Queensland Government
Department of Natural Resources and Mines
Department of Emergency Services
Environmental Protection Agency

Queensland Climate Change and Community Vulnerability to Tropical Cyclones

OCEAN HAZARDS ASSESSMENT - Stage 1a

Operational Manual

March 2004

MMSURGE Modelling System and Parametric Model Processing

SEA Systems Engineering Australia Pty Ltd
Numerical Modelling and Risk Assessment

Australian Government Bureau of Meteorology

In association with:
JCU James Cook University
Marine Modelling Unit

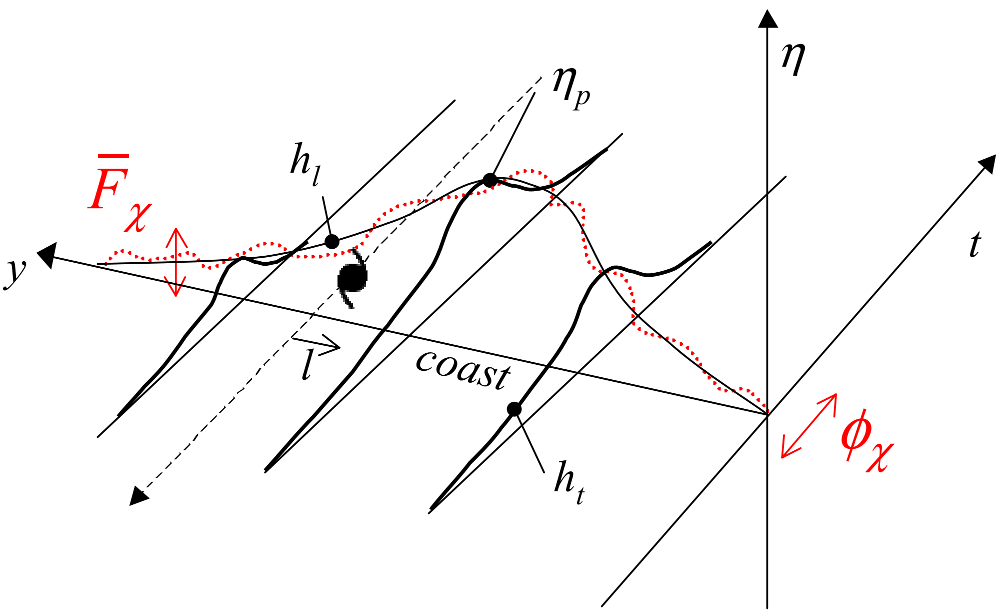
Australian Government
Bureau of Meteorology

Queensland Regional Office

MMUSURGE MODELLING SYSTEM AND PARAMETRIC MODEL PROCESSING

OPERATIONAL MANUAL

March 2004



SEA
Systems Engineering Australia Pty Ltd
Numerical Modelling and Risk Assessment

J0103-PR001B

Contents

Tables

Figures

1. Introduction

The *Queensland Climate Change and Community Vulnerability to Tropical Cyclones – Ocean Hazards Assessment* project commenced in 2000. Stage A.1 of the project was awarded to Systems Engineering Australia Pty Ltd (SEA) in association with the James Cook University Marine Modelling Unit (MMU). Stage A.1(a) consisted of a technical review of the methodology, data and approaches required to undertake the study, while Stage A.1(b) required the supply and installation of the MMUSURGE numerical storm surge model and associated numerical model domains for the Queensland coast onto the Queensland Regional Office computer system.

Stage A.1(a) (Harper 2001) recommended that the optimum long-term storm tide forecast and warning system would be based on real-time application of numerical storm surge and spectral wave models integrated into Bureau of Meteorology forecasting systems. However, it was also recommended that a shorter-term implementation of an efficient MEOW-style capability could be based on the development of parametric models for tide, storm surge and waves. The Queensland Regional office subsequently decided to pursue in-house development of a parametric storm tide modelling system with initial specialist assistance from SEA.

This document provides:

1. An overview of the Stage A.1(b) implementation of the MMUSURGE numerical hydrodynamic model and its operational needs, and

2. The specification, implementation and documentation of a parametric storm tide model development system forming Stages A.1(c) and A.1(d).

The work described here was undertaken over a number of years, commencing with the installation of the MMUSURGE model at the end of 2000 (Stage A.1(b)), followed by short-term SEA assistance with the in-house parametric model development (Stage A.1(c)), mainly between March and October of 2001. Initial processing was facilitated by the use of *OCTAVE* software, an Open Software Foundation emulation of *MATLAB*™. However, the capabilities offered by *OCTAVE* were limited and under Stage A.1(d), additional external short-term *C* language programming assistance was obtained by the Queensland Regional Office in early 2002. Production **A** and **B** grid modelling for the Queensland east coast was then completed in 2004 for coast-crossing storm scenarios. The parallel-to-coast case is yet to be considered and there has been no wave modelling or wave setup analyses undertaken.

The structure of this document, which was finalised in its present form following the 2004 *Project Synthesis Report* (Harper 2004), is as follows:

- Section 2 provides an overview of the system directory structure and the various Unix™ scripts developed to automate the storm surge modelling production work;

- Section 3 discusses the numerical model domains supplied under Stage A.1(a) and the indicative CIRRUS1 workstation performance obtained during Stage A.1(b);

- Section 4 provides details of operations for the various software sub-systems under Stage A.1(c) and (d);

- Section 5 details the statistical analyses done under Stage A.1(c) by SEA to provide an objective basis for deciding on the storm parameter ranges to be modelled by MMUSURGE for each **A** and **B** grid;

- Section 6 explains the technical basis of the parametric storm surge model;

- Section 7 outlines how the processing of the MMUSURGE output is specifically organised to provide the form of the parametric model;

- Appendix A contains the set of mapped domains developed under Stage A.1(c) where the Stage A.1(b) supplied domains were augmented with locations from the official gazetteer;

- Appendix B contains a detailed description of the parametric processing software developed in the *C* language during Stage A.1(d);

- Appendix C describes how the parametric prediction is undertaken.

This report should also be read in conjunction with the MMUSURGE User's Guide provided by James Cook University (Mason and McConochie 2001).

2. Operational Software Overview

2.1 General

The following aliases can be used to run the MMUSURGE model and its support programs:

mmu	runs the MMUSURGE model, which will expect input files to be located in the appropriate directory.
wav	converts model WAV binary time history data into ASCII format ASC files for subsequent import into an MS Excel spreadsheet MMUWAV.XLS
huv	converts model HUV binary spatial snapshots into ASCII format for viewing and/or importing into other packages.
g2c	converts between lat/long and grid coordinates to allow specification of additional grid locations in the RUN file for model output.
gtabs	produces a set of MAPINFO-compatible source tables for domain plotting
makec	creates a “dummy” C grid BED file given the B grid location and extent
coast	converts MAPINFO-compatible coastline sites into format for *maketrack*
maketrack	generates straightline constant parameter storm tracks based on user specifications to give all the necessary RUN and CYC files for an A to B grid simulation. MAPINFO-compatible track files are also produced together with a batch control file.

2.2 Source Directory Structure

The following provides an overview of the major project and model directories:

/ssurge/meow			project root directory
/mmusurge			MMUSURGE model root directory
	/bin		all MMUSURGE model executables
	/doc		model documentation
	/src		JCU supplied model source code
		/maketrack	SEA automatic track file generator
		/nspcg	numerical subroutine library
	/bed_files		master bed files for the Qld coast
	/run_files		master run files with "top 10" sites
	/verify		Althea hindcast verification
	/tst		test area
	/<domain>		MMUSURGE domain root (refer 2.4 for details)
/maps			MAPINFO grid plotting sub-system
/tcstats			Statewide climatology of MEOW parameters
/param_model			BoM parametric model root directory
	/basin01		parametric basin reference (01 only has source)
		/prepare_src	source of parametric model processing
		/prepare_bin	executables of parametric model processing
		/predict_src	source of parametric model prediction
		/predict_bin	executables of parametric model prediction
		/utils_bin	utility scripts for parametric processing etc

/basin*nn*	parametric basin reference (more than one per MMUSURGE domain is possible)

Refer to individual readme.txt files in each directory for further details.

2.3 MMUSURGE Environment Variables

A special script file *setup* should be executed in the /ssurge/meow directory to establish a range of file pointers and provide aliases for executing the various model programs, e.g.

. *setup*

will set up the necessary environment variables and useful aliases.

The model and many of its support programs will evaluate system environment variables to determine the path for various input and/or output files. The *setup* script provides initial definitions as follows:

The system-specific environment variables and default values are:

MEOW	project root directory	/ssurge/meow
MMU	model root directory	$MEOW/mmusurge
MMUBIN	executables directory	$MMU/bin
MMUTST	default test directory	$MMU/tst
MMUBED	all production bed files	$MMU/bed_files
MMUCST	coastal thistory files	$MMUBED/b_coasts

The following model-specific runtime environment variables are initially set to $MMUTST:

MMURUN	run files
MMUCYC	cyc files
MMUPUT	putbdy output
MMUGET	getbdy input
MMUTID	all tide files
MMUHOT	hot start files
MMUASC	wav and huv ASCII output files
MMUMAP	MAPINFO-compatible CYC file for plotting and checking tracks
MMUOUT	all other output files

The alias *sethere* (via script $MEOW/setheremmu) will set all runtime environment variables to $PWD for testing purposes.

2.4 MMUSURGE Production Directory Structure

The alias *setpath* (script setpathmmu <top> <domain> <output>) will establish the following standard production directory structure:

```
        Environment Variable:                              Path:

$MMUTOP                                                    $MEOW/<top>
   |____ $MMUDOM                                           $MMUTOP/<domain>
           |_______   $MMUOUT                              $MMUDOM/<output>
                            |_____ $MMURUN                 $MMUOUT/run
                            |_____ $MMUCYC                 $MMUOUT /cyc
                            |_____ $MMUGET                 $MMUOUT /put
                            |_____ $MMUPUT                 $MMUOUT /put
                            |_____ $MMUHOT                 $MMUOUT /hot
                            |_____ $MMUTID                 $MMUOUT /tid
                            |_____ $MMUASC                 $MMUOUT /asc
                            |_____ $MMUMAP                 $MMUOUT /map
```

In practice, several <top> <domain> <output> schemes were used. Refer to Appendix B for further information, where a specific domain is used as an example

2.5 MMUSURGE Production Runs

The utility program *maketrack* (refer later for details) will generate all required input data files automatically and place them in the directories defined by *setpath*. In addition, a batch control file is placed in $MMUDOM which can be supplied to the system *at* command, e.g.

at –f <control_file> now

will start a background process to execute all the production runs generated by *maketrack*. An email is sent to the user with the stdout and stderr text output from the background process to enable confirmation of the successful completion of the job.

2.6 MMUSURGE Forecasting

The present operation of the model for real-time forecasting is facilitated by a PERL script written by Matt Saunderson, located in $MEOW \run_script*run_mmu.pl*

The model requires both an **A** grid and a **B** grid for each simulation. The script ensures that the appropriate grids are invoked by copying dummy RUN and CYC files from the requisite source directories and editing to include the required parameters. Note that the supplied *.run files provide a series of coastal locations which will need to be augmented as required.

This can be combined with an EXCEL spreadsheet provided to allow graphical output of time histories of water levels at up to 10 coastal locations. This is via *wav2asc*, which is executed following the script and the procedure explained in the spreadsheet must then be followed to import the *.asc file into MMUWAV.XLS.

Refer to later OCTAVE scripts and *C* source for greater analysis functionality.

2.7 Building the MMUSURGE Model Executables

All executables reside in $MMUBIN but are built by various scripts, as follows:

Alias	Source	Build File	Build File Type
mmu	mmusurge.f	/src/makemmusurge	MAKE file ie "make –f makemmusurge"
wav	wav2asc.f	/src/bld_asc.b	script
huv	huv2asc.f	/src/bld_asc.b	script
g2c	grid2coord.f	/src/bld_grid2coord.b	script
gtabs	grid_tables.f	/src/bld_grid_tables.b	script
makec	make_Cgrid.f	/src/bld_make_cgrid.b	script
coast	make_coast.f	/src/bld_make_coast.b	script
maketrack	maketrack.f	/src/maketrack/bld_maketrack.b	script

All code is FORTRAN 90, except for maketrack and make_coast, which are FORTRAN 77. The AIX™ XLF90 and XLF FORTRAN compilers are used.

2.8 Building the Parametric Processing Model Executables

All source code for the parametric processing routines reside in:

```
/surge/meow
        /param_model
                /basin01
                        /prepare_src
```

and rely on the const_value.h library file in the same directory. All routines are built by a compilation script in the same directory. Output of this compilation script is to:

```
/surge/meow
        /param_model
                /basin01
                        /prepare_bin
```

Executable	Source	Build File	Build File Type
process_wav_data	process_wav_data.c	bld_prepare.b	Script
	process_4_tracks_time.c	bld_prepare.b	Script
	convert_lspace_time.c	bld_prepare.b	Script
	process_wavs.c	bld_prepare.b	Script
	convert_l_space.c	bld_prepare.b	Script
	create_envelope.c	bld_prepare.b	Script

	create_ratios.c	bld_prepare.b	Script
	find_env_mean.c	bld_prepare.b	Script
	get_wavs.c	bld_prepare.b	Script
	table_interp.c	bld_prepare.b	Script
comp_model	comp_model.c	bld_prepare.b	Script
prepare_param	prepare_param.c	bld_prepare.b	Script
water_level	water_level.c	bld_prepare.b	Script
param_ratio_fchi	param_ratio_fchi.c	bld_prepare.b	Script
param_ratio_hp	param_ratio_hp.c	bld_prepare.b	Script
param_ratio_mh	param_ratio_mh.c	bld_prepare.b	Script
param_ratio_mt	param_ratio_mt.c	bld_prepare.b	Script
param_ratio_phi	param_ratio_phi.c	bld_prepare.b	Script
param_ratio_ts	param_ratio_ts.c	bld_prepare.b	Script

2.9 Building the Parametric Predictive Model Executables

All source code for the parametric predictive routines reside in:

/surge/meow
    /param_model
        /basin01
            /predict_src

and rely on the const_value.h library file in the same directory. The final predictive routine is built by a compilation script in the same directory. Output of this compilation script is to:

/surge/meow
    /param_model
        /basin01
            /predict_bin

Executable	Source	Build File	Build File Type
surge_param	surge_param_model.c	bld_predict.b	Script
	closest_site.c	bld_predict.b	Script
	calc_l_func.c	bld_predict.b	Script
	interp3d_hp_func.c	bld_predict.b	Script
	interp3d_wl_func.c	bld_predict.b	Script
	table_2d_interp_func.c	bld_predict.b	Script
	table_3d_interp_func.c	bld_predict.b	Script
	table_interp.c	bld_predict.b	Script

3. Model Definitions of the Queensland Coast

3.1 Model Domains

These are summarised in Table 3.1 and Figure 3.1. It should be noted that the origins of the grids vary depending on their orientation. The N-S grids **A00**, **A04** and **B09** have a conventional lower-left origin position. Each grid on the eastern coast (**B01** to **B08** plus **B12**) has the origin at the top-left corner, with +ve x directed southwards along the coast and +ve y directed offshore. Each grid on the western coast (**B10** and **B11**) has its origin at the lower-right with the +ve x axis directed northwards and the +ve y axis directed offshore.

A series of "dummy" **C** grids have also been defined to cover the majority of the **B** grid coastline. Table 3.2 summarises the locations and sizes of the proposed **C** grids relative to each **B** grid. The orientation is assumed to be the same as the parent **B** grid but with a resolution of 555 m (0.0050 deg), representing a 5:1 reduction over the 2.8 km **B** grid. These parameters are read by *makec* when constructing dummy BED files for these locations so that the information can be interpreted by MMUSURGE when the **B** grid simulations are being generated. This will allow open boundary conditions to be made available at future times for **C** grid simulations. Actual **C** grid bathymetry will then need to be assembled to enable the model operation. Figure 3.2 shows the **B** and **C** grid relationships and Appendix A contains detailed maps of the **C** grid coverage within each **B** grid.

Table 3.1 MMUSURGE **A** and **B** model domain parameters.

Grid No.	Location	Timestep	Resolution		Extent		Dimension (km)	
		s	' arc	km	Alongshore	Offshore	Alongshore	Offshore
A00	Queensland Coast	1800	7.5	13.9	201	57	2780	778
A01	Southern Queensland Coast	1800	7.5	13.9	101	57	1390	778
A02	Central Queensland Coast	1800	7.5	13.9	101	57	1390	778
A03	Northern Queensland Coast	1800	7.5	13.9	101	57	1390	778
A04	Gulf of Carpentaria	1800	7.5	13.9	105	117	1966	1612
B01	Brisbane	900	1.5	2.8	161	65	448	179
B02	Hervey Bay - Gladstone	900	1.5	2.8	153	61	426	168
B03	Mackay	900	1.5	2.8	161	125	448	347
B04	Whitsunday	900	1.5	2.8	201	101	560	280
B05	Townsville	900	1.5	2.8	161	65	448	179
B06	Cairns	900	1.5	2.8	177	45	493	123
B07	Cooktown	900	1.5	2.8	161	73	448	202
B08	Lockhart River	900	1.5	2.8	149	65	414	179
B09	Torres Strait	900	1.5	2.8	131	141	364	392
B10	Weipa	900	1.5	2.8	161	41	448	112
B11	Karumba	900	1.5	2.8	169	49	470	134
B12	Mornington Island	900	1.5	2.8	161	57	448	157

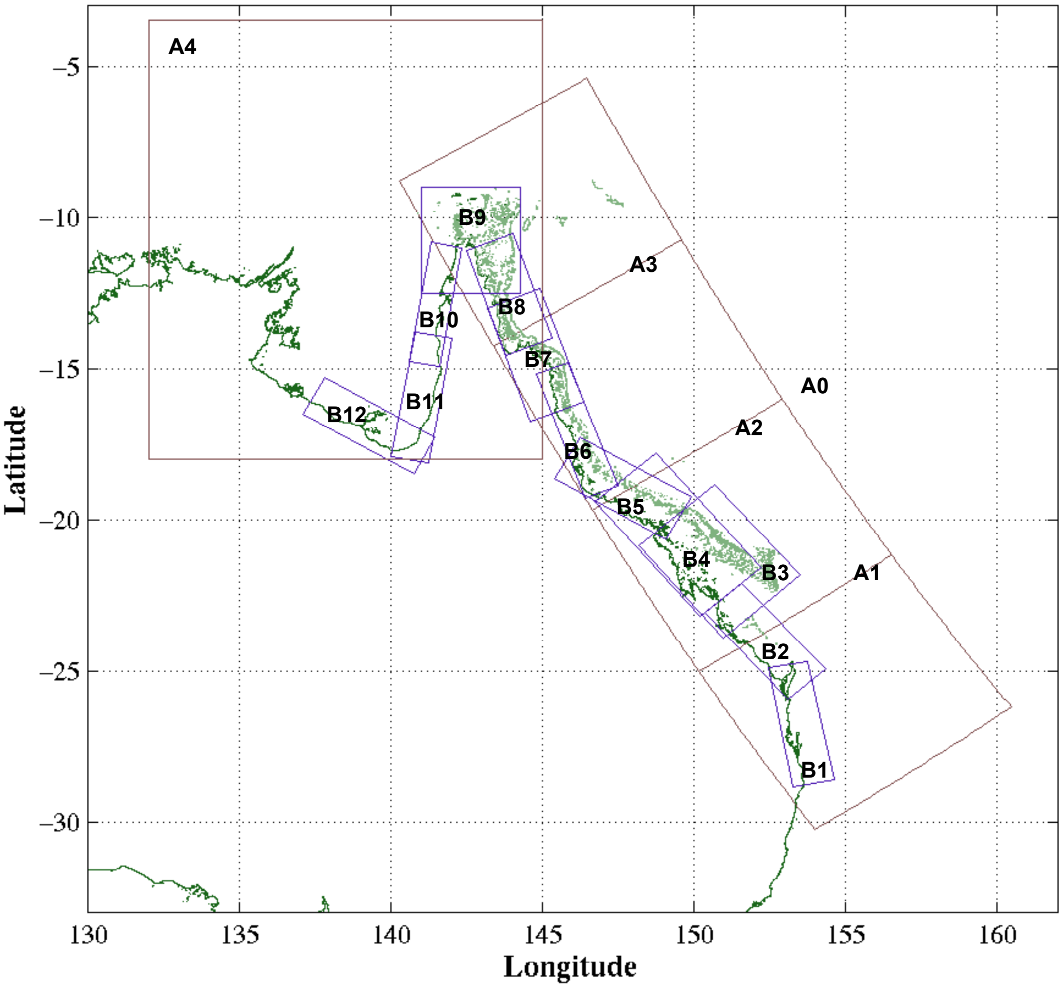


Figure 3.1 Storm surge model **A** and **B** domains for the Queensland coast.

Table 3.2 MMUSURGE **C** domain definitions.

Parent	Local	C Grid Origin in B Grid		C Grid Extent		Unit C Grid Size		C Grid Extent		Dimension (km)		C Grid
B Grid	C Grid	I	J	IMAX	JMAX	deg	m	IMAX	JMAX	Along shore	Off shore	Name
		B units	B units	B units	B units			C units	C units			
B01	C01	118	12	29	8	0.0050	555	140	35	78	19	Gold Coast
	C02	79	4	50	19	0.0050	555	245	90	136	50	Moreton Bay
	C03	47	10	37	10	0.0050	555	180	45	100	25	Sunshine Coast
	C04	17	10	31	12	0.0050	555	150	55	83	31	Great Sandy Strait
B02	C01	120	1	30	33	0.0050	555	145	160	80	89	Hervey Bay
	C02	78	5	54	14	0.0050	555	265	65	147	36	Bundaberg
	C03	40	4	39	18	0.0050	555	190	85	105	47	Gladstone
	C04	3	1	38	21	0.0050	555	185	100	103	56	Yeppoon
B03	C01	75	19	55	19	0.0050	555	270	90	150	50	Shoalwater Bay
	C02	53	2	48	21	0.0050	555	235	100	130	56	Broad Sound
	C03	23	16	32	21	0.0050	555	155	100	86	56	Mackay
B04	C01	81	24	27	13	0.0050	555	130	60	72	33	Proserpine
	C02	68	34	26	18	0.0050	555	125	85	69	47	Whitsunday Islands
	C03	38	22	31	19	0.0050	555	150	90	83	50	Bowen
B05	C01	67	3	38	16	0.0050	555	185	75	103	42	Ayr
	C02	32	6	37	11	0.0050	555	180	50	100	28	Townsville
	C03	27	15	21	13	0.0050	555	100	60	56	33	Palm Islands
B06	C01	122	2	29	13	0.0050	555	140	60	78	33	Cardwell
	C02	84	7	41	13	0.0050	555	200	60	111	33	Innisfail
	C03	70	10	15	12	0.0050	555	70	55	39	31	Cairns
	C04	45	5	26	11	0.0050	555	125	50	69	28	Port Douglas
B07	C01	98	41	39	16	0.0050	555	190	75	105	42	Cooktown
	C02	54	20	26	27	0.0050	555	125	130	69	72	Bathurst Bay
B08	C01	62	8	21	13	0.0050	555	100	60	56	33	Lockhart River
B09	C01	42	61	33	25	0.0050	555	160	120	89	67	Thursday Island
B10	C01	61	2	42	22	0.0050	555	205	105	114	58	Weipa
B11	C01	10	12	27	22	0.0050	555	130	105	72	58	Karumba
B12	C01	79	3	56	20	0.0050	555	275	95	153	53	Burketown
	C02	69	15	17	25	0.0050	555	80	120	44	67	Mornington Island

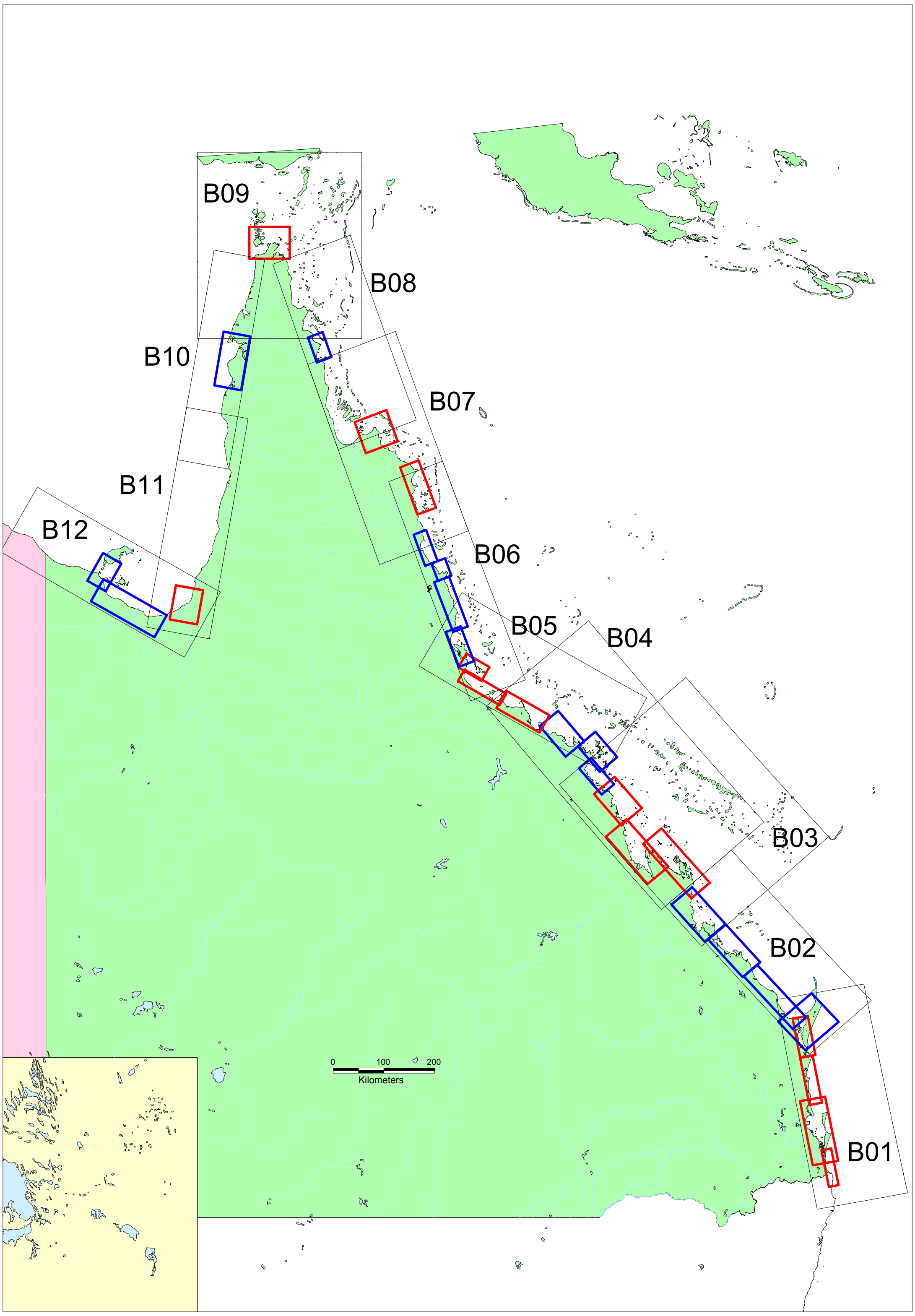


Figure 3.2 **B** and **C** domains for the Queensland coast

3.2 Estimated MEOW Run Times and Storage Requirements

The main study report describes the background to the following summary table, which provides an estimate of the run times and storage requirements. Actual times and storage will vary depending on the run time options chosen.

