ARBEITSBERICHT
**PROZESS- UND PRODUKT-ENGINEERING: METHODEN**

2008/18, ISSN 1814-1307 (Print), ISSN 1814-1315 (Online)

# AN INTRADAY SPOTMARKET-PRICE MODEL BASED ON CLUSTERING

Hans Vollbrecht thomas.breuer@fhv.at

25. April 2008

Fachhochschule Vorarlberg
Forschungszentrum
Prozess- und Produkt-Engineering

# An Intraday Spotmarket-Price Model based on Clustering

Working Paper 2008/01
Dr. Hans Vollbrecht
Research Center PPE at the FH Vorarlberg

## Abstract

A price model for the intraday spot-market of electricity at the European Exchange Market (EEX) is being presented. This model is based on clustering of normalized historical intraday price-curves and on subsequent sampling from those clusters and on final denormalization. Clustering is performed by a standard clustering technique (k-means). The innovation of the clustering, however, consists in establishing the optimal number of clusters by means of an entropy-based quality measure of a clustering. This measure indicates how well the clusters of a clustering reflect some calendaric classification of the historic price-curves, which is taken as a reference classification. The assumption of some, albeit unknown calendaric interpretation of clusters of similar price-curves seems to be justified by many investigations of different authors, and furthermore it has some well-known explanations based on fundamental influences peculiar to the price process of electricity. The problem of finding an optimal number of clusters in this way is non-trivial and requires for reasons of complexity a search algorithm with a meta-heuristic, here a genetic algorithm, which is shown to outperform our best search algorithm based on an ad-hoc heuristic. It is shown empirically in this work that intraday price-curves generated from this model have the advantage of reflecting better their true variance/volatility than those generated by well-established models such as autoregression-based (ARMA) or diffusion-based (two-factor Pilipovic-) models. Modelling electricity prices is of primary importance for stochastic optimization in the energy market, as for example in portfolio optimization, evaluation of contracts for power supply, or operational planning of power plants. Since stochastic optimization is advantageous over deterministic optimization for its taking into account the possibility of a decision recourse which is based on the variance of a stochastic input, a valid model of the variance of the price process is particularly important to a reliable stochastic optimization in the energy market.

## 1 Introduction

It is more than a decade now since the liberalization of the electricity market took place. In the meantime, a liquid market has grown up. Thus, the motivation for operating a power plant or not is given by the price established by the market, and not so much by the regional demand for electric power. A central role for this development is taken by the energy exchanges. The EEX is the exchange market for central Europe, and the work of this paper regards the intraday price behaviour of the spot market at the EEX. This market trades for the 24 hours of the following day an hour-by-hour price of a MWh. These prices are also the de-facto basis for bilateral OTC trading outside the exchange market.

Consequently, the energy prices at the spot market form the basis for many important decisions of an energy trading company, such as the evaluation of contracts for power supply, the operational planning of power plants, or a portfolio optimization. A portfolio might contain thermic or hydraulic power generation units, energy supply contracts, or energy futures. A model of future electricity prices is thus of paramount importance for energy trading and producing companies.

Looking at the price process itself, a number of important characteristics to be taken into account for such a model are immediately evident from historic price curves (see fig. 1 to 3):

- prices have strong long term and short term volatilities,
- prices have extreme spikes, and

- prices have strong periodical structures: daily, weekly and (not evident from the figures) seasonal.

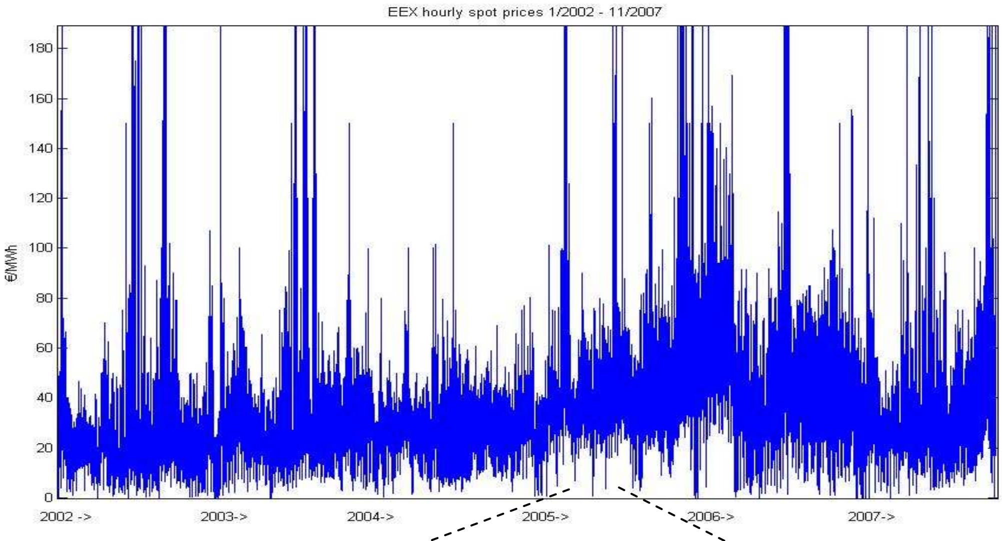


*figure 1: the EEX spot market price since 2002 (note: prices are cut off at €180)*

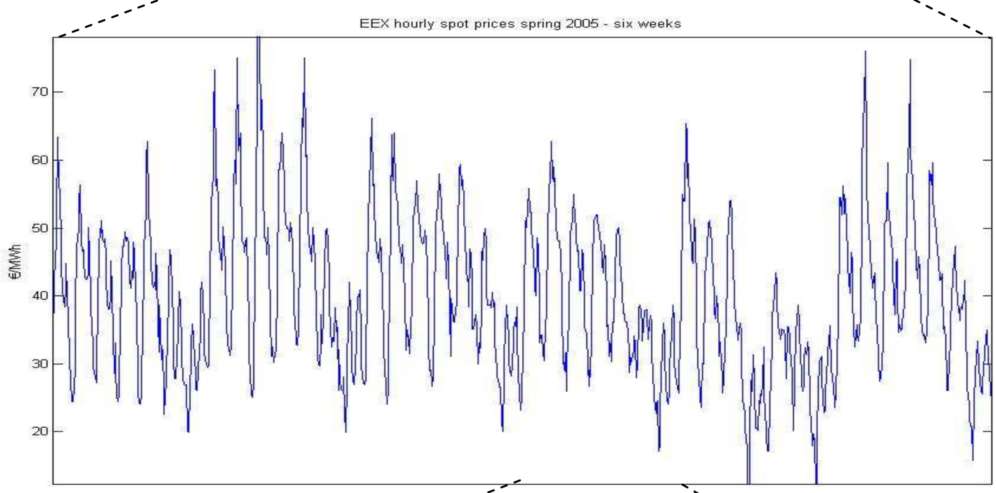


*figure 2: the EEX spot market price: weekly structure (6 weeks in spring 2005)*

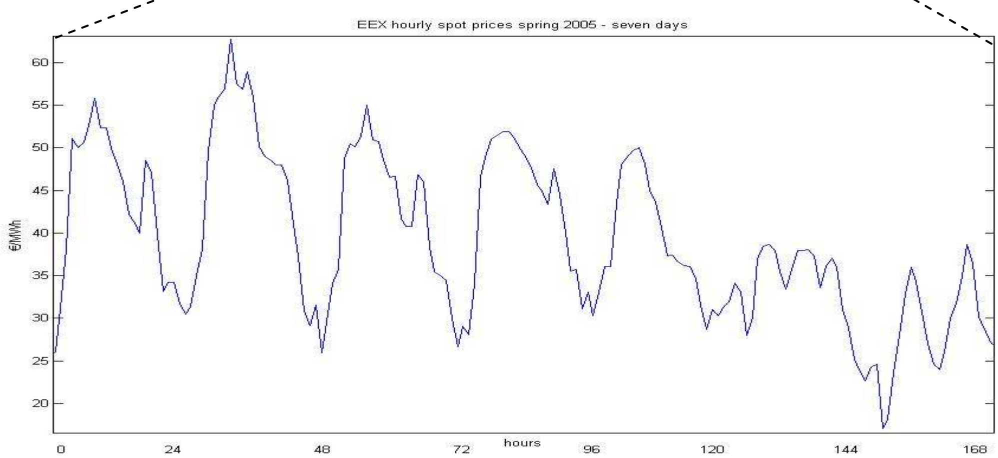


*figure 3: the EEX spot market price: daily structure (monday to sunday in spring 2005)*

There are a number of further characteristics of the spot market price of electricity, such as mean reversion. Some of these characteristics differ from those of other price processes such as stock exchanges or foreign exchange rates. This is due to the peculiarity of electricity of not being a storable commodity and therefore of being tightly correlated with the demand of electricity. For a detailed description see [1]. The long-term volatility, as evident from figure 1, is hard to model with a purely stochastic approach since many fundamental influences are responsible for it [2]. *This long-term behaviour is not in the focus of this work.*
The daily behaviour is characterized by a midday peak in the summer and an evening peak in the winter, by a generally lower price at the weekend and lower prices at night in general. The intraday volatility can be seen from figures 4-7. Note that all depicted price-curves have been normalized: no outliers, zero mean and standard deviation of one as a 24-dimensional vector.

