decision variables, whose coefficients are expressed by an $m \times n$ matrix A.

An LP problem can be written in different equivalent forms through algebraic manipulations. An LP problem is said to be in standard form if all the constraints are expressed by equations and all the variables are non-negative. Equation $\sum_{j=1}^{n} a_{ij} x_j = b_i$ is the i-th constraint, a_{ij} are the coefficients, and constraints $x_j \geq 0$ are the non-negativity constraints, one for each variable.

The fundamental theorem of LP states that if an LP problem whose feasible region is a polytope has a solution, then it will occur in a vertex of the polytope, or on a line segment connecting two vertices.

Many techniques have been developed for evaluating the objective function value only in a subset of all the vertices: one of these approaches is the simplex method [4].

4. Efficient energy dispatching

Consider a microgrid represented by a directed graph having n nodes and m arcs. Nodes can be active or passive. A node is active (passive) if it provides (absorbs) energy. The sets of active and passive nodes will be indicated, respectively, with $\mathcal{O}$ and $\mathcal{E}$, where $\mathcal{O}, \mathcal{E} \subset \mathcal{N}$ and $\mathcal{O} \cap \mathcal{E} = \emptyset$.

A producer p is an active node with the following attributes:

- *distance*: is the distance $d_p \in \mathbb{R}^+$ from the dispatcher node which the producer is connected to, expressed in kilometers;
- *power*: is the suppliable electrical power, measured in kW, represented by a negative integer;
- *type of source*: is the way by which the producer generates energy, e.g., fossil fuels, etc.;
- *environmental impact*: is the amount of pollutants (e.g., carbon dioxide, dioxins, etc.) released into the environment. The impact is supposed to be $\pi_p \in \Pi$, where, e.g., $\Pi = \{1, 2, 3, 4, 5\}$ contains increasing levels of contaminants emitted with energy production;
- *cost*: is the cost $k_p \in \mathbb{R}^+$ per kWh of energy produced.

A consumer c is a passive node with the following attributes:

- *distance*: as previously defined;

- *power*: is the power required by the consumer, measured in kW. It is represented by a positive integer or zero.

A prosumer s is a node which produces or consumes electricity. At any instant, a prosumer is active if it can supply electricity, otherwise it is passive. A prosumer has all attributes of producers and consumers. As a prosumer produces using renewable sources, we assume a zero environmental impact.

A dispatcher is a node which neither consumes nor produces energy. Its task is dispatching the power available in the active nodes to the passive nodes, according to criteria of efficiency. The set containing the dispatchers is indicated with $\mathcal{D}$. Dispatchers are connected to each other through a node *CC* which is called the *central controller*. Like the dispatchers, the central controller neither produces nor consumes energy.

Each producer or consumer is connected to a dispatcher $D_i \in \mathcal{D}$, $i \in \{1, 2, \ldots, |\mathcal{D}|\}$, through an arc. In the case of a producer (consumer), the arc which connects it to a dispatcher has its head in the dispatcher (consumer) and its tail in the producer (dispatcher). Further, each prosumer is connected to a dispatcher through a pair of arcs: one directed to the dispatcher, the other directed to the prosumer. Finally, each dispatcher is connected to the central controller via a pair of arcs: one directed from the dispatcher to the central controller, the other directed in the opposite direction. Arcs are de-
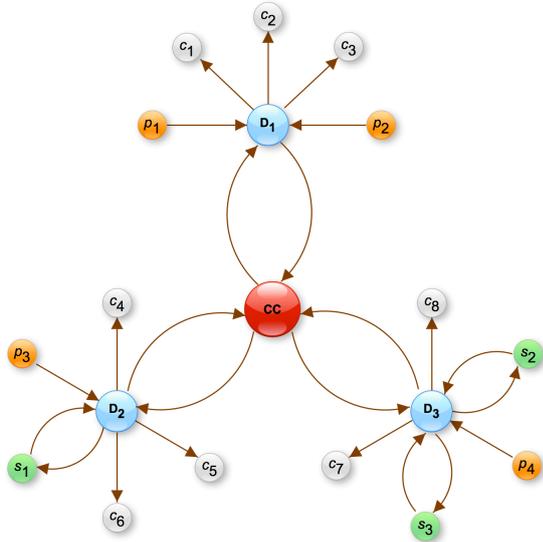

