Proceedings of the 7th Symposium FENS, Lublin, May 14-17, 2014

Independent Component Analysis for Ensemble Predictors with Small Number of Models

R. SZUPILUK^a, T. ZĄBKOWSKI^{b,*} AND K. GAJOWNICZEK^{b,c}

^aWarsaw School of Economics, Niepodleglosci 162, 02-554 Warsaw,

^bWarsaw University of Life Sciences, Faculty of Applied Informatics and Mathematics,

Nowoursynowska 159, 02-776 Warsaw, Poland

^cSystems Research Institute, Polish Academy of Sciences, Newelska 6, 01-447 Warsaw, Poland

The article presents independent component analysis (ICA) applied to the concept of ensemble predictors. The use of ICA decomposition enables to extract components with particular statistical properties that can be interpreted as destructive or constructive for the prediction. Such process can be treated as noise filtration from multivariate observation data, in which observed data consist prediction results. As a consequence of the ICA multivariate approach, the final results are combination of the primary models, what can be interpreted as aggregation step. The key issue of the presented method is the identification of noise components. For this purpose, a new method for evaluating the randomness of the signals was developed. The experimental results show that presented approach is effective for ensemble prediction taking into account different prediction criteria and even small set of models.

DOI: 10.12693/APhysPolA.127.A-139

PACS: 05.45.Tp, 05.40.Ca, 07.05.Kf, 07.05.Mh

1. Introduction

In this paper we develop an independent component analysis (ICA) approach for ensemble predictions. Its main idea is based on decomposition of the prediction results into underlying independent components [1]. Some of these components may be associated with the true value prediction and some of them can be treated as noise or interference. Elimination of noises, termed as destructive components, should result in prediction improvement. The process can be perceived as data filtration aimed to reveal hidden noises in a way that is typical for blind source separation techniques [2]. Standard filtering using ICA involves the components separation into source signals (separation step), the identification and elimination of noise components, and then inverse procedure with respect to separation (remixing step). The use of ICA filtration for prediction problem in which the historical data are available (target values) enables, while remixing, to develop more general form of mixing system which can be the MLP neural network.

The term ensemble or aggregation is a consequence of the fact that the final result is a combination of individual results from different models. Combination methods have gained more ground in the forecasting literature, largely due to the strength of the empirical evidence suggesting that these methods can perform better than alternatives based on forecasts from a single model [3–6]. Unlike the other popular ensemble methods like bagging or boosting there are no assumptions to both, the form of aggregated models (e.g. decision trees, regression models) and the criteria for model assessment (e.g. mean square error, mean absolute percentage error, misclassification error). In other words, we can aggregate models (more specifically, the results of their prediction) regardless to specific criterion and the modeling technique. Another advantage of the presented method is its effectiveness of improving the prediction results with a relatively small number of aggregated models.

One of main problems in this concept is proper destructive component identification and its transformation. In the simplest case for small number of models we can perform full computational search with each basis component elimination and checking its impact for final prediction. Unfortunately, in practice we can't expect that our basis component are pure destructive or constructive, especially taking into account limited number of models. Therefore, we should apply filtration processes rather than elimination, which can be realized as supervised learning.

An important and utilitarian dimension of our work concerns its application for practical business problem which was electricity load forecasting on Polish market. Forecasting electricity demand is an important issue from an economic point of view [7, 8], due to the fact that direct financial incentives are related to the characteristics of the energy market, on which the possibility to store the electricity is very limited. Therefore, any mismatch between the size of demand and supply results in tangible losses. Over estimation, due the storage problems, causes its irretrievable loss, while under estimating leads to urgent purchase on balancing market on which the prices are higher.

In this circumstances our concept delivers a reliable tool for accurate short term forecasting on electricity market, which is crucial for the economic efficiency of

^{*}corresponding author; e-mail: tomasz_zabkowski@sggw.pl

power sector entities, since it is associated with costly transaction realized on balancing market.

