

THESIS



LIBRARY Michigan State University



.

This is to certify that the

dissertation entitled

MULTIWEIGHT OPTIMIZATION IN OPTIMAL BOUNDING ELLIPSOID ALGORITHMS

presented by

Dale Joachim

has been accepted towards fulfillment of the requirements for

Ph.D. degree in Electrical Eng

John R. Delle Ju Major professor Date 26 Que

MSU is an Affirmative Action/Equal Opportunity Institution

0-12771

PLACE IN RETURN BOX to remove this checkout from your record. TO AVOID FINES return on or before date due. MAY BE RECALLED with earlier due date if requested.

DATE DUE	DATE DUE	DATE DUE

1/98 c/CIRC/DateDue.p85-p.14

MULTIWEIGHT OPTIMIZATION IN OPTIMAL BOUNDING ELLIPSOID ALGORITHMS

By

Dale Joachim

A DISSERTATION

Submitted to Michigan State University in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Department of Electrical Engineering

1998

ABSTRACT

MULTIWEIGHT OPTIMIZATION IN OPTIMAL BOUNDING ELLIPSOID ALGORITHMS

By

Dale Joachim

Optimal Bounding Ellipsoid (OBE) algorithms offer an attractive alternative to traditional least squares methods for identifying linear-in-parameters signal and system models due to their low computational efficiency, superior tracking ability, and selective updating that permits processor sharing among tasks.

In existing OBE algorithms, optimization takes place *pointwise*; all conditions (in particular, previous weights) extant at the time of optimization remain fixed. The *globally* optimal solution at a given time would diminish the solution set in light of *all* observations.

This research introduces a new class of OBE algorithms with improved convergence speed and tracking capability, the multiple weight optimal bounding ellipsoid (MW-OBE) algorithms. Given a system of order m, a MW-OBE algorithm "revisits" K past weights when the observation set at time n is received and deemed innovative, so that the ellipsoid is optimally diminished with respect to the current and past K observations, conditioned upon information known at time n - K - 1. The simultaneous optimization over multiple weights in MW-OBE algorithms offers greater flexibility with respect to conventional methods in shaping the hyperellipsoid, thus potentially decreasing the solution set. This dissertation derives a general MW-OBE algorithm form. This general algorithm is then optimized in the framework of two existing OBE algorithms, the quasi-OBE (QOBE) and set membership-weighted least square (SM-WRLS). Simulation results are then presented, demonstrating the potential of the developed algorithms, MW-QOBE and MW-SM-WRLS. Copyright by Dale Joachim

1998

.

Acknowledgments

First and foremost, thanks be to God, who probably watches me with great amusement and concern, for the gift of life and the opportunity to learn. I acknowledge my ancestors and the many who have sacrificed heavily to improve my world.

I am greatly indebted to my dissertation advisor, Dr. John R. Deller, Jr., for his guidance, many hours of discussion and countless words of encouragement (some subliminal.) Sincere thanks to Dr. Majid Nayeri, for his invaluable ideas and comments during the initial phases of my research, and to the other members of my Ph.D. guidance committee, Drs. Percy Pierre, Michael Frazier and Robert Nowak, I remain indebted.

To Dr. Sonneborn in the Department of Mathematics and Dr. Khalil in the Department of Electrical Engineering, who took the time to discuss my linear algebra problems; to the many graduate students who made themselves available to discuss difficult problems of absolutely no concern to them; to Dr. O'Kelly for proof-reading my dissertation; and to the department secretaries for their continued support, I remain indebted.

My five years at Michigan State University would not have been possible without the reliable funding of the National Science Foundation, the Department of Electrical Engineering, and the Office of the Associate Dean.

To my family and relatives who observed my struggles through these not-so-easy times with great concern; to Neighbour for friendship and stimulating discussions; to the many comrades who made MSU enjoyable, I am indebted.

To Dramey, who, with whimsical flair, disrupted my linear approach to life, I remember music.

I remain indebted.

Contents

L	ist o	f Tables	viii
L	ist o	f Figures	x
1	Int	roduction and Background	1
	1.1	Introduction	1
	1.2	OBE in system identification	2
	1.3	Review of OBE algorithms	5
	1.4	Motivation for a new algorithm	9
	1.5	Notation, variables and acronyms	9
2	OB	E with Multiple Weight Optimization	13
	2.1	Introduction and overview	13
	2.2	Recursions	15
	2 . 3	A posteriori error vector and energy matrices	17
		2.3.1 A posteriori error, $\varepsilon_{n n}$	17
		2.3.2 A posteriori weighted energy matrix, $G_{n n}$	18
	2.4	Ellipsoid volume	19
	2.5	Algorithm	20
	2.6	Computational costs	21
	2.7	Optimizations	23
3	Mu	ltiple Weight Quasi-OBE Algorithm	25
	3.1	Introduction	25
	3.2	κ_n minimization	25
	3.3	Existence and uniqueness of an optimal solution	30
	3.4	Incremental gain	34
	3.5	Geometric interpretation of the κ_n minimization	35
	3.6	Algorithm	35
	3.7	Computational cost	37
	3.8	MW-QOBE with $K = 1$	38
	3.9	Illustrative examples	39
		3.9.1 Decrease in ellipsoid volume	39
		3.9.2 Weight assignments	40
4	Mul	tiple Weight Set-Membership Weighted RLS	45
	4.1	Introduction	45
	4.2	Volume minimization	45
	4.3	Optimal solution existence and uniqueness	47
	4.4	MW-SW-WRLS with $K = 1$	54
	4.5	Algorithm	57
	4.6	Illustrative examples	59
	~	4.6.1 Decrease in ellipsoid volume	59

		4.6.2 Weight assignments	59
	4.7	Volume minimization, alternative approach	60
5	Al	ternate MW-OBE Algorithms	68
	5.1	Introduction	68
	5.2	Forward-looking and backward-looking MW-OBE	69
	5.3	One-step MW-OBE	73
	5.4	Proof of Theorem 3.5	75
	5.5	Example contrasting forward-looking, backward looking, and one-step	
		MW-OBE algorithms	75
6	Pra	actical Benefits of MW-OBE Algorithms	77
	6.1	Introduction	77
	6.2	QOBE, SM-WRLS, RLS and LMS	77
	6.3	Signal cataloging	81
	6.4	Parameter convergence and volume	81
		6.4.1 MW-QOBE vs. QOBE	84
	6.5	Tracking performance	94
		6.5.1 MW-QOBE vs. QOBE	94
	6.6	Selective updating	100
	6.7	General findings, conclusions, recommendations for practice	101
7	Fut	ure Directions and Complementary Work	105
	7.1	Introduction	105
	7.2	Normalized MW-OBE algorithm	105
	7.3	Hyperellipsoid volume	106
	7.4	Application to linear-prediction analysis of speech signals	108
	7.5	Alternative volume minimization: Trace	111
	7.6		112
	7.7	MW-OBE in the Kalman-Bucy framework	113
	7.8	Conclusions, contributions and further work	114
Aŗ	pene	dix A	118
	A.1	Definitions	118
	A.2	Matlab programs	122

List of Tables

1.1	Vector and matrix notation. A, B represent matrices, a a vector and	
19	x, a scalar	J
1.4	m when considering K past weights n is the "current" time and t	
	represents the general sequence index	1
1 3	Acronyme and abbreviations	1
21	The MW-OBE algorithm	5
2.1	Detailed computational costs of MW-OBE algorithms per update. The	-
2.2	computation of ε_{-1-1} and G_{-1-1} are included although they are usu-	
	ally considered part of the innovation check. For simplicity, the nota-	
	tion \tilde{K} is used to denote $K + 1$.	2
2.3	Computational costs of MW-OBE algorithms per update for $K = 0$.	-
	1, 2 and 3 as functions of m .	2
3.1	The MW-QOBE algorithm (non-zero past weights).	3
3.2	Computational cost of the MW-QOBE per-innovation check. For sim-	
	plicity, the notation \tilde{K} is used to denote $K + 1$.	3
3.3	Computational cost of the MW-QOBE per-observation innovation	
	check when $K = 1$	3
3.4	First 30 weights assigned by QOBE and MW-QOBE ($K = 1,2$) al-	
	gorithms in the identification of the AR(3) system $y_n = 0.49y_{n-1} +$	
	$0.61y_{n-2}+0.58y_{n-3}+\varepsilon_{n*}$, where the "true" measurement error sequence	
	$\{\varepsilon_{n*}\}$ is uniformly distributed over the interval $(-1, +1)$. The QOBE	
	and MW-QOBE ($K = 1,2$) algorithms selected 32, 10, 3 observations,	
	respectively, from a total of 100	4
4.1	Notation used in Chapter 4	4
4.2	Function definitions used in Chapter 4.	4
4.3	The MW-SM-WRLS algorithm for $K = 1$	5′
4.4	The MW-SM-WRLS algorithm (non-zero past weights)	58
4.5	First 30 weights assigned by SM-WRLS and MW-SM-WRLS $(K = 1)$	
	algorithms in the identification of the AR(3) system $y_n = 0.49y_{n-1} + $	
	$0.61y_{n-2} + 0.58y_{n-3} + \varepsilon_{n*}$, where the "true" measurement error sequence	
	$\{\varepsilon_{n*}\}$ is uniformly distributed over the interval $(-1, +1)$. The SM-	
	WRLS and MW-SM-WRLS $(K = 1)$ algorithms selected 51 and 45	
	observations, respectively, from a total of 100	6

5.1	The forward MW-OBE algorithm.	71
5.2	The one-step MW-OBE algorithm (overlapping weight adjustment	
	blocks applied to non-zero past weights.)	74
5.3	Equivalence of the forward and backward MW-QOBE algorithms, and	
	weight assignments in the one-step MW-QOBE. First eight weights	
	assigned in the identification of the AR(2) system $y_n = -0.10y_{n-1} - 0.10y_{n-1}$	
	$0.56y_{n-2} + \varepsilon_{n*}$, where the "true" measurement error sequence $\{\varepsilon_{n*}\}$	
	is uniformly distributed and bounded by 1. MW-QOBE (backward),	
	MW-QOBE (forward) and one-step MW-QOBE used the same number	
	of points.	76
6.1	Parameter vector for the AR systems used in the simulations of Chap-	
	ter 6, both time-invariant (TI) and time-varying. The operator $'$	
	denotes the division remainder	82
6.2	Excitation noises for the AR systems used in the simulations of Chapter 6.	83
6.3	Matrix of systems used in the simulations of Chapter 6. See Tables 6.1	
	and 6.2.	83
7.1	"Other" AR parameter estimators inside the ellipsoid (Figure 7.2). The	
	coordinates and radii $(\ \boldsymbol{\theta}_{i,L} - \boldsymbol{\theta}_{c,L}\ ^2)$ are in relation to the final SM-	
	WRLS ellipsoid center (M = 14, volume = 1.4×10^{13} and 4.67×10^{14}	100
7 0	\leq semi-axes \leq 43.0). \ldots	109
1.2	SM-WRLS 14-dimensional hyperellipsoid semi-axes (see Figure 1.2). Volume $= 4.22 \times 10^{17}$	110
7 2	Volume = $4.32 \times 10^{\circ}$	110
1.3	(see Figures 7.2 and 7.3)	110
71	Poles of the 14th order system (SM-WRLS) based on the ellipsoid	10
	center estimate (see Figure 7.2 and 7.3)	10
		10

List of Figures

1.1	Hyperstrips ω_t and their intersection Ω_n as described in (1.3) for a	
	system of order 2 ($m = 2$) at time $n = 3$	4
1.2	The ellipsoid superset $\Omega_n \supseteq \Omega_n$ corresponding to the system of Figure 1.1.	5
1.3	The polytope, ellipsoid and least squares surface for a system of order	
	2. The "true" and ellipsoid center estimators are denoted by $*$ and \circ	
	respectively.	6
2.1	Weight assignments in the MW-OBE algorithm.	15
2.2	Rectangular error bound constraint (1.2) (dashed line) when $K = 1$	
	and $m = 2$. An error $\varepsilon_{n n-1}$ originally violating the bound condition in	
	light of new observations is mapped to a location inside the rectangle	
	$\{\boldsymbol{u}: \boldsymbol{u} \leq \boldsymbol{\gamma}\}$ in its a posteriori $\boldsymbol{\varepsilon}_{n n}$ form.	19
2.3	Computational costs (number of multiplications) of a single MW-OBE	
	iteration for $K = 0, 1, 2$ and 3 for system orders $m = 2, 4, 8, 12$ and 16.	
	K = 0 represents the conventional OBE algorithm case	22
3.1	Error bound constraint (1.2) when $K = 1$ and $m = 2$. An error $\varepsilon_{n n-1}$	
	originally violating the bound condition is mapped to a location inside	
	the rectangle $\{\boldsymbol{u}: \boldsymbol{u} \leq \boldsymbol{\gamma}\}$ in its a posteriori $\boldsymbol{\varepsilon}_{n n}$ form	36
3.2	OBE ellipsoids resulting from the system identification of AR(2) sys-	
	tem $y_n = -0.10y_{n-1} - 0.56y_{n-2} + \varepsilon_{n*}$ by QOBE (dashed line) and	
	MW-QOBE (solid line, K=1) at times $n = 8$ and 13. The star (*)	
	represents the "true" parameter and the circles (0) the central estima-	
	tors (superimposed). The underlying polytopes (exact feasible sets)	
	are also shown.	41
3.3	Convergence of the parameter estimator $\theta_n(2)$ to the "true" parameter	
	$\theta_{n*}(2) = 0.61$ and associated ellipsoid volume in the system identifi-	
	cation of $y_n = 0.49y_{n-1} + 0.61y_{n-2} + 0.58y_{n-3} + \varepsilon_{n*}$ by QOBE and	
	MW-QOBE $(K=1,2)$ as in Table 3.4.	43
3.4	κ_n and a posteriori error $\varepsilon_{n n}$ associated with the identification of the	
	system of Figure 3.3.	44

.

(6 6. 6. 6. 6.f 6.7 6.8 6.9

4.1	OBE ellipsoids resulting from the system identification of AR(2) sys-	
	tem $y_n = -0.10y_{n-1} - 0.56y_{n-2} + \varepsilon_{n*}$ by SM-WRLS (dashed line) and	
	MW-SM-WRLS (K=1, solid line) at times $n = 8$ and 10. The star (*)	
	represents the "true" parameter and the circles (\circ) the central estima-	
	tors (superimposed). The underlying polytopes (exact feasible sets)	
	are also shown.	64
4.2	Convergence of the parameter estimator $\theta_n(2)$ to the "true" parameter	
	$\theta_{n*}(2) = 0.61$ and associated ellipsoid volume in the system identifica-	
	tion of $y_n = 0.49y_{n-1} + 0.61y_{n-2} + 0.58y_{n-3} + \varepsilon_{n*}$ by SM-WRLS and	
	MW-SM-WRLS ($K=1$) as in Table 4.5	66
4.3	κ_n and a posteriori error $\varepsilon_{n n}$ associated with the identification of the	
	system of Figure 4.2.	67
5.1	Forward-looking and backward-looking MW-OBE algorithms' duality.	69
6.1	Parameter $\theta_*(2) = 0.562$ convergence of QOBE and SM-WRLS vs.	
	that of LMS and RLS in the identification of the AR(2) system $y_n =$	
	$-0.104y_{n-1} + 0.562y_{n-2} + \varepsilon_{n*}$, where ε_{n*} is uniformly distributed over	
	(-1,1). The QOBE and SM-WRLS algorithms used 22 and 121 points,	
	respectively, from a total of 1200 points.	79
6.2	Parameter $\theta_n(2) = 0.562$ convergence and volumes of QOBE and	
	SM-WRLS algorithms in the identification of the AR(2) system $y_n =$	
	$-0.104y_{n-1} + 0.562y_{n-2} + \varepsilon_{n*}$, where ε_{n*} is uniformly distributed over	
	(-1,1). The QOBE and SM-WRLS algorithms used 22 and 121 points	
	respectively from a total of 1200 points.	80
6.3	Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the	
	identification of SYSTEM 1 by MW-QOBE and QOBE algorithms,	
	which used 94 and 108 points, respectively, from a total of 200 points.	86
6.4	Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the	
	identification of SYSTEM 2 by MW-QOBE and QOBE algorithms,	
<u> </u>	which used 75 and 87 points, respectively, from a total of 700 points.	87
6.5	Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the	
	identification of SYSTEM 3 by MW-QOBE and QOBE algorithms,	
	which used 100 and 82 points, respectively, from a total of 700 points.	88
0.0	Parameter $\theta_{n*}(1) = -0.10$ convergence and volumes resulting from the	
	identification of SYSTEM 4 by MW-QUBE and QUBE algorithms,	00
07	which used 11 and 42 points, respectively, from a total of 100 points.	89
0.1	Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the	
	identification of SYSTEM 5 by MW-QUBE and QUBE algorithms,	00
~ •	which used 117 and 116 points, respectively, from a total of 1500 points.	90
6.8	Parameter $\theta_{n*}(2) = 0.92$ convergence and volumes resulting from the	
	identification of SYSTEM 6 by MW-QOBE and QOBE algorithms,	
0 -	which used 174 and 166 points, respectively, from a total of 3000 points.	91
6.9	Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the	
	identification of SYSTEM 7 by MW-QOBE and QOBE algorithms,	
	which used 214 and 203 points, respectively, from a total of 1500 points.	92

6.1	0 Parameter $\theta_{n*}(1) = -0.10$ convergence and volumes resulting from the identification of SYSTEM 8 by MW-QOBE and QOBE algorithms	
	which used 9 and 43 points respectively from a total of 500 points	93
61	1 Parameter $\theta_{-}(1)$ tracking and volumes resulting from the identifica-	00
0.1	tion of SYSTEM 9 by MW-QOBE and QOBE algorithms, which used	
	109 and 110 points respectively from a total of 700 points	96
61	2 Parameter $\boldsymbol{\theta}$ (1) tracking and volumes resulting from the identifica-	50
0.1	tion of SYSTEM 10 by MW-OOBE and OOBE algorithms, which used	
	70 and 72 points respectively from a total of 350 points	07
613	3 Parameter θ (1) tracking and volumes resulting from the identifica-	51
0.1	tion of SYSTEM 11 by MW-OOBE and OOBE algorithms, which used	
	52 and 95 points respectively from a total of 700 points	98
614	4 Parameter $\theta_{-1}(1)$ tracking and volumes resulting from the identifica-	50
0.1	tion of SYSTEM 12 by MW-OOBE and OOBE algorithms, which used	
	37 and 67 points, respectively, from a total of 350 points.	99
6.15	5 Ratio of points taken, averaged over 10 runs, in the identification of	00
0.10	an $AR(2)$ system with uniformly distributed error sequence by QOBE	
	and MW-QOBE (adaptive K .).	101
6.16	Ratio of points taken, averaged over 10 runs, in the identification of	
	an AR(3) system with uniformly distributed error sequence by QOBE	
	and MW-QOBE (adaptive K .)	102
6.17	Ratio of points taken, averaged over 10 runs, in the identification of an	
	AR(12) system with uniformly distributed error sequence by QOBE	
	and MW-QOBE (adaptive K .)	103
6.18	Ratio of points taken, averaged over 10 runs, in the identification of	
	an $AR(2)$ system with uniformly distributed error sequence by QOBE	
	and MW-QOBE (fixed K .)	103
6.19	Ratio of points taken, averaged over 10 runs, in the identification of	
	an AR(3) system with uniformly distributed error sequence by QOBE	
c 00	and MW-QOBE (fixed K .)	104
6.20	Ratio of points taken, averaged over 10 runs, in the identification of an	
	AR(12) system with uniformly distributed error sequence by QUBE	104
71	and MW-QUBE (fixed A.)	104
1.1	in the united /// phoneme. We show the log of the final volumes	
	In the voiced /1/ phoneme. We show the log of the linal volumes	100
79	(normalized to the initial empsoid volume) using the model orders $7-14$.	100
1.2	sequence is shown in (a) and the FFT of the impulse reports of center	
	(\mathbf{A}_{1}) COV $(\mathbf{A}_{2}, \mathbf{a}_{2})$ and non central estimators inside the ellipsoid	
	$(\mathbf{v}_{c,L})$, COV $(\mathbf{v}_{LSE,L})$ and non-central estimators inside the empsoid $(\mathbf{a}_{c,L})$ are shown in (b). The speech segment is modeled by an AB(14)	
	$(v_{1-3,L})$ are shown in (0). The speech segment is induced by an AR(14) system. Ellipsoid volume -1.4×10^{19} and 4.67×10^{-1} < axes < 110.10	
	System. Emploid volume = 1.4×10^{-1} and $4.07 \times 10^{-1} \ge axes \ge 110.10$. Son Tables 7.1. 7.2. 7.3 and 7.4 for more information on estimator and	
	ollipsoid	116
7 2	Polos of the 14th order system based on the COV and ORE ellipsoid	110
, .J	conter estimates (see Figure 7.2. Tables 7.3 and 7.4)	117
	center commates (see rigure 1.2, lautes 1.3 and 1.4).	111

Chapter 1

Introduction and Background

1.1 Introduction

System identification techniques are used in diverse fields, ranging from engineering, to natural and social sciences, to finance and economics. Indeed, many physical problems benefit from modeling and identification. Linear-in-parameters systems form a broad and important class of well-studied models for which extensive analysis tools have been developed. In particular, models that are linear in the data as well as in the parameters have been widely studied and applied in statistics [47], economics [38], biology [4, 42, 48], engineering [8, 19, 33] and other fields [24, 37]. In the *autoregressive* (AR) model, for example, the output at a given time is a linear function of past outputs and an excitation sequence. This relatively simple model has been used to represent many physical systems [26, 45], a notable example of which is speech [9].

Several batch and recursive methods have been developed to identify linear-inparameters system. Of these, set-membership (SM) algorithms are unique in providing a set of feasible parameter vectors (a solution set) instead of a single point estimate. This is achieved through successive refinements of an initial solution set, consistent with a priori constraints on the signal or system model. Optimal bounding ellipsoid (OBE) algorithms belong to the class of recursive SM algorithms and iteratively assign a weight to each incoming data vector that reflects the current observation set's potential to refine the solution set [14]. Each weight is determined by minimizing a measure of the size of an hyperellipsoidal feasibility set to which the "true" parameter vector must belong.

In existing OBE algorithms, optimization takes place *pointwise*; all conditions (in particular, previous weights) extant at the time of optimization remain fixed. The *globally* optimal solution at a given time would diminish the solution set in light of *all* observations.

This research introduces a new class of OBE algorithms with improved convergence speed and tracking capability, the *multiple weight optimal bounding ellipsoid* (MW-OBE) algorithms. Given a system of order m, a MW-OBE algorithm "revisits" K past weights when the observation set at time n is deemed innovative, so that the ellipsoid is optimally diminished with respect to the current and past K observations, conditioned upon information known at time n - K - 1. The number of revisited past weights, K, must be less than the system order m and may be time-varying. The corresponding past data vectors are not required to be sequential. The simultaneous optimization over multiple weights in MW-OBE algorithms offers greater flexibility with respect to conventional methods in shaping the hyperellipsoid, thus potentially better fit to exact feasibility set (described later.)

This dissertation begins with the derivation of a general MW-OBE algorithm form. This general algorithm is then optimized in the framework of two existing OBE algorithms, the *quasi-OBE* (QOBE) [39] and *set membership-weighted least square* (SM-WRLS) [13]. Simulation results are then presented, demonstrating the potential of the developed algorithms, MW-QOBE and MW-SM-WRLS.

1.2 OBE in system identification

Several batch methods are available to identify linear-in-parameters models system models, including minimum squared error (MMSE), maximum likelihood (ML) and least square error (LSE) [25, 26, 45]. The recursive least square (RLS) [26], least mean square (LMS) [25], instrumental variable [45] and optimal bounded ellipsoid (OBE) [8, 14, 17, 21] algorithms are recursive (in the estimates) techniques for use in on-line applications. RLS and LMS require the whiteness of the model disturbance, and they fail to perform adequately in colored noise [26]. OBE algorithms do not impose any statistical requirements on the disturbance, but require that the sequence squared, say $\{\varepsilon_{n*}^2\}$, be pointwise bounded by a known sequence $\{\gamma_n^2\}$ [34]. OBE identification algorithms (e.g., [14, 15, 21]) have strong potential for application to signal-processing problems involving linear-in-parameters models. With respect to conventional least-square-error identification methods (e.g., [26]), OBE identifiers offer superior adaptation, improved accuracy, efficient use of innovation in the data, improved computational efficiency, robustness to measurement noise, robustness to deviation from the assumed input model, a set of feasible solutions rather than a single point estimate, and the ability to compute the solution recursively in time without block processing or windows (e.g., [2, 13, 15, 16]).

OBE algorithms are used to identify linear-in-parameters models of the form

$$y_n = \boldsymbol{\theta}_*^T \boldsymbol{x}_n + \varepsilon_{n*} \tag{1.1}$$

in which $\theta_* \in \Re^m$ is the unknown "true" parameter vector to be identified; $\{x_n\}$ is a sequence of measurable vectors of dimension m; and $\{\varepsilon_{n*}\}$ is a "true" but unknown error sequence. OBE algorithms are based on the premise that, for each n, the model error has a known pointwise energy bound,

$$\varepsilon_{n*}^2 \le \gamma_n^2. \tag{1.2}$$

This "true" model is posed only for analysis purposes and provides the background from which actual parameter vector estimates are derived. Given data on times $t \in [1, n]$, an exact feasibility set, say Ω_n , of estimates for θ_* whose elements are **Consistent** with these bounds is formally described by (Figure 1.1)

$$\Omega_n = \bigcap_{t=1}^n \omega_t, \text{ where } \omega_t = \{ \boldsymbol{\theta} : \varepsilon_{t*}^2 = |y_t - \boldsymbol{\theta}^T \boldsymbol{x}_t|^2 \le \gamma_t^2 \}.$$
(1.3)

OBE algorithms work with an hyperellipsoidal set, say $\bar{\Omega}_n$, that is guaranteed to **Contain** Ω_n , hence θ_* . The observations are scrutinized with respect to their ability **to** "shrink" $\bar{\Omega}_n$, hence to more tightly bound Ω_n . At time *n*, the hyperellipsoid is



Figure 1.1. Hyperstrips ω_t and their intersection Ω_n as described in (1.3) for a system of order 2 (m = 2) at time n = 3.

given by (e.g., [14]) (Figure 1.2)

$$\bar{\Omega}_n \stackrel{\text{def}}{=} \left\{ \boldsymbol{\theta} : (\boldsymbol{\theta} - \boldsymbol{\theta}_n)^T \boldsymbol{C}_n (\boldsymbol{\theta} - \boldsymbol{\theta}_n) \leq \kappa_n \right\}$$
(1.4)

in which C_n is a weighted covariance matrix of the observations,

$$\boldsymbol{C}_{\boldsymbol{n}} = \sum_{t=1}^{n} q_{t,\boldsymbol{n}} \boldsymbol{x}_{t} \boldsymbol{x}_{t}^{T}, \qquad (1.5)$$

 κ_n is the scalar

$$\kappa_n = \boldsymbol{\theta}_n^T \boldsymbol{C}_n \boldsymbol{\theta}_n + \sum_{t=1}^n q_{t,n} (\gamma_t^2 - y_t^2), \qquad (1.6)$$

and θ_n , the center of $\overline{\Omega}_n$, is a weighted least-square-error estimator of θ_* at time *n* (Figure 1.3),

$$\boldsymbol{\theta}_n = \boldsymbol{P}_n \boldsymbol{c}_n, \quad \text{with } \boldsymbol{P}_n \stackrel{\text{def}}{=} \boldsymbol{C}_n^{-1} \text{ and } \boldsymbol{c}_n \stackrel{\text{def}}{=} \sum_{t=1}^n q_{t,n} y_t \boldsymbol{x}_t.$$
 (1.7)

The weighting sequence in this process at time n, $\{q_{t,n}\}_{t=1}^{n}$, is chosen to optimally **dimin**ish some set measure of the hyperellipsoid.



Figure 1.2. The ellipsoid superset $\tilde{\Omega}_n \supseteq \Omega_n$ corresponding to the system of Figure 1.1.

OBE algorithms make selective use of incoming data in updating the ellipsoid and central estimator. Frequently, the observations at time n contain no innovation in the sense that they cannot be used to reduce the size of $\bar{\Omega}_{n-1}$. This is manifest in the failure to find valid weights, and the effort of updating can be avoided at this time. Depending on the properties of the sequence $\{\varepsilon_{n*}\}$, OBE algorithms often update only 10 percent of the time or less.

1.3 Review of OBE algorithms

Schweppe published one of the first OBE-type algorithm in early 1968 [44] in the **context** of estimating state parameters of linear dynamic systems using noisy observations. Assuming bounded inputs and bounded observation error, Schweppe's **algorithm** estimates the state of the system using bounding ellipsoids. However, as **Schweppe** notes [44], this novel algorithm is presented without convergence proof, **Processes** all available data, and is computationally impractical.

Witsenhausen in 1968 [49], and Bertsekas and Rhodes in 1971 [5], tackled the state-estimation problem from a SM approach. Under similar assumptions as those



Figure 1.3. The polytope, ellipsoid and least squares surface for a system of order 2. The "true" and ellipsoid center estimators are denoted by * and \circ respectively.

of Schweppe, Bertsekas and Rhodes examined filtering, prediction and smoothing problems. The algorithm of Schweppe, and of Berstekas and Rhodes have a Kalman-Bucy filter structure which is optimal as a state estimator in Gaussian white noise.

In 1979, Fogel [20] published an OBE identification algorithm for the ARX model (e.g., [36]) based on *a priori* knowledge of the cumulative error energy. Fogel proves the convergence of the hyperellipsoid central estimator to the true parameter in a deterministic setting, by demonstrating that the hyperellipsoid asymptotically reduces to a point set.

In 1982, Fogel and Huang [21] published an OBE algorithm with selective updating (F-H/OBE) which processes only relevant data, a key feature of modern OBE algorithms. This data-selection process is achieved by assigning weights to each incoming data vector, with a zero weight indicating data rejection. A pre-processing information check $\mathcal{O}(m^2)$ determines the possibility of a non-zero weight, thereby potentially eliminating redundant computations. Fogel and Huang present a sufficient condition for the convergence of the F-H/OBE hyperellipsoid to a point, provided the observation error is white noise. The validity of this proof remains controversial, and a more recent proof in a stochastic setting with a more general class of OBE algorithms is presented in the paper by Nayeri *et al.* [40].

The F-H/OBE data selection process was improved to $\mathcal{O}(m)$ complexity in 1987 by Dasgupta and Huang [8] in their OBE-type algorithm (D-H/OBE). By minimizing κ_n , a scalar not apparently related to the hyperellipsoid volume [see (1.4)], Dasgupta and Huang derive a simple but effective algorithm and prove the asymptotic convergence (at an exponential rate) of its central estimator to a region around the true parameter.

Deller et al. introduced the set-membership weighted recursive least squares (SM-WRLS) algorithm in 1989. SM-WRLS is similar to F-H/OBE but is derived in a much different manner [13, 18]. A major difference between the F-H/OBE algorithm, derived from geometric considerations, and SM-WRLS, derived as an RLS algorithm with special weighting, is in the weighting strategy. With the introduction of SM-WRLS, the relationship between OBE and WRLS was formally established. In 1993, the set-membership stochastic approximation (SM-SA) was introduced in a paper that unifies all previously known versions of OBE algorithms [14, 15]. The first stochastic proof of convergence (in probability) of an OBE algorithm is achieved with the SM-SA algorithm in [14], and the unification in [14] implies that the convergence result is generally applicable to all published OBE algorithms.

In 1991, Cheung [6, 7] published the optimal volume algorithm (OVE) based on an affine transformation which reduces the hyperellipsoid volume without imposing the condition that the hyperellipsoid center be equivalent to θ_n . This relaxation on the hyperellipsoid center improves reduction in the hyperellipsoid size with a minimal increase in computational cost.

The compounded effect of (1.2) at each *n* restricts the "true" parameter vector estimate to an *exact polytope*. In the 1991 paper by Veres and Norton [46], it is proved that, under heavy-tailed conditions on $\{\varepsilon_{n*}\}$ (see Appendix A.1) and certain *persistency of excitation* [26] (PE) conditions on $\{x_n\}$, these polytopes converge in probability to a point set.

In spite of the potential benefits of OBE identification, a significant practical impediment precludes widespread application. The practical difficulty concerns the estimation of proper model error bounds which are *theoretically* necessary for reliable identification. Failure to prescribe accurate bounds in OBE processing is potentially

catastrophic. Underestimated bounds may cause divergence to a parameter vector that is not even capable of generating the observed data (i.e. outside the set Ω_n), while overestimated bounds may cause the estimate to "freeze" at a biased estimate. Although improper bounds imply statistical inconsistency in theory, OBE algorithms have been successfully applied to a range of applications in which bounds can only be estimated.

The OBE algorithm with automatic bound estimation (OBE-ABE) developed by Lin in 1996 [34] is the first OBE method to theoretically solve the difficult problem of accurately estimating "true" model error bounds in ARX models with "true" error $\{\varepsilon_{n*}\}$ and exogenous input $\{u_n\}$ (included as part of the measurements $\{x_n\}$). In theory, OBE-ABE removes the practical roadblock to model identification using OBE algorithms. OBE-ABE converges consistently under conditions on the "true" model error sequence $\{\varepsilon_{n*}\}$ that are met by many practical signals, and additionally provides the customary OBE set of feasible solutions. The basis for the "ABE" in OBE-ABE is Lin's proof [34] that, if the error bounds are overestimated, there exists an interval \mathcal{I} of length N over which no update takes place for any finite N. Thus, the need to adjust the bound is practically indicated by a sufficiently long period over which no update takes place. When such an interval is found, OBE-ABE invokes its bound re-estimation recursion which depends on N and an "adjustment constant" ϵ . The choices (and possible iterative refinements) of parameters N and ϵ have a significant impact on OBE-ABE performance in practice [29].

