Multiscale Models for Data Processing: An Experimental Sensitivity Analysis

Stefano Ferrari, N. Alberto Borghese, *Member, IEEE*, and Vincenzo Piuri, *Senior Member, IEEE*

***Abstract*—Hierarchical radial basis functions (HRBFs) networks have been recently introduced as a tool for adaptive multiscale image reconstruction from range data. These are based on local operation on the data and are able to give a sparse approximation. In this paper, HRBFs are reframed for the regular sampling case, and they are compared with wavelet decomposition. Results show that HRBFs, thanks to their constructive approach to approximation, are much more tolerant on errors in the parameters when errors occur in the configuration phase.**

***Index Terms*—Basis functions, function spaces, iterative decomposition, multiresolution analysis, multiscale signal decomposition, quantization error, RBF networks, robustness, sensitivity, signal processing, wavelets.**

I. Introduction

MULTIRESOLUTION techniques are widely used in signal processing, as they are able to analyze the signal properties and produce a local description both in the temporal and frequency domains. This feature is of fundamental importance when a continuous measurement field has to be recovered from spot measurements.

Wavelet decomposition [1], [2], based on multiresolution analysis (MRA) theory, is the most-used tool for multiscale signal processing thanks to the fast machinery adopted to compute its coefficients. An interesting alternative is offered by multiscale approximation through Gaussian bases. In [3], [4], hierarchical radial basis functions (HRBFs) neural networks have been introduced, and it is shown that, although they do not perform a wavelet decomposition, they do enjoy the same asymptotic approximation properties. Moreover, HRBFs are able to better achieve a given approximation error. This is useful when measurement noise characterization is available.

Real-time requirements, which an embedded system commonly has to cope with, often need hardware implementation of the algorithms [5]. Embedded systems have to satisfy also other requirements, such as robustness, compactness, power consumption, and economical cost. The notation and precision used to represent the numerical quantities affect the characteristics of a computing system in different ways. Hence, in embedded systems design, the choice of the computational framework and its representation are a part of the compromise between complexity, speed, and precision that the designer has to meet [6], [7]. A detailed understanding of how representation and precision affect the quality of the solution is therefore fundamental for an effective and efficient design.

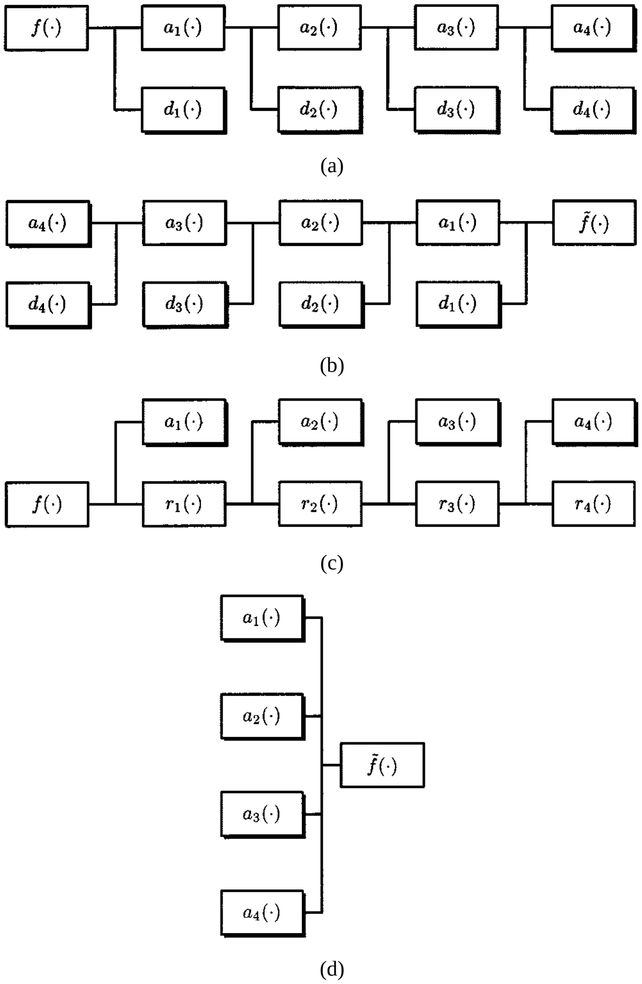


Fig. 1. Functional schemes for analysis and synthesis with MRA and HRBF. (a) MRA determination of the coefficients; (b) MRA Reconstruction; (c) HRBF determination of the coefficients; (d) HRBF Reconstruction.

The aim of this work is to reframe the HRBF configuration procedure for regular sampled data and to compare the accuracy in the reconstruction of a given signal through HRBF and wavelet decomposition when error, due to numerical representation of their parameters, is introduced. In the following section, MRA and HRBFs are introduced with special attention to digital implementation aspects. In Section III, a comparison between