Figure 2: An example of a microgrid.

noted by ordered pairs. Given two nodes $i, j \in \mathcal{N}$, the arc whose head is j and whose tail is i is (i, j). Let $\mathcal{A}_1$ be the set containing all the possible arcs connecting a dispatcher to the central controller and let $\mathcal{A}_2$ be the set containing all the possible arcs connecting a dispatcher to an active or passive node. The set $\mathcal{A}$ containing the arcs of the microgrid is: $\mathcal{A} \subseteq (\mathcal{A}_1 \cup \mathcal{A}_2)$.

In compact notation, a microgrid is $(\mathcal{N}, \mathcal{A})$. Given a configuration of energy flow on the arcs of the grid, the optimization process of dispatching energy considers the following criteria:

- environmental impact;
- cost of the energy;
- distance of supply;
- load level of the power lines.

The problem is addressed through an MCDM approach using a hybrid method based on fuzzy AHP and TOPSIS: fuzzy AHP assigns the weights to each criterion and TOPSIS selects the best alternative. In our model, an alternative is a flow, i.e., a given configuration of energy flows on the arcs of the graph.

Of course, it is important to start with a reasonable number of good alternatives. To this aim, we use linear programming.

5. Strategy for generating alternatives

5.1. Mathematical model

Consider a microgrid $(\mathcal{N}, \mathcal{A})$. Given a node $i \in \mathcal{N}$ let $\mathcal{N}^+(i) = \{j \in \mathcal{N} \mid \exists (i, j) \in \mathcal{A}\}$ be the set of its successors and $\mathcal{N}^-(i) = \{j \in \mathcal{N} \mid \exists (j, i) \in \mathcal{A}\}$ the set of its predecessors. The arcs of the graph can be described by an $n \times m$ incidence matrix E, whose generic element e_{ih} is:

$$e_{ih} = \begin{cases} -1 & \text{if } h \text{ is an outgoing arc from the node } i \\ 1 & \text{if } h \text{ is an incoming arc to the node } i \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Each node $i \in \mathcal{N}$ is characterized by an energy requirement $b_i \in \mathbb{Z}$ measured in kW, which is the power required (or suppliable) by the node. The central controller and the dispatchers have a zero energy requirement.

Consider an active node $i \in \mathcal{O}$ connected to a dispatcher D_t, where $t \in \{1, 2, \ldots, |\mathcal{D}|\}$. For the sake of simplicity, in the following we will indicate with D the dispatcher D_t. The arc $(i, D) \in \mathcal{A}$ which connects i to the dispatcher is characterized by a distance $d_{iD} \in \mathbb{R}^+$ from the dispatcher and an energy cost $k_{iD} \in \mathbb{R}^+$ for each kWh provided by i. In addition, the arc (i, D) is associated with minimum and maximum capacities φ_{iD}^{min} and φ_{iD}^{max} of sustainable energy flow, where φ_{iD}^{min}, $\varphi_{iD}^{max} \in \mathbb{Z}^+$. On the contrary, if i is passive, the arc $(D, i) \in \mathcal{A}$ has no energy cost, i.e., $k_{Di} = 0$, a minimum and maximum capacities φ_{Di}^{min} and φ_{Di}^{max} of sustainable energy flow, where $\varphi_{Di}^{min}, \varphi_{Di}^{max} \in \mathbb{Z}^+$.