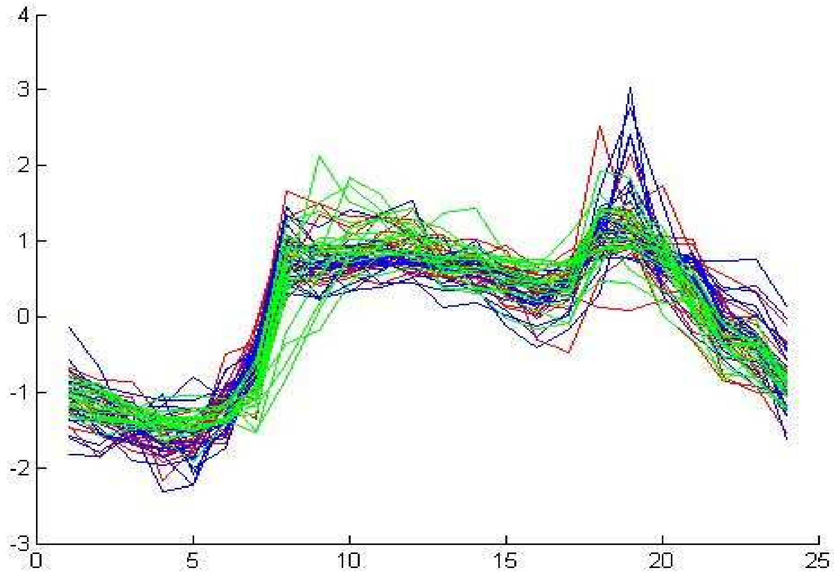


*figure 4: normalized EEX intraday price-curves: working days in January 04-06*

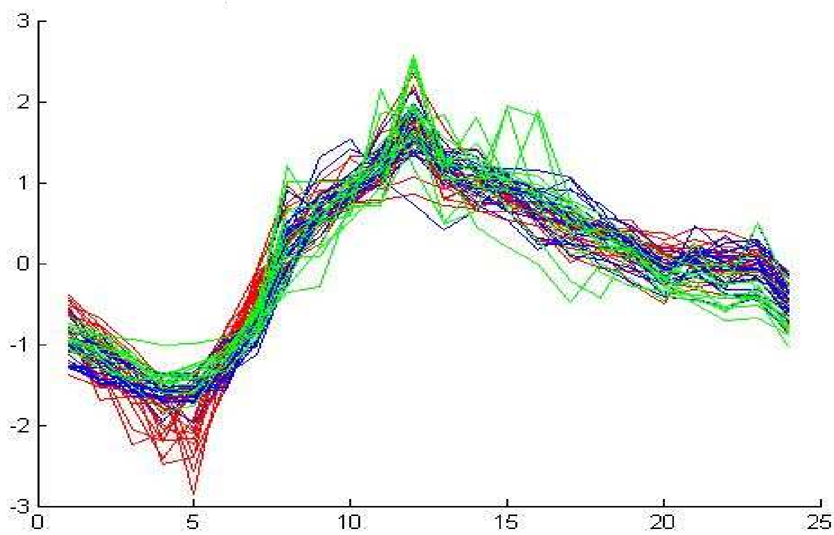


*figure 5: normalized EEX intraday price-curves: working days in July 04-06*

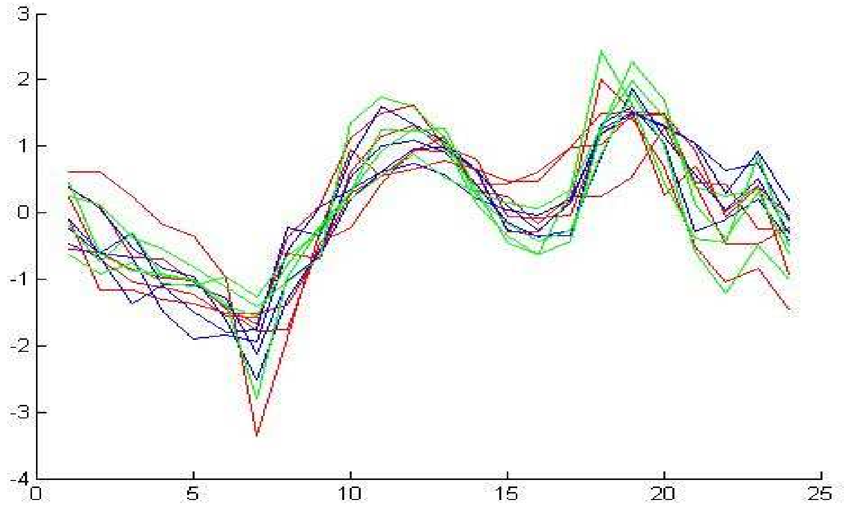


*figure 6: normalized EEX intraday price-curves: saturdays in January 04-06*

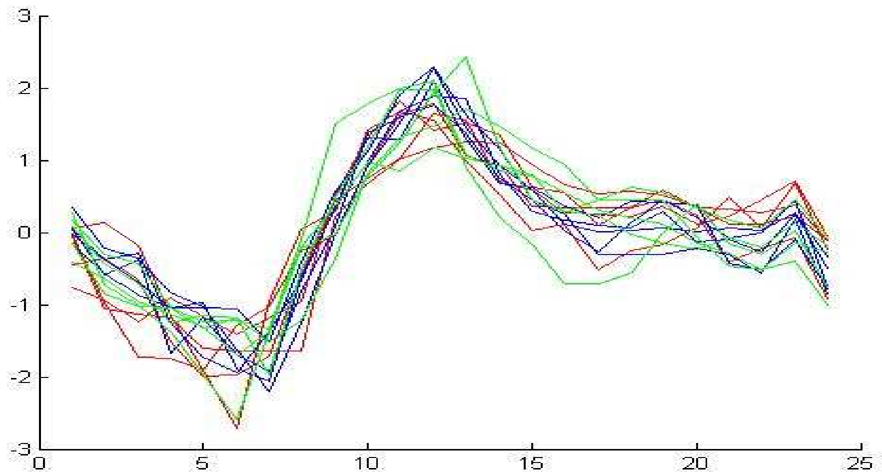


*figure 7: normalized EEX intraday price-curves: saturdays in July 04-06*

The daytype-specific shape and volatility of intraday price-curves is obvious. This leads to the idea that a clustering of historic, normalized intraday price-curves should be possible in such a way that it reflects a calendaric interpretation. The usefulness of such a clustering stems from the observation that electricity traders at the EEX, such as those working at our research cooperation partner Vorarlberger Kraftwerke AG (VKW), construct future price scenarios from historical price-curves by means of analogy, based on calendaric information of future days. These future price scenarios are then used in optimization tasks as those mentioned above.

The objectives we followed in this work are

- to construct an optimal clustering of historic, intraday price-curves with respect to a calendaric interpretation (a calendaric reference classification),
- where the calendaric classification is unknown in advance, and
- to develop a Monte-Carlo simulation technique for simulating future intraday price-curves based on sampling from these clusters, a simulation which then gets
- integrated into a complete price scenario simulation system,
- improving thereby deficiencies of well-known price models (see next chapter) which have too high an intraday volatility (shown in the results section).

The report is structured in the following way. In the next chapter, we introduce price models of spot market prices of electricity which on the one hand we benchmarked our model against, and which on the other hand we adapted to model the medium and long-term price process of the overall price process into which the intraday model gets integrated. This overall process will be described as well. In particular, we will present data normalization prior to clustering, and data denormalization after clustering prior to simulation by sampling. The denormalization follows a non-trivial model which will be presented. In the third chapter, we will discuss quality measures for clusterings and in particular our quality measure for comparing clusterings with different numbers of clusters, based on a calendaric reference classification. For the latter we will give a precise definition, and define a search problem that optimizes this quality measure. In the forth chapter, we present a genetic algorithm that searches for an optimal clustering. In the fifth chapter, results are presented both regarding the performance improvement of our genetic search algorithm in comparison to our best algorithm using an ad-hoc heuristic, and regarding the improvement of the simulated price-curves using sampling from the clusters. The latter improvement is judged when compared to price-curves simulated by standard price models presented in the second chapter. Finally, some

# 2 Spotmarket price models for electricity

A model for daily price-curves as presented in this work is just one component out of a complex model structure for the spot-market electricity prices. In this section, we present this total model which we are using for scenario generation, and present three model variants. Two of them are standard and already published and are used here as a benchmark which we tested the third variant against. This third variant uses the clustering technique which is the new development of this work.

## 2.1 The basic price model

From the numerous electricity price models published so far, we have chosen the one described in [3]. Details and their justification can be found there. In this section we just sketch this model and outline where our model differs from it. The spot price $S_t$ ($t$ always indicates an hour in the sequel) is decomposed into four components: two deterministic ones, the trend $S_t^{tr}$ and the seasonal part $S_t^{seas}$ , and two stochastic ones, the outlier part $S_t^{out}$ and the residual part $S_t^{res}$ . The price $S_t$ without outliers is modeled as

$$S_t = S_t^{tr} \cdot S_t^{sais} \cdot S_t^{res} \qquad (1)$$

The trend follows an exponential model:

$$S_t^{tr} = S_0 \cdot e^{\gamma \cdot t}$$

All parameters of the above model are estimated on historical EEX spot market prices which have been preprocessed by eliminating outliers. Outliers are defined by a threshold which we define on a yearly basis by the median plus three times the interquantile distance of the 25% and 75% quantile.

*a) seasonal part*

The seasonal part is modeled by a trigonometric polynomial with a basic oscillation of one year plus the first harmonic of half a year. For the daily profiles, we define five categories of days: Monday or day after or between holidays, Tuesday to Thursday, Friday or day before holiday, Saturday, Sunday or holiday. For each of these categories $k$ (1-5) and for each hour $j$ (1_24), we estimate an independent yearly model based on the trigonometric polynomial

$$S_{jkt}^{sais} = \alpha_{jk} + \sum_{i=1,2}\left(\beta_{ijk}\cos\left(2\pi\frac{t-\tau}{8766}\right) + \gamma_{ijk}\sin\left(2\pi\frac{t-\tau}{8766}\right)\right) \quad (2)$$

*b) residual part*

The residuals are assumed to follow an ARMA process with time lags of $t = 1$, $t = 24$ and $t = 25$:

$$S_t^{res} = \sum_{i=1..p}\alpha_i \cdot S_{t-i}^{res} + \sum_{j=1..q}\beta_j \cdot \varepsilon_{t-j} + \varepsilon_t \quad (3)$$

where $\varepsilon_t$ are error terms assumed to be IID normally distributed. For estimating the parameters, we remove outliers from the historical EEX data, factor out the deterministic parts, and finally transform the remaining historical residuals to obtain a normal distribution. These transformed residuals are then used for estimation of the ARMA parameters.

*c) outliers*

Outliers are then added to the model by a random process that creates an outlier at $t$ with probability $p_t^{out}$ depending on whether there has been an outlier at certain preceding points of time. The values of outliers are modeled by a gamma-distribution.

## 2.2 The Pilipovic model

As a second price model for benchmarking we use the two-factor Pilipovic model [4]. Starting from the same overall model structure (1), the residual part (3) gets substituted by a two factor Pilipovic model:

$$\begin{aligned} S_{t+1}^{res} &= S_t^{res} + \kappa_1 \cdot (\tilde{S}_t^{res} - S_t^{res}) + \sigma_1 \cdot \varepsilon_1 \\ \tilde{S}_{t+1}^{res} &= \tilde{S}_t^{res} - \kappa_2 \cdot \tilde{S}_t^{res} + \sigma_2 \cdot \varepsilon_2 \end{aligned} \quad (4)$$

The first difference equation models a component for the short-term volatility, oscillating around a second component modelling a long-term volatility ($\sigma_1 > \sigma_2$). Both components have a mean-reversion controlled by the $\kappa_i$. We estimate the parameters by transforming the model in a state space representation and then maximize the likelihood of these parameters using a Kalman filter. Again, estimation is based on the historical residuals prepared as described in 2.1.