One of Lin's principal results [34] asserts that the OBE-ABE bound re-estimation recursion results in point-set convergence of $\overline{\Omega}_n$ in probability if, in addition to a PE requirement on $\{x_n\}$, and a statistical *infinite visitation* (IV) criterion on $\{\varepsilon_{n*}\}$, $\{\varepsilon_{n*}\}$ and $\{u_n\}$ are asymptotically independent (see Definitions A.1 and A.2 in Appendix A.1). However, the choice of N and ϵ may still pose practical challenges in certain applications [29]. Lin's convergence proof depends upon asymptotic analyses in which $n \to \infty$, $N \to \infty$, and $\epsilon \to 0$.

Huang and his research group at the University of Notre Dame (UND) and Deller working at UND and later at Michigan State University (MSU) recently focused on the quasi-OBE (QOBE) algorithm, an OBE algorithm featuring a very simple innovation data check, similar to that of D-H/OBE [22, 39] but with a weighting strategy similar to SM-WRLS. Working together, these groups proved that in a deterministic setting the parameter estimator asymptotically converges to the "true" parameter vector given PE and disturbance sequence IV of any arbitrary neighborhoods of the true bounds [11, 22]. A convergence proof of the QOBE ellipsoid volume is forthcoming and requires joint conditions on $\{x_n\}$ and $\{\varepsilon_{n*}\}$ that would seem to be rare in practice [11, 39].

1.4 Motivation for a new algorithm

All published OBE algorithms can be manipulated into the formal framework described in Section 1.2, provided that we allow for time-varying ("*n*-dependent") weight sequences as we have done in (1.5) - (1.7) [14]. In all published cases, however, this time dependence (if any) has a simple structure arising from a either a generalized "forgetting factor" (e.g., c.f. [15] and [21]) or some heuristic measures to induce adaptation (e.g., [17]). In no published case is there an attempt to *reoptimize* any of the "previous" weights at time *n* in light of the new measurements x_n and y_n . That is, all optimization in existing OBE algorithms can be accomplished by manipulation of the *current* weight only except for possible inherent scaling of past weights. This can be inferred directly from the work in [14]. The globally optimal solution at time *n* would optimize all weights $\{q_{t,n}\}_{t=1}^{n}$, in light of all known information $\{(x_t, y_t)\}_{t=1}^{n}$.

In this work, we develop an algorithm that can potentially approach the performance of a globally-optimized algorithm at each time, by "revisiting" $K \ge 0$ past weights when data at time *n* are received and deemed innovative. The revision of past weights is made by additive adjustments to existing weight values, subject to the constraint that any revised weight remain nonnegative and that the number of revisited weights be less than the system order.

1.5 Notation, variables and acronyms

The vector and matrix notation, acronyms and abbreviations employed throughout the remaining developments are defined in Tables 1.1 - 1.3.

Table 1.1. Vector and matrix notation. A, B represent matrices, a a vector and x, a scalar.

Notation	Definition
$I A^T$	identity matrix with appropriate dimensions transpose of matrix \boldsymbol{A}
A^{-T}	$\left(\boldsymbol{A}^{-1} \right)^T$ when matrix \boldsymbol{A}^{-1} exists
\overline{A}	adjoint of the matrix A , i.e. $\overline{A} = \det(A)A^{-T}$
$\boldsymbol{A}(*,i)$	N-vector comprising the <i>i</i> th column of A , an $N \times M$ matrix
$\boldsymbol{A}(i, *)$	M-vector comprising the <i>i</i> th row of A , an $N \times M$ matrix
$\boldsymbol{A}(i,j)$	(i, j) element of matrix A
$\searrow(A)$	matrix formed by setting off-diagonal elements of $N \times N$ matrix A to zero
$\boldsymbol{A} \circ \boldsymbol{B}$	Hadamard product of two $N \times M$ matrices A and B defined to be the $N \times M$ matrix with (i, j) element $A(i, j)B(i, j)$ [27]
<i>A</i> > 0	each element of the matrix \boldsymbol{A} is positive (similar notation applies to vectors)
A	matrix with (i, j) element $ A(i, j) $ (similar notation applies to vectors)
A > B	A - B > 0 (similar notation applies to vectors)
sign(A)	matrix with (i, j) element $\frac{A_{(i,j)}}{ A_{(i,j)} }$ (similar notation applies to vectors)
$\boldsymbol{a}(i)$	ith element of vector a
$\mathcal{D}(\boldsymbol{a})$	diagonal matrix with <i>i</i> th diagonal element $a(i)$, where a is an N -vector
d(A)	$N \times 1$ vector with <i>i</i> th element $A(i, i)$, with A an $N \times N$ matrix ∂r
$\frac{\partial X}{\partial A}$	matrix with (i, j) element $\frac{\partial x}{\partial A(i, j)}$.

Table 1.2. MW-OBE vector and matrix variables at time n for a system of order m when considering K past weights. n is the "current" time and t represents the general sequence index.

Quantity	Notation	Dimension
Observation matrix	$X_n \stackrel{\text{def}}{=} \begin{bmatrix} x_{n-K} & \cdots & x_{n-1} & x_n \end{bmatrix}$	Note: $\tilde{K} \stackrel{\text{def}}{=} K + 1$ $m \times \tilde{K}$
Output vector	$\boldsymbol{y}_{-} \stackrel{\text{def}}{=} \begin{bmatrix} y_{n-K} & \cdots & y_{n-1} & y_n \end{bmatrix}^T$	$\tilde{K} \times 1$
Weight adjustments vector	$\boldsymbol{\lambda}_{n} \stackrel{\text{def}}{=} \begin{bmatrix} \lambda_{n-K,n} & \dots & \lambda_{n-1,n} & \lambda_{n,n} \end{bmatrix}^{T}$	$\tilde{K} \times 1$
Weight adjustments matrix	$\Lambda_n \stackrel{ ext{def}}{=} \mathcal{D}(\lambda_n)$	$ ilde{K} imes ilde{K}$
Cumulative weight vector	$\boldsymbol{q}_{t,n} \stackrel{\mathrm{def}}{=} \left[\begin{array}{cccc} q_{t-K,n} & \ldots & q_{t-1,n} & q_{t,n} \end{array} ight]^T$	$ ilde{K} imes 1$
Cumulative weight matrix	$oldsymbol{Q}_{t,n} \stackrel{ ext{def}}{=} \mathcal{D}(oldsymbol{q}_{t,n})$	$ ilde{K} imes ilde{K}$
Error bound vector	$\boldsymbol{\gamma}_{n} \stackrel{\text{def}}{=} \left[\begin{array}{ccc} \gamma_{n-K} & \cdots & \gamma_{n-1} & \gamma_{n} \end{array} \right]^{T}$	$ ilde{K} imes 1$
Error bound matrix	$\Gamma_n \stackrel{\text{def}}{=} \mathcal{D}(\gamma_n)$	$ ilde{K} imes ilde{K}$
Conditional predic- tion error vector	$\boldsymbol{\varepsilon}_{n t} \stackrel{\text{def}}{=} \left[\begin{array}{ccc} \varepsilon_{n-K t} & \cdots & \varepsilon_{n t} \end{array} \right]^T$	$ ilde{K} imes 1$
Conditional predic- tion error matrix	$oldsymbol{E}_{n t} \stackrel{ ext{def}}{=} \mathcal{D}(oldsymbol{arepsilon}_{n t})$	$ ilde{K} imes ilde{K}$
Sign vector (signs of elements are chosen in context)	$\boldsymbol{s_n} \stackrel{\text{def}}{=} \left[\begin{array}{ccc} \pm 1 & \cdots & \pm 1 \end{array} \right]^T$	$\tilde{K} \times 1$
Sign matrix	$\boldsymbol{S_n} \stackrel{\mathrm{def}}{=} \mathcal{D}(\boldsymbol{s_n})$	$ ilde{K} imes ilde{K}$
Instantaneous covariance-weighted observation energy matrix	$G_{n t} \stackrel{\text{def}}{=} X_n^T P_t X_n$	$ ilde{K} imes ilde{K}$
(Inverse) relative en- ergy gain	$\boldsymbol{H}_n \stackrel{\text{def}}{=} \boldsymbol{I} + \Lambda_n \boldsymbol{G}_{n n-1}$	$ ilde{K} imes ilde{K}$

Acronym	Definition
ABE	automatic bound estimation
AR	autoregressive (parametric model)
ARX	autoregressive with exogenous input (parametric model)
IV	infinite visitation
IID	independent and identically distributed
LMS	least mean square (algorithm)
ML	maximum likelihood
MW-OBE	multiple weight optimal bounding ellipsoid (algorithm)
MW-QOBE	multiple weight strategy applied to QOBE (algorithm)
MW-SM-WRLS	multiple weight strategy applied to SM-WRLS (algorithm)
MMSE	minimum mean squared error
OBE-ABE	optimal bounding ellipsoid with automatic bound estima-
	tion (algorithm)
OBE	optimal bounding ellipsoid (algorithm)
OVE	optimal volume ellipsoid (algorithm)
PE	persistently exciting, or persistency of excitation (as appro-
	priate in context)
QOBE	quasi-OBE (algorithm)
SM	set-membership
SM-SA	set-membership-stochastic-approximation (algorithm)
SM-WRLS	set-membership weighted least squares (algorithm)
UCT	uniformly conditionally tailed
WRLS	weighted recursive least squares (algorithm)

Table 1.3. Acronyms and abbreviations

Chapter 2

OBE with Multiple Weight Optimization

2.1 Introduction and overview

This chapter introduces the general class of MW-OBE algorithms and presents background formulation for the MW-QOBE and MW-SM-WRLS algorithms (presented in the following two chapters).

At each time *n*, conventional OBE algorithms update the previous covariance matrix C_{n-1} , by incorporating a weighted outer product of the current data vector, if this vector is deemed innovative (e.g., [14]). This process, although efficient, may result in large ellipsoid volumes [13, 34], often due to the shape and size of the underlying exact polytope [46]. However, when adequate PE and IV conditions are present (Appendix A.1), pointwise reduction in ellipsoid volume may be improved by a joint weight assignment [29, 30]. If Ω_n is too large, adding weights to OBE will not help. A principal objective of this research is to develop methods for taking optimal advantage of the information in the data by optimizing a present and reoptimizing K past weights over a block of measurements.

The block optimization process begins by separating the last K + 1 outer products in the covariance matrix (1.5)

$$\boldsymbol{C}_{n} = \sum_{t=1}^{n-K-1} q_{t,n} \boldsymbol{x}_{t} \boldsymbol{x}_{t}^{T} + \sum_{t=n-K}^{n} q_{t,n} \boldsymbol{x}_{t} \boldsymbol{x}_{t}^{T}$$
(2.1)

where the block of weights $\{q_{t,n}\}_{t=n-K}^{n}$ is reoptimized at each time n. To express (2.1) in matrix form, let X_n represent the observation matrix (the block of K + 1data vectors beginning at index n - K), $X_n \stackrel{\text{def}}{=} [x_{n-K} \ x_{n-K+1} \ \cdots \ x_n]$, and $q_{t,n}$ the vector containing the weights applied to the vectors in X_n at time t. In these terms, (2.1) is written

$$\boldsymbol{C}_{n} = \sum_{t=1}^{n-K-1} q_{t,n} \boldsymbol{x}_{t} \boldsymbol{x}_{t}^{T} + \boldsymbol{X}_{n} \boldsymbol{Q}_{n,n} \boldsymbol{X}_{n}^{T}$$
(2.2)

where $Q_{n,n} = \mathcal{D}(q_{n,n})$. Let $\lambda_n \stackrel{\text{def}}{=} q_{n,n} - q_{n,n-1}$ represent the adjustments to the weights applied to observation matrix X_n at time *n*, with corresponding diagonal matrix $\Lambda_n = \mathcal{D}(\lambda_n)$. The weight adjustments represent the difference between the *a priori* and newly computed weights, with the most recent "adjustment" being a modification to a zero (by definition) weight. The recursive expression (2.2) in terms of the weight adjustments becomes

$$\boldsymbol{C}_{\boldsymbol{n}} = \boldsymbol{C}_{\boldsymbol{n}-1} + \boldsymbol{X}_{\boldsymbol{n}} \boldsymbol{\Lambda}_{\boldsymbol{n}} \boldsymbol{X}_{\boldsymbol{n}}^{T}. \tag{2.3}$$

The (composite) weights at any time must be non-negative, $q_{n,n} = \lambda_n + q_{n,n-1} > 0$, in order to retain proper meaning. The block weight assignment strategy is comparable to a sliding window over the sequence of data vectors, where the weights assigned to a data vector vary with time n, but only vary during the time interval [n, n + K]. Accordingly, at time n, the time-varying weights are computed by (Figure 2.1)

$$q_{t,n} = \begin{cases} \sum_{i=t}^{t+K} \lambda_{t,i}, & 0 \le t < n-K \\ \sum_{i=t}^{n} \lambda_{t,i}, & n-K \le t \le n \\ 0, & n < t \end{cases}$$
(2.4)

with the constraint $q_{t,n} \ge 0$ for any t and n. The weight adjustments $\lambda_{t,n}$ are zero outside the time interval $[n - K \le t \le n]$.

This formulation allows the covariance matrix to be updated at each n with a set of weight adjustments acting upon the present and past K observation vectors. In the next section, we derive the general MW-OBE recursions, beginning with (2.3).



Figure 2.1. Weight assignments in the MW-OBE algorithm.

2.2 Recursions

As in other OBE algorithms, at each time t, the MW-OBE computes the *inverse* covariance matrix, $P_n^{-1} \stackrel{\text{def}}{=} C_n$ as part of its recursion [15]. In the present formulation, the matrix P_n is similarly obtained by applying the matrix-inversion lemma (see Appendix A.1) to the covariance, (2.3), yielding

$$P_{n} = P_{n-1} - P_{n-1} X_{n} H_{n}^{-1} \Lambda_{n} X_{n}^{T} P_{n-1}.$$
(2.5)

where $H_n \stackrel{\text{def}}{=} I + \Lambda_n G_{n|n-1}$ and $G_{n|n-1} \stackrel{\text{def}}{=} X_n^T P_{n-1} X_n$. The existence of the matrix $H_n^{-1} \Lambda_n = [\Lambda_n^{-1} + G_{n|n-1}]^{-1}$ is contingent upon all weight adjustments in the matrix Λ_n being non-zero and $[\Lambda_n^{-1} + G_{n|n-1}]^{-1}$ being invertible. In practice this constraint is adhered to by omitting data vectors corresponding to zero weights [see (2.3)]. Another constraint, the reason for which is not yet obvious, is that K be no larger than m. The general reason for this inequality is the necessity to invert $G_{n|n-1}$ in future developments.

As in conventional OBE algorithms, the MW-OBE algorithm recursively computes the ellipsoid center θ_n and the scalar κ_n . The ellipsoid center represents the parameter vector estimate at time n, and κ_n , a defining scalar for the ellipsoid [see (1.4)]. The equalities

$$\boldsymbol{c_n} = \boldsymbol{c_{n-1}} + \boldsymbol{X_n} \boldsymbol{\Lambda_n} \boldsymbol{y_n} \tag{2.6}$$

$$\boldsymbol{P}_{n-1}\boldsymbol{X}_n\boldsymbol{H}_n^{-1} = \boldsymbol{P}_n\boldsymbol{X}_n \tag{2.7}$$

are first noted from (1.7) and (2.5) in order to simplify further developments. The recursion for the ellipsoid center θ_n is derived in a manner similar to that used by Deller and Luk [13]. By employing (2.6) and (2.7) in (1.7), we obtain

$$\theta_{n} = P_{n}c_{n}$$

$$= P_{n}c_{n-1} + P_{n}X_{n}\Lambda_{n}y_{n}$$

$$= \theta_{n-1} - P_{n-1}H_{n}^{-1}\Lambda_{n}X_{n}^{T}\theta_{n-1} + P_{n}X_{n}^{T}\Lambda_{n}y_{n}$$

$$= \theta_{n-1} + P_{n}X_{n}\Lambda_{n}(y_{n} - X_{n}^{T}\theta_{n-1})$$

$$= \theta_{n-1} + P_{n}X_{n}\Lambda_{n}\varepsilon_{n|n-1}.$$
(2.8)

Recursion (2.8) is similar in form to its conventional OBE counterpart (reduces to the conventional recursion when K = 0) but updates the ellipsoid center as a function of K past data vectors. By substituting the recursion

$$\begin{aligned} \theta_n^T C_n \theta_n &= (\theta_{n-1} + P_n X_n \Lambda_n \varepsilon_{n|n-1})^T C_n (\theta_{n-1} + P_n X_n \Lambda_n \varepsilon_{n|n-1}) \\ &= \theta_{n-1}^T C_n \theta_{n-1} + \theta_{n-1}^T C_n P_n (X_n \Lambda_n \varepsilon_{n|n-1}) + (X_n \Lambda_n \varepsilon_{n|n-1})^T P_n C_n \theta_{n-1} \\ &+ (X_n \Lambda_n \varepsilon_{n|n-1})^T P_n C_n P_n (X_n \Lambda_n \varepsilon_{n|n-1}) \\ &= \theta_{n-1}^T C_n \theta_{n-1} + \theta_{n-1}^T (X_n \Lambda_n \varepsilon_{n|n-1}) + (X_n \Lambda_n \varepsilon_{n|n-1})^T \theta_{n-1} \\ &+ (X_n \Lambda_n \varepsilon_{n|n-1})^T P_n (X_n \Lambda_n \varepsilon_{n|n-1}) \\ &= \theta_{n-1}^T C_{n-1} \theta_{n-1} + (y_n - \varepsilon_{n|n-1})^T \Lambda_n (y_n - \varepsilon_{n|n-1}) \\ &+ 2(y_n - \varepsilon_{n|n-1})^T \Lambda_n \varepsilon_{n|n-1} + \varepsilon_{n|n-1}^T \Lambda_n G_{n|n} \Lambda_n \varepsilon_{n|n-1} \\ &= \theta_{n-1}^T C_{n-1} \theta_{n-1} + y_n^T \Lambda_n y_n - \varepsilon_{n|n-1}^T \Lambda_n \varepsilon_{n|n-1} \\ &= \theta_{n-1}^T C_{n-1} \theta_{n-1} + y_n^T \Lambda_n y_n - \varepsilon_{n|n-1}^T \left[\Lambda_n G_{n|n} \Lambda_n - \Lambda_n \right] \varepsilon_{n|n-1} \\ &= \theta_{n-1}^T C_{n-1} \theta_{n-1} + y_n^T \Lambda_n y_n - \varepsilon_{n|n-1}^T \left[\Lambda_n G_{n|n} \Lambda_n - \Lambda_n \right] \varepsilon_{n|n-1} \\ &= \theta_{n-1}^T C_{n-1} \theta_{n-1} + y_n^T \Lambda_n y_n - \varepsilon_{n|n-1}^T \left[\Lambda_n G_{n|n} \Lambda_n - \Lambda_n \right] \varepsilon_{n|n-1} \end{aligned}$$

into (1.6), we obtain

$$\kappa_{n} = \kappa_{n-1} - (\gamma_{n} - \boldsymbol{y}_{n})\Lambda_{n}(\gamma_{n} + \boldsymbol{y}_{n})^{T} + \boldsymbol{\theta}_{n}^{T}\boldsymbol{C}_{n}\boldsymbol{\theta}_{n} - \boldsymbol{\theta}_{n-1}^{T}\boldsymbol{C}_{n-1}\boldsymbol{\theta}_{n-1}$$
$$= \kappa_{n-1} + \gamma_{n}^{T}\Lambda_{n}\gamma_{n} - \boldsymbol{\varepsilon}_{n|n-1}^{T}\boldsymbol{H}_{n}^{-1}\Lambda_{n}\boldsymbol{\varepsilon}_{n|n-1}.$$
(2.9)

Expression (2.9) also yields the conventional SM-WRLS recursion for κ_n [13] when K = 0.

2.3 A posteriori error vector and energy matrices

The overall objective of any OBE algorithm is to define a set of solutions, or a single solution, that closely identifies the parameters of a system. This objective is achieved by seeking to reduce either the distance (in some sense) between the parameter estimate and the "true" parameter vector, or the ellipsoid size. The latter in effect brings the estimator to a closer neighborhood of the true parameter vector when $\overline{\Omega}_n$ becomes "small." The progress of OBE algorithms in achieving the given objective is observable in the effects of each recursion at time *n* on error vector $\boldsymbol{\varepsilon}_{n|n-1}$ and energy matrix $\boldsymbol{G}_{n|n-1}$. In the process of reducing the current ellipsoid size, OBE algorithms "re-map" the error (scalar or vector) to satisfy inequality (1.2). As a result, $\boldsymbol{\varepsilon}_{n|n}$ and $\boldsymbol{G}_{n|n}$ provide important insights into algorithm behavior. In this section we express the error vector and energy matrix in their *a posteriori* representations, to better illustrate MW-OBE behavior.

2.3.1 A posteriori error, $\varepsilon_{n|n}$

To satisfy the pointwise error bound constraint (1.2), the *a posteriori* error vector $\varepsilon_{n|n}$ must belong to the hyper-box $\{u \in \Re^K : |u(i)| \le \gamma_n\}$ (Fig. 2.2). The transformation which maps $\varepsilon_{n|n-1}$ to this hyper-box¹ at time *n* is found by expressing $\varepsilon_{n|n}$ in terms of $\varepsilon_{n|n-1}$ (2.8),

$$\boldsymbol{\varepsilon}_{n|n} = \boldsymbol{y}_n - \boldsymbol{X}_n^T \boldsymbol{\theta}_n$$

¹The hyper-box is defined with the inclusion of its boundaries.

$$= \boldsymbol{y}_{n} - \boldsymbol{X}_{n}^{T} \boldsymbol{\theta}_{n-1} - \boldsymbol{G}_{n|n} \boldsymbol{\Lambda}_{n} \boldsymbol{\varepsilon}_{n|n-1}$$

$$= \boldsymbol{\varepsilon}_{n|n-1} - \boldsymbol{G}_{n|n} \boldsymbol{\Lambda}_{n} \boldsymbol{\varepsilon}_{n|n-1}$$

$$= \left[\boldsymbol{I} - \boldsymbol{G}_{n|n} \boldsymbol{\Lambda}_{n} \right] \boldsymbol{\varepsilon}_{n|n-1}. \qquad (2.10)$$

By Lemma A.1 (see Appendix A.1), the inverse of matrix $H_n = I + \Lambda_n G_{n|n-1}$ becomes

$$H_n^{-1} = I - I\Lambda_n X_n^T \left(C_{n-1} + X_n I\Lambda_n X_n^T \right)^{-1} X_n$$

= $I - \Lambda_n G_{n|n},$ (2.11)

which, combined with (2.10) yields

$$\boldsymbol{\varepsilon}_{n|n} = \boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1}. \qquad (2.12)$$

From (2.12) we note that the mapping of error vector $\boldsymbol{\varepsilon}_{n|n-1}$ to its a posteriori image $\boldsymbol{\varepsilon}_{n|n}$ is achieved through the transformation \boldsymbol{H}_n^{-T} . The expression for the scalar κ_n takes a simpler form through the use of $\boldsymbol{\varepsilon}_{n|n}$; substituting (2.12) in (2.9), we have

$$\kappa_n = \kappa_{n-1} + \gamma_n^T \Lambda_n \gamma_n - \varepsilon_{n|n-1}^T \Lambda_n \varepsilon_{n|n}, \qquad (2.13)$$

or, using the Hadamard product notation (see Table 1.1),

$$\kappa_n = \kappa_{n-1} + \lambda_n^T \left(\boldsymbol{\gamma}_n \circ \boldsymbol{\gamma}_n - \boldsymbol{\varepsilon}_{n|n-1} \circ \boldsymbol{\varepsilon}_{n|n} \right).$$
 (2.14)

2.3.2 A posteriori weighted energy matrix, $G_{n|n}$

The weighted energy matrix (divided by κ_n) represents the composite projection of the data vectors in X_n on the current ellipsoid axes. The *a posteriori* matrix $G_{n|n}$ is found by substituting the right side of (2.5) into $X_n^T P_n X_n$, yielding

$$G_{n|n} = X_n^T P_n X_n$$

= $X_n^T P_{n-1} X_n - X_n^T P_{n-1} X_n H_n^{-1} \Lambda_n X_n^T P_{n-1} X_n$
= $G_{n|n-1} - G_{n|n-1} H_n^{-1} \Lambda_n G_{n|n-1}$



Figure 2.2. Rectangular error bound constraint (1.2) (dashed line) when K = 1 and m = 2. An error $\epsilon_{n|n-1}$ originally violating the bound condition in light of new observations is mapped to a location inside the rectangle $\{\boldsymbol{u} : |\boldsymbol{u}| \leq \gamma\}$ in its a posteriori $\epsilon_{n|n}$ form.

$$= H_n^{-T} G_{n|n-1}. (2.15)$$

(Since $G_{n|n}$ is symmetric, (2.15) is also equal to $G_{n|n-1}H_n^{-1}$.) The matrix H_n^{-1} may be expressed as

$$H_n^{-1} = (G_{n|n-1})^{-1} (G_{n|n}).$$
 (2.16)

Theorem 2.1 The matrices $G_{n|n-1}$, $G_{n|n}$ are positive definite.

Proof: Since P_{n-1} is positive definite, $a^T G_{n|n-1}a = (aX_n^T)^T P_{n-1}(aX_n^T) \ge 0$ for all $a \in \Re^K$, or $G_{n|n-1}$ is positive semi-definite [41]. The same is deduced for $G_{n|n}$. With both $G_{n|n-1}$, $G_{n|n}$ non-singular (no zero eigenvalues), we conclude that they are positive definite.

2.4 Ellipsoid volume

The volume of ellipsoid $\bar{\Omega}_n$ is proportional to the determinant of $\kappa_n P_n$ and therefore the ratio

$$\frac{\det(\kappa_n \boldsymbol{P}_n)}{\det(\kappa_{n-1} \boldsymbol{P}_{n-1})} = \left(\frac{\kappa_n}{\kappa_{n-1}}\right)^m \frac{\det(\boldsymbol{P}_n)}{\det(\boldsymbol{P}_{n-1})}$$
(2.17)

represents an appropriate measure for the change in volume [13]. The ratio of determinants in (2.17) reduces to

$$\frac{\det(P_n)}{\det(P_{n-1})} = I - X_n \left[\Lambda_n^{-1} + G_{n|n-1} \right]^{-1} X_n^T P_{n-1}
= \frac{\det\left[\Lambda_n^{-1} + G_{n|n-1} - G_{n|n-1} \right]}{\det(\Lambda_n^{-1} + G_{n|n-1})}
= \frac{1}{\det(I + \Lambda_n G_{n|n-1})}
= \det(H_n^{-1}) = 1/\det(H_n),$$
(2.18)

or,

$$\det (\boldsymbol{H}_n) = \frac{\det (\boldsymbol{P}_{n-1})}{\det (\boldsymbol{P}_n)}.$$
(2.19)

Incorporating (2.19) into the inverse of (2.17), we obtain

$$\frac{\det (\kappa_{n-1} \boldsymbol{P}_{n-1})}{\det (\kappa_n \boldsymbol{P}_n)} = \det (\boldsymbol{H}_n) \left(\frac{\kappa_{n-1}}{\kappa_n}\right)^m.$$
(2.20)

Results (2.19) and (2.20) will facilitate future developments.

2.5 Algorithm

MW-OBE algorithms are similar to other OBE algorithms in that the observation matrix is checked for innovation at each iteration. If the observation matrix is deemed useful, a new inverse covariance matrix and parameter vector estimate are computed. An advantage of MW-OBE over conventional OBE algorithms is the flexibility in selection and number of past weights to revisit. Indeed, MW-OBE does not specify which past weights are to be revisited, but limits their number to (m - 1), m being the system order, due to the necessity to invert the matrix $G_{n|n-1}$ (see Section 2.2 and Chapter 3). The simulations presented in Chapter 6 are generated using MW-OBE algorithms that revisit the past K non-zero weights. Updating the past sequentially numbered K weights often results in reconsideration of previously rejected observation vectors (zero weighted observation vectors) which may not offer any new information
represents an appropriate measure for the change in volume [13]. The ratio of determinants in (2.17) reduces to

$$\frac{\det(P_n)}{\det(P_{n-1})} = I - X_n \left[\Lambda_n^{-1} + G_{n|n-1} \right]^{-1} X_n^T P_{n-1}
= \frac{\det\left[\Lambda_n^{-1} + G_{n|n-1} - G_{n|n-1} \right]}{\det(\Lambda_n^{-1} + G_{n|n-1})}
= \frac{1}{\det(I + \Lambda_n G_{n|n-1})}
= \det(H_n^{-1}) = 1/\det(H_n),$$
(2.18)

or,

$$\det (\boldsymbol{H}_n) = \frac{\det (\boldsymbol{P}_{n-1})}{\det (\boldsymbol{P}_n)}.$$
(2.19)

Incorporating (2.19) into the inverse of (2.17), we obtain

$$\frac{\det (\kappa_{n-1} \boldsymbol{P}_{n-1})}{\det (\kappa_n \boldsymbol{P}_n)} = \det (\boldsymbol{H}_n) \left(\frac{\kappa_{n-1}}{\kappa_n}\right)^m.$$
(2.20)

Results (2.19) and (2.20) will facilitate future developments.

2.5 Algorithm

MW-OBE algorithms are similar to other OBE algorithms in that the observation matrix is checked for innovation at each iteration. If the observation matrix is deemed useful, a new inverse covariance matrix and parameter vector estimate are computed. An advantage of MW-OBE over conventional OBE algorithms is the flexibility in selection and number of past weights to revisit. Indeed, MW-OBE does not specify which past weights are to be revisited, but limits their number to (m - 1), m being the system order, due to the necessity to invert the matrix $G_{n|n-1}$ (see Section 2.2 and Chapter 3). The simulations presented in Chapter 6 are generated using MW-OBE algorithms that revisit the past K non-zero weights. Updating the past sequentially numbered K weights often results in reconsideration of previously rejected observation vectors (zero weighted observation vectors) which may not offer any new information I. Initialization: 1. $\boldsymbol{\theta}_{K} = \mathbf{0}, \, \kappa_{K} = 1 \text{ and } \boldsymbol{P}_{K} = \frac{1}{\mu} \boldsymbol{I}, \, \mu \text{ small.}$ 2. $q_{\kappa} = 0$. **II. Recursion:** For n = K + 1, K + 2, ...Form X_n matrix from present and chosen past K data vectors along with corresponding \boldsymbol{y}_n and $\boldsymbol{\gamma}_n$ vectors. $\boldsymbol{\varepsilon}_{n|n-1} = \boldsymbol{y}_n - \boldsymbol{X}_n^T \boldsymbol{\theta}_{n-1}$ $R_n = P_{n-1}X_n$ and $G_{n|n-1} = X_n^T R_n$ If current and past K observations are innovative, determine optimal weight vector, λ_n (optimization criterion dependent). Otherwise, next n. $\boldsymbol{K}_{n} = \boldsymbol{R}_{n} [\boldsymbol{\Lambda}_{n}^{-1} + \boldsymbol{G}_{n|n-1}]^{-1}$ $\boldsymbol{P}_{n} = \boldsymbol{P}_{n-1} - \boldsymbol{K}_{n} \boldsymbol{R}_{n}^{T}$ $\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + \boldsymbol{K}_n \boldsymbol{\varepsilon}_{n|n-1}$ $\kappa_n = \kappa_{n-1} + \gamma_n^T \Lambda_n \gamma_n - \varepsilon_{n|n-1}^T [\Lambda_n^{-1} + G_{n|n-1}]^{-1} \varepsilon_{n|n-1} \text{ (when necessary)}$ Next n.

on the system. This issue needs more study although empirically, observations taken in the past are more likely to be informative.

The MW-OBE algorithm is described in Table 2.1. Intermediate matrices R_n and K_n are introduced there to simplify the recursions.

2.6 Computational costs

The following discussion is restricted to the computational costs of the general form of the MW-OBE algorithm. These recursions, common to all MW-OBE algorithms, are only performed when a data matrix is deemed informative. MW-OBE algorithms compute the error vector $\varepsilon_{n|n-1}$ and the energy matrix $G_{n|n-1}$ in the process of checking for innovation in the observation matrix. Although these costs are included in the following considerations, costs due to particular optimization methods (actual innovation check and weight generation) are excluded.

Detailed per update cost is shown in Table 2.2. When $\tilde{K} = 1$ (K = 0), corresponding to the conventional OBE algorithm, this cost becomes $3/2(m^2 + 3m + 4)$,

comparable to that of the algorithm described in [15] at $\mathcal{O}(m^2)$. This $\mathcal{O}(m^2)$ performance is maintained for $m \gg K$. When $K \approx m$, the per-update cost increases to $\mathcal{O}(m^4)$.

In order to keep the cost performance of this algorithm to real-time (on-line) status, we limit this study to the re-visitation of one or two past weights. In so doing, the computational cost are kept within a "reasonable" range (see Table 2.3 and Figure 2.3). Note that with small values of such K, the inversion of matrix $[\Lambda_n^{-1} + G_{n|n-1}]^{-1}$ is relatively inexpensive. The expression for computational cost shown in Table 2.2 includes the computations of $\epsilon_{n|n-1}$ and $G_{n|n-1}$, which are used in the innovation check.