To obtain alternatives, we generate sub-optimal solutions of an LP problem. Consider the objective function *aggregate cost* $c : \mathbb{R}_+^2 \to [0, \gamma]$ to minimize, defined as

$$c(d_{ij}, k_{ij}) = \gamma \left(\alpha \frac{d_{ij} - d_{min}}{d_{max}} + \beta \frac{k_{ij} - k_{min}}{k_{max}} \right), \tag{4}$$

where $(i, j) \in \mathcal{A}$, $\alpha \in [0, 1]$, $\beta = 1 - \alpha$ and $\gamma \in \mathbb{Z}^+$. k_{min} and k_{max} are, respectively, the minimum and maximum energy costs; d_{min} and d_{max} are, respectively, the minimum and maximum distances of all the nodes from the dispatcher.

For each arc $(i, j) \in \mathcal{A}$, let $x_{ij} \in \mathbb{Z}^+$ be the energy flow on the arc, and let φ_{ij}^{min} and φ_{ij}^{max} be, respectively, the minimum and maximum flows of the arc

(i, j). Let $x^T \in \mathbb{Z}_+^m$ be the vector containing a possible configuration of energy flow on the arcs of the grid, and let $\varphi_{min}^T, \varphi_{max}^T \in \mathbb{Z}_+^m$ be, respectively, the vector of minimum flows and the vector of maximum sustainable flows of the arcs. In addition, let $c^T \in \mathbb{Z}_+^m$ be the vector containing the aggregate cost of each arc of the grid. For the sake of simplicity, we denote by c_{ij} the aggregate cost $c(d_{ij}, k_{ij})$ of the arc (i, j). Finally, for each node $i \in \mathcal{N}$, let b_i be its energy requirement and let $b \in \mathbb{Z}_+^n$ be the vector containing the requirements of the nodes of the grid. Chosen a zero minimum energy flow on each arc and assuming that the total energy produced by the active nodes meets the energy requirement of all the passive nodes, the LP model of the problem is:

$$
\begin{cases}
\min \sum_{(i,j) \in \mathcal{A}} c_{ij} x_{ij} \\
\sum_{(i,j) | j \in \mathcal{N}^+(i)} x_{ij} \leq -b_i & \forall i \in \mathcal{O} \\
\sum_{(j,i) | j \in \mathcal{N}^-(i)} x_{ji} = b_i & \forall i \in \mathcal{E} \\
\sum_{j \in \mathcal{N}^-(i)} x_{ji} - \sum_{j \in \mathcal{N}^+(i)} x_{ij} = 0 & \forall i \in \mathcal{D} \\
x_{ij} \leq \varphi_{ij}^{max} & \forall (i,j) \in \mathcal{A} \\
x_{ij} \geq 0 & \forall (i,j) \in \mathcal{A} \\
\sum_{i \in \mathcal{O}} |b_i| - \sum_{j \in \mathcal{E}} b_j \geq 0.
\end{cases} \quad (5)
$$

We are interested in sub-optimal solutions to the problem (5) to be evaluated on the criteria the previous model does not take into account. We adopt a perturbation strategy in order to obtain a family $\mathbb{P}$ of LP problems, so that, once solved, each problem provides a sub-optimal solution to (5).

5.2. Perturbation of the model

Solving the problem (5) means to provide the amount of energy required by the passive nodes by saturating the arcs connecting active nodes to the dispatcher and having a low aggregate cost.

Given the optimal solution $x^\star$ to (5), we are interested in the saturated arcs connecting active nodes to the dispatcher, forming the set $\mathcal{S}$. An arc (i, D) such that $i \in \mathcal{O}$ is said to be saturated if the flow x_{ij} on it equals the energy requirement $|b_i|$ of the active node i.

The perturbation strategy is firstly based on desaturating, one by one, the saturated arcs, obtaining every time a new problem whose solution is a sub-optimal solution to (5). In this way, the flow units no longer provided by the active nodes connected to a saturated arc, will be provided by active nodes having no saturated arc in $\mathcal{S}$. In particular, considered an arc $(i, D) \in \mathcal{S}$, the new maximum capacity of the arc is obtained as $\varphi_{ij}^{max'} = \lfloor \varepsilon b_i \rfloor$, $\varepsilon \in [0, 1)$, where $\lfloor \ \rfloor$ denotes the floor, i.e., the largest previous integer. By means of this strategy, at most $|\mathcal{S}|$ sub-optimal solutions to (5) can be obtained.

