Technical Reference Documentation for

TTT

version 4.0.1

Software Development Kit

for the prediction of tsunami travel times

Geoware, September, 2018

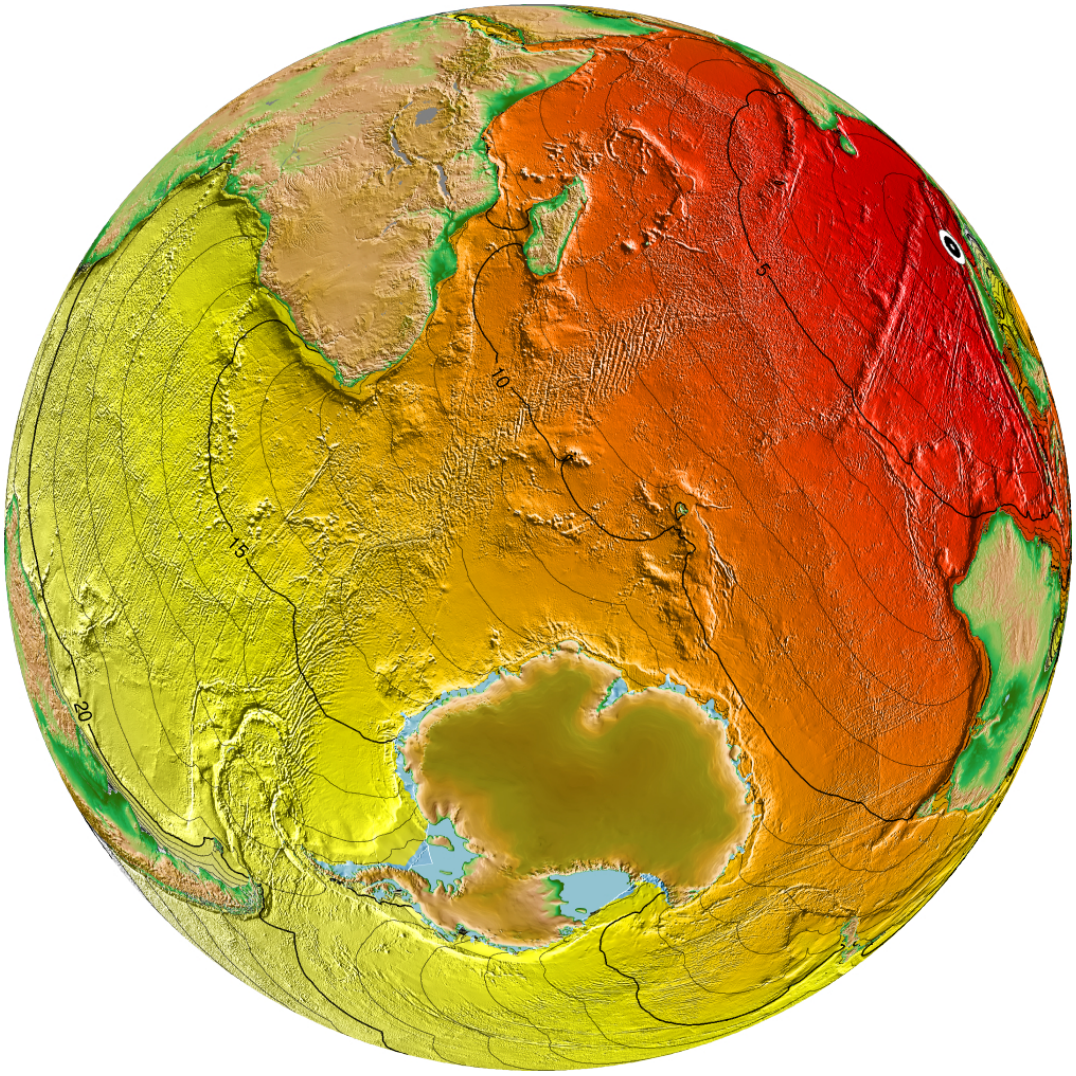


Table of Contents

0. Changes from version 3.0–4.0.1 3
1. Overview 5
2. TTT installation 6
3. TTT SDK functions 7
4. Bathymetry grids 8
5. Job parameters 9
6. Sequence of TTT calls 11
A. Huygens construction – technical details 12
B. Application programming interface 14
C. Program documentation
ttt_client man page 17
D. Analysis of Observed and Predicted Tsunami Travel Times for the Pacific and Indian Oceans[1] 20

[1] Reprint of *Wessel, P.*, 2009, Pure & Applied Geophysics, **166**, 310–324, doi 10.1007/s00024-008-0437-2.

0. Changes from version 3.0–4.0.1

4.0 to 4.0.1: Fixed a bug that crashed -N32, -N48, and -N64. Now also exit if no epicenter is given.

3.3.2 to 4.0: A major update has taken place:

1. The maximum number of nodes in the Huygens construction has been increased from 64 to 120 via the new option value –N120. This means the maximum uncertainty in the method has been reduced by a factor of 2, to about ±0.13% of the travel time. Because almost twice as many nodes are used in the construction, the number of intermediate best travel times are much larger and the calculation takes ~ 2.5 times as long as –N64 (the previous default in TTT 3.x.y). Of course, computers are now faster as well.
2. The building of the TTT SDK now uses the cross-platform configuration system CMake to handle system and library configurations.
3. We supply binary installers for Windows and OS X and installation script for Linux.

3.3.1 to 3.3.2: A few minor changes have taken place:

1. The `ttt_client.c` example tool has a new option **–P** that allows for reading back in a pre-calculated travel time grid. This way one can call ttt_client **–A** many times on the same grid without recomputing the travel times. If **–PT** is given then output formatting suitable for PTWC's TideTool is used.
2. The file given by **–A** is now optional, and if missing we read *stdin*.
3. Return NaN for travel times to stations that cannot be reached instead of error.
4. For **–A** (with or without **–Z**) we only search for the nearest non-NaN travel time node within a 125-km radius of a station.

3.3 to 3.3.1: A few minor changes have taken place:

5. The `ttt_client.c` example tool has a new option **–Z** to be used for ETA calculations by ensuring the water depth at any station exceeds the given limit [0]. It also accepts –**h**|**m**|**s** for decimal travel time format [Default is *hh mm ss*].
6. The meta-data in the travel time grid header now contains the origin time of the earthquake, if it was provided.
7. A minor bug when reading across periodic grid boundaries has been fixed.
8. The API has undergone some minor changes to allow the changes 1–2 above.

3.2 to 3.3: Several important changes have taken place:

1. The TTT SDK is now internally 64-bit compliant and can handle huge grids if compiled with the 64-bit flags (see configure for using --enable-64 and --enable-large; there is also --enable-universal to build Universal binaries under OS X). There is also an --enable-32 option to force 32-bit compilation.

2. New option –**D***zmin* for ttt_client, which limits how shallow bathymetry can be for slowness calculations. If *zmin* > 0 then depths between 0 and 2**zmin* are quadratically tapered to go from *zmin* to 2**zmin* instead. To implement this feature we have extended the 7-item parameter list to 8. The TTT uncertainty estimate is still returned as the 7th parameter, and *zmin* is input as the 8th.
3. We have reprocessed the bathymetry files based on the ETOPO1 data set and added ttt_topo_1m.i2 for global 1x1 arc minute calculations.
4. We have also added a global 30x30 arc second grid (based on SRTM30+ [Becker et al, 2009] for high-resolution calculations. For global calculations this may require the --enable-64 settings discussed above.
5. Minor improvements in precision by using 0.2 ms time unit internally.

3.1 to 3.2: The main change from version 3.1 to 3.2 has to do with the argument list for the functions `ttt_calc_eta` and `ttt_save_eta`. These now pass their list of ETA stations as an array of structures so that ETA estimates can be sorted on arrival time. The older functions that took separate arrays are still available as `ttt_calc_eta2` and `ttt_save_eta2`.

3.0 to 3.1: Ongoing development of the tsunami travel time tools have led to several significant improvements that have now been encoded into the TTT Software Development Kit. These enhancements are:

1. Ability to handle global grids with appropriate geographic boundary conditions.
2. Dramatic reduction in overall execution time by introducing smart, binary tree management for internal queue. The finer the grid the bigger the time savings. E.g., for Pacific-wide calculations on the 5x5 min grid the savings are ~75%.
3. Option to specify a sub-region when providing a grid.

Apart from having to add an explicit **–R** option to the `ttt_client.c` example tool, these enhancements were all implemented without having to change the formal API interface specifications (Appendix B). However, given the ability to extract subsets we now only supply a global rather than three regional grids. The single global grid is of course provided for each of the usual grid resolutions.

1. Overview

The TTT SDK facilitates predictions of tsunami travel times on a geographic (lat–lon) grid derived from a supplied bathymetric data grid. By assuming a long-wave approximation the propagation speed, v, of the tsunami front (i.e., the first arriving wave) is given by

$$v(x,y) = \sqrt{g(y) \cdot d(x,y)} \tag{1}$$

where g is the normal gravitational acceleration (considered a function of the latitude, y) and d is the water depth (positive down). TTT uses Huygens circle constructions to integrate the travel times from the epicenter to all nodes on the grid, i.e., we must add up increments of the form

$$\Delta t(r) = \int_0^r \frac{dx}{v(x)} = \int_0^r s(x)dx \tag{2}$$

where r is the distance from the current node to another node that lies on a circle of radius r and $s(x) = v^{-1}(x)$ is the slowness along the path. The slowness along this radial line is represented as a piecewise, linear function derived from the grid of velocities. The circles are necessarily approximated by polygons with up to 120 nodes to minimize any directional bias; thus a completely flat bottom bathymetry results in travel time contours that are close to concentric circles, and the travel times are everywhere within of 0.13% of the theoretical values (Appendix A).

The TTT SDK comes with a linkable library that contains eight subroutines (functions) that application programmers may call upon to (1) specify a bathymetry data set and other required parameters, (2) carry out the calculations of tsunami travel times, (3) submit queries regarding arrival times at specific locations, and (4) save the grid or the arrival times to disk files. The SDK is implemented in C following the portable POSIX standard, and all functions are callable from a range of programming environments since the functions only use the basic data types of integer, floating-point values and character strings. Fortran bindings are provided in the library.