Table 3.3 CPU and disk storage requirements for MMUSURGE on the CIRRUS1 workstation.

Grid No.	Location	CPU Minutes for Duration				Disk Storage MB for Duration			
		24 h	48 h	72 h	96 h	24 h	48 h	72 h	96 h
A01	Southern Queensland	1.5	3.0	4.5	6.0	3.3	6.5	9.8	13.0
A02	Central Queensland	1.9	3.8	5.6	7.5	3.3	6.5	9.8	13.0
A03	Northern Queensland	1.8	3.6	5.5	7.3	3.3	6.5	9.8	13.0
A04	Gulf of Carpentaria	2.2	4.5	6.7	9.0	5.8	11.7	17.5	23.4
B01	Brisbane	3.7	7.3	11.0	14.7	11.3	22.7	34.0	45.3
B02	Hervey Bay - Gladstone	3.5	7.0	10.4	13.9	10.3	20.5	30.8	41.1
B03	Mackay	13.9	27.8	41.7	55.6	22.1	44.2	66.4	88.5
B04	Whitsunday	13.7	27.5	41.2	55.0	22.3	44.6	66.9	89.3
B05	Townsville	6.7	13.4	20.1	26.8	11.5	23.0	34.5	46.0
B06	Cairns	5.9	11.7	17.6	23.4	8.8	17.5	26.3	35.0
B07	Cooktown	8.0	16.0	24.1	32.1	12.9	25.8	38.8	51.7
B08	Lockhart River	8.2	16.3	24.5	32.7	10.7	21.3	32.0	42.6
B09	Torres Strait	12.3	24.6	36.9	49.2	20.3	40.6	60.9	81.2
B10	Weipa	1.7	3.4	5.2	6.9	7.3	14.5	21.8	29.1
B11	Karumba	2.1	4.2	6.4	8.5	9.1	18.2	27.3	36.4
B12	Mornington Island	2.4	4.7	7.1	9.4	10.1	20.2	30.3	40.4

Maketrack generates RUN files with *advection 0* and hence a saving of approximately 15% is likely on individual **A** and **B** grid runs relative to the individual times shown is possible.

Note: Following a modification to the field.f model subroutine on 10 October 2001, model run times may now be up to 3 times faster for situations with high reef coverage.

4. Detailed Operational Notes

4.1 Pre-Processing Software

4.1.1 Creation of MAPINFO-compatible display products

MAPINFO is used to enable visualisation and checking of the model domains and location of tracks, output locations etc. This requires any model definition data (BED files, thistory locations, CYC tracks etc) to be converted into formats which can be read by MAPINFO.

The program grid_tables.f (via alias *gtabs*) will read any BED file and create a series of TXT files with headers which can, after manual processing, be used to display in MAPINFO, e.g.

*_bor.txt file of the grid outline	--> *_border.TAB
*_xax.txt file of x axis	--> *_xaxis.TAB
*_yax.txt file of y axis	--> *_yaxis.TAB
*_msh.txt file of all grid lines	--> *_mesh.TAB
*_dep.txt file af all depoths	--> *_depth.TAB
*_lnd.txt file of all land points	--> *_land.TAB

where * represents a domain name e.g. B05 etc.

The actual series of steps required to produce these plots is complex and the user should consult the various readme.txt files in the $MEOW/maps directory tree. A series of WOR files are available which provide a base plotting environment for each domain.

After manual checking and additions to the model "coastline" thistory definitions a reverse process in carried out by make_coast.f (via alias *coast*) which reads a MAPINFO-compatible TXT file in the /maps directory system and reformats the time history locations into a MMUSURGE-compatible text file on $MMUCST, which is read by *maketrack*. The input and output filenames are hardwired into the program to ensure the correct filenames are produced for *maketrack*, e.g.

Input: $MEOW/maps/domain_data/b*/b*_cst.txt
Output: $MMUCST/b*_cst.his

4.1.2 Creation of "dummy" C Grid BED Files

This is facilitated by make_cgrid.f (alias *makec*), which accepts the series of parameters needed to define a dummy **C** grid, which is initially output to $MMUBED/c_grids and must then be manually copied to the production directoy $MMUBED.

The operation of *makec* is automated by an answer file c_coords.ans which is generated by c_coords.xls (from where Table 3.2 information is sourced). All the dummy C grid BED files are then able to be generated via

makec < c_coords.ans

4.1.3 Track File Generation

The utility program maketrack.f (via alias *maketrack*) greatly facilitates the creation of all necessary input data and control files to undertake large numbers of model simulations.

It defines a straightline track of constant parameter values relative to a "target site", as shown schematically in Figure 4.1, assuming:

- Each **B** grid has a "target site" from where the track is referenced
- A default fixed storm duration of 24 h; 18 h before "landfall" and 6 h after "landfall"
- "Landfall" is taken as the point of coastal crossing on the **B** grid (time=0)
- The **A** grid run only produces boundary data for the **B** grid
- The **B** grid run produces time history output along the entire defined "coastline" and ***huv*** output at –12, -6, -3, -2, -1, 0, +1, +2, +3, +6 h.
- The CYC file produces wind output only at the target site (for checking purposes)
- The default ambient atmospheric pressure is assumed to be 1008 hPa for Queensland

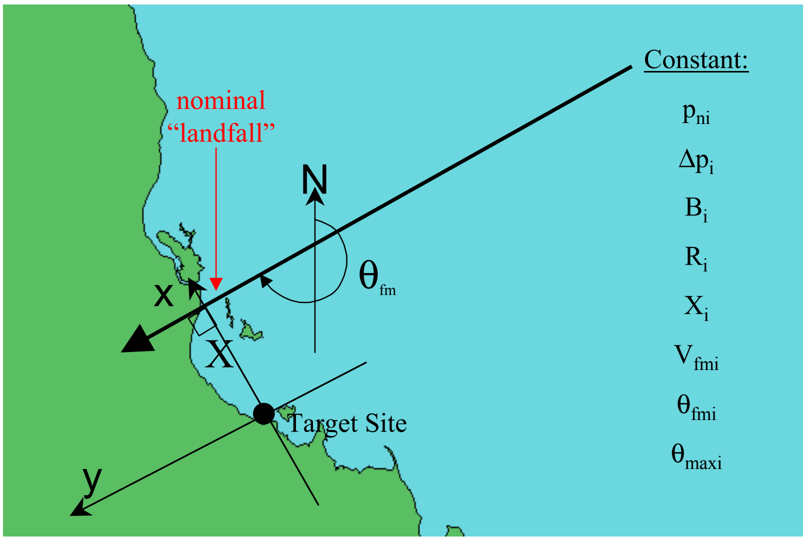


Figure 4.1 Maketrack automatic track generation.

Maketrack expects the following user input, either manually or as a parameter (.PAR) file:

Line 1: Identification line only
Line 2+: <Bnum>,<Deltap>,<R>,<B>,<X>,<Vfm>,<Thetafm>,<Thetamax>

where <Bnum> is the **B** grid name, e.g. B05. In PAR file mode optional parameters may be specified to change the default buildup time, overall duration of run and time after landfall.

The user-supplied track parameters are composited into a unique id-string to identify the input and/or output relevant to each track, e.g.

PPP_RRR_BBB_sXXX_VV_TTT_MMM

Where each fixed field is related to the supplied parameters as:

PPP	DeltaP	hPa
RRR	R	km

BBB	B*10	-	
sXXX	X	km	s=”p” for +ve X; s =”m” for –ve “X”; else s= “0”
TTT	Thetafm	deg	(bearing towards)
MMM	Thetamax	deg	(usually 115 or 065)

Input files needed by maketrack include:

1. $MMUBED/<B_num>.bed which is used to determine the coastline so that the storm track can be positioned making landfall at time 0 h.
2. $MMUCST/<B_num>_cst.his supplies the coastal thistory definition to be included in the RUN file

Supplying a PAR file will produce a batch control file on $MMUDOM which can be used to execute the generated set of model runs via the *at* command.

Output files produced by maketrack are prefixed by the applicable grid name and include:

	e.g.
A grid RUN file	$MMURUN/a01b01_040_030_015_m900_04_270_115.run
B grid RUN file	$MMURUN/b01c01_040_030_015_m900_04_270_115.run
CYC file	$MMUCYC/b02_040_030_015_m900_04_270_115.cyc
MAPINFO track file	$MMUMAP/b02_040_030_015_m900_04_270_115.txt
Batch control file	$MMUDOM/<infile>.ctl

Plus the following BPT filenames are composited for the RUN file:

e.g. a01b01_050_030_015_m500_04_270_115.bpt A -> B
b01c??_050_030_015_m500_04_270_115.bpt B -> C
(maketrack is hardwired with the number of C grids available for each B grid)

Maketrack performs a range of consistency and range checks on the supplied parameters and issues warning messages if necessary. The “coastline” is determined from the model boundary codes on the BED file and although logic is included which attempts to avoid “landfall” on offshore islands and some types of narrow capes, there may be cases where manual intervention may be desirable.

The **B** grid reference, or “target” sites, are arbitrary but easily locatable open coast sites defined as:

Grid	Target Site	Lat	Lon	I	J
B1	Cape_Moreton	-27.0278	153.4874	91	22
B2	Round_Hill_Head	-24.1584	151.9120	73	15
B3	Cape_Palmerston	-21.5232	149.4953	53	22
B4	Cape_Hillsborough	-20.8877	149.0548	103	30
B5	Cape_Bowling_Green	-19.3159	147.3877	80	16
B6	Cape_Grafton	-16.8608	145.9086	79	18
B7	Barrow_Point	-14.3496	144.6573	72	36
B8	Cape_Weymouth	-12.6292	143.4339	71	15
B9	Cape_York	-10.7000	142.5250	62	73
B10	Thud_Point	-12.9889	141.5837	78	16
B11	Nassau_River_Mouth	-15.8964	141.3619	90	12
B12	Point_Parker	-17.0419	139.1804	81	22

Figure 4.2 is an example of tracks generated for the Townsville domain.

Figure 4.2 Example MakeTrack generated storm tracks for Townsville.

4.2 MMUSURGE Model Operation

The basic sequence of operation of the model sub-system for production modelling is:

cd /ssurge/meow
. setup

[… manually determine the required parameter ranges required for the MEOW e.g. use $MEOW/tcstats/meow_stats.xls to produce a <B_num>/<output>.PAR file for maketrack]

setpath <top> <domain> <output>

maketrack < <parameter file>

The model mmusurge.f can be invoked via the alias *mmu* or, in the case of *maketrack* production runs, a CTL file specifies the absolute path to the executable and the name of the RUN file (without ".RUN"), e.g.

mmu <run-filename>	for manual execution, or
at –f <control-file> *now*	for automatic background execution and monitor progress via *ps x*

Refer to the JCU user guide (Mason and McConochie 2001) for details of the model input file requirements and general operation.

The BoM version has been enhanced to allow for the environment variable pathnames as specified in Section 2.3 but is otherwise identical to the as-supplied JCU version in terms of file formats, control file keywords etc.

4.3 Post-Processing Software

4.3.1 JCU MATLAB GUI

This is the JCU-supplied output display system. Refer to $MEOW/mmugui for details.

4.3.2 Binary to ASCII Conversion

In the absence of MATLAB, the following binary to ASCII converters are available:

- wav2asc.f (via alias *wav*) will read a nominated *.wav time history trace file and produce an ASCII file *.ASC which can be read by Excel (e.g. present MMUWAV.XLS procedure, but limited there to 10 sites only). An additional summary of the maximum values of H,U,V at each site is output to file *.max.
- huv2asc.f (via alias *huv*) will read a nominated *.uvl, *.vvl or *.hvl file and produce an ASCII dump.
- grid2coord.f (via alias *g2c*) is a utility program provided by JCU which allows conversion between grid coordinates (I, J) and real world (lat, lon). Principal usage is to assist in locating specific output locations.

4.3.3 OCTAVE Analysis Environment

OCTAVE is an OSF language system compatible with MATLAB™ which runs in an MS Windows environment. Refer http://www.octave.org for further details.

4.3.4 C Processing Environment

A series of utility programs are invoked to call the processing routines described in Section 2.8. The source code for these scripts resides in:

```
/ssurge/meow/
        param_model/
                basin01/
                        utils_src/
```

and are compiled using the build script bld_utils.b

The executable files for these utilities reside in:

```
/ssurge/meow/
        param_model/
                basin01/
                        utils_bin/
```

and are used as described in the following table:

Script	Purpose	Invoked via:
prelim.c	Creates text file of stations in a single wav file to be loaded into MAPINFO	prelim
prep.c	Copies ignorelist file to each data directory and reports any errors or missing directories	prep
make_runp.c	Creates the script that invokes all prepare scripts as listed in Section 2.8	make_runp

The final runp script is then invoked to perform the processing of all datafiles for a single basin. The output from this script is to the directories described in Appendix B Section 1.6.

5. Tropical Cyclone Statistical Analyses

A simplified tropical cyclone climatological analysis was undertaken to support the selection of appropriate and objective values for the model storm parameters required in each coastal region, e.g. tracks, intensities, forward speeds, *R* and *B* values. Use of statistically-based parameters should ensure the most efficient and most likely selection of parameters spanning the expected dynamic range in each region.

This was achieved through the use of the SEA climatology analysis program *ReadMET*, referencing a modified best track database of tropical cyclones for Queensland (as used in the parent study by Harper 2001). This differs from the current official BoM track database in a number of details, mainly related to removal of "east coast lows" and several duplicated storms. The *ReadMET* program was applied to a series of regions around the Queensland coast, as summarised in Table 5.1. All tropical cyclones since 1959/1960 which passed within a 500 km radius of the coast at these latitudes were extracted and their various parameters were analysed and summarised. Extreme value analyses of central pressure estimates within the radius were also performed using the SEA program *GUMMML*. A spreadsheet was then used to collate the information in terms of a basic latitudinal variability and smoothed lines of best fit were devised to summarise the emerging trends. Some specific assumptions were also applied and these are addressed in the discussion which follows. The analysis spreadsheet, which should be referenced for any specific details ($MEOW/tcstats/*meow_stats.xls*), also functions as a decision support tool (refer Section 5.7).

Table 5.1 Statistical analysis regions.

Region	Latitude
	°
Brisbane	-27.47
Hervey Bay	-25.25
Gladstone	-23.85
Mackay	-21.14
Townsville	-19.25
Cairns	-16.92
Cp Melville	-14.16
Cp York	-10.69
Holroyd R	-14.21
Burketown	-17.74

5.1 Central Pressure Variation

This is summarised in Figure 5.1, firstly in terms of the assumed *Maximum Potential Intensity* (MPI) and secondly as estimates of the variation of minimum central pressures for a range of Return Periods (or Average Recurrence Intervals, ARI).

The MPI estimates for present climate "sea to land" classes are based on advice from Holland (1996), while the "land to sea" class is based on advice from Callaghan (2001). The "Greenhouse" class applies to the "sea to land" class only and represents a nominal 10% increase in intensity relative to an assumed environmental pressure of 1008 hPa.

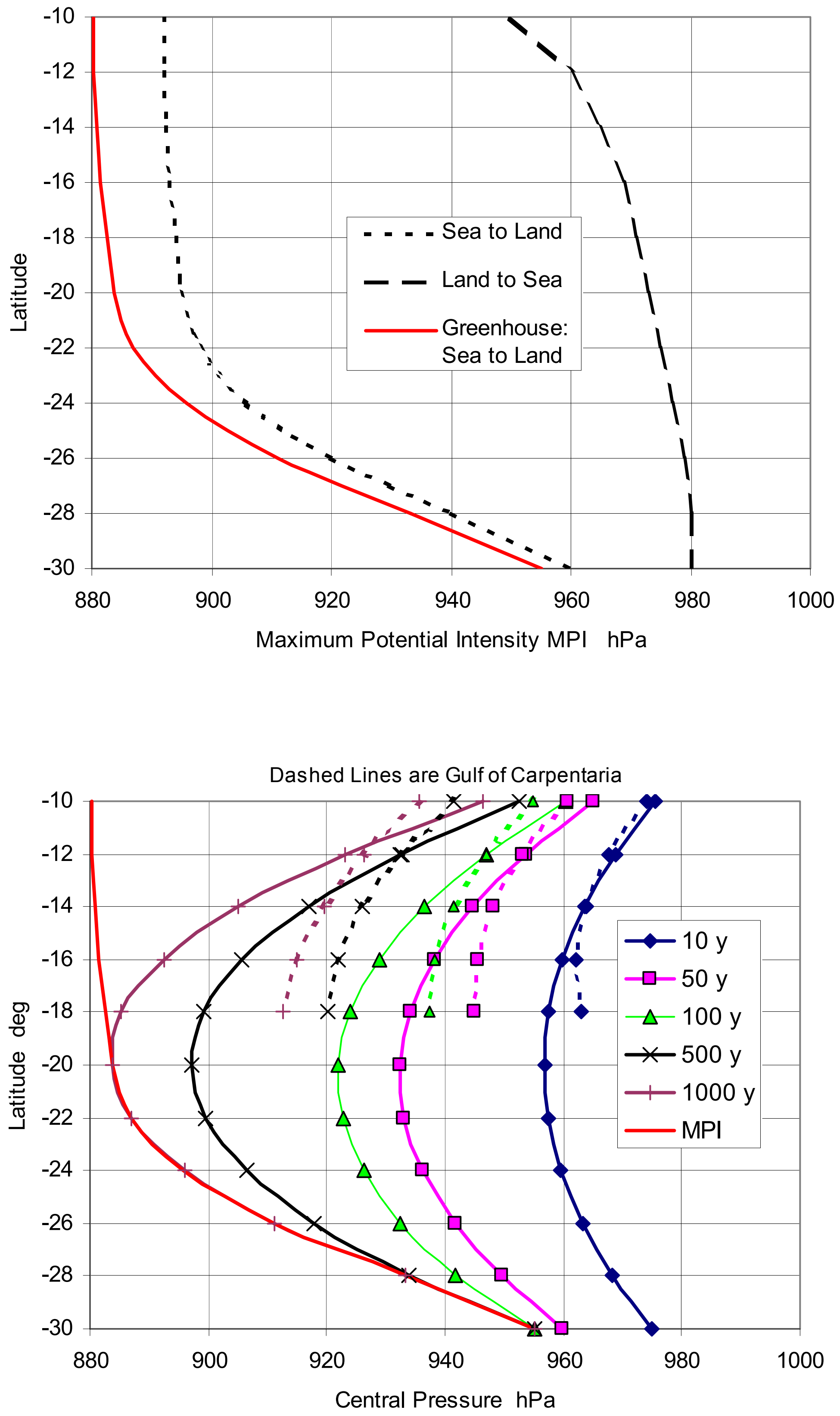


Figure 5.1 Variation in minimum central pressure.

The second graph presents the variation in the central pressure as a function of latitude, based on the extreme value analysis but limited to the Greenhouse MPI estimate. Note that separate curves are shown for east coast and Gulf of Carpentaria sub-regions and that these statistics apply to the minimum central pressure within 500 km radius of the indicated latitude at the coastline, irrespective of storm track.

5.2 Frequency of Occurrence by Track

For this description, each storm track was allocated to one of either:

- E-W moving,
- Parallel to coast, or
- W-E moving.

The individual track variability with latitude is shown in Figure 5.2 together with the total frequency of occurrence for all tracks combined. The E-W class is dominated by the east coast population, peaking in the vicinity of latitude -20° (Bowen). Likewise the parallel class frequency peaks near latitude –23° (Gladstone). The W-E class is comprised of contributions from Gulf of Carpentaria storms far to the north and also recurving storms further to the south.

Note that these statistics apply to any storm tracks within 500 km of the coast and not just to storms which may have crossed the coast.

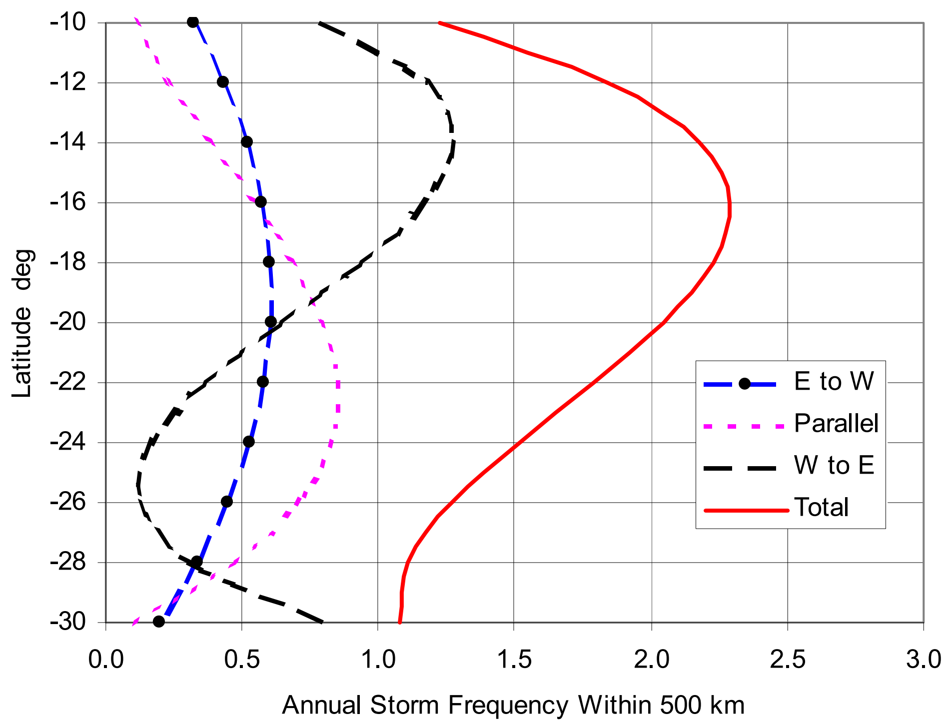


Figure 5.2 Frequency of occurrence analysis.

5.3 Track Characteristics

Figure 5.3 presents the typical variability in track bearing θ_{fm} (direction of motion relative to North) and forward speed V_{fm} for the E-W class of tracks. These are presented in terms of a mean line and (nominal) upper and lower standard deviations. The forward speed is reasonably constant with latitude for this class.

Figure 5.4 presents the Parallel class and Figure 5.5 presents the W-E class. The forward speed can be seen to become more variable with increasing latitude in each of these classes.

5.4 Radius to Maximum Winds R

This parameter is not available from the database in any appreciable numbers and so is estimated here based on local experience (Callaghan 2001) and the broad principles thought to influence radius to maximum winds. For example, it is generally accepted that there is a tendency towards higher *R* at higher latitudes, dependent on Coriolis and also storm age. The assessment here in Figure 5.6 ascribes a mean and standard deviation for both "Sea to Land" and "Land to Sea" classes, the latter being significantly limited in variation.

5.5 Windfield Peakedness B

This parameter can only be assessed on the basis of measured windspeed profiles and so is also not available in the database. Again, it is estimated here (Callaghan 2001) based on expected behaviour only. Figure 5.7 again separates "Sea to Land" and "Land to Sea" classes.

5.6 Correlated Parameters

In addition to the above, some thought has been given to the likely dependence of both *R* and *B* either on each other or with intensity. These relationships are presented in Figure 5.8 for *R* and *B* as functions of central pressure, although it is equally likely that they are related to each other directly.