Subsequently, in order to increase the number of sub-optimal solutions to (5), we consider, in pairs, the arcs saturated by the optimal solution $x^\star$ to the problem P and simultaneously lower their maximum capacity. More rigorously, given a pair of saturated arcs $\{s_h, s_k\} \in \mathcal{S}$, where $h \neq k$, the new maximum capacity of the arcs forming the pair is obtained as $\varphi_{s_h}^{max'} = \lfloor \varepsilon b_i \rfloor$, $\varphi_{s_k}^{max'} = \lfloor \varepsilon b_i \rfloor$, $\varepsilon \in [0, 1)$. By repeating the procedure for each pair of saturated arcs, additional sub-optimal solutions to the original problem (5) are obtained. In particular, by combining both previous approaches, it is possible to obtain at most $|\mathcal{S}| + \frac{|\mathcal{S}|!}{2(|\mathcal{S}|-2)!}$ solutions.

6. Hybrid fuzzy AHP-TOPSIS approach for efficient energy dispatching

Once the alternatives have been generated by the perturbation of the model, the system has to evaluate them in order to establish the best one, with respect to all the criteria. We propose the use of fuzzy AHP to prioritize the criteria. The alternatives are subsequently evaluated by means of TOPSIS. As said in Section 4, the considered criteria are: the environmental impact; the cost of the energy; the distance of supply; the load level of the power lines.

First, the expert has to fill a matrix $\tilde{H}$ which compares the criteria with each other by means of a fuzzified Saaty's scale of preference based on triangular fuzzy numbers from $\tilde{1}$ to $\tilde{9}$, whose meaning is showed in Table 1, with the addition of uncertainty. The expert has also to express an index of optimism ζ about the judgements as explained in Section 2.2.2. The system performs a defuzzification of the matrix $\tilde{H}$ according to equation (2), obtaining matrix $\hat{H}$. Finally the system computes the principal eigenvector w of $\hat{H}$, which contains the weights of the criteria.

For each alternative, the system computes the environmental impact, the cost of the energy, the distance of supply and the load level of the power lines. In particular, given an alternative x, its environmental impact π_x is defined as:

$$
\pi_x = \frac{\sum_{i | \exists (i,D) \in \mathcal{A}} \pi_i x_{iD}}{\sum_{i | \exists (i,D) \in \mathcal{A}} \vartheta_i x_{iD}}, \quad \vartheta_i = \begin{cases} 1 & \forall i | x_{iD} \neq 0 \\ 0 & \text{otherwise.} \end{cases} \quad (6)
$$

In addition, the cost of the energy k_x of the alternative x is:

$$
k_x = \sum_{i | \exists (i,D) \in \mathcal{A}} k_i x_{iD}. \quad (7)
$$

Further, the distance of supply d_x is expressed by:

$$
d_x = \sum_{(i,j) \in \mathcal{A}} d_{ij} x_{ij}. \quad (8)
$$

Finally, the load level l_x of the power lines is defined as:

$$
l_x = 1 - \frac{\sum_{(i,j) \in \mathcal{A}} \varphi_{ij}^{max} - \sum_{(i,j) \in \mathcal{A}} x_{ij}}{\sum_{(i,j) \in \mathcal{A}} \varphi_{ij}^{max}}, \quad (9)
$$

where $0 < l_x \leq 1$.