Figure 2.3. Computational costs (number of multiplications) of a single MW-OBE iteration for K = 0, 1, 2 and 3 for system orders m = 2, 4, 8, 12 and 16. K = 0 represents the conventional OBE algorithm case.

As pointed out in [10, 13, 15, 17], OBE algorithms update the parameter estimator by using a small percentage of observation vectors, often less than 5% (on simulated data with known bounds). MW-OBE algorithms use an even smaller number of observation vectors [30, 31] and thereby compensate for the higher per-update computational cost. This tradeoff makes MW-OBE algorithms attractive in applications where per-point convergence is the dominant concern.

Table 2.2. Detailed computational costs of MW-OBE algorithms per update. The computation of $\varepsilon_{n|n-1}$ and $G_{n|n-1}$ are included although they are usually considered part of the innovation check. For simplicity, the notation \tilde{K} is used to denote K + 1.

Expression	Dimension	No. of Multiplications
$\varepsilon_{n n-1}$ $R_n = P_n X_n$ $G_{n n-1}$ $K_n = R_n [\Lambda_n^{-1} + G_{n n-1}]^{-1}$ $P_n = P_{n-1} - K_n R_n^T$ $\theta_n = \theta_{n-1} + K_n \varepsilon_{n n-1}$ κ_n	$ \tilde{K} \times 1 $ $ m \times \tilde{K} $ $ \tilde{K} \times \tilde{K} $ $ m \times \tilde{K} $ $ m \times m $ $ \tilde{K} \times 1 $ $ 1 \times 1 $	$\begin{array}{c} m \tilde{K} \\ m^2 \tilde{K} \\ (m \tilde{K}/2) (\tilde{K}+1) \\ \tilde{K}^3 + m \tilde{K}^2 + \tilde{K} \\ (m \tilde{K}/2) (m+1) \\ m \tilde{K} \\ 3 \tilde{K} + \tilde{K}^2 \end{array}$
Total		$(3m ilde K/2)(m+ ilde K+2)+ ilde K(ilde K^2+ ilde K+4)$

Table 2.3. Computational costs of MW-OBE algorithms per update for K = 0, 1, 2 and 3 as functions of m.

		For typical m		
K	No. of Multiplications per Update	m = 10	m = 100	m = 1000
0	$(3/2)(m^2 + 3m + 4)$	201	15456	1504506
1	$3m^2 + 12m + 20$	440	31220	3012020
2	$3/2(3m^2+15m+32)$	723	47298	4522548
3	$6(m^2+6m+16)$	1056	63696	6036096

The computational costs in Table 2.2 exploit symmetries in the computations. Additional reduction in cost is achieved by noting that expensive recursion of P_n may be evaluated in a total of $\frac{\tilde{K}}{2}(\frac{\tilde{K}^2}{3} + \tilde{K}m + m^2)$ [23, 28] multiplications using LUdecomposition (Choleski factorization) of $H_n^{-1}\Lambda_n(\frac{\tilde{K}^3}{6})$. This technique also simplifies the computation of κ_n .

2.7 Optimizations

The matrix defining the hyperellipsoid at time *n* is given by [see (1.4)] P_n^{-1}/κ_n . The measure det $(\kappa_n P_n)$ is proportional to the square of the volume of the ellipsoid and is most often minimized in the OBE optimization process. Minimization of the trace of $\{\kappa_n P_n\}$ is also a meaningful measure of size (e.g., [14, 21]). Minimization of the parameter κ_n , first suggested in [8], had been controversial with respect to its interpretability [14] until recently when Huang's research group developed the QOBE algorithm [22, 39]. QOBE, which minimizes κ_n in conjunction with a specific weighting strategy, provides interesting interpretations of this optimization process. Because of the relative algebraic simplicity of the algorithm, we use the QOBE-like approach of κ_n minimization to develop a specific instance of the MW-OBE algorithm, MW-QOBE. MW-QOBE is presented in Chapter 3. On the other hand, the SM-WRLS OBE algorithm optimizes the volume of the ellipsoid at each step. This is used to develop a second MW-OBE algorithm, MW-SM-WRLS. In Chapter 4, the MW-SM-WRLS is presented with simulations.

Chapter 3

Multiple Weight Quasi-OBE Algorithm

3.1 Introduction

In this chapter we develop the MW-QOBE, a MW-OBE algorithm based on minimizing the scalar κ_n with respect to the present weight and past K weights (the diagonal elements of Λ_n) if the new data admit further reduction. We compare this novel approach to the QOBE algorithm developed by Huang *et al.* [22]. We prove the uniqueness of the optimal solution for general $K \leq m$, and experimentally study the case of K = 1.

3.2 κ_n minimization

Recent study of the QOBE algorithm has shown the merit of minimizing the scalar κ_n [22, 39]. This simple yet efficient algorithm offers good convergence of the parameter estimator to the true parameter vector [11]. When the prediction error $\varepsilon_{n|n-1}$ generated by the current parameter estimator θ_{n-1} and observation vector x_n falls outside the error bound constraint (1.2), the QOBE generates a new parameter estimator θ_n which re-maps the prediction error to (exactly) the bound ($|\varepsilon_{n|n}| = \gamma_n$). Although developed from an "OBE" point of view, the condition for data acceptance ($|\varepsilon_{n|n-1}| < \gamma_n$) and the process of mapping $\varepsilon_{n|n-1}$ into $\varepsilon_{n|n}$ is found to be "decoupled" from the ellipsoid, a departure from other OBE algorithms (and reason for the name

"QOBE"). This independence from the ellipsoid makes QOBE particularly interesting in time-varying applications due to its robustness to a "true" parameter θ_{\star} moving outside the ellipsoid. In this section we develop a specific MW-OBE algorithm by using as the optimization criterion the minimization of κ_n with respect to the weight vector λ_n .

In the following theorem, we derive the optimal weight vector λ_n at time *n* which minimizes κ_n over the present and past K weight adjustments.

Theorem 3.1 The scalar κ_n is minimized by the weight adjustments

$$\lambda_n = (G_{n|n-1}S_n\Lambda_n)^{-1}(\varepsilon_{n|n-1}-S_n\gamma_n)$$

where S_n is a diagonal matrix with diagonal elements ± 1 .

Proof: In minimizing κ_n (2.9), we encounter the term $\varepsilon_{n|n-1}^T H_n^{-1} \Lambda_n \varepsilon_{n|n-1}$, a quadratic expression in $\varepsilon_{n|n-1}$ which involves the inverse matrix $H_n^{-1} \Lambda_n = [\Lambda_n^{-1} + G_{n|n-1}]^{-1}$. We use the notation of Table 1.1 in writing to $\lambda_n(i)$ to mean the *i*th element of the vector λ_n at time n. $\lambda_n(i)$ is equivalent to $\Lambda_n(i,i)$ since $\Lambda_n = \mathcal{D}(\lambda_n)$. Differentiating the term $H_n^{-1} \Lambda_n$ with respect to an arbitrary (scalar) weight $\lambda_n(i)$, we obtain

$$\frac{\partial (\boldsymbol{H}_{n}^{-1}\boldsymbol{\Lambda}_{n})}{\partial \boldsymbol{\lambda}_{n}(i)} = \frac{\partial \boldsymbol{H}_{n}^{-1}}{\partial \boldsymbol{\lambda}_{n}(i)}\boldsymbol{\Lambda}_{n} + \boldsymbol{H}_{n}^{-1}\frac{\partial \boldsymbol{\Lambda}_{n}}{\partial \boldsymbol{\lambda}_{n}(i)} \\
= -\boldsymbol{H}_{n}^{-1}\frac{\partial \boldsymbol{\Lambda}_{n}}{\partial \boldsymbol{\lambda}_{n}(i)}\boldsymbol{G}_{n|n-1}\boldsymbol{H}_{n}^{-1}\boldsymbol{\Lambda}_{n} + \boldsymbol{H}_{n}^{-1}\frac{\partial \boldsymbol{\Lambda}_{n}}{\partial \boldsymbol{\lambda}_{n}(i)} \\
= \boldsymbol{H}_{n}^{-1}\frac{\partial \boldsymbol{\Lambda}_{n}}{\partial \boldsymbol{\lambda}_{n}(i)}\left(\boldsymbol{I} - \boldsymbol{G}_{n|n-1}\boldsymbol{H}_{n}^{-1}\boldsymbol{\Lambda}_{n}\right) \\
= \boldsymbol{H}_{n}^{-1}\frac{\partial \boldsymbol{\Lambda}_{n}}{\partial \boldsymbol{\lambda}_{n}(i)}\left(\boldsymbol{H}_{n}^{-1}\right)^{T}.$$
(3.1)

From (3.1) we conclude that

$$\boldsymbol{\varepsilon}_{n|n-1}^{T} \frac{\partial \left(\boldsymbol{H}_{n}^{-1} \boldsymbol{\Lambda}_{n}\right)}{\partial \boldsymbol{\lambda}_{n}(i)} \boldsymbol{\varepsilon}_{n|n-1} = \boldsymbol{\varepsilon}_{n|n-1}^{T} \boldsymbol{H}_{n}^{-1}(*,i) \left[\boldsymbol{H}_{n}^{-1}(*,i)\right]^{T} \boldsymbol{\varepsilon}_{n|n-1}, \qquad (3.2)$$

where the column vector $H_n^{-1}(*,i)$ (see Table 1.1) is the *i*th column of matrix H_n^{-1} .

By incorporating (3.2) in the differentiation of (2.9) with respect to $\lambda_n(i)$ we obtain

$$\frac{\partial \kappa_{n}}{\partial \lambda_{n}(i)} = (\gamma_{n}(i))^{2} - \varepsilon_{n|n-1}^{T} \boldsymbol{H}_{n}^{-1}(*,i) \left[\boldsymbol{H}_{n}^{-1}(*,i) \right]^{T} \varepsilon_{n|n-1}$$
$$= \left(\gamma_{n}(i) + \varepsilon_{n|n-1}^{T} \boldsymbol{H}_{n}^{-1}(*,i) \right) \left(\gamma_{n}(i) - \varepsilon_{n|n-1}^{T} \boldsymbol{H}_{n}^{-1}(*,i) \right). \quad (3.3)$$

Hence,

$$\frac{\partial \kappa_n}{\partial \Lambda_n} = \Gamma_n^2 - \left[\mathcal{D}(\boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1}) \right]^2, \qquad (3.4)$$

where $\frac{\partial \kappa_n}{\partial \Lambda_n}$ is the diagonal matrix whose *i*th diagonal element is $\frac{\partial \kappa_n}{\partial \lambda_n(i)}$.

Using the Hadamard product notation (see Table 1.1), and recalling that $\Lambda_n = \mathcal{D}(\lambda_n)$, (3.4) is expressed as

$$\frac{\partial \kappa_n}{\partial \lambda_n} = \left(\gamma_n + H_n^{-T} \varepsilon_{n|n-1} \right) \circ \left(\gamma_n - H_n^{-T} \varepsilon_{n|n-1} \right).$$
(3.5)

The optimal diagonal weight matrix Λ_n (or vector λ_n) is the solution of the equation

$$\frac{\partial \kappa_n}{\partial \lambda_n} = \mathbf{0}. \tag{3.6}$$

Using $|\mathbf{A}|$ to denote matrix with (i, j) element $|\mathbf{A}(i, j)|$, (3.5) and (3.6) imply that

$$|\boldsymbol{H}_{\boldsymbol{n}}^{-T}\boldsymbol{\varepsilon}_{\boldsymbol{n}|\boldsymbol{n}-1}| - \boldsymbol{\gamma}_{\boldsymbol{n}} = \boldsymbol{0}. \tag{3.7}$$

To solve (3.7), we define the column vector of signs, ± 1 , $s_n \stackrel{\text{def}}{=} \text{sign}\{H_n^{-T} \varepsilon_{n|n-1}\}$ (see Table 1.1). By incorporating

$$|\boldsymbol{H}_{\boldsymbol{n}}^{-T}\boldsymbol{\varepsilon}_{\boldsymbol{n}|\boldsymbol{n}-1}| = (\boldsymbol{H}_{\boldsymbol{n}}^{-T}\boldsymbol{\varepsilon}_{\boldsymbol{n}|\boldsymbol{n}-1}) \circ \boldsymbol{s}_{\boldsymbol{n}}$$
(3.8)

into (3.7), and multiplying on the left by H_n^T , the optimal weight vector is found to be the solution in λ_n of

$$\boldsymbol{\varepsilon}_{n|n-1} = \boldsymbol{H}_{n}^{T}(\boldsymbol{\gamma}_{n} \circ \boldsymbol{s}_{n})$$

$$= (\boldsymbol{\gamma}_{n} \circ \boldsymbol{s}_{n}) + \boldsymbol{G}_{n|n-1}\boldsymbol{\Lambda}_{n}(\boldsymbol{\gamma}_{n} \circ \boldsymbol{s}_{n}).$$

$$(3.9)$$

After some manipulation, we obtain

$$\boldsymbol{\lambda}_{n} = \left[\boldsymbol{G}_{n|n-1}^{-1} \left(\boldsymbol{\varepsilon}_{n|n-1} - \boldsymbol{\gamma}_{n} \circ \boldsymbol{s}_{n} \right) \right] \circ \left(\boldsymbol{g}_{n} \circ \boldsymbol{s}_{n} \right)$$
(3.10)

where g_n denotes the column vector whose elements are the diagonal elements of Γ_n^{-1} . If we further define the matrix $S_n \stackrel{\text{def}}{=} \mathcal{D}(s_n)$, then (3.10) can be also expressed as

$$\lambda_n = (G_{n|n-1}S_n\Gamma_n)^{-1}(\varepsilon_{n|n-1}-S_n\gamma_n). \qquad (3.11)$$

We verify that the solution point is a minimum by demonstrating a positive determinant of the Hessian, the matrix whose (i, j) element is

$$\frac{\partial^2 \kappa_n}{\partial \lambda_n(i) \,\partial \lambda_n(j)}.\tag{3.12}$$

Using the differentiation vector notation of (3.5), the Hessian (3.12) is alternatively represented by the (symmetric) matrix whose *i*th column is

$$\frac{\partial}{\partial \lambda_n(i)} \left(\frac{\partial \kappa_n}{\partial \lambda_n} \right). \tag{3.13}$$

The derivative of (3.5) with respect to a scalar weight $\lambda_n(i)$ is expressed as

$$\frac{\partial}{\partial \lambda_n(i)} \left(\frac{\partial \kappa_n}{\partial \lambda_n} \right) = -2 \left(\boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1} \right) \circ \left[\frac{\partial}{\partial \lambda_n(i)} \left(\boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1} \right) \right]$$
(3.14)

which, by incorporating (3.1), becomes

$$\frac{\partial}{\partial \lambda_n(i)} \left(\frac{\partial \kappa_n}{\partial \lambda_n} \right) = -2 \left(\boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1} \right) \circ \left[-\boldsymbol{H}_n^{-T} \frac{\partial \boldsymbol{H}_n^{T}}{\partial \lambda_n(i)} \boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1} \right]. \quad (3.15)$$

Evaluating (3.15) at any root, say λ_n^* , of the first derivative (3.5) and by substituting $\mathbf{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1} = S_n \boldsymbol{\gamma}_n$ into (3.14), we obtain the columns of the Hessian matrix

$$\frac{\partial}{\partial \lambda_n(i)} \left(\frac{\partial \kappa_n}{\partial \lambda_n} \right)_{\lambda_n^*} = 2S_n \gamma_n \circ H_n^{-T} \frac{\partial H_n^T}{\partial \lambda_n(i)} S_n \gamma_n$$
$$= 2S_n \gamma_n \circ H_n^{-T} G_{n|n-1} \frac{\partial \Lambda_n}{\partial \lambda_n(i)} S_n \gamma_n$$

$$= 2S_n\gamma_n \circ H_n^{-T}G_{n|n-1}\Gamma_n S_n \frac{\partial \lambda_n}{\partial \lambda_n(i)}$$
$$= 2S_n\gamma_n \circ G_{n|n-1}H_n^{-1}\frac{\partial \Lambda_n}{\partial \lambda_n(i)}S_n\gamma_n.$$

The determinant of the Hessian is then

$$\det \left[\frac{\partial^2 \kappa_n}{\partial \lambda_n(1) \partial \lambda_n}, \cdots, \frac{\partial^2 \kappa_n}{\partial \lambda_n(K+1) \partial \lambda_n} \right] = 2^{K+1} \det \left(S_n \Gamma_n \right) \det \left(G_{n|n-1} H_n^{-1} S_n \Gamma_n \right)$$
$$= \frac{2^{K+1}}{\det \left(H_n \right)} \det^2 \left(S_n \Gamma_n \right) \det \left(G_{n|n-1} \right).$$
(3.16)

A necessary and sufficient condition for a non-negative Hessian determinant (3.16) is det $(H_n) \ge 0$. This condition is satisfied with valid weight adjustments, made evident by expressing H_n in its *a posteriori* representation (2.16).

Remark: It is noteworthy that setting K = 0 reduces this expression to the optimal scalar weight in the QOBE algorithm [22, 39].

Corollary 3.1 Optimizing κ_n over several weights at each n results in a nonincreasing sequence $\{\kappa_n\}$.

Proof: By substituting (3.7) into (2.9) we obtain

$$\kappa_{n} = \kappa_{n-1} + \gamma_{n}^{T} \Lambda_{n} (\gamma_{n} - S_{n} \varepsilon_{n|n-1})$$

$$= \kappa_{n-1} - \gamma_{n}^{T} \Lambda_{n} S_{n} (\varepsilon_{n|n-1} - S_{n} \gamma_{n})$$

$$= \kappa_{n-1} - \lambda_{n}^{T} \Gamma_{n} S_{n} (\varepsilon_{n|n-1} - S_{n} \gamma_{n})$$

$$= \kappa_{n-1} - (\varepsilon_{n|n-1} - S_{n} \gamma_{n})^{T} G_{n|n-1}^{-1} (\varepsilon_{n|n-1} - S_{n} \gamma_{n})$$
(3.17)
$$(3.17)$$

where $G_{n|n-1}^{-1}$ is positive definite. Therefore, κ_n is a non-increasing function when **evaluated** at λ_n of (3.11).

Equation (3.18) provides an algebraic proof of Corollary 3.1 along with the amount of decrease in κ_n at each step. To simply prove that $\kappa_n \leq \kappa_{n-1}$ [using weights (3.11)], we note that $\kappa_n|_{\lambda_n=0} = \kappa_{n-1}$.

3.3 Existence and uniqueness of an optimal solution

A major benefit of OBE algorithms is the avoidance of computationally costly recursions by a simple redundancy check that detects when observations supply no new information. Indeed, by pre-determining the absence of an optimal solution through a computationally inexpensive test, an OBE algorithm avoids laborious computations that ultimately yield zero or negative weights (indicating no optimal solution). The QOBE algorithm is particularly attractive due to very its simple test for innovation, notably $|\varepsilon_{n|n-1}| > \gamma_n$.

In the previous section, the MW-QOBE algorithm optimal weights were derived as functions of a sign vector (or equivalent diagonal matrix) s_n (at time n). This $(K + 1) \times 1$ sign vector is formed from a set of 2^{K+1} possible permutations of ± 1 elements, with the constraint (see Section 2.1) that the (composite) weights at time n be element-wise positive,

$$\boldsymbol{q}_{n,n} = \boldsymbol{\lambda}_n + \boldsymbol{q}_{n,n-1} > \boldsymbol{0}. \tag{3.19}$$

An efficient test for new information in the MW-OBE observation matrix presents new challenges. Although a sufficient test for the presence of an optimal solution is not yet found, we report a necessary condition in Theorem 3.1.

The following lemma represents the principal algebraic result needed to prove a condition for existence of the weight adjustment vector λ_n .

Lemma 3.1 Let A represent an $n \times n$ symmetric positive definite matrix and let uand v be two $n \times 1$ vectors with v(i) > 0, for all $i \in [1, n]$. There exists a sign vector s with corresponding diagonal matrix $S = \mathcal{D}(s)$ (see Table 1.1) which satisfies the vector inequality SA(u - Sv) > 0, only if

$$\boldsymbol{v}^T \boldsymbol{D}^2 \left(\boldsymbol{v} - |\boldsymbol{u}| \right) \leq 0$$

where D is a diagonal matrix of eigenvalues of A.

Proof: We multiply the vector inequality SAu > SASv by the vector v > 0 and

obtain the scalar inequality

$$\boldsymbol{v}^T \boldsymbol{S} \boldsymbol{A} \boldsymbol{u} > \boldsymbol{v}^T \boldsymbol{S} \boldsymbol{A} \boldsymbol{S} \boldsymbol{v} > 0 \tag{3.20}$$

which can be re-written

$$s^T V A u > s^T V A V s$$
 or (3.21)

$$\boldsymbol{z}^T \boldsymbol{w} > \boldsymbol{z}^T \boldsymbol{z} \tag{3.22}$$

where $V = \mathcal{D}(v)$, $z \stackrel{\text{def}}{=} RDVs$, $w \stackrel{\text{def}}{=} RDu$, $A = R^T D^2 R$ with D a diagonal matrix comprised of the positive square roots of the eigenvalues of A (all positive since A is positive definite) and R unitary. The functions $f_1(z) \stackrel{\text{def}}{=} z^T w$ and $f_2(z) \stackrel{\text{def}}{=} z^T z$ represent a hyperplane and a quadratic function in z, respectively. The inequality (3.22) is only achieved in the interior of the spheroid represented by

$$\boldsymbol{z}^T \boldsymbol{z} - \boldsymbol{z}^T \boldsymbol{w} = \boldsymbol{0}. \tag{3.23}$$

In order to express (3.23) in the form

$$\|\boldsymbol{z}-\boldsymbol{z}_c\|^2-\alpha=0,$$

where z_c represents the center of the spheroid and α is a scalar, we choose

$$\boldsymbol{z}_{c}=rac{1}{2}\boldsymbol{w} ext{ and } \boldsymbol{\alpha}=rac{1}{4}\|\boldsymbol{w}\|^{2}.$$

The spheroid equation then becomes

$$\left(\frac{4}{\|\boldsymbol{w}\|^2}\right)\|\boldsymbol{z} - \frac{1}{2}\boldsymbol{w}\|^2 = 1, \text{ or } \|\boldsymbol{z} - \frac{1}{2}\boldsymbol{w}\|^2 = \|\frac{1}{2}\boldsymbol{w}\|^2.$$
(3.24)

Substituting the expressions for z and w, and recalling that R is unitary, (3.24) implies that

$$\|RD(2Vs - u)\|^2 = \|RDu\|^2$$

 $\|D(2Vs - u)\|^2 = \|Du\|^2$ or

$$\boldsymbol{v}^T \boldsymbol{D}^2 \left(\boldsymbol{v} - \boldsymbol{S} \boldsymbol{u} \right) = 0.$$

Any feasible vector \boldsymbol{u} therefore satisfies

$$\{\boldsymbol{u}: \boldsymbol{v}^T \boldsymbol{D}^2 (\boldsymbol{v} - \boldsymbol{S} \boldsymbol{u}) \leq 0\}$$
(3.25)

where the vector s is one of 2^{K+1} possible sign vectors. The smallest value on the left side of (3.25) occurs whenever s = sign(u). Therefore any u consistent with (3.25) must satisfy

$$\boldsymbol{v}^T \boldsymbol{D}^2 \left(\boldsymbol{v} - |\boldsymbol{u}| \right) \leq 0. \tag{3.26}$$

Hence the existence of a solution requires that there be a vector \boldsymbol{u} satisfying (3.26).

Theorem 3.2 (Necessity) There exists a weight adjustment vector λ_n (3.11) satisfying (3.19) only if

$$\boldsymbol{\gamma}_{\boldsymbol{n}}^{T} \boldsymbol{D}_{\boldsymbol{n}} (\boldsymbol{\gamma}_{\boldsymbol{n}} - |\boldsymbol{\varepsilon}_{n|n-1}|) \leq 0 \tag{3.27}$$

where D_n is the diagonal matrix of eigenvalues of $G_{n|n-1}$.

Proof: We may assume without loss of generality that $\lambda_n > 0$ as demonstrated in Chapter 5. Let $\mathbf{A} = \mathbf{G}_{n|n-1}^{-1}$, $\mathbf{v}_n = \boldsymbol{\varepsilon}_{n|n-1}$ and $\mathbf{u}_n = \boldsymbol{\gamma}_n$. Apply Lemma 3.1 to prove the existence of an appropriate sign vector, hence, a weight adjustments vector.

Next we prove that at most one weight adjustment vector λ satisfies (3.11) with constraint (3.19) beginning with the following lemma.

Lemma 3.2 Let A represent an $n \times n$ symmetric positive definite matrix and let **u** and **v** be two $n \times 1$ vectors with v(i) > 0, for all $i \in [1, n]$. There exists at most one sign vector with corresponding diagonal matrix S = D(s) such that

$$SA(u - Sv) > 0. \tag{3.28}$$

Proof: Let $s_1 \neq s_2$ be sign vectors with corresponding diagonal matrices S_1 and S_2 both satisfying inequality (3.28). Without loss of generality, we assume that the mismatched elements of S_1 and S_2 are consecutively arranged in the top left quadrants as S_{1a} and S_{2a} , since we may re-order the basis and preserve the positive definitiveness of A. The partitioned matrices (including A appropriately partitioned) are

$$\boldsymbol{S}_{1} = \begin{bmatrix} \boldsymbol{S}_{1a} & \boldsymbol{0} \\ \hline \boldsymbol{0} & \boldsymbol{S}_{1b} \end{bmatrix}, \quad \boldsymbol{S}_{2} = \begin{bmatrix} -\boldsymbol{S}_{1a} & \boldsymbol{0} \\ \hline \boldsymbol{0} & \boldsymbol{S}_{1b} \end{bmatrix}, \quad \boldsymbol{A} = \begin{bmatrix} \boldsymbol{A}_{a} & \boldsymbol{A}_{c} \\ \hline \boldsymbol{A}_{c}^{T} & \boldsymbol{A}_{b} \end{bmatrix}. \quad (3.29)$$

٢

We now add the inequalities corresponding to S_1 and S_2 ,

$$0 < (S_1 + S_2)Au - (S_1AS_1 + S_2AS_2)v$$

and incorporate (3.29), to obtain

$$\mathbf{0} < \left[\frac{0}{0} \mid 0 \\ 0 \mid 2S_{1b} \right] - 2 \left[\frac{S_{1a}A_aS_{1a}}{0} \mid 0 \\ \frac{S_{1b}A_bS_{1b}}{0} \right] \left[\frac{v_a}{v_b} \right]$$
$$\mathbf{0} < \left[\frac{-2S_{1a}A_aS_{1a}v_a}{2(S_{1b} - S_{1b}A_bS_{1b})v_b} \right]$$
(3.30)

where $[v_a | v_b]^T$ is an appropriately partitioned vector v. Multiplying each side of the top partition inequality by the vector $v_a > 0$ maintains the inequality, therefore

$$0 < -\boldsymbol{v}_a^T \boldsymbol{S}_{1a} \boldsymbol{A}_a \boldsymbol{S}_{1a} \boldsymbol{v}_a,$$

a contradiction since A_a is positive definite. Hence a sign vector s satisfying (3.28) is unique.

Theorem 3.3 (Uniqueness) At most one weight adjustment vector λ_n solving (3.11) also satisfies (3.19).

Proof: Define A, u_n and v_n as in the proof Theorem 3.2. Apply Lemma 3.2.

3.4 Incremental gain

The merit of MW-QOBE with respect to QOBE is its ability to further reduce κ_n and the ellipsoid volume at each update. In this section we explore the incremental gains in such improvements.

Although "valid" weights are required to be non-zero for the non-singularity of $H_n^{-1}\Lambda_n$ (see Section 2.2), in practice a zero weight may be replaced by a small number². This substitution allows us to prove the following theorem.

Theorem 3.4 If valid optimal weights $\lambda_{n,K}^*$ and $\lambda_{n,K-1}^*$ exist at time n for the optimization of $\kappa_{n,K}$ and $\kappa_{n,K-1}$, then $\kappa_{n,K}(\lambda_{n,K}^*) < \kappa_{n,K-1}(\lambda_{n,K-1}^*)$.

Heuristic proof: We can always achieve $\kappa_{n,K} = \kappa_{n,K-1}$ by taking $\lambda_{n,K} = [0 \mid \lambda_{n,K-1}(2:K)] \approx [\epsilon \mid \lambda_{n,K-1}(2:K)]$ where ϵ is a small number. If this solution produces the smallest possible $\kappa_{n,K}$, then the new (Kth) observation provides no innovation and therefore $\kappa_{n,K}(\lambda_{n,K}^*) = \kappa_{n,K-1}(\lambda_{n,K-1}^*)$. Otherwise $\kappa_{n,K}(\lambda_{n,K}^*) < \kappa_{n,K-1}(\lambda_{n,K-1}^*)$.

Proof with exact incremental gain: The energy matrix at time n when optimizing over K weights may be partitioned in the following manner:

$$\boldsymbol{G}_{n|n-1,K} = \begin{bmatrix} g_{n,K} & \boldsymbol{g}_{n,K}^T \\ \boldsymbol{g}_{n,K} & \boldsymbol{G}_{n|n-1,K-1} \end{bmatrix}$$
(3.31)

with $g_{n,K} \stackrel{\text{def}}{=} x_{n-K}^T P_{n-1} x_{n-K}$, $g_{n,K}^T \stackrel{\text{def}}{=} x_{n-K}^T P_{n-1} X_{n,K-1}$, $X_{n,K-1}$. We define the scalar $\Delta_{n,K} \stackrel{\text{def}}{=} g_{n,K} - g_{n,K}^T G_{n|n-1,K-1}^{-1} g_{n,K}$. Since $G_{n,K}$ is positive definite, $\Delta_{n,K}^{-1} = G_{n,K}^{-1}(1,1)$ is positive. Let us recall (3.18),

$$\kappa_n = \kappa_{n-1} - (\varepsilon_{n|n-1} - S_n \gamma_n)^T G_{n|n-1}^{-1} (\varepsilon_{n|n-1} - S_n \gamma_n)$$

and define $\boldsymbol{u}_{n,K} = \boldsymbol{\varepsilon}_{n|n-1,K} - \boldsymbol{S}_{n,K} \boldsymbol{\gamma}_{n,K}$. Then, by Lemma A.5, we rewrite $\kappa_{n,K}$ as

$$\kappa_{n,K} = \kappa_{n,K-1} - \Delta_{n,K}^{-1} \left(u_{n,K}(1) G_{n,K-1}^{-1} X_{n,K-1}^T P_{n-1} x_{n,K-1} - u_{n,K-1} \right)^2$$

²Necessary for the existence of Λ_n^{-1}

and conclude that $\kappa_{n,K} < \kappa_{n,K-1}$.

Theorem 3.5 The sequence of ellipsoid volumes (over the times of updates) in MW-OBE algorithms is decreasing.

Proof: Deferred until Chapter 5.

3.5 Geometric interpretation of the κ_n minimization

The geometric interpretation of the MW-QOBE algorithm provides considerable insight into its behavior. The QOBE algorithm (MW-QOBE with K = 0) maps the absolute value of the (scalar) *a posteriori* error, $|\varepsilon_{n|n-1}|$, to the bound γ_n [11, 39]. We observe a similar behavior in the MW-QOBE algorithm by incorporating (3.7) into (2.12) to obtain

$$\boldsymbol{\gamma}_n \circ \boldsymbol{\gamma}_n - \boldsymbol{\varepsilon}_{n|n} \circ \boldsymbol{\varepsilon}_{n|n} = \mathbf{0}. \tag{3.32}$$

This reveals that the MW-QOBE algorithm maps the component-wise absolute value of a posteriori error vector $\boldsymbol{\varepsilon}_{n|n}$ to the error vector bound $\boldsymbol{\gamma}_n$ by requiring

$$\boldsymbol{\gamma}_n = |\boldsymbol{\varepsilon}_{n|n}|. \tag{3.33}$$

This phenomenon is illustrated in Figure 3.1 for the case K = 1. At each iteration, the MW-QOBE algorithm attempts to map the error vector $\varepsilon_{n|n-1}$ to the unique $S_n \gamma_n$ vector (one of 2^{K+1}) which satisfies condition (3.19) through the transformation (3.7). This condition imposes a more stringent requirement on the acceptance of observation vectors and as a result provides a more selective screening process.

3.6 Algorithm

The MW-QOBE algorithm appears in Table 3.1. The initial conditions mirror those in QOBE supplemented by the initial "accumulated" weight vector $q_{n,n-1}$ which is



Figure 3.1. Error bound constraint (1.2) when K = 1 and m = 2. An error $\varepsilon_{n|n-1}$ originally violating the bound condition is mapped to a location inside the rectangle $\{\boldsymbol{u} : |\boldsymbol{u}| \leq \gamma\}$ in its a posteriori $\varepsilon_{n|n}$ form.

appropriately assumed to contain all zero elements. This vector contains the accumulation of all adjustments made to the weights over the window of time that it currently represents, n - K, ..., n. At each iteration, the elements of $q_{n,n-1}$ are "shifted" to reflect the new time window. The check for innovation at time n in the conventional QOBE algorithm (K = 0) is

$$|\boldsymbol{\varepsilon}_{n|n-1}| > \boldsymbol{\gamma}_n. \tag{3.34}$$

With weight reoptimization (K > 0), satisfaction of this simple test is still necessary for any further computation to be required on the window $n - K, \ldots, n$. Indeed, if there is no innovation in the observation at time n, then the past K weights are already optimal. The use of the QOBE check for innovation (3.34) also becomes the basis for a simple adaptive (in K) version of MW-QOBE algorithm. When an optimal weight adjustment vector for K past weights cannot be found, the algorithm may opt to use the optimal weight for K = 0. This adaptive process guarantees further decrease in κ_n (per update, see Section 3.4) and allows use of QOBE algorithm convergence results [11, 39]. The computational cost of the method is drastically reduced by recourse to the simple check (3.34) prior to further optimization.