2. Prediction results improvement

We assume that after learning various models we have a set of prediction results. We collect particular prediction results $x_i(k)$, i = 1..., m, in one multivariate variable $\mathbf{x} = [x_1(k), ..., x_m(k)]^T$. Then, we assume that prediction result is a mixture of the m latent components: constructive s_j , $j \in D_1$ and destructive s_i , $j \in D_2$, where $D_1 \cup D_2 = \{1, 2, ..., m\}$, $D_1 \cap D_2 = \emptyset$ and $D_1 \neq \emptyset$. Next, we assume that the prediction result are the linear combination of latent components

$$\mathbf{x}(k) = \mathbf{A}\mathbf{s}(k) = \mathbf{A}\left[s_1(k), ..., s_m(k)\right]^T,$$
(1)

where $\mathbf{s}(k) \in \mathbb{R}^m$, matrix $\mathbf{A} = [a_{ij} \in \mathbb{R}^{m \times m}$ represents the mixing system. The relation (1) stands for decomposition of prediction results \mathbf{x} into latent components vector \mathbf{s} and mixing matrix \mathbf{A} . If the destructive part of the signal is removed (the signals are replaced with zero, $s_i(k)$ for $i \in D_2$) and the constructive components are mixed back, the modified prediction results $\hat{\mathbf{x}}(k)$ will be improved (or filtered): $s_j, j \in D_1$

$$\hat{\mathbf{x}}(k) = \mathbf{A}\hat{\mathbf{s}}(k) = \hat{\mathbf{A}}\mathbf{s}(k), \qquad (2)$$

where $\hat{s}_j(k) = s_j(k)$ for $j \in D_1$, $\hat{s}_j(k) = 0$ for $j \in D_2$,
and $\hat{\mathbf{A}} = [\hat{a}_{ij}]$, where $\hat{a}_{ij} = a_{ij}$ for $j \in D_1$, $\hat{a}_{ij} = 0$ for
 $j \in D_2$.

The crucial point of the above concept is proper A and S estimation. It is difficult task because we don't have information which decomposition is most adequate. The most adequate methods to solve the first problem seem to be the blind signal separation (BSS) methods. Let's us note that, the ICA approach represents chosen data decomposition from wide set of blind signal separation methods, but other techniques like smooth component analysis (SmCA), sparse component analysis (SCA), principal component analysis (PCA), Nonnegative Matrix Factorisation or Tensor Decompositions can be applied either [2, 9–11].

As a consequence of incomplete purity of the estimated components some extension to the re-composition concept must be introduced, in which, instead of the elimination of a component that was found to be disruptive, we change its impact by reducing the weight or applying the filtration. These operations can be realized by introducing a complex, adaptive re-composition system with its optimization (learning). In this way the system gains some flexibility as destructive impact of the component does not have to be zero in order to obtain improved estimates. Assuming that, the basis latent components might be not pure so their impact should have weight other than 0 or 1, we can try to find the better mixing system than described by **A**. The new mixing system can be formulated in more general form than linear, e.g. we can take MLP neural network as the mixing system

$$\hat{\mathbf{X}} = \mathbf{g}^{(2)} \left(\mathbf{B}^{(2)} \left[\mathbf{g}^{(1)} \left(\mathbf{B}^{(1)} \mathbf{S} + \mathbf{b}^{(1)} \right) \right] + \mathbf{b}^{(2)} \right), \qquad (3)$$

where $\mathbf{g}^{(i)}(.)$ is a vector of nonlinearities, $\mathbf{B}^{(i)}$ is a weight matrix and $\mathbf{b}^{(i)}$ is a bias vector respectively for *i*-th hidden layer, i = 1, 2. The first weight layer will produce results related to (3) if we take $\mathbf{B}^{(1)} = \hat{\mathbf{A}}$. Due to some nonlinearity introduced by the first and the second layer, the proposed mixing system gained some flexibility in comparison to the linear form. If we learn the whole structure starting from system described by $\hat{\mathbf{A}}$ with initial weights of $\mathbf{B}^{(1)}(0) = \hat{\mathbf{A}}$, we can expect the results will be improved.

3. Destructive components as noisy signal identification

In the prediction improvement procedure, after the independent signals are identified the next step is to distinguish the informative elements from the noises and disturbances [12, 13]. Therefore, we need to classify latent components as constructive or destructive. In general, this can be difficult task because the obtained components might be somewhere between constructive and destructive. It means that particular component can have constructive impact on one model and destructive on the other. Also, there may exist components destructive as a single but constructive in a group. All these effects can appear due to many reasons like improper assumption about the linear structure for model results, transformation choice or estimation algorithm features. Therefore, we have to check the impact of almost all components and their combinations on final results. This can be done in the following way.