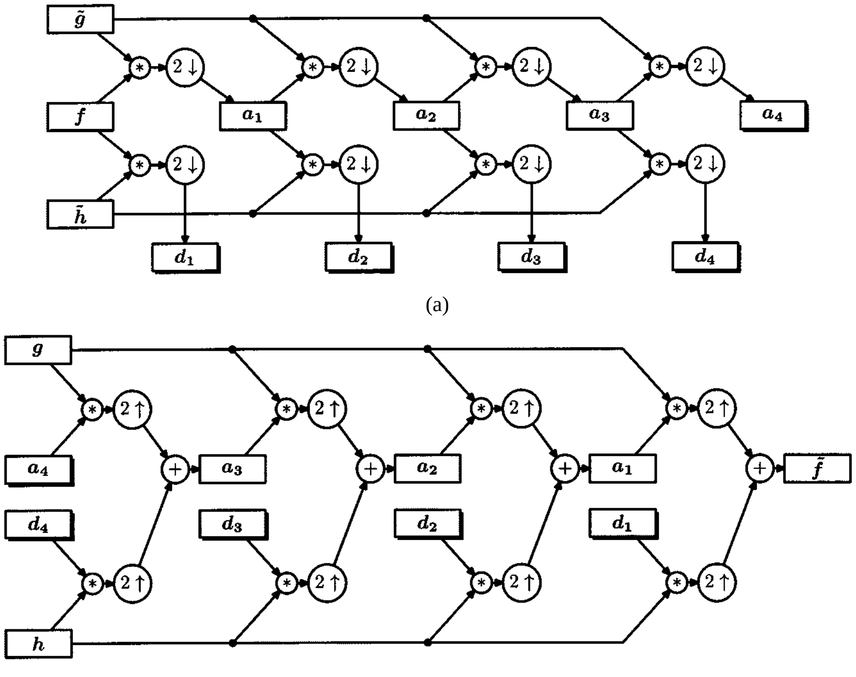
Manuscript received May 4, 2000; revised May 9, 2001.

S. Ferrari is with the Department of Electronics and Information, Politecnico di Milano, Milano, Italy (e-mail: sferrari@elet.polimi.it).

N. A. Borghese is with the Laboratory of Human Motion Analysis and Virtual Reality—MAVR, Istituto Neuroscienze e Bioimmagini—CNR, Scientific Institute H. S. Raffaele, LITA, Segrate (MI) Italy (e-mail: borghese@inb.mi.cnr.it; http://www.inb.mi.cnr.it/Borghese.html).

V. Piuri is with the Department of Information Technologies, University of Milan, Crema (CR), Italy (e-mail: piuri@elet.polimi.it).

Publisher Item Identifier S 0018-9456(01)07346-6.



(a)

(b)

Fig. 2. Implementation schemes for analysis and synthesis with MRA. (a) MRA determination of the coefficients; (b) MRA reconstruction.

the theoretical and experimental properties of the two frameworks is carried out.