The following "rules" have been considered when constructing these functions:

(a) Radius to Max Winds *R*

- Radius = *f*(intensity) for a given storm
- "Mature" storms e.g. 940 tend towards a "typical" 40 km eye diameter where an approximate *R* = eye_dia*0.5 + 5 km
- Distribution of *R* in any region will tend to reflect the intensity distribution
- The variation in *R* during a storm lifecycle will tend to occur within a "band" such that "small" storms tend to remain "small" in an overall sense, and "large" storms tend to remain "large" in an overall sense
- High intensity tends towards small *R*
- No eye until below 990 hPa hence $R > 40$ - 50 km until about 985 hPa
- Unlikely for $R < 10$ km for > 940 to 950 hPa
- Unlikely for $R > 40$ km for < 940 to 950 hPa

(b) Windfield Peakedness *B*

- Peakedness increases during intensification cycles
- Intensification cycles normally result in eye shrinkage =f(*R*)
- Holland suggests $1 < B < 2.5$ and *B* = constant – central pressure / 160

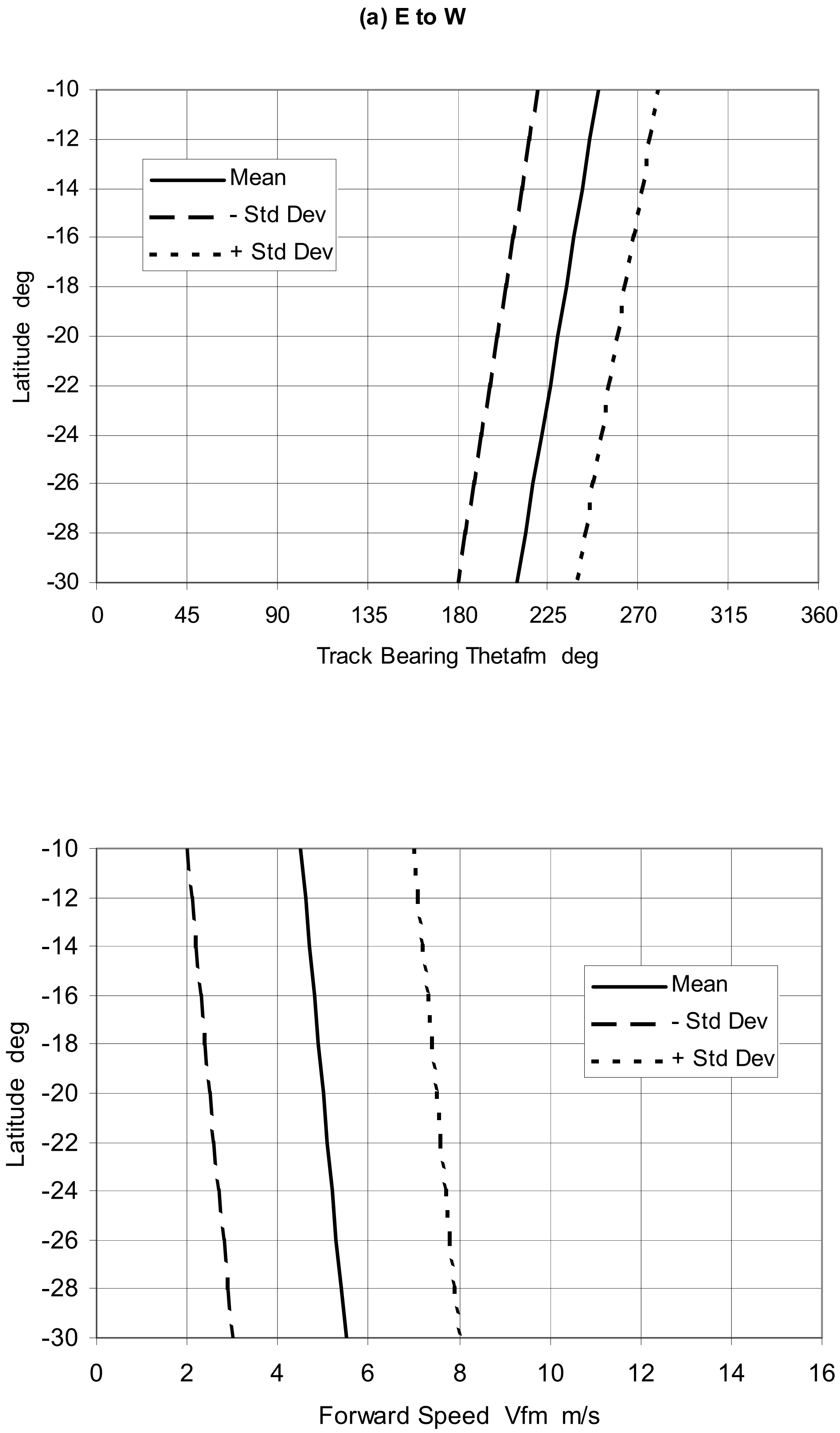
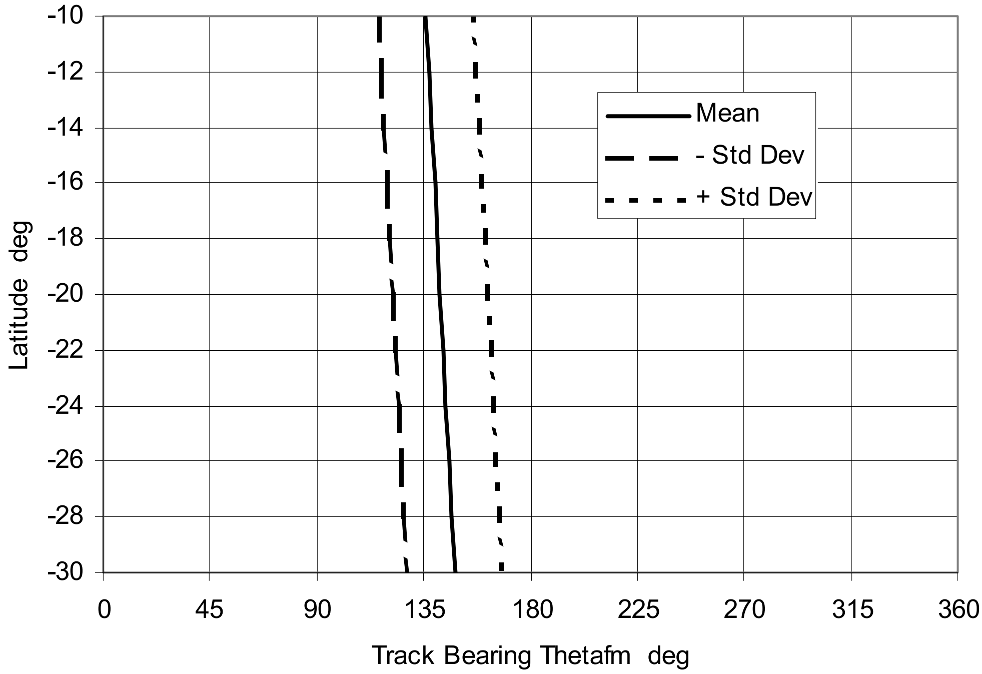


Figure 5.3 E-W storm track class.

(b) Parallel to Coast



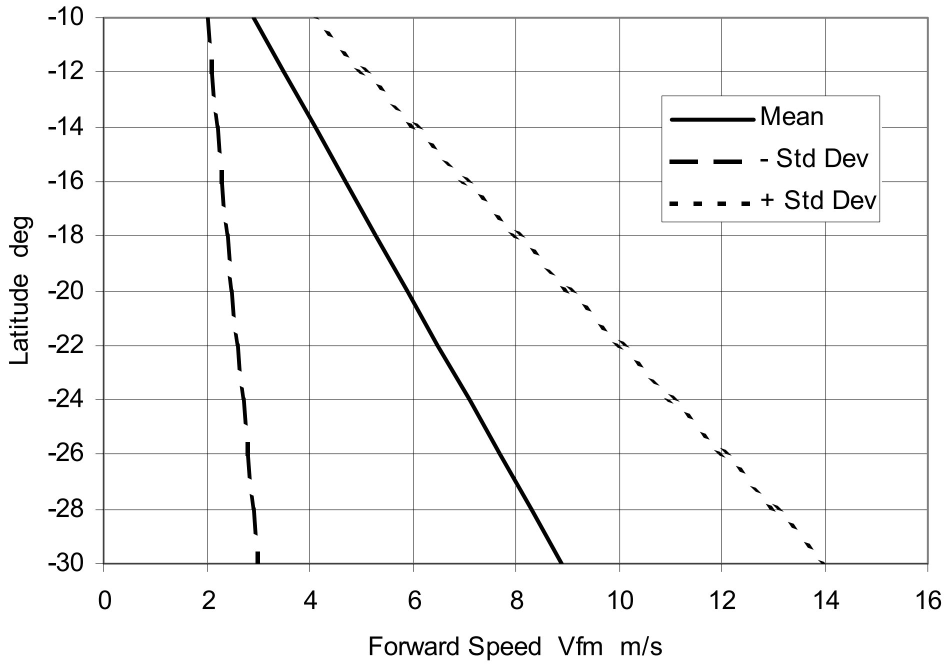
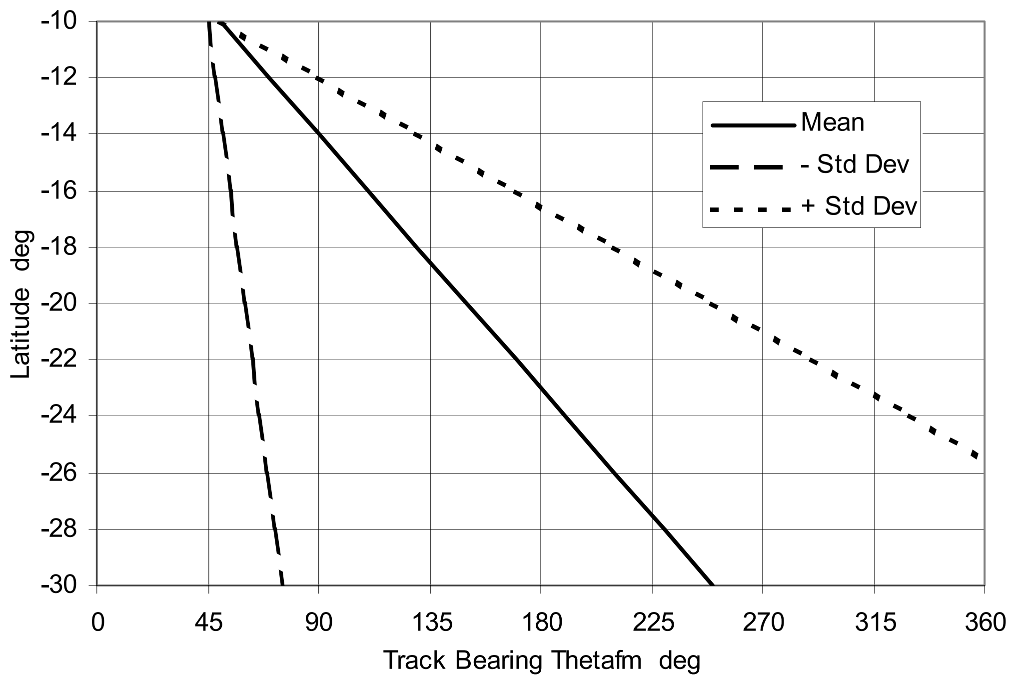


Figure 5.4 Parallel to coast storm track class.

(c) W to E



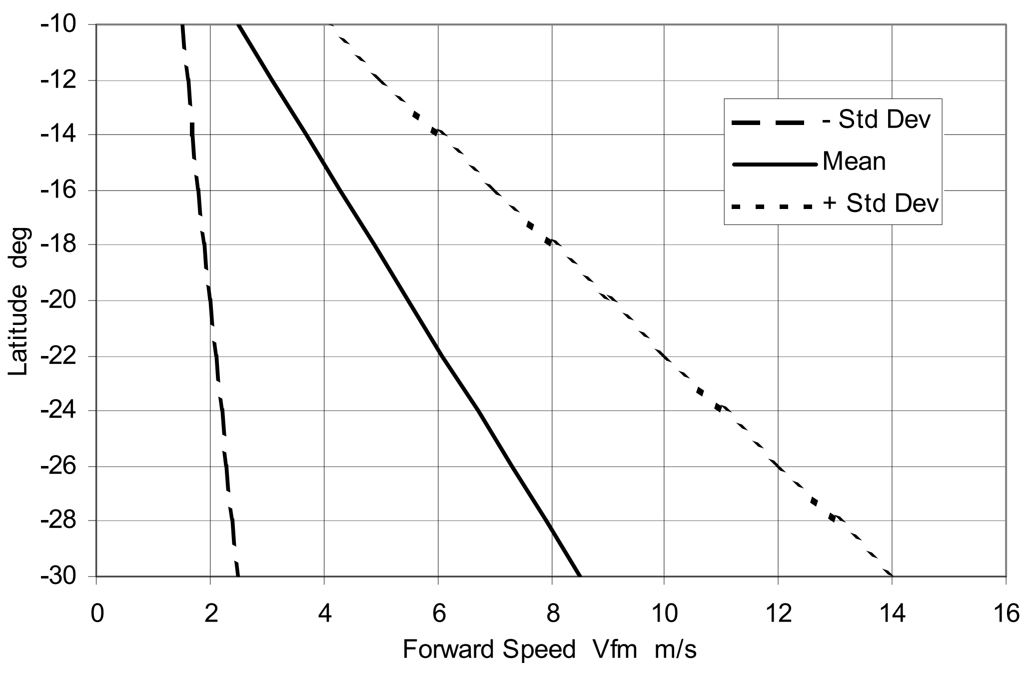
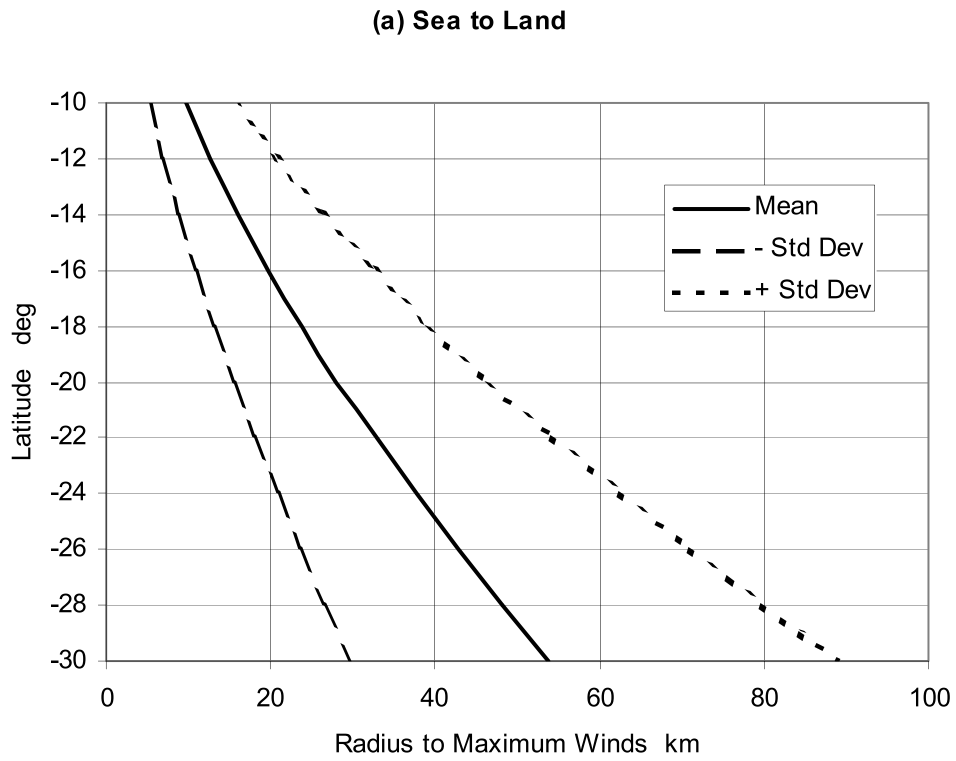


Figure 5.5 W-E storm track class.



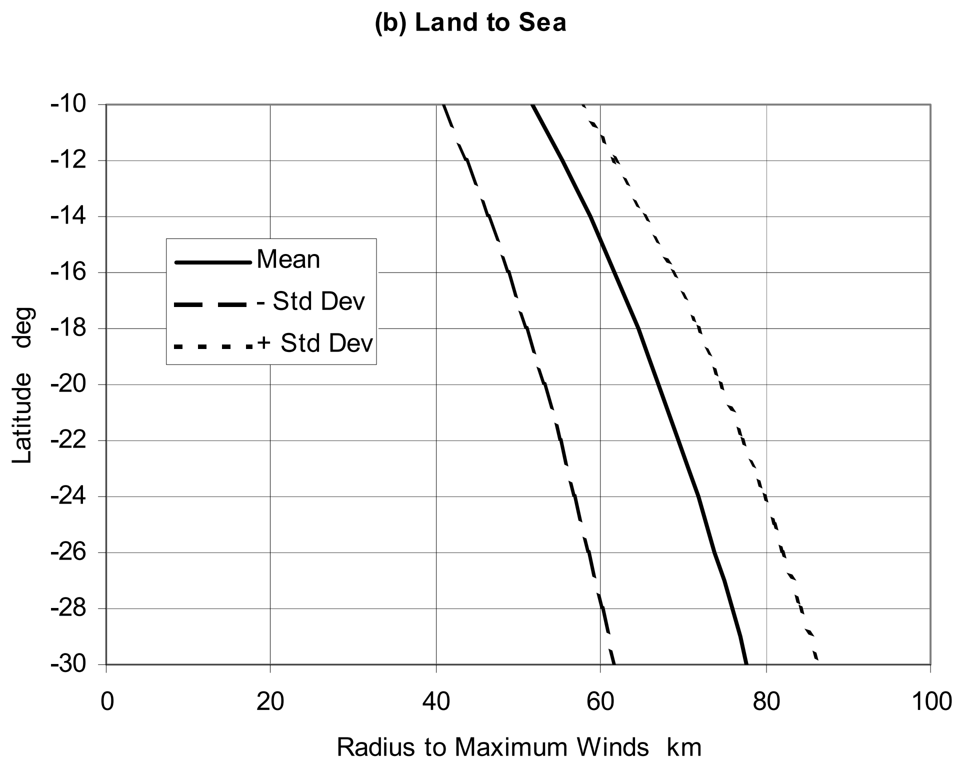
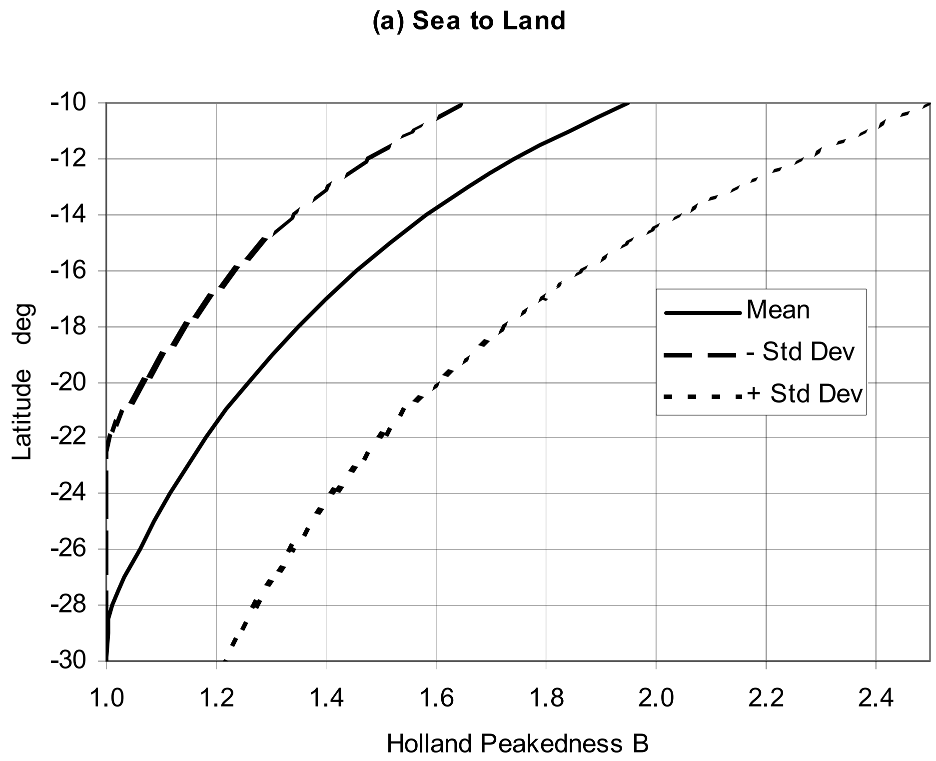


Figure 5.6 Radius to maximum winds *R*.



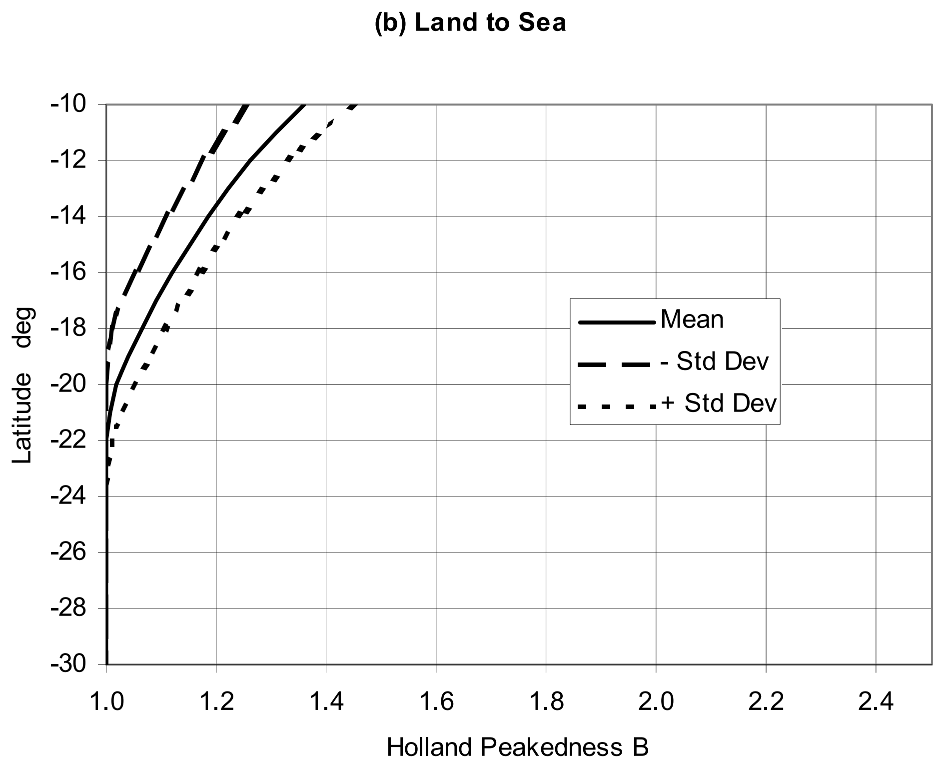
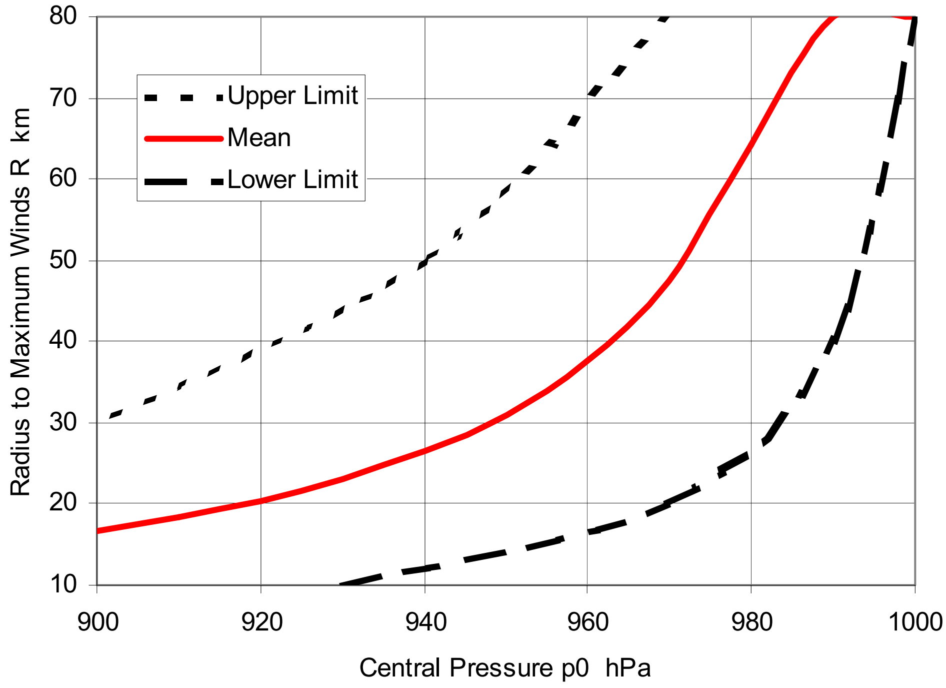


Figure 5.7 Windfield peakedness *B*.



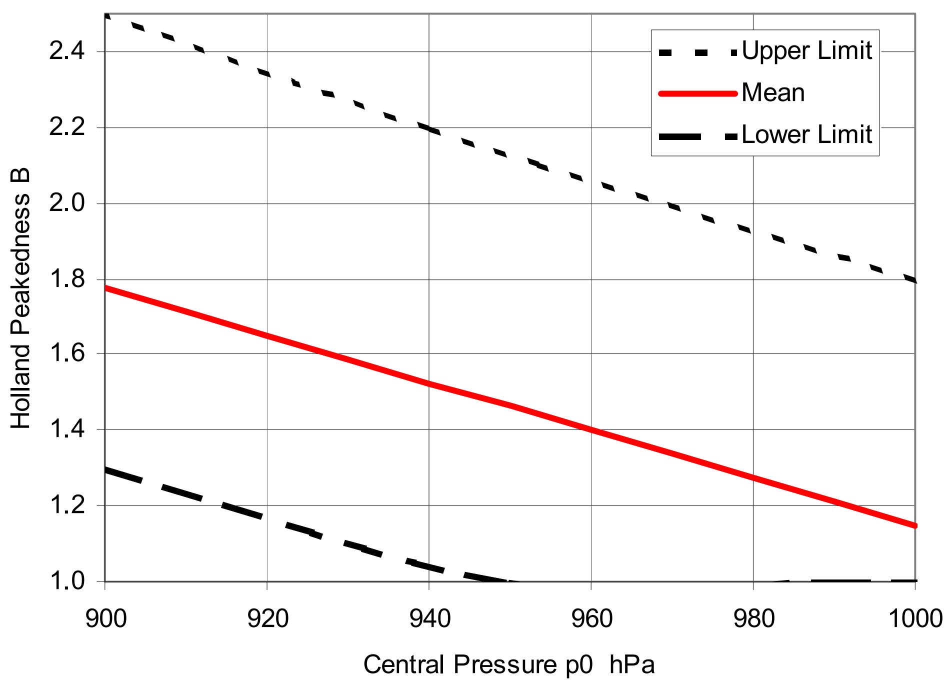


Figure 5.8 Correlated storm parameters.

5.7 Decision Support Tool

The spreadsheet *meow_stats.xls* has been designed to provide a decision support role for the selection of appropriate parameter sets for use by *maketrack* when setting up large scale MMUSURGE model runs.

This function is possible because all of the climatology relationships (as summarised in the foregoing figures) are derived from analytical functions, typically referenced by latitude. Accordingly, for any specific surge model domain's nominal latitude, a range of parameter values can be generated by the spreadsheet. These are then discretised into appropriate rounded value choices.