The simplest, but somehow trivial method is to eliminate particular components combinations and combining the rest by mixing matrix. The main disadvantage of this method is the curse of dimensionality because the large number of models gives large number of basis signals and huge number of basis signals combinations.

The other method for component classification can be based on some theoretical assumptions and is addressed for the prediction models optimized under MSE error and using ICA or PCA decompositions. Yet another approach is based on the assumption that the destructive components may be interpreted in terms of noise, for which we can deliver mathematical characteristics describing them. This method is particularly appropriate if the destructive components can be modeled as some stochastic processes including white noise. However, in practice, we rarely deal with white noises, and the components are the mixture of random and deterministic signals. As a result, we aim to analyze the degree of randomness of each component rather than classifying them into white noise or deterministic signals. Therefore, we need some measure to compare the signal with the noise and some procedure to evaluate the these results.

Identification of destructive components may be considered as benchmarking the signal to the noise using a given similarity measure. The choice of the appropriate measure is an important issue. It may include application of correlational approach, p-norms approach or divergence functions. The other approach is related to signal randomness assessment which includes analysis of such signal characteristics as volatility, smoothness or the existence of internal relations. Among the other traditionally used techniques, there are analysis of variance, R/S and Hurst exponent analysis, or methods based on the autocorrelation functions [14, 15].

In our approach, to assess the similarity of the signal to the noise and its randomness we propose a general scheme for comparative analysis, described with the following formula:

$$\varphi_{1,2}(y_1, y_2) = 2 \frac{u(y_1 + y_2)}{u(y_1) + u(y_2)},\tag{4}$$

where u(y) is a measure of randomness assessment. The choice of this measure may be related to the nature of the problem, or it can be to some extent automatic and arbitrary. It is also possible to test several different functions u(y) such as variance of the signal u(y) = var(y) or Hurst's exponent of the signal u(y) = H(y).

If one of the signals is assumed to be the reference noise then the interpretation of (4) is relatively simple and intuitive and for chosen functions u(y) take some constant value. Having y_1 and y_2 with similar statistical characteristics, e.g. both of them are Gaussian noise, then for Hurst exponent the measure is equal to 1. In other case, we would get the values different than 1. In addition, by selecting an appropriate form of u(y), the measure may be largely independent of the distribution and this is evident for the Hurst exponent.

To explore directly the temporal characteristics of the noise signals we propose following variability measure:

$$u(y) = \frac{\frac{1}{N} \sum_{k=2}^{N} |y(k) - y(k-1)|}{\rho(\max(y) - \min(y))},$$
(5)

where k is observation index and $\rho(z)$ is zero indicator, for $\rho(z) = 1$, z = 1 and $\rho(z) = z$ where $z \neq 1$. Although other potential delays with higher lags than just y(k-1) are possible we propose such structure as most natural and easy to apply. The measure (5) has simple interpretation: it is maximal when the changes in each iteration step are equal to range (maximal change), and is minimal when data are constant. The possible values are ranging from 0 to 1. Function $\rho(z)$ was introduced to avoid dividing by zero.

In practical applications, it may be necessary to use some smoothness measure as the objective function in optimization problems, thus approximating the absolute value function $\log(\cosh(u))$, can be obtained by a modified form of a differentiable smoothness measure of the form:

$$u(y) = \frac{\frac{1}{N} \sum_{k=2} \log(\cosh(y(k) - y(k-1)))}{\log(\cosh(\rho(\max(y) - \min(y))))},$$
(6)

To maintain proportional scales of numerator and denominator, the expression in the denominator (6) was also converted by $\log(\cosh(u))$ non-linearity. If we assume that the signal was normalized to the interval (0, 1)then the expression (6) takes the form:

$$\varphi_{1,2}(y_1, y_2) = (7) \\
\frac{\sum_{k=2}^{N} \log(\cosh(y_1(k) + y_2(k) - y_1(k-1) - y_2(k-1))))}{\sum_{k=2}^{N} (\log(\cosh(y_1(k) - y_1(k-1))) + \log(\cosh(y_1(k) - y_1(k-1))))}.$$

In case of multiple signals, their mutual similarity $\varphi_{i,j}(y_i, y_j)$ may be presented in a matrix form:

$$\Phi = \begin{bmatrix} \varphi_{1,1} & \cdots & \varphi_{1,m} \\ \vdots & \ddots & \vdots \\ \varphi_{n,1} & \cdots & \varphi_{n,m} \end{bmatrix}.$$
(8)

Additional characteristics which can be explored using the above measures (5)-(7) are directly related to the existence of the temporal structure. In case of random noise the mixing procedure should not significantly change the value of volatility (variability, smoothness) measure and consequently the similarity measure. However, for signals in which the observation order is important then the random mixing of signal samples should significantly change the values of measures (5)-(7).