The TTT SDK is distributed with a fully functioning C client demo application that shows how applications may call the TTT library functions. This application, called **ttt_client**, simulates the functionality of the application **ttt** shipped in older versions of TTT.

The installation of TTT is discussed in Section 2. The required TTT functions are summarized in Section 3.1 while the optional functions are listed in Section 3.2; all are available via the SDK, and details of the use of these functions is provided in Appendix B. The details of bathymetry grid formats are discussed in Section 4, while an overview of the job parameters can be found in Section 5. Finally, in Section 6 we demonstrate how each of the eight functions might be called from a typical application written in C.

2. TTT Installation

The installation of the TTT SDK library is straightforward. We distinguish between two programming environments:

2.1 Unix-like systems

This environment includes all standard UNIX systems (e.g., Solaris, HP-UX, IRIX), all Linux distributions, Mac OS X, Cygwin, and Microsoft Services For Unix (SFU). A standard approach is taken:

1. Uncompress and un-tar the distribution package to create the tttAPI directory and all included files:
   tar xzvf tttAPI.tbz *or* gzip -dc tttAPI.tbz | tar xvf -
2. Run the configure script
   ./configure [--enable-64 --enable-large]
   If you have an unusual setup or require specific compiler flags you may want to set these first via the environmental parameters **CC** and **CFLAGS** prior to running configure. See ./configure --help for how to set up 64-bit compilation.
3. Run the makefile to install the shared library and create the example client application:
   make install
4. Run the test example to verify a successful installation:
   make test

If any of the steps fail you may need to examine your programming environment and make sure standard tools such as make, C compiler (cc or gcc), ar, and ranlib have been installed. The configure script recognizes most systems and will provided preset compiler flags; you can override these as described under the configure description.

2.2 Microsoft Windows

Simply run the tttAPI installer, which will take you through the installation of data grids and program executables.

3. TTT SDK functions

The library contains a total of eight functions. Of these, two are required in order to obtain travel times, while the remaining six are optional and only provided to facilitate program development.

3.1 Travel time calculation functions

These two functions are the main functions that are required to undertake travel time calculations and estimation of arrival times.

1. **`ttt_calc_ttt`**: This function accepts a bathymetry grid, earthquake location(s), and job parameters, and replaces the bathymetry with a grid of calculated tsunami travel times (in hours).
2. **`ttt_calc_eta`**: This function allows one to use the travel time grid and a set of station locations to estimate the arrival times at this set of discrete locations. It assumes that **`ttt_calc_ttt`** has been called first.

3.2 Travel time utility functions

The following six functions can facilitate the use of the main TTT functions by allowing programmers to load in bathymetry grids and save travel time grids and tables of estimated arrival times. Of course, programmers can choose to accomplish these tasks in any other way depending on the functionality they want implemented.

1. **`ttt_load_gmtgrid`**: This convenience function allows users to load in bathymetry grids prepared with the Generic Mapping Tools (GMT), including the premade bathymetry grids that are distributed with TTT.
2. **`ttt_save_gmtgrid`**: This function allows users to write the travel time grid to a file, using either a 2-byte short or a 4-byte float GMT file format.
3. **`ttt_save_eta`**: This function can be used to save the estimated arrival times to a select group of stations to an ASCII file on disk.
4. **`ttt_date2time`**: Utility function that converts the time of an earthquake (given by year, month, day, hour, minute, and seconds) into a single integer time-representation known as UNIX time (seconds since 1-Jan-1970). When this time is passed to **`ttt_calc_eta`** we obtain absolute arrival times rather than just travel times relative to an arbitrary origin time.
5. **`ttt_version`**: This function simply returns a text-string which describes the current version of the API library.
6. **`ttt_message`**: This function can be used to write out error messages that corresponds to the error numbers returned by other functions.

4. Bathymetry grids

Bathymetry must be provided on a geographic (latitude–longitude) equidistant grid. The data unit must be meters, with negative values indicating water depth and positive values (or NaNs) indicating land. The data must be loaded into memory by the user's application before travel time calculations can be started. If the bathymetry data reside in a GMT native binary grid (either in 2-byte, short integer format or 4-byte floating-point format), it is convenient to use the **`ttt_load_gmtgrid`** function. Should it be desirable to obtain data by a different mechanism, make sure that the process will eventually provide a pointer to a 2-D, 4-byte floating point array and a set of associated parameters that describe the grid. The following is the list of required parameters:

- We need a double precision array of length 4 that holds the min and max longitude and min and max latitude defining the rectangular data region.
- We need a 4-byte integer array of length 3 that holds the number of columns and rows in the grid, as well as the dimension of the fastest-changing vector (row or column dimension depending on your array structure). For instance, if you have your custom bathymetry in a grid in memory which is essentially organized in columns going from west to east, with the first value in each column being at the northern latitude, you have a "Fortran column format" and must use the dimension of the column as the 3rd entry into the dimension array. If you allocated the memory to exactly fit your grid then this dimension will be the same as the column length, but if your grid is smaller than your allocated memory you must supply a column length that reflects this larger length.
- Specify the grid registration. Grids are either gridline (0) or pixel (1) registered.
- A format identifier describes how the data is structured in the 2-D array (see discussion on dimensions above).

Details of these parameters are provided in Appendix B.

5. Job parameters

In addition to accepting the bathymetry data and descriptors for the data domain, **`ttt_calc_ttt`** also accepts (1) information about the earthquake epicenter and (2) parameters controlling the calculation of travel times.

While in most situations the epicenter will be a single (latitude, longitude) pair, there may be situations where enough information is known about the seismic source to warrant a more accurate description of the shape of the source. Thus, it is possible to provide a list of locations that collectively define the source region (e.g., a line approximation). The order of the points is not significant.

There are seven job control parameters that can be specified via a double precision array of length 8; the 7th position is used to output an uncertainty factor. The parameters, given in the order they must appear in the array, are:

1. **nodes:** The 1st parameter controls how many neighboring nodes should be used in the Huygens construction approximation. Valid choices are 8, 16, 32, 48, 64, and 120, with the higher numbers giving the best approximations. [120]
2. **search:** One of several tests is to check if any of the epicenter locations fall on land (as determined by the supplied bathymetry grid), and if so give an error message. However, if the 2nd argument is set to 1 we will try to move the epicenter to the nearest node that is over water. This search uses the two parameters **radius** and **depth** described below. [0]
3. **radius:** The 3rd parameter sets the search radius to use in the "on-land" test. If set to 0.0 we use the default radius of 5 spherical degrees; otherwise we use the supplied radius value (in spherical degrees). If no water-node is found within the search radius we return an error. [5]
4. **depth:**. In addition to move the epicenter to a node over water, we can select that the final node must have a water depth larger than a given threshold. Set this threshold with the 4th parameter; note the depth must be negative. The purpose of this constraint is to avoid long travel-time delays due to propagation in very shallow water (see Appendix C). [0]
5. **ignore-bias:** When circles are approximated on a rectangular grid we end up constructing *n*-sided polygons, with *n* being the **nodes** parameter discussed above. This approximation leads to a slight over-estimation of travel times. We adjust for this by normalizing all travel times by the square root of the ratio between the areas of the polygon and the circle. While we recommend this normalization, one may want to disable it for testing purposes; if so supply 1 as the 5th parameter. [0]
6. **verbose:** Normally the calculations are performed without any intermediate feed-back. By setting the 6th parameter to 1 we will report progress to standard error (usually the console). This is useful for testing but also for assuring the user that the calculations are in progress. [0]
7. **tt_error:** Output uncertainty estimate.
8. **min_depth:** [New] Normally we use the bathymetry values directly to calculate slowness. However, areas of very shallow bathymetry may cause unreasonably long travel times. One way to investigate this effect is to impose a minimum water

depth via this setting. If non-zero, we reset water depths between 0 and 2* **min_depth** to ramp from **min_depth** to 2* **min_depth**, using a quadratic equation.

If the earthquake is a single point then the 7th parameter position will hold a travel-time uncertainty factor, giving an indication of how sensitive the travel time is to positional errors in the epicenter location. The factor has units of seconds/km; multiply by your estimate of positional error to see the corresponding travel time uncertainty.

6. Sequence of TTT calls

The following is the core of the **`ttt_client.c`** application; it illustrates what steps a typical application need to take. Here, we show usage of all eight SDK functions.

```
/* This section of the code shows the steps relevant to the TTT SDK */

ttt_version (path); /* Report to stderr which version of the library is used */

/* Load in a preexisting GMT bathymetry grid or give error message */

if ((err = ttt_load_gmtgrid (path, wesn, dim, &reg, in_format, &z)))
   ttt_message (argv[0], error);

/* Calculate travel times from bathymetry and return values in bathy array */

if ((err = ttt_calc_ttt (wesn, dim, reg, 0, z, n, q_lon, q_lat, params)))
   ttt_message (argv[0], error);

/* Save the travel times to a GMT-compatible grid */

if ((err = ttt_save_gmtgrid (tttfile, wesn, dim, reg, out_format, z)))
   ttt_message (argv[0], error);

/* Get Unix time based on the user's date/clock of the earthquake */

q_time =  ttt_date2time (yy, mm, dd, hh, mi, ss, UTC);

/* Determine travel times to a list of stations */

if ((err = ttt_calc_eta (wesn, dim, reg, z, n_sites, eta)))
   ttt_message (argv[0], error);

/* Write out sorted estimated arrival times in UTC time to standard output */

if ((err = ttt_save_eta ("", n_sites, eta, q_time, UTC)))
   ttt_message (argv[0], error);

/* That's it, we are done */
```

This example is implemented in POSIX C, just like the tttAPI library itself. The calling sequence should be identical from C++ and Fortran programs (using the provided bindings). For other environments (e.g., Java), please consult the technical references for using external native C functions.

Appendix A: Huygens Construction Technical Details

For TTT 4, we analyzed all possible rays in the Huygens construction and determined the most isotropic distribution for the fewest rays. By requiring that there should be at least one ray within every 3°-wide sector we needed to consider a Huygens radius of 13 nodes. Fig. A1 shows the nodes (for one octant; they are all symmetric) that we are computing incremental travel times to via integration of the slowness grid interpolated along each ray. We keep track of the shortest travel time to any node beyond the current wave front.

