5-25-2006

Performance Evaluation of the Self-Organizing Map for Feature Extraction

Yonggang Liu

Robert H. Weisberg
University of South Florida, weisberg@marine.usf.edu

C. N. K. Mooers

Performance evaluation of the self-organizing map for feature extraction

Yonggang Liu,[1] Robert H. Weisberg,[1] and Christopher N. K. Mooers[2]

Received 22 June 2005; revised 22 December 2005; accepted 3 February 2006; published 25 May 2006.

[1] Despite its wide applications as a tool for feature extraction, the Self-Organizing Map (SOM) remains a black box to most meteorologists and oceanographers. This paper evaluates the feature extraction performance of the SOM by using artificial data representative of known patterns. The SOM is shown to extract the patterns of a linear progressive sine wave. Sensitivity studies are performed to ascertain the effects of the SOM tunable parameters. By adding random noise to the linear progressive wave data, it is demonstrated that the SOM extracts essential patterns from noisy data. Moreover, the SOM technique successfully chooses among multiple sets of patterns in contrast with an Empirical Orthogonal Function method that fails to do this. A practical way to apply the SOM is proposed and demonstrated using several examples, including long time series of coastal ocean currents from the West Florida Shelf. With improved SOM parameter choices, strong current patterns associated with severe weather forcing are extracted separate from previously identified asymmetric upwelling/downwelling and transitional patterns associated with more typical weather forcing.

Citation: Liu, Y., R. H. Weisberg, and C. N. K. Mooers (2006), Performance evaluation of the self-organizing map for feature extraction, *J. Geophys. Res.*, *111*, C05018, doi:10.1029/2005JC003117.

1. Introduction

[2] Our understanding of ocean processes steadily improves with increasing information obtained from in situ and remotely sensed data [such as from moored Acoustic Doppler Current Profilers (ADCP), satellite sea surface temperature (SST), sea surface height (SSH), and Chlorophyll, and high-frequency radar sampling of surface currents], and numerical models. However, the percentage of data actually used is low, in part because of a lack of efficient and effective analysis tools. For instance, it is estimated that less than 5% of all remotely sensed images are ever viewed by human eyes [*Petrou*, 2004]. With the increasing quantity of data there is a need for effective feature extraction methods.

[3] Temporal and spatial averaging are the simplest methods, but it is difficult to define suitable time and length scales over which to average. For example, currents on a continental shelf may exhibit different isotropic or anisotropic behaviors depending on the processes and timescales involved [*Liu and Weisberg*, 2005]. Thus mean values may be very misleading.

[4] The Empirical Orthogonal Function (EOF) analysis technique is useful in reducing large correlated data sets into an often much smaller number of orthogonal patterns ordered by relative variance, with wide oceanographic and meteorological applications [e.g., *Weare et al.*, 1976; *Klink*, 1985; *Lagerloef and Bernstein*, 1988; *Espinosa-Carreon et al.*, 2004]. However, conventional EOF, as a linear method, may not be as useful in extracting nonlinear information [*Hsieh*, 2001, 2004].

[5] The Self-Organizing Map (SOM), based on an unsupervised neural network [*Kohonen*, 1982, 2001], appears to be an effective method for feature extraction and classification. It maps high-dimensional input data onto a low-dimensional (usually 2-d) space while preserving the topological relationships between the input data. Thousands of SOM applications are found among various disciplines [*Kaski et al.*, 1998; *Oja et al.*, 2002], including climate and meteorology [*Hewitson and Crane*, 1994, 2002; *Malmgren and Winter*, 1999; *Cavazos*, 2000; *Ambroise et al.*, 2000; *Hsu et al.*, 2002; *Hong et al.*, 2004, 2005] and biological oceanography [*Ainsworth*, 1999; *Ainsworth and Jones*, 1999; *Silulwane et al.*, 2001; *Richardson et al.*, 2002; *Hardman-Mountford et al.*, 2003]. Recent SOM applications include SST and wind pattern extractions from satellite data [*Richardson et al.*, 2003; *Risien et al.*, 2004; *Liu et al.*, 2006] and ocean current pattern extractions from moored ADCP data [*Liu and Weisberg*, 2005]. All of these applications suggest that the SOM may be useful for meteorological and oceanographic feature extraction. However, for meteorologists and oceanographers unfamiliar with neural network techniques, the SOM remains a "black box," with associated skepticism.

[6] Early evaluations of SOM performance focused on comparisons with other techniques, such as principal component analysis and k-means clustering [*Murtagh and Hernandez-Pajares*, 1995; *Kiang and Kumar*, 2001]. In an introduction to the SOM Toolbox [*Vesanto et al.*, 1999,

[1] College of Marine Science, University of South Florida, St. Petersburg, Florida, USA.
[2] Rosenstiel School of Marine and Atmospheric Science, University of Miami, Miami, Florida, USA.

0148-0227/06/2005JC003117$09.00

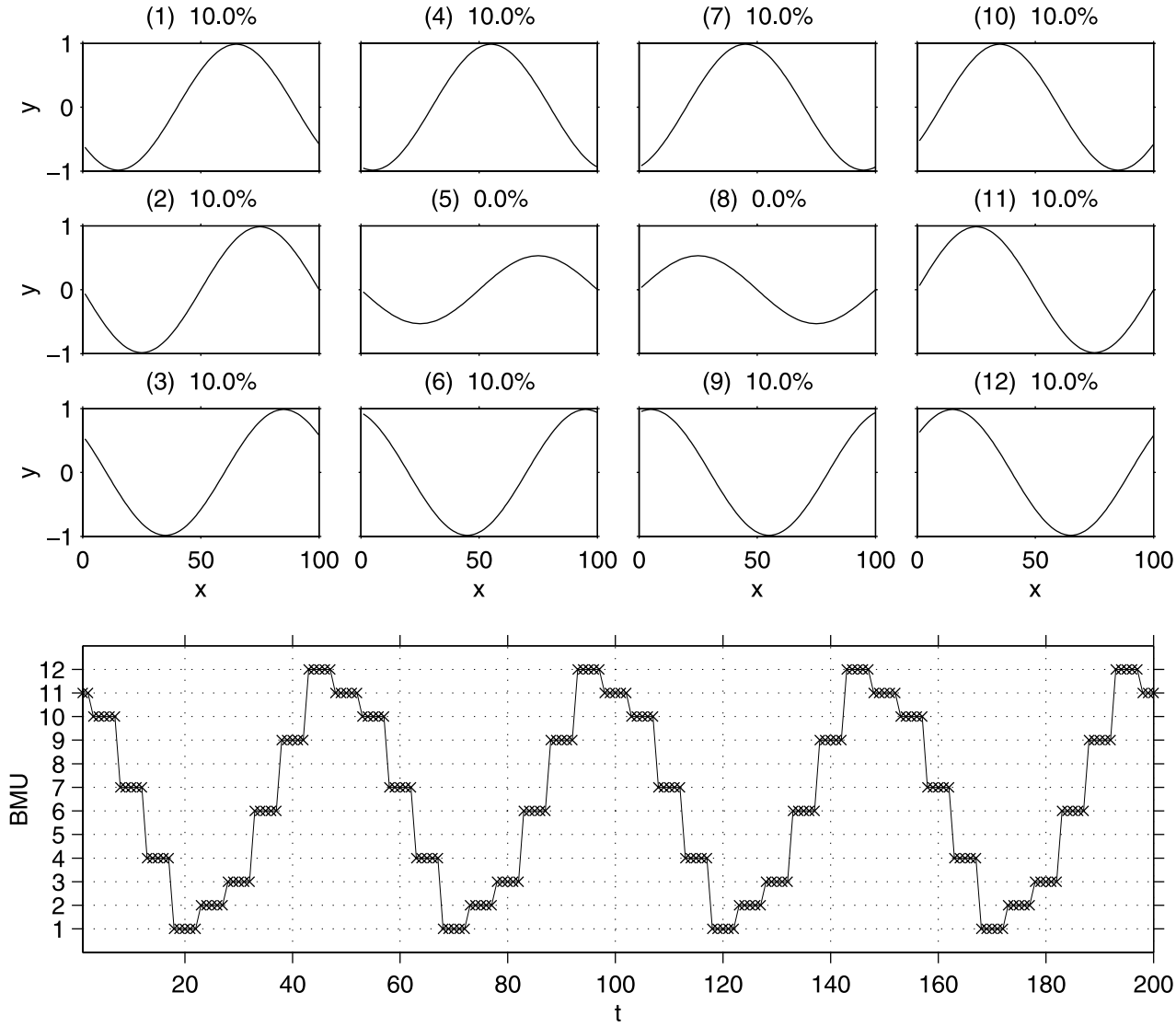

Figure 1. A 3 × 4 SOM representation of the linear progressive wave data. The top 12 plots show the SOM patterns with the frequency of occurrence given at the top of each plot. The bottom plot is the BMU time series.

2000] performance tests were only on the computational requirements of the algorithms, i.e., computing time for different training methods, not on the quality of the mappings or the sensitivity to different SOM parameter choices.

[7] Here we attempt to evaluate the performance of the SOM in feature extraction by using time series generated from known patterns along with sensitivity tests under various tunable parameter choices and signal-to-noise levels. Some of the questions addressed are: does the SOM technique recover known patterns reliably, are any artifices created, and which parameter choices provide the "best results?" Given these findings from synthetic data sets, we then apply the approach to an actual data set consisting of observed current velocity profiles.

[8] The remainder of this paper is arranged as follows. Section 2 introduces the SOM. In section 3, time series of linear progressive wave data are used to train and evaluate the SOM method. The results of using various map sizes, lattice structures, initializations, and different neighborhood functions are tested against a control run. By adding random noise to the known time series, the capability of the SOM in extracting essential features from noisy data is also examined. Section 4 considers a more complex synthetic data set consisting of multiple patterns and compares SOM extractions with those by EOF. The SOM is then applied to an oceanographic data set in section 5. Section 6 concludes with a summary and discussion.

2. Brief Introduction to the SOM

[9] The SOM performs a nonlinear projection from the input data space to a set of units (neural network nodes) on a two-dimensional grid. Each unit has a weight vector $\mathbf{m}_i$, which may be initialized randomly. Here the unit number i varies from 1 to M, M being the size of the SOM array. Adjacent units on the grid are called neighbors. In the Matlab SOM Toolbox [*Vesanto et al.*, 2000] there are three types of training algorithms: sequential, batch, and sompak.