To illustrate how the measure works we will consider the set of source signals, as presented in Fig. 1a. Figure 1b shows the same signals but the observation order was randomly assigned. Signals were also decomposed by ICA algorithm and therefore, they are also mutually decorrelated.

The matrices corresponding to the signals presented in Fig. 1 have the following values for source signals (Φ_1) and the signals with observation order randomly assigned (Φ_2) :

	1	0.8220	0.9282	0.7649	1.5188	1.5814	1
$\Phi_1 =$	0.8220	1	0.8030	0.7998	1.8006	1.8741	
	0.9282	0.8030	1	0.8643	1.7822	1.9042	
	0.7649	0.7998	0.8643	1	1.8291	1.7849	
	1.5188	1.8006	1.7822	1.8291	1	0.9902	
	1.5814	1.8741	1.9042	1.7849	0.9902	1	
$\Phi_2 =$	1	0.7308	0.7136	0.7358	0.5878	0.5437	
	0.7308	1	0.7121	0.8286	0.6094	0.5387	
	0.7136	0.7121	1	0.7376	0.5821	0.5358	
	0.7358	0.8286	0.7376	1	0.7375	0.6912	•
	0.5878	0.6094	0.5821	0.7375	1	0.9452	
	0.5437	0.5387	0.5358	0.6912	0.9452	1	

As we can see, according the matrix Φ_1 , the last two noise signals (values in bold) are identified as these which have the highest similarity. It is also apparent that the value of similarity did not changed substantially after mixing the signals, as presented in matrix Φ_2 (in bold). It should be noted that because of the fact that the signals are decorrelated there is no possibility to determine their similarity using correlation approach. Also, calculating the distance measured by the chosen p-norm does not give unambiguous results, too.



Fig. 1. Source signals (a) and the signals with observation order randomly assigned (b).

4. Algorithm outline

Based on the proposed aggregation concept and having the method for components classification, we can formulate aggregation algorithm as follows.

- 1. Store prediction results x_i in $\mathbf{x} = [x_1, x_2, ..., x_m]^T$ and decompose the models results \mathbf{x} , into independent latent components $\mathbf{s}(t) = [s_1(t), s_2(t), ..., s_n(t)]^T$ by the chosen ICA algorithm;
- 2. Classify latent components as destructive $s_i(t) = \tilde{s}_i(t)$ or constructive $s_i(t) = \hat{s}_i(t)$ on prediction results. For MSE criterion it can be made by correlation analysis, see Sect. 3, or in general case by computational test with inverse to ICA process.
- 3. Learn MLP neural network with latent components as an input to obtain effective mixing system, the initial weights matrix for the first layer is inverse matrix to matrix from ICA decomposition, where columns related to particular destructive components are replaced by vectors of zeros.

5. Number of aggregated models — methodological issues

The models aggregation process assumes itself somehow the increase in the complexity of the solution. It is desirable when the effects of aggregation are satisfactory for the researcher in terms of prediction results improvement. It seems that the greatest value added can be achieved by combining different forecasting techniques. In practice, the aggregation concerns models of similar type or structure, using different samples of the training dataset (i.e. bootstrap aggregating) and finally the most popular way to integrate this information is averaging (for regression) or voting (for classification) [4]. This is due to the fact that existing methods for models aggregation (ensemble methods) are addressed to a relatively different situation, in which we have a large number of results obtained with the same method.

Based on these findings, the article highlights a small number of aggregated models. This is designed to show the difference between this method and the typical aggregation methods like boosting and bagging. These latter ones aggregate models with relatively restrictive conditions to the form of aggregated models. Therefore, we deal with some averaging of the results generated by the same model (in terms of architecture) but learned on different subsets rather than the actual use of information coming from different approaches. What is important, these techniques have relatively strong assumptions about the statistical characteristics of the models errors which means strong assumptions about their nature and the way these models are built.