Alternative Flow

id	p_1D_1	p_2D_1	p_3D_2	p_4D_3	s_1D_2	s_2D_3	s_3D_3	D_2s_1	D_3s_2	D_3s_3	D_1CC	D_2CC	D_3CC	CCD_1	CCD_2	CCD_3	π	k	d	l
1	9	14	9	13	4	0	9	0	4	0	0	4	0	2	0	2	2.9375	177	576	0.2833
2	8	14	9	14	5	0	9	0	4	0	0	4	0	3	0	1	2.9167	179	579	0.2833
3	7	14	9	15	4	0	9	0	4	0	0	4	0	4	0	0	2.8958	181	582	0.2833
4	6	14	9	16	4	0	9	0	4	0	0	4	0	5	0	0	2.8750	183	585	0.2833
5	9	13	9	14	4	0	9	0	4	0	0	4	0	3	0	1	2.9167	178	578	0.2833
6	9	13	9	14	4	0	9	0	4	0	0	4	0	3	0	1	2.9167	178	578	0.2833
7	9	11	9	16	4	0	9	0	4	0	0	5	0	5	0	0	2.8750	180	582	0.2833
8	9	10	9	17	5	0	9	0	4	0	0	4	1	6	0	0	2.8542	181	590	0.2875
9	9	9	9	18	5	0	9	0	4	0	0	5	2	7	0	0	2.8333	182	598	0.2917
10	9	14	8	14	5	0	9	0	4	0	0	3	0	2	0	1	2.8958	179	575	0.2792
11	9	14	7	15	4	0	10	0	4	0	0	3	0	2	0	0	2.8542	181	574	0.2750
12	9	14	6	16	5	0	9	0	4	0	0	2	0	2	0	0	2.8125	183	573	0.2708
13	10	14	5	17	5	0	10	0	4	0	0	1	1	2	0	0	2.7708	185	578	0.2708
14	**9**	**14**	**4**	**18**	**5**	**0**	**9**	**0**	**4**	**0**	**0**	**0**	**2**	**2**	**0**	**0**	**2.7292**	**187**	**583**	**0.2708**
15	8	13	9	15	4	0	10	0	4	0	0	4	0	4	0	0	2.8958	180	581	0.2833
16	7	12	9	17	5	0	9	0	4	0	0	4	1	6	0	0	2.8542	183	592	0.2875
17	6	11	9	19	5	0	9	0	4	0	0	5	3	8	0	0	2.8125	186	609	0.2958
18	9	13	8	15	4	0	9	0	4	0	0	3	0	3	0	0	2.8750	180	577	0.2792
19	9	12	7	17	5	0	9	0	4	0	0	3	1	4	0	0	2.8125	183	584	0.2792
20	9	14	9	17	2	0	7	0	4	0	0	2	0	2	0	0	2.8654	183	586	0.2750
21	9	14	9	19	1	0	6	0	4	0	0	2	0	2	0	0	2.8333	186	591	0.2708

Table 3: Alternative flows with their performance values with respect to the criteria.

If n alternatives are generated, the system organizes them as rows of a matrix X, so as each row of X contains the unities of flow for each arc of the grid. The performance values of each alternative with respect to the criteria are organized as rows of a matrix Y, therefore given a row i of Y, where $i = 1, 2, \ldots n$, such row contains, respectively, the pollution level, the cost of the energy, the distance of supply and the load level of the power lines related to the i-th alternative. Matrix Y is the decision matrix the system evaluates by means of TOPSIS, using the weights in w. At the end of the computation, the system returns the best alternative with respect to all the considered criteria.

7. Experiments

The system was implemented in MATLAB and was tested on a prototype microgrid consisting of four producers, eight consumers, three prosumers, three dispatchers and a central controller, having the structure shown in figure 2.