Figure 3.1. Error bound constraint (1.2) when K = 1 and m = 2. An error $\varepsilon_{n|n-1}$ originally violating the bound condition is mapped to a location inside the rectangle $\{\boldsymbol{u} : |\boldsymbol{u}| \leq \gamma\}$ in its a posteriori $\varepsilon_{n|n}$ form.

appropriately assumed to contain all zero elements. This vector contains the accumulation of all adjustments made to the weights over the window of time that it currently represents, n - K, ..., n. At each iteration, the elements of $q_{n,n-1}$ are "shifted" to reflect the new time window. The check for innovation at time n in the conventional QOBE algorithm (K = 0) is

$$|\boldsymbol{\varepsilon}_{n|n-1}| > \boldsymbol{\gamma}_n. \tag{3.34}$$

With weight reoptimization (K > 0), satisfaction of this simple test is still necessary for any further computation to be required on the window $n - K, \ldots, n$. Indeed, if there is no innovation in the observation at time n, then the past K weights are already optimal. The use of the QOBE check for innovation (3.34) also becomes the basis for a simple adaptive (in K) version of MW-QOBE algorithm. When an optimal weight adjustment vector for K past weights cannot be found, the algorithm may opt to use the optimal weight for K = 0. This adaptive process guarantees further decrease in κ_n (per update, see Section 3.4) and allows use of QOBE algorithm convergence results [11, 39]. The computational cost of the method is drastically reduced by recourse to the simple check (3.34) prior to further optimization.

Table 3.1. The MW-QOBE algorithm (non-zero past weights).

I. Initialization: 1. $\boldsymbol{\theta}_{K} = \mathbf{0}, \, \kappa_{K} = 1 \text{ and } \boldsymbol{P}_{K} = \frac{1}{\mu} \boldsymbol{I}, \, \mu \text{ small.}$ 2. $q_{K} = 0$. **II. Recursion:** For $n = K + 1, K + 2, \dots, L$ (available data length) If $|\varepsilon_{n|n-1}| > \gamma_n$ Form X_n matrix from present and chosen past K data vectors along with corresponding \boldsymbol{y}_n and $\boldsymbol{\gamma}_n$ vectors. $\boldsymbol{\varepsilon}_{n|n-1} = \boldsymbol{y}_n - \boldsymbol{X}_n^T \boldsymbol{\theta}_{n-1}$ $R_n = P_{n-1}X_n$ and $G_{n|n-1} = X_n^T R_n$ If S_n exists such that $S_n \overline{G}_{n|n-1}(\varepsilon_{n|n-1} - S_n \gamma_n) > 0$ $\boldsymbol{\lambda}_n = (\boldsymbol{G}_{n|n-1}\boldsymbol{S}_n\boldsymbol{\Gamma}_n)^{-1}(\boldsymbol{\varepsilon}_{n|n-1} - \boldsymbol{S}_n\boldsymbol{\gamma}_n)$ Otherwise, next n. $\boldsymbol{K}_{n} = \boldsymbol{R}_{n} [\boldsymbol{\Lambda}_{n}^{-1} + \boldsymbol{G}_{n|n-1}]^{-1}$ $\boldsymbol{P}_{n} = \boldsymbol{P}_{n-1} - \boldsymbol{K}_{n} \boldsymbol{R}_{n}^{T}$ $\boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + K_n \boldsymbol{\varepsilon}_{n|n-1}$ $\kappa_n = \kappa_{n-1} + \gamma_n^T \Lambda_n (\gamma_n - S_n \varepsilon_{n|n-1}^T)$ (if necessary) Next n.

The computational complexity of the reoptimized algorithm is significantly worse per update than that of QOBE without reconsideration of past data. However, these selective algorithms tend to incorporate so few data that, even with the additional burden at times of update, the overall complexity remains $\mathcal{O}(m)$, the complexity of the computations at a time for which no update occurs. Further, there is empirical evidence that reoptimization may result in a significant reduction in the number of updates over conventional OBE optimization.

3.7 Computational cost

The basic computational costs are described in Section 2.6. The initial data acceptance check remains the same as in the QOBE algorithm. If a QOBE optimal weight exists, the algorithm attempts an optimization over the past K weights (3.19) which requires the computation of $\overline{G}_{n|n-1}$ (K³ floating-point operations (flops)) as well as its multiplication to a vector (K² flops) (3.11). Note that the computation of det ($G_{n|n-1}$) is not necessary to check inequality (3.19) since det ($G_{n|n-1}$) > 0 and

Table 3.2. Computational cost of the MW-QOBE per-innovation check. For simplicity, the notation \tilde{K} is used to denote K + 1.

Expression	Dimension	No. of Multiplications
$\begin{aligned} & \varepsilon_{n n-1} \\ & R_n = P_n X_n \\ & G_{n n-1} \\ & \overline{G}_{n n-1} \\ & S_n \overline{G}_{n n-1} (\varepsilon_{n n-1} - S_n \gamma_n) \\ & \lambda_n \end{aligned}$	$ \tilde{K} \times 1 $ $ m \times \tilde{K} $ $ \tilde{K} \times \tilde{K} $ $ \tilde{K} \times 1 $ $ \tilde{K} \times 1 $	$egin{aligned} & m ilde{K} \ & m^2 ilde{K} \ & (m ilde{K}/2)(ilde{K}+1) \ & < ilde{K}^3 \ & ilde{K}^2 \ & (imes 2^{ ilde{K}}) \ & ar{K}^2 pprox ilde{K}^4 \ & + \det \left(oldsymbol{G}_{n n-1} ight) \end{aligned}$
Total		$rac{3m}{ ilde{K}/2}(m+ ilde{K}+2) \ + ilde{K}(ilde{K}^2+ ilde{K}+4)$

therefore does not change this inequality. Table 3.2 summarizes the computational cost of the MW-QOBE algorithm.

The computational cost is decreased with increased K due the reduction in observations deemed innovative (avoiding laborious recursions).

3.8 MW-QOBE with K = 1

Readjusting a single past weight offers important insight into behavior of the general MW-QOBE algorithm. The innovation check requires the computation of the vector $\varepsilon_{n|n-1}$ (2m flops), the 2 × 2 matrix $G_{n|n-1}$ (2m² + 3m flops) and a maximum of four tests of condition (3.19), each requiring four multiplications (see Table 3.3).

Expression	No. of Multiplications
$c_{n n-1}$ $R_n = P_n X_n$ $G_{n n-1}$ $\overline{G}_{n n-1}$ $S_n \overline{G}_{n n-1} (\epsilon_{n n-1} - S_n \gamma_n)$ λ_n	$2m$ $2m^2$ $3m$ $-$ $4 (\times 4, \max)$ 4
Total	$2m^2 + 5m + [8 - 20]$

Table 3.3. Computational cost of the MW-QOBE per-observation innovation check when K = 1.

3.9 Illustrative examples

3.9.1 Decrease in ellipsoid volume

In order to illustrate the effect of increased K in MW-QOBE algorithms, we consider the AR(2) system,

$$y_n = -0.10y_{n-1} - 0.56y_{n-2} + \varepsilon_{n*} \tag{3.35}$$

where ε_{n*} is uniformally distributed over the interval (-1, +1). We use the MW-QOBE with K = 0 (QOBE) and MW-QOBE (K = 1) to identify this AR(2) system of length 100. The QOBE algorithm found 21 points relevant to the optimization process as compared to the five used by MW-QOBE with K = 1. Figure 3.2 shows the ellipsoid generated by the two techniques at times n = 8 and n = 13 with corresponding polytopes, ellipsoid centers and true parameters. The MW-QOBE ellipsoids show improved alignment with the major axis of the polytope, and reduced volumes.

We note that QOBE does not focus on decreasing volume. In fact, it more-or-less "ignores" the ellipsoid altogether in the attempt to minimize κ_n .

3.9.2 Weight assignments

In this section, we illustrate the weight assignment mechanism in the MW-QOBE algorithms and their data screening behavior. We consider the AR(3) system,

$$y_n = 0.49y_{n-1} + 0.61y_{n-2} + 0.58y_{n-3}\varepsilon_{n*}$$
(3.36)

where ε_{n*} is biased and found in the interval [-0.5, +1]. Table 3.4 show the first 38 weight assignments. As expected, the QOBE algorithms uses the largest number of observations in the optimization process, followed by the MW-QOBE (K = 1) and MW-QOBE (K = 2) algorithms. At time n = 17 (K = 2), the non-zero weight $q_{17,17} = \lambda_{17} = [9.34 \times 10^{-4}, 1.904 \times 10^{-3}, 4.148 \times 10^{-3}]^T$ is assigned to observations previously ignored. Therefore, observation vectors \boldsymbol{x}_{15} and \boldsymbol{x}_{16} became relevant in light of the new observation vector \boldsymbol{x}_{17} . We also observe at times n=15, 16 and 17 (K = 1) the compounding effects of weight adjustment as consecutive observation vector \boldsymbol{x}_{15} increased from 6.242×10^{-3} to 6.551×10^{-3} and the one applied to \boldsymbol{x}_{16} decreased from 1.906×10^{-3} to 5.443×10^{-4} . This example illustrates the importance of allowing negative adjustments (within constraints) to past data vectors which may not convey as much information as previously computed in light of a current observation.

The MW-QOBE (K = 2) only used three observation vectors in identifying (3.36), compared with the 32 and 10 needed by MW-QOBE (K = 1) and QOBE, respectively. Using this reduced number of observations, the MW-QOBE (K = 2) was able to identify system (3.36) in a comparable amount time to that required by smaller K, as seen in Figure 3.3a. The associated volume is greater (Figure 3.3b) due to the smaller number of points taken (even though, per update, increasing K decreases volume). A similar remark is true of the plot of κ_n in Figure 3.4b. The *a posteriori* error in Figure 3.4a shows its mapping to the error bound (when an observation is accepted).



(b) Time n = 13.

Figure 3.2. OBE ellipsoids resulting from the system identification of AR(2) system $y_n = -0.10y_{n-1} - 0.56y_{n-2} + \varepsilon_{n*}$ by QOBE (dashed line) and MW-QOBE (solid line, K=1) at times n = 8 and 13. The star (*) represents the "true" parameter and the circles (\circ) the central estimators (superimposed). The underlying polytopes (exact feasible sets) are also shown.

Table 3.4. First 30 weights assigned by QOBE and MW-QOBE (K = 1,2) algorithms in the identification of the AR(3) system $y_n = 0.49y_{n-1} + 0.61y_{n-2} + 0.58y_{n-3} + \varepsilon_{n*}$, where the "true" measurement error sequence $\{\varepsilon_{n*}\}$ is uniformly distributed over the interval (-1, +1). The QOBE and MW-QOBE (K = 1,2) algorithms selected 32, 10, 3 observations, respectively, from a total of 100.

	QOBE	MW-QOBE (K=1)		MW	-QOBE	(K=2)
Time,n	$q_{n,n}$	$q_{n,n}$	$q_{n,n-1}$	$q_{n,n}$	$q_{n,n-1}$	$q_{n,n-2}$
	$\times 10^{-4}$	×10 ⁻⁴	×10 ⁻⁴	$\times 10^{-4}$	×10 ⁻⁴	×10 ⁻⁴
5	1.837	-	-		-	-
7	6.203	12.002	7.117	-	-	-
8	-	-	12.002	-	-	-
9	50.115	- 1	-	- 1	-	-
14	15.073	-	-	-	-	-
15	16.917	62.422	25.913	-	-	-
16	20.392	19.061	65.508	-	-	-
17	6.189	21.261	5.443	9.340	19.039	41.481
18	2.025	- 1	21.261	-	9.340	19.039
19	-	- 1	-	-	-	9.340
20	0.624	-	-	-	-	-
24	1.798	-	-	-	-	-
25	-	15.458	10.908	-	-	-
26	-	-	15.458	-	-	-
28	-	13.325	1.661	-	-	-
29	1.546	-	13.325	-	-	-
30	0.475	-	-	-	-	-
32	0.612	-	-	-	-	-
34	0.059	-	-	-	-	-
36	0.034	-	-	-	-	-
39	0.357	3.191	4.099	-	-	-
40	0.515	1.301	3.764	-	-	-
41	-	-	1.301	-	-	-
44	2.757	-	-	-	-	-
46	3.933	-	-	-	-	-
47	0.685	-	-	-	-	-
54	0.009	-	-	-	-	-
57	0.146	-	-	-	-	-
61	0.219	-	-	-	-	-
62	0.200	-	-	-	-	-



(b) Volume.

Figure 3.3. Convergence of the parameter estimator $\theta_n(2)$ to the "true" parameter $\theta_{n*}(2) = 0.61$ and associated ellipsoid volume in the system identification of $y_n = 0.49y_{n-1} + 0.61y_{n-2} + 0.58y_{n-3} + \varepsilon_{n*}$ by QOBE and MW-QOBE (K=1,2) as in Table 3.4.



Figure 3.4. κ_n and a posteriori error $\varepsilon_{n|n}$ associated with the identification of the system of Figure 3.3.

Chapter 4

Multiple Weight Set-Membership Weighted RLS

4.1 Introduction

Reducing the ellipsoid volume in OBE algorithms shrinks the set of feasible solutions, and assuming convergence of Ω_n to $\{\theta_*\}$, asymptotically forces the central estimator to a closer neighborhood of the true parameter vector. This approach is successfully used in several OBE algorithms including the SM-WRLS [13, 14], which pointwise minimizes the ellipsoid volume in the context of an RLS-like framework. In this chapter we develop an OBE algorithm named after the SM-WRLS, the MW-SM-WRLS algorithm, which extends the pointwise SM-WRLS minimization of volume to an algorithm that optimizes the volume over several past weights.

4.2 Volume minimization

The volume of ellipsoid $\overline{\Omega}_n$ is measured by either the trace or determinant of $\kappa_n P_n$ [14]. The SM-WRLS algorithm minimizes the determinant of $\kappa_n P_n$, which is proportional within a constant to the actual ellipsoid volume, in its derivation of optimal weights. Here, we use the same optimization approach, with the added flexibility of K past weights adjustments. Hereafter the term "ellipsoid volume" or "volume" at time n refers to the determinant of the matrix $\kappa_n P_n$

$$V_n \stackrel{\text{def}}{=} \det\left(\kappa_n \boldsymbol{P}_n\right) = \kappa_n^m \det\left(\boldsymbol{P}_n\right) = \kappa_n^m \frac{\det\left(\boldsymbol{P}_n\right)}{\det\left(\boldsymbol{P}_{n-1}\right)} \det\left(\boldsymbol{P}_{n-1}\right). \tag{4.1}$$

Using (2.18), (4.1) can be written

$$V_n = \kappa_n^m \frac{\det\left(\boldsymbol{P}_{n-1}\right)}{\det\left(\boldsymbol{H}_n\right)} = \frac{\det\left(\boldsymbol{P}_{n-1}\right)\kappa_n^m}{h_n}$$
(4.2)

where $h_n \stackrel{\text{def}}{=} \det(H_n)$. We minimize (4.2) over several weights, in a manner similar to that used in [13]. Minimizing the volume³ with respect to the weight matrix, alternatively represented by (see Section 2.2)

$$\Lambda_{n} = (H_{n} - I) G_{n|n-1}^{-1} = H_{n} G_{n|n-1}^{-1} - G_{n|n-1}^{-1} = G_{n|n-1}^{-1} H_{n}^{T} - G_{n|n-1}^{-1}, \qquad (4.3)$$

is equivalent to minimizing volume with respect to the matrix H_n . The derivative of the volume function with respect to matrix H_n ,

$$\frac{\partial V_n}{\partial \boldsymbol{H}_n} = \det\left(\boldsymbol{P}_{n-1}\right) \left[m \frac{\partial \kappa_n}{\partial \boldsymbol{H}_n} \kappa_n^{m-1} h_n - \frac{\partial h_n}{\partial \boldsymbol{H}_n} \kappa_n^m \right], \qquad (4.4)$$

is minimized by determining a matrix H_n that solves

$$\mathbf{0} = m \left[\frac{\partial \kappa_n}{\partial \boldsymbol{H}_n} \right] h_n - \kappa_n \left[\frac{\partial h_n}{\partial \boldsymbol{H}_n} \right]$$
(4.5)

 $(\kappa_n^{m-1} \text{ and } \det(P_{n-1}) \text{ are both strictly positive numbers.})$ In finding the derivative of κ_n with respect to H_n , we first write κ_n as a function of H_n ,

$$\kappa_{n} = \kappa_{n-1} + \gamma_{n}^{T} \left[(H_{n} - I) G_{n|n-1}^{-1} \right] \gamma_{n} - \varepsilon_{n|n-1}^{T} \left[H_{n}^{-1} (H_{n} - I) G_{n|n-1}^{-1} \right] \varepsilon_{n|n-1}$$

$$= \tilde{\kappa}_{n-1} + \gamma_{n}^{T} \left[H_{n} G_{n|n-1}^{-1} \right] \gamma_{n} + \varepsilon_{n|n-1}^{T} \left[G_{n|n-1}^{-1} (H_{n} G_{n|n-1}^{-1})^{-1} G_{n|n-1}^{-1} \right] \varepsilon_{n|n-1}$$

$$= \tilde{\kappa}_{n-1} + \gamma_{n}^{T} \left[H_{n} G_{n|n-1}^{-1} \right] \gamma_{n} + \tilde{\varepsilon}_{n|n-1}^{T} \left[H_{n} G_{n|n-1}^{-1} \right]^{-1} \tilde{\varepsilon}_{n|n-1}$$
(4.6)

where $\tilde{\kappa}_{n-1} \stackrel{\text{def}}{=} \kappa_{n-1} - \gamma_n^T G_{n|n-1}^{-1} \gamma_n - \varepsilon_{n|n-1}^T G_{n|n-1}^{-1} \varepsilon_{n|n-1}$ and $\tilde{\varepsilon}_{n|n-1} \stackrel{\text{def}}{=} G_{n|n-1}^{-1} \varepsilon_{n|n-1}$. Using the formulae given by Athans [1] (see Lemma A.6), we then find

$$\frac{\partial h_n}{\partial H_n} = h_n H_n^{-T} \tag{4.7}$$

$$\frac{\partial \kappa_n}{\partial H_n} = \gamma_n \gamma_n^T G_{n|n-1}^{-1} - H_n^{-T} G_{n|n-1} \tilde{\varepsilon}_{n|n-1} \tilde{\varepsilon}_{n|n-1}^T H_n^{-T}.$$
(4.8)

³Minimizing the "volume ratio" as in [13] or "volume" lead to identical optimizations.

The volume minimizing equation (4.5) becomes

$$0 = m \gamma_n \gamma_n^T G_{n|n-1}^{-1} - m H_n^{-T} G_{n|n-1} \tilde{\varepsilon}_{n|n-1} \tilde{\varepsilon}_{n|n-1}^T H_n^{-T} - \kappa_n H_n^{-T}$$

$$= m H_n^T \gamma_n \gamma_n^T G_{n|n-1}^{-1} H_n^T - m G_{n|n-1} \tilde{\varepsilon}_{n|n-1} \tilde{\varepsilon}_{n|n-1}^T - \kappa_n H_n^T$$

$$= m G_{n|n-1}^{-1} H_n^T \left(\gamma_n \gamma_n^T \right) G_{n|n-1}^{-1} H_n^T - m \left(\tilde{\varepsilon}_{n|n-1} \tilde{\varepsilon}_{n|n-1}^T \right) - \kappa_n G_{n|n-1}^{-1} H_n^T$$
(4.9)

or, in matrix form,

$$\mathbf{0} = \begin{bmatrix} \mathbf{G}_{n|n-1}^{-1} \mathbf{H}_{n}^{T} & | \mathbf{I} \end{bmatrix} \begin{bmatrix} m \boldsymbol{\gamma}_{n} \boldsymbol{\gamma}_{n}^{T} & | -\frac{1}{2} \kappa_{n} \mathbf{I} \\ ---- & | ----- \\ -\frac{1}{2} \kappa_{n} \mathbf{I} & | -m \tilde{\boldsymbol{\varepsilon}}_{n|n-1} \tilde{\boldsymbol{\varepsilon}}_{n|n-1}^{T} \end{bmatrix} \begin{bmatrix} \mathbf{G}_{n|n-1}^{-1} \mathbf{H}_{n}^{T} \\ ---- \\ \mathbf{I} \end{bmatrix}$$
(4.10)

where $G_{n|n-1}^{-1}H_n^T$ is the unknown, symmetric positive-definite, matrix.

Remark: When K = 0, the solution to (4.9),

$$\frac{H_n}{G_{n|n-1}} = \frac{\tilde{\kappa}_{n-1} + \sqrt{\tilde{\kappa}_{n-1}^2 + 4(m^2 - 1)\gamma_n^2 \tilde{\varepsilon}_{n|n-1}^2}}{2(m-1)\gamma_n^2}$$

corresponds to the SM-WRLS weight

$$\lambda_{n} = \frac{G_{n|n-1}[H_{n}/G_{n|n-1}] - 1}{G_{n|n-1}}$$

=
$$\frac{G_{n|n-1}\tilde{\kappa}_{n-1} - 2(m-1)\gamma_{n}^{2} + G_{n|n-1}\sqrt{\tilde{\kappa}_{n-1}^{2} + 4(m^{2}-1)\gamma_{n}^{2}\tilde{\varepsilon}_{n|n-1}^{2}}}{2(m-1)G_{n|n-1}\gamma_{n}^{2}},$$

where $\tilde{\kappa}_{n-1} = \kappa_{n-1} - \varepsilon_{n|n-1}^2 / G_{n|n-1} - \gamma_{n|n-1}^2 / G_{n|n-1}$, $\tilde{\varepsilon}_{n|n-1} = \varepsilon_{n|n-1} / G_{n|n-1}$, and where H_n and $G_{n|n-1}$ are the scalar versions of H_n and $G_{n|n-1}$.

4.3 Optimal solution existence and uniqueness

In this section we provide a method for solving (4.9) for the matrix H_n , describe a condition for the existence of a solution and show the uniqueness of this solution.

Without loss of generality, we assume that the weight adjustment vector λ_n has all positive elements, as demonstrated in Chapter 5. Notation and definitions needed

Table 4.1.	Notation	used in	Chapter	4.
------------	----------	---------	---------	----

Notation	Definition
u	$n \times 1$ vector
v	$n \times 1$ vector with $\boldsymbol{v}(i) > 0$, for all $i \in [1, n]$
m	$m \in \aleph$, with $m \ge n$
\mathcal{M}_{n^+}	set of all real symmetric positive definite $n \times n$ matrices
X	$X \in \mathcal{M}_{n^+}$
W	$W \in \mathcal{M}_{n^+}$ with diagonal elements $w_i \stackrel{ ext{def}}{=} W(i,i)$
D	diagonal matrix of eigenvalues, d_i , of X [i.e., $d_i \stackrel{\text{def}}{=} D(i, i)$]
R	orthogonal matrix such that $X = R^T D R$
α_i	$lpha_i \stackrel{ ext{def}}{=} rac{oldsymbol{u}(i)}{oldsymbol{v}(i)} \left[moldsymbol{u}(i)oldsymbol{v}(i) - oldsymbol{u}^Toldsymbol{v} ight] ext{ for all } i \in [1,n]$
eta_i	$eta_i \stackrel{ ext{def}}{=} rac{m{v}(i)}{m{u}(i)} \left[mm{u}(i)m{v}(i) + m{u}^Tm{v} ight] ext{ for all } i \in [1,n]$

Table 4.2. Function definitions used in Chapter 4.

FunctionDefinition
$$b$$
 $b = c - u^T W^{-1} u - v^T W^{-1} v$ where $c > 0$ f $f: \mathcal{M}_{n^+} \to \Re$ $f(Z) = b + u^T Z u + v^T Z^{-1} v$ F $F: \mathcal{M}_{n^+} \to \mathcal{M}_{n^+}$ $F(Z) = Z u u^T Z - f(Z) Z - m v v^T$

throughout the rest of this chapter given in Tables 4.1 and 4.2.

The following lemma is necessary to find a condition for the existence of a valid weight adjustment vector λ_n (or matrix $G_{n|n-1}H_n^T$) which solves (4.9).

Lemma 4.1
$$f(X) \to \infty$$
 and $\frac{f^m(X)}{\det(X)} \to \infty$ as trace $(X) \to \infty$.

Proof: We re-write the function f(X) as

$$f(\boldsymbol{X}) = b + \|\boldsymbol{u}\|^2 \frac{\boldsymbol{u}^T}{\|\boldsymbol{u}\|} \boldsymbol{X} \frac{\boldsymbol{u}}{\|\boldsymbol{u}\|} + \|\boldsymbol{v}\|^2 \frac{\boldsymbol{v}^T}{\|\boldsymbol{v}\|} \boldsymbol{X}^{-1} \frac{\boldsymbol{v}}{\|\boldsymbol{v}\|}$$
$$= b + \|\boldsymbol{u}\|^2 \bar{\boldsymbol{u}}^T \boldsymbol{D} \overline{\boldsymbol{u}} + \|\boldsymbol{v}\|^2 \bar{\boldsymbol{v}}^T \boldsymbol{D}^{-1} \overline{\boldsymbol{v}}$$

where $\overline{\boldsymbol{u}} = \boldsymbol{R}\boldsymbol{u}/\|\boldsymbol{u}\|$ and $\overline{\boldsymbol{v}} = \boldsymbol{R}\boldsymbol{v}/\|\boldsymbol{v}\|$, each having unity norm. Since $d_{\min} < \overline{\boldsymbol{u}}^T \boldsymbol{D}\overline{\boldsymbol{u}} < d_{\max}$ and $d_{\max}^{-1} < \overline{\boldsymbol{v}}^T \boldsymbol{D}^{-1}\overline{\boldsymbol{v}} < d_{\min}^{-1}$, the function $f(\boldsymbol{X})$ is bounded by

$$b + \|\boldsymbol{u}\|^2 d_{\min} + \|\boldsymbol{v}\|^2 d_{\max}^{-1} < f(\boldsymbol{X}) < b + \|\boldsymbol{u}\|^2 d_{\max} + \|\boldsymbol{v}\|^2 d_{\min}^{-1}.$$

Let j be the index of the largest eigenvalue of X. The function f(X) is then decomposed into the two terms

$$f(\boldsymbol{X}) = b + \|\boldsymbol{u}\|^2 \,\overline{\boldsymbol{u}}^2(j) d_j + g(\boldsymbol{X})$$

where, $g(\boldsymbol{X}) = \|\boldsymbol{u}\|^2 \sum_{i \neq j} \overline{\boldsymbol{u}}^2(i) d_i + \|\boldsymbol{v}\|^2 \sum_i \frac{\overline{\boldsymbol{v}}^2(i)}{d_i}.$

The function g(X) is a sum of positive elements and is therefore positive. We conclude therefore that $f(X) \to \infty$ as trace $(X) \to \infty$ $(d_j \to \infty \text{ implies trace } (X) \to \infty)$. We rewrite f(X) as

$$f(\boldsymbol{X}) = d_j \left[\|\boldsymbol{u}\|^2 \overline{\boldsymbol{u}}^2(j) + \frac{b}{d_j} + \sum_{i=1}^n \frac{\overline{\boldsymbol{v}}^2(i)}{d_i d_j} \right],$$

so that

$$\frac{f^m(\boldsymbol{X})}{\det(\boldsymbol{X})} = \frac{d_j^m}{\prod_{i=1}^n d_i} \left[\|\boldsymbol{u}\|^2 \overline{\boldsymbol{u}}^2(j) + \frac{b}{d_j} + \sum_{i=1}^n \frac{\overline{\boldsymbol{v}}^2(i)}{d_i d_j} \right]^m.$$

Since the term

$$\left[\frac{b}{d_j} + \sum_{i=1}^n \frac{\overline{v}^2(i)}{d_i d_j}\right] \to \infty \text{ as } d_j \to \infty,$$

and since m > n implies that $d_j^m > \prod_{i=1}^n d_i$, we have

$$\frac{f^m(\boldsymbol{X})}{\det(\boldsymbol{X})} \to \left[\|\boldsymbol{u}\| \overline{\boldsymbol{u}}(j) \right]^{2m} \frac{d_j^m}{\prod_{i=1}^n d_i} \to \infty \text{ as } d_j \to \infty,$$

from which we obtain

$$rac{f^m(oldsymbol{X})}{\det{(oldsymbol{X})}}
ightarrow\infty ext{ as } d_j
ightarrow\infty.$$

Hence,

$$rac{f^m(X)}{\det(X)} o \infty \ \ ext{as trace} (X) o \infty.$$

Theorem 4.1 (Existence) A sufficient condition for existence of a positive weight vector that solves (4.9) at time n is

$$\left(\boldsymbol{\gamma}_{n}\boldsymbol{\gamma}_{n}^{T}-\boldsymbol{\varepsilon}_{n\mid n-1}\boldsymbol{\varepsilon}_{n\mid n-1}^{T}\right)\boldsymbol{G}_{n\mid n-1}^{-1}<\frac{\kappa_{n-1}}{m}\boldsymbol{I}.$$
(4.11)

Proof: The volume (4.1) is a continuous function in the positive hyperquadrant of \Re^{K+1} (see Chapter 2). From the definition of H_n , we have

trace
$$(\boldsymbol{H}_n) = \sum_{i=1}^{K+1} \left[1 + \lambda_n(i) \boldsymbol{G}_{n|n-1}(i,i) \right].$$

Therefore $\|\lambda_n\| \to \infty$ implies that trace $(H_n) \to \infty$. Let $X = H_n G_{n|n-1}^{-1}$, $b = \tilde{\kappa}_{n-1}$, $u = \gamma_n$ and $v = \tilde{\epsilon}_{n|n-1}$ with $\tilde{\kappa}_{n-1}$ and $\tilde{\epsilon}_{n|n-1}$ as in (4.6). By Lemma 4.1 we have

$$\kappa_n \to \infty$$
 and $\frac{\kappa_n^m}{h_n} \to \infty$ as $\|\lambda_n\| \to \infty$,

hence $V_n(\lambda_n) \to \infty$ as $||\lambda_n|| \to \infty$ (see Equation 4.1). We also know that $V_n(0)$ must be greater that $V_n(\lambda_n)$ if an optimal weight adjustment, λ_n , exits. Therefore, the condition that all partial derivatives be negative at $\lambda_n = 0$ is sufficient for the existence of a solution in the positive hyperquadrant. By substituting $\Lambda_n = 0$ in (4.7) and (4.8) we obtain

$$\frac{\partial h_n}{\partial H_n}\Big|_{\lambda_n=0} = I$$

$$\frac{\partial \kappa_n}{\partial H_n}\Big|_{\lambda_n=0} = \gamma_n \gamma_n^T G_{n|n-1}^{-1} - G_{n|n-1} \tilde{\varepsilon}_{n|n-1} \tilde{\varepsilon}_{n|n-1}^T$$

$$= \gamma_n \gamma_n^T G_{n|n-1}^{-1} - \varepsilon_{n|n-1} \varepsilon_{n|n-1}^T G_{n|n-1}^{-1}$$

which, when incorporated into (4.4), produces the matrix

$$\frac{\partial V_n}{\partial H_n}\Big|_{\boldsymbol{\lambda}_n=\boldsymbol{0}} = \frac{m}{\kappa_{n-1}} \left(\boldsymbol{\gamma}_n \boldsymbol{\gamma}_n^T \boldsymbol{G}_{n|n-1}^{-1} - \boldsymbol{\varepsilon}_{n|n-1} \boldsymbol{\varepsilon}_{n|n-1}^T \boldsymbol{G}_{n|n-1}^{-1} \right) - \boldsymbol{I}$$
$$= \left(\boldsymbol{\gamma}_n \boldsymbol{\gamma}_n^T - \boldsymbol{\varepsilon}_{n|n-1} \boldsymbol{\varepsilon}_{n|n-1}^T \right) \boldsymbol{G}_{n|n-1}^{-1} - \frac{\kappa_{n-1}}{m} \boldsymbol{I}$$
(4.12)

with trace

trace
$$\frac{\partial V_n}{\partial H_n}\Big|_{\boldsymbol{\lambda}_n=\mathbf{0}} = \boldsymbol{\gamma}_n^T \boldsymbol{G}_{n|n-1}^{-1} \boldsymbol{\gamma}_n - \boldsymbol{\varepsilon}_{n|n-1}^T \boldsymbol{G}_{n|n-1}^{-1} \boldsymbol{\varepsilon}_{n|n-1} - \frac{(K+1)}{m} \kappa_{n-1}.$$
 (4.13)

The check for existence of a valid weight is therefore the matrix inequality (4.11)

Remark: From (4.13) we obtain

$$\frac{(K+1)}{m}\kappa_{n-1}-\boldsymbol{\gamma}_n^T\boldsymbol{G}_{n|n-1}^{-1}\boldsymbol{\gamma}_n > \boldsymbol{\varepsilon}_{n|n-1}^T\boldsymbol{G}_{n|n-1}^{-1}\boldsymbol{\varepsilon}_{n|n-1} > 0 \qquad (4.14)$$

which implies $\kappa_{n-1} - \gamma_n^T G_{n|n-1}^{-1} \gamma_n > \varepsilon_{n|n-1}^T G_{n|n-1}^{-1} \varepsilon_{n|n-1} > 0$, thus $\tilde{\kappa}_{n-1} > 0$.