## 2.3 The Cluster-based model

Our third price model has as a component for the short term (intraday) volatility the cluster-based model of intraday price-curves described in detail in the following chapters. The model is the sum of a daily mean price plus the intraday price curve:

$$S_t = \bar{S}_{d(t)} + \hat{S}_{t-24\cdot d(t)}^{d(t)} \quad (5)$$

where $d(t)$ is the day that belongs to hour $t$, $\bar{S}_d$ is the daily mean price (over 24 values) of day $d$, and $\hat{S}_t^d$ is the hourly spot price for hour $t$ (from 0..23) of day $d$.

The model structure of the daily mean price is similar to (1):

$$\bar{S}_d = \bar{S}_d^{tr} \cdot \bar{S}_d^{sais} \cdot \bar{S}_d^{res} \quad (6)$$

with exponential deterministic trend, deterministic seasonal component analogous to (2), however now just with 5 different models indexed by $k$ (daytype), but not indexed any more

by the hour of the day, and a stochastic component for the residuals that is modeled as in (3) by an ARMA process with time lags of $t$ = 1, $t$ = 7 and $t$ = 8.
The model of the intraday price-curve is the following:

$$\hat{S}_t^d = f(Sn_t^d, s_d, \omega) \qquad (7)$$

where $f$ is a random function (on some probability space $\Omega$) which denormalizes the normalized (i.e. daily mean 0 and standard dev. 1) intraday price-curve $Sn_t^d$. This denormalization multiplies the 24 hourly prices of $Sn_t^d$ with a random factor modelling the standard deviation which depends on the daily mean $s_d$. The stochastic model of this factor is described in 2.6. The model of $Sn_t^d$ is not analytical any more and is described in 2.4 and 2.5.

## 2.4 Clustering and simulation of intraday price-curves

The following diagram illustrates schematically the process of building clusters of normalized daily profiles from historical EEX data.

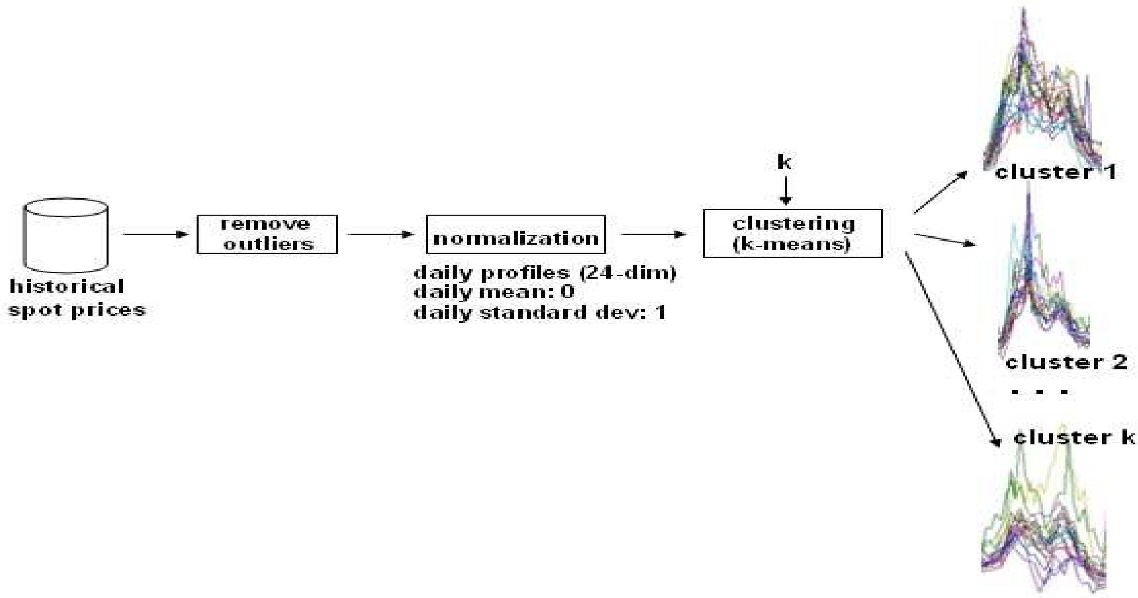


*figure 8: the clustering process – building clusters of daily (residual) price curves*

After having removed the outliers as described in 2.1, the historical daily profiles (24-dimensional vectors) get normalized by subtracting from each profile its mean on the 24 hourly prices, and then dividing each of the 24 values by the standard deviation of this profile.

These normalized 24-dimensional vectors get clustered by a standard clustering algorithm (k-means, see section 3) into k clusters. From such a clustering, we generate normalized daily profiles as illustrated in figure 9. Starting with a given time period, we determine for each day of that period its day-type and its month. We use nine distinct day-types: the seven different weekdays, a holiday, and a bridge-day which is a working day between two non-working days. Together with the month of the day, we get 108 different pairs of day-type and month which is called the calendaric label of a particular date within the simulation period. For each calendaric label, we determine the empirical cluster distribution of historical intraday price-curves with this label. Now, simulation of price-curve scenarios is performed by Monte-Carlo simulation in which for each day within the simulation period, we sample historical normalized price-curves out of a cluster sampled from the k clusters according to the empirical distribution of the label of the simulated day. The number of price-curves sampled randomly from the selected cluster is one in our experiments, but may alternatively be any small number. In the latter case, we would average over the samples price-curves to get the simulated price-curve.

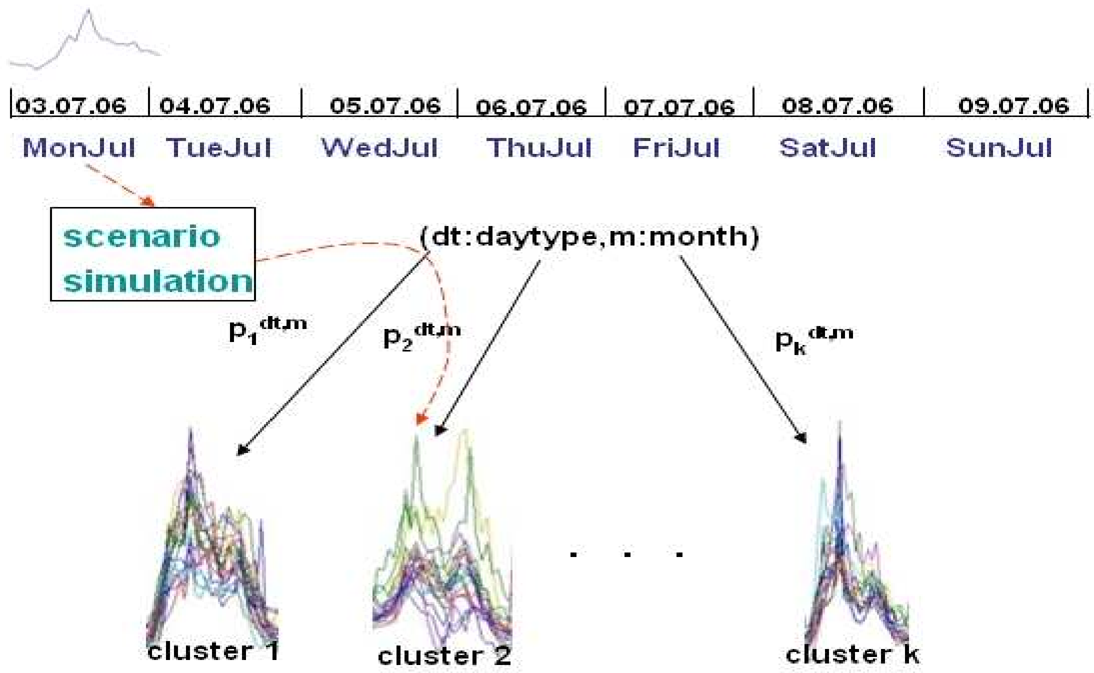


*figure 9: simulation of intraday price curves – $p_i^{dt,m}$ denotes the empirical probability of a day with daytype=dt and month=m being in cluster i*

Since the number of 108 different empirical distributions is too high for the amount of historical data available, and since it can be easily observed that certain labels (pairs of daytype-month) have very similar empirical distributions over the clusters, we define the scenario simulation in a more general way by defining a calendaric reference classification as a classification of daytype/months:

*calendaric reference classification*:

$$Ca^g := (ca_i)_{1 \le i \le g}, \text{ class } ca_i \subset DT := \{(\text{daytype},\text{month}) \mid 1 \le \text{daytype} \le 9, 1 \le \text{month} \le 12\}$$
$$\text{with } \bigcup_{i=1..g} ca_i = DT \quad \text{and } ca_i \cap ca_m = \varnothing \quad \forall_{i \ne m} \qquad (8)$$

Now, instead of using empirical cluster distributions for each daytype/month, we use such distributions for each calendaric class $ca_i$ , as illustrated in figure 10. The main problem is to determine the optimal number of clusters k and then, for a selected k, an optimal calendaric reference classification. This problem and its solution is described in detail in the next sections.

The overall simulation procedure is illustrated in figure 11. All models but the denormalization have been described so far. The latter is described in the next subsection.

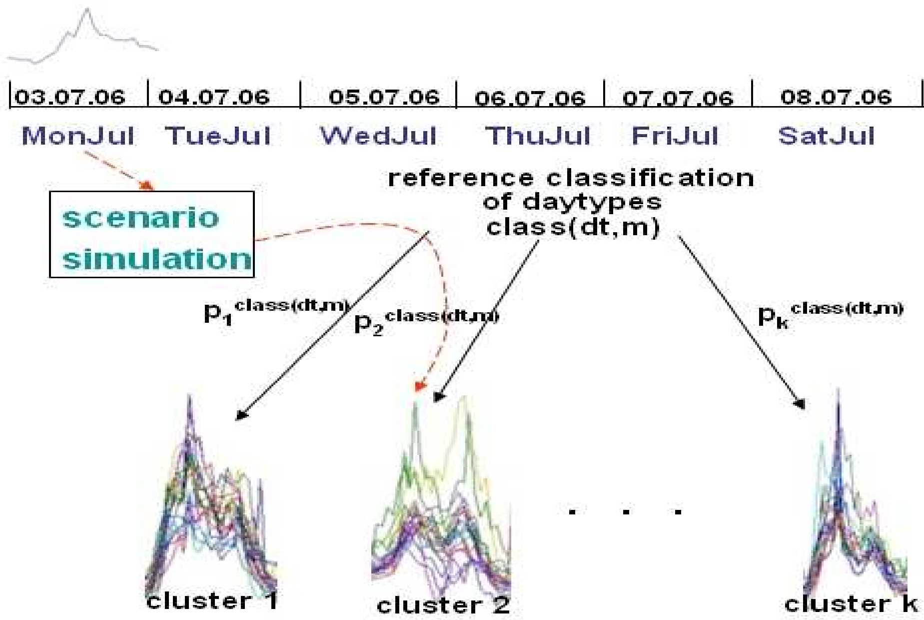


*figure 10: simulation of intraday price curves based on calendaric reference classification $C^k$ (class(dt,m) is the class $c_i$ with (dt,m) $\in c_i$)*