Initially, we compared in pairs the criteria discussed in Section 4 so as to fill the fuzzy matrix needed by fuzzy AHP:

$$\tilde{H} = \begin{array}{c} \\ E \\ C \\ D \\ L \end{array} \begin{array}{cccc} E & C & D & L \\ \begin{pmatrix} 1 & \tilde{2} & \tilde{3} & \tilde{5} \\ \tilde{2}^{-1} & 1 & \tilde{5} & \tilde{7} \\ \tilde{3}^{-1} & \tilde{5}^{-1} & 1 & \tilde{4} \\ \tilde{5}^{-1} & \tilde{7}^{-1} & \tilde{4}^{-1} & 1 \end{pmatrix} \end{array}. \quad (10)$$

The judgements in $\tilde{H}$ are used to prioritize the criteria in relation to a particular context, therefore they should be set by an expert. Nevertheless, we estimated the judgements according to common sense, providing them with an uncertainty expressed by triangular fuzzy numbers having a support equal to an interval with length 2. The elements of the pairwise comparison matrix $\tilde{H}$ are taken by the function computeWeights() as argument. The function performs a defuzzification according to equation (2), by using a further parameter ζ, i.e., the index of optimism we have described in Section 2.2.2. We chose $\zeta = 0.5$, based on heuristic considerations. The returned consistency index was $CI = 0.093$, therefore the decision matrix $\tilde{H}$ was said to be consistent. In addition, the function returned the vector containing the weights of the criteria $w = (0.434, 0.381, 0.132, 0.053)$. At this point, the system asked the following parameters to be entered: the incidence matrix E of the network; the vector b of the energy requirements of the nodes; the vectors φ^{max} and φ^{min} of the maximum and minimum flows; the vector d of the distances between the connected nodes; the vector k of the energy costs. We entered the parameters according to a given order. Nodes were entered in groups, and the nodes in each group were in lexicographic order. Groups of nodes were composed, respectively, by consumers, producers, prosumers, dispatchers and finally the central controller. Arcs were entered in the following order: the ones connecting each dispatcher to the consumers, the ones connecting each producer to a dispatcher, the ones connecting each prosumer to a dispatcher and vice versa. Finally the system requires the arcs connecting each dispatcher to the central controller and the arcs connecting the central controller to the dispatchers. The parameters we inserted are summarized in the following: $b = (8, 12, 7, 2, 3, 5, 15, 7, -10, -15, -10, -20, -5, 4, -10)$; $c = (0, 0, 0, 0, 0, 0, 0, 0, 2, 3, 2, 4, 2, 2, 3, 0, \ldots, 0)$; $\varphi^{max} = (20, \ldots, 20)$; $\varphi^{min} = (0, \ldots, 0)$; $d = (3, 4, 2, 5, 8, 4, 4, 4, 5, 6, 3, 8, 2, 1, 3, 2, 1, 3, \ldots, 3)$. For reasons of space, we omit the incidence matrix of the network. However, the incidence matrix can be easily obtained

according to equation (3), observing Figure 2.

By using the previous arguments, the system called the function `getSubOptimalSolutions()` which solved the optimization problem (5), finding the optimal solution $x^\star$, by means of the simplex algorithm. The optimal solution of the problem was $x^\star = (8, 12, 7, 2, 3, 5, 15, 7, 10, 15, 10, 13, 5, 0, 10, 0, 4, 0, 0, 5, 0, 2, 0, 3)$. The system then began to perturb the model lowering the maximum capacities of the saturated arcs, firstly one at a time, successively in pairs, obtaining suboptimal solutions, i.e., the alternatives to be evaluated. For each alternative, the system calculated the values of the criteria, according to equations (6-9), by calling the function `evaluateAlternativesOnCriteria()`. In Table 3 we have summarized the suboptimal solutions. In the header of the table there are, from left to right, an identifier *id* of the alternative, and the arcs connecting, respectively, producers to dispatchers, prosumers to dispatchers, dispatchers to the central controller and central controller to the dispatchers. Note that, for reasons of space, arcs are represented without the arrow. For example, the arc $p_1 D_1$ represents the arc $p_1 \rightarrow D_1$. The header of the table is terminated with the symbols denoting the criteria, i.e., the pollution level π, the cost of the energy k, the distance of supply d and the load level l of the power lines. We have also omitted the first eight columns which would contain the incoming flows of the consumers. Such flows are the same for each alternative, i.e., $\{8, 12, 7, 2, 3, 5, 15, 7\}$.