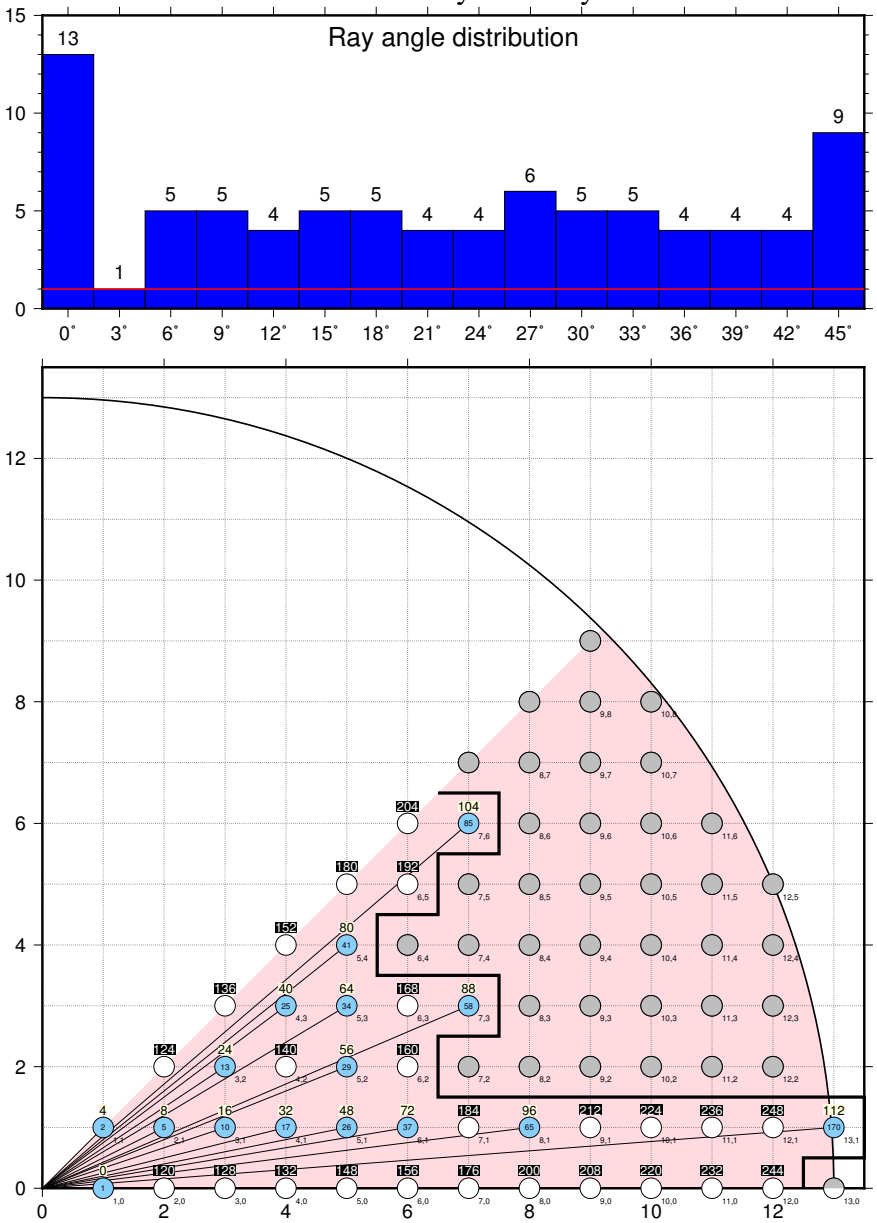


Fig. A1. (bottom) Distances to the various nodes are calculated using a flat-earth approximation (where the *y*-distance is scaled by the cosine of the latitude). For instance, the line from the central node to node #56 is internally referred to as $\sqrt{5^2+2^2}=\sqrt{29}$. If $\Delta x = \Delta y$ then the distance would be $\Delta x\sqrt{29}$; of course, here Δy varies with latitude. By insisting we have rays in every direction to the nearest 3° we pick the blue nodes only. (top) For a 3° sector width we see all directions have at least one or more rays.

Fig. A1 (bottom) shows the central node (at 0,0) and all neighboring nodes considered when calculating incremental travel times (we are just showing one octant but symmetries prevail across every direction that is a multiple of 45˚. The central node represents one of the grid nodes that currently lie on the tsunami front (at time 0, this is typically a single node at the epicenter). From there we calculate travel times to our selected 120 nodes (blue); these lie on 16 concentric circles. White nodes do not need to be calculated since they can be reached indirectly via intermediate white nodes and repositioning of the central node. Gray nodes are beyond our search as they do not improve isotropy very much.

For each ray to a blue node, linear interpolation along radial lines is used to develop the slowness function $s(x)$. Figure A1 shows the geometries that have been used. For instance, the travel time from the current node to the node marked 16 is a 3-term function: For each tile, we express how $s(x)$ varies linearly along this path, integrate the terms analytically over the piecewise distances, and add up the three terms. The 120 separate paths considered yield 120 separate expressions, which are coded separately and stored as macro definitions. The distances and constants used in these macros vary with latitude. Note that tttAPI considers grids representing the full 360˚ longitude range or grids including the north or south poles; in both cases, we apply geographical boundary conditions.

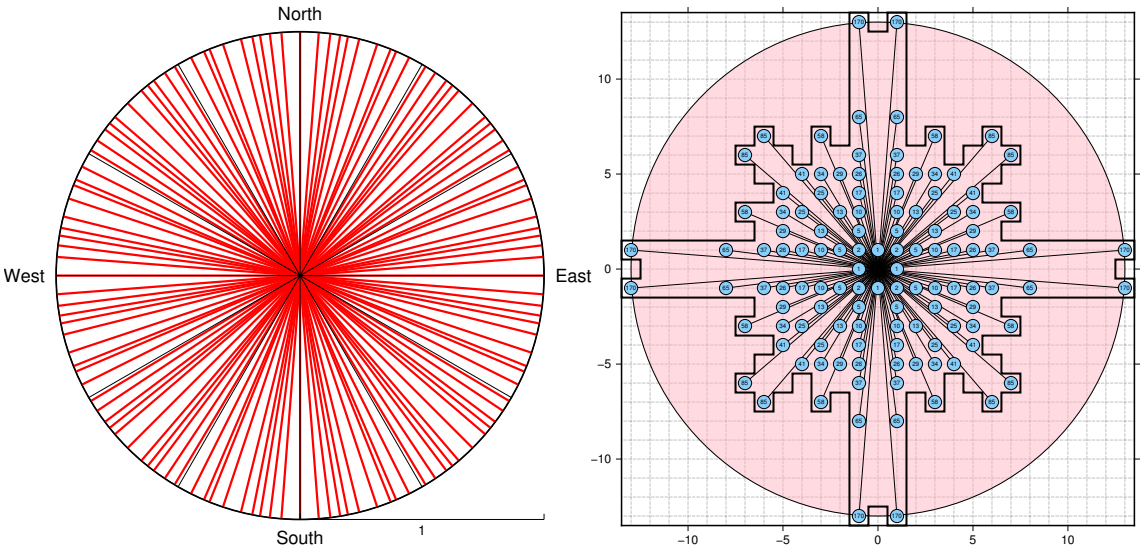


Fig. A2. (left) Ray distribution resulting from the chosen nodes in Fig. A1. The distribution is close to isotropic, with small variations in angular separation between neighboring rays. The biggest weakness of the Huygens construction method as implemented by TTT is along the cardinal direction. (right) By extending our search to a 13-node radius but only considering the selected nodes we have reduced the theoretical maximum error by a factor of two. This pointy 120-node star is the basis for the most isotropic Huygens construction with this number of nodes.

Appendix B: Application Programming Interface

Here you will find the exact prototypes of each function in the TTT SDK library.

```
extern int ttt_calc_ttt (double wesn[], int dim[], int
 registration, int format, float * topo, struct QUAKE *Q,
 double parameters[])
```

wesn:	Specifies west longitude, east longitude, south latitude, and north latitude of the rectangular grid domain, in decimal degrees.
dim:	Specifies the number of columns, the number of rows, and the length of the fastest-changing dimension (row or column) in your input grid.
registration:	Specify 0 for gridline- and 1 for pixel-registration.
format:	Specifies the internal order of your grid matrix: (1) grid has rows LR starting with top row (scan lines), (2) grid has rows LR starting with bottom row, (3) grid has rows RL starting with top row, (4) grid has rows RL starting with bottom row, (5) grid has columns TB starting with left column, (6) grid has columns TB starting with right column, (7) grid has columns BT starting with left column, (8) grid has columns BT starting with right column.
topo:	Name of the bathymetry grid variable.
Q:	Structure with information about the earthquake.
parameters:	List of 7 job input parameters; (1) nodes, (2) search, (3) radius, (4) depth, (5) ignore-bias, (6) verbose, (8) min_depth, and one output parameter (7) uncertainty rate; multiply by uncertainty in epicenter position (in km) to get corresponding travel time uncertainty. Array must be at least 8 items long.

```
extern int ttt_calc_eta (double wesn[], int dim[], int
 registration, float * ttt, float *z, double zdepth, int
 n_sites, struct ETA eta[])
```

wesn:	Specifies west longitude, east longitude, south latitude, and north latitude of the rectangular grid domain, in decimal degrees.
dim:	Specifies the number of columns, the number of rows, and the length of the fastest-changing dimension (row or column) in your input grid.
registration:	Specify 0 for gridline- and 1 for pixel-registration.
ttt:	Name of the travel time grid variable.
z:	Name of the co-registered depth grid, if need to limit search to nodes deeper than `zdepth` (see below). Pass NULL if not used.
zdepth:	Shallowest depth to select ETA for a station (see `z` above).
n_sites:	The number of points for which we desire estimates of arrival times.
eta:	Array of structures that contain these attributes for each station double longitude, latitude [station location] double eta [arrival time estimate]. double dist [distance from station to nearest point on the grid from which the eta was estimated]