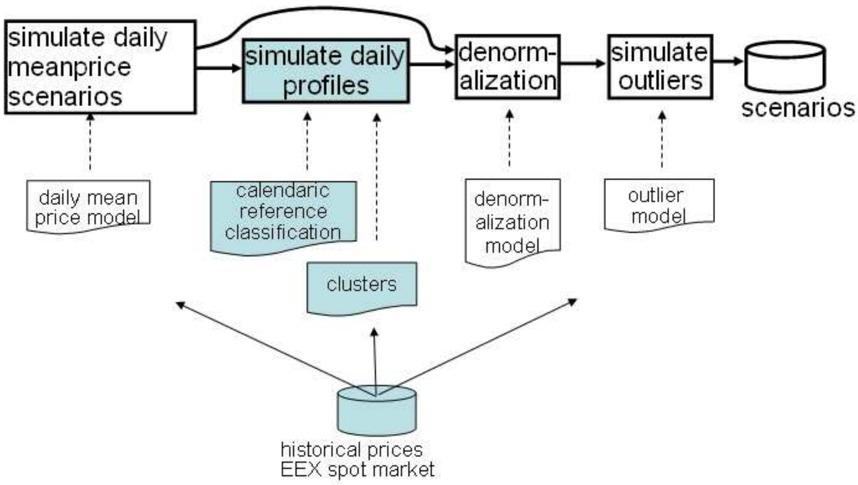


*figure 11: overall scenario simulation process*

### 2.5 The denormalization model

Denormalization reintroduces a random standard deviation into the normalized intraday price-curve. It is well known that the standard deviation of electricity prices depends on the daily mean price [2]: the higher the daily mean, the higher the standard deviation. The denormalization function as given in (7) is modelled as a random factor $v$ by which a normalized price-curve gets multiplied point-wise. The dependence of $v$ on the daily mean $s_d$ is depicted on the plot in figure 12 which gives a scatter plot of the historical daily mean against daily standard deviation of the EEX spot market. The dependence is obviously non-linear. We found best a quadratic fit $c_2 \cdot s_d^2 + c_1 \cdot s_d + c_0$. The distribution of $v$ is modelled in the following way:

$$\nu = normal(\mu_n, \sigma_n) \quad with \quad \mu_n = c_2 \cdot s_d^2 + c_1 \cdot s_d + c_0 \quad and \quad \sigma_n = c_3 \cdot \sqrt{s_d} + c_4 \qquad (9)$$

The fitted values of the parameters are the following:

| $c_0$ | $c_1$ | $c_2$ | $c_3$ | $c_4$ |
|---|---|---|---|---|
| 3.9123 | 0.08119 | 0.00282 | 0.5217 | -0.3893 |

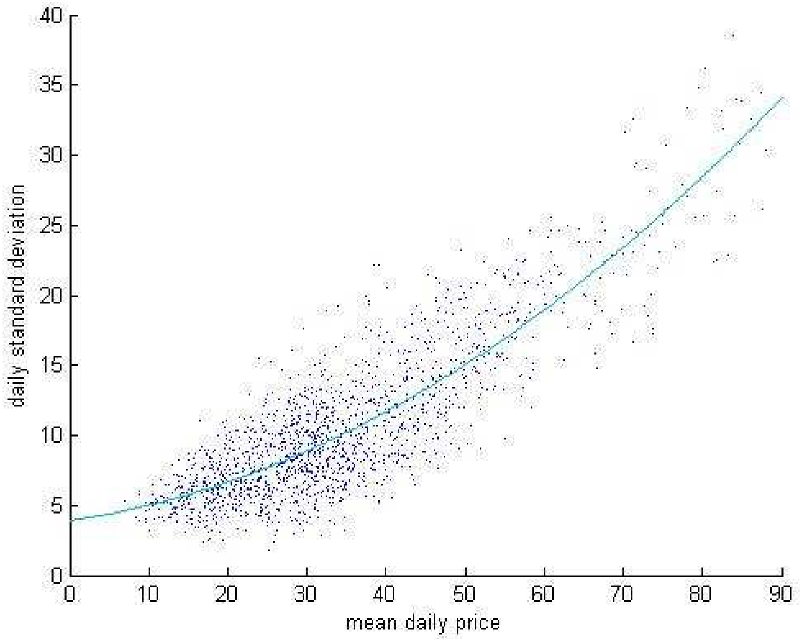


figure 12: scatter plot of daily mean prices against daily standard deviation, and quadratic fit

## 3 Quality measures for clustering price-curves

The clustering of daily, normalized price-curves is performed by a standard k-means algorithm. This algorithm minimizes the mean *intracluster distance*, sometimes also called the "within-cluster" point scatter:

$$\sum_{k=1..K} \sum_{C(i)=k} \sum_{C(j)=k} \left\| x_i - x_j \right\|^2$$ where *C(i)* is the cluster of point *i*, assuming *K* clusters

This algorithm is not guaranteed to converge to the optimum, so it has to be replicated a given number of times in order to get the global optimum with a given probability.
The distance between two price-curves (24-dim. real-valued vectors) is simply calculated as the L2-norm. k-means and its convergence results are valid only for this norm. It should be stated here that other distance measures might be more appropriate for specific optimization problems using price scenarios. However, in the absence of further information about the specific optimization problem, the L2-norm seems to be the natural candidate.

The main problem this work has started from is that of determining the "best" number of clusters *K*. This optimal value must be based on some definition of a clustering quality. There exist a number of measures which can be classified as external and internal quality measures [5].

*a) internal (unsupervised) quality measures*

Internal quality measures depend only on the information the clustering algorithm (here k.means) itself uses for clustering. Examples are the silhouette coefficients [6] or the Intra/Inter Cluster Dissimilarity [5]. The latter is the quotient of mean intracluster distance and intercluster distance:

$$D = \frac{\sum_{k=1..K} \frac{1}{n_k} \sum_{C(i)=k} \sum_{C(j)=k} \left\| x_i - x_j \right\|^2}{\frac{1}{K} \sum_{i,j=1..K} \left\| c(i) - c(j) \right\|}$$

where *c(i)* is the cluster centroid of cluster *i*

The lower this measure, the better is the clustering. Although this measure does not exhibit any obvious dependency on the number of clusters *K*, it is not applicable in our application for deciding on the optimal number of *K* since it has a monotonous dependency on K. What is required from a quality measure of a clustering is, however, a dependency with a maximal value not being at the extremes of the interval within which we search for an optimal K. In the case of clustering daily price curves, there are some natural lower and upper bounds for K as will be discussed in the next point

*b) external (supervised) quality measures*

This type of clustering requires some a priori known structure of the samples which the clustering should reflect. This could be a reference classification of the samples. In case of price curves, it is quite obvious (fig. 4-7) that there is a strong calendaric dependency of the shape of the curves. A natural reference classification would be a calendaric one as defined in (8). This classification gives a natural lower and upper bound for K: since the distinction between summer and winter daily profiles (high midday peak in summer, high evening peak in winter) and the distinction between working days, Saturdays and Sundays is evident, K should be at least 6, and the upper bound is 108 since this is the maximum number of calendaric distinctions. This upper bound should be in practice even much lower since the number of available daily profiles at the EEX is in the order of 2000.

There exist a number of external quality measures some of which are well-known in the domain of document retrieval methods [5]:

- *Purity*
- *F-measure*
- *Entropy*

These measures are based on the following notation:

| | |
|---|---|
| $Ca^g := (ca_i)_{1 \le i \le g}$ : | a classification into g calendaric classes, see (8) |
| $Cu^k := (cu_i)_{1 \le i \le k}$ : | a clustering into *k* clusters (similar to (8), i.e. a crisp clustering) |
| $n_{ij}$ : | the number of samples (days) of class *j* in cluster *i* |
| $n_i$ : | number of samples in cluster *i* |
| $n^j$ : | number of samples in class *j* |
| $p_{ij} := n_{ij} / n_i$ : | empirical probability of a day of class *i* being in cluster *j* |
| $n$ : | the total number of samples |

The *Purity* of a clustering $P(Cu^k)$ is based on the dominant class of each cluster:

$$P(Cu^k) := \sum_{i=1..k} \frac{n_i}{n} \cdot P(cu_i) \qquad \text{where} \qquad P(cu_i) := \max_{1 \le j \le g}(p_{ij})$$

The *F-measure* measures the extent to which a cluster contains only objects of this class *and* all objects of that class:

$$F := \sum_{j=1..g} \frac{n^j}{n} \max_{i=1..k} F_{ij} \qquad \text{where} \qquad F_{ij} := \frac{2 \cdot p_{ij} \cdot \frac{n_{ij}}{n^j}}{p_{ij} + \frac{n_{ij}}{n^j}}$$

Finally, the *Entropy* measures, on an information theoretic background, the entropy of the distribution of the classes of the reference classification within a cluster:

$$Ent(Cu, Ca) := \frac{1}{k} \sum_{i=1..k} Ent(cu_i, Ca) \quad \text{where} \quad Ent(cu_i, Ca) := -\frac{1}{\log g} \sum_{j=1..g} p_{ij} \cdot \log p_{ij} \qquad (10)$$

A high entropy $Ent(cu_i,Ca)$ indicates a cluster that is unspecific about the classification: its maximum is achieved when $p_{ij} = 1/g$ , i.e. when each class $j$ has the same probability if being in cluster $i$. The Entropy is minimal when $p_{ij}=1$ for one class $j$, i.e. when the cluster matches exactly one class.

From these quality measures for clusterings, we only found the *Entropy* to fulfill the requirement of presenting a minimum inside the range of $k$. Thus, we can define the following precise statement of an optimization problem:

*Optimization Problem Statement:*
Find a clustering $Cu^{k'}$ and a reference classification $Ca^{g'}$ such that:

$$(Cu^{k'}, Ca^{g'}) = \arg\min_{Cu^k, Ca^g}(Ent(Cu^k, Ca^g) : 6 \le k, g \le 108) \qquad (11)$$

A solution to this optimization problem is a clustering and a reference classification. Both are required for our scenario simulation process as illustrated in figure 10.
The general clustering algorithm now has the following form:

**General Clustering Algorithm**

*min_entropy* = MAXFLOAT

for *k=minK to maxK*
1. find best *k*-clustering *Cu* by *m* replications of K-means
2. find calendaric reference classification *Ca* for *Cu* with minimal *entropy*
3. if *entropy* < *min_entropy*
   *min_entropy = entropy*
   *best_k = k*
   *best_Cu = Cu*
   *best_Ca = Ca*

end for

return *best_k, best_Cu, best_Ca*

# 4 A genetic algorithm for searching an optimal calendaric interpretation of clusters of price-curves

The optimization problem (11) is an NP-hard problem (because of the combinatorial search space of all partitions of the set of daytypes/months), and some (meta-)heuristic search method must be employed. We decided for a genetic algorithm which will be described in detail in this section. Before we do this, a first simplification of (11) will be introduced:

$$(Cu^{k'}, Ca^{k'}) = \underset{Cu^k, Ca^k}{\arg\min}(Ent(Cu^k, Ca^k) : 6 \leq k \leq 108) \tag{11a}$$

Thus, we are searching only for calendaric reference classifications which have the same number of classes as the corresponding clustering has clusters. This means that we expect all clusters to have a calendaric interpretation, and that different clusters have distinct calendaric interpretations.