II. Methodological Background

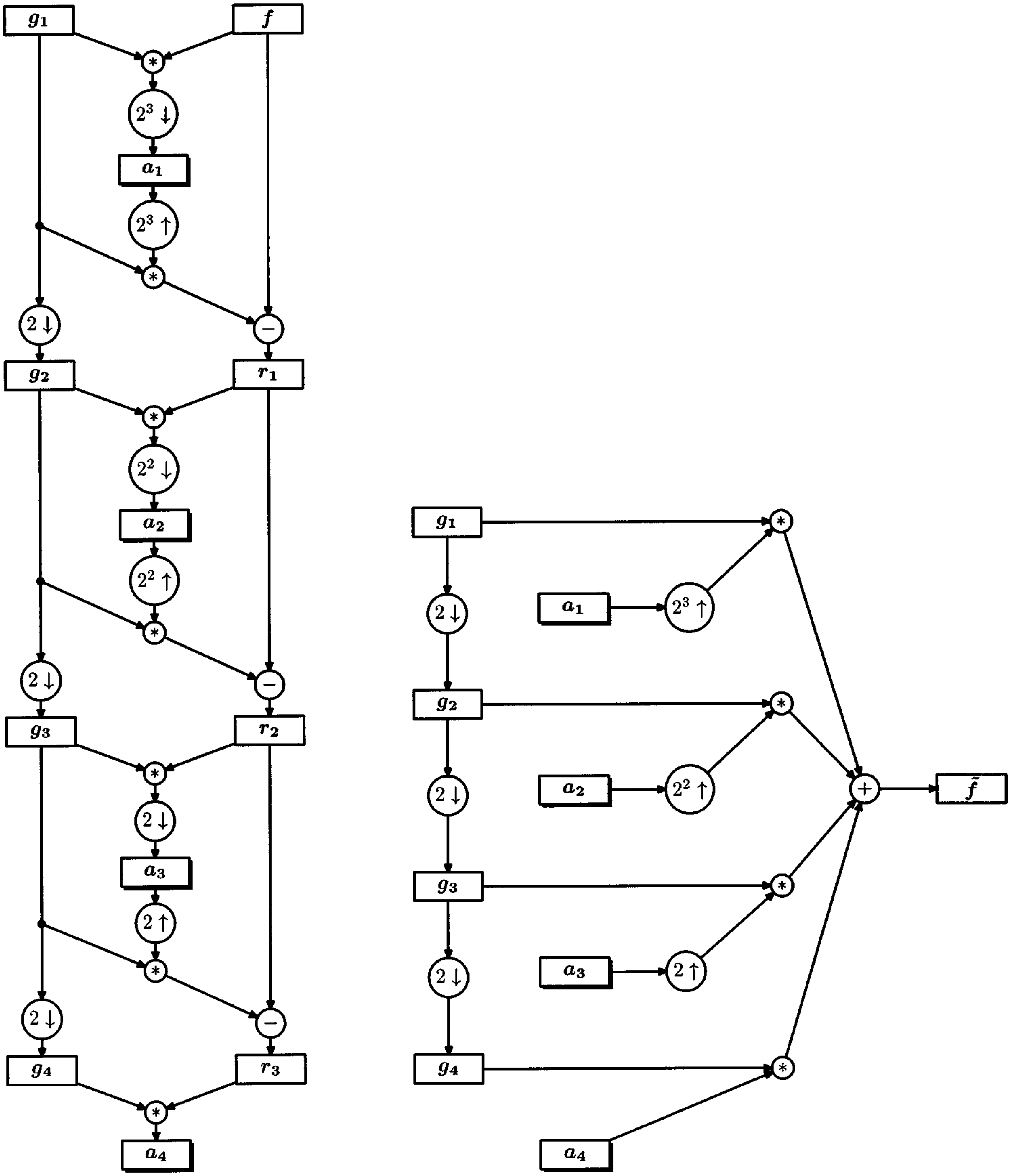
A. Multiresolution Analysis

An MRA is a sequence of function spaces $\{V_j\}_{j \in Z}$ where each space completely includes the previous ones $(V_{j-1} \subseteq V_j)$. These spaces are completely characterized by a *scaling function* $\varphi(\cdot)$, as, for each j, the set of functions $\{\varphi_{j,k}(\cdot) \,|\, k \in Z\}$ where $\varphi_{j,k}(x) = \sqrt{2^j}\varphi(2^j x - k)$ constitutes a basis for V_j. That is, each space V_j is spanned by a set of suitably scaled, dilated, and translated copies of $\varphi(\cdot)$. Thanks to the scaling factor 2^j, the scale of $\varphi(\cdot)$ doubles every layer. Approximations at the j-th scale are obtained as a linear combination of $\varphi_j(\cdot)$ and are contained in the corresponding V_j. As the union of $\{V_j\}$ is dense in L_2—the space of the square integrable functions—every signal $f(\cdot)$ belonging from this functional space (which contains all the common functions) can be reconstructed with an arbitrary accuracy. Therefore, it is possible to define a set of spaces $\{W_j\}$ such that they complement $\{V_j\}$ in $\{V_{j+1}\} : V_{j+1} = V_j \oplus W_j$. The $\{W_j\}$ are the function spaces that contain the details: the portion of $f(\cdot)$ which is contained in V_{j+1} but not in V_j. The spaces $\{W_j\}$ are characterized by a single function $\psi(\cdot)$ called a *wavelet*. Analogously to the approximation spaces V_j, each space W_j is spanned by the base $\{\psi_{j,k}(\cdot) \,|\, k \in Z\}$, where $\psi_{j,k}(x) = \sqrt{2^j}\psi(2^j x - k)$. The details are therefore represented as a linear combination of (equally spaced) translated copies of $\psi(\cdot)$, whose scale also doubles every layer. The coefficients can be obtained by projecting the measured signal onto the wavelets and the scaling functions. Hence, a given function $f(\cdot) \in L_2$ can be represented as the sum of an approximation $a_j(\cdot)$ and a detail $d_j(\cdot)$ at the scale j. Applying an iterative decomposition process J times the function $f(\cdot)$ can be represented as the sum of an approximation and J details at a decreasing scale [Fig. 1(a)–(b)].

When the signal is digitized, MRA theory allows one to design a fast algorithm for wavelet decomposition, the *cascade algorithm*, which decomposes the signal by convolving it with two (suitable) mirror quadrature finite impulse response (FIR) filters [8]. To obtain the multiresolution description, the convolution is iterated on the coefficients, which are obtained at each pass of the convolution.

More formally,[1] given a sequence $\boldsymbol{f}$ obtained by regularly sampling the signal $f(\cdot)$, the approximation coefficients of the first layer are obtained as $\boldsymbol{a}_1 = \downarrow 2(\boldsymbol{f} * \tilde{\boldsymbol{g}})$, and the detail coefficients as $\boldsymbol{d}_1 = \downarrow 2(\boldsymbol{f} * \tilde{\boldsymbol{h}})$, where $\tilde{\boldsymbol{g}}$ and $\tilde{\boldsymbol{h}}$ are respectively the lowpass and the highpass decomposition FIR filters (called *dual* filters) corresponding to the considered MRA. $\downarrow 2(\cdot)$ is a sub-sampling operator: it discards one of every two samples. The procedure is iterated in the higher layers using the coefficients computed in the previous layer: $\boldsymbol{a}_j = \downarrow 2(\boldsymbol{a}_{j-1} * \tilde{\boldsymbol{g}})$ and $\boldsymbol{d}_j = \downarrow 2(\boldsymbol{a}_{j-1} * \tilde{\boldsymbol{h}})$. After n iterations, the signal $\boldsymbol{f}$ is represented by the following collection of coefficients $\{\boldsymbol{a_n}, \boldsymbol{d_1}, \ldots, \boldsymbol{d_n}\}$. Given $\boldsymbol{a_n}$ and $\boldsymbol{d_n}$, the approximation at the next higher resolution level can be computed as $\boldsymbol{a_{n-1}} = \uparrow 2(\boldsymbol{a_n}) * \boldsymbol{g} + \uparrow 2(\boldsymbol{d}_n) * \boldsymbol{h}$, where $\boldsymbol{g}$ and $\boldsymbol{h}$ are the lowpass and highpass reconstruction FIR filters (also called *primal* filters). $\uparrow 2(\cdot)$ is a super-sampling operator, which inserts zeros in between the coefficients. The filters $\boldsymbol{g}$ and $\boldsymbol{h}$ are related by orthogonality or biorthogonality condition to $\tilde{\boldsymbol{g}}$ and $\tilde{\boldsymbol{h}}$. The decomposition and reconstruction procedure through the cascade algorithm are schematized, respectively, in Figs. 2(a)–(b).

[1] In order to simplify the notation, vectors will be typed in bold italics, with function and scalar in plain italics.



(a) (b)

Fig. 3. Implementation schemes for analysis and synthesis with HRBF. (a) HRBF determination of the coefficients; (b) HRBF Reconstruction.