In our approach, in contrast to other methods, the forecast errors are treated in terms of physical disturbance rather than purely mathematical value of the predetermined criterion. This means that the removal of such a true distortion can improve any error criterion. Therefore, there is a question of how can we separate these noises. This is the issue which has been widely developed on the basis of general filtration and signal separation methods. However, the classical filtration/separation assumes spectral variation of filtered signals, what, in practice, significantly limits the number of problems possible to be solved. The situation changed significantly with the development of methods for blind signals separation, allowing for separation and consequently the elimination of signals based on other characteristics than the frequency. The only drawback is the necessity to have multiple signals/observations, since blind signal separation problem and its solution is considered in multidimensional terms.

Consequently, the elimination of noises from the prediction results using blind signal separation methods requires two separate prediction results and the final result is a combination (linear or nonlinear) of improved basic models, which is the consequence of the adopted method. Therefore, aggregation even of two neural models with different structures is possible, what so far, used to be associated with a simple averaging.

6. Practical experiment

In this paragraph, we will consider the short term forecasting problem of electricity demand. For this reason a 24-hour ahead forecast was prepared based on historical data about energy consumption from Polish power system. The data set included 86400 observations (hourly data) covering time span between years 1988 and 1998, please see Fig. 2 for data characteristics.



Fig. 2. Hourly load observed during in 1997 (vertical axis shows normalized load and horizontal axis shows subsequent observations).

The available variables to create the forecast included energy demand from the last 24 hours and calendar variables such as month, day of the month, day of the week, and holiday indicator. In this case, we don't expect to create a single and precise descriptive model, but rather we focus, as in the data mining methodology, on testing various methods and models, which are then aggregated. For this reason, there were six neural networks models build with different learning algorithms and one hidden layer with 12, 18, 24, 27, 30, 33 neurons respectively and labeled as M1:MLP12, M2:MLP18, M3:MLP24, M4:MLP27, M5:MLP30, M6:MLP33.

The experiment was conducted for various types of decomposition and different algorithms realizing it. We proposed three ICA algorithms such as: FPICA, SANG and JADE [16]. The test results with the MAPE criterion are presented in Table. The best results were typed in bold.

TABLE

The results of BSS aggregation measured with MAPE error.

MAPE $\times 10^{-3}$		M1	M2	M3	M4	M5	M6
Primary models		23.9431	23.5021	23.6750	23.9850	24.1374	23.5776
	FPICA	23.8712	23.0011	23.5819	23.6068	24.0282	23.5715
ICA	SANG	24.1180	22.3682	23.8860	24.0345	24.2017	23.7412
	JADE	24.1081	23.5996	23.7717	23.7653	22.5332	22.3239

The best results were obtained for the JADE algorithm, and the improvement rate observed reached 5% for model M6, calculated as the best model after aggregation to the best primary model (MAPE error $\times 10^{-3}$ 22.3239 vs. 23.5021).

Identification of noise components was made based on Φ matrix values. As the reference noise a Gaussian noise was proposed. The set of reference signals, among which the first six are the extracted components, and the seventh signal is the Gaussian noise, is presented below:

	1.0000	1.3028	1.2878	1.3337	1.0882	1.1248	1.2519
	1.3028	1.0000	0.9787	0.9627	1.0812	0.9945	0.8347
	1.2878	0.9787	1.0000	0.7792	1.3205	1.0916	0.8182
$\Phi =$	1.3337	0.9627	0.7792	1.0000	0.8768	1.0478	0.8217
	1.0882	1.0812	1.3205	0.8768	1.0000	0.9130	0.7008
	1.1248	0.9945	1.0916	1.0478	0.9130	1.0000	0.9305
	1.2519	0.8347	0.8182	0.8217	0.7008	0.9305	1.0000

The values of matrix Φ in bold present the similarity of the individual components to the reference noise.

The other algorithms delivered slightly worse results. The SANG algorithm achieved improvement rate of 4.8% for model M2, calculated as the best model after aggregation to the best primary model (MAPE error $\times 10^{-3}$ of 22.3682 vs. 23.5021). Finally, FPICA algorithm reached improvement of 2.1% for model M2 (MAPE error $\times 10^{-3}$ of 23.0011 vs. 23.5021).