The optimization problem (11a) requires a search for minimization in the space of S(n,k) (Stirling number) partitionings *Ca* into *k* calendaric classes, given a k-clustering *Cu*. This is accomplished by a genetic algorithm. This type of algorithm [7] uses a special coding of feasible solutions called chromosomes, and generates a sequence of populations of chromosomes by manipulating single chromosomes (so-called mutation) and recombining pairs of chromosomes (crossover). Each chromosome/solution is evaluated by its so-called fitness which is the entropy as defined in (10).

The general algorithm specification is the following:

**MERGA**: **M**inimal_**E**ntropy_**R**eferenceClassification **G**enetic **A**lgorithm

initialize *pop_size, mutation_rate, num_mutations, max_no_change* , *no_change*←0,
    *best_entropy*, clustering *Cu* (by k-means)
initialize *population* with *pop_size* chromosomes (see 4.4)
do while *no_change* ≤ *max_no_change*
    *new_population* ← *population* ∪ crossover( *population* ) (see 4.2)
    *new_population* ← *new_population* ∪ mutation( *new_population* ) (see 4.3)
    calculate entropies of *new_population* for clustering *Cu*
    *new_population* ← sort *new_population* by entropy in ascending order
    *population* ← best (i.e. first) *pop_size* chromosomes of *new_population*
    if entropy of first element of *population* < *best_entropy*
        then *best_entropy* ← entropy of first element of *population*
            *no_change* ← 0
        else *no_change* ← *no_change* + 1
end while

## 4.1 Chromosome Coding

Each chromosome *c* is defined as an integer-valued, 108-dimensional vector whose *i*-th component indicates the class *ca* of the datype-month *i* of which there exist 108:

*chromosome definition*

chromosome *c*: $class_c[i] \in \{1..k\}$, $1 \leq i \leq 108$, such that for each $ca \in \{1..k\}$, there exists at least one $i$, $1 \leq i \leq 108$, with $ca = class_c[i]$

Note that there are 9 daytypes and 12 months, as defined in 2.4. We use two different enumeration schemes for indicating a pair (*daytype,month*):

*Enum1*: the pairs are ordered by months: i = 1-9: the 9 daytypes in january,
i = 10-18: the 9 daytypes in February,
…

*Enum2*: the pairs are ordered by daytypes: i = 1-12: the monday for 12 months,
i = 13-24: the tuesday for 12 months
…

Each chromosome is logically held in two versions, each for a different enumeration schema. Using two different enumeration schemes is motivated by the crossover operation.

### 4.2 Crossover

Crossover selects randomly two different chromosomes and cuts their vectors at a random position each into two parts, and then recombining them in a crossover way, resulting in two new chromosomes:

*Crossover Operation*:

1. select an enumeration schema (1 or 2) randomly (uniform distribution)
2. select two chromosomes $c_1 \neq c_2$ from the population
3. select an $i$, $1 \leq i < 108$, randomly (uniform distribution): the cutting position
4. create two new child chromosomes $c_{12}$ and $c_{21}$:
   $class_{c12}[i] = class_{c1}[i] \;\; \forall j \leq i$ and $class_{c12}[i] = class_{c2}[i] \forall i < j$
   $class_{c21}[i] = class_{c2}[i] \;\; \forall j \leq i$ and $class_{c21}[i] = class_{c1}[i] \forall i < j$
5. if $c_{12}$ and $c_{21}$ do not satisfy the chromosome definition
   then go to 2
   else add $c_{12}$ and $c_{21}$ to *new_population*

The two enumeration schemes have been defined for recombining subsequences of chromosomes with a bias towards preserving months already well classified (schema 1), or a bias towards preserving day-types already well classified (schema 2). We have no experimental proof of whether this technique is superior to using just one enumeration schema.

### 4.3 Mutation

Mutation selects randomly a chromosome $c$ (from the previous population or from the new chromosomes constructed by crossover), and for a fixed number *num_mutations* (parameter of the algorithm) of randomly selected single positions $i$ in a chromosome, a change in the class $class_c[i]$ , $i \in \{1..k\}$, is being forced. This mutation occurs for each chromosome with a probability defined by the *mutation_rate*. A heuristic has been added to this standard mutation technique, giving a bias in the random selection of a new class $class_c[i]$ such that daytype-month $i$ has a high relative frequency in the cluster with the same number than that of the new class. This results in driving the search process towards a classification in which a class $i$ is dominant in cluster $i$. Again, we have no experimental proof of whether this technique is superior to using a random mutation without any bias.

### 4.4 Initialization

As with most search algorithms with meta-heuristics, the performance of the above algorithm is very sensible to an intelligent initialization of the first population. The initialization is done with the same heuristic as that described in 4.3: each of the *pop_size* new chromosomes $c$ gets assigned for an $i$ a class $class_c[i]$ such that daytype-month $i$ has a high relative frequency in the cluster with the same number than that of the assigned class. This results in starting the search process from a classification in which a class *i* is dominant in cluster *i*. Again, we have no experimental proof of whether this technique is superior to using a random mutation without any bias.

# 5 Results

In this chapter, results will be presented regarding

- the performance of the genetic algorithm searching optimal calendaric classifications
- the search of an optimal number of clusters
- the simulated intraday price-curves using the simulation process from figure 10
- the simulated spot-market price process using the simulation process from figure 11

All results are based on historical, normalized innerday-curves from the EEX spot market in the time of January first of 2002 until December thirty-first of 2005. Simulated prices cover the whole year of 2006.

## 5.1 The performance of the genetic algorithm

We performed the genetic algorithm with the following model parameters:

| *pop_size* | *mutation_rate* | *num_mutations* | *max_nochange* |
|---|---|---|---|
| 500 | 0.5 | 10 | 20 |

The genetic algorithm requires (for one value of k) about 30 minutes on a 1.6GHz PC with MatLab©. Figure 13 shows an example of the performance of the genetic algorithm searching for an optimal classification (15 classes) of a clustering with 15 clusters.

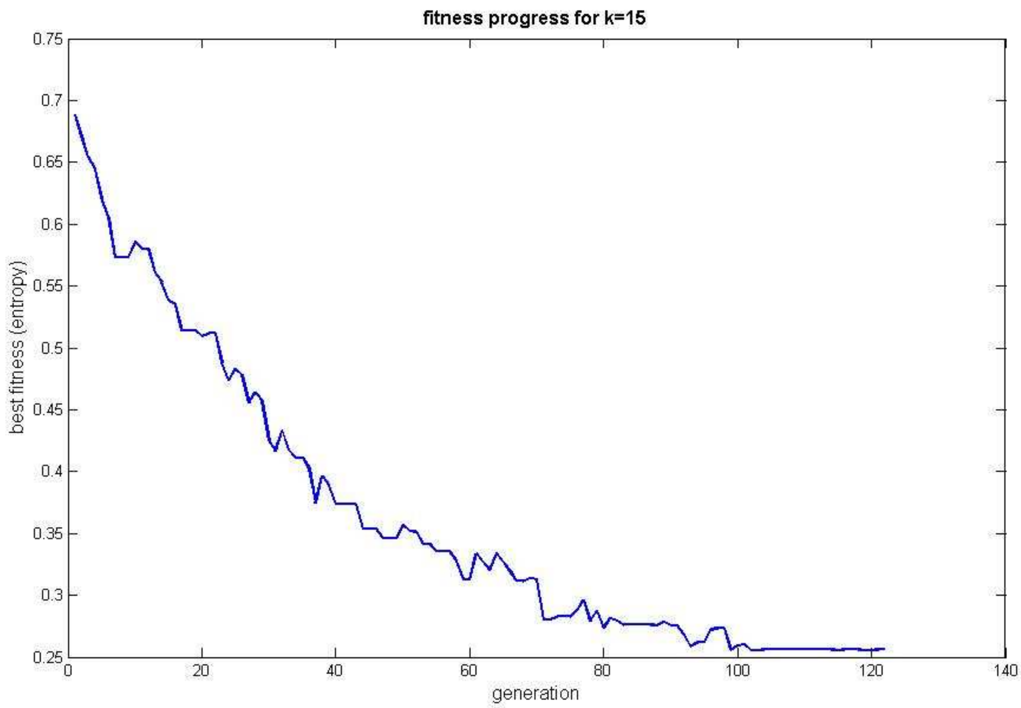


figure 13: fitness (entropy) progress for genetic algorithm MERGA for k=15

We compared the performance with our best ad-hoc heuristic algorithm. This algorithm starts with a relaxed classification in which different classes may overlap (same daytype-month in different classes are allowed). The initial relaxed classification has classes which have the same daytype-months as the clusters of the corresponding clustering. The algorithm performs a successive elimination (from classes) of daytypes-months which occur in several classes, applying „hill climbing“ with respect to the entropy, until all classes are non-overlapping. Figure 14 shows the comparison between the two algorithms. Results of the genetic algorithm are obviously better than of the ad-hoc algorithm.