Analytically, the original signal $f(x)$ can be represented as

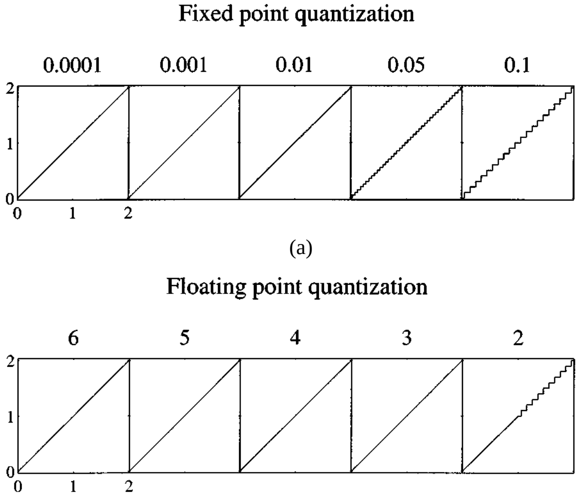
$$\tilde{f}(x) = a_n(x) + \sum_{j=1}^{n} d_j(x) \tag{1}$$

where $a_n(\cdot)$ is the approximation at the n-th level of resolution $a_n(x) = \Sigma_K \boldsymbol{a_n}_{,k}\varphi_{n,k}(x)$. Similarly, the details functions $d_j(\cdot)$ are a linear combination of the wavelet functions: $d_j(x) = \Sigma_k \boldsymbol{d_j}_{,k}\psi_{j,k}(x)$. Substituting the previous expressions in (1), the result is

$$\tilde{f}(x) = \sum_k a_{n,k}\varphi_{n,k}(x) + \sum_{j=1}^{n}\sum_k d_{j,k}\psi_{j,k}(x). \tag{2}$$

B. Hierarchical Radial Basis Functions

In their original formulation [3], [4], HRBF networks have been designed for range data (sparse data); their configuration



(a)

(b)

Fig. 4. Visual representation of the quantization error in (a) fixed and (b) floating point notation. The points of the straight line are represented with the five precision levels used in our experiments.

algorithm is reframed here to the regular sampling case. An HRBF network is composed of a hierarchical set of sub-networks $\{a_j(\cdot)\}$ called *layers*. The j-th layer is composed of regularly spaced Gaussian units, with the same variance σ_j, and its output is constituted of a linear combination of these

$$\begin{aligned}\tilde{a}_j(x) &= \sum_k a_{j,k} g_{j,k}(x)\\ &= \sum_k a_{j,k} \exp\left(-(x-k\Delta x_j)^2/\sigma_j^2\right)/(\sqrt{\pi}\sigma_j)^D.\end{aligned} \tag{3}$$

where D is the input space dimension.

The ensemble of the Gaussian units of each layer, j, can therefore be seen as a function basis which spans the input space at the scale j. The first layer, $\tilde{a}_1(\cdot)$, features the largest scale and it captures only the average behavior of the measurement field. The higher layers feature smaller scales and are devoted to reconstructing the details. To the scope, a residual function is computed at the output of each layer j, as the difference between the signal and the sum of the output of the first j layers

$$r_j(x) = f(x) - \sum_{k=0}^{j} \tilde{a}_j(x). \tag{4}$$

The configuration procedure is depicted in Fig. 1(c).

When x is sampled with sampling step Δx_j, (3) can also be regarded as a low-pass Gaussian filter. Under this perspective, it is possible to determine a relationship between the scale σ_j and the spacing between the units Δx_j by giving a maximum attenuation in the passband and a maximum amplitude in the stopband [9]. Moreover, linear filtering theory can be used to design a noniterative configuration algorithm. The simplest choice is to substitute the residual $r_{j-1}(\cdot)$, sampled in the points $k \cdot \Delta x_j$ times Δx_j, to the coefficients $a_{j,k}$ in (2). This is a poor choice when samples are affected by noise. Exploiting the correlation between neighbor data, a better result can be obtained.

We propose here a schema that takes full advantage of the regular spacing of the data points. We start observing that each $a_{j,k}$ is proportional to the projection of the residual $r_{j-1}(\cdot)$ onto the base function $g_{j,k}(\cdot)$

$$a_{j,k} = \sum_x g_{j,k}(x) r_{j-1}(x) = \sum_x g_j(x - x_k) r_{j-1}(x). \tag{5}$$

This is equivalent to the convolution of the residual with a Gaussian filter, evaluated in the position occupied by the center of the Gaussian basis. Hence, an FFT can be used in the coefficients computation, followed by a proper down-sampling to single out the values corresponding to the Gaussian centers. The $a_{j,k}$ can therefore be computed as

$$\boldsymbol{a_j} = \downarrow 2^{M-j}(\boldsymbol{r_{j-1}} * \boldsymbol{g_j}) \tag{6}$$

where M is the maximum number of layers.

The function samples $\boldsymbol{f}$ are considered as residuals for the first layer ($\boldsymbol{r}_0 = \boldsymbol{f}$). The whole set of sampled data points is considered in computing the weights $a_{j,k}$. We explicitly notice that (6) is equivalent to the analysis through filtering in the MRA.