We next turn our attention to solving (4.9). At present, no closed form solution for K > 1 has been found, but we present a general approach for solving the problem, which simplifies the quest for solutions. The following lemma represents the principal algebraic result needed to derive the optimal weight adjustment vector λ_n .

Lemma 4.2 F(D) = 0 with constraint f(D) > 0 when

$$d_{i} = \frac{1}{2\alpha_{i}}(b + \sqrt{b^{2} + 4\alpha_{i}\beta_{i}}). \qquad (4.15)$$

Proof: The (i, j) equation in F(D) = 0 when $i \neq j$:

$$m(d_i d_j)[\boldsymbol{u}(i)\boldsymbol{u}(j)] - m[\boldsymbol{v}(i)\boldsymbol{v}(j)] = 0$$

or $d_i d_j = \frac{\boldsymbol{v}(i)\boldsymbol{v}(j)}{\boldsymbol{u}(i)\boldsymbol{u}(j)}.$ (4.16)

The (i, j) equation in F(D) = 0 when i = j:

$$0 = md_{i}^{2}\boldsymbol{u}(i)^{2} - b\,d_{i} - d_{i}\sum_{j}\boldsymbol{u}^{2}(j)d_{j} - d_{i}\sum_{j}\frac{\boldsymbol{v}^{2}(j)}{d_{j}} - m\boldsymbol{v}(i)^{2}$$

$$= (m-1)d_{i}^{2}\boldsymbol{u}(i)^{2} - b\,d_{i} - (m+1)\boldsymbol{v}(i)^{2} - \sum_{j\neq i}\left(\boldsymbol{u}^{2}(j)d_{i}d_{j} + \boldsymbol{v}^{2}(j)\frac{d_{i}}{d_{j}}\right)$$

$$= (m-1)d_{i}^{2}\boldsymbol{u}(i)^{2} - b\,d_{i} - (m+1)\boldsymbol{v}(i)^{2}$$

$$- \sum_{j\neq i}\left(\boldsymbol{u}^{2}(j)d_{i}d_{j} + \boldsymbol{v}^{2}(j)\frac{d_{i}^{2}}{d_{i}d_{j}}\right). \qquad (4.17)$$

Substituting (4.16) in (4.17), we obtain

$$0 = \left[(m-1)u(i)^{2} - \sum_{j \neq i} \frac{u(i)u(j)v(j)}{v(i)} \right] d_{i}^{2} - b d_{i} - (m+1)v(i)^{2} - \sum_{j \neq i} \frac{v(i)v(j)u(j)}{u(i)}$$

$$= \left[(m-1)u(i)^{2} - \frac{u(i)}{v(i)} \sum_{j \neq i} u(j)v(j) \right] d_{i}^{2} - b d_{i} - (m+1)v(i)^{2} - \frac{v(i)}{u(i)} \sum_{j \neq i} u(j)v(j)$$

$$= \left[(m-1)u(i)^{2} - \frac{u(i)}{v(i)} (u^{T}v - u(i)v(i)) \right] d_{i}^{2} - b d_{i} - (m+1)v(i)^{2}$$

$$- \frac{v(i)}{u(i)} (u^{T}v - u(i)v(i))$$

$$= \left[mu(i)^{2} - \frac{u(i)}{v(i)} u^{T}v \right] d_{i}^{2} - b d_{i} - \left[mv(i)^{2} + \frac{v(i)}{u(i)} u^{T}v \right]$$

$$= d_{i}^{2} - \frac{u(i)}{v(i)} \frac{b}{[mu(i)v(i) - u^{T}v]} d_{i} - \frac{[mu(i)v(i) + u^{T}v]}{[mu(i)v(i) - u^{T}v]}$$
(4.18)
$$= d_{i}^{2} - \frac{b}{\alpha_{i}} d_{i} - \frac{\beta_{i}}{\alpha_{i}}.$$
(4.19)

Equation (4.19) is quadratic in d_i with solutions (4.15).

We now revisit the condition for existence of the MW-SM-WRLS optimal solutions through Lemma 4.3 by determining a condition for the existence for the roots of (4.19).

Lemma 4.3 A matrix D satisfying F(D) = 0 with constraint f(D) > 0 and $d_i > w_i$ exists if and only if

$$\alpha_i(\beta_i + bw_i) > (\alpha_i w_i)^2. \qquad (4.20)$$

Proof: From $d_i > w_i$ we obtain

$$\frac{1}{\alpha_i}(\beta_i + bw_i) > w_i^2 \tag{4.21}$$

$$\alpha_i(\beta_i + bw_i) > (\alpha_i w_i)^2. \tag{4.22}$$

Remark: When n = 1, we have $\alpha = u^2(m-1)$, $\beta = v^2(m+1)$, w = 1/G and $b = c - \frac{u^2}{G} - v^2 G$. Equation (4.20) becomes

$$u^2v^2(m-1)(m+1) > \frac{u^2}{G}(m-1)\left[\frac{u^2}{G}(m-1)-b\right]$$

which reduces to

$$mG^2v^2 > mu^2 - cG.$$
 (4.23)

We obtain the SM-WRLS innovation check by $u = \gamma$, $v = \varepsilon_{n|n-1}/G_n$ and $c = \kappa_{n-1}$.

Theorem 4.2 When the weight adjustment vector λ_n exists, the matrix $H_n G_{n|n-1}^{-1}$ solves (4.9), and λ_n solves (4.3).

Proof: Let $X = H_n G_{n|n-1}^{-1}$, $b = \tilde{\kappa}_{n-1}$, $u = \gamma_n$ and $v = \tilde{\epsilon}_{n|n-1}$, where $\tilde{\kappa}_{n-1}$ and $\tilde{\epsilon}_{n|n-1}$ are as in (4.6). Apply Lemma 4.2 to find $H_n G_{n|n-1}^{-1}$.

Next we prove that at most one weight adjustment vector λ_n (or matrix $G_{n|n-1}^{-1}H_n^T$) satisfies (4.9) with constraint (3.19). The following lemma is required.

Lemma 4.4 There exists at most one $X \in \mathcal{M}_{n^+}$ satisfying F(X) = 0.

Proof: Let X_1 and $X_2 \in \mathcal{M}_{n^+}$ such that $F(X_1) = 0$ and $F(X_2) = 0$. Let M be a non-singular matrix such that $MX_1M^T = I$ and $MX_2M^T = D$, where D is a diagonal matrix with positive elements. The existence of such a matrix D is proven in [41]. Let $\overline{u} = Mu$ and $\overline{v} = M^{-T}v$. We then have

$$0 = m \boldsymbol{D} \overline{\boldsymbol{u}} \overline{\boldsymbol{u}}^T \boldsymbol{D} - (b + \overline{\boldsymbol{u}}^T \boldsymbol{D} \overline{\boldsymbol{u}} + \overline{\boldsymbol{v}}^T \boldsymbol{D}^{-1} \overline{\boldsymbol{v}}) \boldsymbol{D} - m \overline{\boldsymbol{v}} \overline{\boldsymbol{v}}^T$$
(4.24)

$$0 = m\overline{\boldsymbol{u}}\,\overline{\boldsymbol{u}}^T - (b + \overline{\boldsymbol{u}}^T\overline{\boldsymbol{u}} + \overline{\boldsymbol{v}}^T\overline{\boldsymbol{v}})\boldsymbol{I} - m\overline{\boldsymbol{v}}\,\overline{\boldsymbol{v}}^T.$$
(4.25)

When $i \neq j$, the (i, j) equations of (4.24) and (4.25) become

$$m(d_i d_j)[\boldsymbol{u}(i)\overline{\boldsymbol{u}}(j)] - m[\boldsymbol{v}(i)\overline{\boldsymbol{v}}(j)] = 0$$
$$m[\boldsymbol{u}(i)\overline{\boldsymbol{u}}(j)] - m[\boldsymbol{v}(i)\overline{\boldsymbol{v}}(j)] = 0$$

implying that $d_i d_j = 1$. From the diagonal equations (i, i) of (4.24) and (4.25) we have

$$0 = \left[m \boldsymbol{u}(i)^2 - \frac{\boldsymbol{u}(i)}{\boldsymbol{v}(i)} \bar{\boldsymbol{u}}^T \bar{\boldsymbol{v}} \right] d_i^2 - b d_i - \left[m \boldsymbol{v}(i)^2 + \frac{\boldsymbol{v}(i)}{\boldsymbol{u}(i)} \bar{\boldsymbol{u}}^T \bar{\boldsymbol{v}} \right]$$
$$0 = \left[m \boldsymbol{u}(i)^2 - \frac{\boldsymbol{u}(i)}{\boldsymbol{v}(i)} \bar{\boldsymbol{u}}^T \bar{\boldsymbol{v}} \right] - b - \left[m \boldsymbol{v}(i)^2 + \frac{\boldsymbol{v}(i)}{\boldsymbol{u}(i)} \bar{\boldsymbol{u}}^T \bar{\boldsymbol{v}} \right]$$

thus $d_i = 1$ for all *i*. The matrix **D** is therefore the identity matrix, indicating that X_1 must equal X_2 . Therefore, a solution to F(X) = 0 must be unique.

Theorem 4.3 (Uniqueness) At most one weight adjustment vector λ_n (or $G_{n|n-1}^{-1}H_n^T$) satisfies (4.9).

Proof: Let $X = G_{n|n-1}^{-1}H_n^T$, $b = \tilde{\kappa}_{n-1}$, $u = \gamma_n$ and $v = \tilde{\epsilon}_{n|n-1}$, with $\tilde{\kappa}_{n-1}$ and $\tilde{\epsilon}_{n|n-1}$ as in (4.6). Apply Lemma 4.4.

4.4 MW-SW-WRLS with K = 1

Optimizing the ellipsoid volume by re-visiting a single past weight is the simplest form of MW-SM-WRLS. It offers significant insight into the behavior of the algorithm and presently is the only case with a closed form solution. We solve for the matrix $H_n G_{n|n-1}^{-1}$ from which we obtain the weight adjustments. In order to simplify
notation, the optimal solution to (4.9) when K = 1 is presented through the following lemma.

Lemma 4.5 Let
$$n = 2$$
. The solution to $F(\mathbf{X}) = 0$ with $f(\mathbf{X}) > 0$ is $\mathbf{X} = \mathbf{R}^T D \mathbf{R}$,
with $\mathbf{R} = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix}$, $\mathbf{D} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}$ and $\mathbf{X}(1, 2) = g_{12}$, where
 $\psi = \frac{\phi}{2} - \frac{pi}{4}$
 $\phi = \arctan \frac{1}{2} \frac{m || \mathbf{v} || (\mathbf{u}_1 - \mathbf{u}_2) (\mathbf{u}_1 + \mathbf{u}_2) + 2\mathbf{u}^T \mathbf{v} (\mathbf{u}_1 \mathbf{v}_1 - \mathbf{u}_2 \mathbf{v}_2)}{m || \mathbf{v} || \mathbf{u}_1 \mathbf{u}_2 + \mathbf{u}^T \mathbf{v} (\mathbf{u}_1 \mathbf{v}_2 + \mathbf{u}_2 \mathbf{v}_1)}$
 $d_i = \frac{1}{2\alpha_i} (b + \sqrt{b^2 + 4\alpha_i \beta_i})$ for $i = 1, 2$.

Proof: Let

$$\boldsymbol{R} = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix} \text{ and } \boldsymbol{D} = \begin{bmatrix} d_1 & 0 \\ 0 & d_2 \end{bmatrix}$$
(4.26)

where ψ is a rotation angle. We then obtain

$$g_{12} = (d_2 - d_1) \cos \psi \sin \psi = \frac{1}{2} (d_2 - d_1) \sin 2\psi.$$
 (4.27)

From (4.27) and $d_1d_2 = \frac{\overline{v}_1\overline{v}_2}{\overline{u}_1\overline{u}_2}$, we obtain the quadratic equation in d_1 ,

$$0 = d_1^2 + \frac{2g_{12}}{\sin 2\psi} d_1 - \frac{\overline{\boldsymbol{v}}_1 \overline{\boldsymbol{v}}_2}{\overline{\boldsymbol{u}}_1 \overline{\boldsymbol{u}}_2}$$
(4.28)

which, compared with (4.19), yields the set of equations

$$\frac{\overline{\boldsymbol{v}}_1}{\overline{\boldsymbol{u}}_1} \frac{b}{m\overline{\boldsymbol{u}}_1\overline{\boldsymbol{v}}_1 - \boldsymbol{u}^T\boldsymbol{v}} = \frac{-2g_{12}}{\sin 2\psi}$$
(4.29)

$$\frac{\overline{\boldsymbol{u}}_{2}\overline{\boldsymbol{v}}_{1}}{\overline{\boldsymbol{v}}_{2}\overline{\boldsymbol{u}}_{1}}\frac{m\overline{\boldsymbol{u}}_{1}\overline{\boldsymbol{v}}_{1}+\boldsymbol{u}^{T}\boldsymbol{v}}{m\overline{\boldsymbol{u}}_{1}\overline{\boldsymbol{v}}_{1}-\boldsymbol{u}^{T}\boldsymbol{v}} = 1$$
(4.30)

$$\frac{\overline{\boldsymbol{v}}_2}{\overline{\boldsymbol{u}}_2} \frac{b}{m\overline{\boldsymbol{u}}_2\overline{\boldsymbol{v}}_2 - \boldsymbol{u}^T\boldsymbol{v}} = \frac{2g_{12}}{\sin 2\psi}$$
(4.31)

$$\frac{\overline{\boldsymbol{u}}_1 \overline{\boldsymbol{v}}_2}{\overline{\boldsymbol{v}}_1 \overline{\boldsymbol{u}}_2} \frac{m \overline{\boldsymbol{u}}_2 \overline{\boldsymbol{v}}_2 + \boldsymbol{u}^T \boldsymbol{v}}{m \overline{\boldsymbol{u}}_2 \overline{\boldsymbol{v}}_2 - \boldsymbol{u}^T \boldsymbol{v}} = 1.$$
(4.32)

Combining (4.29-4.32) we obtain the system of equations

$$m\overline{\boldsymbol{u}}_1\overline{\boldsymbol{u}}_2\overline{\boldsymbol{v}}_1 + \overline{\boldsymbol{u}}_2(\boldsymbol{u}^T\boldsymbol{v}) = \frac{-b}{2g_{12}}\overline{\boldsymbol{v}}_2\sin 2\psi \qquad (4.33)$$

$$m\overline{\boldsymbol{u}}_1\overline{\boldsymbol{u}}_2\overline{\boldsymbol{v}}_2 + \overline{\boldsymbol{u}}_1(\boldsymbol{u}^T\boldsymbol{v}) = \frac{b}{2g_{12}}\overline{\boldsymbol{v}}_1\sin 2\psi \qquad (4.34)$$

which simplifies to

$$m \|\boldsymbol{v}\|^2 \overline{\boldsymbol{u}}_1 \overline{\boldsymbol{u}}_2 + \boldsymbol{u}^T \boldsymbol{v} (\overline{\boldsymbol{u}}_1 \overline{\boldsymbol{v}}_2 + \overline{\boldsymbol{u}}_2 \overline{\boldsymbol{v}}_1) = 0.$$
(4.35)

Incorporating the rotation matrix (4.26) into (4.35), we obtain

$$0 = \left[m \| \boldsymbol{v} \|^2 \boldsymbol{u}_1 \boldsymbol{u}_2 + \boldsymbol{u}^T \boldsymbol{v} (\boldsymbol{u}_1 \boldsymbol{v}_2 + \boldsymbol{u}_2 \boldsymbol{v}_1) \right] \cos 2\psi \\ + \frac{1}{2} \left[m \| \boldsymbol{v} \|^2 (\boldsymbol{u}_1 - \boldsymbol{u}_2) (\boldsymbol{u}_1 + \boldsymbol{u}_2) + 2 \boldsymbol{u}^T \boldsymbol{v} (\boldsymbol{u}_1 \boldsymbol{v}_1 - \boldsymbol{u}_2 \boldsymbol{v}_2) \right] \sin 2\psi \quad (4.36)$$

which we solve for the angle ψ . Let

$$\phi = \arctan \frac{1}{2} \frac{m \|v\| (u_1 - u_2) (u_1 + u_2) + 2u^T v (u_1 v_1 - u_2 v_2)}{m \|v\| u_1 u_2 + u^T v (u_1 v_2 + u_2 v_1)}.$$
 (4.37)

The solution ψ of (4.36) becomes

$$0 = \cos\phi\cos 2\psi + \sin\phi\sin 2\psi = \cos(\phi - 2\psi)$$
(4.38)

or,
$$\psi = \frac{\phi}{2} - \frac{\pi}{4} - k\pi \quad k = 0, 1, \cdots$$
 (4.39)

Since the addition of the term $k\pi$ $k = 0, 1, \dots$ in (4.39) does not change the solution of F(X), we have

$$\psi = \frac{\phi}{2} - \frac{\pi}{4}.$$
 (4.40)

	-

I. Initialization: 1. $\boldsymbol{\theta}_{K} = \mathbf{0}, \, \kappa_{K} = 1 \text{ and } \boldsymbol{P}_{K} = \frac{1}{\mu} \boldsymbol{I}, \, \mu \text{ small.}$ 2. $\boldsymbol{q}_{\boldsymbol{K}} = \boldsymbol{0}$. **II. Recursion:** For n = K + 1, K + 2, ...If $\varepsilon_{n|n-1}^2 > \gamma_n^2 - \frac{\kappa_{n-1}G_{n|n-1}}{m}$ Form X_n matrix from present and chosen past K observation vectors along with corresponding \boldsymbol{y}_n and $\boldsymbol{\gamma}_n$ vectors. $\varepsilon_{n|n-1} = y_n - X_n^T \theta_{n-1}$ $M_n = P_{n-1} X_n \text{ and } G_{n|n-1} = X_n^T M_n$ $\tilde{\varepsilon}_{n|n-1} = G_{n|n-1}^{-1} \varepsilon_{n|n-1}, \tilde{\gamma}_n = G_{n|n-1}^{-1} \gamma_n$ If $\left(\gamma_n \gamma_n^T - \varepsilon_{n|n-1} \varepsilon_{n|n-1}^T\right) G_{n|n-1}^{-1} > \frac{\kappa_{n-1}}{m} I$ $\tilde{\kappa}_{n-1} = \kappa_{n-1} - \varepsilon_{n|n-1}^T \tilde{\varepsilon}_{n|n-1} - \gamma_{n|n-1}^T \tilde{\gamma}_{n|n-1}$ Compute the rotation matrix R_n from (4.40) and (4.26) Compute the diagonal matrix D_n from (4.15) $\Lambda_n = \boldsymbol{R}_n^T \boldsymbol{D}_n \boldsymbol{R}_n - \boldsymbol{G}_{n|n-1}^{-1},$ Otherwise, next n. $K_n = M_n [\Lambda_n^{-1} + G_{n|n-1}]^{-1}$ $P_n = P_{n-1} - K_n M_n^T$ $\theta_n = \theta_{n-1} + K_n \varepsilon_{n|n-1}$ $\kappa_{n} = \tilde{\kappa}_{n-1} + \gamma_{n}^{T} \left[\boldsymbol{H}_{n} \boldsymbol{G}_{n|n-1}^{-1} \right] \gamma_{n} + \tilde{\boldsymbol{\varepsilon}}_{n|n-1}^{T} \left[\boldsymbol{H}_{n} \boldsymbol{G}_{n|n-1}^{-1} \right]^{-1} \tilde{\boldsymbol{\varepsilon}}_{n|n-1}$ Next n.

4.5 Algorithm

To date, a closed form for the MW-SM-WRLS optimal weights hasn't been found. We offer an outline of the general algorithmic steps on Table 4.4. An algorithm for the case K = 1 is presented in Table 4.3. When K = 1, we first determine an appropriate rotation angle with corresponding eigenvalues for the matrix $HG_{n|n-1}^{-1}$, then generate the optimal weight vector.

Table 4.4. The MW-SM-WRLS algorithm (non-zero past weights).

I. Initialization: 1. $\theta_K = 0$, $\kappa_K = 1$ and $P_K = \frac{1}{\mu}I$, μ small. 2. $q_K = 0$. II. Recursion: For n = K + 1, K + 2, ...If $\varepsilon_{n|n-1}^2 > \gamma_n^2 - \frac{\kappa_{n-1}G_{n|n-1}}{m}$ Form X_n matrix from present and chosen past K observation vectors along with corresponding y_n and γ_n vectors. $\varepsilon_{n|n-1} = y_n - X_n^T \theta_{n-1}$ $M_n = P_{n-1}X_n$ and $G_{n|n-1} = X_n^T M_n$ $\bar{\varepsilon}_{n|n-1} = G_{n|n-1}^{-1}\varepsilon_{n|n-1}, \ \bar{\gamma}_n = G_{n|n-1}^{-1}\gamma_n$ If $(\gamma_n\gamma_n^T - \varepsilon_{n|n-1}\varepsilon_{n|n-1}^T)G_{n|n-1}^{-1} > \frac{\kappa_{n-1}}{m}I$ $\bar{\kappa}_{n-1} = \kappa_{n-1} - \varepsilon_{n|n-1}^T \bar{\ell}_{n|n-1} - \gamma_{n|n-1}^T \bar{\eta}_{n|n-1}$ Compute optimal $G_{n|n-1}^{-1} H_n^T = R_n^T D_n R_n$, where D_n is diagonal (positive), and R_n , orthogonal. $\Lambda_n = G_{n|n-1}^{-1} H_n^T - G_{n|n-1}^{-1}$ $P_n = P_{n-1} - K_n M_n^T$ $\theta_n = \theta_{n-1} + K_n \varepsilon_{n|n-1}]^{-1}$ $\kappa_n = \tilde{\kappa}_{n-1} + \gamma_n^T [H_n G_{n|n-1}^{-1}] \gamma_n + \tilde{\varepsilon}_{n|n-1}^T [H_n G_{n|n-1}^{-1}]^{-1} \tilde{\varepsilon}_{n|n-1}$ Next n.

4.6 Illustrative examples

4.6.1 Decrease in ellipsoid volume

In order to illustrate the effect of increased K in MW-SM-WRLS algorithms, we consider the AR(2) system (3.35). We use the SM-WRLS and MW-SM-WRLS (K = 1) to identify this AR(2) system of length 100. Figure 4.1 shows the ellipsoid generated by the two techniques at times n = 8 and n = 10 with corresponding polytopes, ellipsoid centers and true parameters. The MW-SM-WRLS ellipsoids show reduced volumes. Note the significant size differences between the MW-QOBE ellipsoids of Figure 3.2 and the MW-SM-WRLS (K = 1) ellipsoids of Figures 4.1. This difference reinforces the remark in section 3.9.1, that MW-QOBE algorithms do not focus on minimizing the ellipsoid volume and therefore may generate large ellipsoids.

4.6.2 Weight assignments

In this section, we illustrate the weight assignment mechanism in the MW-SM-WRLS algorithms and their data screening behavior. We consider the AR(3) system (3.36). Table 4.5 shows the first 38 weight assignments. As expected, the SM-WRLS algorithms use more observations in the optimization process than the MW-SM-WRLS. At time n = 5 (K = 1), the non-zero weight $q_{5,5} = \lambda_5 = [0.857 \times 10^1, 1.437 \times^1]^T$ is assigned to an observation vector previously ignored. Therefore, observation vector \boldsymbol{x}_4 became relevant in light of the new observation vector \boldsymbol{x}_5 . We also observe that, between times n=11 and 12 (K = 1), the weight applied to observation vector \boldsymbol{x}_{11} decreased from 21.24×10^1 to 15.31×10^1 . This example illustrates the importance of allowing negative adjustments (within constraints) to past data vectors which may not convey as much information as previously computed in light of a current observation.

The MW-SM-WRLS (K = 1) used 45 observation vectors in identifying (3.36), compared the 51 needed by SM-WRLS. Using this reduced number of observations, the MW-SM-WRLS (K = 1) was able to identify system (3.36) in a comparable amount time to that required by SM-WRLS, as seen in Figure 4.2a. As expected, the MW-SM-WRLS (K = 1) ellipsoid volume is smaller than that of SM-WRLS (Figure 4.2). No particular conclusion is drawn about the effect of increased K on κ_n (shown in Figure 4.3b.) The *a posteriori* error in Figure 4.3a shows its mapping to less than (or equal to) the error bound (when an observation is accepted.)

4.7 Volume minimization, alternative approach

In this section we describe an alternative approach for solving for the optimal weight matrix that minimizes volume.

Multiplying (4.5) from the right by $\left[\frac{\partial h_n}{\partial \Lambda_n}\right]^{-1}$ we obtain

$$\mathbf{0} = m \left[\frac{\partial \kappa_n}{\partial \Lambda_n} \right] \left[\frac{\partial h_n}{\partial \Lambda_n} \right]^{-1} - \frac{\kappa_n}{h_n}$$
(4.41)

۰.

which, by substituting (3.4), becomes

$$\mathbf{0} = m \left[\Gamma_n^2 - \mathcal{D} (\boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1})^2 \right] \left[\searrow (\boldsymbol{H}_n^{-T} \boldsymbol{G}_{n|n-1}) \right]^{-1} - \kappa_n \boldsymbol{I}$$
(4.42)

$$\mathbf{0} = m \left[h_n^2 \Gamma_n^2 - \mathcal{D}(\overline{H}_n \boldsymbol{\epsilon}_{n|n-1})^2 \right] \left[\searrow (\overline{H}_n G_{n|n-1}) \right]^{-1} - h_n \kappa_n \boldsymbol{I}. \quad (4.43)$$

Minimizing the ellipsoid volume in the MW-OBE context is seen to be equivalent to solving for the weight adjustments, roots of either (4.42) or (4.43). We re-write (3.4) as

$$\frac{\partial \kappa_n}{\partial \Lambda_n} = \Gamma_n^2 - \sum \left(\boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1} \boldsymbol{\varepsilon}_{n|n-1}^T \boldsymbol{H}_n^{-1} \right)$$
(4.44)

to obtain the volume minimizing equation

$$0 = m\Gamma_n^2 - m \searrow \left(\boldsymbol{H}_n^{-T} \boldsymbol{\varepsilon}_{n|n-1} \boldsymbol{\varepsilon}_{n|n-1}^T \boldsymbol{H}_n^{-1} \right) - \kappa_n \searrow \left(\boldsymbol{G}_{n|n-1} \boldsymbol{H}_n^{-1} \right)$$

$$0 = \sum \left[\boldsymbol{H}_n^{-T} \left(\boldsymbol{H}_n^T \Gamma_n^2 \boldsymbol{H}_n - \boldsymbol{\varepsilon}_{n|n-1} \boldsymbol{\varepsilon}_{n|n-1}^T + \frac{\kappa_n}{m} \boldsymbol{H}_n^T \boldsymbol{G}_{n|n-1} \right) \boldsymbol{H}_n^{-1} \right]. \quad (4.45)$$

Equation (4.45) may be re-written the following way:

$$\mathbf{0} = \sum \left[\boldsymbol{H}_n^{-T} \boldsymbol{G}_{n|n-1} (\boldsymbol{T}_n) \boldsymbol{G}_{n|n-1} \boldsymbol{H}_n^{-1} \right]$$

with the symmetric matrix T_n defined by

$$T_{n} = G_{n|n-1}^{-1} \left(H_{n}^{T} \Gamma_{n}^{2} H_{n} - \varepsilon_{n|n-1} \varepsilon_{n|n-1}^{T} - \frac{\kappa_{n}}{m} H_{n}^{T} G_{n|n-1} \right) G_{n|n-1}^{-1}$$

$$= \Gamma_{n}^{2} \Lambda_{n}^{2} + G_{n|n-1}^{-1} \Gamma_{n}^{2} \Lambda_{n} + \Lambda_{n} \Gamma_{n}^{2} G_{n|n-1}^{-1} - \frac{\kappa_{n}}{m} \Lambda_{n}$$

$$+ G_{n|n-1}^{-1} \left(\Gamma_{n}^{2} - \varepsilon_{n|n-1} \varepsilon_{n|n-1}^{T} - \frac{\kappa_{n}}{m} G_{n|n-1} \right) G_{n|n-1}^{-1}.$$
(4.46)

We re-write T_n in the following block form:

$$\boldsymbol{T}_{n} = \begin{bmatrix} \boldsymbol{\Lambda}_{n} & | & \boldsymbol{I} \end{bmatrix} \begin{bmatrix} \boldsymbol{A}_{n} & | & \boldsymbol{B}_{n}^{T} \\ -- & | & -- \\ \boldsymbol{B}_{n} & | & \boldsymbol{C}_{n} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Lambda}_{n} \\ -- \\ \boldsymbol{I} \end{bmatrix} - \boldsymbol{J}_{n} \qquad (4.47)$$

where

$$\begin{aligned} A_n &= \Gamma_n^2 \\ B_n &= G_{n|n-1}^{-1} \Gamma_n^2 - \frac{\kappa_n}{2m} \\ C_n &= G_{n|n-1}^{-1} \left(\Gamma_n^2 - \epsilon_{n|n-1} \epsilon_{n|n-1}^T - \frac{\kappa_n}{m} G_{n|n-1} \right) G_{n|n-1}^{-1}. \end{aligned}$$

When A_n and C_n are positive definite, T_n is positive-semi-definite. Therefore T_n must be zero in order for (4.45) to hold as shown in the following lemma.

Lemma 4.6: Let A and B be symmetric non-singular and positive-semi-definite matrices respectively. Then $\sum [ABA] = 0$ implies that B = 0.

Proof: By congruence, B positive semi-definite implies that the product ABA is also positive-semi-definite. The trace of ABA, sum of non-negative eigenvalues, being zero implies in turn that all the eigenvalues are zero. In addition, the matrix ABA with non-negative eigenvalues is symmetric therefore it must be zero. However, A is non-singular; therefore B must be zero.

We therefore solve $T_n = 0$ for a general $\kappa_n > 0$. Note that $\kappa_n - \kappa_{n-1} = 0$ gives (4.47) a *Ricatti* equation form [3].

Definition 4.1: The operator $\mathbf{r}_{H}^{i}(\mathbf{u}^{T})$ creates a matrix identical to H except that its ith row is replaced by the vector \mathbf{u}^{T} . (This is an adaptation of the notation in [43]).

Focusing on κ_n and $h_n \kappa_n$, the following observations are made:

$$\overline{H}_{n}(*,j)\varepsilon_{n|n-1} = \det [\mathbf{r}_{H_{n}}^{j}(\varepsilon_{n|n-1}^{T})]$$

$$\varepsilon_{n|n-1}(j)\lambda_{n}(j)\overline{H}_{n}(*,j)\varepsilon_{n|n-1} = \det [\mathbf{r}_{H_{n}}^{j}(\varepsilon_{n|n-1}(j)\lambda_{n}(j)\varepsilon_{n|n-1}^{T})]$$

$$\det H_{n}\left(\gamma_{n}^{2}(j)\lambda_{n}\right) = \det \left[\mathbf{r}_{H_{n}}^{j}(\gamma_{n}^{2}(j)\lambda_{n}H_{n}(j,*))\right].$$

Since the expression for $h_n \kappa_n$ sums these determinants (from similar matrices except for one row), we may combine their determinants (see [41]) as

$$h_n \kappa_n = h_n \kappa_{n-1} + \sum_{j=1}^K \det \left[\mathbf{r}_{\boldsymbol{H}_n}^j (\boldsymbol{\gamma}_n^2(j) \boldsymbol{\lambda}_n(j) \boldsymbol{H}_n(j, *) - \boldsymbol{\varepsilon}_{n|n-1}(j) \boldsymbol{\lambda}_n(j) \boldsymbol{\varepsilon}_{n|n-1}^T) \right].$$

By defining the matrix M_n

$$M_{n} \stackrel{\text{def}}{=} \Gamma_{n}^{2} \Lambda_{n}^{2} G_{n|n-1} + \Lambda_{n} \left(\Gamma_{n}^{2} - \epsilon_{n|n-1} \epsilon_{n|n-1}^{T} \right)$$

$$= \Lambda_{n} \left[\Gamma_{n}^{2} H_{n} - \epsilon_{n|n-1} \epsilon_{n|n-1}^{T} \right],$$

$$(4.48)$$

 $h_n \kappa_n$ becomes

$$h_n \kappa_n = h_n \kappa_{n-1} + \sum_{j=1}^K \det \left[\mathbf{r}^j_{H_n}(\boldsymbol{M}_n(j, *)) \right]$$
(4.49)

which is converted into the matrix form

$$\kappa_n - \kappa_{n-1} = \operatorname{trace}\left(\boldsymbol{M}_n \boldsymbol{H}_n^{-1}\right) = \operatorname{trace}\left(\boldsymbol{H}_n^{-T} \boldsymbol{M}_n^{T}\right)$$
(4.50)

$$h_n \kappa_n - h_n \kappa_{n-1} = \operatorname{trace} (\boldsymbol{M}_n \overline{\boldsymbol{H}}_n^T) = \operatorname{trace} (\overline{\boldsymbol{H}}_n \boldsymbol{M}_n^T).$$
 (4.51)

By expressing M_n and J_n in block form,

$$\boldsymbol{M}_{n} = \left[\boldsymbol{\Lambda}_{n} \mid \boldsymbol{I} \right] \boldsymbol{\Lambda}_{n} \boldsymbol{G}_{n|n-1} \left[\begin{array}{c} \boldsymbol{G}_{n|n-1} \boldsymbol{A}_{n} \\ \boldsymbol{----} \\ \boldsymbol{C}_{n} + \frac{\kappa_{n-1}}{m} \boldsymbol{G}_{n|n-1}^{-1} \end{array} \right] \boldsymbol{G}_{n|n-1} \quad (4.52)$$

and
$$J_n = \operatorname{trace} \left(H_n^{-T} M_n \right) \left[\Lambda_n \mid I \right] \left[\begin{array}{c} I \\ -- \\ G_{n|n-1}^{-1} \end{array} \right].$$
 (4.53)

_

•

ł

The initial steps provided in this section for finding the optimum MW-SM-WRLS weights may provide important information for future research in MW-SM-WRLS algorithms.