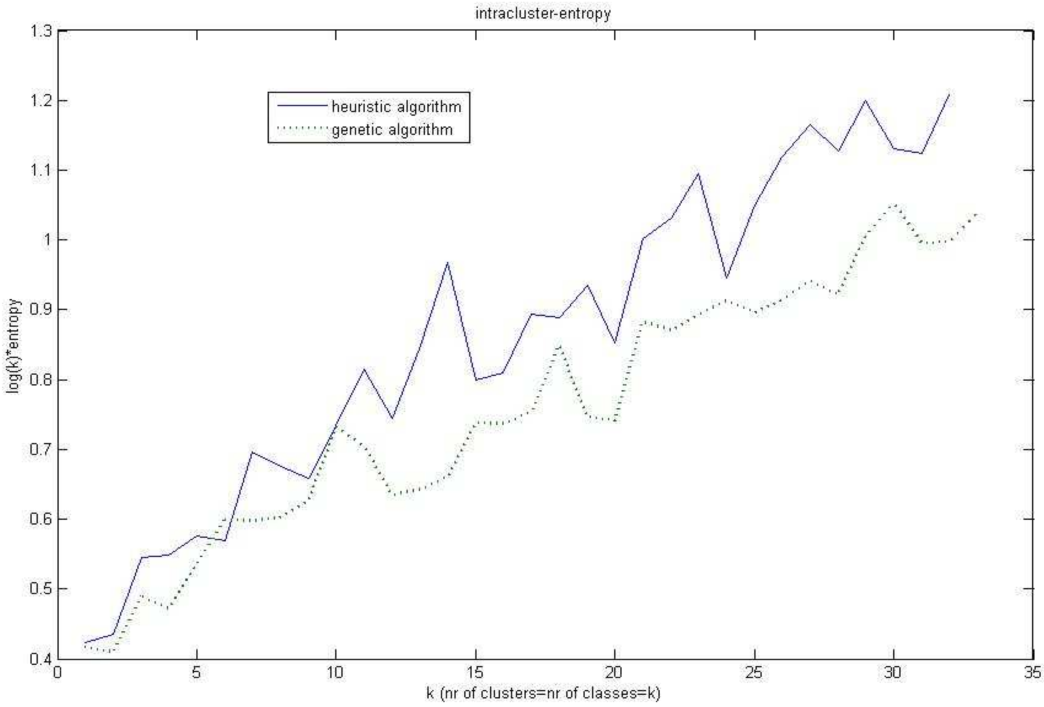


figure 14: comparison of genetic algorithm with ad-hoc heuristic algorithm

## 5.2 Optimal number of clusters

Experiments have shown that the range of search for an optimal number of clusters *k* [minK,maxK] in the general clustering algorithm (page 12) can be restricted to minK=4 and maxK=35. The results for the "entropy" quality measure is shown in figure 15.

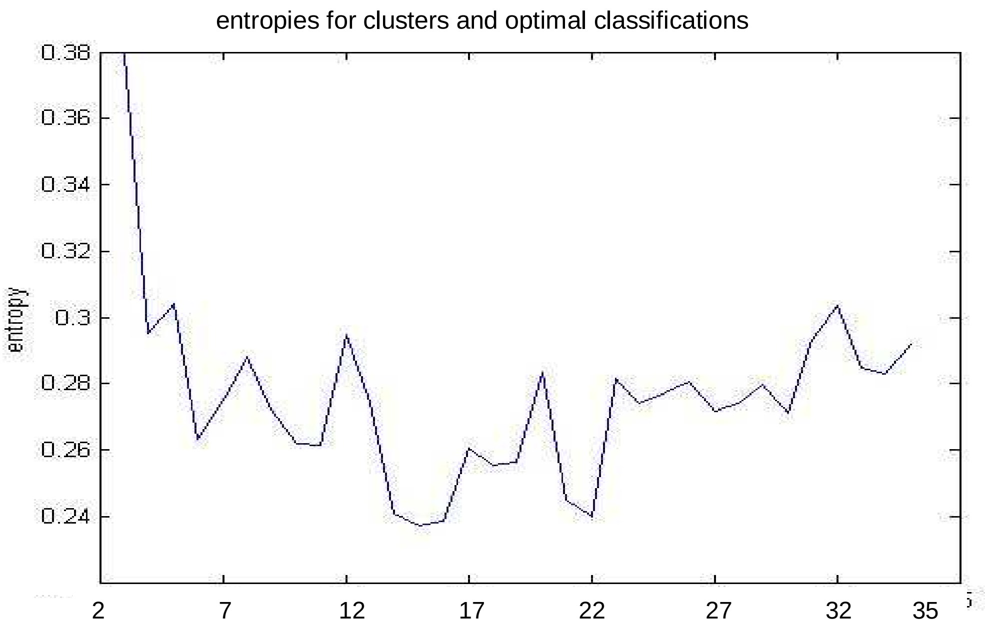


figure 15: entropies for clusters and its optimal calendaric classification

It can be seen, as a general tendency, that the clusterings in the range of 14 to 22 are best (lowest entropy). In particular, k=15 and k=22 are good candidates. We present further results for the choice k=15. In figure 16, an illustration of the optimal calendaric classification with 15 classes is presented. Each colour represents one class.

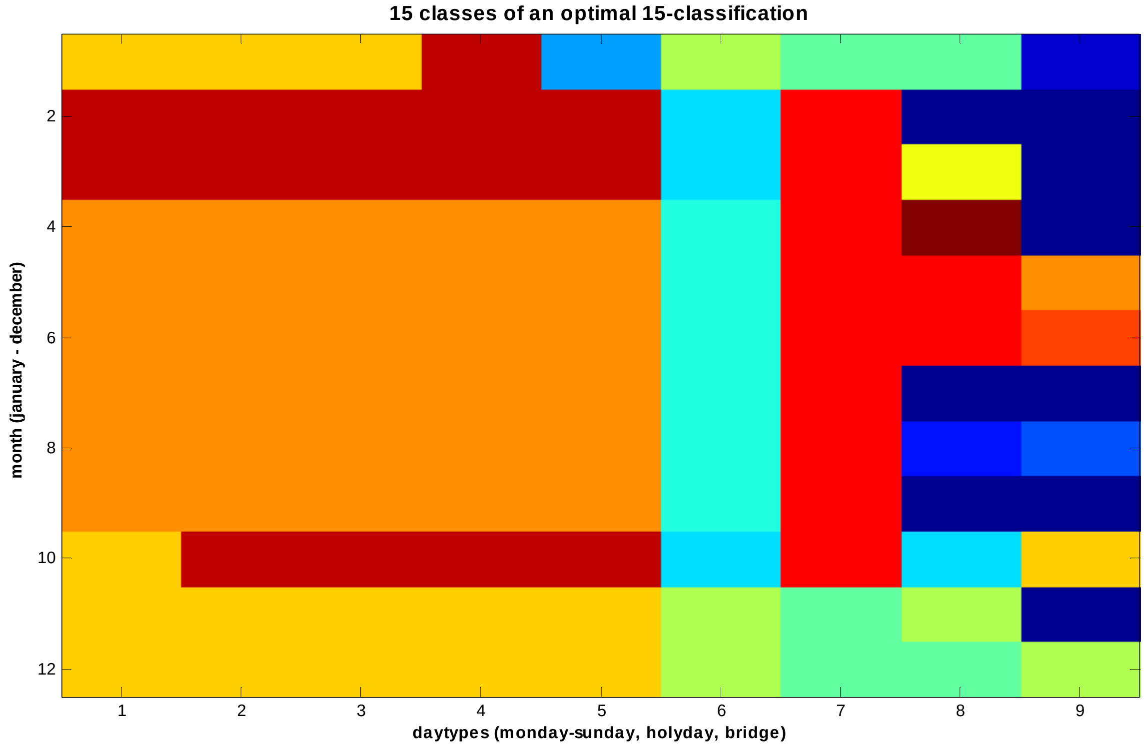


figure 16: an optimal calendaric 15-classification

The classification shows clearly a calendaric interpretability. Workdays in the summer and in part of spring and autumn (April to September) form the biggest class. Workdays in February and in March  together with the Thursday of January form another big class, as well as those in November and in December and the October Monday. Saturdays are clearly separated, as well as Sundays. Only holydays and bridge-days are more fragmented which might be due to the small number of samples.

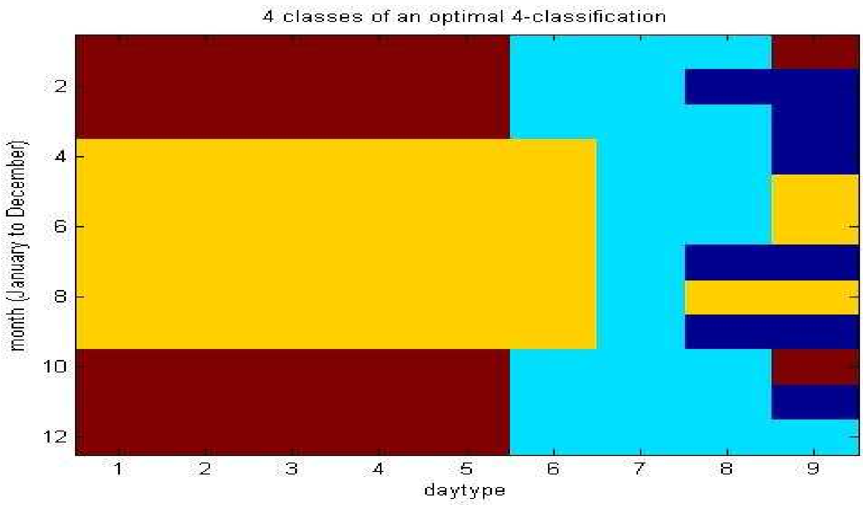


figure 17: an optimal calendaric 4-classification

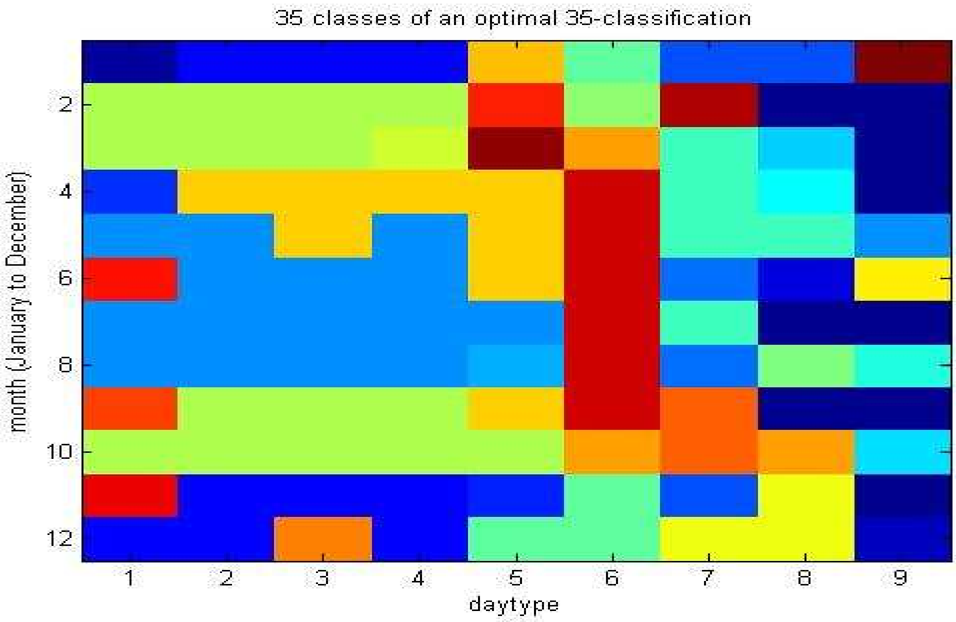

figure 18: an optimal calendaric 35-classification

Figures 17 and 18 show the extreme classifications (higher entropies) for k=4 and k=35. Calendaric interpretations are still obvious.