Table 5.2 presents a summary of these parameter ranges, which are able to be conveniently linked into formats suitable for preparation of a *.par* file for *maketrack*.

5.8 Adopted Domain Parameters

The choice of domain-specific tropical cyclone parameters is based on the indicated ranges from the decision support tool, summarised in **Table 5.3** below.

In the case of the central pressure p_c, the range is taken as the lower anchor, 50 y, 500 y and the MPI value, each converted to Δp based on an assumed p_n of 1008 hPa. In the case of θ_{fm} the mean and upper and lower standard deviations are chosen for the most severe coast crossing situation (W-E crossing for E coast, E-W for W coast etc)[1]. In the case of B, R and V_{fm} the mean and upper and lower standard deviations are chosen, with the mean values representing the so-called *base case* for these parameters. The spacing of the coast crossing points was fixed at 10 km but the actual start and end of coast crossing locations relative to the domain target sites was decided subjectively because of the overlapping domains and the desire for an even representation.

The domain tide level was normally set at MSL but a special set of sensitivity tests were also undertaken using a level slightly greater than HAT (refer Appendix B).

The indicated number of *base case* cyclone runs for each domain is also indicated in **Table 5.3**. These are calculated on the basis of the number of Δp times the number of X crossing locations, which are then repeated for each value of θ_{fm}, B, R and V_{fm} (refer Appendix B).

The model angle to maximum winds ($\boldsymbol{\theta_{max}}$) was fixed as 115°, based on the recommendation from Harper (2001).

[1] The case of the less severe parallel-to-coast tracks is yet to be considered.

Table 5.2 Decision support parameter selections.

East Coast			***E - W and Parallel***					user selectable values			hardwired values	
			Rounding	10	hPa			0.1			5	km
				Return Period				**Holland B**			**Radius to Maximum Winds**	
Domain	Target	Latitude	Anchor	50	500	MPI	- Std Dev	Mean	+ Std Dev	- Std Dev	Mean	+ Std Dev
		deg	hPa	y	y	hPa				km	km	km
B09	Cape_York	-11	985	960	940	880	1.6	1.8	2.4	5	10	20
B08	Cape_Weymouth	-13	985	950	920	880	1.4	1.7	2.1	10	15	25
B07	Barrow_Point	-14	985	940	920	880	1.3	1.6	2.0	10	15	25
B06	Cape_Grafton	-17	985	940	900	880	1.2	1.4	1.8	10	20	35
B05	Cape_Bowling_Green	-19	985	930	900	880	1.1	1.3	1.7	15	25	45
B04	Cape_Hillsborough	-21	985	930	900	890	1.0	1.2	1.6	15	30	50
B03	Cape_Palmerston	-22	985	930	900	890	1.0	1.2	1.5	20	35	55
B02	Round_Hill_Head	-24	985	940	910	900	1.0	1.1	1.4	20	40	60
B01	Cape_Moreton	-27	985	950	930	920	1.0	1.0	1.3	25	45	75
West Coast			***W - E and Parallel***					**Holland B**			**Radius to Maximum Winds**	
B12	Point_Parker	-17	985	940	920	880	1.2	1.4	1.8	10	20	35
B11	Nassau_River_Mouth	-16	985	950	920	880	1.2	1.5	1.9	10	20	30
B10	Thud_Point	-13	985	950	930	880	1.4	1.7	2.1	10	15	25

East Coast			***E - W and Parallel***						user selectable values				hardwired values	
					E - W				**E - W**					
					2	m/s			20	deg				
					Forward Speed Vfm				**Track Bearing Thetafm**					
Domain	Target	Latitude		- Std Dev	Mean	+ Std Dev		- Std Dev	Mean	+ Std Dev				
		deg		m/s	m/s	m/s		deg	deg	deg				
B09	Cape_York	-11		2	4	8		220	240	280				
B08	Cape_Weymouth	-13		2	4	8		220	240	280				
B07	Barrow_Point	-14		2	4	8		220	240	280				
B06	Cape_Grafton	-17		2	4	8		200	240	260				
B05	Cape_Bowling_Green	-19		2	4	8		200	240	260				
B04	Cape_Hillsborough	-21		2	6	8		200	220	260				
B03	Cape_Palmerston	-22		2	6	8		200	220	260				
B02	Round_Hill_Head	-24		2	6	8		200	220	260				
B01	Cape_Moreton	-27		2	6	8		180	220	240				
West Coast		***W - E and Parallel***			**W - E**				**W - E**					
					Forward Speed Vfm				**Track Bearing Thetafm**					
B12	Point_Parker	-17		2	4	8		60	120	200				
B11	Nassau_River_Mouth	-16		2	4	8		60	120	180				
B10	Thud_Point	-13		2	4	6		40	80	120				

East Coast			***E - W and Parallel***						user selectable values				hardwired values	
					Parallel				**Parallel**			**Combined Tracks**		
					2	m/s			20	deg				
					Forward Speed Vfm				**Track Bearing Thetafm**			**Track Bearing Thetafm**		
Domain	Target	Latitude		- Std Dev	Mean	+ Std Dev		- Std Dev	Mean	+ Std Dev		Min	Max	
		deg		m/s	m/s	m/s		deg	deg	deg		deg	deg	
B09	Cape_York	-11		2	4	4		120	140	160		120	280	
B08	Cape_Weymouth	-13		2	4	6		120	140	160		120	280	
B07	Barrow_Point	-14		2	4	6		120	140	160		120	280	
B06	Cape_Grafton	-17		2	6	8		120	140	160		120	260	
B05	Cape_Bowling_Green	-19		2	6	8		120	140	160		120	260	
B04	Cape_Hillsborough	-21		2	6	10		120	140	160		120	260	
B03	Cape_Palmerston	-22		2	6	10		120	140	160		120	260	
B02	Round_Hill_Head	-24		2	8	12		120	140	160		120	260	
B01	Cape_Moreton	-27		2	8	12		120	140	160		120	240	
West Coast		***W - E and Parallel***			**Parallel**				**Parallel**			**Combined Tracks**		
					Forward Speed Vfm				**Track Bearing Thetafm**			**Track Bearing Thetafm**		
B12	Point_Parker	-17		2	6	8		120	140	160		60	200	
B11	Nassau_River_Mouth	-16		2	4	8		120	140	160		60	180	
B10	Thud_Point	-13		2	4	6		120	140	160		40	160	

Table 5.3 Adopted model cyclone parameters for each domain.

Domain	**Name**	ΔP	θ_{fm}	***X*** **Tracks**	***B***	***R***	V_{fm}	**Tide**	**# Base Case**
		hPa	°	km	-	km	ms^{-1}	m MSL	
B01	SE Qld	23 58 78 88	200 220 240	m070 to p150	0.9 *1.0* 1.3	25 *45* 75	2.0 *6.0* 8.0	*0.0* 2.5	*88*
B02	Hervey Bay to Gladstone	23 68 98 108	200 220 260	m160 to p070	1.0 *1.1* 1.4	20 *40* 60	2.0 *6.0* 8.0	*0.0* 3.0	*92*
B03	Mackay	23 78 108 118	200 220 260	m200 to p130	1.0 *1.2* 1.5	20 *35* 55	2.0 *6.0* 8.0	*0.0* 5.5	*132*
B04	Whitsunday	23 78 108 118	200 220 250	m080 to p240	1.0 *1.2* 1.6	15 *30* 50	2.0 *6.0* 8.0	*0.0* 3.5	*128*
B05	Townsville	23 78 108 128	200 240 260	m060 to p180	1.1 *1.5* 1.7	15 *25* 35	2.0 *4.0* 8.0	*0.0* 2.5	*96*
B06	Cairns	23 68 108 128	200 240 260 280	m135 to p125	1.2 *1.3* 1.8	10 *20* 35	2.0 *4.0* 8.0	*0.0* 2.5	*104*
B07	Cooktown	23 68 88 128	220 240 280	m140 to p140	1.3 *1.6* 2.0	10 *15* 25	2.0 *4.0* 8.0	*0.0* 2.5	*112*
B08	Lockhart River	23 58 88 128	220 240 280	m080 to p130	1.4 *1.7* 2.1	10 *15* 25	2.0 *4.0* 8.0	*0.0* 2.5	*84*
B09	Torres Strait	*(not yet considered)*							
B10	Weipa								
B11	Karumba								
B12	Mornington Island								

6. Basis of the Parametric Storm Surge Model

The present methodology is an extension and generalisation of the approach originally devised for the Beach Protection Authority's Storm Tide Statistics project in the early 1980s (Harper and McMonagle 1985).

6.1 Premise

The storm surge response η at any coastal site *(x,y)* can be represented by a combination of the incident tropical cyclone parameters *{TC}* and the characteristics of the coastal basin *{Basin}*, *viz*

$$\eta = \boldsymbol{f}[\ \{TC\}\ ,\ \{Basin\}\ ]$$

where

$$\{TC\} = \boldsymbol{f}[\ \Delta p\ ,\ R\ ,\ B,\ V_{fm}\ ,\ \theta_{fm}\ ,\ \theta_{max}\ ]$$
$$\{Basin\} = \boldsymbol{f}[\ l\ ,\ \chi_b\ ,\ \chi_c\ ]$$

and l = perpendicular distance from the storm track centreline to *(x,y)*

χ_b = the *global basin* characteristics, e.g. depths, shelf width, slope, reefs etc

χ_c = the *local coastal* influences, e.g. bays, capes etc

This is illustrated by considering a hypothetical straightline coast basin with invariant shelf and slope characteristics, as shown in Figure 6.1 below.

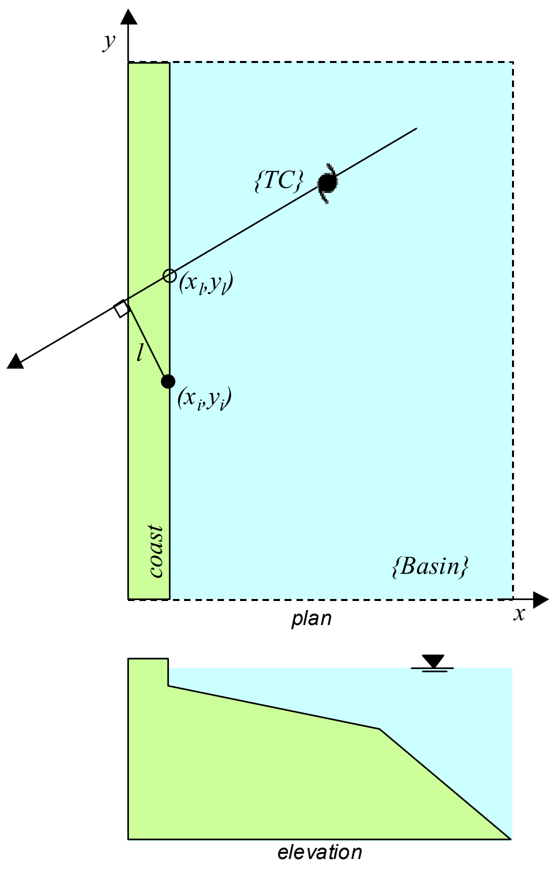


Figure 6.1 Schematic of a tropical cyclone impinging on a hypothetical straightline uniform basin.

In Figure 6.1, the location (x_i,y_i) is any coastal location of interest in the schematic *{Basin}*, situated a distance l perpendicular from the (assumed straightline) track of a tropical cyclone with characteristics *{TC}*, which makes landfall at location (x_l,y_l).

The expected shape of the time history of surge height $\eta(t)$ at (x_i,y_i) is illustrated in Figure 6.2 below.

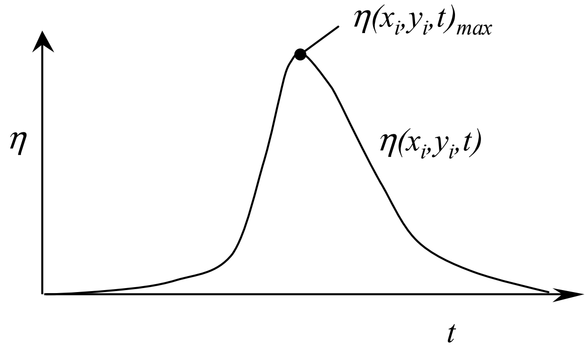


Figure 6.2 Surge height time history at any coastal location.

In a real-world basin the coastline is irregular and the offshore bathymetry varies, such that χ_c is not constant, and

$$\eta(x_i,y_i,t)_{max} \quad = \quad f\,[\ \{TC\},\ \chi_b\ ,\ l\ ,\ \chi_c\ ]$$

However, for the schematic uniform basin, it matters not where the point of interest lies but only the relative distance between the point and the storm track. In that case χ_c is constant, hence

$$\eta(x_i,y_i,t)_{max} \quad = \quad f\,[\ \{TC\},\ \chi_b\ ,\ l\ ]$$

Now consider the maximum envelope of surge height throughout the schematic basin after a surge event has occurred, where the peak "open coast" surge height (not affected by χ_c) is η_p.

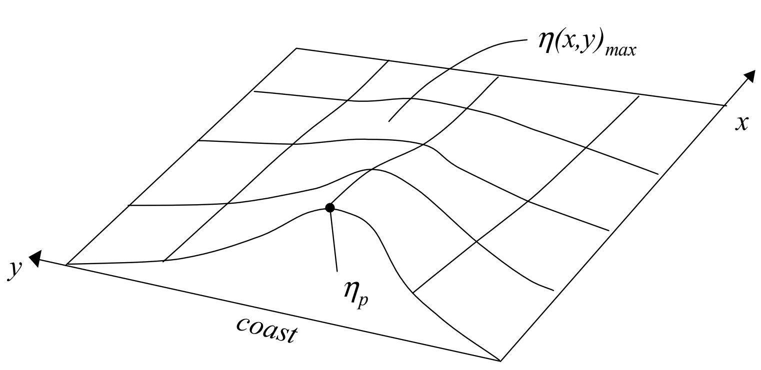


Figure 6.3 Maximum envelope of surge height.

It is then assumed that η_p within any specific basin will, to first order accuracy, be a joint function of storm intensity and track; viz

$\eta_p = \boldsymbol{O_1} [\, \Delta p \mid \theta_{fm} \mid \chi_b \,]$

However, θ_{fm} *and* χ_b may be essentially combined because of the interaction between storm track and line of coastline, with χ_b essentially being bound to that result, hence

$\eta_p = \boldsymbol{O_1} [\, \Delta p \mid \theta_{fm} \,]$

The remaining parameters are then assumed to be second-order and independent, i.e.

$\eta_p = \boldsymbol{O_1} [\, \Delta p \mid \theta_{fm} \,] + \boldsymbol{O_2} [\, R , B , V_{fm} , \theta_{max} \,]$

This is then evaluated numerically as follows:

$\eta_p = h_p (\, \Delta p , \theta_{fm} \,) \times F_R \times F_B \times F_{Vfm} \times F_{\theta max}$

where h_p is a joint response function of Δp and θ_{fm} based on a set of reference values for R, B, V_{fm}, and θ_{max} which yields the peak open coast surge magnitude.

F_R , F_B , F_{Vfm} and $F_{\theta max}$ are then dimensionless open coast response scaling factors or multipliers, relative to the adoped reference value for each parameter in the h_p function, e.g.

$F_R = [h_p]_{R=R'} \;/\; [h_p]_{R=R*}$

Where $R = R*$ is the reference value for R used in the definition of the h_p function (selected from mean climatology) and R' is the actual value required.

Hence, in summary

$h_p = \boldsymbol{f}(\, \Delta p , \theta_{fm} \,)_{R=R*,\, B=B*,\, Vfm=Vfm*,\, \theta max=\theta max*}$

which is obtained via numerical experimentation to yield, e.g.

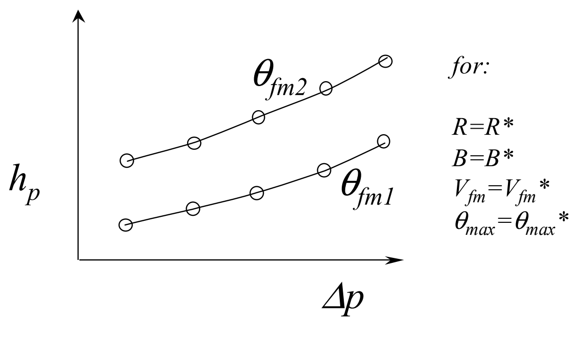


Figure 6.4 Construction of the base peak surge response function h_p.

6.2 Spatial Distribution of Open Coast Surge Magnitude

The spatial distribution of the open coast surge magnitude is then assumed to be a joint function of the following variables:

$h_l = \boldsymbol{f}\,[\,l \mid \theta_{fm} \mid R\,]$

and expressed as a normalised function, as illustrated schematically in Figure 6.5 below.

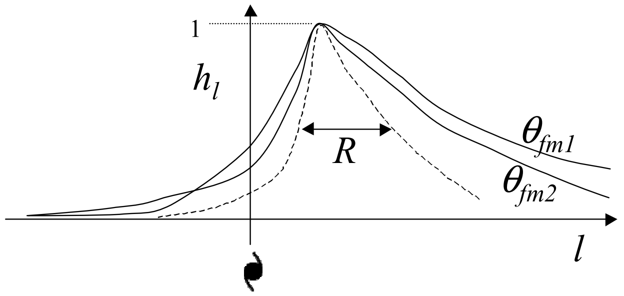


Figure 6.5 Alongshore open coast surge magnitude distribution h_l.

It is argued that *R*, being the essential horizontal scale parameter of the tropical cyclone, will likewise scale the alongshore "width" of the surge magnitude.

6.3 Temporal Variation in Open Coast Surge Magnitude

The temporal distribution of the open coast surge magnitude is then assumed to be a joint function of the following variables:

$h_t = \boldsymbol{f}\,[\,t \mid \theta_{fm} \mid V_{fm}\,]$

and expressed as a normalised function, as illustrated schematically in Figure 6.6 below.

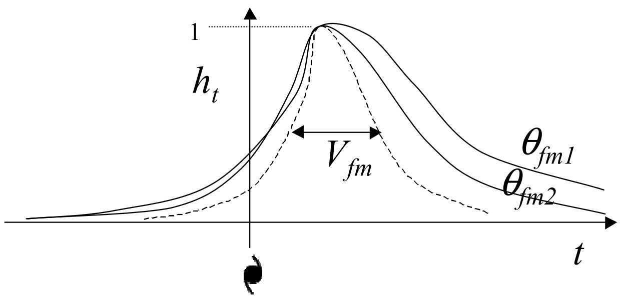


Figure 6.6 Temporal variation in open coast surge magnitude h_t.

It is argued that V_{fm}, being the essential scale parameter of the tropical cyclone movement, will likewise scale the temporal "width" of the surge hydrograph.

6.4 Local Coastal Influences

The site specific coastal influences χ_c are then assumed to act locally in modifying the open coast surge magnitude and timing, whereby

F_χ = a dimensionless surge magnitude multiplier; and

ϕ_χ = a relative phase shift value in hours.

with the expected variation in each of these being schematised in Figure 6.7 as a function of the relative alongshore position y. F_χ is found from experiment to be best represented as a local function of l but is represented schematically here by its mean value (refer later).

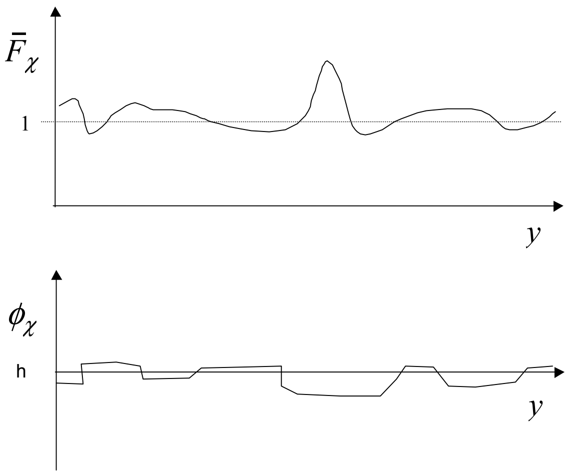


Figure 6.7 Localised open coast surge magnitude and phase modification.

6.5 Combined Storm Surge Response

The combined storm surge time history at any coastal location (x,y) can then be expressed in terms of a combination of each component, as illustrated in Figure 6.8:

$$\eta(x, y, t) = \eta_p \, . \, h_l(l, \theta_{fm}, R) \, . \, h_t(t + \phi_\chi(y), \theta_{fm}, V_{fm}) \, . \, F_\chi(y,l)$$

where

$$\eta_p = h_p(\Delta p, \theta_{fm}) \, . \, F_R \, . \, F_B \, . \, F_{Vfm} \, . \, F_{\theta max}$$

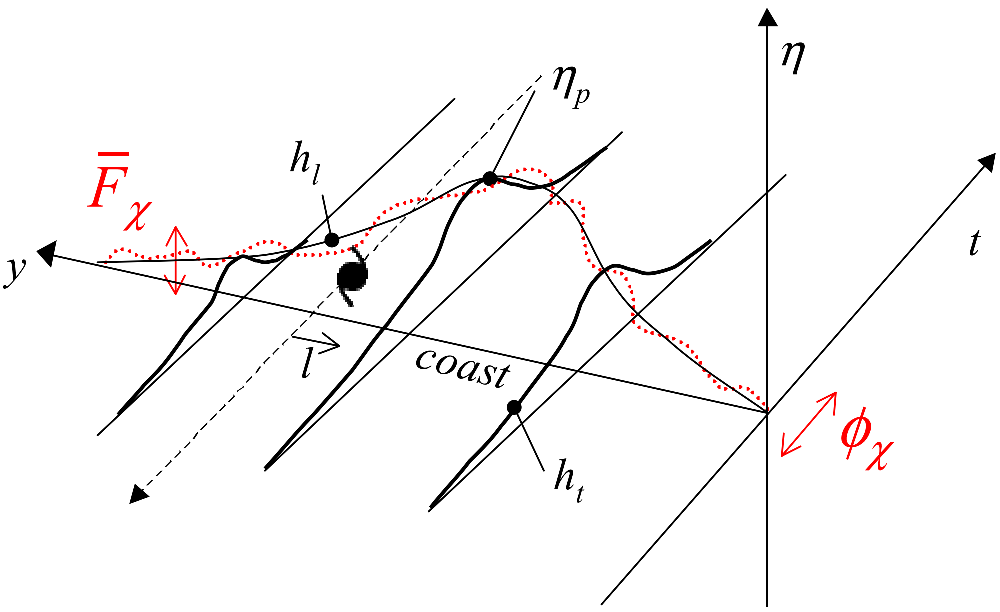


Figure 6.8 Combined storm surge response including local coastal effects.

6.6 Comparison with 1985 Parametric Model

The following describes the "1985 model" (Harper and McMonagle 1985) in terms of the present nomenclature:

F_R = 1 R^* = 30 km
F_B = 1 B^* = 1.0
F_{Vfm} = Jelesnianski formula based on V_{fm}^* = 7.8 m/s (28 km/h)
$F_{\theta max}$ = 1 θ_{max} = 115°

$h_l = \boldsymbol{f}\,[\,l \mid \theta_{fm}\,]$

$h_t = \boldsymbol{f}\,[\,t \mid \theta_{fm}\,]$

$F_\chi (y)$ = derived from amplification response due to a single coast-parallel storm; and

$\phi_\chi (y) = 0$

7. Extraction of Parametric Forms

7.1 General Approach

Construction of a parametric storm surge model for any given {Basin} then proceeds as follows:

- Establish optimum parameter ranges
- Devise sequence of numerical experiments
- Perform the modelling
- Analyse the results systematically

The analysis sequence then proceeds in reverse order to the theoretical development of Section 1, concentrating on determining the *local coastal* response functions first so that their influence can be removed from the full response and so obtain the equivalent *open coast* response. Appendix B provides a more detailed step-by-step programming description of this process as it has been implemented by Queensland Regional Office.

7.2 Local Alongshore Coastal Response

For this purpose, it is necessary to examine the surge response variation over the range of sites of interest (i.e. y) for each set of fixed storm reference parameters, e.g. Δp, R, B, V_{fm}, θ_{fm}, θ_{max}. To this end a series of closely spaced storm tracks (X) are devised for the region and processing of the results proceeds as follows:

Step 1 : Smoothing of the alongshore maximum surge envelope

This consists of assembling all $\eta\ (x,y,t\ )_{max}$ for the case of $x=x_1,x_2,x_3\ ...\ x_n$ and $y=y_1,y_2,y_3\ ...\ y_n$ being those points defining the contiguous open coastline (but excluding islands for the moment) and at the time of the local maximum value of η. To assist in schematising the results, the (x,y) spatial references are firstly converted to the storm landfall reference l . Because of this spatial transformation, which interacts with the storm track angle and the actual coastline shape, it is possible that l will not be unique and not monotonically increasing. Accordingly, the alongshore series is sorted in l and the maximum envelope is sampled on the basis of a defined resolution interval δl and the mean η_{max} is found within the interval. This is illustrated in Figure 7.1 below.

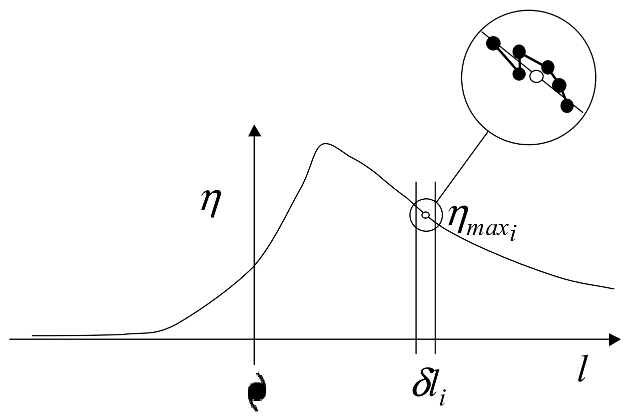


Figure 7.1 Sampled alongshore maximum surge envelope for a single storm event.

Multiple instances of the alongshore profiles are then combined from all of the closely spaced storm tracks. These are also averaged within each interval δl to form a *mean alongshore maximum envelope* which is considered to represent the "open coast response" for the basin. Figure 7.2 refers.