double rate [Uncertainty rates in sec/km for location; multiply by uncertainty in position (in km) to get corresponding travel time uncertainty.

```
extern int ttt_load_gmtgrid (char * file, double wesn[], int
 dim[], int * registration, int format, struct QUAKE *Q,
 float ** z);
```

file:	Name of the input file from which we will read bathymetry grid using the specified GMT grid format.
wesn:	On input contains the coordinates of a subset, or all zeros to return the entire grid. Returns west longitude, east longitude, south latitude, and north latitude of the rectangular grid domain, in decimal degrees.
dim:	Returns the number of columns, the number of rows, and the length of the fastest-changing dimension (row or column) in your input grid.
registration:	Returns 0 for gridline- and 1 for pixel-registration.
format:	Set to 1 for 4-byte floating-point grid and 2 for 2-byte short int grid.
Q:	Structure to hold information about the earthquake if reading a precalculated travel time grid; otherwise pass NULL.
z:	Name of the pointer to the travel time grid.

```
extern int ttt_save_gmtgrid (char * file, double wesn[], int
 dim[], int registration, int format, struct QUAKE *Q, float
 * ttt);
```

file:	Name of the output file for travel time grid in GMT format.
wesn:	Specifies west longitude, east longitude, south latitude, and north latitude of the rectangular grid domain, in decimal degrees.
dim:	Specifies the number of columns, the number of rows, and the length of the fastest-changing dimension (row or column) in your input grid.
registration:	Specify 0 for gridline- and 1 for pixel-registration.
format:	Set to 1 for 4-byte floating-point grid and 2 for 2-byte short int grid.
Q:	Structure with information about the earthquake.
ttt:	Name of the travel time grid variable.

```
extern int ttt_save_eta (char * file, int n_sites, struct ETA
 eta[], time_t q_time, int UTC)
```

file:	Name of the output file where we will write the ASCII eta table.
n_sites:	The number of points with estimates of arrival times.
eta:	Array of structures with information for all station.
lat:	See `ttt_calc_eta` for details of structure attributes.
q_time:	UNIX time of earthquake (s from Jan 1, 1970), or 0 for relative time.
UTC:	1 for saving time in UTC, 0 for using local time (requires valid nonzero q_time).

```
extern int ttt_version (char * prefix)
```

prefix:	Prefix used to report the version of the current TTT library.

```
extern int ttt_message (char * prefix, int error)
```

prefix: Prefix used to report the error message associated with the error number.
error: Error number, presumably returned by one of the above functions.

```
extern time_t ttt_date2time (int year, int month, int day,
  int hour, int minute, int second, int UTC)
```

year: Year of the earthquake (e.g., 2005).
month: Month of the earthquake (1–12).
day: Day of the earthquake (1–31).
hour: Hour of the earthquake (0–23).
minute: Minute of the earthquake (0–59).
second: Second of the earthquake (0–59).
UTC: 1 for returning time in UTC, 0 for using local time.

List of return values:

Code	Explanation
TTT_ERROR_BAD_NODES	Nodes not 8|16|32|48|64!
TTT_ERROR_UNKNOWN_FORMAT	Grid format not recognized!
TTT_ERROR_WESN_CHECK	West, east, south, and north not in proper order!
TTT_ERROR_DIM_CHECK	Grid nodes zero or negative (increment bad?)!
TTT_ERROR_INC_CHECK	Grid increment is <= 0!
TTT_ERROR_NO_SOURCE	No epicenter given!
TTT_ERROR_SOURCE_ON_LAND	Epicenter lies on land!
TTT_ERROR_SOURCE_OUTSIDE	Epicenter lies outside selected region!
TTT_ERROR_SOURCE_BAD_SEARCH	Unable to relocate epicenter to nearest node over water!
TTT_ERROR_BAD_VERBOSITY	Verbosity must be 0 (FALSE) or 1 (TRUE)!
TTT_ERROR_BAD_RADIUS	Search radius is negative!
TTT_ERROR_BAD_DEPTH	Epicenter depth threshold is above sealevel!
TTT_ERROR_ALLOCATE	Unable to allocate requested memory!
TTT_ERROR_REALLOCATE	Unable to reallocate memory!
TTT_ERROR_BAD_MONTH	Month number must be between 1 and 12 inclusive!
TTT_ERROR_BAD_DAY	Day number must be between 1 and 31 inclusive!
TTT_ERROR_BAD_HOUR	Hour number must be between 0 and 23 inclusive!
TTT_ERROR_BAD_MINUTE	Minute number must be between 0 and 59 inclusive!
TTT_ERROR_BAD_SECOND	Second number must be between 0 and 59 inclusive!
TTT_ERROR_FILE_R_ERROR	Error opening file for reading!
TTT_ERROR_FILE_W_ERROR	Error opening file for writing!
TTT_ERROR_HEADER_R_ERROR	Error reading grid header
TTT_ERROR_HEADER_S_ERROR	Error trying to byte-swab header
TTT_ERROR_HEADER_W_ERROR	Error writing grid header
TTT_ERROR_ROW_R_ERROR	Error reading grid data
TTT_ERROR_ROW_W_ERROR	Error writing grid data
TTT_ERROR_STATION_ON_LAND	Station location for ETA calculation is on land!
TTT_ERROR_STATION_OUTSIDE	Station location for ETA calculation is outside region!
TTT_ERROR_STATION_BAD_SEARCH	Unable to relocate station to nearest node over water!
TTT_ERROR_BAD_DOMAIN	Sub-region not compatible with grid domain!
TTT_ERROR_BAD_MINDEPTH	Shallow water depth threshold is above sealevel!

NAME

ttt_client v. 4.0.1 – Calculate tsunami travel times on a grid

SYNOPSIS

ttt_client [*input-file*] [**–A**[*stations.txt*]] [**–B**] [**–D***zmin*] [**–E***lon/lat*] [**–e***sources*] [**–I**] [**–N***nodes*] [**–O|o***yyyy/mm/dd/hh/mi/ss*] [**–P**[**T**]] [**–R***west/east/south/north*] [**–S**[*radius*[*depth*]]] [**–T***tttfile*] [**–U**] [**–V**[**L**]] [**–Z***depth*] [**–h|m|s**]

DESCRIPTION

ttt_client is an example of an application built to use the TTT SDK functions. It can calculate predicted tsunami travel-times using a provided bathymetry grid. Note for most operating systems the **ttt_client** executable(s) are called **ttt_client32** and/or **ttt_client64** to reflect if they are 32- or 64-bit executables; this man page simply use the name **ttt_client**. It may use the directory pointed to by the environmental variable **TTT_DIR** as the place in which to look for bathymetry. If this variable is not set then only the current directory is used. To set this variable to point to the directory *your_dir*, try one of

setenv TTT_DIR *your_dir* [for cshell]
export TTT_DIR=*your_dir* [Bourne shell]
set TTT_DIR=*your_dir* [DOS]

The following command line arguments are recognized:

input-file

ttt_client attempts to decode *input-file* using the following sequence of steps:

1. If *input-file* ends in ".b" it is read as a bathymetry grid [GMT native float format].
2. If *input-file* ends in ".i2" it is read as a bathymetry grid [GMT native short format].
3. If **$TTT_DIR/***input_file*.i2 exists it will be read as a bathymetry grid.

Input bathymetry data must be in GMT float or short int binary native format (see FILE FORMATS). To avoid ambiguities in travel time construction it is best that landlocked bodies of water be set to positive (topography) values or NaN (== 32767 for short int). Grids may be either pixel- or gridline-registered, and geographical boundary conditions will be applied for global grids. Note that the *input-file* is only required when calculating the travel time grid or when sampling it later while **–Z** is also used.

OPTIONS

–A Specify filename with multiple stations (records of lon lat name) to which we will estimate ETAs. Results are written to stdout. if no filename is given we will read standard input. We search for the nearest non-NaN travel time node within 125 km radius of the station. If the search fails we print a warning and return a travel time of NaN.

–B Do NOT normalize the travel times by minimizing the geometric prediction error (caused by the rectangular nature of grids). By default, the travel times are normalized to yield approximately a zero mean prediction error.

–D Specify a minimum water depth *zmin* [0]. For depths shallower than 2**zmin* the depth will be adjusted to go quadratically from *zmin* to 2**zmin* instead of 0-2**zmin*.

–E Sets the location of the epicenter. This location will be rounded off to the nearest grid node. If the given epicenter is on land the program will notify the user and exit (but see **–S**). Give each coordinate in the format *ddd:mm:ss.xx*, *ddd:mm.xx*, or *ddd.xx*. Use signed degrees or append W, E, S, or N to indicate hemisphere.

–e Instead supply an ASCII table with a list of coordinates that make up a non-point source rupture zone. Each record must have longitude and latitude in the first two columns.

–I Store output travel times using 2-byte short ints (in deca-seconds) rather than the default 4-byte floats (in hours).

−N Sets the number of nodes to use in the Huygens construction. Choose among 8, 16, 32, 48, 64, and 120. More nodes gives more precise results but are slower to execute [Default is 120].

−O Specify UTC Origin time of earthquake to be included in the output file's header. Use lower case **−o** if local time is used.

−P Instead of calculating travel time, just read in a pre-calculated travel time grid, specified via the file given in **−T**. The *input-file* is only required if **−Z** is used. Append **T** to simulate the output formatting expected by TideTool.

−R Specify a subset of he given grid [Default uses entire grid]. Give each coordinate in the format *ddd:mm:ss.xx*, *ddd:mm.xx*, or *ddd.xx*. Use signed degrees or append W, E, S, or N to indicate hemisphere.

−S If the given epicenter(s) is on land the program will find the nearest node that is over water. Optionally, specify a search radius in spherical degrees [Default is 5]. If a search radius is set, you can furthermore choose to add */depth* which means we want to find the nearest node over water where the water depth is deeper than *depth* (very shallow depths can lead to long travel time delays) [Default depth is 0].

−T Sets the name of the output gridfile with travel times in hours (but see **−I**). [Default is ttt.b]

−U Use UTC time when reporting ETAs (requires **−A**) [Default is UTC].

−V Verbose operation. The program will inform you about the various steps taken. Append **L** to get progress messages from within the tttAPI library.