In a similar way, the output of the j-th HRBF layer $\tilde{\boldsymbol{a}}_j$ can be computed by up-sampling the coefficients vector $\boldsymbol{a}_j$ and convolving the resulting vector with a Gaussian filter $g_j(\cdot)$

$$\tilde{\boldsymbol{a}}_j = \uparrow 2^{M-j}(\boldsymbol{a_j}) * \boldsymbol{g_j}. \tag{7}$$

Then, the residual $\boldsymbol{r}_j$ can be computed as

$$\boldsymbol{r}_j = \boldsymbol{r}_{j-1} - \tilde{\boldsymbol{a}}_j = \boldsymbol{r}_{j-1} - \uparrow 2^{M-j}(\boldsymbol{a}_j) * \boldsymbol{g}_j. \tag{8}$$

This schema is based on projecting the sampled data on the same Gaussian basis j and then down-sampling the result.

The filter $g_j(\cdot)$ is obtained from $g_{j-1}(\cdot)$ by contracting it by a factor of two, similarly to the MRA scaling function

$$g_{j+1,k}(x) = 2g_{j,k}(2x). \tag{9}$$

Down-sampling (6) and contraction of the Gaussian basis (9) by a factor of two is not mandatory in the HRBF as it was in the MRA framework. An arbitrary integer value can be adopted and even a different value for each of the layers.

The configuration procedure is iterated until a stop criterion is met, e.g., the predetermined number of levels is reached or until a uniform error over the entire input domain is achieved. At the end, the original signal $\boldsymbol{f}$ is represented by the collection of HRBF approximation coefficients $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a_n}\}$ [Fig. 3(a)] as

$$\tilde{f}(x) = \sum_{j=1}^{n} a_j(x) = \sum_{j=1}^{n} \sum_k a_{j,k} g_{j,k}(x). \tag{10}$$

Once the network has been configured, it offers in a fast way a multiscale approximation of the signal [Fig. 3(b)].

III. Comparison

A. Theoretical Comparison

Although HRBF and MRA both offer a multiresolution approximation, they work in a different way. MRA decomposes the signal, decreasing the level of detail layer by layer

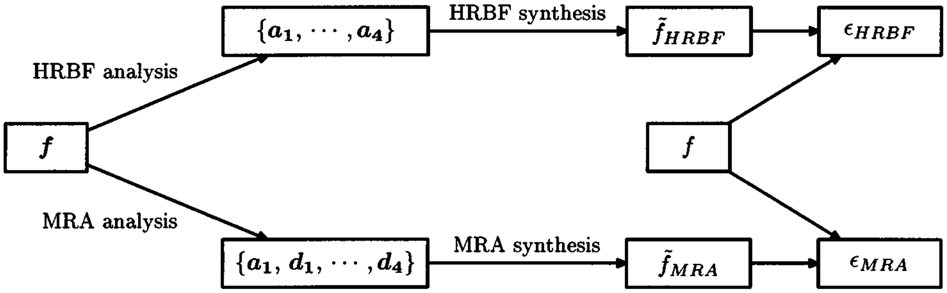


Fig. 5. Schematic representation of the procedure followed to compare the reconstruction accuracy in MRA and HRBF.

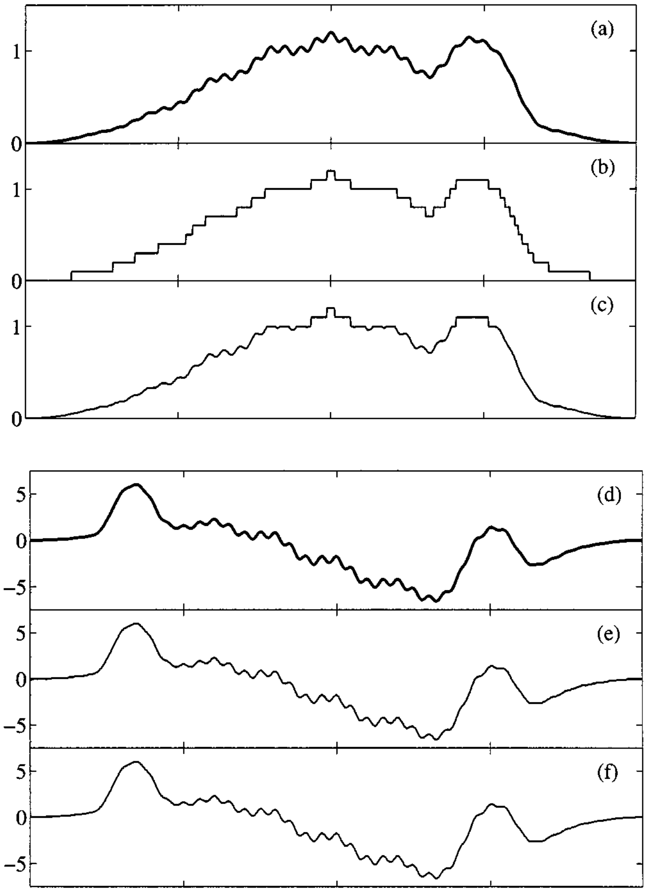


Fig. 6. The two functions used in the experiments are shown in panels (a) and (d). The same data are shown when represented with fixed (b) and (e) and floating point (c) and (f) quantization.

[Fig. 1(a)], where the first layer is constituted by measured samples of the signal. HRBF, instead, works the other way around, i.e., the least detailed approximation is obtained first (Gaussian basis with large variance) and details are progressively added as the number of layers increases and the variance decreases [Fig. 1(c)].