Step 2 : Local point specific alongshore coastal effect

The residual difference between the mean alongshore maximum and the local *(x,y)* maximum is then deemed to represent the influence of the local point-specific coastal effects $F\chi$

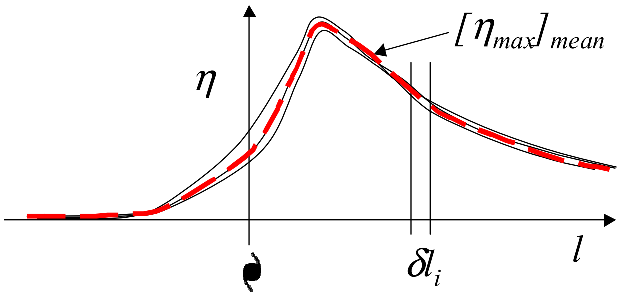


Figure 7.2 Mean alongshore maximum envelope formed from multiple tracks.

Referring to Figure 7.3,

$F\chi_i = (\delta\eta_i + \eta_i) / \eta_i$

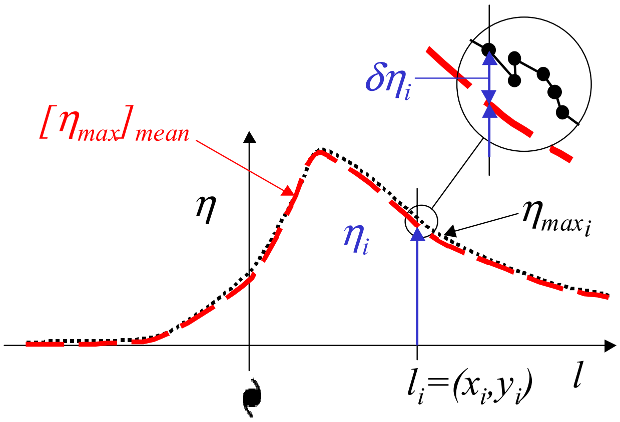


Figure 7.3 Derivation of local coastal response.

$F\chi_i$ was found to be best retained as a local function of *l*, with a typical site specific response function as shown in Figure 7.4.

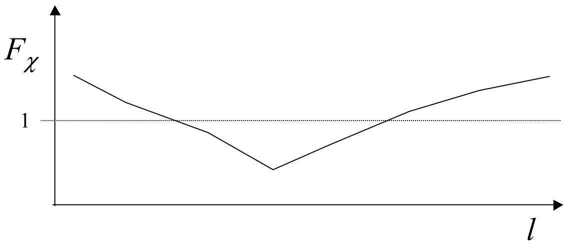


Figure 7.4 Typical alongshore local response factor.

Step 3 : Average the alongshore response for all central pressures

Step 2 is repeated for all central pressure track sets and the alongshore profiles and local response factors are averaged to remove central pressure as an independent variable.

7.3 Time History Response

This is analogous to Step 2 for the alongshore case but carried out in the time domain for each coastal point of interest. The starting context is many storms, each with a constant track θ_{fm} and constant intensity Δp but different coast crossing points.

Firstly, the time response at each point for each storm is normalised by its individual peak surge value to give $ht_i(l,t)$. Then the relative timing of the peak surge $\phi_{\chi i}$ at each point i is used to time shift the normalised time profile ht_i to the same zero time position and the average time profile $ht(l,t)_{mean}$ is calculated for each δl slice along the coast (refer Figure 7.5). The averaged phase offset $\phi(l)_{mean}$ and averaged absolute peak surge $\eta(l)_{mean}$ for each δl slice are also retained at this point.

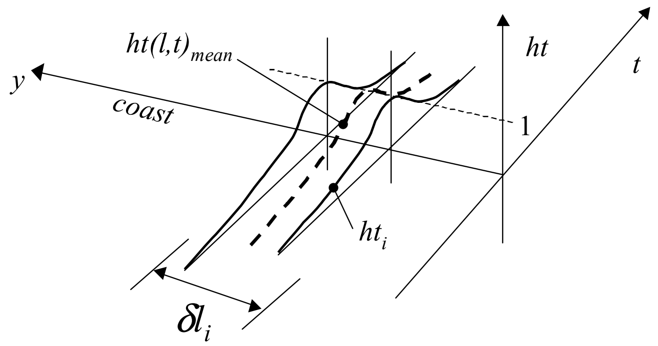


Figure 7.5 Mean time history curve formed from many tracks at δl resolution.

It is then assumed that all coast points will have a similar shape to their time history curve and the weighted average (by their respective $\eta_{max\ i}$) is then taken to yield $[h_t]_{mean}$, being the best estimate of the combined shapes. The residual difference between each ht_i and $[h_t]_{mean}$ then yields a local time shape factor which, referring to Figure 7.6, is

$$Ft_{i,j} = (\delta h_{t\,i} + h_{t\,i,j}) / h_{t\,i,j}$$

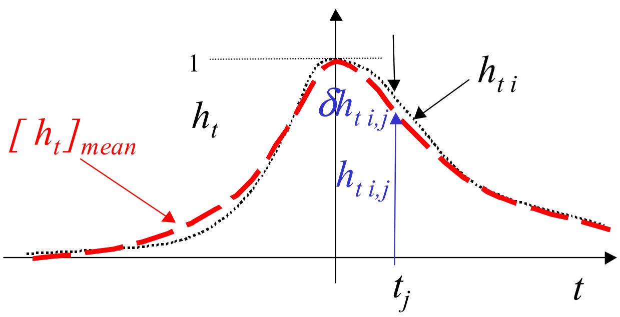


Figure 7.6 Calculation of local time shape factor.

By definition, $Ft_{i,j} = 1$ at t=0.

The $[h_t]_{mean}$ and $Ft_{i,j}$ are then further averaged across the range of different storm intensities.

7.4 Assessing the Sensitivity to Other Parameters

The foregoing development assumes that the model output for each domain is processed firstly in groups of θ_{fm} and then groups of Δp, with constant values of R, B and V_{fm}. To explore the possible behaviour of these other variables, the processing is simply repeated as many times as necessary using the chosen parameter ranges. For example, the mean value of each parameter is chosen for the *base case* and the upper and lower standard deviations can be chosen for the sensitivity runs. To enable linear interpolation between all the possible parameter values, only one value is changed at a time.

If sensitivity to mean water level is also to be investigated, an additional set of tests can be undertaken. However, assuming this effect is global and is expected to be small, it is recommended that only a small series of tests need be undertaken for a representative near-straightline section of the coast in each domain. For example, a series of three Δp on a fixed track with other base case parameters should suffice. If a large sensitivity is uncovered then further tests should be undertaken.

8. References

Callaghan J. (2001) *Personal communication.* Bureau of Meteorology, Queensland Regional Office, Brisbane.

Harper B.A. and McMonagle C.J. (1985) Storm tide statistics - methodology. Report prepared by *Blain Bremner and Williams Pty Ltd* for the Beach Protection Authority Queensland, Jan.

Harper B.A. (*Ed.*) (2001) Queensland climate change and community vulnerability to tropical cyclones: ocean hazards assessment - stage 1. Report prepared by Systems Engineering Australia Pty Ltd in conjunction with James Cook University Marine Modelling Unit, *Queensland Government*, March, 382pp.

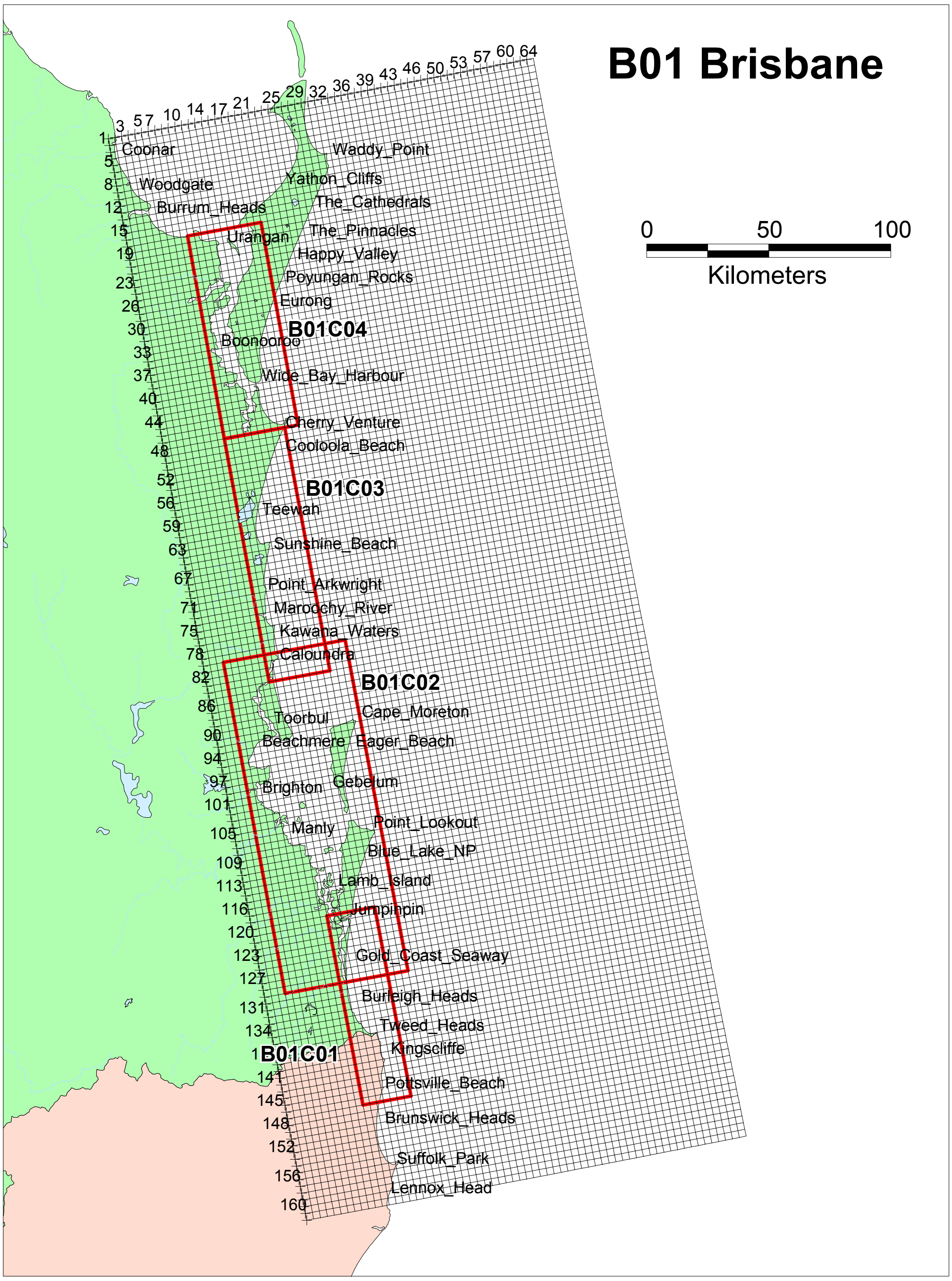
Harper B.A. (2004) Queensland climate change and community vulnerability to tropical cyclones: ocean hazards assessment – synthesis report. Report prepared by Systems Engineering Australia Pty Ltd, *Queensland Government*, March, 31pp.

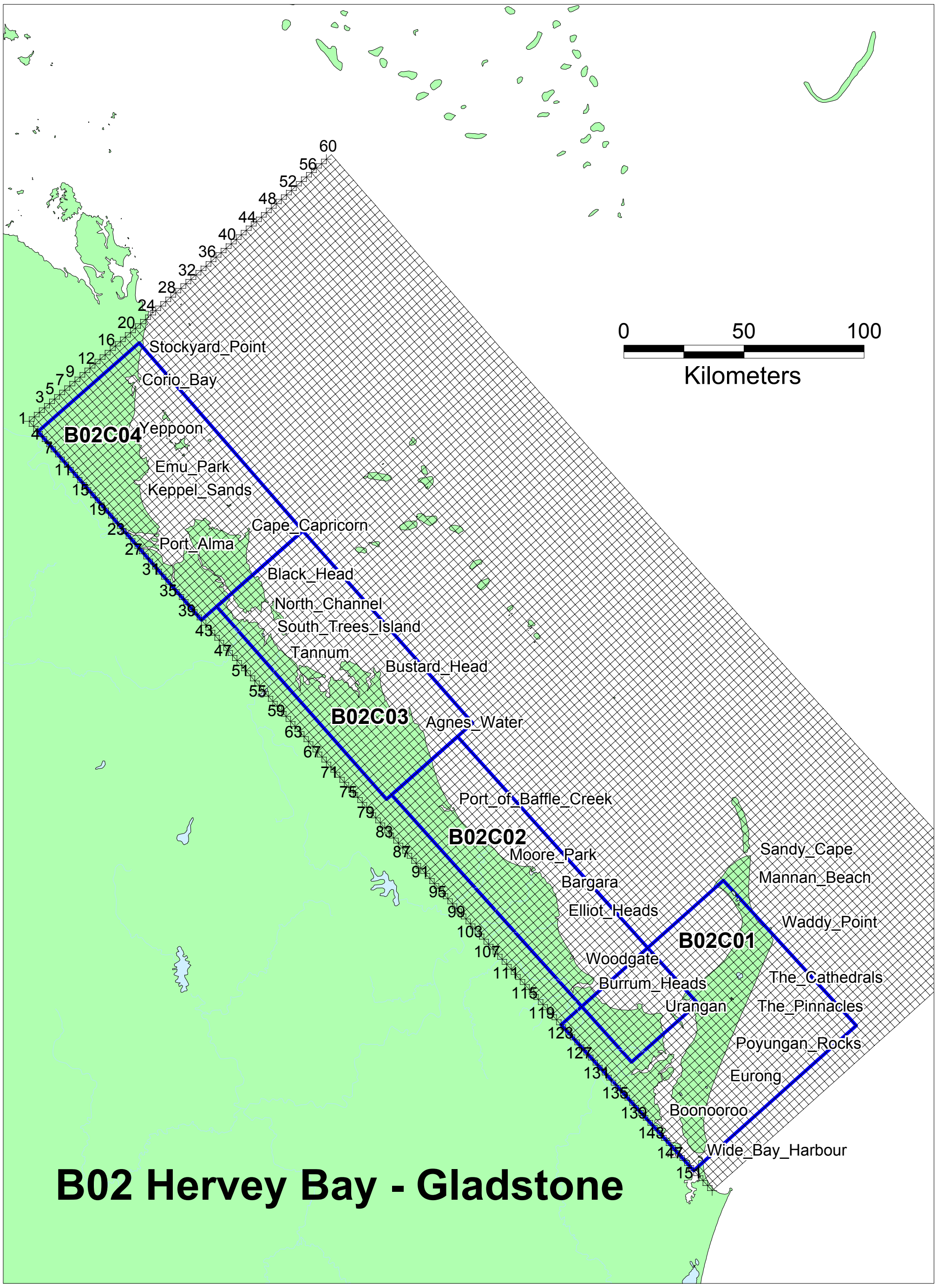
Holland G.J. (1997) *Personal communication.* Bureau of Meteorology Research Centre, Melbourne.

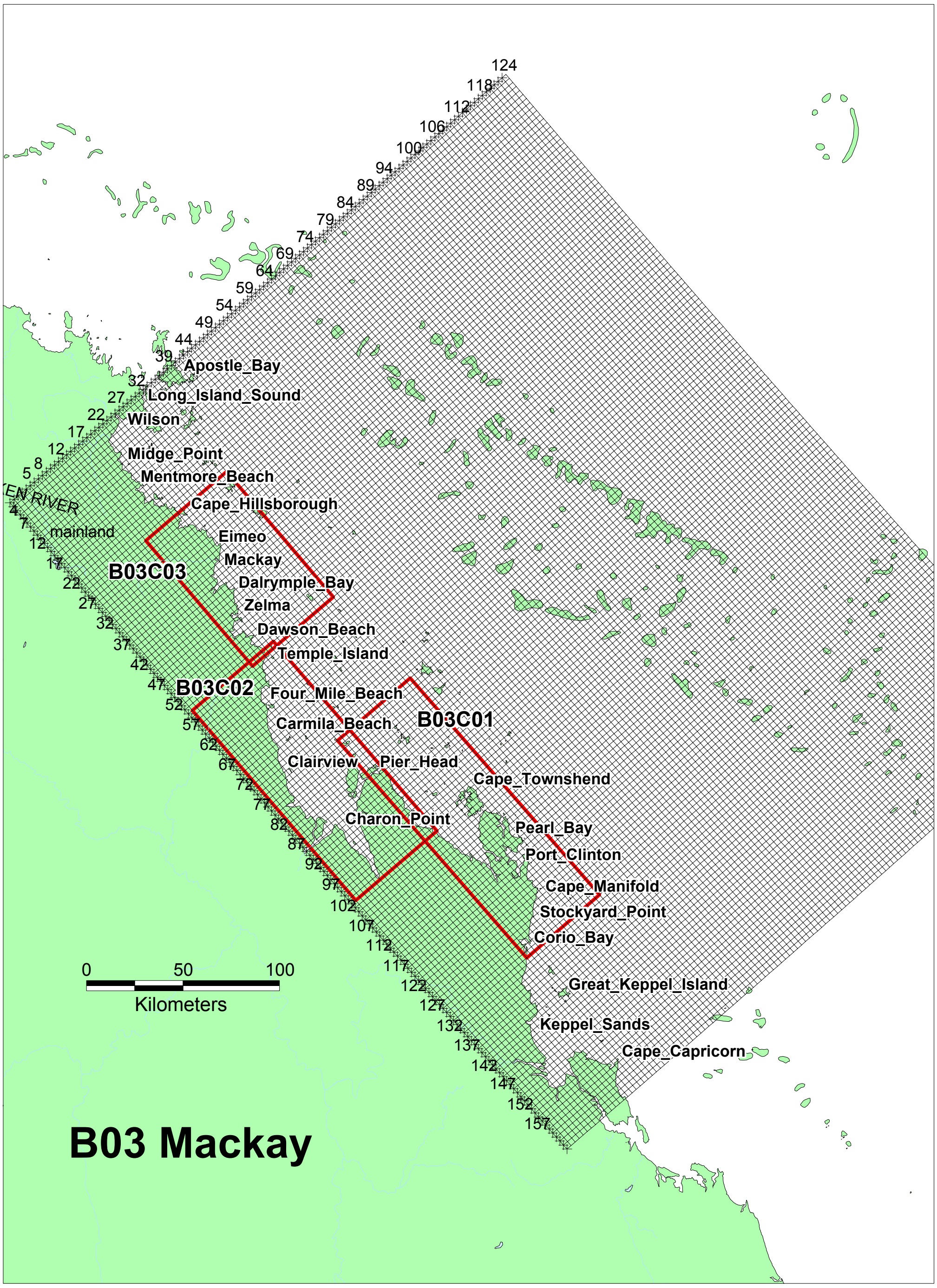
Mason L.B. and McConochie J.D. (2001) MMUSURGE user's guide. School of Engineering, Marine Modelling Unit, *James Cook University*, Townsville.