(b) Time n = 10.

Figure 4.1. OBE ellipsoids resulting from the system identification of AR(2) system $y_n = -0.10y_{n-1} - 0.56y_{n-2} + \varepsilon_{n*}$ by SM-WRLS (dashed line) and MW-SM-WRLS (K=1, solid line) at times n = 8 and 10. The star (*) represents the "true" parameter and the circles (\circ) the central estimators (superimposed). The underlying polytopes (exact feasible sets) are also shown.

Table 4.5. First 30 weights assigned by SM-WRLS and MW-SM-WRLS (K = 1) algorithms in the identification of the AR(3) system $y_n = 0.49y_{n-1} + 0.61y_{n-2} + 0.58y_{n-3} + \epsilon_{n*}$, where the "true" measurement error sequence $\{\epsilon_{n*}\}$ is uniformly distributed over the interval (-1, +1). The SM-WRLS and MW-SM-WRLS (K = 1) algorithms selected 51 and 45 observations, respectively, from a total of 100.

	SM-WRLS	MW-SM-WRLS (K=1			
Time, <i>n</i>	$q_{n,n}$	$q_{n,n}$	$q_{n,n-1}$		
×10 ¹	×10 ¹	×10 ¹	×10 ¹		
4	9.994	-	-		
5	14.912	0.857	1.437		
6	22.302	10.541	0.857		
7	24.355	14.597	10.541		
8	21.644	16.438	14.597		
9	28.240	20.155	16.438		
10	23.885	21.241	20.155		
11	24.891	7.621	15.312		
12	-	-	7.621		
14	46.303	25.753	-		
15	26.187	17.355	25.753		
16	19.828	18.332	17.355		
17	12.770	5.007	18.332		
18	24.471	4.723	5.007		
19	2.471	-	4.723		
20	3.639	-	-		
24	41.686	15.696	8.232		
25	43.606	17.851	15.696		
26	-	-	17.851		
27	4.092	6.023	-		
28	0.332	-	6.023		
29	47.938	14.348	-		
30	13.976	7.815	14.348		
31	-	-	7.815		
32	12.784	6.149	-		
33	-	-	6.149		
34	12.027	3.715	-		
35	0.607	-	3.715		
39	32.757	8.178	-		
40	22.735	6.752	8.178		



Figure 4.2. Convergence of the parameter estimator $\theta_n(2)$ to the "true" parameter $\theta_{n*}(2) = 0.61$ and associated ellipsoid volume in the system identification of $y_n = 0.49y_{n-1} + 0.61y_{n-2} + 0.58y_{n-3} + \varepsilon_{n*}$ by SM-WRLS and MW-SM-WRLS (K=1) as in Table 4.5.



Figure 4.3. κ_n and a posteriori error $\varepsilon_{n|n}$ associated with the identification of the system of Figure 4.2.

Chapter 5

Alternate MW-OBE Algorithms

5.1 Introduction

The general MW-OBE algorithm introduced in Chapter 2 does not place any constraints on the selection of observations targeted for weight adjustments (corresponding to distinct observation vectors), nor on how many times a weight may be adjusted. This algorithm was presented in the context of weight adjustments acting on past observation vectors because this is how the algorithm was originally conceived and developed. In fact, MW-OBE algorithms can be structured to target any set of observation vectors for weight adjustments, as long as the cumulative weight on each observation vector remains positive. In this chapter we identify and study two broad classes of the MW-OBE algorithms. First, we make the distinction between the "forward-looking" MW-OBE algorithms, which solely act on "future" observation vectors, based on a "current" covariance matrix, and the "backward-looking" MW-OBE algorithms, which make adjustments of only past observation vectors. It is the latter category that has thus far been the subject of this work. We establish a duality between the two classes of MW-OBE algorithms and use this duality to simplify the proof of Theorem 3.5. Secondly, we focus on MW-OBE algorithms that only modify non-zero weights associated with successive overlapping blocks of observation vectors (overlap of exactly one observation vector) and generate a simplified MW-OBE algorithm.

5.2 Forward-looking and backward-looking MW-OBE

MW-OBE algorithms may be constructed as "forward-looking" or "backwardlooking." At time n, backward-looking MW-OBE algorithms adjust K selected previous weights (in addition to the current weight), based on the current observation matrix X_n and the covariance matrix of time n - 1. The backward-looking algorithm, although not labeled as such when presented in Chapter 2, is restricted to the modification of past weights.

The forward-looking algorithm acts on the "same" observation matrix X_n , but considers it to represent the "future" based on a "current" covariance matrix at time n - K - 1. The relationship between the forward-looking and backward-looking algorithms (see Figure 5.1) has some similarities with the relationship introduced by Deller *et al.* in [15] concerning the weighting strategy in conventional OBE algorithms. The labels "forward-looking" and "backward-looking" are references to the time relationships between the covariance matrix and the "current" observation matrix.



Figure 5.1. Forward-looking and backward-looking MW-OBE algorithms' duality.

We now develop the different recursions associated with these two MW-OBE algorithms. We shall see that these two approaches lead to formally different algorithms, but generate exactly the same sequence of permanent (composite) weights $\{q_{n,n}\}$.

Recursions (2.5), (2.8) and (2.9) are expressed as functions of Λ_n , but can be re-written as functions of $Q_{n,n} = \Lambda_n + Q_{n-1,n}$, where $Q_{n-1,n}$ is a matrix of known past weights at time n. This is achieved by writing the covariance matrix C_{n-1} in

the form (2.2),

$$\boldsymbol{C}_{n-1} = \tilde{\boldsymbol{C}}_{n_{\mathcal{K}}} + \boldsymbol{X}_{n} \boldsymbol{Q}_{n,n-1} \boldsymbol{X}_{n}^{T}, \qquad (5.1)$$

where $n_K \stackrel{\text{def}}{=} n - K - 1$ and $\tilde{C}_{n_K} \stackrel{\text{def}}{=} \sum_{t=1}^{n_K} q_{t,n} \boldsymbol{x}_t \boldsymbol{x}_t^T$ is a positive definite matrix⁴. Applying the matrix inversion lemma to (5.1), we obtain

$$P_{n-1} = \tilde{P}_{n_{\kappa}} - \tilde{P}_{n_{\kappa}} X_n \left[Q_{n,n-1}^{-1} + \tilde{G}_{n|n_{\kappa}} \right]^{-1} X_n^T \tilde{P}_{n_{\kappa}}$$
(5.2)

where $\tilde{P}_{n_{\kappa}} \stackrel{\text{def}}{=} \tilde{C}_{n_{\kappa}}^{-1}$ and $\tilde{G}_{n|n_{\kappa}} \stackrel{\text{def}}{=} X_{n}^{T} \tilde{P}_{n_{\kappa}} X_{n}$ (positive definite similar to $G_{n|n-1}$ in Section 2.3.2). Pre- and post-multiplying the left and right sides of (5.2) by X_{n}^{T} and X_{n} respectively yields

$$\begin{aligned}
G_{n|n-1} &= \tilde{G}_{n|n_{K}} - \tilde{G}_{n|n_{K}} \left[Q_{n,n-1}^{-1} + \tilde{G}_{n|n_{K}} \right]^{-1} \tilde{G}_{n|n_{K}} \\
&= \tilde{G}_{n|n_{K}} \left(I - \left[Q_{n,n-1}^{-1} + \tilde{G}_{n|n_{K}} \right]^{-1} \bar{G}_{n|n_{K}} \right) \\
&= \tilde{G}_{n|n_{K}} \left[Q_{n,n-1}^{-1} + \tilde{G}_{n|n_{K}} \right]^{-1} Q_{n,n-1}^{-1} \\
&= \left[\tilde{G}_{n|n_{K}}^{-1} + Q_{n,n-1} \right]^{-1},
\end{aligned}$$
(5.3)

an important relation between $G_{n|n-1}$ and $\tilde{G}_{n|n_{\kappa}}^{-1}$ which will be needed later.

The recursion for the inverse covariance matrix $P_{n_{\kappa}+1}$ is found similarly to (5.2),

$$\boldsymbol{P}_{n_{K}+1} = \boldsymbol{\tilde{P}}_{n_{K}} - \boldsymbol{\tilde{P}}_{n_{K}} \boldsymbol{X}_{n} \left[\boldsymbol{Q}_{n,n}^{-1} + \boldsymbol{\tilde{G}}_{n|n_{K}} \right]^{-1} \boldsymbol{X}_{n}^{T} \boldsymbol{\tilde{P}}_{n_{K}}.$$
(5.4)

The central parameter estimator $\bar{\theta}_{n_{\kappa}}$ and the scalar $\bar{\kappa}_{n_{\kappa}}$ related to the covariance matrix (5.1) become

$$\tilde{\boldsymbol{\theta}}_{n_{\kappa}+1} = \tilde{\boldsymbol{\theta}}_{n_{\kappa}} + \tilde{\boldsymbol{P}}_{n_{\kappa}+1} \boldsymbol{X}_{n} \boldsymbol{Q}_{n,n} \tilde{\boldsymbol{\varepsilon}}_{n|n_{\kappa}}$$
(5.5)

$$\tilde{\kappa}_{n_{K}+1} = \tilde{\kappa}_{n_{K}} + \boldsymbol{\gamma}_{n}^{T} \boldsymbol{Q}_{n,n} \boldsymbol{\gamma}_{n} - \tilde{\boldsymbol{\varepsilon}}_{n|n_{K}}^{T} \left[\boldsymbol{Q}_{n,n}^{-1} + \tilde{\boldsymbol{G}}_{n|n_{K}} \right]^{-1} \tilde{\boldsymbol{\varepsilon}}_{n|n_{K}}, \qquad (5.6)$$

where $\tilde{\boldsymbol{\epsilon}}_{n|n_{\kappa}} = \boldsymbol{y}_n - \boldsymbol{X}_n^T \tilde{\boldsymbol{\theta}}_{n_{\kappa}}$. Recursions (5.4), (5.5) and (5.6) form the forward MW-OBE algorithm. These are the counterparts to (2.5), (2.8) and (2.9) in the

⁴Note the significant difference between C_{n_K} and \tilde{C}_{n_K} .

Table 5.1. The forward MW-OBE algorithm.

I. Initialization: 1. $\tilde{\boldsymbol{\theta}}_0 = \boldsymbol{0}, \, \tilde{\kappa}_0 = 1 \text{ and } \, \tilde{\boldsymbol{P}}_0 = \frac{1}{\mu} \boldsymbol{I}, \, \mu \text{ small.}$ 2. $q_0 = 0$. **II.** Recursion: For l = 1, 2, ...n = l + KForm $X_n = X_{l+K}$ matrix from future K + 1 observation vectors along with corresponding y_n and γ_n vectors. $\tilde{\boldsymbol{\varepsilon}}_{n|n_{K}} = \boldsymbol{y}_{n} - \boldsymbol{X}_{n}^{T} \tilde{\boldsymbol{\theta}}_{n_{K}}$ $R_n = \tilde{P}_{n\kappa} X_n$ and $\tilde{G}_{n|n\kappa} = X_n^T R_n$ If the current and past K observations are innovative, determine optimal weight vector, $q_{n,n}$ (optimization criterion dependent). Otherwise, next n. $\boldsymbol{K}_n = \boldsymbol{R}_n [\boldsymbol{\Lambda}_n^{-1} + \tilde{\boldsymbol{G}}_{n|n_K}]^{-1}$ $\tilde{\boldsymbol{P}}_{n_{K}+1} = \tilde{\boldsymbol{P}}_{n_{K}} - \boldsymbol{K}_{n}\boldsymbol{R}_{n}^{T}$ $\boldsymbol{\theta}_{n_{K}+1} = \boldsymbol{\theta}_{n_{K}} + \boldsymbol{K}_{n}\tilde{\boldsymbol{\varepsilon}}_{n|n_{K}}$ $\kappa_{n_{K}+1} = \kappa_{n_{K}} + \gamma_{n}^{T} Q_{n,n} \gamma_{n} - \tilde{\varepsilon}_{n|n_{K}}^{T} [Q_{n,n}^{-1} + \tilde{G}_{n|n_{K}}]^{-1} \tilde{\varepsilon}_{n|n_{K}} \text{ (when necessary)}$ Next n.

backward algorithm (Section 2.5). The forward MW-OBE algorithm is summarized in Table 5.1.

At a given time n, the forward and backward forms of MW-OBE algorithms generate an identical weight adjustment vector $\lambda_{n,n}$. The forward-looking algorithm generates the weights $q_{n,n}$ as opposed to weight adjustments λ_n generated by backwardlooking MW-OBE. The quantities are related by $q_{n,n} = q_{n,n-1} + \lambda_n$, where $q_{n,n-1}$ is known. Thus, the composite weight vector $q_{n,n}$ applied to the observation matrix X_n at time n is the same whether generated by the MW-OBE forward or backward algorithm. The forward and backward forms of MW-OBE algorithms are therefore equivalent.

The following theorem, broadens the idea of MW-OBE algorithm equivalence by showing that any MW-OBE optimization (in particular MW-QOBE or MW-SM-WRLS developed in this research) may be implemented using positive weights.

Theorem 5.1 Let $C_n = C_{n-1} + X_n \Lambda_n X_n^T$. There exists a covariance matrix \tilde{C}_{n-1} and a weight matrix $\tilde{\Lambda}_n$ with positive elements such that $C_n = \tilde{C}_{n-1} + X_n \tilde{\Lambda}_n X_n^T$. **Proof:** Let $S_n = \{i_1, \dots, i_{K+1}\}$ represent the indices of observation vectors included in X_n . We write C_n as

$$C_{n} = C_{n-1} + X_{n}\Lambda_{n}X_{n}^{T}$$

$$= \sum_{t \notin S_{n}} q_{t,n-1}x_{t}x_{t}^{T} + \sum_{t \in S_{n}} q_{t,n-1}x_{t}x_{t}^{T} + X_{n}\Lambda_{n}X_{n}^{T}$$

$$= \tilde{C}_{n} + X_{n}\tilde{\Lambda}_{n}X_{n}^{T}$$
(5.7)

where $\bar{C}_n = \sum_{t \notin S_n} q_{t,n} x_t x_t^T$, a symmetric positive definite matrix, and $\tilde{\Lambda}_n = \sum_{t \in S_n} q_{t,n} x_t x_t^T + X_n \Lambda_n X_n^T$, with $\tilde{\Lambda}_n$ containing all positive weights (by definition). Note that $q_{t,n-1} = q_{t,n}$ for $t \notin S_n$.

These equivalent MW-OBE recursions are helpful in simplifying the analysis of MW-QOBE weight adjustments employed in the "backward" class. By inserting (5.3) into (3.11), the weight adjustment vector becomes

$$\lambda_{n} = \Gamma_{n}^{-1} S_{n} G_{n|n-1}^{-1} (\varepsilon_{n|n-1} - S_{n} \gamma_{n})$$

$$= \Gamma_{n}^{-1} S_{n} \left[\tilde{G}_{n|n_{\kappa}}^{-1} + Q_{n,n-1} \right] (\varepsilon_{n|n-1} - S_{n} \gamma_{n})$$

$$= \Gamma_{n}^{-1} S_{n} \tilde{G}_{n|n_{\kappa}}^{-1} (\varepsilon_{n|n-1} - S_{n} \gamma_{n}) + \Gamma_{n}^{-1} S_{n} Q_{n,n-1} \varepsilon_{n|n-1} - \Gamma_{n}^{-1} S_{n} Q_{n,n-1} S_{n} \gamma_{n}$$

$$= \Gamma_{n}^{-1} S_{n} \tilde{G}_{n|n_{\kappa}}^{-1} \left[\left(I + \tilde{G}_{n|n_{\kappa}} Q_{n,n-1} \right) \varepsilon_{n|n-1} - S_{n} \gamma_{n} \right] - \Gamma_{n}^{-1} Q_{n,n-1} \gamma_{n}$$

$$= \Gamma_{n}^{-1} S_{n} \tilde{G}_{n|n_{\kappa}}^{-1} \left[\left(I + \tilde{G}_{n|n_{\kappa}} Q_{n,n-1} \right) \varepsilon_{n|n-1} - S_{n} \gamma_{n} \right] - \Gamma_{n}^{-1} Q_{n,n-1} \gamma_{n}$$

$$= \Gamma_{n}^{-1} S_{n} \tilde{G}_{n|n_{\kappa}}^{-1} \left[\left(I + \tilde{G}_{n|n_{\kappa}} Q_{n,n-1} \right) \varepsilon_{n|n-1} - S_{n} \gamma_{n} \right] - q_{n,n-1}. \quad (5.8)$$

Substituting (5.8) into (3.19) we obtain the inequality

$$\boldsymbol{S}_{\boldsymbol{n}} \boldsymbol{\tilde{G}}_{\boldsymbol{n}|\boldsymbol{n}_{\boldsymbol{K}}}^{-1} \left[\left(\boldsymbol{I} + \boldsymbol{\tilde{G}}_{\boldsymbol{n}|\boldsymbol{n}_{\boldsymbol{K}}} \boldsymbol{Q}_{\boldsymbol{n},\boldsymbol{n}-1} \right) \boldsymbol{\varepsilon}_{\boldsymbol{n}|\boldsymbol{n}-1} - \boldsymbol{S}_{\boldsymbol{n}} \boldsymbol{\gamma}_{\boldsymbol{n}} \right] > 0.$$

$$(5.9)$$

It can be shown that

$$\tilde{\boldsymbol{\varepsilon}}_{n|n_{K}} = \left(\boldsymbol{I} + \tilde{\boldsymbol{G}}_{n|n_{K}} \boldsymbol{Q}_{n,n-1}\right) \boldsymbol{\varepsilon}_{n|n-1}$$
(5.10)

and therefore (3.19) and (5.9) are equivalent to

$$\boldsymbol{S}_{n} \boldsymbol{\tilde{G}}_{n|n_{\kappa}}^{-1} \left(\boldsymbol{\tilde{\varepsilon}}_{n|n-1} - \boldsymbol{S}_{n} \boldsymbol{\gamma}_{n} \right) > 0.$$
(5.11)

Inequality (5.11) is used in the existence and uniqueness proofs of Section 3.3.

5.3 One-step MW-OBE

At each iteration, a MW-OBE algorithm adjusts a set of K weights. These weights may correspond to successive data vectors in time, past non-zero weights, or any other criterion. When the indices of the weights targeted for revision at time n are the same those at time n - 1 with the n - Kth index discarded and a new index appended, we shall refer to this weight as "shifted" by one (from the weight at time n - 1.) The indices of the revised (scalar) weights at time n - 1

$$\mathcal{I}_{n-1} = \{i_{n-1,1}, i_{n-1,2}, \cdots, i_{n-1,K+1}, i_{n-1,K+1}\}$$

are "shifted" to

$$\mathcal{I}_n = \{i_{n-1,2}, i_{n-1,3}, \cdots, i_{n-1,K+1}, i_{n,K+1}\}.$$

When an algorithm shifts indices at each iteration, it computes the last adjustment for the weight indexed $\mathcal{I}_n(1)$. Therefore the $\lambda_{n,\mathcal{I}_n(1)}$ becomes the final additive adjustment to the weight at index $\mathcal{I}_n(1)$. This weight remains permanent but the other K weights will be recomputed. Therefore, the algorithm may be simplified by updating the covariance matrix based on the single permanent weight and discarding the others (since they will be recomputed in future iterations.) The update of the covariance takes on the form of a conventional OBE algorithm.

We therefore use the much simpler OBE recursions (QOBE or SM-WRLS algorithms) [14], in conjunction with an MW-OBE weight strategy. Such a MW-OBE algorithm is shown Table 5.2. The corresponding MW-SM-WRLS and MW-QOBE algorithms are identical to the SM-WRLS and QOBE algorithms [15, 39] aside from the weight computations. We refer to this MW-OBE algorithm as "one-step" (restricted to the overlap described earlier.) Table 5.2. The one-step MW-OBE algorithm (overlapping weight adjustment blocks applied to non-zero past weights.)

I. Initialization: 1. $\tilde{\boldsymbol{\theta}}_0 = \boldsymbol{0}, \, \tilde{\kappa}_0 = 1 \text{ and } \tilde{\boldsymbol{P}}_0 = \frac{1}{\mu} \boldsymbol{I}, \, \mu \text{ small.}$ 2. $q_0 = 0$. **II. Recursion:** For n = K + 1, K + 2, ...l = n - KForm X_n matrix from present and chosen past K observation vectors along with corresponding \boldsymbol{y}_n and $\boldsymbol{\gamma}_n$ vectors. $\tilde{\boldsymbol{\varepsilon}}_{n|n_{\mathcal{K}}} = \boldsymbol{y}_n - \boldsymbol{X}_n^T \tilde{\boldsymbol{\theta}}_{n_{\mathcal{K}}}$ $\boldsymbol{R}_n = \boldsymbol{\tilde{P}}_{n_K} \boldsymbol{X}_n$ and $\boldsymbol{\tilde{G}}_{n|n_K} = \boldsymbol{X}_n^T \boldsymbol{R}_n$ If the current and and (other) chosen K observations are innovative, determine optimal scalar weight, $q_{n,n}$ (optimization criterion dependent). Otherwise, next n. $\boldsymbol{P}_{n_{K}+1} = \boldsymbol{P}_{n_{K}} - q_{n_{K}+1,n_{K}+1} / (1 + q_{n_{K}+1,n_{K}+1} \boldsymbol{G}_{n_{K}+1}) \boldsymbol{P}_{n_{K}} \boldsymbol{x}_{n_{K}+1} \boldsymbol{x}_{n_{K}+1}^{T} \boldsymbol{P}_{n_{K}}$ $\boldsymbol{\theta}_{n_{K}+1} = \boldsymbol{\theta}_{n_{K}} + (q_{n_{K}+1,n_{K}+1}\varepsilon_{n_{K}+1|n_{K}})\bar{\boldsymbol{P}}_{n-1}\boldsymbol{x}_{n_{K}+1}$ $\kappa_{n_{K}+1} = \kappa_{n_{K}} + q_{n_{K}+1,n_{K}+1} \gamma_{n_{K}+1}^{2} - (q_{n_{K}+1,n_{K}+1}\varepsilon_{n_{K}+1|n_{K}})/(1 + q_{n_{K}+1,n_{K}+1}G_{n_{K}+1})$ (when necessary) Next n.

5.4 Proof of Theorem 3.5

In Chapter 3, Theorem 3.5 states that the sequence of ellipsoid generated by MW-OBE algorithms is decreasing. Proving that det $(\kappa_n P_n) \leq \det(\kappa_{n-1} P_{n-1})$ is circumvented by the following proof.

Proof of Theorem 3.5: The backward MW-OBE algorithm described in Chapter 2 is equivalent to a one-step MW-OBE algorithm and therefore generates a single single-positive-weight per recursions. We apply results from the single weight case [11] to prove the theorem. Alternatively, we may apply (2.18) with K = 0 and a single positive weight to derive the same result.

5.5 Example contrasting forward-looking, backward looking, and one-step MW-OBE algorithms

The equivalence of the forward and backward MW-QOBE algorithm is demonstrated in Table 5.5. The AR(2) system,

$$y_n = -0.10y_{n-1} - 0.56y_{n-2} + \varepsilon_{n*},$$

where ε_{n*} is uniformly distributed in (-1, 1) is identified by forward-looking and backward-looking MW-QOBE algorithms. As expected, the weights assigned are equal at each time in both algorithms. The same system is also identified by the onestep MW-QOBE algorithm of Table 5.2 and compared with the backward MW-QOBE algorithm. Table 5.5 shows that the sequence $\{q_{n,n}\}$ is the same in the backward, forward, and one-step MW-QOBE algorithm.

Table 5.3. Equivalence of the forward and backward MW-QOBE algorithms, and weight assignments in the one-step MW-QOBE. First eight weights assigned in the identification of the AR(2) system $y_n = -0.10y_{n-1} - 0.56y_{n-2} + \varepsilon_{n*}$, where the "true" measurement error sequence $\{\varepsilon_{n*}\}$ is uniformly distributed and bounded by 1. MW-QOBE (backward), MW-QOBE (forward) and one-step MW-QOBE used the same number of points.

	MW-QC	OBE (backward)	MW-QC	OBE (forward)	MW-QOBE (one-step)			
Time,n	$q_{n,n}(1)$	$\boldsymbol{q}_{n,n}(2)$	$q_{n,n}(1)$	$\boldsymbol{q}_{n,n}(2)$	$q_{n,n}(1)$	$\boldsymbol{q}_{n,n}(2)$		
	$\times 10^{-4}$	$\times 10^{-4}$	×10 ⁻⁴	×10 ⁻⁴	×10 ⁻⁴	×10 ⁻⁴		
12	15.671	10.908	15.671	10.908	-	10.908		
13	18.316	11.931	18.316	11.931	-	11.931		
14	20.143	23.546	20.143	23.546	-	23.546		
15	-	20.143	-	20.143	-	20.143		
20	0.246	0.253	0.246	0.253	-	0.253		
21	-	0.246	-	0.246	-	0.246		
40	-	-	-	-	-	-		
100	0.002	0.025	0.002	0.025	-	0.025		

6.1

la this

F

tbrough algorith.r MW-Q()

vergence. Percenta₋

possible.

ia previou

6.2 G

We first by Popular id, The SM Utt consid,

 $\theta_n \rightarrow \theta_{n-in}$

justificati Lev

The As a s

Chapter 6

Practical Benefits of MW-OBE Algorithms

6.1 Introduction

In this chapter, we demonstrate the practical benefits of MW-QOBE algorithms through simulations. QOBE and SM-WRLS are briefly compared to LMS and RLS algorithms (for reference) and subsequently to their multiple weight upgrade versions, MW-QOBE and MW-SM-WRLS. The performance benchmarks are parameter convergence, tracking, and data usage. By "data usage," we refer to the number (or percentage) of observations used by the algorithm to update its estimator. Whenever possible, the simulations are generated using observation data similar to that found in previous work ([13, 14, 34]) in order to facilitate comparison.

6.2 QOBE, SM-WRLS, RLS and LMS

We first briefly compare QOBE and SM-WRLS algorithms to each other, and to the popular identification algorithms, LMS and RLS, in order to establish benchmarks.

The SM-WRLS algorithm optimization minimizes the volume of $\bar{\Omega}_n$, without direct consideration of the "distance" between the "true" parameter vector $\boldsymbol{\theta}_{\star}$ and the central parameter vector estimator $\boldsymbol{\theta}_n$. Under proper data conditions (see Chapter 1), $\boldsymbol{\theta}_n \to \boldsymbol{\theta}_{\star}$ asymptotically as the set $\bar{\Omega}_n$ converges to a point [15]. QOBE algorithms, however, reduce (not minimize) both the "distance" between $\boldsymbol{\theta}_n$ and $\boldsymbol{\theta}_{\star}$ and the volume. As a result, the excellent parameter convergence of QOBE algorithms is often accompanied by large ellipsoid volumes. The large terminal volume in QOBE algorithms is often because the algorithm becomes extremely selective of data as its estimator approaches the true parameter vector [12]. Therefore, the QOBE feasible solution set, $\bar{\Omega}_n$, tends to be larger than its SM-WRLS counterpart.

Both QOBE and SM-WRLS algorithms exhibit performance superior to LMS and RLS in parameter convergence and adaptation when the error bound sequence is known (or closely approximated). This is partly due to the discriminatory selection of data points by QOBE and SM-WRLS algorithms which does not exist in RLS and LMS. The choice of pertinent observation vectors in QOBE and SM-WRLS also leads to faster convergence to the "true" parameter vector, increases robustness to additive noise, and improves performance in colored noise [15].

We illustrate performance differences among QOBE, SM-WRLS, LMS and RLS through the following simulation. The AR(2) system

$$y_n = -0.104y_{n-1} + 0.562y_{n-2} + \varepsilon_{n*}, \tag{6.1}$$

where ε_{n} , is uniformly distributed over (-1, 1), is identified by the QOBE, SM-WRLS, LMS and RLS. Figure 6.1 shows the superior performance of QOBE and SM-WRLS over RLS and LMS in parameter convergence. QOBE and SM-WRLS use small percentages of the available data (approximately 2 and 10 percent of the data, respectively.) In Figure 6.2, we illustrate important differences between the QOBE and SM-WRLS algorithms. SM-WRLS generates a much smaller ellipsoid, with a less "stable" convergence, whereas the QOBE, retains a large volume, uses fewer points and exhibits fast parameter estimator convergence to the "true" parameter.





Figure 6.1. Parameter $\theta_{\star}(2) = 0.562$ convergence of QOBE and SM-WRLS vs. that of LMS and RLS in the identification of the AR(2) system $y_n = -0.104y_{n-1}+0.562y_{n-2}+\varepsilon_{n\star}$, where $\varepsilon_{n\star}$ is uniformly distributed over (-1,1). The QOBE and SM-WRLS algorithms used 22 and 121 points, respectively, from a total of 1200 points.



(b) Volume.

Figure 6.2. Parameter $\theta_n(2) = 0.562$ convergence and volumes of QOBE and SM-WRLS algorithms in the identification of the AR(2) system $y_n = -0.104y_{n-1} + 0.562y_{n-2} + \varepsilon_{n*}$, where ε_{n*} is uniformly distributed over (-1,1). The QOBE and SM-WRLS algorithms used 22 and 121 points respectively from a total of 1200 points.

6.3 h th: similar work vector input 1 SY: ious ty: ramete: vector 6 parame. and non 6.4 In this s visited in We consi mputin due to the new infor addition t. SM-WRL ^{to t}conven This fir ^{xent} in th Convergence ¹⁵anown ar baie Berno The rela ¹⁰¹ Well-dei

6.3 Signal cataloging

In this section, we catalog the signals used for the simulation. This cataloging is similar to Lin's [34] in order to facilitate comparison with the results of previous work (e.g., [14, 34, 35]). The systems to be identified are described by a parameter vector (or sequence of parameter vectors in the time-varying case) (Table 6.1) and an input noise type (Table 6.2.) The systems are cataloged in Table 6.2.

SYSTEM 1 through SYSTEM 8 have time-invariant parameter vectors with various types of input noises. SYSTEM 9 through SYSTEM 12 are time-varying parameter vectors; SYSTEM 9 and SYSTEM 10 have gradual variations in parameter vector $\boldsymbol{\theta}_{\cdot n}$ (sine wave), while SYSTEM 11 and SYSTEM 12 have abruptly changing parameters. The excitation noises used are Bernoulli, uniform and colored, with zero and non-zero means.

6.4 Parameter convergence and volume

In this section, we show by simulation that increasing the number of weights revisited improves parameter convergence speed, and further reduces ellipsoid volume. We considerer the case of K = 1 and adjust a single past weight in addition to computing the current data vector weight. We choose to adjust non-zero past weights due to their obvious past relevance (although this choice does not guarantee current new information.) At each iteration, we attempt to adjust one previous weight in addition to computing a present weight but fall back on the single weight algorithm (SM-WRLS or QOBE) when this such attempt fails. Recall that K = 0 corresponds to "conventional" QOBE or SM-WRLS.

This first series of simulations compares the benefits of an extra weight adjustment in the identification of SYSTEM 1 through SYSTEM 8 in terms of parameter convergence speed and volume reduction. We consider the "true" parameter vectors unknown and identify them using MW-OBE algorithms. SYSTEM 1 and SYSTEM 3 have Bernoulli distributed noise sequences.

The relationship between κ_n minimization and the volume of the hyperellipsoid is not well-defined. It can be inferred from recent work, however, that κ_n minimization

Table 6 both til
remain (
La
E
C
D
E
1
r C
6
ĺ

Table 6.1. Parameter vector for the AR systems used in the simulations of Chapter 6, both time-invariant (TI) and time-varying. The operator '\' denotes the division remainder.

Label	Model Order	"True parameter vector", $\boldsymbol{\theta}_{n*}^{T}$	Туре
А	2	$[-0.1, 0.56]^T$	TI
В	2	$\left[\begin{array}{c} 1.6\cos\frac{2\pi n}{600}\\ -0.68\end{array}\right]$	sine wave (slow)
С	2	$\left[\begin{array}{c} 1.6\cos\frac{2\pi n}{300}\\ -0.68 \end{array}\right]$	sine wave (fast)
D	2	$\theta_{*n}(1) = \begin{cases} 1.6, & (n \setminus 600) + 300 < 300 \\ -1.6, & \text{otherwise} \end{cases}$ $\theta_{*n}(2) = -0.68$	square wave (slow)
Е	2	$\boldsymbol{\theta}_{*n}(1) = \begin{cases} 1.6, & (n \setminus 300) + 150 < 150 \\ -1.6, & \text{otherwise} \end{cases}$ $\boldsymbol{\theta}_{*n}(2) = -0.68$	square wave (fast)
F	3	$[2, -1.48, 0.34]^T$	TI
G	12	$[\begin{array}{c} 0.1, \ 0.9175, \ -0.191, \ -0.2253, \ 0.2601, \ 0.0046, \\ -0.00367, \ -0.0209, \ -0.0082, \ 0.0095, \ -0.0052, \ -0.0041 \end{array}]^T$	TI

Ta	Ы	e 6.	2.	Excitation	1 noises 1	for [·]	the	AR	. sysi	tems	used	in (the simu	lation	s of	Ċ	Chap	ter	6 .
----	---	------	----	------------	------------	------------------	-----	----	--------	------	------	------	----------	--------	------	---	------	-----	------------

Label	Distribution	"True error", ε_{n*}	Mean
А	Bernoulli, B(-1,1)	$\begin{cases} 1, & \text{with probability 0.7} \\ -1, & \text{with probability 0.3} \end{cases}$	zero
В	Bernoulli, <i>B</i> (-1, 0.5)	$\begin{cases} 1, & \text{with probability 0.7} \\ -0.5, & \text{with probability 0.3} \end{cases}$	non-zero
С	Uniform, $U(-1, 1)$		zero
D	Colored, $z_n \sim U(-1, 1)$, $w_n = -0.8w_{n-1} + z_n$	$\begin{cases} 1, & \text{if } w_n > -1 \\ -1, & \text{otherwise} \end{cases}$	non-zero
Ε	Colored, $z_n \sim U(-1, 1)$, $w_n = -0.8w_{n-1} + z_n$	$\begin{cases} 1, & \text{if } w_n > -0.5 \\ -0.5, & \text{otherwise} \end{cases}$	non-zero

Table 6.3. Matrix of systems used in the simulations of Chapter 6. See Tables 6.1 and 6.2.