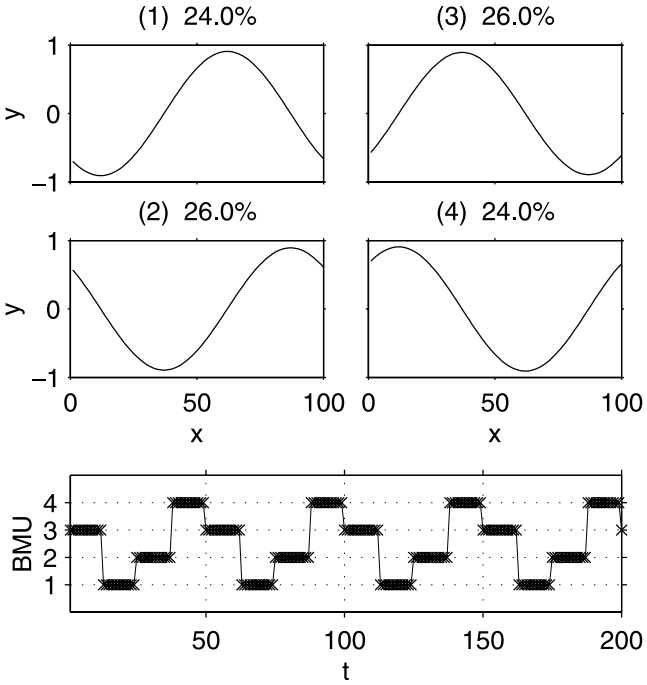

Figure 2. Same as Figure 1 but for a 2 × 2 SOM.

[10] In a sequential training process, elements from the high-dimensional input space, referred to as input vectors **x**, are presented to the SOM, and the activation of each unit for the presented input vector is calculated using an activation function. Commonly, the Euclidian distance between the weight vector of the unit and the input vector serves as the activation function. The weight vector of the unit showing the highest activation (i.e., the smallest Euclidian distance) is selected as the "winner" [or best matching unit (BMU)]. This process is expressed as

$$c_k = \arg\min \| \mathbf{x}_k - \mathbf{m}_i \| \tag{1}$$

where c_k is an index of the "winner" on the SOM for a data snapshot k, and c varies from 1 to M. The "arg" denotes "index." During the training process the weight vector of the winner is moved toward the presented input data by a certain fraction of the Euclidean distance as indicated by a time-decreasing learning rate α. Also, the weight vectors of the neighboring units are modified according to a spatial-temporal neighborhood function h. The learning rule may be expressed as

$$\mathbf{m}_i(t+1) = \mathbf{m}_i(t) + \alpha(t) \cdot h_{ci}(t) \cdot [\mathbf{x}(t) - \mathbf{m}_i(t)], \tag{2}$$

where t denotes the current learning iteration and **x** represents the currently presented input pattern. This iterative learning procedure leads to a topologically ordered mapping of the input data. Similar patterns are mapped onto neighboring units, whereas dissimilar patterns are mapped onto units farther apart.

[11] The batch version of the SOM algorithm is computationally more efficient than the sequential version [*Kohonen*, 1998; *Vesanto et al.*, 1999, 2000]. At each step of the training process, all the input data vectors are simultaneously used to update all the weight vectors. The data set is partitioned into M groups (by minimum Euclidian distance) and each group is used to update the corresponding weight vector. Updated weight vectors are calculated by:

$$\mathbf{m}_i(t+1) = \sum_{j=1}^{M} n_j h_{ij}(t) \overline{\mathbf{x}}_j \Big/ \sum_{j=1}^{M} n_j h_{ij}(t) \tag{3}$$

where $\overline{\mathbf{x}}_j$ is the mean of the n data vectors in group j. The $h_{ij}(t)$ denotes the value of the neighborhood function at unit j when the neighborhood function is centered on the unit i. In the batch algorithm, the learning rate function $\alpha(t)$ of the sequential algorithm is no longer needed, but, like the sequential algorithm, the radius of the neighborhood may decrease during the learning process. In the SOM Toolbox, there are four types of neighborhood functions available: "bubble," "gaussian," "cutgauss," and "ep" (or Epanechikov function).

$$h_{ci}(t) = \begin{cases} \mathrm{F}(\sigma_t - d_{ci}) & \text{bubble} \\ \exp\left(-d_{ci}^2/2\sigma_t^2\right) & \text{gaussian} \\ \exp\left(-d_{ci}^2/2\sigma_t^2\right)\mathrm{F}(\sigma_t - d_{ci}) & \text{cutgauss} \\ \max\left\{0, 1 - (\sigma_t - d_{ci})^2\right\} & \text{ep} \end{cases} \tag{4}$$

where σ_t is the neighborhood radius at time t, d_{ci} is the distance between map units c and i on the map grid and F is a step function

$$\mathrm{F}(x) = \begin{cases} 0 & \text{if x } < 0 \\ 1 & \text{if x } \geq 0 \end{cases} \tag{5}$$

(see *Vesanto et al.* [2000] for the geometries of these neighborhood functions). The default SOM Toolbox neighborhood function is "Gaussian." The neighborhood radius σ_t is either constant or linearly decreasing between specified initial and final values.

[12] The sompak training process is similar, employing C-language programs [*Kohonen et al.*, 1995]. *Vesanto et al.* [1999, 2000] demonstrates that batch training is the fastest of the three algorithms. Our evaluation is therefore based on the batch method only. (The SOM Toolbox version 2.0 can be freely downloaded from the Helsinki University of Technology, Finland: http://www.cis.hut.fi/projects/somtoolbox/.)

3. Performance Evaluations for Progressive Waves

[13] This section uses synthetic, linear progressive wave data to evaluate SOM feature extraction performance. After a brief introduction to the data, control run results are presented followed by sensitivity studies performed by varying map size, lattice structure, initialization, neighborhood function, and random noise levels.

3.1. Synthetic Data

[14] Linear progressive waves are a common topic in meteorology and oceanography. We specify a sinusoidal pattern in space (x) and time (t) of the form:

$$y(x,t) = \sin(kx - \omega t) \tag{6}$$

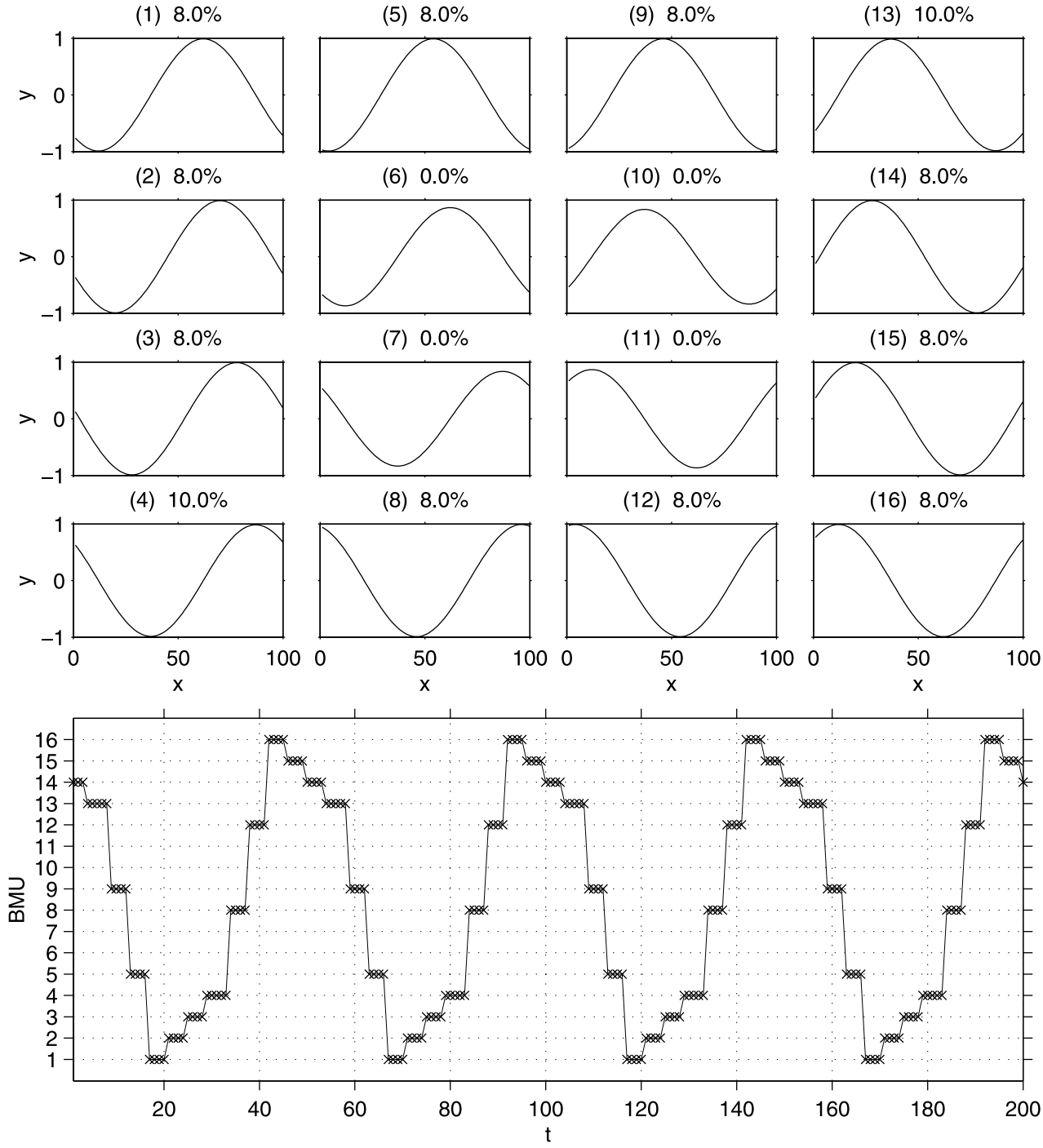

Figure 3. Same as Figure 1 but for a 4 × 4 SOM.

where $k = 2\pi/200$, $\omega = 2\pi/50$, $x = [1{:}100]$, $t = [1{:}200]$, and the amplitude is 1, such that each time step presents a wave of different phase. These data are arranged in a matrix such that each row is a spatial pattern (sine wave) at a given time step.