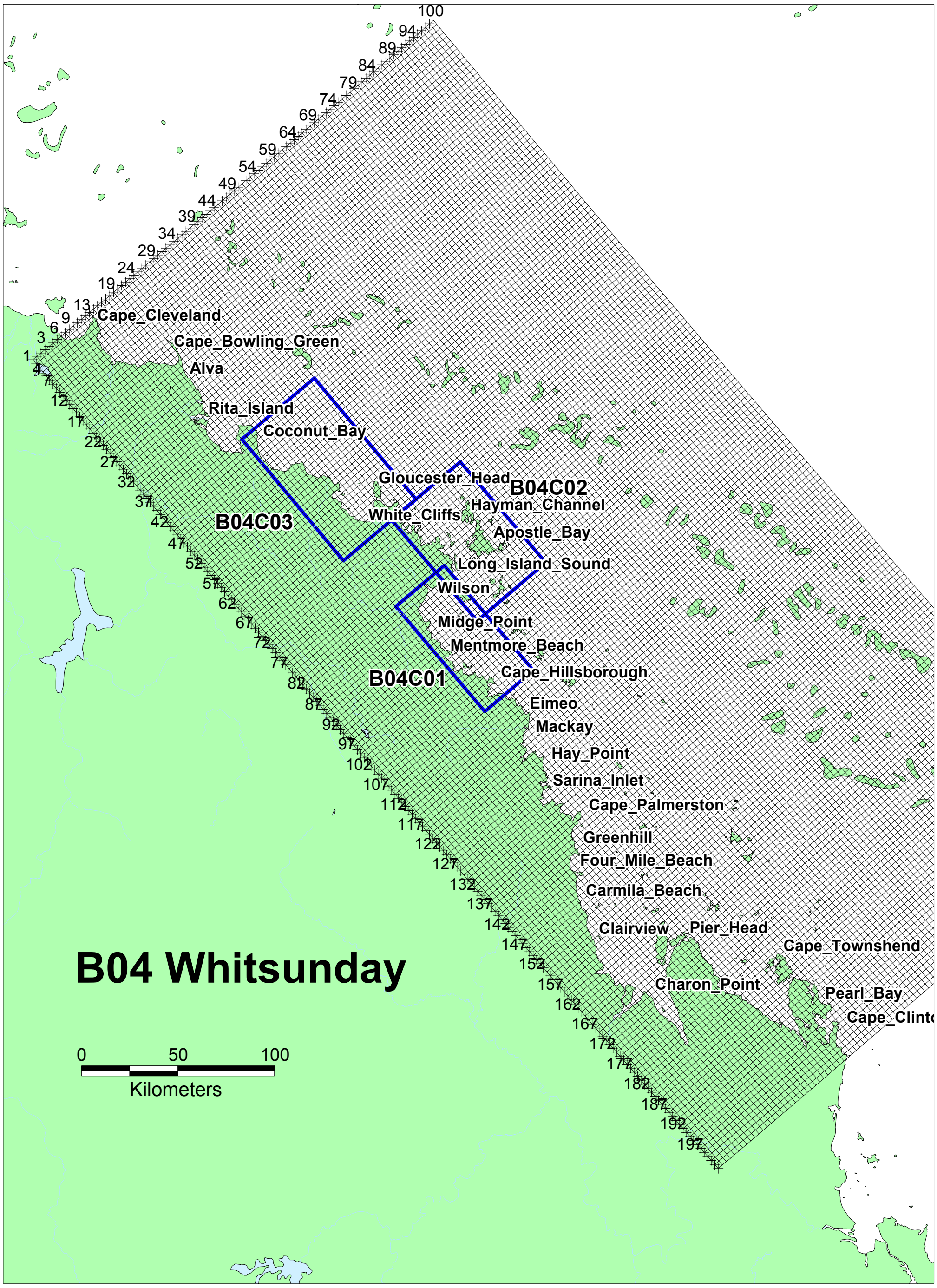
Appendix A

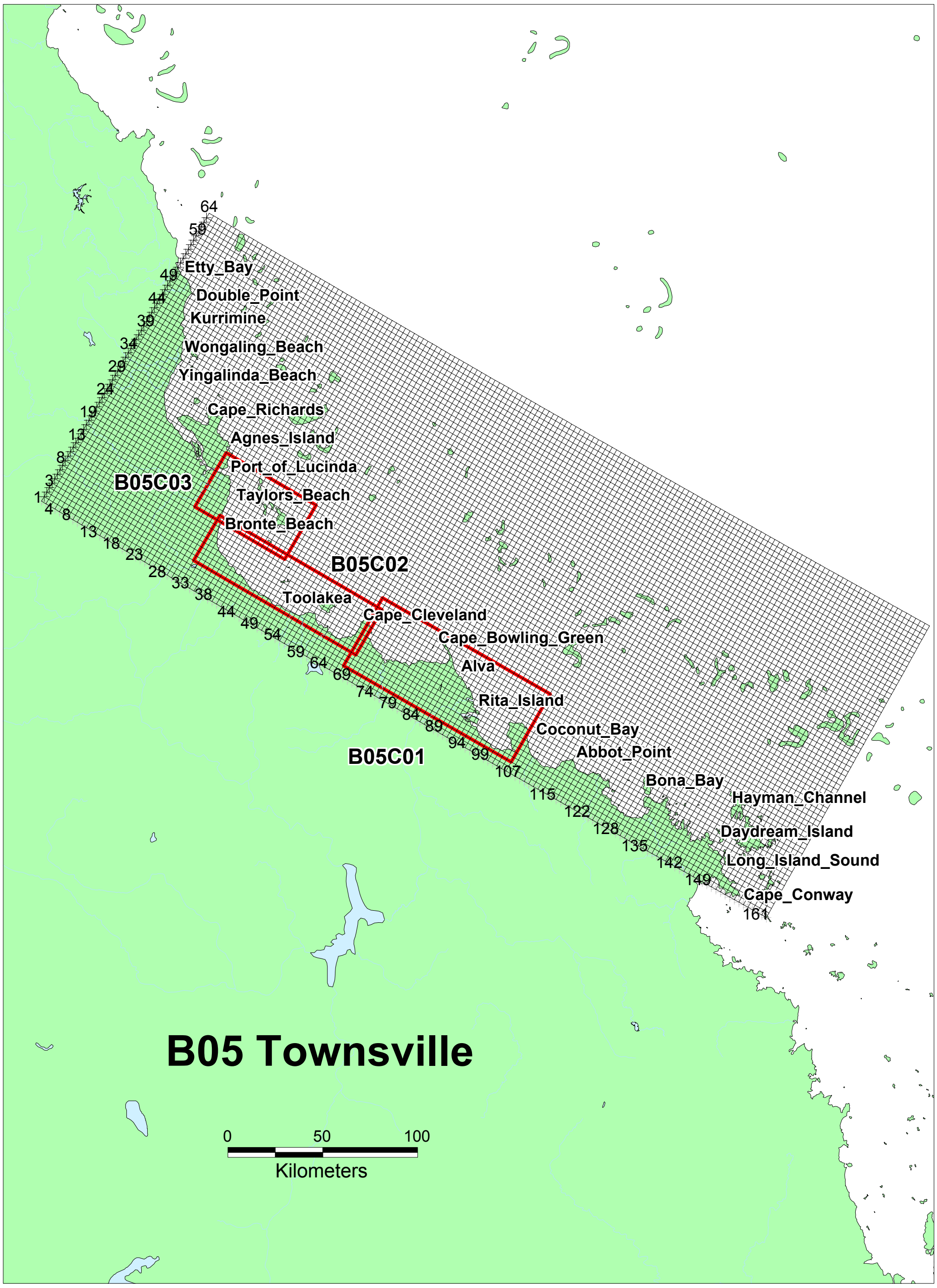
Detailed Model Domain Maps

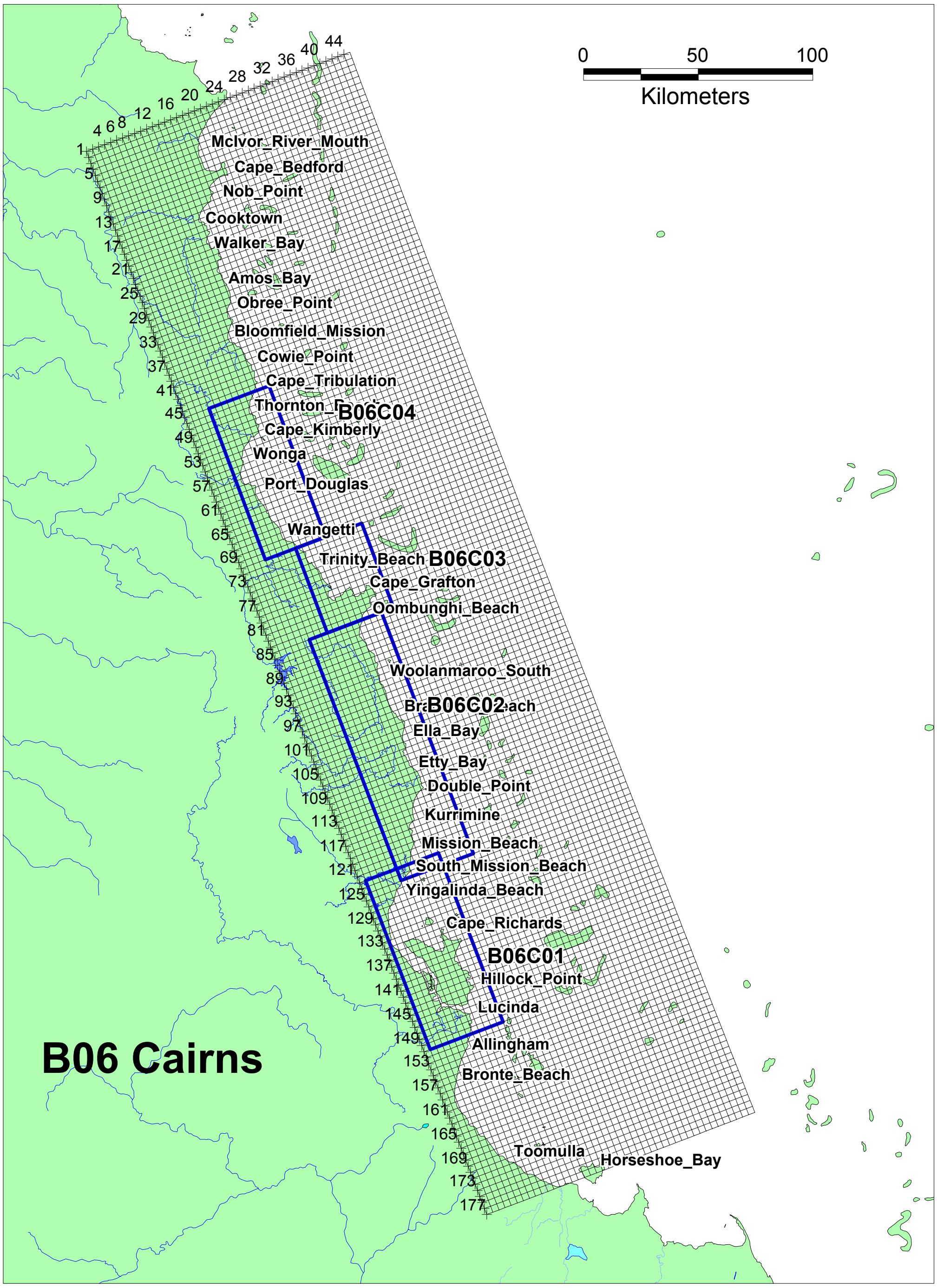

B01 Brisbane
0 50 100
Kilometers
B01C04
B01C03
B01C02
B01C01
Coonar
Woodgate
Burrum_Heads
Urangan
Waddy_Point
Yathon_Cliffs
The_Cathedrals
The_Pinnacles
Happy_Valley
Poyungan_Rocks
Eurong
Boonooroo
Wide_Bay_Harbour
Cherry_Venture
Cooloola_Beach
Teewah
Sunshine_Beach
Point_Arkwright
Marochy_River
Kawana_Waters
Caloundra
Toorbul
Cape_Moreton
Beachmere
Eager_Beach
Brighton
Gebejum
Manly
Point_Lookout
Blue_Lake_NP
Lamb_Island
Jumpinpin
Gold_Coast_Seaway
Burleigh_Heads
Tweed_Heads
Kingscliffe
Pottsville_Beach
Brunswick_Heads
Suffolk_Park
Lennox_Head

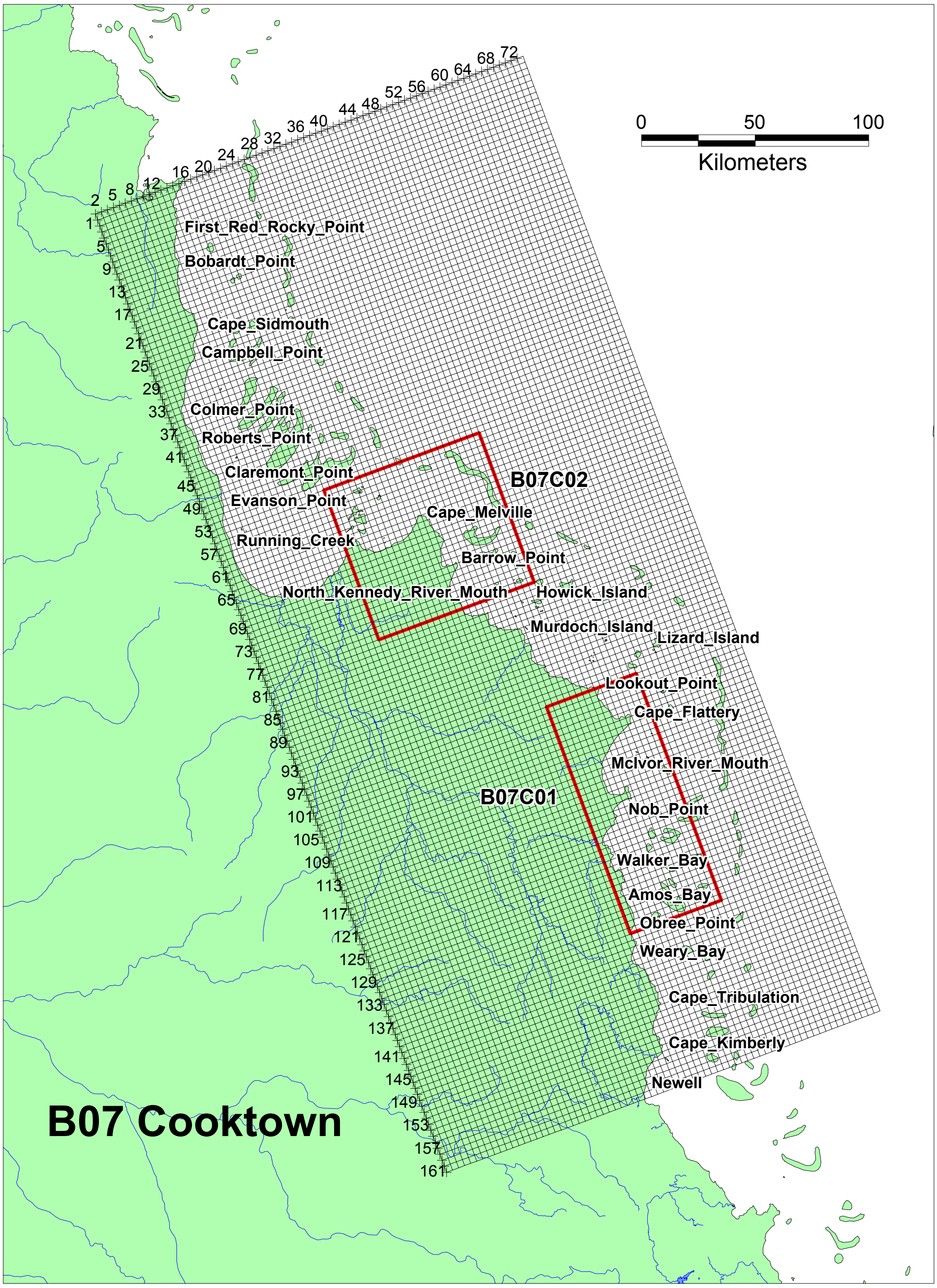

B02 Hervey Bay - Gladstone
B02C04
B02C03
B02C02
B02C01
0 50 100
Kilometers
Stockyard_Point
Corio_Bay
Yeppoon
Emu_Park
Keppel_Sands
Cape_Capricorn
Port_Alma
Black_Head
North_Channel
South_Trees_Island
Tannum
Bustard_Head
Agnes_Water
Port_of_Baffle_Creek
Moore_Park
Bargara
Elliot_Heads
Woodgate
Burrum_Heads
Urangan
Sandy_Cape
Mannan_Beach
Waddy_Point
The_Cathedrals
The_Pinnacles
Poyungan_Rocks
Eurong
Boonooroo
Wide_Bay_Harbour

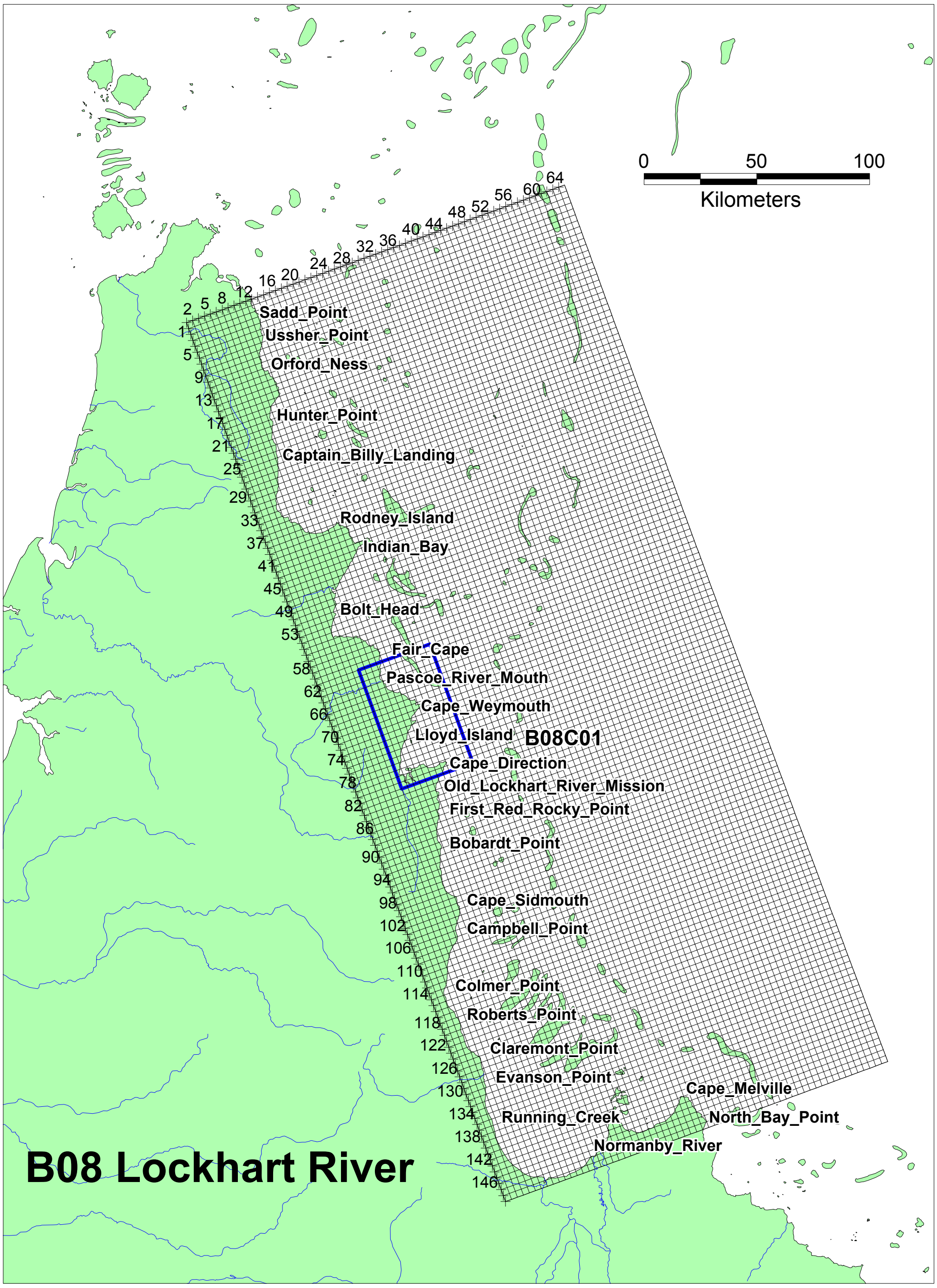

124
118
112
106
100
94
89
84
79
74
69
64
59
54
49
44
39
32
27
22
17
12
8
5
4
7
12
17
22
27
32
37
42
47
52
57
62
67
72
77
82
87
92
97
102
107
112
117
122
127
132
137
142
147
152
157
EN RIVER
mainland
Apostle_Bay
Long_Island_Sound
Wilson
Midge_Point
Mentmore_Beach
Cape_Hillsborough
Eimeo
Mackay
Dalrymple_Bay
Zelma
Dawson_Beach
Temple_Island
Four_Mile_Beach
Carmila_Beach
Clairview
Pier_Head
Cape_Townshend
Charon_Point
Pearl_Bay
Port_Clinton
Cape_Manifold
Stockyard_Point
Corio_Bay
Great_Keppel_Island
Keppel_Sands
Cape_Capricorn
B03C03
B03C02
B03C01
0
50
100
Kilometers
B03 Mackay

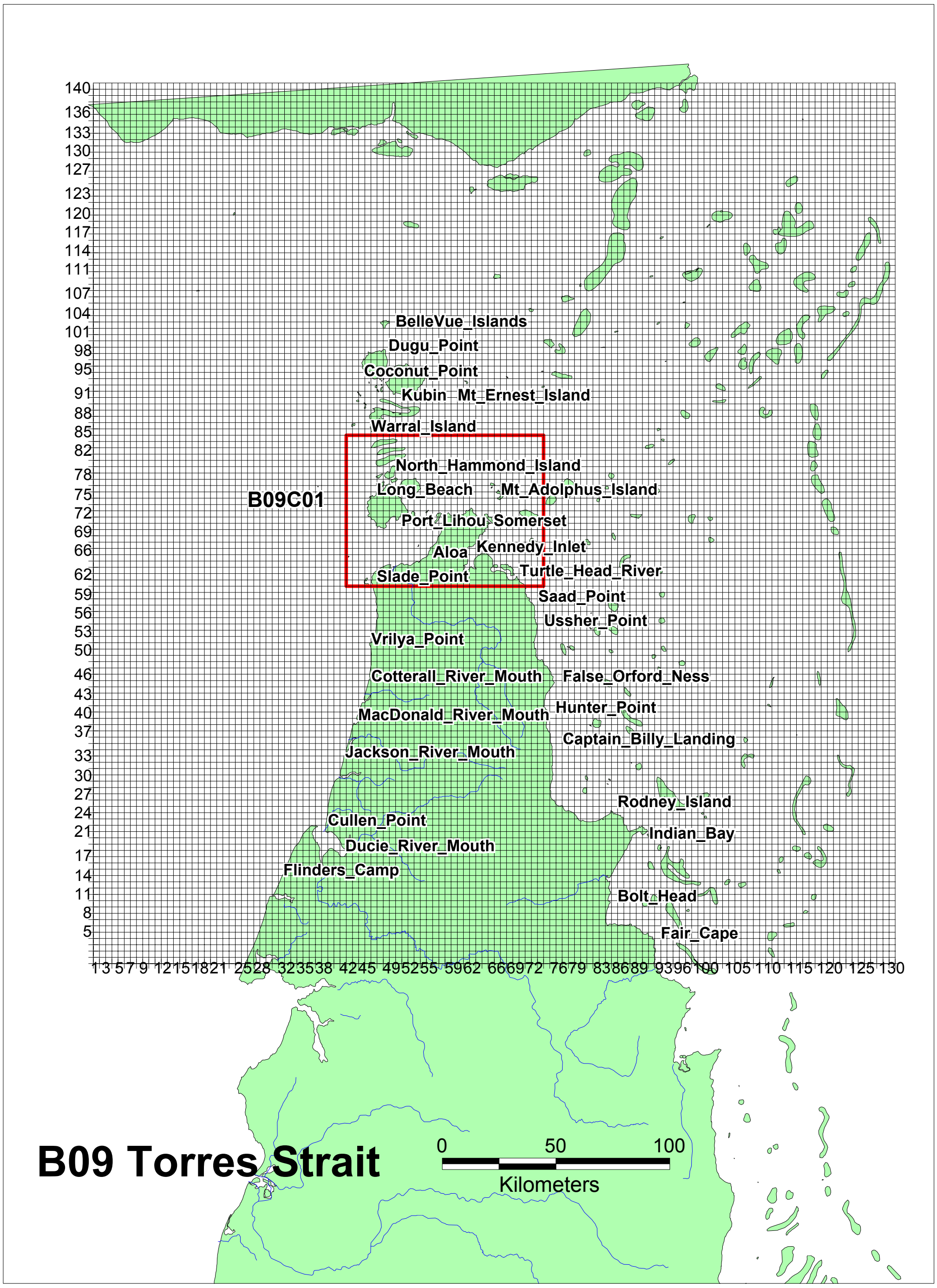

B04 Whitsunday
B04C01
B04C02
B04C03
Cape_Cleveland
Cape_Bowling_Green
Alva
Rita_Island
Coconut_Bay
Gloucester_Head
Hayman_Channel
White_Cliffs
Apostle_Bay
Long_Island_Sound
Wilson
Midge_Point
Mentmore_Beach
Cape_Hillsborough
Eimeo
Mackay
Hay_Point
Sarina_Inlet
Cape_Palmerston
Greenhill
Four_Mile_Beach
Carmila_Beach
Clairview
Pier_Head
Cape_Townshend
Charon_Point
Pearl_Bay
0
50
100
Kilometers

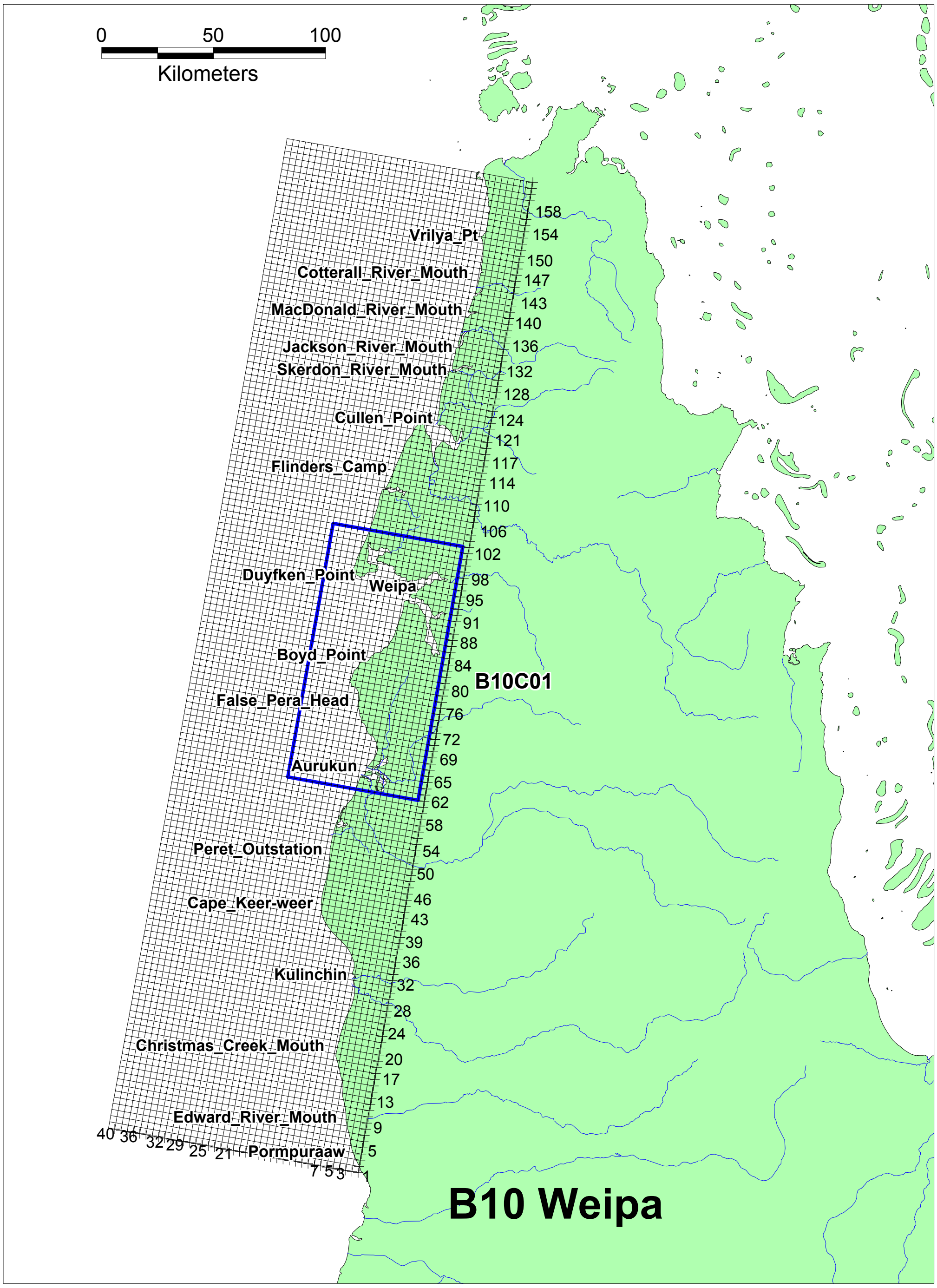

B05 Townsville
B05C01
B05C02
B05C03
Etty_Bay
Double_Point
Kurrimine
Wongaling_Beach
Yingalinda_Beach
Cape_Richards
Agnes_Island
Port_of_Lucinda
Taylors_Beach
Bronte_Beach
Toolakea
Cape_Cleveland
Cape_Bowling_Green
Alva
Rita_Island
Coconut_Bay
Abbot_Point
Bona_Bay
Hayman_Channel
Daydream_Island
Long_Island_Sound
Cape_Conway
0 50 100
Kilometers

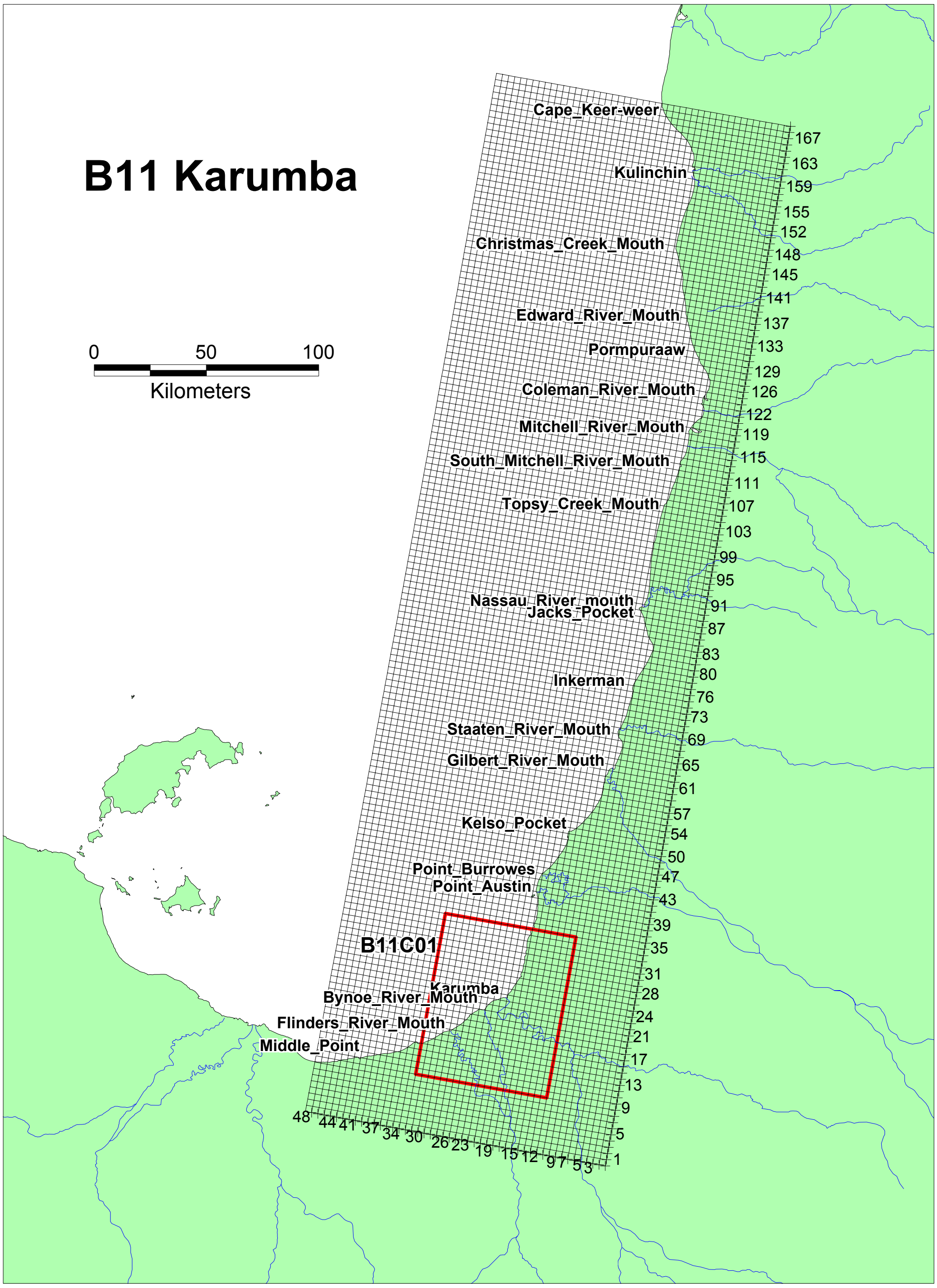

B06 Cairns
0 50 100
Kilometers
McIvor_River_Mouth
Cape_Bedford
Nob_Point
Cooktown
Walker_Bay
Amos_Bay
Obree_Point
Bloomfield_Mission
Cowie_Point
Cape_Tribulation
B06C04
Cape_Kimberly
Wonga
Port_Douglas
Wangetti
Trinity_Beach
B06C03
Cape_Grafton
Oombunghi_Beach
Woolanmaroo_South
B06C02
Ella_Bay
Etty_Bay
Double_Point
Kurrimine
Mission_Beach
South_Mission_Beach
Yingalinda_Beach
Cape_Richards
B06C01
Hillock_Point
Lucinda
Allingham
Bronte_Beach
Toomulla
Horseshoe_Bay