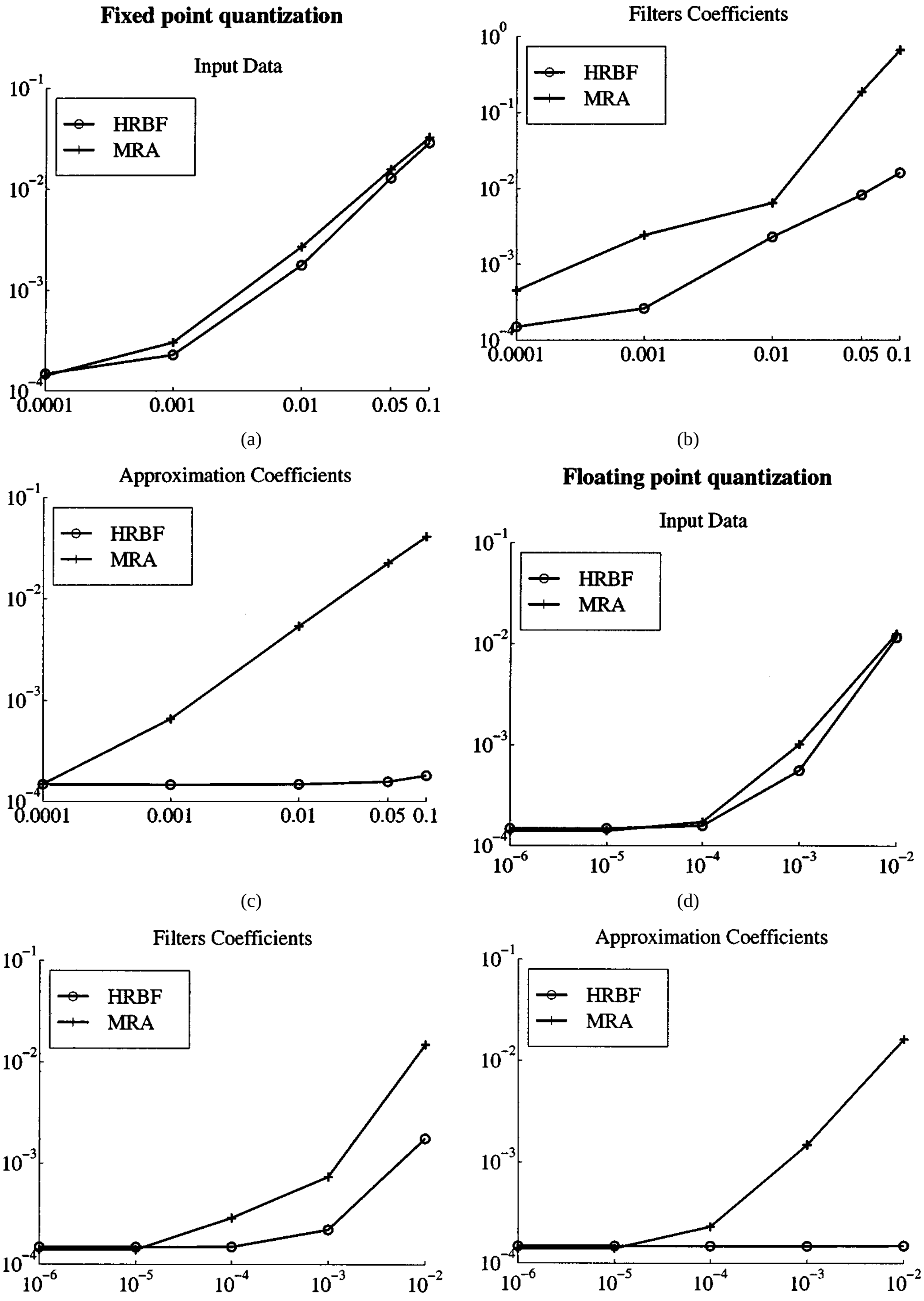
MRA filters are generally shorter than a digital FIR implementation of the Gaussian filter, which, due to its large transition band, spans at least eight samples [4]. Moreover, the number of coefficients in a MRA is equal to the number of data points N, while in HRBF the number is equal to $2N-1$. However, in practical applications, after zeroing the smallest coefficients, the number of coefficients left is usually the same. HRBF, on the contrary, is much simpler as it needs only one filter while MRA requires two pairs $(\boldsymbol{g}, \boldsymbol{h}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{h}})$, one for analysis and one for synthesis.

While in MRA, the reconstruction at the smallest scale, determined by the sampling step, always has to be computed (MRA decomposition starts from the finest approximation), in HRBF, the smallest scale can be determined at run-time, stopping the analysis process at an adequate scale by examining the residuals. The only information required by HRBF, in this computation schema, is the maximum number of layers required by (6) and (7). As shown by the experimental results, the configuration algorithm is also more error tolerant as it is based on the residual, which can easily incorporate errors in the previous computations, in contrast with MRA, which works on approximations. Moreover, many iterative decomposition used in MRA do not have an analytical expression and are able to give the measurement field values only in the sampled points. The reconstruction in between two samples has to be interpolated, while with HRBF, a continuous measurement field is directly output. Moreover, MRA is cast for a digital implementation, while HRBFs are suitable for both a digital and a hybrid implementation where the coefficients are stored digitally and the Gaussian filters can be either digitalized as a FIR filter or computed analogically.

B. *Experimental Comparison*

The approximation quality achieved by MRA and HRBF can be appreciated by comparing quantitatively the original and the reconstructed signals. The quality of the approximation depends, among other factors, on the numerical accuracy in the representation of the parameters.

To assess their impact, we simulated the effect of quantization in the representation of the parameters in a fixed and floating point representation for both MRA and HRBF algorithms. Fixed-point notation is extensively used in a hardware implementation since it allows circuitry simplification. The use of this notation involves—given the fixed number of bits available—a compromise between the range of the possible numbers and the resolution (i.e., the gap between two consecutive possible numbers). To simulate the fixed-point implementation, the parameter values are constrained to assume the closest of a finite number of (equally spaced) values. Reconstruction accuracy is evaluated here for quantization steps of $1/10000, 1/1000, 1/100, 1/50$, and $1/10$ of the maximum absolute value of the parameters [cf. Fig. 4(a)].