3.2. Control Run

[15] As an SOM control run, against which to test varying parameters, we use a map size of 3 × 4, a rectangular lattice, linearly initialized weights, an "ep" neighborhood function with initial and final neighborhood radii of 3 and 1, respectively, and all of these choices will be clarified later. Batch training is performed over 10 iterations, and the results are shown in Figure 1.

[16] The original 200 input frames (50 unique sinusoidal patterns repeated four times) are extracted into 12 units (Figure 1, top). Among these 12 units, numbers 5 and 8 are artifices, because their frequencies of occurrence are zero and their amplitudes are about half of that of the input data. The remaining 10 patterns show equal frequencies of occurrence (10% each of the total input data). From the BMU time series (Figure 1, bottom) we see the sequence of a wave pattern evolution moving in the positive x direction:

Table 1. Minimum and Maximum Wave Amplitudes of the SOM Patterns Excluding Those With Zero Frequency of Occurrence

Map Size	Minimum	Maximum
2 × 2	0.89	0.91
2 × 3	0.95	0.97
3 × 3	0.97	0.98
3 × 4	0.98	0.98
4 × 4	0.98	0.99
5 × 5	0.98	1.00
6 × 6	0.99	1.00
7 × 7	0.99	1.00
8 × 8	1.00	1.00

11→10→7→4→1→2→3→6→9→12→11 consistent with the phase progression of the synthetic sinusoidal pattern input to the SOM. Further examination of the adjacent SOM units shows a constant phase difference consistent with a progressive sine wave. Wave propagation is in the positive x direction, and the wave amplitude is about 1. Thus the SOM unit pattern extractions and BMU time series provide satisfactory descriptions of the input data set.

[17] The fictitious patterns 5 and 8 are a consequence of the smooth ordered mapping by the SOM that attempts to conserve the input data topology. Consequently, similar patterns are arranged in neighboring regions of the map, whereas dissimilar patterns are mapped further apart. Actually, pattern 1 mirrors pattern 12, pattern 2 mirrors pattern 11, and so on around the SOM. So, both patterns 5 and 8 are topological transitional patterns between the two opposite extremes of the SOM. Since these patterns do not occur in the input data set their frequency of occurrence is zero. Thus, though the SOM technique may produce some spurious patterns, it is capable of distinguishing between fictitious and nonfictitious patterns of a given data set, as we will see again later.

3.3. Effects of Varying the Map Size

[18] The training process requires a map size (number of units) specification. Larger map sizes result in more detailed patterns; smaller map sizes result in more general patterns [*Vesanto et al.*, 2000]. To test the map size sensitivity, the control run is repeated by changing the map size only, and the results for 2 × 2 and 4 × 4 rectangular arrays are shown in Figures 2 and 3, respectively.

[19] The 2 × 2 array distinguishes four different phases of the sinusoidal function (Figure 2). The BMU time series show the wave propagation through the following SOM unit sequence: 3→1→2→4→3, with a π/2 phase angle in between. While some detail is lost in going to smaller map size the systematic phase progression is retained. Far from the smooth-

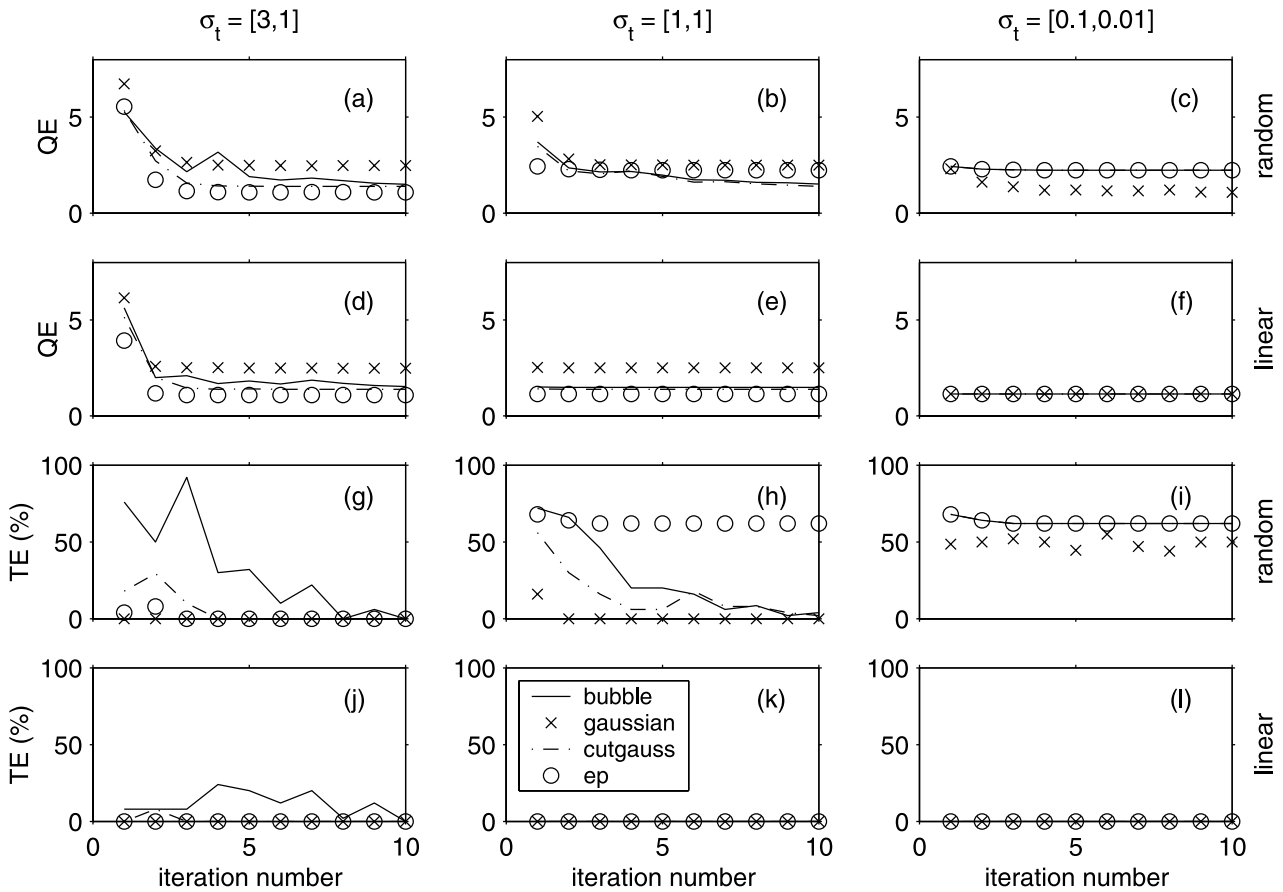

Figure 4. (a–f) QE and (g–l) TE by the four neighborhood functions as a function of iteration number during the SOM batch training process. The first and third rows are for random initialization, and the second and fourth rows are for linear initialization. The left, middle and right columns are for initial and final neighborhood radii of [3, 1], [1, 1] and [0.1, 0.01], respectively.

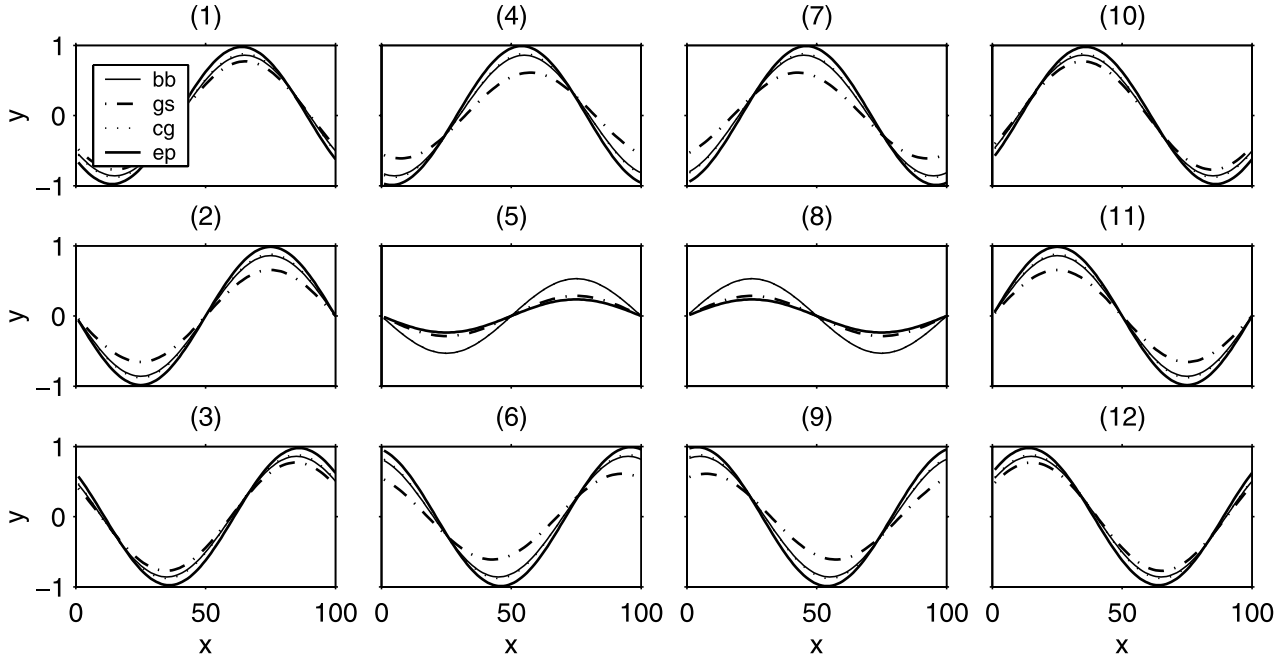

Figure 5. Comparison of SOM results using four different neighborhood functions: "bubble" (bb), "gaussian" (gs), "cutgauss" (cg) and "ep."

ing effect of temporal average, the fluctuating nature of the linear wave is picked up in both the space and time domains.

[20] When the map size is increased to 4 × 4, more detailed patterns (Figure 3) are extracted (more sampling in phase space). The progressive wave is represented by 12 different patterns each with $\pi/6$ phase difference propagating around the four central null patterns (6, 7, 10, and 11) with zero frequencies of occurrence.

[21] The pattern extracted in each SOM unit is the iterative result of the training process. Hence the amplitude may not be exactly that of the sinusoidal function input to the SOM. Table 1 shows the maximum/minimum values obtained for each of the different map sizes tested. While generally close to one, they are not necessarily equal to one. Larger map size leads to more accurate results by virtue of less pattern smoothing. However, larger array size results in more patterns so there is a trade off between compressing information into a manageable few patterns and accuracy, as with any related technique.