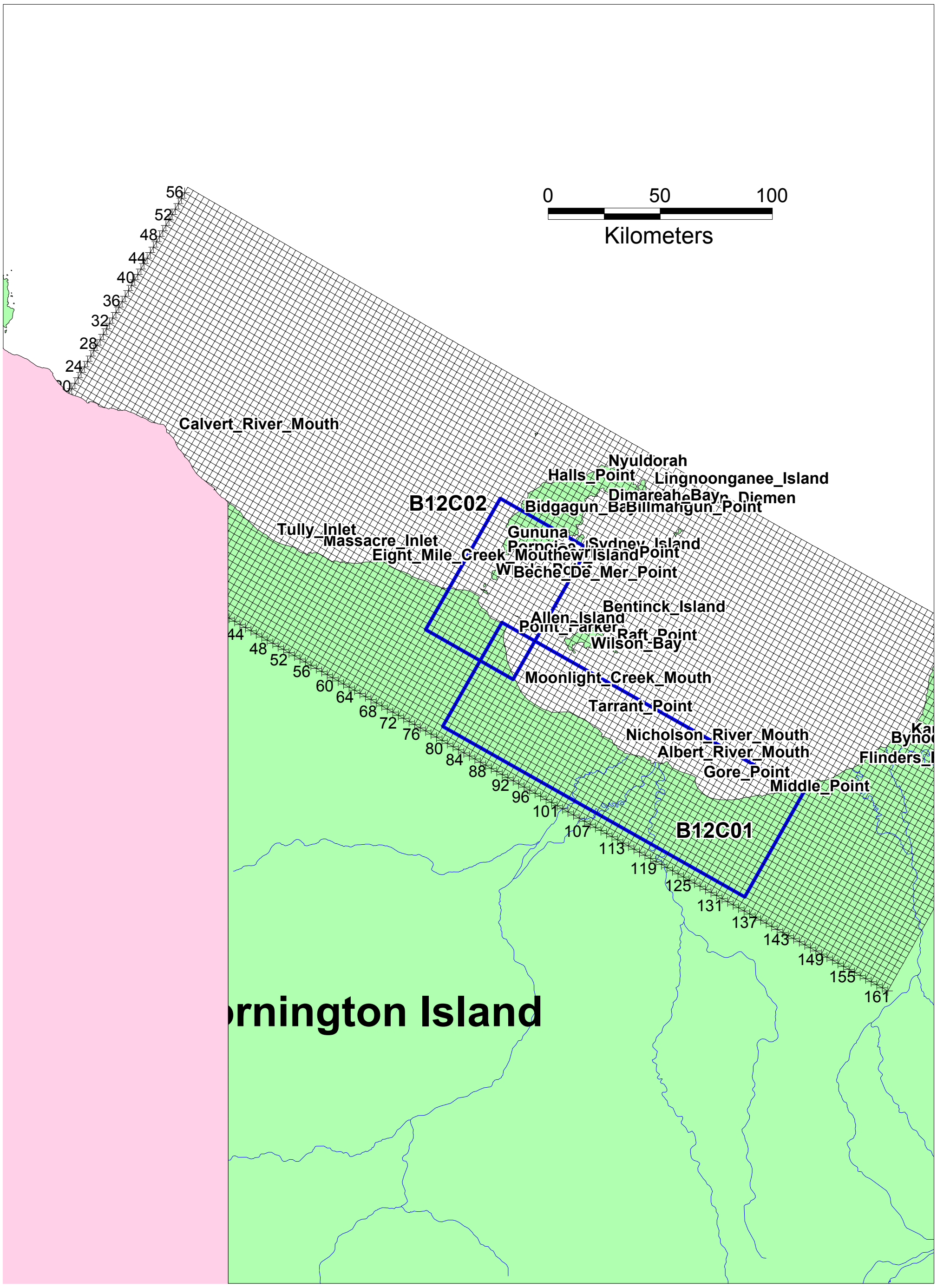

0 50 100
Kilometers
First_Red_Rocky_Point
Bobardt_Point
Cape_Sidmouth
Campbell_Point
Colmer_Point
Roberts_Point
Claremont_Point
Evanson_Point
Running_Creek
North_Kennedy_River_Mouth
B07C02
Cape_Melville
Barrow_Point
Howick_Island
Murdoch_Island
Lizard_Island
Lookout_Point
Cape_Flattery
McIvor_River_Mouth
B07C01
Nob_Point
Walker_Bay
Amos_Bay
Obree_Point
Weary_Bay
Cape_Tribulation
Cape_Kimberly
Newell
B07 Cooktown


0
50
100
Kilometers
B08 Lockhart River
B08C01
Sadd_Point
Ussher_Point
Orford_Ness
Hunter_Point
Captain_Billy_Landing
Rodney_Island
Indian_Bay
Bolt_Head
Fair_Cape
Pascoe_River_Mouth
Cape_Weymouth
Lloyd_Island
Cape_Direction
Old_Lockhart_River_Mission
First_Red_Rocky_Point
Bobardt_Point
Cape_Sidmouth
Campbell_Point
Colmer_Point
Roberts_Point
Claremont_Point
Evanson_Point
Cape_Melville
North_Bay_Point
Running_Creek
Normanby_River


B09 Torres Strait
B09C01
BelleVue_Islands
Dugu_Point
Coconut_Point
Kubin
Mt_Ernest_Island
Warral_Island
North_Hammond_Island
Long_Beach
Mt_Adolphus_Island
Port_Lihou
Somerset
Aloa
Kennedy_Inlet
Slade_Point
Turtle_Head_River
Saad_Point
Ussher_Point
Vrilya_Point
Cotterall_River_Mouth
False_Orford_Ness
MacDonald_River_Mouth
Hunter_Point
Captain_Billy_Landing
Jackson_River_Mouth
Rodney_Island
Cullen_Point
Indian_Bay
Ducie_River_Mouth
Flinders_Camp
Bolt_Head
Fair_Cape
0
50
100
Kilometers


0
50
100
Kilometers
Vrilya_Pt
Cotterall_River_Mouth
MacDonald_River_Mouth
Jackson_River_Mouth
Skerdon_River_Mouth
Cullen_Point
Flinders_Camp
Duyfken_Point
Weipa
Boyd_Point
False_Pera_Head
Aurukun
B10C01
Peret_Outstation
Cape_Keer-weer
Kulinchin
Christmas_Creek_Mouth
Edward_River_Mouth
Pormpuraaw
B10 Weipa


B11 Karumba
0
50
100
Kilometers
Cape_Keer-weer
Kulinchin
Christmas_Creek_Mouth
Edward_River_Mouth
Pormpuraaw
Coleman_River_Mouth
Mitchell_River_Mouth
South_Mitchell_River_Mouth
Topsy_Creek_Mouth
Nassau_River_mouth
Jacks_Pocket
Inkerman
Staaten_River_Mouth
Gilbert_River_Mouth
Kelso_Pocket
Point_Burrowes
Point_Austin
B11C01
Karumba
Bynoe_River_Mouth
Flinders_River_Mouth
Middle_Point
167
163
159
155
152
148
145
141
137
133
129
126
122
119
115
111
107
103
99
95
91
87
83
80
76
73
69
65
61
57
54
50
47
43
39
35
31
28
24
21
17
13
9
5
1
48
44
41
37
34
30
26
23
19
15
12
9
7
5
3


0
50
100
Kilometers
Calvert_River_Mouth
Nyuldorah
Halls_Point
Lingnoonganee_Island
B12C02
Tully_Inlet
Massacre_Inlet
Eight_Mile_Creek_Mouth
Gununa
Beche_De_Mer_Point
Bentinck_Island
Allen_Island
Wilson_Bay
Moonlight_Creek_Mouth
Tarrant_Point
Nicholson_River_Mouth
Albert_River_Mouth
Gore_Point
Middle_Point
B12C01
rnington Island

Appendix B

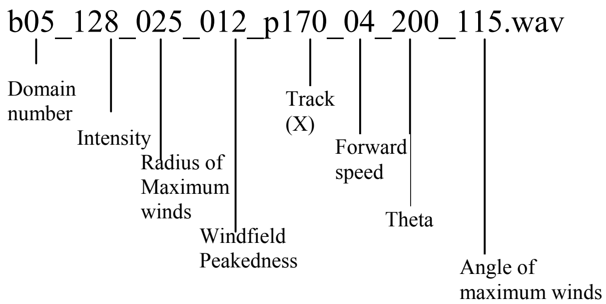
Parametric Model Data Processing

1. Detailed Description of the Processing Steps in the Development of the Parametric Model of Storm Surge[2]

The output of the JCU MMUSURGE numerical model consists of binary files that must be interrogated and processed to extract the various functions described in Section 6. The basic WAV binary output file contains an array of water height and current information as time histories for each output site; a vector of output times corresponding to the time histories; and an array of characters containing the names, location and *l* values of the output sites. The *l* values are actually created by *maketrack* during the track generation phase and appended to the coastal location names.

The following description relates to the Townsville domain, labelled B05. **Table 5.3** contains the representative values for the other regions, which can be substituted into the following discussion to describe the derivation process for those domains.

The "wav" filename for any specific B05 domain model run can be decoded as follows:



These are binary files and consist of the following arrays:

Array Name	Rows	Columns	Contents
HUV	121 timesteps	305 sites * 3	Height and u,v current data
wavtime	121 timesteps	1	Times at which output is given
name	305 sites	1	Name & location of output sites

The u,v current data is not required and so is immediately removed, resulting in an array called **hraw**:

Array Name	Rows	Columns	Contents
hraw	121 timesteps	305 sites	Height data

Such arrays, one for the wav file from each model run, look like:

hraw		Sites				
		1	2	…		305
Times	-18.0					
	-17.75					
	⋮					
	+12.0					

[2] These detailed processing notes were prepared by Matt Saunderson, Queensland Regional Office, and describe the *C* software developed during 2002 to 2004.

ie. Each column is a time series of surge height for a site in that basin for a single cyclone simulation.

The following processing steps simply extract different factors from these arrays using various averaging methods, which focus on specific aspects of the surge response. The following description follows the chronological order of the processing. Reference should be made to the *C* source code for more information about the actual steps completed in the processing.

The modelling and analysis process is summarised in the following flowchart, which assumes all wav files for a particular θ_{fm} in a particular domain are being processed:

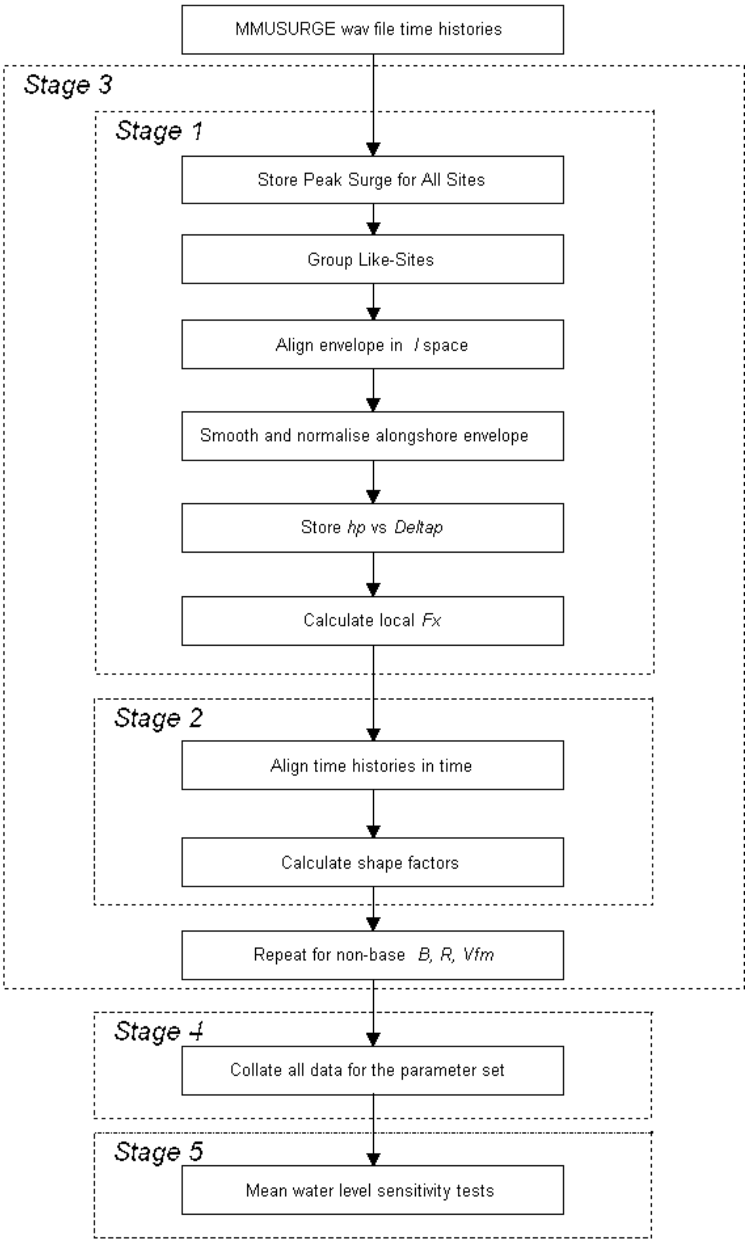


Figure A-1 An overview of the parametric model data processing.

1.1 Stage 1 – Base Case Alongshore Processing

The aim of this stage is to extract the h_p, h_l and $F\chi$ functions described in Section 7. The term "model run" will be used to collectively describe the full set of cyclone simulations for each domain *base case,* which have a common θ_{fm}, R, B and V_{fm} but different ΔP and X values. All wav files from a model run (eg 96 wav files for the *base case* parameters for Basin B05) are processed in sequence and various data extracted.

On the first pass through, the *l* value of every site is extracted and stored in **all_sites_l_array**:

all_sites_l_array		Sites				
		1	2	…		305
Wav file	1					
	2					
	⋮					
	96					

Also, the peak of every time history is extracted in **create_envelope.c** and stored in **as_env_array** (shorthand for "all sites envelope array"). The term "envelope" is used because this data represents the highest water level, or envelope of water, that is created by the cyclone. The peak of each column of a wav file is extracted and stored in a row vector. These 96 row vectors are then combined into a single array:

as_env_array		Sites				
		1	2	…		305
Wav file	1					
	2					
	⋮					
	96					

An extra piece of information is stored during the above calculations – the row index into **hraw** at which the peak occurs. This will later be modified and become **phi** - it describes how much a time history must be shifted in time so that its peak would occur at model time = 0. These indexes are stored in **temp_phi_array**:

temp_phi_array		Sites				
		1	2	…		305
Wav file	1					
	2					
	⋮					
	96					

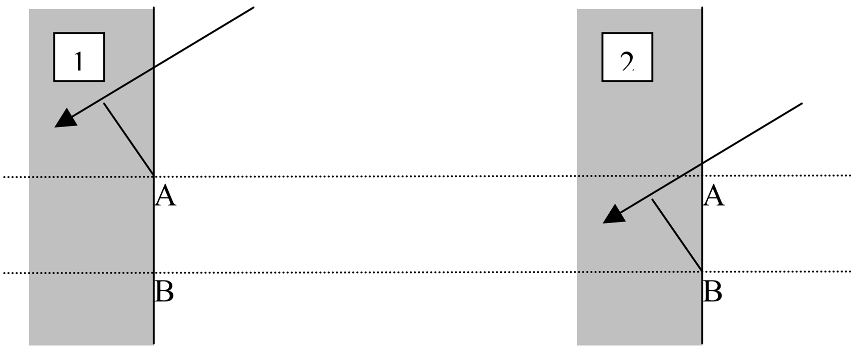
Up to this point, all wav files have been considered identically (no specific consideration of different intensity cyclones or different tracks) and all sites have been included. The next step is to realize that not all sites should be included in the processing. For example islands and passages, being separated from and of different nature to "normal" coastal points, will affect the results we obtain if they are included in the full processing. Hence they are now removed by reading in an **ignorelist** that specifies which sites should be excluded from processing. A different **ignorelist** file exists for all basins, and there can be several ignore lists for a single basin (eg. to create different models for the mainland and a large island). Using this method, other sites (eg. all of Fraser Island or everything but Fraser Island in basin B02) can be excluded for specialized parametric model development. The main way this is used for the B05 domain is to remove all islands and passages as well as some wrap-around points near Cape Bowling Green in preparation for the shift to *l*-space (described later).

After removing these sites we have two modified arrays for future use:

env_array		Required Sites Only				
		1	…			N
Wav file	1					
	2					
	⋮					
	96					

l_array		Required Sites Only				
		1	…			N
Wav file	1					
	2					
	⋮					
	96					

Next is the shift to *l*-space. This means that we will refer to locations on the coast by their perpendicular distance from the cyclone track as opposed to their original (arbitrary) site number. This will allow meaningful averaging of surge heights to take place. To see why, consider a theoretical straight-line coast with two different cyclones approaching:



Point A in scenario 1 is the same perpendicular distance from the cyclone as point B is in scenario 2. The properties of the cyclone in each scenario are identical, so exactly the same surge response must be expected at point A in scenario 1 and B in scenario 2. In our naming convention, the perpendicular distance of a site from the cyclone is labeled *l*, so sites with the same *l* value for the same cyclone can expect the same surge response (assuming the coast is straight-line and the bathymetry is regular).

We will make use of this assumption in averaging our *l*-space data to allow for the irregularity of a real coastline. Hence instead of finding sites with the *same l* values, we will find sites with *similar l* values – all sites within a nominal window width of δl = 4 km will be assigned a representative *l* value and their data values averaged to obtain a single representative surge height that applies to that *l* value.

This change to *l* space is achieved using **convert_lspace.c** which:

- sorts each row of **l_array** in order of increasing *l*
- saves the rearranged *l* values in **new_l_array**
- sorts **env_array** according to this new ordering

and results in:

new_l_array		Increasing *l* values				
		L_1	L_2	…		L_N
Wav files	1					
	2					
	⋮					
	96					

new_env_array		Sites by increasing *l* values				
		L_1	L_2	…		L_N
Wav files	1					
	2					
	⋮					
	96					

To ensure we have regularly spaced *l* values that are consistent between wav files (ie between the rows of **new_env_array**) we now further average the **new_env_array** rows according to fixed *l* values.

This is done by starting at the smallest *l* value in the row and averaging the heights of all the *l* values within a δl window of that smallest *l* value. So if the smallest *l* value in some row is -100, and we use a δl value of 4 km, then we loop through the other columns of **new_l_array** and average the heights of all sites with *l* values between –100 and –96. We can then assign an average *l* value to this average height, in this case –98, and shift the *l* window to the next group of *l* values (in this case, -96 to –92). This process is repeated for all *l* values in a row and then for all rows, resulting in:

lspace_env_array		Increasing *mean l* values				
		L_1	L_2	…		L_M
Wav files	1					
	2					
	⋮					
	96					

and

lspace_l_array		Increasing *mean l* values				
		L_1	L_2	…		L_M
Wav files	1					
	2					
	⋮					
	96					

From our earlier discussion of the benefits of *l*-space, we know that the surge response at a particular *l* value is independent of the track of the cyclone. So if we consider just the rows of **lspace_l_array** with the same *ΔP*, then the response for the same *l* value for the different rows (ie different tracks) should be the same. We are assuming a straight-line coast (and will account for the inaccuracies that this assumption introduces later) and so proceed to average the heights for each *l* value. This requires an intermediate step, as every row has a different range of *l* values. Hence we go through the process of:

- extract just the rows corresponding to a particular ΔP into **temp_env** and **temp_l**

temp_env		Increasing mean *l* values				
		L_1	L_2	…		L_M
Wav files	1					
	2					
	⋮					
	24					

For some delta P value

temp_l		Increasing mean *l* values				
		L_1	L_2	…		L_M
Wav files	1					
	2					
	⋮					
	24					

For some delta P value

- for this *ΔP*, find the overall minimum and maximum *l* values across all rows of **temp_l**
- start at the minimum *l* value, find the entries in all rows within the 4 m δl window of this *l* value, and average them. Once completed, shift the window by 4 km and average again.

When the maximum *l* value is reached, the 24 rows corresponding to the different tracks for this *ΔP* have been averaged into one representative row vector. Hence the above steps can be repeated for all 4 *ΔP* values, resulting in:

mean_env_vector		Increasing mean *l* values				
		L_1	L_2	…		L_N
ΔP	23					
	78					
	108					
	128					

and

mean_l_values		Increasing mean *l* values				
		L_1	L_2	…		L_N
ΔP	23					
	78					
	108					
	128					

Having created these arrays we can immediately extract two of the functions described in Section 6. Firstly, if we find the maximum height of each row of **mean_env_vector**, then we have **max_mean**, the peak height due to a cyclone with that intensity.

Secondly, if we normalize each row of **mean_env_vector** by dividing each element by the peak of that row, and then average across all 4 ΔP values, then we get the mean along coast profile of surge response. This is saved as **norm_av_mean_height_array**.

Hence we now have:

max_mean		Height
ΔP	23	
	78	
	108	
	128	

norm_av_mean_height_array	Increasing *l* values				
	L_1	L_2	…		L_N
Along coast height profile					

From the earlier discussion, if all coastlines were straight then this is all that would be required. Simply extract the peak height for that intensity cyclone from **max_mean** and multiply it by the factor for the *l* value of the site you are interested in from **norm_av_mean_height_array**. However to cater for real-world coastlines that are not straight we need to extract the difference between this theoretical value and the actual value. This is known as $F\chi$ in Section 6. To do this:

- extract a row of actual peak surge values from **env_array** and interpolate into it for every site to grab the various actual peak surge heights

  Numerator = actual peak surge height for site S

- grab the *l* values for each of these sites and interpolate into the **norm_av_mean_height_array** to get the modelled version of the peak heights

  Denominator = modelled peak surge height for site S

- divide the first value by the second value and this gives the required variation in height

  $F\chi$ for site S = Numerator / Denominator

We are therefore able to store this $F\chi$ value for every wav file and site combination, then (assuming $F\chi$ is not a function of ΔP) further average across the 4 ΔP values to give the **average_ratio_array**:

average_ratio_array		Sites				
		1	2	…		305
W av fil	1					
	2					

	⋮					
	24					

By repeating the same process as described thus far for the three different θ_{fm} values we can get a set of results (stored as arrays) for each θ_{fm}. We then combine these into the functions that were described in Section 6 and store them as tables that can be interrogated for any value of any parameter. The final arrays at this stage are therefore:

hp_vs_delp		θ_{fm}		
		200	240	260
ΔP	23			
	78			
	108			
	128			

mean_heights		θ_{fm}		
		200	240	260
L values	L_1			
	L_2			
	…			
	L_m			

fchi_ratio_array		Sites				
		1	2	…		305
Wav files	1					
	2					
	⋮					
	24					

(stacked for θ_{fm} = 200, 240, 260)

1.2 Stage 2 – Base Case Time History Processing

This next stage operates slightly differently from the first stage. Instead of loading all 96 wav files at once and processing them as a whole, we cycle through the 4 different ΔPs in turn and consider initially just the 24 wav files for each of these four groups separately.

For a particular ΔP we load each wav file in turn and create **hraw (**and hence **ignored_hraw)** as in Stage 1. For each of these arrays we again need to convert to *l*-space. However this time instead of first calling **create_envelope** (and hence considering only the peak heights), we wish to keep the full time histories. This will allow us to average these time histories to gain the next function from Section 6 – the average shape of the time profile h_t. So our first step is to call **convert_lspace_time** to:

- reorder the *l* values in increasing order and apply the same ordering to the columns of **ignored_hraw**, resulting in **sorted_hraw**

sorted_hraw		Increasing sorted *l* values				
		L_1	L_2	…		L_n
Times	-18.0					
	-17.75					
	⋮					
	+12.0					

- obtain regularly spaced, consistent *l* values by averaging all columns within (again) a 4 km δl window, giving **lspace_env_array**:

lspace_env_array		Increasing sorted *l* values				
		L_1	L_2	…		L_m
Times	-18.0					
	-17.75					
	⋮					
	+12.0					

- for each converted hraw file, add a new row to **lspace_l_array** for that wav file to keep track of the *l* values:

lspace_l_array		Increasing *l* values				
		L_1	L_2	…		L_m
Wav files	1					
	2					
	⋮					
	24					

Once the **hraw** arrays have been converted to *l*-space we must normalize each time history. This will allow us to average the shapes of the time histories while ignoring any effects of the magnitudes (again assuming shape is independent of ΔP). To do this, we first find the peak height of the time history of each l value and divide each element of that column by the peak. We then store this normalized time history in **norm_growing_array** and also again store the index of the peak height as a measure of ϕ.