$\varepsilon_{n*} \setminus \boldsymbol{\theta}_{n*}$	A	В	C	D	E	F	G
А	4	-	-	-	-	1	-
В	-	-	-	-	-	3	-
С	-	9	10	11	12	2	-
D	8	-	-	-	-	5	6
E	-	-	-	-	-	7	-

will reduce volume at each update [14, 22]. This assertion is verified through the following simulations.

6.4.1 MW-QOBE vs. QOBE

We compare MW-QOBE (K = 1) and QOBE algorithms in terms of parameter convergence and ellipsoid volume reduction. We assume the true error magnitude to be bounded by $\gamma_n = 1.1$, when is it actually bounded by unity. In general, we observe a decrease in the number of updates and an improvement in parameter convergence when adjusting a single past weight.

Figure 6.3 shows the parameter convergence of SYSTEM 1, an AR(3) system driven by a non-zero-mean Bernoulli distributed sequence. Both the QOBE and MW-QOBE perform equally well. This is due to the frequent visitation of the bound. Considering an extra weight slightly decreases the number of points taken as well as the ellipsoid volume.

The disturbance in SYSTEM 2 is uniformly distributed over (-1, 1) therefore the MW-QOBE parameter estimator θ_n is not expected to converge as fast as in SYSTEM 1 (where ε_{*n} visited the bound more frequently). We notice an improvement in MW-QOBE performance when increasing K from zero to one, in both parameter tracking and volume reduction (Figure 6.4). The volumes remain large in all the QOBE experiments due to the algorithm's tendency to stop taking points (or very stringent requirements on accepting incoming data points) once the central estimator approaches the true parameter vector.

The disturbance in SYSTEM 3 is Bernoulli distributed as in SYSTEM 1 but has asymmetric error bounds (see excitation noises labeled A and B on Table 6.1.) The resulting overestimated bound (30% of the time) decreased the performance of SYSTEM 3 (as opposed to SYSTEM 1). As in the previous case, the volume is further decreased by adjusting one past weight (Figure 6.5). In this experiment we notice a significant reduction in number of points taken.

SYSTEM 4 shows an improved performance over SYSTEM 1 because of smaller model order. The incremental improvements are minimal with increased number of weight revisions (Figure 6.6). This improvement may be due to the equality between number of weights revisited and model order.

In SYSTEM 5, the noise sequence is a filtered AR(1) signal. This is usually a difficult case of the type that causes RLS to converge to a biased estimator. MW-QOBE's performance stands out in this experiment with fast convergence, smaller volume, and significantly fewer points selected (Figure 6.7). This improvement may be due to the more stringent criteria for data acceptance in MW-QOBE, resulting in the selection of points with a higher energy concentration in the modes of the system being identified.

In SYSTEM 6, the AR(3) system of SYSTEM 5 is replaced by an AR(12) system. The result of the identification of $\theta_n(2)$ in this higher order system by QOBE and MW-QOBE is shown in Figure 6.8. The MW-QOBE algorithm shows a modest improvement over the QOBE algorithm.

In SYSTEM 7, the mean of SYSTEM 5 is moved closer to zero and therefore induces some improvement in its identification as seen in Figure 6.9.

SYSTEM 8 has the same noise sequence as SYSTEM 5, but with a smaller system order, AR(2), and therefore enjoys faster convergence and smaller volume (Figure 6.10).

In summary, the adjustment of a single past weight improves the convergence speed of the QOBE (MW-QOBE with K = 0) algorithm and further reduces the ellipsoid volume.



(b) Volume.

Figure 6.3. Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the identification of SYSTEM 1 by MW-QOBE and QOBE algorithms, which used 94 and 108 points, respectively, from a total of 200 points.

F ic a


(b) Volume.

Figure 6.4. Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the identification of SYSTEM 2 by MW-QOBE and QOBE algorithms, which used 75 and 87 points, respectively, from a total of 700 points.



(b) Volume.

Figure 6.5. Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the identification of SYSTEM 3 by MW-QOBE and QOBE algorithms, which used 100 and 82 points, respectively, from a total of 700 points.

F iri ar



(b) Volume.

Figure 6.6. Parameter $\theta_{n*}(1) = -0.10$ convergence and volumes resulting from the identification of SYSTEM 4 by MW-QOBE and QOBE algorithms, which used 11 and 42 points, respectively, from a total of 100 points.



(b) Volume.

Figure 6.7. Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the identification of SYSTEM 5 by MW-QOBE and QOBE algorithms, which used 117 and 116 points, respectively, from a total of 1500 points.



(b) Volume.

Figure 6.8. Parameter $\theta_{n*}(2) = 0.92$ convergence and volumes resulting from the identification of SYSTEM 6 by MW-QOBE and QOBE algorithms, which used 174 and 166 points, respectively, from a total of 3000 points.



(b) Volume.

Figure 6.9. Parameter $\theta_{n*}(1) = 2.00$ convergence and volumes resulting from the identification of SYSTEM 7 by MW-QOBE and QOBE algorithms, which used 214 and 203 points, respectively, from a total of 1500 points.



(b) Volume.

Figure 6.10. Parameter $\theta_{n*}(1) = -0.10$ convergence and volumes resulting from the identification of SYSTEM 8 by MW-QOBE and QOBE algorithms, which used 9 and 43 points, respectively, from a total of 500 points.

6.5 Tracking performance

In this section we show the tracking benefits of increased K in MW-OBE algorithms.

6.5.1 MW-QOBE vs. QOBE

While based in the "OBE philosophy," QOBE algorithms may be interpreted as having an update rule that is essentially independent of the current hyperellipsoid [22, 39]. This result is counter-intuitive because the optimization criterion is to minimize the parameter κ_n , a scalar that is directly related to the ellipsoid and its size. Nevertheless, the alternative interpretation reveals why QOBE is strongly robust to parameter variation and highly proficient at adaptive estimation.

In the present work, we use faster systems than those in [14, 34] in order to highlight both the superior tracking performance of QOBE algorithm and the incremental benefits of higher-order weight adjustments in MW-QOBE.

SYSTEM 9 represents a time-varying system in which one of the parameters changes gradually from -1.6 to 1.6 every 400 points. The noise sequence is uniformally distributed in (-1, 1). We observe good tracking by both MW-QOBE and QOBE (Figure 6.11).

In SYSTEM 10 the rate of change of parameter $\theta_{n*}(1)$ is increased compared with that of SYSTEM 9. The noise sequence is the as in SYSTEM 9 but parameter $\theta_{n*}(1)$ is made to change twice as fast. Nevertheless the parameter estimators of both QOBE and MW-QOBE stay on track most of the time (Figure 6.12). MW-QOBE uses fewer points to achieve the same (or better) tracking performance. The final ellipsoid (volume) is very small due to the relatively large number of points taken by MW-QOBE.

SYSTEM 11 represents an abruptly changing system, with parameter $\theta_{n*}(1)$ switching in one sample from -1.6 to 1.6 (see Tables 6.1-6.3). The tracking performance can be observed at switching times. Again the benefit of an additional weight in the optimization process is evident (Figure 6.13).

As in SYSTEM 11, SYSTEM 12 represents a fast switching system with abrupt changes in parameter vector. The MW-QOBE superior tracking is shown in Figure 6.14.

The simulations presented in this section show that MW-QOBE (K = 1) has a tracking performance at least as good as QOBE while using fewer observations.



(b) Volume.

Figure 6.11. Parameter $\theta_{n*}(1)$ tracking and volumes resulting from the identification of SYSTEM 9 by MW-QOBE and QOBE algorithms, which used 109 and 110 points, respectively, from a total of 700 points.



(b) Volume.

Figure 6.12. Parameter $\theta_{n*}(1)$ tracking and volumes resulting from the identification of SYSTEM 10 by MW-QOBE and QOBE algorithms, which used 70 and 72 points, respectively, from a total of 350 points.



(b) Volume.

Figure 6.13. Parameter $\theta_{n*}(1)$ tracking and volumes resulting from the identification of SYSTEM 11 by MW-QOBE and QOBE algorithms, which used 52 and 95 points, respectively, from a total of 700 points.



(b) Volume.

Figure 6.14. Parameter $\theta_{n*}(1)$ tracking and volumes resulting from the identification of SYSTEM 12 by MW-QOBE and QOBE algorithms, which used 37 and 67 points, respectively, from a total of 350 points.

6.6 Selective updating

MW-QOBE algorithms utilize fewer points than QOBE algorithms due to their more stringent data selection. The extra per-point-taken costs are often offset by the decrease in the number of observations processed. In this section, we chart the average number of data points accepted by QOBE and MW-QOBE in the identification of SYSTEM 2.

In the first experiment we identify SYSTEM 2 by MW-QOBE and QOBE using an MW-QOBE algorithm with an "adaptive" number of weights. That is, the algorithm first attempts to optimize by incorporating one past weight, but is allowed to revert to K = 0 (QOBE) if the attempt fails. We then compare the average number (over 10 runs) of points taken by MW-OBE and QOBE per length of data. Figures 6.15-6.17 show plots of the results.

The average number of updates of the central estimator in both MW-QOBE and QOBE algorithm decreases as the duration of the observation vector sequence increases. This is expected, since many of the observations are redundant after a certain time. However, the MW-QOBE selects fewer points on average. We observe performance gains in each case. Increasing the number of weights revisited results in a more stringent test for acceptance of observations but increases the per-point convergence and decreases the ellipsoid volume.

Figures 6.15 - 6.17 compare the average number of points used by MW-QOBE and QOBE in identifying AR(2), AR(3) and AR(12) [systems SYSTEM 4, SYSTEM 2 and SYSTEM 12] with error sequence uniformly distributed over (-1, 1). We observe performance gains in each case.

Next we consider an MW-QOBE algorithm which attempts to optimize κ_n over K weights but does not revert to the QOBE algorithm when the optimization over K weights fails. Figures 6.18 - 6.20 compare the average number of points utilized by MW-QOBE and QOBE in identifying the systems of the preceding experiments.

The per-update MW-QOBE algorithm computational costs (flops) for a system of order m is $(3/2)(m^2 + 3m + 4)$ for K = 0 (QOBE), and $3m^2 + 12m + 20$ for K = 1(see Chapter 3). When m = 3 the computational cost per update becomes 33 and 83 (flops). For a sequence of length 1000 (see Fig 6.19), this corresponds to an average



Figure 6.15. Ratio of points taken, averaged over 10 runs, in the identification of an AR(2) system with uniformly distributed error sequence by QOBE and MW-QOBE (adaptive K.)

of 34 and 10 updates for K = 0 and K = 1, or 112 and 83 multiplications. These numbers support the assertion that the extra per-update cost is often offset by a decrease in the number of observations processed. Using a fixed K in MW-QOBE, demonstrates its strongest asset, namely the drastic decrease in number of observation taken.

6.7 General findings, conclusions, recommendations for practice

Simulation results show improvements in parameter convergence speed, reduction in ellipsoid volume and tracking ability, when an extra weight is used in MW-OBE algorithms. Both QOBE and MW-QOBE have excellent tracking capabilities, with MW-QOBE performing at least as well as QOBE when tracking fast systems with abrupt parameter transitions. In practice, revisitation of a large number of weights becomes impractical. We recommend the use of a small number of weight revisions,



Figure 6.16. Ratio of points taken, averaged over 10 runs, in the identification of an AR(3) system with uniformly distributed error sequence by QOBE and MW-QOBE (adaptive K.)

typically one or two.

In the simulations presented, the MW-QOBE algorithm was given the option of reverting to the QOBE algorithm when no optimal solution could be found considering K weights. When this option is removed, as shown in Figures 6.18-6.20, the number of observations necessary to update the estimator is dramatically reduced. The drawback is the more stringent data acceptance criteria which may prevent the algorithm from sufficiently updating its estimator in a short interval.

We suggest the use of MW-OBE with an adaptive number of adjustments when identifying parameters in short data segments and for parameter tracking of timevarying parameters. This adaptive version (in K) guarantees maximum tracking performance. For long data sequences where the concern is to reduce the number of points processed (in updating the estimator), we recommend using an MW-OBE algorithm with fixed K.



Figure 6.17. Ratio of points taken, averaged over 10 runs, in the identification of an AR(12) system with uniformly distributed error sequence by QOBE and MW-QOBE (adaptive K.)



Figure 6.18. Ratio of points taken, averaged over 10 runs, in the identification of an AR(2) system with uniformly distributed error sequence by QOBE and MW-QOBE (fixed K.)



Figure 6.19. Ratio of points taken, averaged over 10 runs, in the identification of an AR(3) system with uniformly distributed error sequence by QOBE and MW-QOBE (fixed K.)



Figure 6.20. Ratio of points taken, averaged over 10 runs, in the identification of an AR(12) system with uniformly distributed error sequence by QOBE and MW-QOBE (fixed K.)

Chapter 7

Future Directions and Complementary Work

7.1 Introduction

This chapter contains a collection of topics that were explored as part of this research. These topics contain information and ideas potentially helpful for future research. We also summarize the results obtained throughout the course of this research and conclude the dissertation.

7.2 Normalized MW-OBE algorithm

Including a normalization strategy (as in [6]) offers new insights in the behavior of MW-OBE algorithms. This "normalization" is achieved by an (affine) transformation of the parameter estimator space, which normalizes the ellipsoid semi-axis lengths to unity and shifts its center, θ_{n-1} , to the origin. We begin by expressing the positive definite matrix P_{n-1} using its normalized eigenvector representation

$$P_{n-1} = R_{n-1}R_{n-1}^T$$
(7.1)

where \mathbf{R}_{n-1} is a unitary matrix. We define the change of variable in the parameter space (at each iteration) by

$$\overline{\boldsymbol{\theta}} \stackrel{\text{def}}{=} \boldsymbol{R}_{n-1}^{T} (\boldsymbol{\theta} - \boldsymbol{\theta}_{n-1}) \text{ or } \boldsymbol{\theta} = \boldsymbol{R}_{n-1} \overline{\boldsymbol{\theta}} + \boldsymbol{\theta}_{n-1}$$
(7.2)

and the projection of the matrix of data vectors in the parameter space spanned by R_{n-1} by

$$\boldsymbol{Z}_{\boldsymbol{n}-1} \stackrel{\text{def}}{=} \boldsymbol{R}_{\boldsymbol{n}-1}^T \boldsymbol{X}_{\boldsymbol{n}-1}. \tag{7.3}$$

It follows that $G_{n|n-1} = Z_n^T Z_n$ and $H_{n|n-1} = I + \Lambda_n Z_n^T Z_n$. The *a posteriori* ellipsoid set at time n-1 (a priori at time n) in the new coordinate system becomes

$$\bar{\Omega}_{n-1} = \left\{ \boldsymbol{R}_{n-1} \overline{\boldsymbol{\theta}} + \boldsymbol{\theta}_{n-1} : (\boldsymbol{R}_{n-1} \overline{\boldsymbol{\theta}}_{n-1}^T)^T \boldsymbol{C}_{n-1} (\boldsymbol{R}_{n-1} \overline{\boldsymbol{\theta}}_{n-1}) < \kappa_{n-1} \right\}$$

or, by multiplying each side by R_{n-1}^T (a rotation),

$$\bar{\Omega}_{n-1} = \left\{ \boldsymbol{\theta} : \bar{\boldsymbol{\theta}}^T (\boldsymbol{R}_{n-1}^T \boldsymbol{P}_{n-1} \boldsymbol{R}_{n-1})^{-1} \overline{\boldsymbol{\theta}} < \kappa_{n-1} \right\}$$

= $\left\{ \boldsymbol{\theta} : \bar{\boldsymbol{\theta}}^T \overline{\boldsymbol{\theta}} < \kappa_{n-1} \right\}.$

The MW-OBE recursions take on the new form:

$$\boldsymbol{R}_{n-1}^{T}\boldsymbol{P}_{n}\boldsymbol{R}_{n-1} = \boldsymbol{I} - \boldsymbol{Z}_{n}\boldsymbol{H}_{n}^{-1}\boldsymbol{\Lambda}_{n}\boldsymbol{Z}_{n}^{T}$$
(7.4)

$$\overline{\boldsymbol{\theta}}_{n} = \overline{\boldsymbol{\theta}}_{n-1} + \boldsymbol{Z}_{n} \boldsymbol{\Lambda}_{n} \boldsymbol{\varepsilon}_{n|n-1}$$
(7.5)

$$\kappa_n = \kappa_{n-1} + \gamma_n^T \Lambda_n \gamma_n - \varepsilon_{n|n-1}^T H_n^{-1} \Lambda_n \varepsilon_{n|n-1}.$$
(7.6)

Equations (7.4-7.6) may be used as alternatives to the MW-OBE recursions introduced in Chapter 2 and have a simpler form.

7.3 Hyperellipsoid volume

The determinant of the matrix representing a hyperellipsoid is proportional to the volume. The need to compare volumes in different dimensional spaces (models of different order) sometimes arises, leading to the need for a more exact representation of volume. Some of the following ideas are used in Section 7.4.

Lemma 7.1 Let $\boldsymbol{\theta} = [\theta_1, \theta_2, \dots, \theta_{m-1}]$ represent a vector of angles, describing a direction in m-dimensional space, and $\boldsymbol{u} = [u_1, u_2, \dots, u_m]$ is the unit vector in that direction. Without loss of generality, we assume that $0 \le \theta_1 < 2\pi$ and $-\pi/2 \le \theta_i < \pi/2$,

for $i \neq 1$. The *i*th element of **u** is expressed as follows:

$$u_i = S_i \prod_{j=i}^m C_j \tag{7.7}$$

where
$$S = \begin{bmatrix} 1 & \sin \theta_1 & \cdots & \sin \theta_{m-1} \end{bmatrix}$$
 and $C = \begin{bmatrix} \cos \theta_1 & \cdots & \cos \theta_{m-1} & 1 \end{bmatrix}$.

Proof: Start with the two-dimensional case and iterate.

Lemma 7.2 Let M be a positive definite matrix of order m describing a hyperellipsoid. The distance from the center to the surface of the hyperellipsoid in any given direction θ is given by:

$$d_{\boldsymbol{\theta}} = \left(\boldsymbol{u}^T \boldsymbol{M} \boldsymbol{u}\right)^{-\frac{1}{2}} \tag{7.8}$$

where the vector \mathbf{u} is derived as in Lemma 7.1.

Proof: Let \boldsymbol{v} be the vector from the origin to the ellipsoid in the $\boldsymbol{\theta}$ direction. Then \boldsymbol{v} may be expressed as a multiple of the vector \boldsymbol{u}^T , or $\boldsymbol{v} = \alpha \boldsymbol{u}$, where α is a scalar and \boldsymbol{u} , the unit vector in the $\boldsymbol{\theta}$ direction. From the hyperellipsoid equation $\boldsymbol{v}^T \boldsymbol{M} \boldsymbol{v} = 1$, we obtain $\boldsymbol{u}^T \boldsymbol{M} \boldsymbol{u} = 1/\alpha^2$. Equation 7.8 follows.

Lemma 7.3 The volume of an m-dimensional hypersphere of radius r is given by

$$v = \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \cdots \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \int_{0}^{2\pi} \int_{0}^{r} l \sin \theta_{1} \cos \theta_{2} \cdots \cos \theta_{m-1} \, dl \, d\theta_{1} \cdots d\theta_{m-1}$$

$$= \frac{2\pi}{m} 2^{(m-1)} r^{m}$$

$$= \frac{2^{m-1}}{m} \pi r^{m}$$
(7.9)

In Figure 7.1, we compare the effects of model order on the volume in OBE identification of a speech segment. Volume corresponding to each model order is normalized by scaling by an appropriate constant (see Lemma 7.3).

Lemmas 7.1 and 7.2 are used for generating the "other" parameter vectors inside the ellipsoid in Section 7.4. These "other" points are generated by choosing a random



Figure 7.1. Effect of the model order in the identification of the LP parameters in the voiced /I/ phoneme. We show the log of the final volumes (normalized to the initial ellipsoid volume) using the model orders 7-14.

direction inside the ellipsoid (m - 1 angle vector), determining the length of a semiaxis in the chosen direction, then choosing a random length.

7.4 Application to linear-prediction analysis of speech signals

Speech signals are often modeled as AR signals with 10-14 linear prediction (LP) parameters [9]. These parameters represent the spectral properties of a speech segment and are usually identified by a least-squares method, often in batch form. However, the least-squares estimator does not necessarily provide the best perceptual representation of the original speech sequence. OBE algorithms have interesting properties with applications to speech processing. Fast convergence and good tracking abilities of MW-QOBE make it an ideal candidate for spectral estimation in very short time frames. However, what distinguishes OBE from most system identification methods is its data selection ability and its generation of a feasible set. These two properties offer great potential for improving perceptual qualities in speech analysis.

An objective in speech analysis may be to find a parameter estimator that would enhance a certain perceptual quality. Such an estimator may be found within the ellipsoid, aside from the central estimator and the conventional LS estimators. OBE algorithms provide a set of feasible parameters from which a "better" estimator can be chosen.

In Figure 7.2a we show the spectrum of a segment of length L (L=512 points) of the vowel /I/, generated by fast Fourier transform (FFT) and assume that the speech segment is accurately modeled with an AR(14) signal with "true" parameter θ_{\bullet} . In Figure 7.2b, we show the impulse response FFT of the "covariance method" estimator ($\theta_{COV,L}$), the LSE estimator [9], the central estimator ($\theta_{c,L}$) and three "other" non-central estimators ($\theta_{1-3,L}$) of θ_{\bullet} inside the ellipsoid (see Section 7.3). These alternative estimators, with close spectra to the least-squares estimators, may offer other desirable properties in terms of perceptual quality. We suggest the use of these alternative parameters in improving speech quality.

Parameter	$\boldsymbol{\theta}_{1,L}$	$\theta_{2,L}$	$\boldsymbol{\theta}_{3,L}$		
1	0.07	-0.00	0.04		
2	0.17	0.00	-0.07		
3	-0.04	-0.00	-0.06		
4	0.27	0.00	0.05		
5	-0.16	0.03	-0.57		
6	0.13	-0.25	-6.16		
7	-0.62	-0.15	5.24		
8	0.57	2.00	15.31		
9	22.61	4.21	73.82		
10	-1.24	23.52	17.00		
11	141.98	79.60	336.44		
12	16.86	66.57	75.48		
13	329.19	547.10	-288.21		
14	-147.43	-374.22	73.76		
$\ \boldsymbol{\theta}_{i,L} - \boldsymbol{\theta}_{c,L}\ ^2$	388.66	671.34	461.98		

Table 7.1. "Other" AR parameter estimators inside the ellipsoid (Figure 7.2b). The coordinates and radii ($\|\boldsymbol{\theta}_{i,L} - \boldsymbol{\theta}_{c,L}\|^2$) are in relation to the final SM-WRLS ellipsoid center (M = 14, volume = 1.4×10^{19} and $4.67 \times 10^{-1} \leq \text{semi-axes} \leq 43.6$).

Certain challenges remain in applying OBE algorithms to speech processing. First, determining reasonable error bounds for speech signals poses a challenge, since the "true" bounds are unknown [9, 29]. Second, OBE algorithms generate large ellipsoid volumes [13]. A closer look at these ellipsoids shows a great disparity between axis lengths as shown in Table 7.2. (Tables 7.3 and 7.4 are provided for reference.) It is conjectured that this is largely due to the lack of persistency of excitation in most voiced speech data.

Lengths	43.62	35.51	17.33	13.85	13.47	5.10	4.07
	3.81	0.47	0.91	2.47	2.62	1.86	1.82

Table 7.2. SM-WRLS 14-dimensional hyperellipsoid semi-axes (see Figure 7.2). Volume = 4.32×10^{17} .

Magnitudes ($\times 10^{-1}$)	7.98	9.66	9.66	9.69	9.69	9.33	9.33
	8.45	8.45	9.86	9.86	9.35	3.83	3.83
Angles (deg)	180.00	124.03	-124.03	80.38	-80.38	95.69	-95.69
	113.81	-113.81	10.10	-10.10	0.00	27.59	-27.59

Table 7.3. Poles of the 14th order system, based on the batch covariance estimate (see Figures 7.2 and 7.3).

Magnitudes ($\times 10^{-1}$)	10.1	10.1	9.10	9.62	9.62	9.27	9.27
	9.58	9.58	8.22	7.44	7.44	2.57	1.99
Angles (deg)	9.93	-9.93	0.00	79.51	-79.51	96.54	-96.54
	123.10	-123.10	180.00	113.14	-113.14	180.00	0.00

Table 7.4. Poles of the 14th order system (SM-WRLS), based on the ellipsoid center estimate (see Figure 7.2 and 7.3).

7.5 Alternative volume minimization: Trace

Minimizing the volume can also be done by minimizing the trace of the ellipsoid matrix [15]. In this section initial steps toward finding a solution with multiple-weight adjustment are described. More work is necessary to find an actual solution to the derived system of equations. Using the fact that trace AB = trace BA, and defining $R_n \stackrel{\text{def}}{=} [P_{n-1}X_nH_n^{-1}\Lambda_nX_n^TP_{n-1}]$, we can then find the trace of

trace
$$\mathbf{R}_n$$
 = trace $\left[\mathbf{P}_{n-1}\mathbf{X}_n\mathbf{H}_n^{-1}\mathbf{\Lambda}_n\mathbf{X}_n^T\mathbf{P}_{n-1}\right]$
= trace $\left[\mathbf{H}_n^{-1}\mathbf{\Lambda}_n\mathbf{X}_n^T\mathbf{P}_{n-1}\mathbf{P}_{n-1}\mathbf{X}_n\right]$
= trace $\left[\mathbf{H}_n^{-1}\mathbf{\Lambda}_n\mathbf{B}_n\right]$

where $B_n \stackrel{\text{def}}{=} X_n^T P_{n-1}^2 X_n$. Using the formulae of Athans [1] and defining $A_n \stackrel{\text{def}}{=} \Lambda_n^{-1} H_n$, the derivative (with respect to the matrix) is found as followed:

$$\frac{\partial \operatorname{trace} \boldsymbol{R}_n}{\partial \boldsymbol{A}_n} = -\boldsymbol{A}_n^{-1} \boldsymbol{B}_n \boldsymbol{A}_n^{-1}.$$

By the chain rule, we can obtain the partial derivative with respect to a particular weight by using the fact that $\frac{\partial \operatorname{trace}(A_n)}{\partial A_n} = [\operatorname{trace} \frac{\partial A_n}{\partial a_n}]$ and

$$\frac{\partial A_n}{\partial \lambda_i} = \begin{cases} 0, & i \neq j \\ -\frac{1}{\lambda_n^2}, & i = j \end{cases}$$

We then obtain

$$\frac{\partial \operatorname{trace} \boldsymbol{R}_{n}}{\partial \lambda_{i}} = \frac{\partial \boldsymbol{R}_{n}}{\partial a_{(i,i)}} \frac{\partial a_{(i,i)}}{\partial \lambda_{i}} \\ = \left[\frac{\partial \operatorname{trace} \boldsymbol{R}_{n}}{\partial \boldsymbol{A}_{n}} \right]_{(i,i)} \frac{\partial a_{(i,i)}}{\partial \lambda_{i}}$$

which can be expressed in matrix form as

$$\frac{\partial \operatorname{trace} \boldsymbol{R}_n}{\partial \boldsymbol{\Lambda}_n} = \sum \boldsymbol{\Lambda}_n^{-2} \boldsymbol{A}_n^{-1} \boldsymbol{B}_n \boldsymbol{A}_n^{-1}.$$

The task becomes to solve the equation:

$$0 = \frac{\partial \kappa_n}{\partial \Lambda_n} \operatorname{trace} \left(\boldsymbol{P}_n \right) + \kappa_n \left[\searrow \Lambda_n^{-2} \boldsymbol{A}_n^{-1} \boldsymbol{B}_n \boldsymbol{A}_n^{-1} \right].$$
(7.10)

A solution of (7.10) minimizes the trace of the $\kappa_n P_n$ and therefore the volume of the corresponding ellipsoid.

7.6 Approximations

The recursions (2.8) require the relatively expensive computation of the term $H_n^{-1}\Lambda_n$. The product $H_n^{-1}\Lambda_n$ can be expanded as an infinite series $H_n^{-1}\Lambda_n = \sum_{i=0}^{\infty} (-1)^i (G_{n|n-1}\Lambda_n)^i$ which can then be approximated by a small number of terms.⁵ For example, $(M < \infty)$

$$\begin{split} [\Lambda_n^{-1} + G_{n|n-1}]^{-1} &= \left[\Lambda_n - \Lambda_n [G_{n|n-1}^{-1} + \Lambda_n]^{-1} \Lambda_n\right] \\ &= \left[\Lambda_n - \Lambda_n [G_{n|n-1} - G_{n|n-1} (\Lambda_n^{-1} + G_{n|n-1})^{-1} G_{n|n-1}] \Lambda_n\right] \\ &= \Lambda_n - \Lambda_n G_{n|n-1} \Lambda_n + \Lambda_n G_{n|n-1} (\Lambda_n^{-1} + G_{n|n-1})^{-1} G_{n|n-1} \Lambda_n \\ &\approx \sum_{i=0}^M (G_{n|n-1} \Lambda_n)^i. \end{split}$$

Similarly,

$$H_n^{-1} = \sum_{i=0}^{\infty} (-1)^{-i} (\Lambda_n G_{n|n-1})^{-i}.$$

Approximating each polynomial with the first two terms:

$$[\Lambda_n^{-1} + G_{n|n-1}]^{-1} \approx \Lambda_n - \Lambda_n [G_{n|n-1}^{-1} + \Lambda_n]^{-1} \Lambda_n$$
(7.11)

$$\boldsymbol{H}_{n}^{-1} \approx \boldsymbol{I} - \boldsymbol{\Lambda}_{n} \boldsymbol{G}_{n|n-1}, \qquad (7.12)$$

⁵The primary purpose of this development is not to reduce the computational complexity of the inverse, since K is usually small (1 to 3), but rather to express κ_n and the volume in a way that facilitates the derivation of an approximate closed-form solution.

from which we obtain:

$$P_n \approx P_{n-1} - P_{n-1}X_n\Lambda_nX_n^TP_{n-1} - P_{n-1}X_n\Lambda_nG_{n|n-1}\Lambda_nX_n^TP_{n-1}$$

$$\kappa_n \approx \kappa_{n-1} + \gamma_n^T\Lambda_n\gamma_n - \varepsilon_{n|n-1}^T\Lambda_n\varepsilon_n - \varepsilon_{n|n-1}^T\Lambda_nG_{n|n-1}\Lambda_n\varepsilon_n.$$

The corresponding approximation for κ_n is

$$\kappa_n \approx \kappa_{n-1} + \gamma_n^T \Lambda_n \gamma_n - \varepsilon_{n|n-1}^T \Lambda_n \varepsilon_n - \varepsilon_{n|n-1}^T \Lambda_n G_{n|n-1} \Lambda_n \varepsilon_n.$$
(7.13)

As demonstrated in [31], these approximations are useful in deriving an optimum solution for a MW-OBE algorithm. Future work may involve the derivation of a closed form solution for the MW-SM-WRLS and a study of the benefits associated with each additional term (in the series.)

7.7 MW-OBE in the Kalman-Bucy framework

The structural similarities between the Kalman-Bucy (K-B) recursions and MW-OBE are addressed in this section. MW-OBE may be framed in a state-space setting in which the parameter estimator represents the state of a time-varying system. Let us re-write the system of MW-OBE recursions as

$$\begin{cases} \boldsymbol{\theta}_n = \boldsymbol{\theta}_{n-1} + \boldsymbol{K}_n \boldsymbol{\varepsilon}_{n|n-1} \\ \boldsymbol{y}_n = -\boldsymbol{X}_n^T \boldsymbol{\theta}_{n-1} + \boldsymbol{\varepsilon}_{n|n-1} \end{cases}$$

where

$$K_{n} = P_{n-1}X_{n}H_{n}^{-1}\Lambda_{n}$$

= $P_{n-1}X_{n}(G_{n|n-1} + \Lambda_{n}^{-1})^{-1}$

and

$$P_n = K_n X_n^T P_{n-1}.$$

These recursions can be compared with the K-B commonly expressed as

$$\hat{X}_{n|n} = \hat{X}_{n|n-1} + K_n (Y_n - H_{n|n-1} \hat{X}_{n|n-1})$$

$$K_n = \Sigma_{n|n-1} H_n^T (H_{n|n-1} \Sigma_{n|n-1} H_n^T + R_n)^{-1}$$

$$\Sigma_{n|n} = \Sigma_{n|n-1} - K_n H_{n|n-1} \Sigma_{n|n-1}$$

where $\hat{X}_{n|n}$ is the updated state vector, $\Sigma_{n|n}$ is the regressor vector $(H_{n|n-1})$ covariance, and R_n the error-vector covariance, all at time n. In the MW-OBE algorithm, the K-B computation of the matrix R_n is replaced the computation of weights and the state variable κ_n . Therefore a structure identical to the K-B may be used to implement the MW-OBE with the addition of the weights matrix and κ_n .