3.4. Sensitivity to Map Lattice Structure

[22] Along with map size, the map lattice and shape must be specified. The SOM lattice gives the local topology of the map, i.e., the connectivity of the map units. The lattice can be either rectangular or hexagonal in the SOM toolbox [*Vesanto et al.*, 2000]. Different shapes may also be chosen: sheet, cylinder or toroid. For simplicity, only the flat sheet is considered here. Relative to the (control run) rectangular lattice, a hexagonal lattice yields comparable, but more complex results (not shown).

3.5. Sensitivity to Initialization

[23] The unit weights may be initialized either randomly or linearly. In random initialization the map weights are initialized with random values in the range of [min(**x**), max(**x**)]. In linear initialization [e.g., *Kohonen*, 2001, p. 142] the SOM toolbox initializes the map weights by first performing an EOF decomposition and then linearly interpolating between the first two leading EOFs. The toolbox also provides two quantitative measures of mapping quality: average quantization error (QE) and topographic error (TE). The QE is the average distance between each data vector and the BMU. The TE gives the percentage of the data vectors for which the first BMU and the second BMU are not neighboring units. Lower QE and TE values indicate better mapping quality. For comparison with the control run, both the initialization methods and the neighborhood functions (and with three sets of neighborhood radii) were varied. The QE and TE in these experiments are shown as a function of iteration time in Figure 4.

[24] For all neighborhood functions with different radii the QE generally decreases and stabilizes after several training iterations (Figures 4a–4f). For a given neighborhood function and radius, linear initialization leads to

Table 2. Minimum and Maximum Wave Amplitudes of the SOM Patterns From Different Neighborhood Functions[a]

	"Bubble"		"Gaussian"		"Cutgauss"		"ep"	
σ_t Range	Minimum	Maximum	Minimum	Maximum	Minimum	Maximum	Minimum	Maximum
[3, 1]	0.82	0.89	0.62	0.78	0.88	0.88	0.98	0.98
[1, 1]	0.86	0.86	0.61	0.77	0.88	0.88	0.98	0.99
[0.1, 0.01]	0.98	0.99	0.98	0.99	0.98	0.99	0.98	0.99

[a]The SOM patterns with zero frequency of occurrence are excluded.

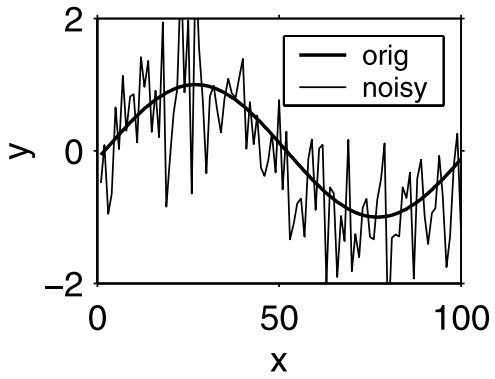

Figure 6. Data representation for the case of a sinusoidal function, plus random noise. Superimposed on the original noise-free data (thick line) are noisy data (thin lines) with a signal-to-noise ratio of one.

shorter QE stabilization time than random initialization. This linear initialization advantage is increasingly more important with larger, more complex data sets. Also, the QE with linear initialization is slightly less than with random initialization. For all neighborhood functions the TE is also smaller when linear initialization is used (Figures 4g–4l). Linear (versus random) initialization, therefore, saves iteration time and may provide for better SOM results.

3.6. Sensitivity to Neighborhood Function

[25] Different neighborhood functions result in different smoothing (equation (4)). Among the four neighborhood types, the "ep" gives the least smoothing to the SOM units. With a radius of 1 or less, the "ep" does little smoothing during the iterative learning process; if the initial TE is large (e.g., from the random initialization), it cannot be reduced very much (Figures 4h and 4i).

[26] Focusing on the second and the fourth rows of Figure 4, we see the effects of different neighborhood functions and radii on the QE and TE with linear initialization. With larger neighborhood radii both the "gaussian" and "ep" neighborhood functions give the smallest TE (Figure 4j). However, for a given neighborhood radius the "ep" leads to the smallest QE among the four neighborhood functions (Figures 4d and 4e). For small radii (e.g., $\sigma_t < 0.1$ in Figures 4f and 4l), both the QE and the TE for different neighborhood functions are the same.

[27] Given these measures, the SOM spatial patterns are further examined by changing neighborhood functions from the ("ep") control run (Figure 5). Different neighborhood functions give slightly different wave amplitudes, but they all extract the progressive wave pattern. The "ep" control run is the best since its amplitude is closest to the true value, 1. The "cutgauss" and "bubble" give similar results, whereas the "gaussian" deviates the most. These findings are consistent with the QE results (Figure 4d).

[28] Tests with smaller radii give similar amplitude results, with the control run being the best (Table 2). With very small radii, however, all of the neighborhood functions give similar results. The explanation follows from equation (4), which shows that the neighborhood functions reduce to a delta function for very small σ_t values.

$$h_{ci} = \begin{cases} 1 & \text{if } c = i \\ 0 & \text{if } c \neq i \end{cases} \tag{7}$$

such that it is only the weights of the "winner" that are updated in the training process.

3.7. Sensitivity to Noise

[29] Can the SOM extract known patterns in the presence of random noise? This question is examined by adding white noise to the linear progressive wave function. Results are given for a signal-to-noise ratio of 1 (equal variance for

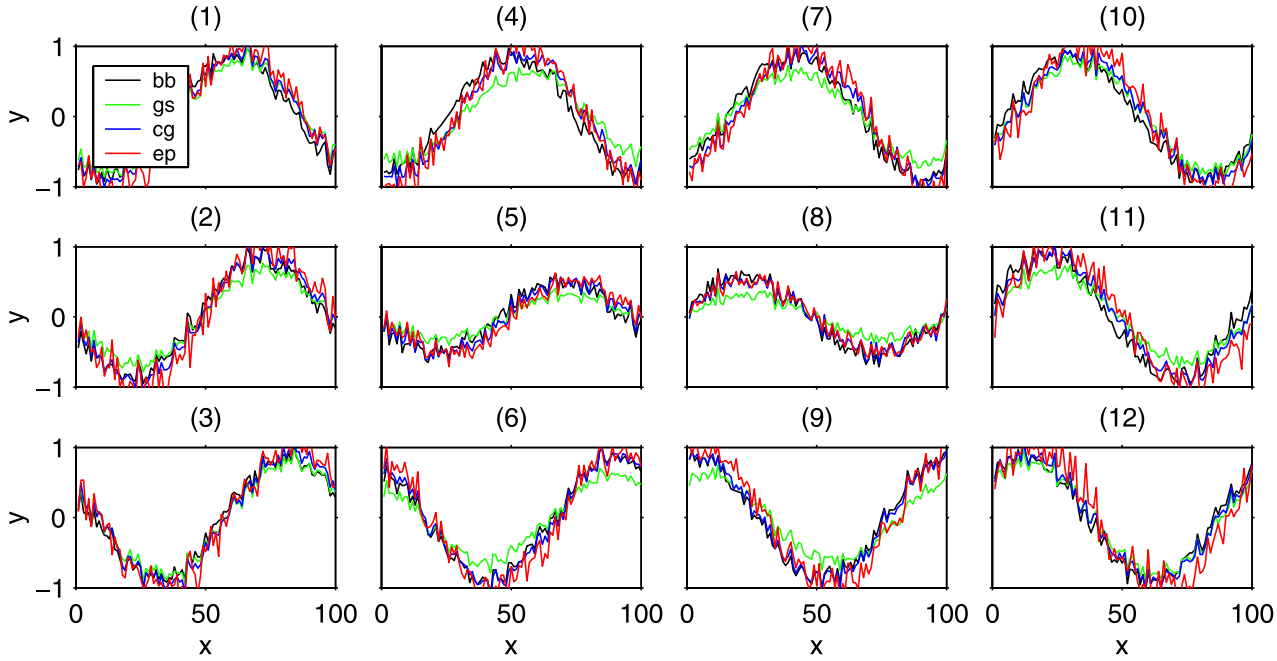

Figure 7. Same as Figure 5 except with random noise (at a signal-to-noise ratio of one) added to the sinusoidal wave data prior to the training processes.

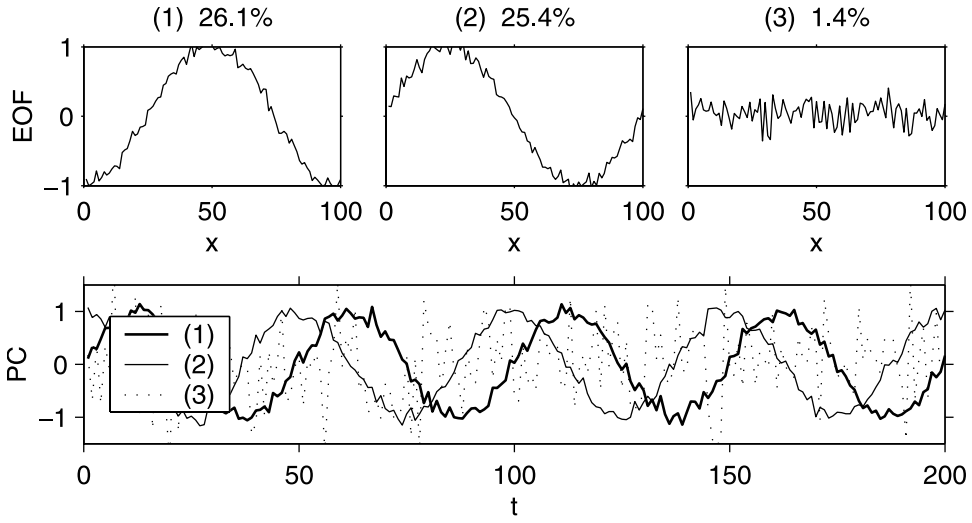

Figure 8. Time domain EOF analysis of the sinusoidal wave data with random noise added (with a signal-to-noise ratio of one). The top three plots are the first three leading mode EOFs (with the variance accounted for by each mode listed on top of each plot). The bottom plot contains the corresponding PC time series. The PC variance is normalized to be 0.5.

signal and noise as shown in Figure 6). Comments will also be made for experiments with higher and lower signal-to-noise ratios.