As last step, the system passed the alternatives shown in Table 3 to the function `rankAlternatives()` which, by means of the TOPSIS algorithm and using the weights in w, chose the alternative 14 as the best one. In Table 3 the best alternative is represented in bold.

8. Conclusions

In this paper we presented a new approach based on multi-criteria optimization and decision making to optimize the energy dispatching in smart (micro)grids. The optimization method we have proposed is based on four criteria: environmental impact, energy cost, distance of supply, and load level of the power lines. The problem has been modeled by means of linear programming, by arbitrarily considering two of the four criteria, i.e., the distance of supply and the cost of the energy. By modifying the maximum flow constraints of the problem it has been possible to generate sub-optimal flows. Such flows have been subsequently evaluated on the criteria the LP model does not take into account, by means of a hybrid multi-criteria decision making approach based on fuzzy AHP and TOPSIS. Fuzzy AHP has been used to associate a weight with each criterion according to the judgements of an expert, in order to obtain a vector of weights so as to prioritize the criteria. Subsequently, TOPSIS has been used to select the best configuration of flow among the available ones, with respect to all the criteria.

We believe the proposed approach is interesting since

it is based on evaluations of the importance of the criteria which can be expressed with different levels of uncertainty and vagueness. In addition, an MCDM approach allows to manage the problem quite easily thanks to the use of pairwise comparison matrices for the prioritization of the criteria and an evaluation of the alternatives simply based on the performance they attain on each criterion.

The proposed approach can be extended by considering further criteria and guiding the generation of the alternatives according to different strategies of perturbation, to be used even in parallel. Finally, it is also possible to extend the model to deal with larger grids, maybe composed by a potentially high number of microgrids.

References

[1] D. Ruan C. Kahraman, U. Cebeci. Multi-attribute comparison of catering service companies using fuzzy ahp: the case of turkey. *International Journal of Production Economics*, 87:171–184, 2004.

[2] Kwangsun Yoon Ching-Lai Hwang. *Multiple attribute decision making*. Springer-Verlag, 1981.

[3] David Cornforth. The role of microgrids in the smart grid. *Journal of Electronic Science and Technology*, 9(1):9–16, March 2011.

[4] George B. Dantzig. *Linear Programming and Extensions*. Princeton University Press, 1965.

[5] C. Marnay G. Venkataramanan. A larger role for microgrids. *IEEE Power Energy Magazine*, 5(4):78–82, May/June 2008.

[6] B. Lasseter. Microgrids (distributed power generation). In *Proceedings of IEEE Power Engineering Society Winter Meeting*, volume 1, pages 146–149, Ohio, USA, February 2001.

[7] C. V. Negoiţă. *Expert systems and fuzzy systems*. Benjamin-Cummings Publishing Co., 1985.

[8] Reza Iravani Chris Marnay Nikos Hatziargyriou, Hiroshi Asano. Microgrids. *Power and Energy Magazine, IEEE*, 5(4):78–94, August 2007.

[9] Thomas L. Saaty. *The analytic hierarchy process: Planning priority setting*. McGraw Hill, 1980.

[10] Thomas L. Saaty. How to make a decision: The analytic hierarchy process. *European Journal of Operational Research*, (48):9–26, 1990.

[11] Thomas L. Saaty. Decision-making with the ahp: Why is the principal eigenvector necessary. *European Journal of Operational Research*, 145:85–91, 2003.

[12] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, 1965.

[13] Furong Li Zhenjie Li, Yuan Yue Yuan. Evaluating the reliability of islanded microgrid in an emergency mode. In *Universities Power Engineering Conference (UPEC), 2010 45th International*, pages 1–5, 2010.

[14] H. J. Zimmermann. *Fuzzy set theory and its applications*. Kluwer, 1985.