## 5.3 The simulated intraday price-curves

We simulated innerday price-curves by the simulation process as described in section 2.4, figure 10, using an optimal 15-clustering. We simulated innerday price-curves by sampling for a given daytype-month from the clusters each time just one historical price-curve. This sampling occurs according to the empirical cluster distribution of the classification. Results for the year 2006 are compared with normalized innerday price-curves from the EEX, and from scenarios simulated by our price models described in section 2: the ARMA and the Pilipovic models. Figure 19 shows the comparison of the four sources of price-curves for Tuesdays, Wednesdays and Thursdays from May to September of 2006. It can be seen easily by visual inspection that the innerday variance of the ARMA and the Pilipovic models is much higher than that of the EEX, while the simulation by clustering is closer to reality. Figure 20 shows the same type of results for Saturdays of the same period, again with a much higher variance of the two price traditional models, while our model based on clustering demonstrates a more realistic behaviour.

## 5.4 The complete spot-price scenarios

Figures 21 and 22 shows 3 different scenarios simulated by the overall simulation model as described in section 2.4, figure 11. Remind that the volatility of the daily mean prices have been simulated by an ARMA(8,8)-model. The outliers have been simulated by the model described in section 2.1c. Comparing these scenarios with the EEX spot price during 2006, it is evident that the stochastic components (volatility and outliers) are not satisfactory yet. In particular, it is obvious that the simulated long term volatility does not match the real volatility. This deficiency leads to unsatisfactory results in optimization tasks using stochastic programming, as has been analyzed quantitatively in [8], and is a future topic for further research.

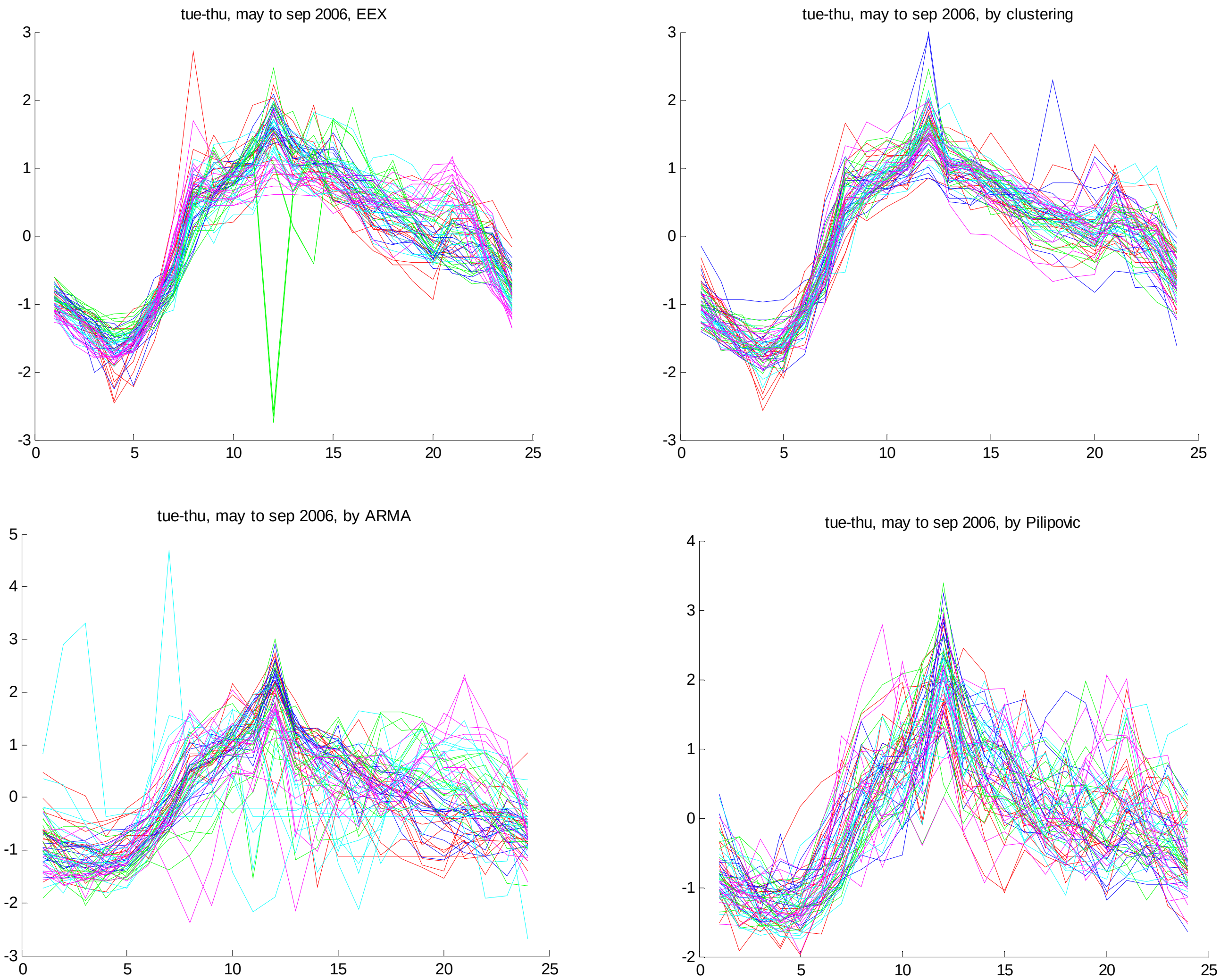


figure 19: price-curves from the EEX, simulated by clustering, by ARMA and by Pilipovic models: Tue, Wed, Thu of 05-09/2006. (take care of different scalings in *y* axis!)

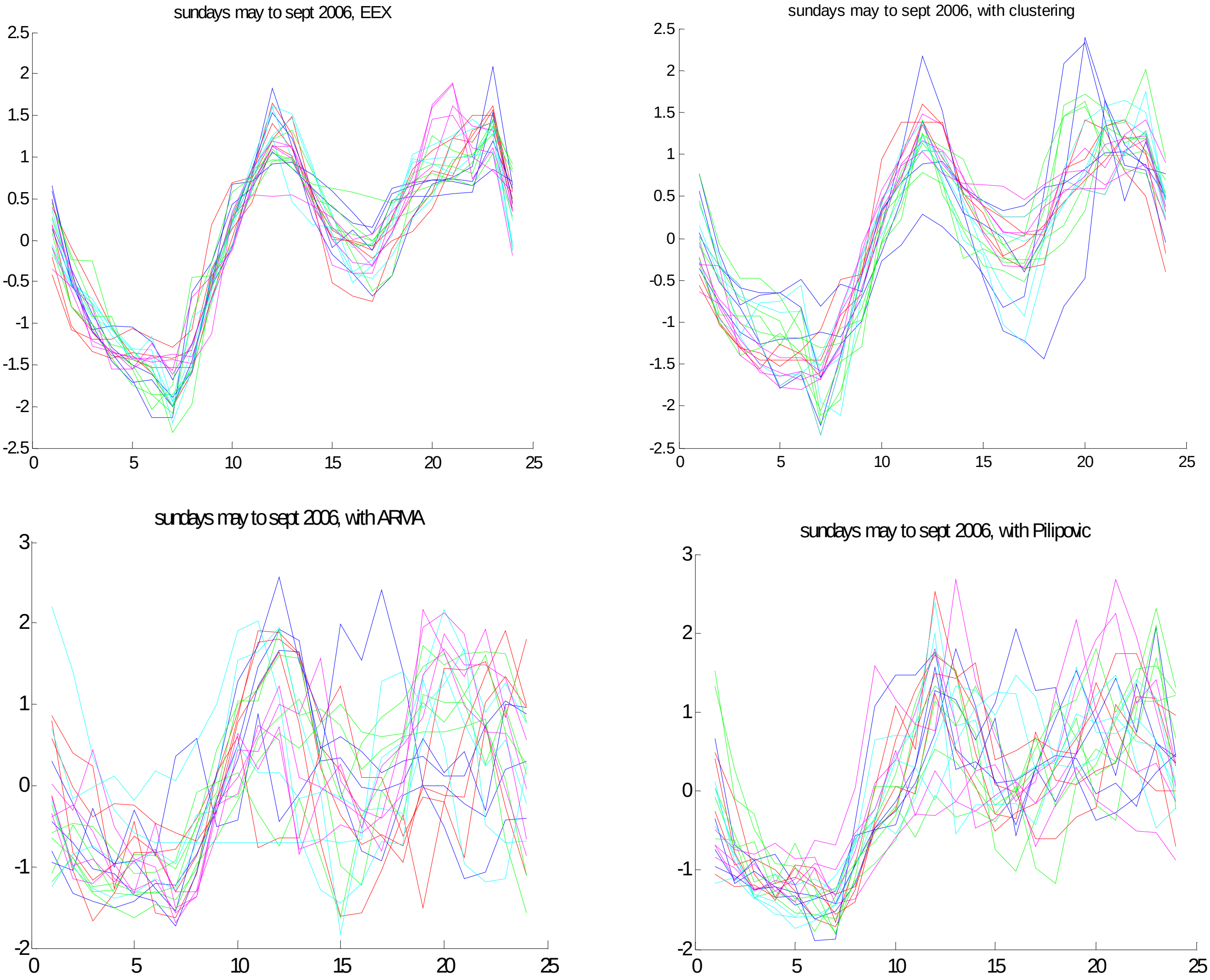


figure 20: price-curves from the EEX, simulated by clustering, by ARMA and by Pilipovic models: Sundays of 05-09/2006 (take care of different scalings in *y* axis!)