−Z For each station given in **−A**, search for the nearest point where the depth is below the given *depth* setting [0].

−h Write travel time output in decimal hours.

−m Write travel time output in decimal minutes.

−s Write travel time output in decimal seconds.

FILE FORMATS

Bathymetry files may be prepared by the user or you may use the files supplied by Geoware. The public-domain software package GMT is highly recommended to assist in data manipulations. The bathymetry grids (in meters positive above sea level) must be in the GMT format # 1 (or 2): native float (or short int) binary file format. This file contains a header structure (see **tttAPI_lib.h** for technical information) followed by the rows of the 4-byte float (or 2-byte short int) grid in scanline orientation. We recommend using the suffix ".b" (or ".i2") for all files using this format. The output travel time grid uses the GMT format # 1: native 4-byte floating point binary file format, with the same header structure and scan-line orientation as above. We recommend the suffix ".b" for these files. However, see **−I**). See the GMT documentation on how to make these suffices automatically recognized by GMT programs.

EXAMPLES

To obtain predicted tsunami travel times for an earthquake in the Aleutians and restrict calculations to the Pacific subset of the global 10 arc minute grid, try

ttt_client ttt_topo_10m -E-175/50 -Taleutian.b -V -R90/295/-72/65

To model the global propagation from the Sumatra 2004 tsunami using the locations in the file rupzone.txt as a non-point source and the 5 arc minute global bathymetry, try

ttt_client ttt_topo_5m -erupzone.txt -TSumatra_2004.b -V

To extract estimated tsunami travel times for this earthquake to specified lon, lat locations (stored in a simple multicolumn ASCII table *stations.txt* where the first two columns must be longitude and latitude), run:

ttt_client ttt_topo_5m -erupzone.txt -TSumatra_2004.b -V -Astations.txt > ETA.txt

USING TTT_CLIENT OUTPUT WITH GMT

GMT programs can read the binary *.b format created by **ttt_client** as well as the bathymetry files in the *.i2 format. However, GMT must be "told" what format a gridded file is in. When using the *.b files, simply append the string =bf to the filename; when using the *.i2 files, append =bs/1/0/32767 (See script ttt_fancy.sh for usage). See GMT documentation for how to implement a shorthand file suffix mechanism.

USING TTT_CLIENT IN 64-BIT MODE

ttt_client is fully 64-bit compliant and can be compiled as such. This is only required if you plan to use such large grids that 32-bit addressing is not enough. Also, some operating systems limits the memory of any given process to 4 Gb. During the configuration, you can use --enable-64, which will set up the compiler flags to obtain 64-bit executables. Furthermore, if you are using a grid that contains more than 2^31 nodes you need to add --enable-large, which will use 8-byte integers to store node indices. Note: Even for the largest grid currently supplied (global 30x30 arc seconds) the 32-bits offered by regular integers are adequate. Only use --enable-large if you need to use global grids that exceed 2^31 nodes, e.g., a global grid with smaller than 20 arc second spacing.

REFERENCES

The TTT SDK 3.3 Documentation

The GMT home page at URL http://gmt.soest.hawaii.edu, and the Geoware home page at URL http://www.geoware-online.com.

Wessel, P., and W. H. F. Smith, 2017, The Generic Mapping Tools (GMT) version 5.4.2 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., W. H. F. Smith, R. Scharroo, J. F. Luis, and F. Wobbe, 2013, Generic Mapping Tools: Improved version released, Eos Trans. AGU, 94(45), 409-410, doi:10.1002/2013EO450001.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of the Generic Mapping Tools Released, EOS Trans. AGU, 79, p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans. AGU, 76, p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, http://www.agu.org/eos_elec/95154e.html, Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free software helps map and display data, EOS Trans. AGU, 72, 441.

AUTHOR

Geoware, c/o Paul Wessel, 91-1350 Kaikohola Street, Ewa Beach, HI 96706−3901, USA.
(geoware@geoware-online.com)

Pure appl. geophys. 166 (2009) 301–324
0033–4553/09/010301–24
DOI 10.1007/s00024-008-0437-2

Pure and Applied Geophysics

Analysis of Observed and Predicted Tsunami Travel Times for the Pacific and Indian Oceans

Paul Wessel

Abstract—I have examined over 1500 historical tsunami travel-time records for 127 tsunamigenic earthquakes that occurred in the Pacific and Indian Oceans. After subjecting the observations to simple tests to rule out gross errors I compare the remaining reports to simple travel-time predictions using Huygens method and the long-wave approximation, thus simulating the calculations that typically take place in a tsunami warning situation. In general, I find a high correspondence between predicted and reported travel times however, significant departures exist. Some outliers imply significantly slower propagation speeds than predicted; many of these are clearly the consequences of observers not being able to detect the (possibly weak?) first arrivals. Other outliers imply excessively long predicted travel times. These outliers reflect peculiar geometric and bathymetric conditions that are poorly represented in global bathymetric grids, leading to longer propagation paths and consequently increased travel times. Analysis of Δt, the difference between observed and predicted travel time, yields a mean Δt of 19 minutes with a standard deviation of 131 minutes. Robust statistics, being less sensitive to outliers, yield a median Δt of just 18 seconds and a median absolute deviation of 33 minutes. Care is needed to process bathymetry to avoid excessive travel-time delays in shallow areas. I also show that a 2×2 arc minute grid yields better results that a 5×5 arc minute grid; the latter in general yielding slightly slower propagation predictions. The largest remaining source of error appears to be the inadequacy of the point-source approximation to the finite tsunami-generating area.

Key words: Tsunami travel-time prediction, statistics, bathymetry.

1. Introduction

Historically, the Pacific has experienced several basin-wide tsunamis following large tsunamigenic earthquakes from various areas of the subducting plate boundary (e.g., Dudley, 1998). Of particular importance is the April 1, 1946 Aleutian earthquake whose powerful tsunami led to widespread destruction and numerous deaths (e.g., Shepard *et al.*, 1950); it also gave birth to the early U.S. tsunami warning system. In contrast, tsunamigenic earthquakes in other oceans have been much less frequent and thus warning centers were generally lacking; the calamitous 2004 Sumatra tsunami has now ushered in a new era in tsunami detection and preparedness. Designed to monitor their regions for potentially destructive tsunamis, warning centers, such as the U.S. Pacific and Alaska

Department of Geology and Geophysics, School of Ocean and Earth Sciences and Technology, University of Hawaii at Manoa, 1680 East-West Rd. Honolulu, HI 96822, U.S.A. E-mail: pwessel@hawaii.edu

tsunami warning centers, must routinely evaluate predicted tsunami travel times from the epicenters of potentially tsunamigenic earthquakes. Typically, it is not known until tide gauge or tsunameter data become available whether or not a particular large earthquake has generated an ocean-wide tsunami. In the mean time, the authorities may calculate travel times to a large number of tide stations and warning points in the Pacific. These estimated times of arrival (ETA) are incorporated into various communications from the warning agencies to local, state, and international civil defense agencies so that first responders will have an accurate estimate of when the first wave is likely to arrive. Because the premium is on responding quickly in a possible emergency situation, many warning agencies employ a rapid first-arrival methodology where no dynamic calculation of the waves is performed; i.e., no prediction of wave amplitude is attempted. Such dynamic calculations require detailed knowledge of the source, are usually done after an event, and may require considerable computational power (e.g., KOWALIK *et al.*, 2005). Simple estimates can be obtained by using the long-wave approximation (e.g., MADER, 2004; MEI, 1989), i.e., it is assumed that the tsunami will propagate away from the epicenter at a velocity given by

$$v(\vec{x}) = \sqrt{g(\vec{x})z(\vec{x})}, \tag{1}$$

where g is the vertical gravitational attraction, z is the local water depth, and $\vec{x}$ is the position vector. The program TTT from Geoware (GEOWARE, 2007) calculates these velocities based on an input bathymetry grid and uses Huygens' constructions to propagate the wave front from the epicenter to all nodes on the grid.

There are several situations in which these predicted ETAs may not match observed arrival times of the tsunami waves, including but not limited to the following:

1. The bathymetry grid is not accurate.
2. The epicenter is not well located, or the origin time is uncertain.
3. The epicenter is on land and a pseudo-epicenter off the coast must be selected.
4. The point approximation to the epicenter inadequately represents the rupture zone.
5. Nonlinear propagation effects may be important in shallow water.
6. The observed travel times represent later arrivals.

It is therefore of interest to examine historical tsunamigenic events in the Pacific and Indian Oceans and compare observed travel times to predictions made with the methodology currently in place at many warning centers. Given such data one may derive statistical information about the accuracy of these rapidly calculated ETAs. In particular, I wish to examine the statistical properties of Δt, the discrepancy between observed and predicted travel times, and determine if there are significant systematic variations in Δt. For instance, given that earthquakes with epicenters on land can excite tsunamis, how does Δt vary with location of the pseudo-epicenter location chosen for such earthquakes? Finally, I will examine to what degree the various error sources listed above are

responsible for large Δt and what can be done to ensure the most accurate predictions in an emergency situation.