(a)

(b)

(c)

(d)

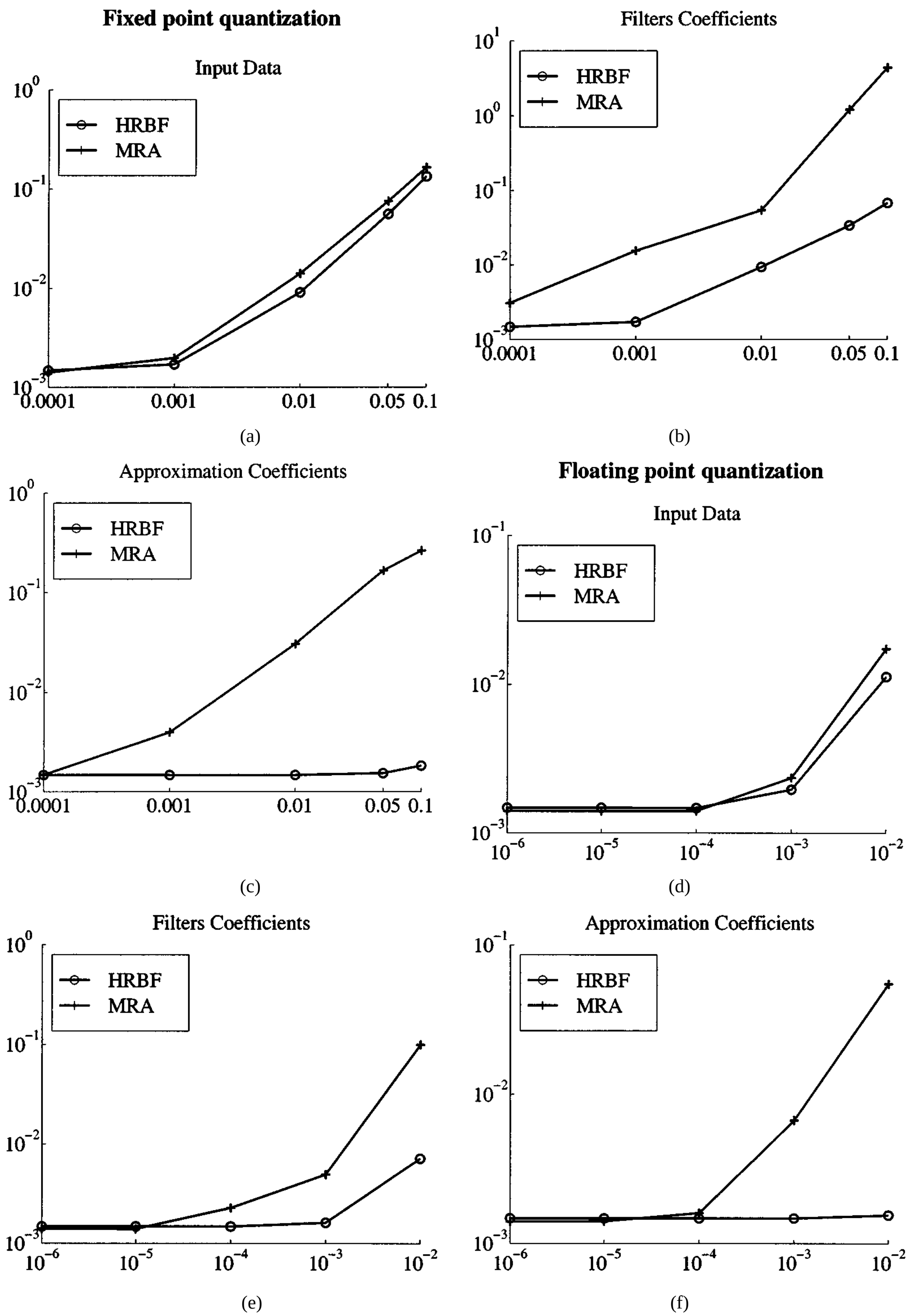
(e)

(f)

Fig. 7. Generalization error for the functions in Figs. 6(a)–(c). Error on: input data (a), filter coefficients (b), and approximation coefficients (c) with fixed point quantization. The same figures with floating point representation are reported in panels (d), (e), and (f).

Floating point notation allows one to represent the data up to a given relative accuracy. In the floating-point representation, the parameters are rounded according to the given number of bits allocated for the mantissa and the exponent. In our simulation, we pose a limit only on the number of bits used to represent the mantissa. This is equivalent to assigning a relative precision of the actual parameters (truncation). We used from two to six decimal digits to represent the mantissa [cf. Fig. 4(b)].

Each experimental session is characterized by the type of the quantization error (floating or fixed point) and by the parameter sets affected; the latter can be either the input data $\boldsymbol{f}$ or the filters coefficients, $\boldsymbol{g}$ for HRBF and $\{\boldsymbol{g}, \boldsymbol{h}, \tilde{\boldsymbol{g}}, \tilde{\boldsymbol{h}}\}$ for MRA, or the basis functions coefficients, $\{\boldsymbol{a}_1, \ldots, \boldsymbol{a_n}\}$ for HRBF and $\{\boldsymbol{a_n}, \boldsymbol{d}_1, \ldots, \boldsymbol{d_n}\}$ for MRA. In each session, HRBF and MRA receive the input data $\boldsymbol{f}$ and calculate the approximation coefficients (Fig. 5). Computation precision is considered infinite for



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 8. Generalization error for function in Fig. 6(d)–(f). Error on: input data (a), filter coefficients (b), and approximation coefficients (c) with fixed point quantization. The same figures with floating point representation are reported in panels (d), (e), and (f).

the scope of the experiment, i.e., only the effects of the limited precision in the numerical representation of the entities are considered.