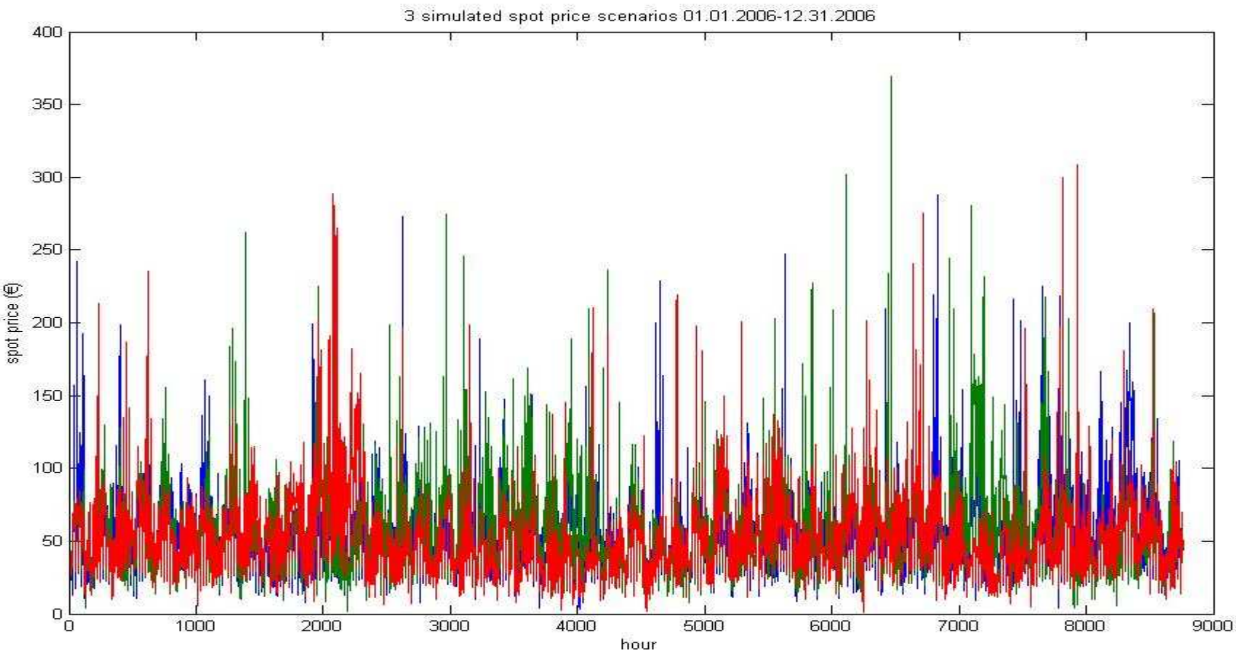


figure 21: 3 simulated spot-price scenarios with innerday-prices by clustering: year 2006

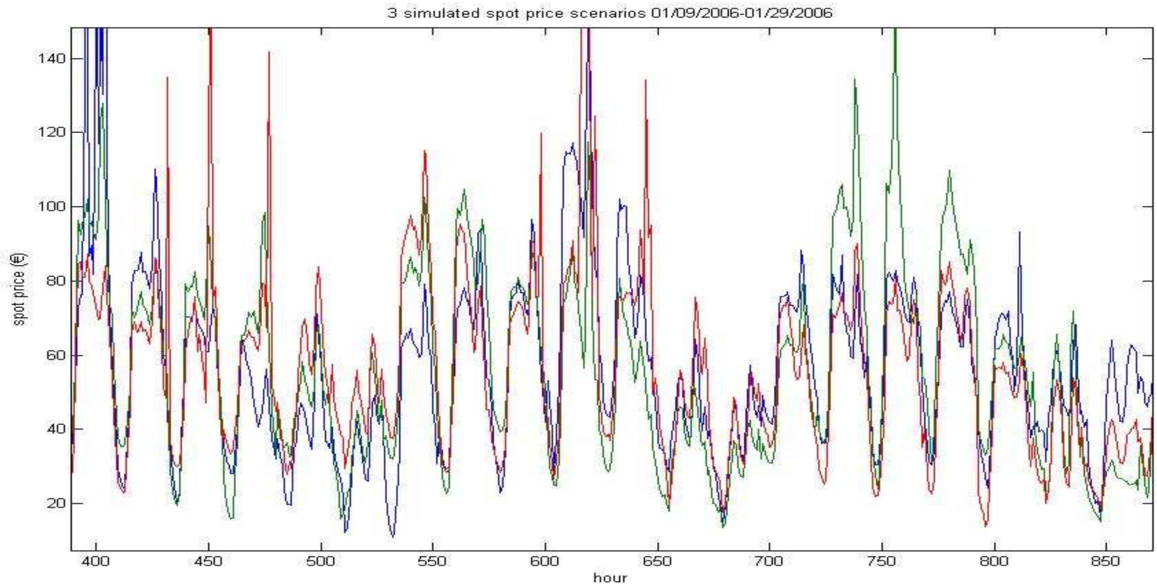


figure 22: 3 simulated spot-price scenarios with innerday-prices by clustering: 3 weeks

# 6 Conclusion and Outlook

A new model and simulation technique for innerday price-curves of the spotmarket at the energy exchange EEX has been presented. It is based on a clustering of normalized, historical price-curves. An optimal clustering and a corresponding optimal calendaric reference classification are used for sampling historical price-curves according to the empirical distribution of the classification within the clustering. The complex search for an optimal calendaric classification has been solved in a satisfactory way by a genetic algorithm which has been presented in detail and compared to an ad-hoc algorithm. The simulated price-curves are superior to those simulated by classical price models which we compared our model to. In particular, it demonstrates a more realistic innerday variance than those. It has been shown how our model fits into an overall price model which has been presented in detail.
Further research has to be done in order to improve this latter model, since it shows deficiencies in the stochastic component modelling long-term volatility of the spotmarket prices. This latter type of uncertainty is particularly important for energy providers since it is the source of risk in energy trading in mid- to long-term portfolio optimization.

**Acknowledgement:**
This work has been done in collaboration with the Vorarlberger Kraftwerke AG (VKW), and has been funded by a FHPlus grant of the FFG of the Austrian Government.

## Literatur:

[1] M. Burger, B. Klar, A. Müller, G. Schindlmayr: A spot market model for pricing derivatives in electricity markets. In: Quantitative Finance, 4, 109–122. Routledge 2006

[2] D. W. Bunn, N. Karakatsani: Forecasting electricity prices, London Business School Working Paper, 2003.

[3] H. Schmoeller: Modellierung von Unsicherheiten bei der mittelfristigen Stromerzeugungs- und Handelsplanung. Dissertation. Aachener Beiträge zur Energieversorgung Band 103, 2005

[4] D. Pilipovic: Energy Risk: Valuing and Managing Energy Derivatives. McGraw-Hill 1997

[5] P.-N. Tan, M. Steinbach, V. Kumar: Introduction to Data Mining. Addison-Wesley 2005

[6] Kaufman L., and P. J. Rousseeuw, Finding Groups in Data: An Introduction to Cluster Analysis, Wiley, 1990

[7] D.E. Goldberg: Genetic Algorithms for Search, Optimization and Machine Learning. Addison-Wesley 1989

[8] M.C. Steinbach, H. Vollbrecht: Efficient Stochastic Programming Techniques for Electricity Swing Options. (Accepted in) J. Kallrath, P. Pardalos: Optimization in the Energy Industry, Springer 2008

**Weitere Arbeiten**
**Forschungszentrum Prozess- und Produkt-Engineering**

**ANWENDUNGEN**
Kurzfristige Prognose des Stromverbrauchs in Vorarlberg auf Stunden- und Viertelstundenbasis
Thomas Steinberger, 2004

Weiterbildungs- und Qualifizierungsbedarf kleinerer und mittlerer Unternehmen in Vorarlberg bezüglich Prozess- und Projektmanagement, Führung, Strategie und Innovationsmanagement
Markus Reichart, Julia Schneider, Isabella Gratzer, 2004

Netzwerke für Innovationen
Martin Meusburger, Markus Reichart, Karin Feurstein, 2005

Neue Technologien im Produktinnovationsprozess
Julia Schneider, Markus Reichart, 2005

Bezug von externen Leistungen in der Produktentwicklung Aktueller Stand - Trends - Verbesserungspotenziale
Julia Schneider, 2005

project orientation [vorarlberg]
Martin Meusburger, Markus Reichart, Bratislav Veljovic, 2005

project orientation [vorarlberg II]
Martin Meusburger, Markus Reichart, Stefan Fink, 2006

Adverse Inter-Risk Diversification Effects for FX Forwards[1]
Thomas Breuer, Martin Jandacka, 2007

Optimierung eines Vertrages zum variablen Strombezug
Hans Vollbrecht, 2007

Szenarioanalyse mit unvollständiger Information: Beispiel Pflegekostenmodell Vorarlberg
Thomas Breuer, Martin Herburger, Manfred Hellrigl, Bertram Meusburger, Ruth Weiskopf, Falko Wilms, 2007

A Review and Redefinition of Knowledge Work from a Management-Oriented Perspective
Rainer Erne, Sonja Sackmann, 2006

Was bedeutet Produktivität in der Produktentwicklung und welche Prozessstandards sind dafür wirksam?
Thomas Breuer, Rainer Erne, 2007

Strategisches Management
Falko E. P. Wilms, 2008

**METHODEN**
Identifying Worst Case Scenarios of Security Portfolios with Quasi-Random Search Algorithms
Thomas Breuer, Filip Pistovcak, 2004

A General Noise Model and Its Effects on Evolution Strategy Performance
Hans-Georg Beyer, Dirk V. Arnold, 2004

Using Quasi-Monte Carlo Scenarios in Risk Management
Thomas Breuer, Filip Pistovcak, 2004

An Explicit Characterization of Calogero-Systems
Fritz Gesztesy, Karl Unterkofler, Rudi Weikard, 2004

Reliability of old and new Ventricular Fibrillation Detection Algorithms for Automated External Defibrillators
Anton Amann, Robert Tratnig, Karl Unterkofler, 2005

Towards an Integrated Measurement of Credit and Market Risk
Thomas Breuer, Martin Jandacka, Gerald Krenn, 2005

Umgang mit Szenarien
Falko E. P. Wilms, 2005

Umgang mit unscharfen Informationen
Falko E. P. Wilms, 2005

A new ventricular fibrillation detection algorithm for automated external defibrillators
Anton Amann, Robert Tratnig, Karl Unterkofler, 2005

Removal of Resuscitation Artefacts from Ventricular Fibrillation ECG Signals Using Kalman Methods
Anton Amann, M. Baubin, Klaus Rheinberger, Karl Unterkofler, 2005

Detecting ventricular fibrillation by time-delay methods
Anton Amann, Robert Tratnig, Karl Unterkofler, 2005

Der Einsatz vagen Wissens bei Entscheidungsprozessen
Thomas Breuer, Hans Vollbrecht, Andreas Juen, 2005

Szenarien sind Systeme
Falko E. P. Wilms, 2006

Portfolio Selection with Transaction Costs under Expected Shortfall Constraints
Thomas Breuer, Martin Jandacka, 2006

An optimization model for storing and delivering spare parts
Hans-Georg Beyer, Stefan Röhl 2007

Folgenabschätzung von Massnahmen
Falko E. P. Wilms, 2007

Regulatory Capital for Market and Credit Risk
Interaction: Is Current Regulation Always Conservative?
Thomas Breuer, Martin Jandacka, Klaus Rheinberger, Martin Summer, 2007

An Intraday Spotmarket-Price Model based on Clustering
Hans Vollbrecht, 2008

Fachhochschule Vorarlberg
Forschungszentrum
Prozess- und Produkt-Engineering
Hochschulstraße 1
A-6850 Dornbirn

T +43 (0)5572 792 7100
F +43 (0)5572 792 9510

www.fhv.at/res/ppe

Fachhochschule Vorarlberg
Forschungszentrum
Prozess- und Produkt-Engineering