2. Methodology

I have examined the NGDC database of tsunamigenic earthquakes and associated observed first arrival tsunami travel times to numerous stations (NATIONAL GEOPHYSICAL DATA CENTER, 2007). From this database 127 tsunamigenic earthquakes were identified as having produced observable tsunamis with well-determined origin times and locations. In selecting this subset I examined definite tsunamis since 1800 with runup reports, an earthquake magnitude of 6 or above, and an epicenter in the Pacific or Indian Oceans (Fig. 1); the overwhelming majority of events are from the Pacific basin. Travel-time calculations relied on the global 2×2 arc minute bathymetry grid ETOPO2 (NATIONAL GEOPHYSICAL DATA CENTER, 2006) which itself derives most of its oceanic depths from the predicted/calibrated bathymetry based on satellite altimetry and shipboard bathymetry (SMITH and SANDWELL, 1994; 1997). To prevent excessive travel time overestimates in cases when the earthquake occurred beneath very shallow water (or for epicenters on land) I relocated the epicenter to the nearest node with a depth of

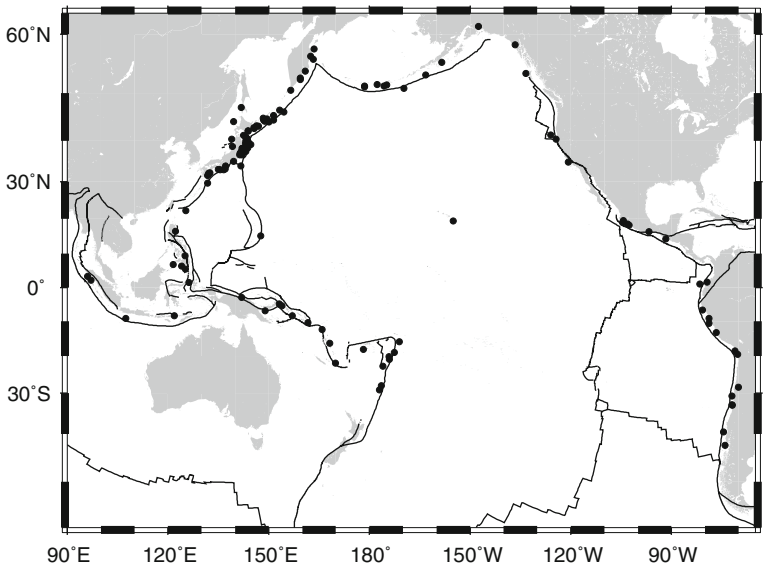


Figure 1
Location of 127 earthquakes identified as tsunamigenic events in the NOAA database. For each event I computed a global 2'×2' travel-time grid and sampled the travel times at all stations that reported an observed travel time, yielding over 1500 pairs of reported and predicted travel times.

at least 25 m. For consistency, and to examine far-field propagation of the most devastating tsunamis, global 2×2 arc minute travel-time grids were generated for each of the 127 events considered, even though only a few are known to have propagated beyond the Pacific (or Indian) basins. For warning center operations, typically only a regional (e.g., Pacific-wide or Indian-wide) calculation is required; at 2×2 arc minute resolution a Pacific-wide ETA grid is obtained within 1–2 minutes on a fast workstation; a slightly cruder 5×5 arc minute solution takes less than 10 seconds. Because most warning operations will automatically determine the epicenter and magnitude of an earthquake (or obtain this information from other agencies), the tsunami travel-time calculations may be launched automatically for earthquakes over a certain magnitude threshold and the resulting travel-time grid will be ready for analysis almost immediately. The output travel-time grids are compatible with the Generic Mapping Tools (e.g., WESSEL and SMITH, 1998), which were used extensively in this analysis, and are available upon request.

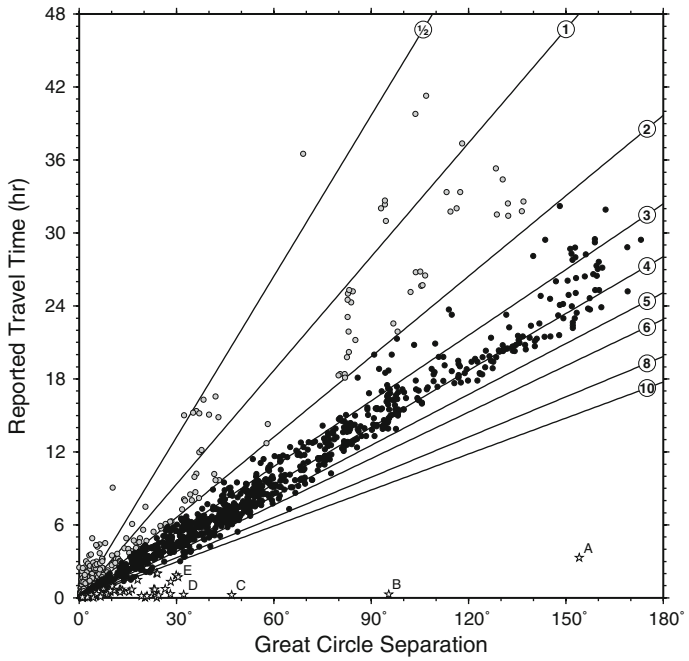


Figure 2

Reported travel times for the 127 earthquakes considered in the study, plotted versus the great circle distance between epicenter and recording station. Straight lines show travel times predicted by Eq. (1) for various average water depths (see labels, in km). Reported travel times above the 2 km-depth prediction (gray circles) are possibly late arrivals whereas times below the 11 km depth prediction may largely reflect erroneous tabulations (open stars), especially for the longer distances.

3. Analysis

3.1. Consistency of Reported Tsunami Travel Times

Prior to analyzing predictions I examined reported travel times versus the great circle separation between epicenter and reporting station; this distance represents the minimum path length traveled by any tsunami wave. Given Eq. (1) one can predict this relationship for a constant water depth. Figure 2 reveals several outliers that clearly indicate problems with the reported data. For instance, several reported travel times are much too short given the minimum distance the waves must have traveled. The outlier labeled "A" is the reported travel time from a 1922 earthquake in northern Chile to Aburatsubo, Japan. The distance is thus correct but one would expect a travel time closer to one day instead of the reported 198 minutes (3.3 hours). Perhaps the observed travel time originally was 19.8 hours (which is still too fast) but somehow ended up in the NOAA archive as 198 minutes. Outlier "B" from 2006 is more humorous, as the reasonable travel time from an Indonesian tsunami to Christmas Island (Indian Ocean) became associated with the other Christmas Island located in the Pacific, thus being archived with wrong metadata. Outlier C is simply a seiche registered in Freeport, Texas that was excited by the seismic tremors of the momentous 1964 Good Friday earthquake; thus, the travel time does not represent a typical tsunami phenomenon. Outlier D reflects another clerical error where the travel time from a 2006 earthquake in the Kuril Islands reportedly only took 16 minutes to reach the Shumagin Islands, Alaska over 3500 km to the west. Finally, outlier E is another Japanese recording (from Tsurushima) following a 1923 earthquake in Kamchatka. Again, I suspect the reported 10.2 minutes might originally have been 10.2 hours, and that many of the remaining outliers are likely to have similarly trivial explanations.

These data are further analyzed in Figure 3, which displays the equivalent average water depths, z_{ave}, required to reconcile reported travel times and their minimum distance of travel (via Eq. 1). All in all, 61 reports gave z_{ave} exceeding 11,022 m, which is the oceans' largest observed depth. These 61 are clearly all outliers and will be excluded from further consideration. Obviously, many others with slightly smaller z_{ave} are likely to be outliers as well but I have no clear cut-off criteria to apply and the distribution appears fairly continuous (see Fig. 3). Figure 2 also shows (as gray circles) reported travel times that appear too slow (equivalent average depth < 2 km). Certainly, for the more distant events these excessive travel times most likely reflect later arrivals, implying the first wave simply was too small to be noticed. Figure 3 suggests a possible hachured region where observations most likely come from later arrivals; again, no clear-cut criterion is available to separate these from first-wave arrivals and I will retain the remaining 1476 data pairs in the subsequent analysis.

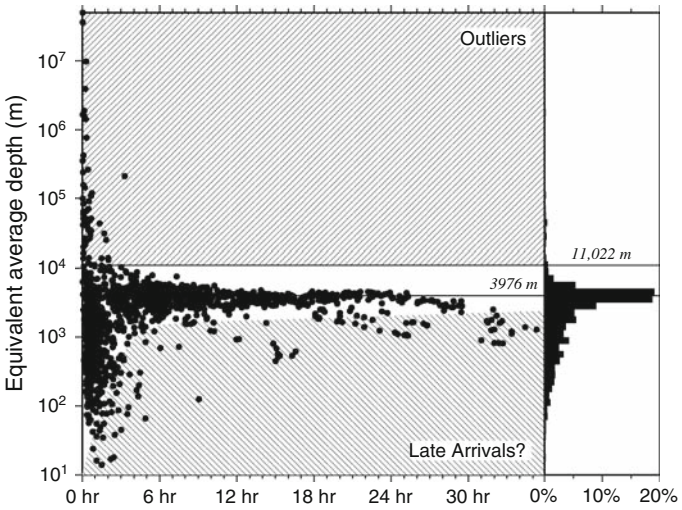


Figure 3

Equivalent average water depth (z_{ave}) versus reported travel times for data points in Figure 2. Most reported values are compatible with the Pacific mean ocean floor depth (3976 m). A total of 61 observations have a z_{ave} that exceeds the largest depth on Earth (11,022 m); these are considered clear outliers and are most likely clerical errors. Many values correspond to very shallow depths, probably reflecting overestimates of actual travel times (e.g., the detection of later rather than first arrivals). The lower hachured regions suggest an envelope for such later arrivals.