[30] Regardless of the neighborhood function employed, the noise becomes distributed among the SOM units (Figure 7, which can be compared directly with Figure 5). The same progressive waveform is extracted by the SOM, but with some additive noise. Similar to the noise-free data, the QEs are stable after four iterations (not shown) with the "ep" resulting in the smallest QE (the most accurate SOM patterns). When the SOM patterns of Figure 5 are subtracted from those of Figure 7, the residuals are white noise with variance dependent on the neighborhood functions. The residual (noise) variances are 0.035, 0.026, 0.013, and 0.008 for the "bubble," "ep," "cutgauss," and "Gaussian," respectively. Thus, among the four neighborhood functions, the "gaussian" gives the smoothest SOM patterns, but at the expense of detail (e.g., the amplitude of the known pattern is least accurate).

[31] For comparison, an EOF analysis is performed on the same noisy data (Figure 8). The first two leading EOF modes account for 26.1% and 25.4% of the variance, respectively. The sum of the two variances is 51.9%, which is close, but not equal, to the true value, 50%. Noise appears in both the EOF spatial and temporal patterns. Here the principal component (PC) time series are normalized to variances of 0.5 (the variance of the noise free data) so that the spatial eigenvectors have the same units as the original data, i.e., the wave amplitudes of the first two EOFs are about 1. Fitting smoothed cosine and sine waves to the first two EOFs, respectively, results in residual white noise with variances of 0.027 and 0.014, respectively; they are about the same magnitude as those in the SOM patterns. These results indicate that the SOM extracts the known pattern from the noise as well as the EOF.

[32] Tests with different noise levels of up to 200% of the original data variance (not shown) demonstrate that the SOM remains capable of extracting a known waveform from a noisy data set. The lower the noise level, the smoother the result, but, even with 200% noise added, the SOM yields a progressive wave with residual variance among units varying between 0.014 and 0.065.

4. Performance Evaluation for More Complex Patterns

[33] All the above analyses are based on a given repetitive linear progressive wave, and when comparing SOM with EOF neither technique shows a clear advantage in feature extraction. What is the outcome with multiple repetitive patterns? To address this question, a sequence of sine, step, sawtooth, and cosine waves are constructed and both SOM and EOF analyses are applied to them. The four input spatial patterns (Figure 9) are repeated for 50 cycles and these cycles are connected to form a time series of 200 frames.

4.1. The 2 × 2 Self-Organizing Map Result

[34] We first use a 2 × 2 SOM with rectangular lattice structure and linear initialization to extract the four known

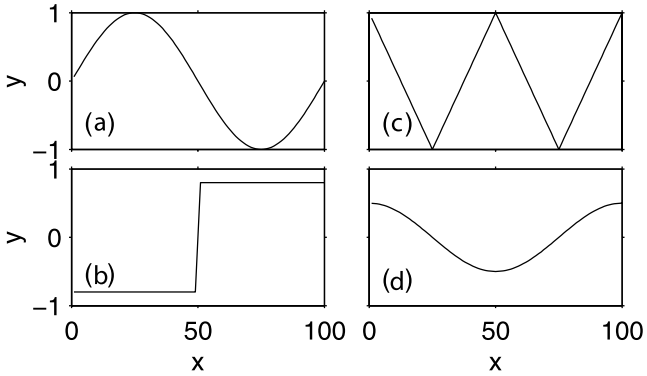

Figure 9. Four unique wave patterns analyzed by SOM and EOF in section 4: (a) sine, (b) step, (c) sawtooth, and (d) cosine functions with amplitudes of 1, 0.8, 1, and 0.5, respectively.

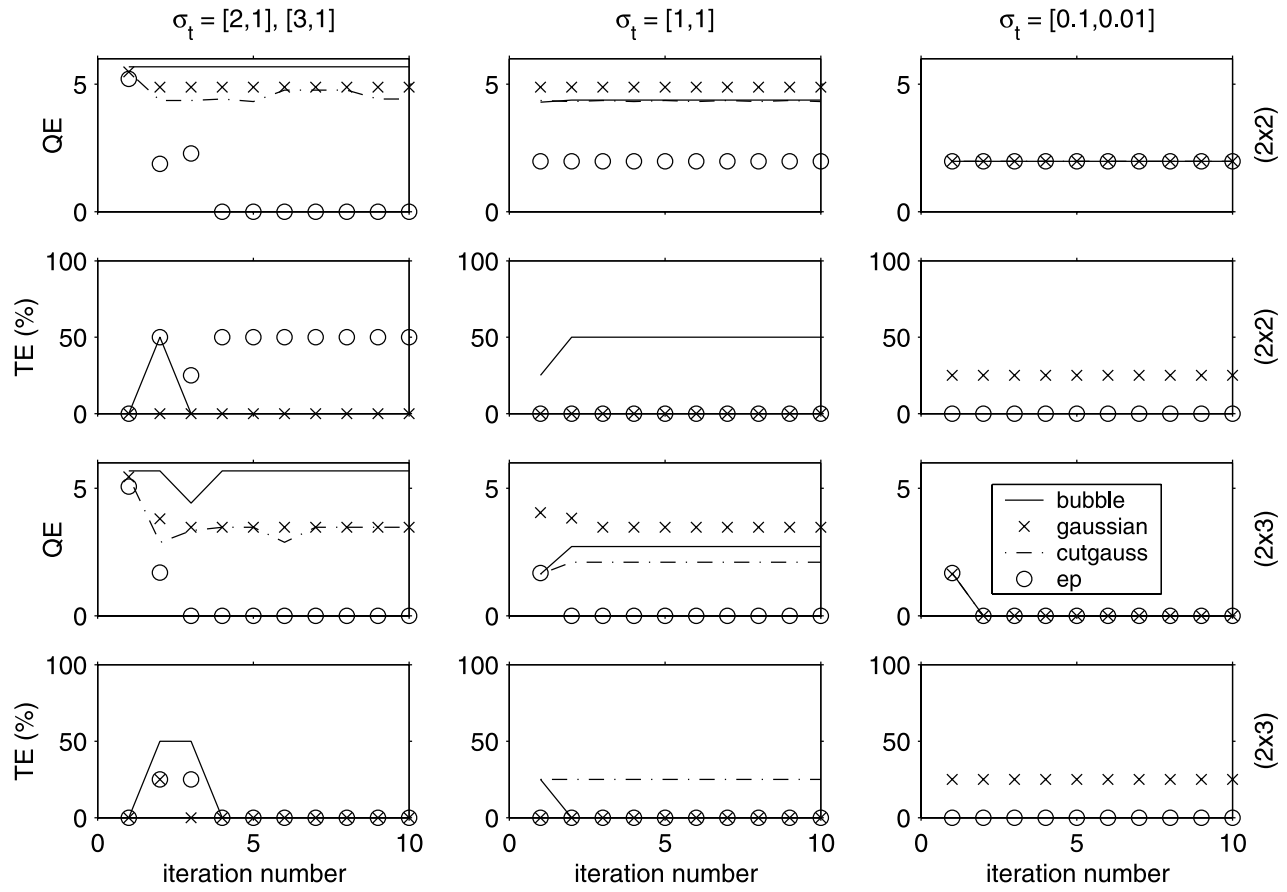

Figure 10. QE (first and third rows) and TE (second and fourth rows) as functions of the SOM training iteration number for different neighborhood functions and radii. The top two rows are for map size of 2×2, and the bottom two rows are for map size of 2×3. The left, middle and right columns are for larger ([2,1] for 2×2 SOM and [3,1] for 2×3 SOM), medium [1, 1] and small [0.1, 0.01] initial and final neighborhood radii, respectively.

patterns. Experiments are repeated using all the four neighborhood functions with three sets of neighborhood radii to determine the best results by monitoring QE and TE over the batch training process (Figure 10, top two rows). By definition, smaller QE means higher accuracy, and zero QE means that all of the patterns in the input data are completely extracted. In this set of experiments the best result is with an "ep" neighborhood function of initial and final radii of 2 and 1, respectively. The four input patterns are exactly reproduced and with the proper frequencies of occurrence (25%), and the BMU time series further follow the input sequence (not shown).

[35] While the TE is not the smallest in this case it is an irrelevant measure since it applies to the topological arrangement of the patterns. Since there are only four patterns their arrangement is not important. The TE becomes relevant as more patterns are extracted.

4.2. The 2×3 Self-Organizing Map Result

[36] Increasing the map size to 2×3 allows investigation of the appearance of fictitious patterns when the array size exceeds the number of known patterns. The same procedures are used as in 4.1, and the resulting QE and TE are shown in Figure 10 (bottom two rows). Since fictitious patterns now appear with zero frequency of occurrence (Figure 11), there are more cases for which QE is zero than in the 2×2 SOM. Similarly the TEs are mostly zero. Figure 11 shows the SOM result with the "ep" neighborhood function and initial and final radii of 3 and 1, respectively. As in the 2×2 array the four known patterns are extracted (in units 1, 4, 5 and 6), and each with 25% frequency of occurrence. The array contains two fictitious patterns (units 2 and 3), each with zero frequency of occurrence that occur as transitions between the adjacent true patterns, and it is these fictitious patterns that help to minimize the TE. Thus, even if the number of unique patterns in the input data set is unknown a priori, it may be safe to use a larger array SOM for feature extraction.