The second part of the experiment was proposed to

illustrate how the prediction improvement depends on the number of models used for BSS aggregation. The procedure was as follows.

Having set of six models we create subsets of n models, where n = 2, 3, 4, 5, 6. For a given set of n models we make ICA decomposition using FPICA, SANG and JADE algorithms, and then we calculate the average value of the best results over all nelement models subsets. In particular, for six models we deal with only one set of models (M1-M2-M3-M4-M5-M6). For five models we deal with six subsets (M1-M2-M3-M4-M5, M1-M2-M3-M4-M6, M2-M3-M4-M5, M1-M2-M3-M4-M5, M1-M2-M3-M5-M6), etc. The experiment was performed for MAPE and MSE error, please see Fig. 3a for details of MAPE and Fig. 3b for MSE.

We observed that with increasing number of aggregated models the efficiency of aggregation was improved. In terms of MAPE, we could observe growing effectiveness of the improvement procedure with increasing number of models, for JADE and SANG algorithms. Although, we observed a slight drop in improvement ratio for FPICA algorithm using 5 models for aggregation. Nevertheless, even a small number of aggregated models, using just two or three of them, delivered noticeable improvement which may be considered relevant from a practical point of view. In terms of MSE, there was clear dependence between the number of aggregated models and effectiveness of the procedure.



Fig. 3. Prediction improvement observed for different number of aggregated models and measured with (a) MAPE, (b) MSE.

7. Results and concluding remarks

In this article we aim to develop new aggregation method applicable for predictive models. Its main idea is based on decomposition of the prediction results into underlying independent components and identification of components showing strong similarity to the noise.

The experiments were run both, on illustrative simulated data and the real problem of electricity forecasting, and confirmed the effectiveness of the proposed approach. Based on the electricity load data we showed that presented approach is effective for ensemble prediction taking into account MAPE error criteria and different number of aggregated models. As a result we could benefit of about 5% of MAPE reduction (best primary model vs. best model after decomposition) which may be regarded as significant improvement for power sector entities to maintain high efficiency in terms of balancing the market better.

It should be noted that the proposed method is adequate for a small number of aggregated models which meets typical requirement for ensemble methods.

Acknowledgments

The study is cofounded by the European Union from resources of the European Social Fund. Project PO KL "Information technologies: Research and their interdisciplinary applications", Agreement UDA-POKL.04.01.01-00-051/10-00.

References

- R. Szupiluk, P. Wojewnik, T. Ząbkowski, *Lect. Notes Comput. Sc.* 6594, 206 (2011).
- [2] P. Comon, Ch. Jutten, Handbook of Blind Source Separation: Independent Component Analysis and Applications, Academic Press, 2010.
- [3] L. Breiman, Mach. Learn. 24, 123 (1996).
- [4] R.T. Clemen, Int. J. Forecasting 5, 559 (1989).
- [5] J.H. Friedman, Ann. Stat. 19, 1 (1991).
- [6] A. Timmermann, Handbook of Economic Forecasting 1, 135 (2006).
- [7] J.W. Taylor, L.M. de Menezes, P.E. McSharry, Int. J. Forecasting 22, 1 (2006).
- [8] R. Weron, Modeling and Forecasting Electricity Loads and Prices: A Statistical Approach, John Wiley, Chichester 2006.
- [9] A. Cichocki, JCMSI 7, 507 (2013).
- [10] G. Zhou, A. Cichocki, Q. Zhao, S. Xie, *IEEE Signal Proc. Mag.* **31**, 54 (2014).
- [11] R. Szupiluk, P. Wojewnik, T. Ząbkowski, *IJICIC* 10, 1435 (2014).
- [12] R. Szupiluk, Przegląd Elektrotechniczny (Electrotechnical Review) 86, 144 (2010).
- [13] R. Szupiluk, P. Wojewnik, T. Zabkowski, *Lect. Notes Artif. Int.* **6113**, 471 (2010).
- [14] H.E. Hurst, Trans. Am. Soc. Civil Engineers 116, 770 (1951).
- [15] E. Peters, Fractal market analysis, John Wiley, Chichester 1996.
- [16] A. Cichocki, S. Amari, Adaptive Blind Signal and Image Processing: Learning Algorithms and Applications, John Wiley, Chichester 2003.