3.2. Simply Predicted Tsunami Travel Times

For each of the 127 events I calculated predicted travel-times on a global 2×2 minute grid, from which I made a detailed travel-time contour map, showing not only the (global) travel-time predictions but displaying the locations of stations from which reported travel times are available. These maps also include a simple graph of predicted versus observed travel time for these stations, and summarize the differences, Δt, between these pairs of values in standard box-and-whisker diagrams. In this paper I will only highlight some of these events individually; high-resolution PDF versions of all 127 event maps are available from the author's website (http://www.soest.hawaii.edu/pwessel/ttt). Figure 4 shows the results for the propagation of the tsunami wave front following the large 1960 Chile earthquake; here limited to the Pacific region only. The travel times are color-coded, with shading reflecting the shape of the underlying bathymetry. Over 100 tide stations, all in the Pacific, registered the arrival of this tsunami that took numerous lives, particularly in Hawaii and Japan, in addition to the local devastation in Chile (e.g., DUDLEY, 1998). None of the reported values have equivalent average depths exceeding 11,022 m. A direct comparison of observed and predicted travel times gives a correlation of 0.98, with a median Δt of only 14 minutes. However, note the several outlying points in the travel-time graph (Fig. 5). A closer inspection

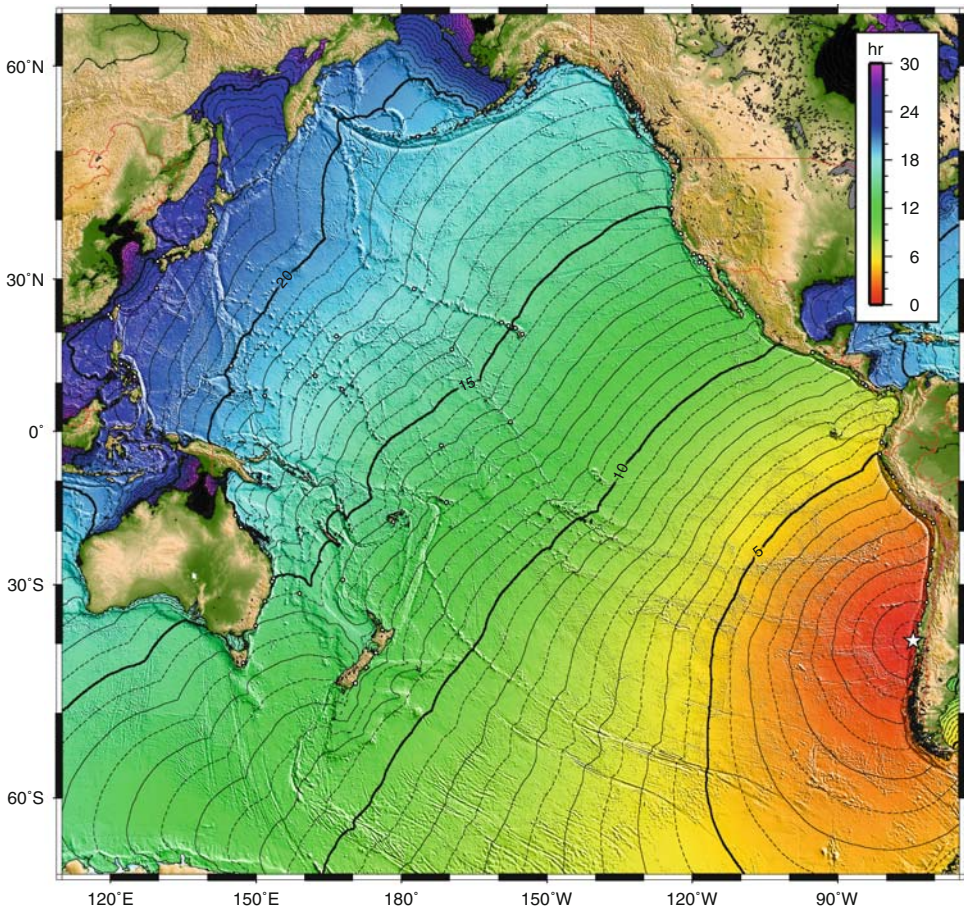


Figure 4
Predicted travel times for the Pacific-wide tsunami produced by the large 1960 Chile earthquake. Star indicates the point-source epicenter used for the calculation, with the 104 stations that reported travel times shown as white circles. Shading of travel times is provided by the bathymetry. Solid contours are hourly with 30-minute dashed contours in between.

shows that the single point for which the prediction exceeds the observed by several hours represents Punta Arenas in the far south of Chile. Given its sheltered position in the Strait of Magellan behind the Chilean Archipelago, the predicted travel time has been overestimated; it is likely that in this situation the simplicity of Eq. (1) poorly approximates the physics of wave propagation. Fortunately, the same island obstructions that lead to the excess in predicted travel time are also likely to attenuate truly dangerous waves before they arrive in Punta Arenas.

While the 1:1 correlation line is a remarkably good lower bound for all remaining observed travel times, there are several observations that are many hours slower than the

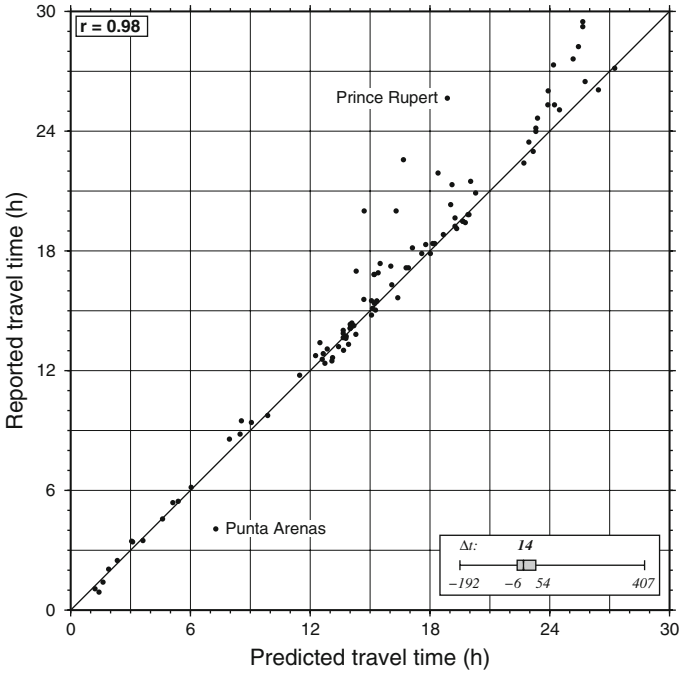


Figure 5

Reported versus predicted tsunami travel times for all 104 stations that observed the 1960 Chile tsunami. The 1:1 line represents perfect correlation. Large departures from this trend appear to be caused by excessive predicted travel times in shallow, narrow fiords (e.g., Punta Arenas, −3 hours) or a failure to detect the first arrivals (e.g., Prince Rupert, +6 hours). The box-and-whisker diagram summarizes the statistics of Δt; the differences between reported and predicted travel time (in minutes). The median Δt is +14 minutes.

corresponding predictions. Examining these points reveals that the slower observations for predictions in the 14–20 hour range are mostly associated with stations on the U.S. and Canadian west coast, many of which are sheltered in narrow inlets and sounds. Similarly, the slow arrivals after the first full day of propagation are mostly stations on the west-facing sides of Japan, Taiwan, and the Philippines. It would seem that these outliers represent later arrivals in locations where the first wave was not particularly energetic.

3.3. Statistical Analysis of Travel Times

Figure 6 shows all observed tsunami travel times plotted against the corresponding predicted travel times; the 61 points with excessive z_{ave} have been excluded. Again, if travel-time observations and predictions both faithfully reflected reality then all points should fall on a straight line with slope 1:1; clearly, this is not the case. However, as in the case of the 1960 event we do find a strong tendency for points to cluster around this

line, however there is significant scatter, some systematic offsets, and some large outliers. A peculiar feature of this plot is the appearance of a secondary trend that parallels the main 1:1 line but shifted by almost three hours of excess predicted travel time. During the analysis it became clear that the 1964 Prince Williams Sound, Alaska earthquake posed a particular problem when comparing predictions to observed travel times. Even a casual inspection of the travel-time correlation chart (white circles in Fig. 6) reveals that the predicted travel times are all close to three hours too long. This consistency for all observations points to a problem originating in the area near the epicenter. Early studies have demonstrated, by backward propagation of travel times, that the tsunami source area had to be located further out on the continental shelf, far from the epicenter (e.g., HATORI, 1981; PARARAS-CARAYANIS, 1967). Figure 7a presents a Mercator map of the Gulf of Alaska and indicates the reported location of the epicenter (star). As reported, the epicenter falls on land (e.g., SHERBURNE *et al.*, 1969) and therefore was relocated to the nearest ocean node of at least 25 m depth. The bathymetry in and near the area is

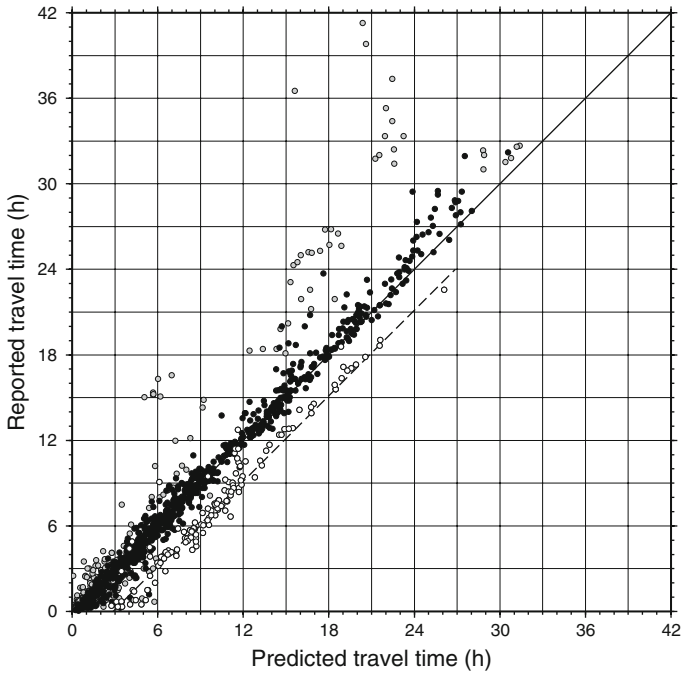


Figure 6
Correlation plot of all 1476 pairs of reported and predicted tsunami travel times. Color-coding as in Figure 2; outliers (stars) have been excluded. Other extreme outliers are noted, both above and below the trend line. The cluster of points (white) sub-parallel to the trend line ensues from reports of the great 1964 Prince Williams Sound earthquake in Alaska (see text).