4.3. Empirical Orthogonal Function Result

[37] Results from an EOF analysis on the same data set are shown in Figure 12. Of the leading three EOFs, none look like the original patterns. Unlike the previous analysis of a simple repetitive waveform, the EOF fails to extract the four different patterns presented here. The order in which the patterns are organized is not relevant to the analysis as demonstrated through random permutations of the pattern sequence (not shown). The SOM extracts the patterns while the EOF does not. The only differences between the ordered

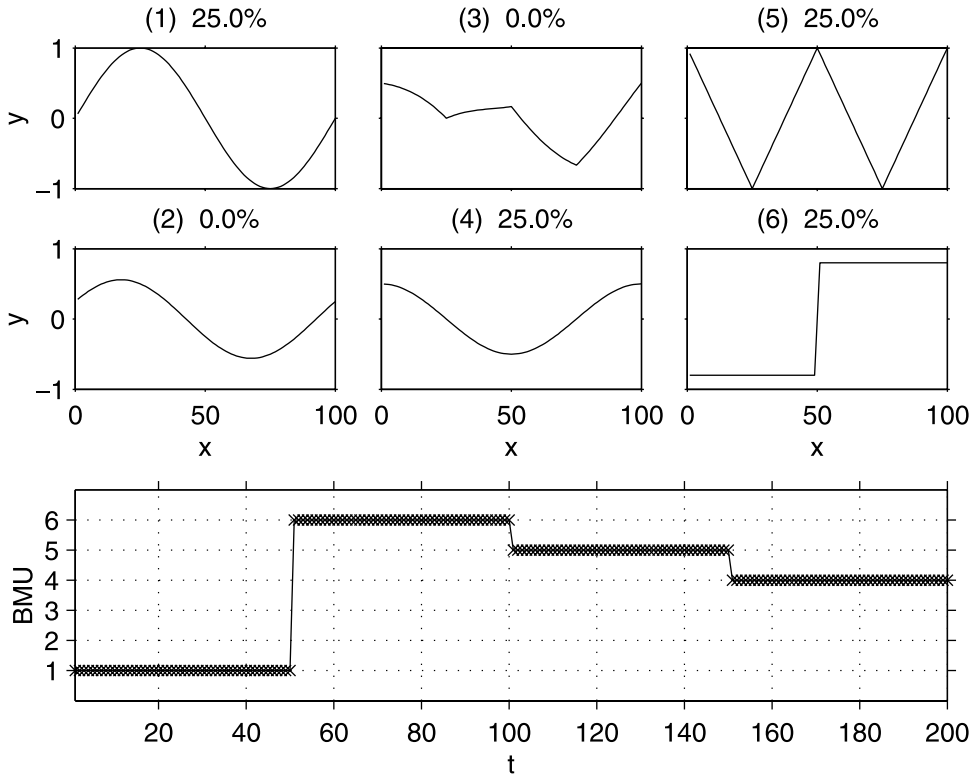

Figure 11. A 2 × 3 SOM representation of the sequential pattern data of Figure 9. The top six plots show the SOM patterns with the frequency of occurrence given at the top of each plot. The bottom plot is the BMU time series.

and random sequence are in the BMU and the PC time series for the SOM and the EOF, respectively.

5. Demonstration With Application to Ocean Current Patterns on the West Florida Shelf (WFS)

[38] An application of the SOM to moored ADCP velocity data on the WFS is given by *Liu and Weisberg* [2005]. The above sensitivity analyses suggest that while the SOM is a reliable tool in feature extraction, different controlling parameters may give slightly different results. For example, as regards neighborhood functions, the "gaussian" may give the smoothest patterns, while the "ep" may give the most accurate mappings. In this section, the *Liu and Weisberg* [2005] 2-day, low-pass filtered ADCP data are reexamined, adding additional details for extreme events at

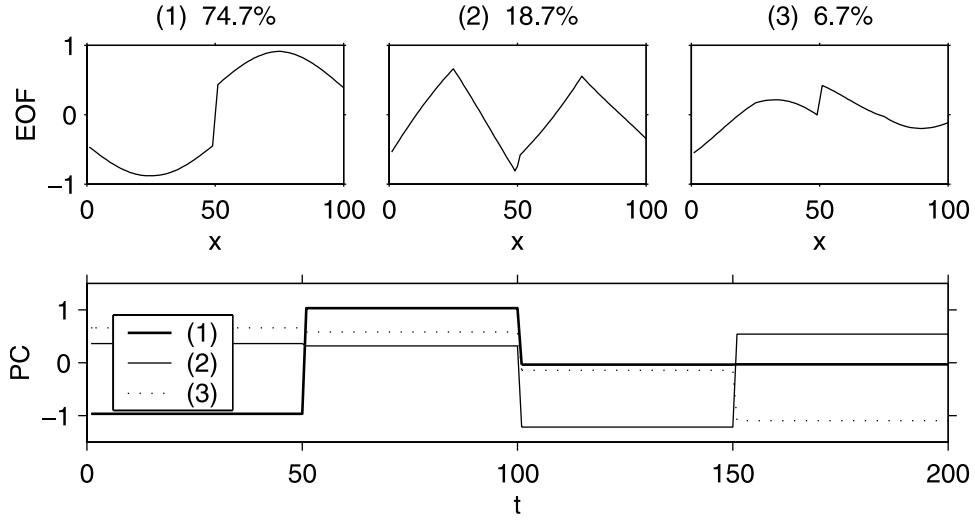

Figure 12. Time domain EOF analysis of the sequential pattern data of Figure 9. The top three plots are the first three leading mode EOFs (with the variance accounted for by each mode listed at the top of each plot). The bottom plot gives the corresponding PC time series. The PC variance is normalized to be 0.5.

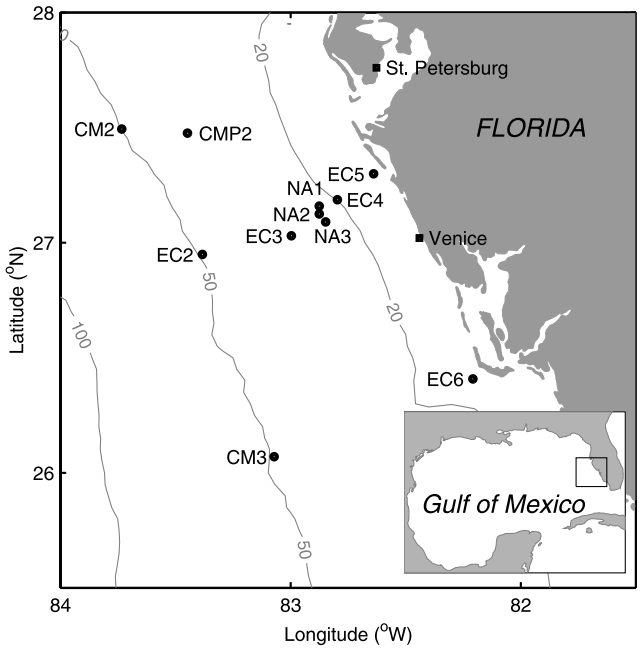

Figure 13. Map of the West Florida Shelf showing topography (isobaths units in m), acoustic Doppler current profiler (ADCP) moorings and wind stations. A map of the entire Gulf of Mexico is inserted in the bottom right corner, and the square box denotes the study area.

the synoptic and longer timescales by varying the SOM parameters. The data set contains ADCP velocity data at the near surface, middle, and near bottom levels from 11 WFS moorings (Figure 13) during the 3-year period, September 1998 through December 2001.

[39] To facilitate comparison with *Liu and Weisberg* [2005], the same map size (3 × 4) is used here. As in 4.1 and 4.2, a rectangular lattice, ''sheet'' map shape, linearly initialized weights, and batch training are used. The SOM experiment is repeated for all the four neighborhood functions with different radius values, and the resulting average QE is examined to search for the most accurate SOM mapping. As expected, the ''ep'' neighborhood function gives the lowest average QE among the four types, and, when the neighborhood radius shrinks to the very small value, all the QE values converge to that of the ''ep'' type with the radius of 1. In accordance with Figure 14, the SOM result using the ''ep'' neighborhood function with the radius of 1 and 10 iterations is chosen as the most accurate mapping as given in Figure 15.

[40] Similar to that of *Liu and Weisberg* [2005] three sets of coherent current patterns are identified: southeastward, transitional and northwestward currents, located in the top, middle and bottom rows of Figure 15, respectively. Flow asymmetries are found between the upwelling patterns (4, 7 and 10) and the downwelling pattern (6), i.e., when southeastward currents are compared with northwestward currents the magnitudes are larger, the coastal jet is located further offshore, and the velocity vector rotations with depth are larger in shallower water.

[41] In addition to the asymmetric upwelling/downwelling patterns in *Liu and Weisberg* [2005], extreme upwelling and downwelling events, identified as patterns 1 and 3, respectively, are extracted, each with maximum current speeds greater than 20 cm/s. These strong current patterns relate to the largest synoptic weather events, with frequencies of occurrence of 1–3% over the 3-year record. As seen in Figure 15 (bottom) (note the order of the BMUs on *y* axis is rearranged for clarity), the strong downwelling pattern 3 appears in August through October of each year because of tropical storms and hurricanes; an example is the 19–22 September 1999 tropical storm Harvey event (Figure 16, top). The strong upwelling pattern 1 appears sporadically in fall and winter because of the passage of the strongest cold fronts and also on the trailing side of tropical storms and hurricanes. Examples of these are the 21–24 January 2001 upwelling event by a winter cold front (Figure 16, bottom) and the 14–15 September 1999 event by hurricane Floyd (Figure 16, top). The extreme upwelling and downwelling patterns identified here are less asymmetric than the moderate event patterns previously identified [*Liu and Weisberg*, 2005], especially for the currents in shallowest water. For example, the coastal jet cores for extreme upwelling and downwelling patterns are located near the 30 m isobath and the velocity vector rotations with depth are reduced. This result is due to increased mixing with increased external forcing intensity. An upwelling and downwelling asymmetry is attributed to stratification through the effects of the ''thermal wind'' on the bottom Ekman layer flows [*Weisberg et al.*, 2001]. Extreme weather events cause increased mixing (reduced

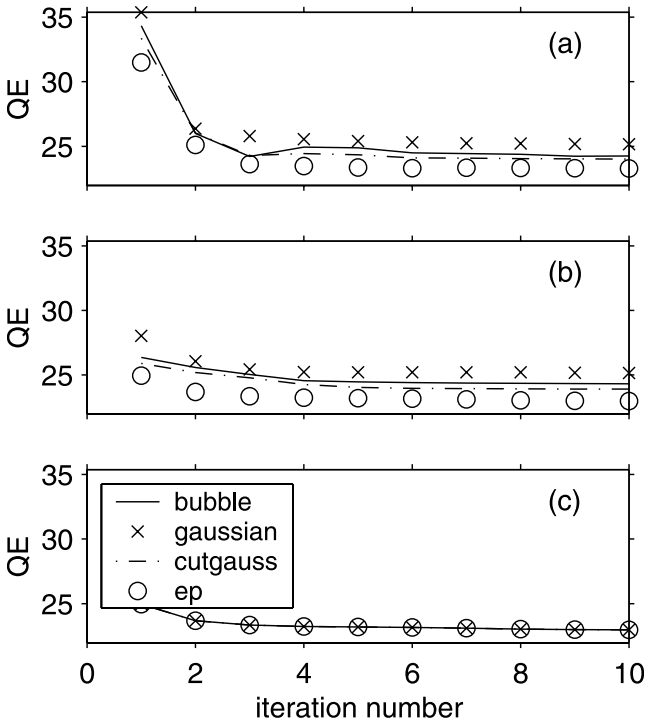

Figure 14. QE as functions of the SOM training process iteration number for the four different neighborhood functions applied to the WFS velocity data. Three different pairs of initial and final neighborhood radii: (a) [3, 1], (b) [1, 1], and (c) [0.1, 0.01].