Error in the reconstruction is assessed through a data set f_V different from the data set f used for the coefficient's computation. The f and f_V data are obtained by regularly sampling the functions $f(\cdot)$ reported in Fig. 6 in 1001 and 32 000 points, respectively. Due to the high sampling density used to obtain it, f_V can be considered a reasonable approximation of $f(\cdot)$. Therefore, it can be used to measure the generalization error of the two models. The results achieved are quite general and have been obtained for many test functions. For sake of comparison, we reported here two functions with different amplitude and shape.

In the six plots of Figs. 7 and 8, the quantitative results obtained by representing the three sets of parameters in limited precision (fixed and floating point) are reported. The reconstruction accuracy is measured as the standard deviation of the (signed) difference between the original function and the synthesized one. Biorthogonal wavelets 3.7, among the most commonly used, have been employed in the MRA experiments.

IV. Results and Conclusions

As Figs. 7 and 8 show, HRBF does not lose accuracy when the representation precision is limited to the configuration procedure (error on the filter and the approximation coefficients). This is due to two main reasons. First, MRA can lose biorthogonality of its basis functions. Second, the residual computed in the HRBF configuration procedure for each layer [see (4)] incorporates any error introduced in the output of the previous layer (e.g., errors in the filter coefficients, in the approximation coefficients, or in the computation). This process allows error compensation in the higher layers. The error introduced in the last layer cannot be corrected, but, as the residual amplitude decreases with the number of layers, only a very small amplitude in the last layer occurs, and its impact on the reconstruction is greatly reduced. This is not the case in MRA, where the error in the coefficients propagates through the cascade algorithm. Overall, the results suggest that HRBF networks can be a much more robust tool for hardware implemented multiresolution reconstruction of measurement fields.

References

[1] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 11, pp. 674–693, July 1989.

[2] I. Daubechies, "CBMS-NSF regional conference series in applied mathematics," presented at the Ten Lectures on Wavelets, vol. 61, 1992.

[3] N. A. Borghese and S. Ferrari, "Hierarchical RBF networks in function approximation," *Neurocomputing*, vol. 19, pp. 259–283, 1998.

[4] N. A. Borghese, M. Maggioni, and S. Ferrari, "Multiscale approximation with hierarchical radial basis functions networks," Dept. Elect. Inform. Politecnico di Milano, Tech. Rep. 99–67, 1999.

[5] C. Alippi, E. Casagrande, V. Piuri, and F. Scotti, "Composite real-time image processing for railways track profile measurement," *IEEE Trans. Instrum. Meas.*, vol. 49, pp. 559–564, June 2000.

[6] C. Alippi and M. Moioli, "A poly-time analysis of robustness in feed-forward neural networks," *Proc. IEEE-VIMS 2001*, May 19–20, 2001.

[7] C. Alippi and L. Briozzo, "Accuracy vs. precision in digital VLSI architectures for signal processing," *IEEE Trans. Comput.*, vol. 47, pp. 472–477, Apr. 1998.

[8] G. Strang and T. Nguyen, *Wavelets and Filters Banks*. Cambridge, MA: Wellesley-Cambridge, 1996.

[9] S. Ferrari and N. A. Borghese, "A portable modular system for automatic acquisition of 3-D objects," *IEEE Trans. Instrum. Meas.*, vol. 49, pp. 1128–1136, Oct. 2000.

Stefano Ferrari received the "laurea" in computer science from University of Milan, Milan, Italy, in 1995 and obtained the Ph.D. degree in computer engineering from Politecnico di Milano, Milan, Italy, in 2001.

He collaborated with the Bioengineering Center, Fondazione ProJuventute, Milano and MAVR, INB in 1996–1997. His research interests are soft-computing algorithms, their high level architectural specification, and their application in the areas of image processing and virtual reality.

N. Alberto Borghese (M'96) received the "laurea" in electrical engineering from Milan Politecnico, Milan, Italy, with 100/100 cum laude.

He joined the Institute of Neuroscience and Bioimaging in 1988, where he is currently Director of the Laboratory for Human Motion Analysis and Virtual Reality (MAVR). He was a Visiting Scientist at the Center for Neural Engineering, University of Southern California, Los Angeles, in 1991–1992; and at the Department of Electrical Engineering of the California Institute of Technology, Pasadena, CA, in 1992. His research interests include quantitative human motion analysis, modeling and synthesis in virtual reality, and artificial learning systems. He is member of the IEEE.

Vincenzo Piuri (S'84–M'86–SM'96) received the Ph.D. degree in computer engineering from Politecnico di Milano, Milan, Italy, in 1989.

From 1992 to 2000, he was Associate Professor of operating systems at Politecnico di Milano. Currently, he is a Full Professor at the University of Milan, Milan, Italy. His research interests include distributed and parallel computing systems, application-specific processing architectures, fault tolerance, neural network architectures, theory and applications of neural techniques for identification, prediction, and control. The original results of this research have been published in more than 100 papers in book chapters, international journals, and proceedings of international conferences.

Dr. Piuri is a member of ACM, IMACS, INNS, and AEI. In the IEEE Instrumentation and Measurement Society, he is founding Co-Chair of both the Technical Committee on Emergent Technologies and the Technical Committee on Intelligent Measurement Systems. He is member of the IMACS Technical Committee on Neural Networks.