Some normalized lspace_env_array		Sites				
		L_1	L_2	…		L_m
Times	-18.0					
	-17.75					
	⋮					
	+12.0					

and

norm_growing_array

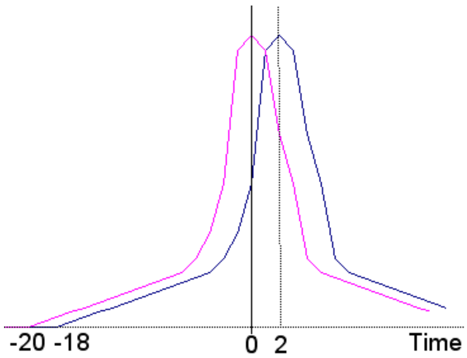
Wav files			
Wav file 1	Wav file 2	…	Wav file 24
Normalized Lspace_env_array Increasing l values: L_1, L_2, …, L_m imes: -18.0, -17.75, ⋮, +12.0	**Normalized** .space_env_array Increasing l values: L_1, L_2, …, L_m -18.0, -17.75, ⋮, +12.0		**Normalized** Lspace_env_array Increasing l values: L_1, L_2, …, L_m imes: -18.0, -17.75, ⋮, +12.0

It should be noted that:

- some time histories stay negative for all times. These are normalized by dividing the column by –1 multiplied by the peak height.
- the actual value of the peak height for each column is temporarily stored as it will later be used to weight the contribution of each time history shape to the overall average, so that the highest surges have the greatest influence on the results.
- **norm_growing_array** contains time histories for all sites in all 24 wav files for this *ΔP*.

Now that we have all the normalized time histories stored in **norm_growing_array** we wish to average their shapes. To do this they must all have peaks at the same time, so we shift them all to have peak at time = 0. This can be done using the **phi** values we saved in Stage 1.

We know the index of time = 0 in the **wavtime** array, and we know the index of the peak of each time history, so the difference between the two indices can tell us how many hours the time history must be shifted to have its peak at t = 0.



Since every time history was initially only from –18 to +12 hours, shifting it in time will mean that it no longer meets those conditions. For example, if a time profile has a peak at t=2, we must shift it by two hours to have a peak at t = 0. Therefore the shifted time history now runs from –20 to +10 hours.

In order to meaningfully average the time histories we will want them to have the same time domain, so our first step is to find the overall minimum and maximum times across all the sites. We can then make sure the time history of every site covers this extended time domain (by padding the start and end of each site's time history as required with that site's first and last height values).

Making these changes to **norm_growing_array** will increase the number of rows (to cover these extended timesteps rather than all being –18 to +12 hours) but will leave the number of columns the same (we are not changing the *l* values at all), and we store the result in **final_converted_height_array**:

final_converted_height_array

		Wav files												
		Wav file 1				Wav file 2				…	Wav file 24			
	Normalized Lspace_env_array	Increasing l values				Increasing l values					Increasing l values			
		L_1	L_2	…	L_m	L_1	L_2	…	L_m		L_1	L_2	…	L_m
imes	-18.0													
	-17.75													
	⋮													
	+12.0													

We are now in a position where we can average all the time histories by simply averaging every value in the rows of **final_converted_height_array**. We don't need to bother with adjusting the *l* values like we did in Stage 1 as we are not interested in where the site is relative to the cyclone – we are assuming *every* site will have a similar shape to its time profile. However to take into account very low surges (eg around 10 cm peaks, or even negative surges) that, when normalized, will have much different shapes to a "normal" positive surge, we will weight the contribution of each *l* value's time history by the un-normalized peak of that time history. Hence low surges, and negative surges, will contribute much less to the overall average time history shape than the large surges (which the model is trying to best capture the behaviour of). By averaging across all rows and multiplying each element from a particular row by its weight (which was stored earlier) we get a single time profile representative of all sites.

Now the whole of Stage 2 to this point is repeated for each ΔP value in turn, so for basin B05 we will ultimately have 4 different mean time profiles stored in **weighted_line_vector** and 4 different pairs of maximum/minimum times of the time histories stored in **max_min_domain_array**:

weighted_line_vector		Extended time steps				
		T_1	T_2	…		T_N
ΔP	23					
	78					
	108					
	128					

min_max_domain_array	ΔP			
	23	78	108	128
Minimum Time				
Maximum Time				

Now each row in **weighted_line_vector** has a potentially different time domain described in **min_max_domain_array**, so again these need to be made identical. We follow the same process as earlier, in which we find the overall minimum and maximum times and pad the 4 different time profiles accordingly so that they all match. The results are stored in **final_weighted_array**:

final_weighted_array		Final extended time steps				
		T_1	T_2	…		T_M
ΔP	23					
	78					
	108					
	128					

We can then find the average of these time histories by averaging the 4 rows, giving **weighted_sum**, and also find the final time domain by stepping from the overall minimum time to the overall maximum time in time steps of δt, giving **final_domain**:

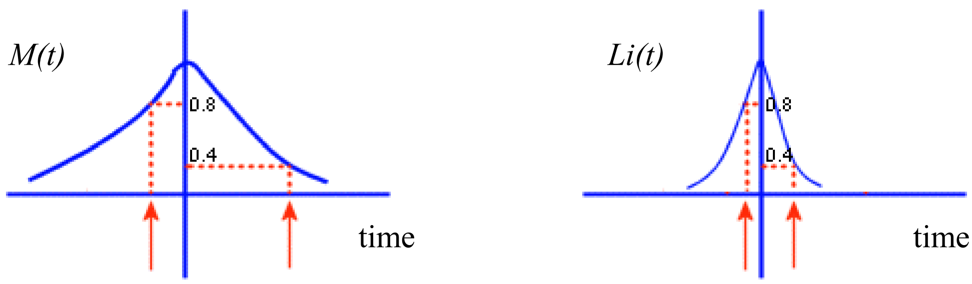
weighted_sum	Extended time steps			
	T_1	T_2	…	T_M
Heights				

final_domain	Extended time steps			
	T_1	T_2	…	T_M
Times				

As with Stage 1, this same process is repeated for all three θ_{fm} values, resulting in three different time histories and time domains. Combining them into a single array gives us the function from Section 6 that we were aiming for – h_t, the normalized time history shape for all cyclones and all sites.

time_history		θ_{fm}		
		200	240	260
Times	T_1			
	T_2			
	⋮			
	T_M			

We need to complete one extra step in Stage 2. We've found the average normalized shape that will be suitable for the base parameters we have considered so far. However, to reduce the data needs of the parametric model it is desirable to compress the time history response down to some simple shape factors. The way this is approached is to use two measures, one for the negative section of the time history (the "lead up" to the peak) and one for the positive. Due to large variations in the shapes between model runs for the times leading up to the peak, a measure close to the peak is used for the negative times. This aims to ensure that the modelled profile for times close to the peak will closely match the actual profile. A lower measure is used for the positive times because there was observed to be much less variation in the shapes of the profiles for these times. The measures chosen were the 0.8 height (leading) and the 0.4 height (trailing). To see how this is useful, suppose we have a mean time profile *M(t)* and a collection of *n* different time profiles for a range of *l* values, each labeled $L_i(t)$. To store the full data for each of the *n* time profiles would be too cumbersome. A single averaged time profile is not suitably accurate to replace them all. Hence we instead store the "shape" measures of the *n* time profiles, safe in the knowledge that we can recreate each of them from the mean time profile when the need arises. To see why this is the case, considering the following setup:



From the diagram above, for all *l* values from 1 to *n* we will store the ratios:

time for *M*=0.8 and time for *M*=0.4
----------------------- -----------------------
time for L_i = 0.8 time for L_i=0.4

for negative and positive times respectively. To see why these are useful, suppose we wish to recreate one of the L_i time profiles. Our recreated version will be labeled $R_i(t)$.

If we define $R_i(t)$ such that:

$R_i(t) = M(\text{ratio} * t)$

Then at the point where $Li(t) = 0.8$ for negative times:

$$R_i(t \text{ for } Li \text{ to be } 0.8) = M(\text{ratio} * (t \text{ for } L_i \text{ to be } 0.8))$$
$$= M\left(\frac{t \text{ for M to be } 0.8}{t \text{ for } L_i \text{ to be } 0.8} * t \text{ for } L_i \text{ to be } 0.8\right)$$
$$= M(t \text{ for } M \text{ to be } 0.8)$$
$$= 0.8$$

Hence at the time we expect L_i to be 0.8, our recreated function R_i also equals 0.8. We are keeping the same values of the *M* range, but modifying its domain (stretching or shrinking it by the ratio calculated above) to recreate our desired shape.

If we repeat the same procedure with the mean function being **weighted_sum**, the corresponding times being **final_domain** and the collection of *n* time profiles being the time profiles of every *l* value in the wav file, then we have a means of recreating the individual sites' time histories from the mean shape with good accuracy. We store all of this shape information in **time_shape_data**:

time_shape_data		*l* value				
		1	2	…		305
ratio	Negative					
	Positive					

1.3 Stage 3 – Sensitivity to R, B, V_{fm}

The preceding process descriptions assumed that the *R*, *B* and V_{fm} values being used were the *base case* values. The same series of model simulations is then repeated, changing only one of the *R*, *B* and V_{fm} values in each case to provide information on how these parameters might affect the storm surge magnitude, along-coast factors and time history shapes etc.

1.4 Stage 4 – Collating the Processed Data for each Parameter Set

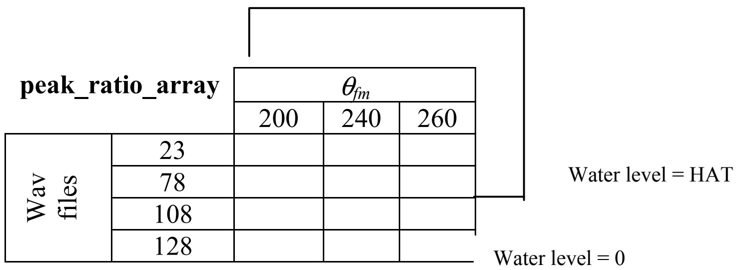
This stage is merely an organizing step where the different data elements are taken from their raw formats and saved to the final data files that will be used by the model. For example, peak height data is initially stored in three raw files, one for each value of θ_{fm}. In this stage, the data for the first θ_{fm} value is written to a new file, and then the same data for the last two θ_{fm} values is appended to that same file. Consequently we build up a library of self-contained data files that can be called by the final predictive parametric model. This stage is completed by **comp_model.c**.

1.5 Stage 5 – Sensitivity to Mean Water Level

The final pre-processing stage is to create a separate data file to take into account the effect of different base water level on the surge response for the basin. After examining the HAT values for the basin, an upper limit is chosen (which slightly exceeds the highest HAT found, so that it will be possible to interpolate for any possible tide value) and the MMUSURGE input data files are updated. This simply requires a change to the "*dmsl*" value in the appropriate bed files (e.g. a02.bed and b05.bed for the Townsville domain).

Then the following additional steps are then undertaken to provide a mean water level response function:

- A near straight-line section of coast in each domain is targeted (to keep any effects of bays, capes etc to a minimum) and a collection of model runs are completed for 3 X values, 4 ΔP values and 4 θ_{fm} values. The tracks (X values) cover a small section of coast and we will be interested in the sites at approximately one radius of maximum winds (R) from these tracks to determine the effect of water level.
- Process the above model runs normally, creating the same data files as earlier, but we are only interested in the **hp_vs_delp_2d** file, as this tells us the peak values.
- Copy the initial base **wav** files that correspond to the above model runs into a separate directory and rerun the processing steps (Stages 1-4). We will now have two **hp_vs_delp_2d** data files to compare – one for the *base case* water level of MSL, and one for the higher water level of HAT.
- We can now simply divide each array by the base water level array and convert these two 2D arrays into a single 3D array called **peak_ratio_array**. The indexes are ΔP and θ_{fm} (as previously) and now water level. The parametric model can now interpolate into this 3D array for values of ΔP, θ_{fm} and water level to extract the multiplier for water level.



1.6 Processing Directory Structure

The \prod root directory is where the *base case* MMUSURGE modelling is done for each domain. Within each domain, the parametric processing is output to \model for the base case and \sens_model for the sensitivity cases (R, B, V_{fm}). All the MMUSURGE sensitivity runs are done under the \sens root directory. All the parametric processing for the HAT sensitivity is done under the \hat root directory.

An example operational directory structure for Basin 01 processing follows. This example can be replicated for any other basin by changing b01 to the appropriate basin number, and then substituting the appropriate parameter values for that basin from **Table 5.3**.

/ssurge/meow project root directory

<top>	<domain>	<output>	<sub>	<description>
/prod				Root MMUSURGE production directory
	/b01			Basin 01 base case root directory
		/t200		MMUSURGE θ_{fm}=200 base case, MSL
		/t220		MMUSURGE θ_{fm}=220 base case, MSL
		/t240		MMUSURGE θ_{fm}=240 base case, MSL
		/t200_hat		repeat above base cases for HAT
		/t220_hat		“
		/t240_hat		“
		/model		Processed parametric model output
			/run1	1st set of included sites
			/run2	2nd set of included sites
			/run3	3rd set of included sites
			/run4	4th set of included sites (if needed)
		/sens_model		Processed parametric sensitivity output
			/B09	B=0.9 sensitivity
			/B13	B=1.3 sensitivity
			/R25	R=25 sensitivity
			/R75	R=75 sensitivity
			/V02	V=2 sensitivity
			/V08	V=8 sensitivity
/sens				Root MMUSURGE sensitivity directory
	/b01B09			MMUSURGE B=0.9 sensitivity; MSL
		/t200		MMUSURGE θ_{fm}=200
		/t220		MMUSURGE θ_{fm}=220
		/t240		MMUSURGE θ_{fm}=240
	/b01B13			MMUSURGE B=1.3 sensitivity; MSL
		/t200		MMUSURGE θ_{fm}=200
		/t220		MMUSURGE θ_{fm}=220
		/t240		MMUSURGE θ_{fm}=240
	/… (repeat for all b01 parameters – R25, R75, V02, V08)			
/hat				Root HAT processing directory
	/b01			basin 01 root directory
		/t200		processed for 3 tracks for θ_{fm}=200
		/t220		processed for 3 tracks for θ_{fm}=220
		/t240		processed for 3 tracks for θ_{fm}=240
		/model		parametric model output for HAT
		/model_base		corresponding output from *base case* data

The \model\run1 … \run2 represent different parametric models for a single domain that are restricted to specific sites (e.g. mainland, islands etc) to improve the model performance in those situations.

Appendix C

Predictive Parametric Model Processing

1. Documentation for the Predictive Parametric Storm Surge Model[3]

This section describes the operation of the predictive parametric model, as summarised in theoretical terms in Section 6.5. That is, how the various data files created to form the parametric model description in Appendix B are grouped and interrogated to recreate the storm surge response for a particular tropical cyclone in a particular basin.

The present parametric model can run in two modes – either a *time history* mode (in which the time history of surge response is generated for a site of interest) or an *along-coast* mode (in which the along-coast "snapshot" of surge height is generated for a time of interest). The processing steps are nearly identical for both, however the following description initially relates just to the time history mode. The different processing steps required for the along-coast mode are described later.

1.1 Phase 1 – Load in all the Parametric Data Files

During the initial processing (see Appendix B), a series of data files were created. Some of these were only temporary and required only for the next step of processing, while others were generated purely for the parametric model. These parametric model data files are loaded into memory so they can be interpolated into to recreate the surge response. The required data files, and the arrays used in the parametric model that relate to them, are listed in Table C-1.

Two other data files are also loaded. The first, stored in a file called **size_information**, holds various useful variables such as array dimensions and numbers of sites and timesteps. The second file, called **lat_lon**, holds arrays of latitude and longitude values of all sites in the domain (indexed by site number).

1.2 Phase 2 – Get the User Information

The parametric model works by interpolating into the arrays loaded in Phase 1 for values of the different parameters input by the user. These parameters are θ_{fm}, ΔP, water level, peakedness (B), radius of maximum winds (R), speed of movement of the cyclone (V_{fm}), site of interest (either a latitude/longitude pair or a site number) for the time history to be produced (or time of interest for the along-coast profile to be produced) and crossing location of the cyclone (either a latitude/longitude pair or a site number). The input parameters are specified in one of three ways:

> <u>Method 1- Interactive</u>
> The user is prompted for the different values for a single run of the model.
> <u>Method 2 – Command line</u>
> The user lists all parameters on the command line using various switches for a single run of the model.
> <u>Method 3 – Input file</u>
> The user lists a collection of parameters in an input file for multiple runs of the model.

Regardless of how the information is passed to the model, the required variables are stored ready for the interpolation to proceed.

[3] These detailed processing notes were prepared by Matt Saunderson, Queensland Regional Office, and describe the *C* software developed during 2002 to 2004.

Table C-1 Data arrays used for constructing a storm surge prediction.

Datafile	Array Created	X dimension and contents	Y dimension and contents	Z dimension and contents	W dimension and contents
hp_vs_delp_2d	hp_vs_delp_2d	hp_vs_delp_xvec; 3 theta	hp_vs_delp_yvec 4 Delta P	-	-
Hp_vs_delp_B	Hp_vs_delp_B	hp_vs_delp_xvec; 3 theta	hp_vs_delp_yvec; 4 Delta P	Hp_vs_delp_zvec_B; 3 B	-
Hp_vs_delp_R	Hp_vs_delp_R	hp_vs_delp_xvec; 3 theta	hp_vs_delp_yvec; 4 Delta P	Hp_vs_delp_zvec_R ; 3 R	-
Hp_vs_delp_V	Hp_vs_delp_V	hp_vs_delp_xvec; 3 theta	hp_vs_delp_yvec; 4 Delta P	Hp_vs_delp_zvec_V; 3 V	-
Overall_mean_height_2d_prepare d	Overall_mean_height_2d	Overall_mean_height_xvec 3 theta values	Overall_mean_height_yvec L values	-	-
Overall_mean_height_2d_B	Overall_mean_height_2d_B	Overall_mean_height_xvec 3 theta values	Overall_mean_height_yvec L values	Overall_mean_height_zvec_B 3 B values	-
Overall_mean_height_2d_R	Overall_mean_height_2d_R	Overall_mean_height_xvec 3 theta values	Overall_mean_height_yvec L values	Overall_mean_height_zvec_R 3 R values	-
Overall_mean_height_2d_V	Overall_mean_height_2d_ V	Overall_mean_height_xvec 3 theta values	Overall_mean_height_yvec L values	Overall_mean_height_zvec_V 3 V values	-
Mean_fchi_3d	Mean_fchi_3d	Mean_fchi_xvec; 305 sites	Mean_fchi_yvec; 24 wav	Mean_fchi_zvec; 3 theta	-
Mean_fchi_3d_B	Mean_fchi_3d_B	Mean_fchi_xvec; 305 sites	Mean_fchi_yvec; 24 wav	Mean_fchi_zvec; 3 theta	Mean_fchi_wvec_B; 3 B
Mean_fchi_3d_R	Mean_fchi_3d_R	Mean_fchi_xvec; 305 sites	Mean_fchi_yvec; 24 wav	Mean_fchi_zvec; 3 theta	Mean_fchi_wvec_R; 3 R
Mean_fchi_3d_V	Mean_fchi_3d_V	Mean_fchi_xvec; 305 sites	Mean_fchi_yvec; 24 wav	Mean_fchi_zvec; 3 theta	Mean_fchi_wvec_V; 3 V
Phi_data_all_thetas	New_phi_array	New_phi_xvec; 305 sites		New_phi_zvec; 3 theta	
Phi_data_B	New_phi_array_B	New_phi_xvec; 305 sites		New_phi_zvec; 3 theta	New_phi_wvec_B
Phi_data_R	New_phi_array_R	New_phi_xvec; 305 sites		New_phi_zvec; 3 theta	New_phi_wvec_R
Phi_data_V	New_phi_array_V	New_phi_xvec; 305 sites		New_phi_zvec; 3 theta	New_phi_wvec_V
Time_mean_lines_2d	Time_mean_lines_2d	Time_mean_lines_xvec	Time_mean_lines_yvec; 3 theta		
Time_mean_lines_2d_B	Time_mean_lines_2d_B	Time_mean_lines_xvec	Time_mean_lines_yvec; 3 theta	Time_mean_lines_zvec_B	
Time_mean_lines_2d_R	Time_mean_lines_2d_R	Time_mean_lines_xvec	Time_mean_lines_yvec; 3 theta	Time_mean_lines_zvec_B	
Time_mean_lines_2d_V	Time_mean_lines_2d_V	Time_mean_lines_xvec	Time_mean_lines_yvec; 3 theta values	Time_mean_lines_zvec_B	
Prepared_time_shape	Time_shape_array	Time_shape_l_array	2 rows	Time_shape_array_zvec; 24 wav	3 theta values
Time_shape_B	Time_shape_array_B				Time_shape_array_wvec_B
Time_shape_R	Time_shape_array_R				Time_shape_array_wvec_R
Time_shape_V	Time_shape_array_V				Time_shape_array_wvec_V

1.3 Phase 3 – Interpolating into the Data Arrays

The general procedure that is followed to recreate the surge response is:

1. Extract the peak height for a cyclone of the input intensity, adjusted for different *B*, *R* and V_{fm} values.
2. Extract the fraction of this peak that the input site will attain, adjusted for different *B*, *R* and V_{fm} values.
3. Modify this amended height to take into account local coastal influences, adjusted for different *B*, *R* and V_{fm} values.
4. Extract the water level multiplier for the input theta, ΔP and chosen water level.
5. Extract the representative normalized time history for this site and cyclone combination, adjusted for different *B*, *R* and V_{fm} values.
6. Adjust the shape of the normalized time history for our input site of interest as well as the input values of *B*, *R* and V_{fm}.
7. Shift the time history by an appropriate amount to give the peak at the correct time.
8. Finalize the result and write to an output file.

The steps required in each of the above stages are listed below.

Step 1 (extract the peak height and adjust appropriately)

- Interpolate into the hp_vs_delp_2d array for the input values of θ_{fm} and ΔP. This gives the model peak height
- Interpolate into each of hp_vs_delp_2d_B, hp_vs_delp_2d_R and hp_vs_delp_2d_V for the input values of θ_{fm}, ΔP and the input value of one of *B*, *R* or V_{fm}. These give the 3 factors to multiply the model peak height by to get the final peak height.
- Multiply the four values from above to get the final model peak height.

Step 2 (extract the fraction of the peak height)

- Interpolate into the overall_mean_height_2d array for the input value of θ_{fm} and the input site's *l* value. This gives the fraction of the peak height found in Step 1 that this particular site of interest will attain.
- Interpolate into each of overall_mean_height_2d_B, overall_mean_height_2d_R and overall_mean_height_2d_V for the input θ_{fm}, the input site's *l* value, and the input value of one of *B*, *R* and V_{fm}. These give the 3 factors to multiply the above adjustment by.
- Multiply the above four values to give the final fraction of the peak height obtained in Step 1 that the input site will attain.

Step 3 (modify the amended height for local coastal influences)

- Due to the way the initial processing was carried out, it is not possible for the parametric model to interpolate for *X* values. That is, it is not possible to interpolate into an array indexed by the track (the distance of the cyclone from the reference point), so alternative means are needed when the track is a variable. To overcome this, we find the closest track to the input cyclone of interest (instead of interpolating between the two closest tracks to get a better result). So if the input cyclone corresponds to an *X* value of 106 km, we simply extract the data from the necessary array for an *X* value of 110 km (which is a valid track).

- Therefore this is the first step when finding the $F\chi$ value – the data for the closest *X* value to the input cyclone is extracted from mean_fchi_3d, resulting in a 2D array. This is then interpolated into for the input θ_{fm} value and the input site number.
- Similarly when finding the adjustments to this factor due to different *B*, *R* and V_{fm} values, the first step is to extract the data from mean_fchi_3d_B, mean_fchi_3d_R and mean_fchi_3d_V for the closest *X* value to the input cyclone. This results in a series of 3d arrays that are interpolated into for the input values of *B*, *R* and V_{fm}.
- The above four values are then multiplied together to give the final adjustment for local coastal influences.

Step 4 (extract the water level multiplier)

- Simply interpolate into the water level ratio array for the input values of θ_{fm}, ΔP and water level.

Step 5 (extract the representative time history shape)

- Simply step through each timestep and extract:
  - The base time history value at that time for the input value of θ_{fm}
  - The adjustment to that height at that time due to a different value of *B*, *R* or V_{fm} for the input value of θ_{fm} and the input value of *B*, *R* or V_{fm}.
  - Multiply each of the four factors together at each timestep.
- The end result is a single vector of heights representing the final normalized time history (and corresponding to the times in time_mean_lines_yvec).

Step 6 (adjust the shape for the input site and the different parameters)

- As with Step 3, first extract the necessary data for the closest *X* value to our input cyclone track. Then interpolate into this 2d array for the input value of θ_{fm} and the input site's *l* value. This results in the multiplying factors for the negative and positive section of the time history, which are used to recreate the shape of the time profile.
- Do the same for the *B*, *R* and V_{fm} arrays and interpolate to get the adjustments to the previously extracted shape factors.
- Multiply the four negative factors to give the final adjustment factor for the negative times, and the four positive factors to give the final adjustment factors for the positive times.
- For all negative times in the time_mean_lines_yvec time vector, divide the time by the negative factor to give the final time vector. Repeat for the positive times, dividing by the positive factor. This results in a stretching or shrinking of the time history to better match the time profile shape of the input site.

Step 7 (shift the time history an appropriate amount)

- As with steps 3 and 5, extract the information for the closest X value to our input cyclone track. Then interpolate into the resulting 2d array for the input value of θ_{fm} and the input site, resulting in the amount the time history has to be shifted from zero (either positive or negative) so that the peak of the input site's time profile occurs at the correct time.
- Repeat for the different B, R and V_{fm} values to find the factors by which the above **phi** value has to be modified for the input values of B, R and V_{fm}.
- Multiply the above 4 factors together to give the final amount that the time history must be shifted.

Step 8 (create the final result and write to an output file)

- The normalized time profile can be multiplied by the final peak height value, the fraction of that peak height the site will attain, and the multiplier representing local coastal influences. This gives the correct peak height and shape for the input site's profile.
- The final value of **phi** is subtracted from the final stretched or shrunk domain to ensure that the peak occurs at the correct time.
- This final height profile and time vector are then written to an output file.