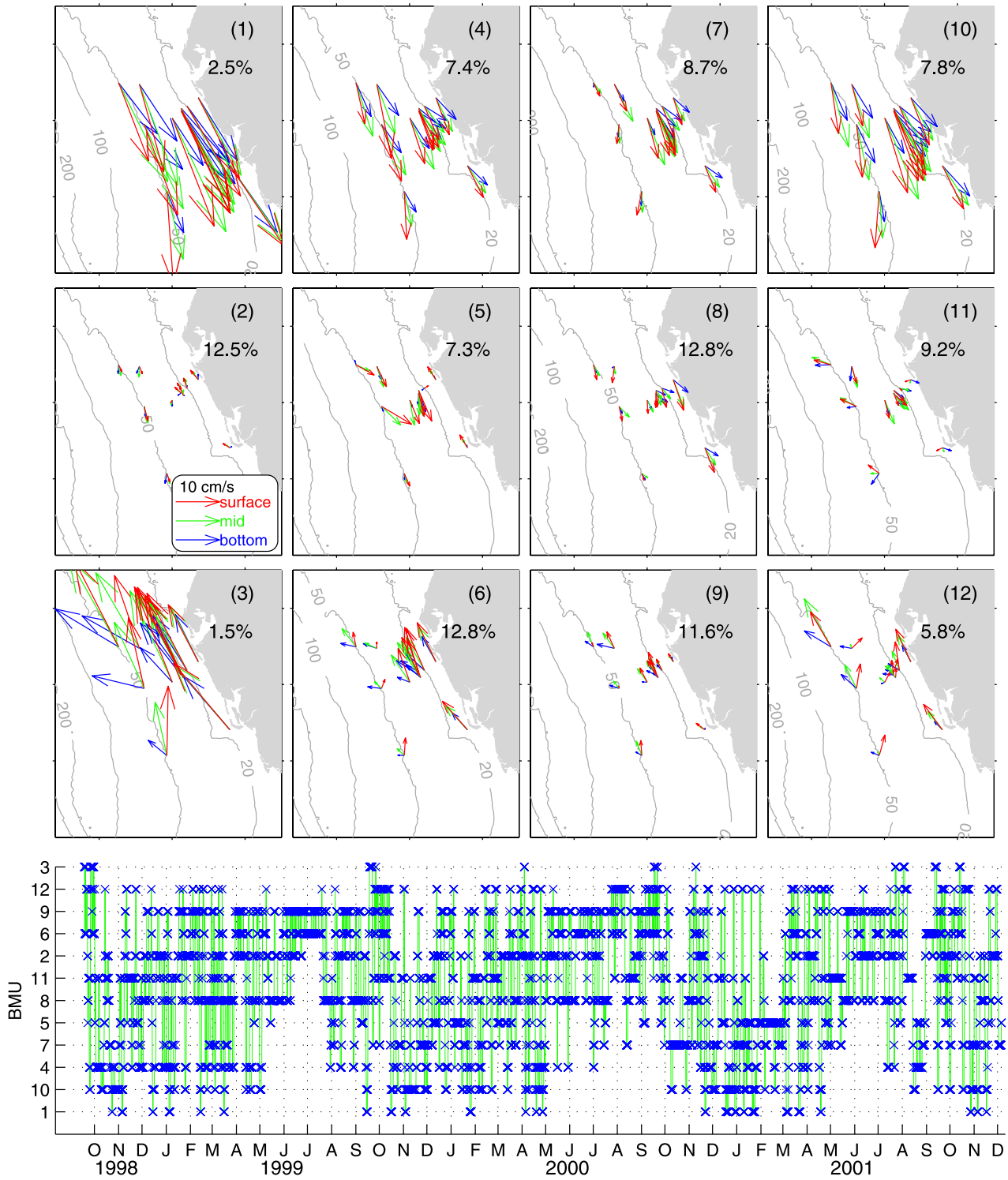

Figure 15. A 3 × 4 SOM representation of the WFS 2-day low-pass filtered ADCP velocity data from September 1998 through December 2001. The top 12 plots are the SOM patterns with the frequency of occurrence given as percentages in each plot. The bottom plot is the BMU time series.

stratification) and, hence, a decrease in current response asymmetry.

[42] In summary, five sets of characteristic ocean current patterns are extracted from the 2-day, low-pass filtered velocity data: strong upwelling and downwelling patterns associated with extreme weather forcing, asymmetric upwelling and downwelling patterns associated with more moderate weather forcing, and a set of transitional current patterns. Less smoothing, through the use of the "ep" neighborhood function here versus the "gaussian" neighbor-

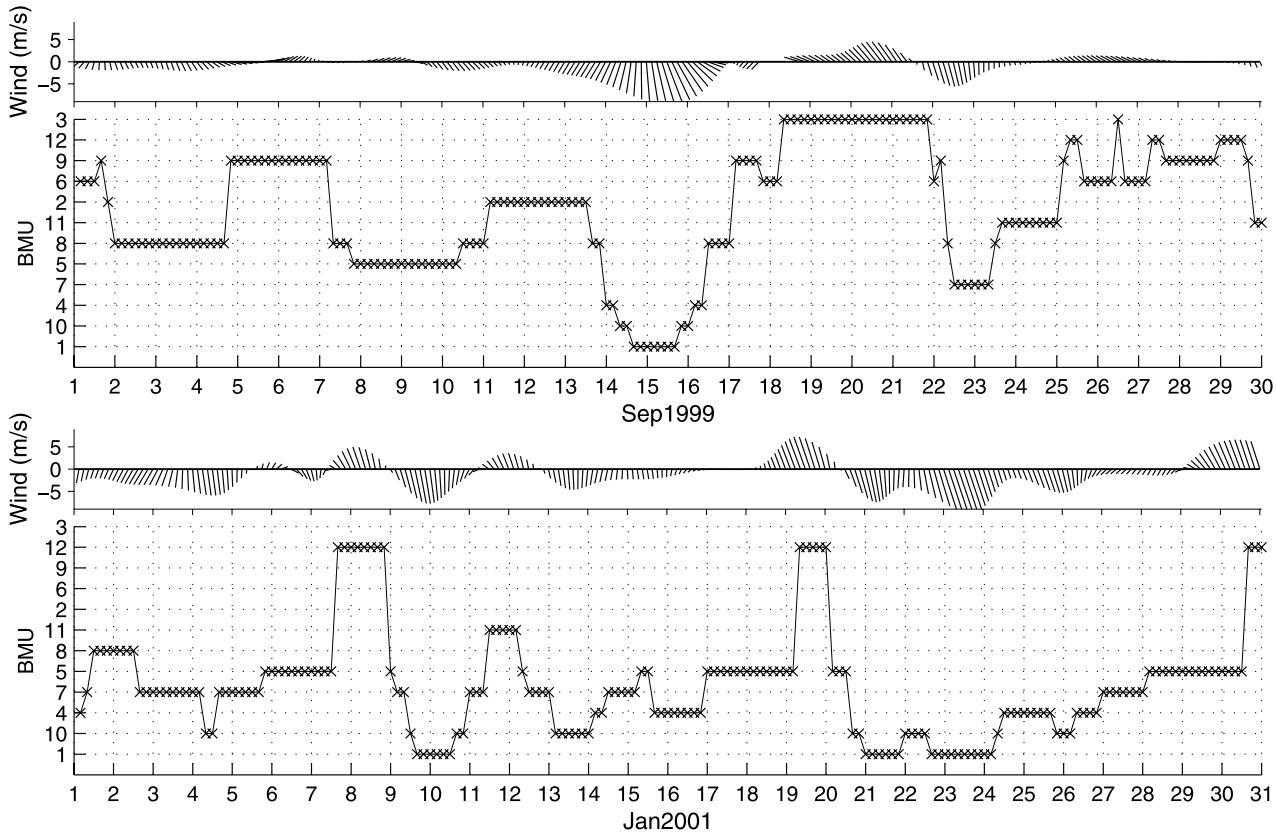

Figure 16. Time series of the local winds at Venice, Florida, and the BMU for September 1999 and January 2001.

hood function used by *Liu and Weisberg* [2005], results in additional details that can be related to external forcing via the BMU time series.

6. Summary and Discussions

[43] The application of the SOM as a feature extraction technique in meteorology and oceanography has been demystified by performing sensitivity studies on the tunable parameters of SOM implementation. A series of SOM feature extraction experiments were performed, first by using artificial data sets comprising known patterns followed by an application to a geophysical data set of coastal ocean currents.

[44] The SOM accurately represented a time series of linear progressive sine waves of fixed amplitude, period and wavelength. The following results were found from studies of the sensitivity to the SOM adjustable parameters. (1) A larger map size resulted in slightly more accurate mapping. (2) A rectangular lattice appeared to be preferable for small size SOMs; however, a hexagonal lattice may be useful for larger map sizes. (3) Linear initialization had two advantages over random initialization: fewer iterations for QE convergence and smaller TE. (4) Among the four neighborhood functions, the "ep" type gave the best results (smallest QE and TE). (5) While fictitious patterns appeared in the SOM, they were of no consequence since their frequencies of occurrence were zero.

[45] The following parameter choices are recommended for SOM applications. For a small map size, a suitable SOM configuration is a rectangular neural lattice of "sheet" shape, linear initialization, "ep" neighborhood function, and batch training algorithm. As to the choices of the neighborhood radius and the training iterations, it is practical to monitor QE and TE in a searching process. Minimum QE indicates the most accurate representation of the input data. Minimum TE indicates the best SOM pattern organization such that adjacent to a BMU in the map lattice is the second BMU. TE is not critical for a small size SOM; however, it may be important for a larger size SOM with increasing data set complexity.

[46] The SOM extracted features from noisy data over a broad range of signal-to-noise ratios. Of the four neighborhood functions, the "ep" gave the most accurate patterns (smallest QE), whereas the "gaussian" gave the smoothest patterns with the lowest noise levels. While noise appeared superimposed upon the SOM units, the known patterns were readily identified, and the SOM and EOF results were comparable.