7.8 Conclusions, contributions and further work

In this dissertation, we derived the MW-OBE algorithm, which was then optimized in the framework of the single-weight QOBE and SM-WRLS algorithms. Simulation studies have revealed enhanced ability to minimize κ_n with increasing K. Enhanced ability to reduce volume also results with increasing K. Reoptimization of weights results in an increased computational cost per update, but the overall complexity remains $\mathcal{O}(m)$ when only adjusting one previous weight since updating is very infrequent. In fact, the number of updates decreased dramatically with increasing K due to a conservative aggregate check for innovation. We also introduced the notion of "forward-looking" and "backward-looking" MW-OBE algorithms as equivalent alternatives.

The major contributions of this research are the following. This research has:

- 1. Introduced the notion of block optimization in OBE algorithms.
- 2. Produced an MW-QOBE algorithm with improved parameter vector convergence speed and reduced ellipsoid volume with respect to the recent QOBE algorithm.
- 3. Produced an MW-SM-WRLS algorithm with improved parameter vector convergence speed and reduced volume with respect to the SM-WRLS algorithm.

- 4. Proved the uniqueness and a condition for existence of the optimal MW-QOBE weights.
- 5. Proved the uniqueness and a condition for existence of the optimal MW-SM-WRLS weights.
- 6. Proved the equivalence of the "forward-looking" and "backward-looking" MW-OBE algorithms.
- 7. Provided simulations showing the performance of both MW-QOBE and MW-SM-WRLS.

8. Suggested the application of MW-QOBE and MW-SM-WRLS in linearprediction analysis of speech signals.



(b) OBE ellipsoid center and other (ellipsoid) points.

Figure 7.2. Spectrum of the voiced /I/ phoneme (512 points). The FFT of the sequence is shown in (a), and the FFT of the impulse response of center $(\theta_{c,L})$, COV $(\theta_{LSE,L})$ and non-central estimators inside the ellipsoid $(\theta_{1-3,L})$ are shown in (b). The speech segment is modeled by an AR(14) system. Ellipsoid volume = 1.4×10^{19} and $4.67 \times 10^{-1} \leq axes \leq 110.10$. See Tables 7.1, 7.2, 7.3 and 7.4 for more information on estimates and ellipsoid.



(b) OBE-ABE ellipsoid center.

Figure 7.3. Poles of the 14th order system based on the COV and OBE ellipsoid center estimates (see Figure 7.2, Tables 7.3 and 7.4).

Appendix A

A.1 Definitions

 ε_n , u_n , and y_n are modeled as random variables defined on a probability space (Ω, \mathcal{F}, P) where Ω is a sample space, \mathcal{F} a σ -field, and P a probability measure. Using Lin's [34] notation [34], define ϵ -neighborhoods of noise bounds as:

$$D_{\epsilon}^{+} = \left[\sqrt{\gamma_{\star}} - \epsilon, \sqrt{\gamma_{\star}}\right]$$

and

$$D_{\epsilon}^{-} = [-\sqrt{\gamma_{\star}}, -\sqrt{\gamma_{\star}} + \epsilon].$$

Definition A.1 [34] A random sequence $\{\varepsilon_n\}$ is called uniformly conditionally tailed (UCT) if given $\epsilon > 0$, there exist a $\delta > 0$ and an infinite subsequence $\{t_i\} \subset N$, such that

$$P(\varepsilon_n \in (D_{\epsilon}^+ \cup D_{\epsilon}^-) | \mathcal{F}_{n-1}) > \delta \ a.s. \ \forall \ n \in \{t_i\}.$$

Definition A.2 The sub-sequence of random vectors, $\{x_{n_k}\}$, is called persistently exciting *(PE)* if there exits $\alpha > 0$, $\beta > 0$ and $T_{PE} < \infty$ such that $\forall n_k$, (subsequence in time)

$$\alpha I < \sum_{i=k}^{k+T_{PE}} \boldsymbol{x}_{n_i} \boldsymbol{x}_{n_i}^T < \beta I$$

Lemma A.1 [41][Matrix Inversion Lemma]: Let the matrix H = A + BDC, where A, D are non-singular matrices of order m, n and B, C are $m \times n$ and $n \times m$

matrices respectively. Then

$$H^{-1} = A^{-1} - A^{-1}B(D^{-1} + CA^{-1}B)^{-1}CA^{-1}$$

Lemma A.2 [41]

$$\det A \det (D + CA^{-1}B) = \det D \det (A + BDC)$$

where A, D are non-singular matrices of order m, n and B, C are $m \times n$ and $n \times m$ matrices respectively.

Lemma A.3 [32] If A^{-1} exists,

$$\begin{bmatrix} \mathbf{A} & \mathbf{D} \\ \mathbf{C} & \mathbf{B} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{A}^{-1} + \mathbf{E} \Delta^{-1} \mathbf{F} & -\mathbf{E} \Delta^{-1} \\ -\Delta^{-1} \mathbf{F} & \Delta^{-1} \end{bmatrix}$$

where $\Delta = B - CA^{-1}D$, $E = A^{-1}D$, $F = CA^{-1}$. If B^{-1} exists, the block (1,1) can be written as $[A - DB^{-1}C]^{-1}$. Δ is known as the Shur complement of A. Also,

$$\det \begin{bmatrix} A & D \\ C & B \end{bmatrix} = \det A \det (B + CA^{-1}D).$$

Lemma A.4 As a extension of Lemma A.3, assuming B^{-1} exists,

$$\begin{bmatrix} \boldsymbol{A} & \boldsymbol{D} \\ \boldsymbol{C} & \boldsymbol{B} \end{bmatrix}^{-1} = \begin{bmatrix} \Delta^{-1} & -\Delta^{-1}\boldsymbol{F} \\ -\boldsymbol{E}\Delta^{-1} & \boldsymbol{B}^{-1} + \boldsymbol{E}\Delta^{-1}\boldsymbol{F} \end{bmatrix}$$

where $\Delta = \mathbf{A} - \mathbf{D}\mathbf{B}^{-1}\mathbf{C}$, $\mathbf{E} = \mathbf{B}^{-1}\mathbf{C}$, $\mathbf{F} = \mathbf{D}\mathbf{B}^{-1}$. If \mathbf{A}^{-1} exists, the block (2,2) can be written as $[\mathbf{B} - \mathbf{C}\mathbf{A}^{-1}\mathbf{D}]^{-1}$.

Proof: Use the development of [32] starting from the top left corner of the matrix. \blacksquare

Lemma A.5 Following Lemma A.4, let A_1 be the $(K \times K)$ symmetric and invertible matrix and u_1 the $(K \times 1)$ vector as described below:

$$oldsymbol{A}_1 = \left[egin{array}{cc} a_1 & a_1^T \ a_1 & oldsymbol{A}_0 \end{array}
ight] \quad oldsymbol{u}_1 = \left[egin{array}{cc} u_1 & oldsymbol{u}_0 \end{array}
ight]$$

where A_0 is invertible $(K - 1 \times K - 1)$, u_1 is $(K - 1 \times 1)$, a_1 and b_1 are scalars. Then,

$$\boldsymbol{A}_{1}^{-1} = \Delta_{1}^{-1} \begin{bmatrix} 1 & | & -(\boldsymbol{A}_{0}^{-1}\boldsymbol{a}_{1})^{T} \\ - - - - - - - | & - - - - - - - \\ - \boldsymbol{A}_{0}^{-1}\boldsymbol{a}_{1} & | & \Delta_{1}\boldsymbol{A}_{0}^{-1} + (\boldsymbol{A}_{0}^{-1}\boldsymbol{a}_{1})(\boldsymbol{A}_{0}^{-1}\boldsymbol{a}_{1})^{T} \end{bmatrix}$$

where $\Delta_1 = a_1 - a_1^T A_0^{-1} a_1$. It follows that

1.
$$A_1^{-1}u_1 = \begin{bmatrix} 0 \\ A_0^{-1}u_0 \end{bmatrix} - \Delta_1^{-1}(u_1 - u_0^T A_0^{-1}a_1) \begin{bmatrix} -1 \\ A_0^{-1}a_1 \end{bmatrix}$$

2. $u_1^T A_1 u_1 = u_0^T A_0 u_0 + 2u_1 a_1^T u_0 + u_1^2 a_1$
3. $\Delta_1 u_1^T A_1^{-1} u_1 = \Delta_1 u_0^T A_0^{-1} u_0 + (u_0^T A_0^{-1}a_1 - u_1)^2$

Proof: Apply Lemma A.4.

Lemma A.6 [1] Let A, B and X be matrices of appropriate dimensions. In addition, let X be invertable with elements independent of each other. Then,

$$\frac{\partial}{\partial X} \operatorname{trace} (AX) = A^{T}$$

$$\frac{\partial}{\partial X} \operatorname{trace} (AXB) = A^{T}B^{T}$$

$$\frac{\partial}{\partial X} \operatorname{trace} (X^{-1}) = -(X^{-1}X^{-1})^{T}$$

$$\frac{\partial}{\partial X} \operatorname{trace} (AX^{-1}B) = -(X^{-1}BAX^{-1})^{T}$$

$$\frac{\partial}{\partial X} \det (X) = \det (X)X^{-T}$$

$$\frac{\partial}{\partial X} \det (AXB) = \det (AXB)X^{-T}$$

$$\frac{\partial}{\partial X} \det (X^{T}) = \frac{\partial}{\partial X} \det (X) - \det (X)X^{-T}$$

where the operator
$$\frac{\partial a}{\partial \mathbf{X}} \stackrel{\text{def}}{=} \left[\frac{\partial a}{\partial \mathbf{X}_{i,j}} \right]$$
 for a scalar *a* (see Table 1.1.)
A.2 Matlab programs

MWOBE, August 25, 1998

```
% Multiple weight OBE algorithm (MWOBE).
 % with volume and kappa minimization options (MW-SM-WRLS and MW-QOBE).
 % Dale Joachim, MSU. 1996, 09/7/97 and 6/9/98.
 %
 % function [thet,param,stat,Ellip] = mwobe(y,w,gam,pm,opt)
 %
 %у
          -> output sequence
 % w
          \rightarrow observable input sequence (when not present use w=y and q=-1)
 % gmax -> gamma max: initial overestimated bound (scalar)
 %
 % pm(1) -> p, order of 'y'
 % pm(2) -> q, order of 'x'
 % pm(3) -> N, non-update maximum window length
 % pm(4) -> ep, 'small number'
 % pm(5) -> reset value of k (kappa)
 % pm(6) -> forgetting factor
%
% opt -> digit(1) Algorithm type: 0->volume (back), 1->kap (back)
%
                                         2->volume (forw), 3->kap (forw)
%
                                         4->volume (simp), 5->kap (simp)
% -> digit(2)
                 Number of past weight revisions
% -> digit(3)
                  Observation vector choice and adaptive K in [K,0] (adaptive K)
% 0 -> non-zero weights & variable K in {0,K}
% 1 -> non-zero weights & K fixed
% 2 -> immediate weights & K fixed
% 3 -> immediate weights & variable K in {0,K}
%
         \rightarrow digit(4)
                       Ellipsoid matrix: 0 -> none, 1-> store
%
% thet <- parameter updates (matrix MxT)
X
% param(1,1:T) <- epsilon, error sequence (vector 1xT) (squared)</pre>
% param(2,1:T) <- vol, ellipsoid volume (vector 1xT)</pre>
% param(3,1:T) <- kappa, kappa (min distance) (vector 1xT)</pre>
% param(4,1:T) <- gamma, estimated error bound (vector 1xT)</pre>
% param(5:7,1:T) <- last 3 weight updates (vector 3xT)</pre>
%
% Ellip
                <- ellipsoid matrices (MxMxT)
%
% stat(1) <- nup, number of data points used (scalar)
% stat(2) <- gup, number of gamma updates (scalar)
% stat(3)
           <- mup, number of mulpiple weight update points</p>
%
```

```
function [thet,param,Ellip,stat] = mwobe(y,w,gam,pm,opt);
```

```
mu = 10^{(-2)}; % small number
 funct = rem(fix(opt/10<sup>0</sup>),10); % function, see next.
 d_typ = rem(fix(opt/10^2), 10); % choice of data.
 nK = rem(fix(opt/10<sup>1</sup>),10) + 1; % TOTAL weight to revise
 % INCLUDING current
 do_ell = rem(fix(opt/10^3),10) == 1; % 1->return each ellipsoid.
 if size(y,1)>size(y,2) y=y'; end; % column vector
 if size(w,1)>size(w,2) w=w'; end;
                                       % column vector
 p = pm(1); q = pm(2); N = pm(3); ep = pm(4); ff = pm(6);
 lY = length(y); % output sequence size
                                % ARX-type model order
M = p+q+1;
 if length(gam) == 1,
gam = gam*ones(size(y));
end;
% ------ Initialization ------
P0 = (1/mu^2) * eye(M); %
P = PO; % initial condition
param = zeros(7,1Y); % epsilon,vol,k,g,k*G
param(2,:) = ones(size(param(2,:)));
thet = zeros(M, 1Y); % model parameters
t = zeros(M,1); % initial parameters
k0 = 1.0; k = k0; P = P*mu; v0 = sqrt(det(k0*P0)); % initial kappa
Ellip = inv(PO)/k; % initial ellipsoid
        v = sqrt(det( k*P )); % volume
        pWb = 0*ones(nK,1); % TOTAL weight revisions
        nup = 0; % number of updates
        mup = 0; % number of multiple weight updates
Y = [zeros(nK-1,1);y(M)];
       U = [gam(1:nK)]'; % initial bound vector
X = [y(M+1-(1:1:p)) w(M+1-(0:1:q))]';
 for i = 1:nK-1,
```

```
X = [zeros(M, 1) X];
      end;
    E = Y - X'*t; % initial error
  ofile = 1; % output device (0 = null)
  % ----- Recursion -----
 fprintf(ofile, '\nMWOBE, K = %d, M = %d: ',nK-1,M);
   for n = M+nK:1Y,
   fprintf(ofile,'.');
 Y1 = [Y(2:end); y(n)]; % new candidates
 U1 = [U(2:end); gam(n)]; % same: gam(n-nK+1:n)]';
   Um = diag(U1); % diagonal matrix
 x1 = [y(n-(1:1:p)) w(n-(0:1:q))]';
 X1 = [X(:,2:end) x1];
 if ( sum(d_typ == [1,2]) ) | (funct > 1) % immediate K observations
 X = X1; Y = Y1; U = U1; \% forward algorithms
 end:
 E = Y1 - X1' * t; Em = diag(E); % error
 G = X1' * P * X1;
                  % weighted energy matrix
d_idx = nK; if(funct>1) d_idx = 1; end;
x = X1(:,d_idx); e = Y1(d_idx) - x'*t; g = x'*P*x;
% initial error & g scalar
u = U1(d_idx);
if sum(funct == [0,2,4])
[L,chk,pWa,S] = mwv_weights(G,U1,E,pWb,[funct,d_typ,k,M]);
else
[L,chk,pWa,S] = mwq_weights(G,U1,E,pWb,[funct,d_typ,k,M]);
end;
switch(chk)
case 1
fprintf(ofile,'\b[%d]',n);
nup = nup + 1; % stat: points used (any adjust.)
X = X1; Y = Y1; U = U1; 1 = L(d_idx); h = 1+1*g;
```

```
P = P - (1/h) * P * x * x' * P;
                                   k = k + (1/h) * (h * u^2 - e^2);
                                   if k < 10e-8, k=0.1; end;
 t = t + 1 + e + P + x;
 case 2
 fprintf(ofile,'\b<%d>',n);
 nup = nup + 1; % stat: no of used pts
 X = X1; Y = Y1; U = U1;
 Lm = diag(L); H = eye(nK)+Lm*G;
                                   P = P - P + X + inv(H) + Lm + X' + P;
                                   k = k + U'*Lm*U - E'*inv(H)*Lm*E;
                                   if k < 10e-8, k=0.1; end;
 t = t + P * X * Lm * E;
 mup = mup + 1;
 end;
 if k<0; k = 0.1; end; % adaptation
 if sum(eig(P)<0), eig(P),</pre>
fprintf('\nMWOBE problem! negative eigenvalues! \n');
keyboard, end;
% ------ saving up the information ------
                 param(5,n) = pWa(1);
                                            % log past 3 weights
                 if nK>1, param(6,n) = pWa(2); end;
                 if nK>2, param(7,n) = pWa(3); end;
                 v = sqrt(det( k*P ));
                                         % volume
                 thet(:,n) = t;
                                          % parameters
                 param(1,n) = Y1(d_idx) - x'*t; param(2,n) = v; %
                 param(3,n) = k; param(4,n) = U(d_idx);
                                                                %
pWb = [pWa(2:nK); 0]; % updated weights
if ( sum(d_typ == [2,3]) ) & (chk==0) % immediate K observations
X = X1; Y = Y1; U = U1; % forward algorithms
end;
```

if do_ell, Ellip = [Ellip inv(P)/k]; end;

end

```
param(1,1:M+nK-1) = param(1,M+nK)*ones(1,M+nK-1); % initial error
param(2,1:M+nK-1) = v0*ones(1,M+nK-1); % initial vol
param(3,1:M+nK-1) = k0*ones(1,M+nK-1); % initial kappa
param(4,1:M+nK-1) = param(4,M+nK)*ones(1,M+nK-1); % initial gamma
```

```
stat = [nup 0 mup];
```

```
MWOBE, August 25, 1998
 % MW-QOBE weights
 % dale joachim, msu, 6/9/98
 %
 % function [L,chk,pWa,S] = mwq_weights(G,U,E,pWb,opt)
 %
 % G
          -> Weighted energy matrix (X'PX)
 % U
         -> upper bound vector
 % E
         -> error vector
 % pWb -> vector of past (cumulated) weights (before)
 % opt -> opt(1): funct (see MWOBE.m)
 %
          opt(2): d_typ (see MWOBE.m)
 %
          opt(3): kappa from previous iteration
 %
          opt(4): system order M
 %
 %
 % L <- weight adjustments vector
 % chk <- weight exists ? (0:no, 1:qobe, 2:mwobe)
 % pWa <- vector of (cumulated) weights (after)
 % S <- sign vector
%
function [L,chk,pWa,S] = mwq_weights(G,U,E,pWb,opt)
nK = size(G,1); \%
  Um = diag(U); % diagonal matrix
  Em = diag(E); % error
  pWa = pWb;
  ddir = opt(1) > 1;
  vK = sum (opt(2) == [0,3]);
  simpl = opt(1) > 3;
  k = opt(3);
  M = opt(4);
chk = 0; L = zeros(nK,1); S = 0; % initialization
if ddir, t = 1; else t = nK; end;
e = E(t); u = U(t); g = G(t,t);
if (nK > 1)
    for i = 0:0 \% (2^nK)-1 \% for this study, do all!
    if rank(G) == nK
```

```
S1 = [((dec2bin(i,nK) == dec2bin(0,nK)) \dots]
        -~ (dec2bin(i,nK) == dec2bin(0,nK)))]';
     Se = sign(E)+(E==0); % sign E
     S1 = Se; % note for now !
     Sm = diag(S1);
B = G * Um * Sm; % other: B = Em * Gm * Em;
L1 = inv(B)*(E-(U.*S1));
if prod(L1+pWb>0)
chk = 2; S = S1; L = L1;
if simpl, % only one weight logged
chk = 1;
pWa(t) = L(t);
else
pWa = pWb + L;
end;
end;
end;
   end;
end;
if (~chk) & (( abs(e) > u )) & ((nK==1)|((nK>1)&vK));
% last cond. forces nK fixed
S = e/abs(e); chk = 1;
    L(t) = (abs(e) - u)/(u*g);
    pWa(t) = L(t);
end;
```

128

```
<u>MWOBE</u>, August 25, 1998
 % MW-SM-WRLS weights
 % dale joachim, msu, 6/9/98
 %
 % function [L,chk,pWa,S] = mwv_weights(G,U,E,pWb,opt)
 %
 % G
          -> Weighted energy matrix (X'PX)
 % U
         -> upper bound vector
 % E
          -> error vector
 % pWb -> vector of past (cummulated) weights
 % opt -> opt(1): funct (see MWOBE.m)
 %
          opt(2): d_typ (see MWOBE.m)
 %
          opt(3): kappa from previous iteration
 %
          opt(4): system order M
 %
 %
% L <- weight adjustments vector
% chk <- weight exists ? (0:yes, 1:sm-wrls, 2:mw-sm-wrls)
% pWa <- vector of (cummulated) weights
% S <- sign vector
%
function [L,chk,pWa,S] = mwv_weights(G,U,E,pWb,opt)
nK = size(G,1); %
  Um = diag(U); % diagonal matrix
  Em = diag(E); % error
  pWa = pWb;
  ddir = opt(1) > 1;
  vK = sum (opt(2) == [0,3]);
  simpl = opt(1) > 3;
  k = opt(3);
  m = opt(4);
chk = 0; L = zeros(nK,1); S = 0; % initialization
if ddir, t = 1; else t = nK; end;
e = E(t); u = U(t); g = G(t,t); Gi = inv(G);
if (e^2 - u^2 + k \cdot g/m > 0) & (nK > 1)
  E1 = Gi * E; U1 = Gi * U;
  b = k - E' * E1 - U' * U1;
  if prod(prod((E*E' - U*U')*Gi + (k/m) > 0))
```

```
% ----- numerical solution ------
 a1 = 0; a2 = 2*pi;
 incr = (a2-a1)/500; tt = [incr:incr:a2];
 uvec = zeros(length(tt),2);
 vvec = zeros(length(tt),2);
 dvec = zeros(length(tt),2);
 tvec = 10^9 * ones(length(tt),1);
 uvec(:,1) = [U(1)*cos(tt) - U(2)*sin(tt)]';
 uvec(:,2) = [U(2)*cos(tt) + U(1)*sin(tt)]';
 vvec(:,1) = [E1(1)*cos(tt) - E1(2)*sin(tt)]';
 vvec(:,2) = [E1(2)*cos(tt) + E1(1)*sin(tt)]';
 alfv = (uvec./vvec).*(m*uvec.*vvec - E1'*U);
 betv = (vvec./uvec).*(m*uvec.*vvec + E1'*U);
 dvec = (b + sqrt((b<sup>2</sup>) + 4*(alfv).*(betv)))./(2*alfv);
 idx = isreal(dvec);
for i=1:length(tt)
if (isreal(dvec(i,:)))
DD = diag(dvec(i,:));
uu = uvec(i,:)'; vv = vvec(i,:)';
f = b + uu'*DD*uu + vv'*DD*vv;
F = m*DD*uu*uu'*DD - f*DD - m*vv*vv';
tvec(i) = trace(F);
end:
end;
\% tvec = (tvec < 0) * 10^50 + (tvec>0). * tvec;
[val,idx] = min(tvec);
angl = tt(idx); D = diag(dvec(idx,:));
R = [cos(angl) -sin(angl); sin(angl) cos(angl)];
XX = R' * D * R; L1 = diag(XX - Gi);
plot(tvec);
for i = 1:1
zoom on;
agl = ginput(1); angl = agl(1);
R = [cos(angl) -sin(angl); sin(angl) cos(angl)];
uvec = R*U; vvec = R*E1;
alfv = (uvec./vvec).*(m*uvec.*vvec - E1'*U);
```

```
betv = (vvec./uvec).*(m*uvec.*vvec + E1'*U);
dvec = (b + sqrt((b^2) + 4*(alfv).*(betv)))./(2*alfv);
D = diag(dvec); XX = R'*D*R; L1 = diag(XX - Gi);
val = prod(L1>0);
end;
```

```
if prod(L1+pWb>0)&(val>0)&isreal(L1)
chk = 2; L = L1;
if simpl, % only one weight logged
chk = 1;
pWa(t) = L(t);
else
pWa = pWb + L;
end;
end;
```

end;

```
end;
```

```
if (~chk) & (( m*(u^2 - e^2) - k*g < 0 )) & ((nK==1)|((nK>1)&vK)) ; %
    a = (m-1)*(u*g)^2;
    b = u^2*g*(2*m-g)+e^2*g-k*g^2;
    c = m*(u^2-e^2)-k*g;
    L(t) = (-b+sqrt(b^2-4*a*c))/(2*a);
    pWa(t) = L(t); chk = 1;
```

end;

References

- [1] Michael Athans. The matrix minimum principle. Information and Control, 11:592-606, January 1968.
- [2] Cs. Bányász and L. Keviczky (editors). In Proc. 9th IFAC/IFORS Symp. on Identification and System Parameter Estimation, volume 1 & 2, Budapest, July 1991.
- [3] Stephen Barnett. Matrices: methods and applications. Oxford University Press, New York, 1990.
- [4] G. Belforte, B. Bona, and M. Milanese. Advanced modelling and identification techniques for metabolic processes. CRC J. Biomedical Engr., 10:275-316, 1983.
- [5] D. P. Bertsekas and I. B. Rhodes. Recursive state estimation for a setmembership description of uncertainty. *IEEE Trans. on Automatic Control*, AC-16:117-128, April 1971.
- [6] M. F. Cheung. On Optimal Algorithms for parameter set estimation. PhD thesis, Ohio State University, Columbus, 1991.
- [7] M. F. Cheung, S. Yurkovich, and K. M. Passino. An optimal volume ellipsoid algorithm for parameter set estimation. In Proc. 30th IEEE Conf. Decision and Control, pages 969-974, Brighton, December 1991.
- [8] S. Dasgupta and Y.-F. Huang. Asymptotically convergent modified recursive least squares with data dependent updating and forgetting factor for systems with bounded noise. *IEEE Trans. on Information Theory*, 33:383-392, 1987.
- [9] John R. Deller, Jr., J.G. Proakis, and J.H.L. Hansen. Discrete-Time Processing of Speech Signals. MacMillan, New York, 1993.
- [10] J.R. Deller, Jr. Set-membership identification in digital signal processing. *IEEE* Acoustics, Speech, and Signal Processing Magazine, 6:4-22, 1989.
- [11] J.R. Deller, Jr. Personal communication. 1997.
- [12] J.R. Deller, Jr., S. Gollamudi, S. Nagaraj, Y.-F. Huang, and S. Kapoor. Convergence analysis of a new "Quasi-OBE" algorithm for real-time signal processing. *In preparation.*
- [13] J.R. Deller, Jr. and T. C. Luk. Linear prediction analysis of speech based on set-membership theory. Computer Speech and Language, 3:301-327, October 1989.
- [14] J.R. Deller, Jr., M. Nayeri, and M. S. Liu. Unifying the landmark developments in optimal bounding ellipsoid identification. Int. J. Adaptive Control and Signal Processing, 8:48-63, Jan.-Feb. 1994.

- [15] J.R. Deller, Jr., Majid Nayeri, and S. F. Odeh. Least-square identification with error bounds for real-time signal processing and control. *Proc. IEEE*, 81(6):813– 849, June 1993.
- [16] J.R Deller, Jr. and M. Nayeri (session chairs). In Proc. International Symp. on Circuits and Systems, volume 1, Chicago, May 1993. (Special session on set-membership-based identification for signal processing and control).
- [17] J.R. Deller, Jr. and S. F. Odeh. Adaptive set-membership identification in O(m) time for linear-in-parameters models. *IEEE Trans. on Signal Processing*, 41(5), May 1993.
- [18] J.R. Deller, Jr. and G. P. Picaché. Advantages of a Givens rotation approach to temporally recursive linear prediction analysis of speech. *IEEE Trans. on Acoustics, Speech, and Signal Processing*, 37(3):429-431, 1989.
- [19] G. Fiorio, S. Malan, and M. Milanese. Guaranteed specification control design of inertial platforms for a high accuracy calibration device. In Proc. 29th Annual IEEE Conf. Decision and Control, Honolulu, 1990.
- [20] E. Fogel. System identification via membership set constraints with energy constrained noise. *IEEE Trans. on Automatic Control*, 24:752-758, 1979.
- [21] E. Fogel and Y.-F. Huang. On the value of information in system identification - Bounded noise case. *Automatica*, 18:229–238, 1982.
- [22] S. Gollamudi, S. Nagaraj, and Y.-F. Huang. SMART: A toolbox for setmembership filtering. In Proc. 1997 European Conf. on Circuit Theory and Design, Budapest, September 1997.
- [23] Gene H. Golub and Charles F. Van Loan. *Matrix Computations*. The John Hopkins University Press, Baltimore, Maryland, 1983.
- [24] R. Gomeni, H. Piet-Lahanier, and E. Walter. Study of the pharmacokinetics of betaxolol using set-membership set estimation. In Proc. 3th IMEKO Congress on Measurements in Clinical Medecine, Edinburgh, 1986.
- [25] D. Graupe. Time Series Analysis, Identification, and Adaptive Systems. Krieger, Malabar, Florida, 1989.
- [26] Simon Haykin. Adaptive Filter Theory. Prentice-Hall, Englewood-Cliffs, New Jersey, 1995.
- [27] Roger A. Horn and Charles R. Johnson. *Topics in Matrix Analysis*, volume 2. Cambridge University Press, 1991.
- [28] Alan Jennings. Matrix Computations for Scientists and Engineers. John Wiley and Sons, 1977.

- [29] D. Joachim, J.R Deller, Jr., and Majid Nayeri. Practical considerations in the use of a new OBE algorithm that blindly estimates error bounds. In Proc. 40th Midwest Symposium on Circuits and Systems, pages 762-765, Sacramento, August 1997.
- [30] D. Joachim, J.R Deller, Jr., and Majid Nayeri. Weight reoptimization in OBE algorithms: An initial study. In Proc. 35th Allerton Conference on Communication, Control and Computing, pages 789–797, Monticello, Illinois, October 1997.
- [31] D. Joachim, J.R. Deller, Jr., and Majid Nayeri. Multiweight optimization in OBE algorithms for improved tracking ans adaptive identification. In Proc. Int. Conf. on Acoustics, Speech, and Signal Processing, Seattle, May 1998.
- [32] Thomas Kailath. Linear Systems. Prentice-Hall, Englewood Cliffs, New Jersey, 1980.
- [33] K. Keesman. Robust identification and prediction for output-error models. In Proc. 9th IFAC/IFORS Symp. on Identification and System Parameter Estimation, volume 2, pages 878-882, Budapest, July 1991.
- [34] T. M. Lin. Optimal bounding ellipsoid algorithms with automatic bound estimation. PhD thesis, Michigan State University, East Lansing, 1996.
- [35] M. S. Liu. Development and analysis of an optimal bounding ellipsoid algorithm using stochastic approximation. PhD thesis, Michigan State University, East Lansing, August 1993.
- [36] L. Ljung and T. Södertröm. Theory and Pratice of Recursive Identification. MIT Press, Cambridge, Mass., 1983.
- [37] M. Milanese and G. Belforte. Estimation theory and uncertainty evaluation intervals evaluation in the presence of unknown but bounded errors: Linear families of models and estimators. *IEEE Trans. on Automatic Control*, AC-27:408-414, 1982.
- [38] M. Milanese, R. Tempo, and A. Vicino. Optimal error prediction for economic models. Int. J. System Science, 19:1189-1200, 1988.
- [39] S. Nagaraj, S. Gollamundi, J.R Deller, Jr., Y.-F. Huang, and S. Kapoor. Performance studies on a "Quasi-OBE" algorithm for real-time signal processing. In Proc. 40th Annual Midwest Symposium on Circuits and Systems, volume 2, pages 770-773, Sacramento, August 1997.
- [40] M. Nayeri, M.S. Liu, and J.R Deller, Jr. Do interpretable optimal bounding ellipsoid algorithms converge? parts I & II. In Proc. 10th IFAC/IFORS Symp. on System Identification (SYSID '94), volume 3, pages 389-400, Copenhagen, July 1994.

- [41] Ben Noble. Applied Linear Algebra. Prentice-Hall, Englewood-Cliffs, New Jersey, 1969.
- [42] J. P. Norton. Problems in identifying the dynamics of biological systems from very short records. In Proc. 25th Annual IEEE Conf. Decision and Control, Athens, 1986.
- [43] Richard E. Phillips. A Course in Matrix Theory (class notes). Michigan State University, 1992.
- [44] F. C. Schweppe. Recursive state estimation: Unknown but bounded errors and system inputs. *IEEE Trans. on Automatic Control*, AC-13:22-28, February 1968.
- [45] T. Söderström and P. Stoica. System Identification. Prentice-Hall, Englewood-Cliffs, New Jersey, 1989.
- [46] S. M. Veres and J. P. Norton. Structure selection for bounded-parameter models: Consistency conditions and selection criterion. *IEEE Trans. on Automatic Control*, 36:474-481, 1991.
- [47] A. Vicino, R. Tempo, R. Genesio, and M. Milanese. Optimal error and GMDH predictors: A comparison with some statistical techniques. Int. J. Forecasting, 2:313-328, 1987.
- [48] E. Walter, Y. Lecourtier, J. Happel, and J. Y. Kao. Identifiability and distinguishability of fundemental parameters in catalytic methanation. J. American Inst. Chemical Engineer, 32:1360-1366, 1988.
- [49] H. S. Witsenhausen. Sets of possible state of linear systems given perturbed observation. *IEEE Trans. on Automatic Control*, AC-13:556-568, October 1968.