[47] As a further test between SOM and EOF, time series were constructed by linking together four unique pattern types. An SOM successfully extracted these four known patterns, whereas an EOF did not. This SOM advantage over EOF in feature extraction adds to the previous findings by *Liu and Weisberg* [2005] and *Liu et al.* [2006] where the SOM identified asymmetries in the response patterns of velocity and SST, respectively, that the EOF did not. The SOM also allows for data gaps and mean values in the input data [*Richardson et al.*, 2003; *Liu and Weisberg*, 2005; *Liu et al.*, 2006], making it more convenient to use than the EOF in some cases. On the other hand the EOF, by preserving variance, is capable of exactly reconstructing

the data from which it derives by summing over all modes, whereas the SOM by preserving topology does not provide a convenient way to exactly reproduce the data. Thus both methods have advantages and disadvantages.

[48] The analyses concluded with a real geophysical data application, in which the SOM was used to extract coastal ocean current patterns on the West Florida Shelf from moored velocity profile time series sampled from September 1998 to December 2001. With the recommended SOM parameter choices and the QE monitoring process discussed herein new insights were gained. Strong current patterns associated with severe weather events (hurricanes, tropical storms, and strong winter cold fronts) were extracted in addition to previously identified [*Liu and Weisberg*, 2005] asymmetric upwelling/downwelling patterns of more moderate strength currents and transitional patterns of relatively weak currents. This additional finding is attributed to the use of the ''ep'' neighborhood function, resulting in less smoothing.

[49] With geophysical applications already made to satellite SST and winds [e.g., *Richardson et al.*, 2003], altimetry [e.g., *Hardman-Mountford et al.*, 2003], in situ ADCP [*Liu and Weisberg*, 2005], and gridded atmospheric data using multiple input variables [e.g., *Cavazos et al.*, 2002], an increased use of the SOM for geophysical feature extractions is anticipated. Other types of data and analyses amenable to SOM treatments include, but are not limited to, surface currents remotely sensed by HF-radar, climate-related data sets relative to the various identified climate indices, numerical model results, climate change scenarios, or any geophysical application for which large fields of information (data or models) are available for pattern recognition.

[50] **Acknowledgments.** Support was by the Office of Naval Research, grants N00014-98-1-0158 and N00014-02-1-0972. The second of these, for the Southeast Atlantic Coastal Ocean Observing System (SEACOOS), is administered by UNC under task order 3-12110-10. The SOM MATLAB Toolbox is by E. Alhoniemi, J. Himberg, J. Parhankangas, and J. Vesanto.

References

Ainsworth, E. J. (1999), Visualization of ocean colour and temperature from multispectral imagery captured by the Japanese ADEOS satellite, *J. Visualization*, *2*, 195–204.

Ainsworth, E. J., and S. F. Jones (1999), Radiance spectra classification from the ocean color and temperature scanner on ADEOS, *IEEE Trans. Geosci. Remote Sens.*, *37*, 1645–1656.

Ambroise, C., G. Seze, F. Badran, and S. Thiria (2000), Hierarchical clustering of self-organizing maps for cloud classification, *Neurocomputing*, *30*, 47–52.

Cavazos, T. (2000), Using self-organizing maps to investigate extreme climate events: An application to wintertime precipitation in the Balkans, *J. Clim.*, *13*, 1718–1732.

Cavazos, T., A. C. Comrie, and D. M. Liverman (2002), Intraseasonal anomalies associated with wet monsoons in southeast Arizona, *J. Clim.*, *15*, 2477–2490.

Espinosa-Carreon, T. L., P. T. Strub, E. Beier, F. Ocampo-Torres, and G. Gaxiola-Castro (2004), Seasonal and interannual variability of satellite-derived chlorophyll pigment, surface height, and temperature off Baja California, *J. Geophys. Res.*, *109*, C03039, doi:10.1029/2003JC002105.

Hardman-Mountford, N. J., A. J. Richardson, D. C. Boyer, A. Kreiner, and H. J. Boyer (2003), Relating sardine recruitment in the northern Benguela to satellite-derived sea surface height using a neural network pattern recognition approach, *Prog. Oceanogr.*, *59*, 241–255.

Hewitson, B. C., and R. G. Crane (1994), *Neural Nets: Applications in Geography*, Springer, New York.

Hewitson, B. C., and R. G. Crane (2002), Self-organizing maps: Applications to synoptic climatology, *Clim. Res.*, *22*, 13–26.

Hong, Y., K. Hsu, S. Sorooshian, and X. Gao (2004), Precipitation estimation from remotely sensed imagery using an artificial neural network cloud classification system, *J. Appl. Meteorol.*, *43*, 1834–1853.

Hong, Y., K. Hsu, S. Sorooshian, and X. Gao (2005), Self-organizing nonlinear output (SONO): A neural network suitable for cloud patch-based rainfall estimation at small scales, *Water Resour. Res.*, *41*, W03008, doi:10.1029/2004WR003142.

Hsieh, W. W. (2001), Non-linear principal component analysis by neural networks, *Tellus*, *53*, 599–615.

Hsieh, W. W. (2004), Nonlinear multivariate and time series analysis by neural network methods, *Rev. Geophys.*, *42*, RG1003, doi:10.1029/2002RG000112.

Hsu, K., H. V. Gupta, X. Gao, S. Sorooshian, and B. Imam (2002), Self-organizing linear output map (SOLO): An artificial neural network suitable for hydrologic modeling and analysis, *Water Resour. Res.*, *38*(12), 1302, doi:10.1029/2001WR000795.

Kaski, S., J. Kangas, and T. Kohonen (1998), Bibliography of Self-Organizing Map (SOM) papers: 1981–1997, *Neural Comput. Surv.*, *1*, 102–350.

Kiang, M. Y., and A. Kumar (2001), An evaluation of Self-Organizing Map networks as a robust alternative to factor analysis in data mining applications, *Inf. Syst. Res.*, *12*(2), 177–194.

Klink, J. (1985), EOF analysis of central Drake Passage currents from DRAKE 79, *J. Phys. Oceanogr.*, *15*, 288–298.

Kohonen, T. (1982), Self-organized information of topologically correct features maps, *Biol. Cyber.*, *43*, 59–69.

Kohonen, T. (1998), The self-organizing map, *Neurocomputing*, *21*, 1–6.

Kohonen, T. (2001), *Self-Organizing Maps*, 3rd ed., *Springer Ser. Inf. Sci.*, *30*, 501 pp.

Kohonen, J., J. Hynninen, J. Kangas, and J. Laaksonen (1995), SOM_PAK, the self-organizing map program, version 3.1, report, 27 pp., Lab. of Comput. and Inf. Sci., Helsinki Univ. Technol., Helsinki, Finland.

Lagerloef, G. S. E., and R. L. Bernstein (1988), Empirical orthogonal function analysis of advanced very high resolution radiometer surface temperature patterns in Santa Barbra Channel, *J. Geophys. Res.*, *93*, 6863–6873.

Liu, Y., and R. H. Weisberg (2005), Patterns of ocean current variability on the West Florida Shelf using the self-organizing map, *J. Geophys. Res.*, *110*, C06003, doi:10.1029/2004JC002786.

Liu, Y., R. H. Weisberg, and R. He (2006), Sea surface temperature patterns on the West Florida Shelf using the Growing Hierarchical Self-Organizing Maps, *J. Atmos. Oceanic Technol.*, *23*(2), 325–338.

Malmgren, B. A., and A. Winter (1999), Climate zonation in Puerto Rico based on principal components analysis and an artificial neural network, *J. Clim.*, *12*, 977–985.

Murtagh, F., and M. Hernandez-Pajares (1995), The Kohenon Self-Organizing Map method: An assessment, *J. Classif.*, *12*, 165–190.

Oja, M., S. Kaski, and T. Kohonen (2002), Bibliography of Self-Organizing Map (SOM) papers: 1998–2001 addendum, *Neural Comput. Surv.*, *3*, 1–156.

Petrou, M. (2004), Preface, *Pattern Recognition Lett.*, *25*, 1459.

Richardson, A. J., M. C. Pfaff, J. G. Field, N. F. Silulwane, and F. A. Shillington (2002), Identifying characteristic chlorophyll a profiles in the coastal domain using an artificial neural network, *J. Plankton Res.*, *24*, 1289–1303.

Richardson, A. J., C. Risien, and F. A. Shillington (2003), Using self-organizing maps to identify patterns in satellite imagery, *Prog. Oceanogr.*, *59*, 223–239.

Risien, C. M., C. J. C. Reason, F. A. Shillington, and D. B. Chelton (2004), Variability in satellite winds over the Benguela upwelling system during 1999–2000, *J. Geophys. Res.*, *109*, C03010, doi:10.1029/2003JC001880.

Silulwane, N. F., A. J. Richardson, F. A. Shillington, and B. A. Mitchell-Innes (2001), Identification and classification of vertical chlorophyll patterns in the Benguela upwelling system and Angola-Benguela Front using an artificial neural network, in *A Decade of Namibian Fisheries Science*, edited by A. I. L. Payne, S. C. Pillar, and R. J. M. Crawford, *S. Afr. J. Mar. Sci.*, *23*, 37–51.

Vesanto, J., J. Himberg, E. Alhoniemi, and J. Parhankangas (1999), Self-organizing map in Matlab: The SOM Toolbox, paper presented at Matlab DSP Conference, Espoo, Finland, Nov.

Vesanto, J., J. Himberg, E. Alhoniemi, and J. Parhankangas (2000), SOM Toolbox for Matlab 5, report, Helsinki Univ. of Technol., Helsinki, Finland.

Weare, B. C., A. R. Navato, and R. E. Newell (1976), Empirical orthogonal analysis of Pacific sea surface temperature, *J. Phys. Oceanogr.*, *6*, 671–678.

Weisberg, R. H., Z. Li, and F. E. Muller-Karger (2001), West Florida shelf response to local wind forcing: April 1998, *J. Geophys. Res.*, *106*, 31,239–31,262.

Y. Liu and R. H. Weisberg, College of Marine Science, University of South Florida, 830 First Street South, St. Petersburg, FL 33701, USA. (yliu18@gmail.com)

C. N. K. Mooers, Rosenstiel School of Marine and Atmospheric Science, University of Miami, Miami, FL 33149, USA.