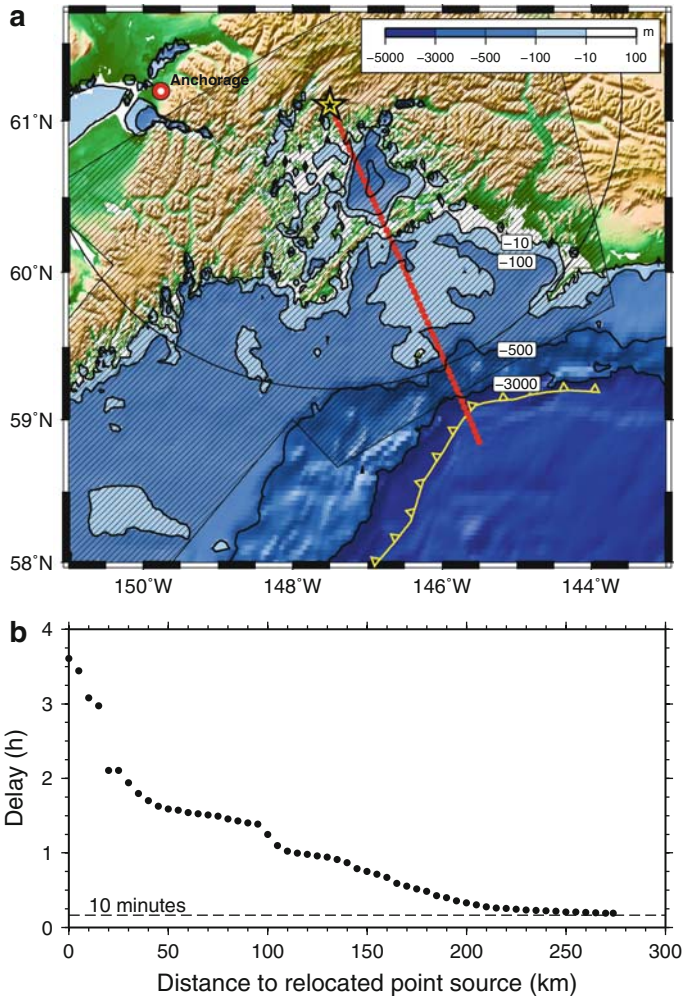


Figure 7

a) Coastlines and bathymetry near the site of the 1964 Alaska earthquake (star). Bathymetry shows extended shallow water depths on the continental margin. Hachured areas are uplifted blocks determined by joint geodetic and tsunami inversion by JOHNSON *et al.* (1996). Red dots indicate alternative point source locations for improved travel-time calculations, up to 275 km from the epicenter and toward the trench. **b**) Average delay (predicted minus reported tsunami travel time) obtained by using different point source locations. The major delays are caused by low propagation speed in shallow waters and the fact that the tsunami generation took place closer to the continental edge.

particularly shallow, which adds considerable propagation time to all stations. However, the main cause of the delays lies in the nature of the tsunami generation. Studies have shown that a large region of the continental shelf experienced significant crustal uplift in response to the earthquake (e.g., CHRISTENSEN and BECK, 1994; JOHNSON *et al.*, 1996; RUFF

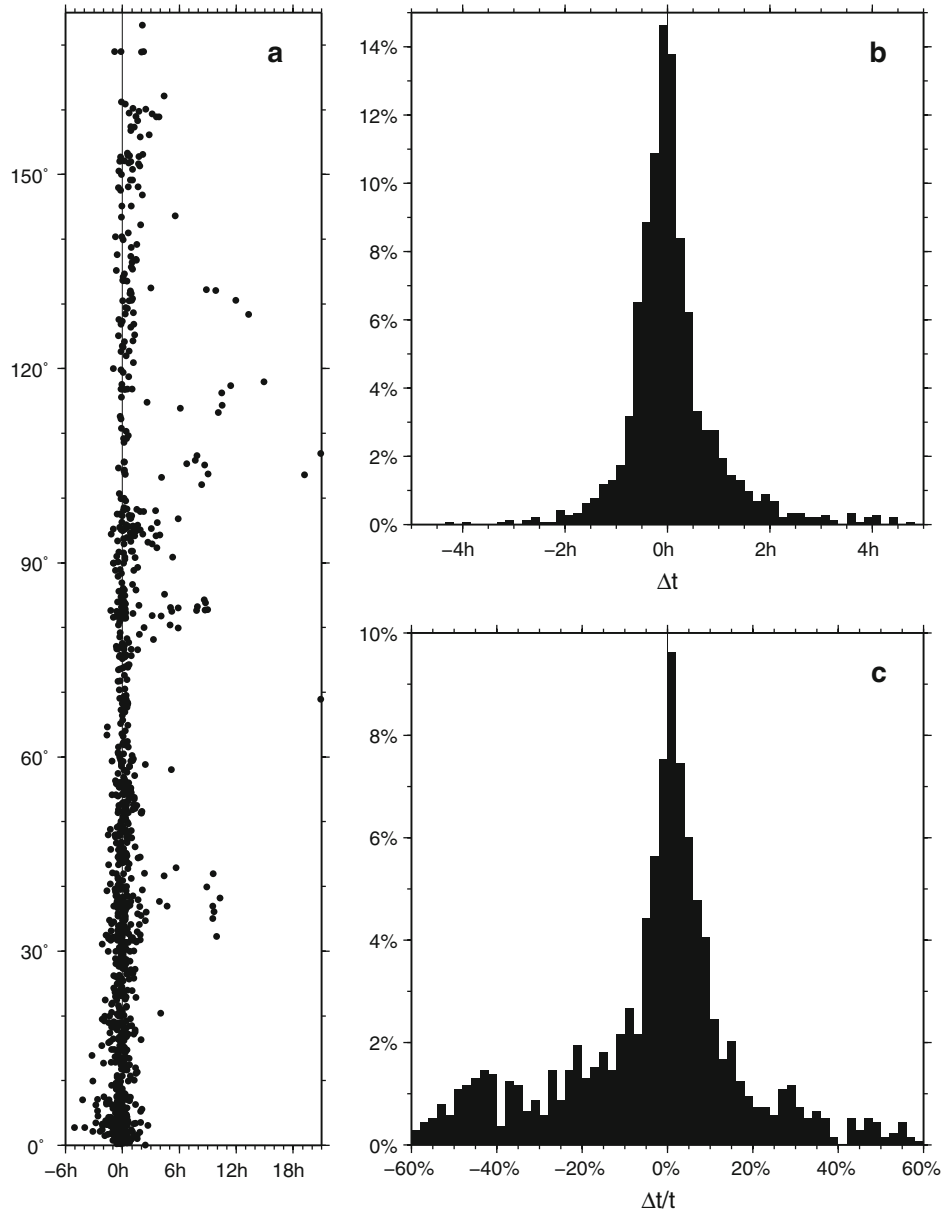


Figure 8

(a) Summary of Δt, the difference between reported and predicted tsunami travel times for all 1537 data pairs versus minimum travel distance (in degrees). (b) Histogram of Δt. The mean (median) value is 12 (−1.5) minutes with standard deviation (median absolute deviation) of 139 (35) minutes. Distribution has a longer tail to the right. (c) Same, but normalized by predicted travel time and reported in percent.

and KANAMORI, 1983), and it is this wide uplift of the water on the continental shelf that initiated the tsunami. In other words, a point source approximation turns out to be particularly poor for this event; however, this realization is in general not achieved until some time after the event.

To test this explanation I relocated the point source to increasingly more distant locations along a great circle from the epicenter to the nearest point on the trench

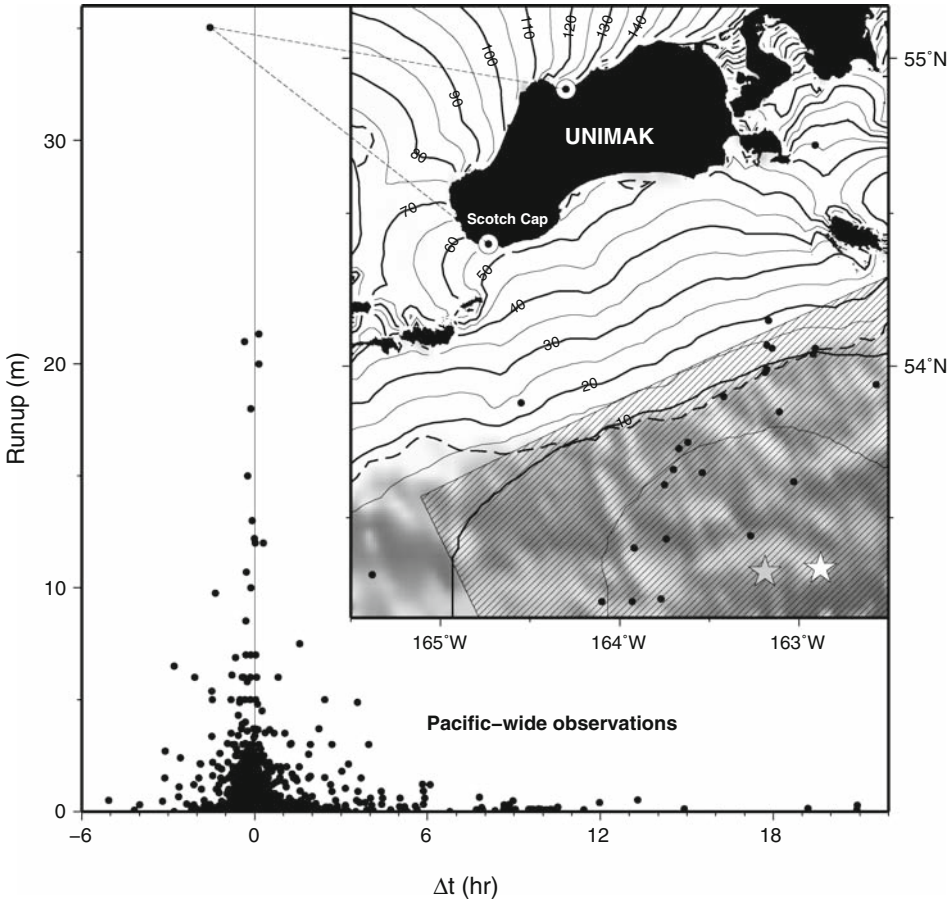


Figure 9

Distribution of tsunami runups (m) versus Δt. The larger runups have small Δt, suggesting larger Δt values may involve later arrivals. Note the large runup for the 1946 tsunami and its substantial travel-time prediction delay of $\sim$1.5 hrs. Inset: Travel-time contours (in minutes) from epicenter (star) of 1946 Aleutian tsunami on shaded bathymetry. Gray star is epicenter reported in NOAA runup catalog. White star, solid circles, and hachured region are relocated epicenter, aftershocks, and best estimate of minimum rupture area, respectively (LOPEZ and OKAL, 2006). Dashed contour is 200-m isobath. The marked delay Δt for the largest runup reflects incorrect coordinates used for Scotch Cap (see text).

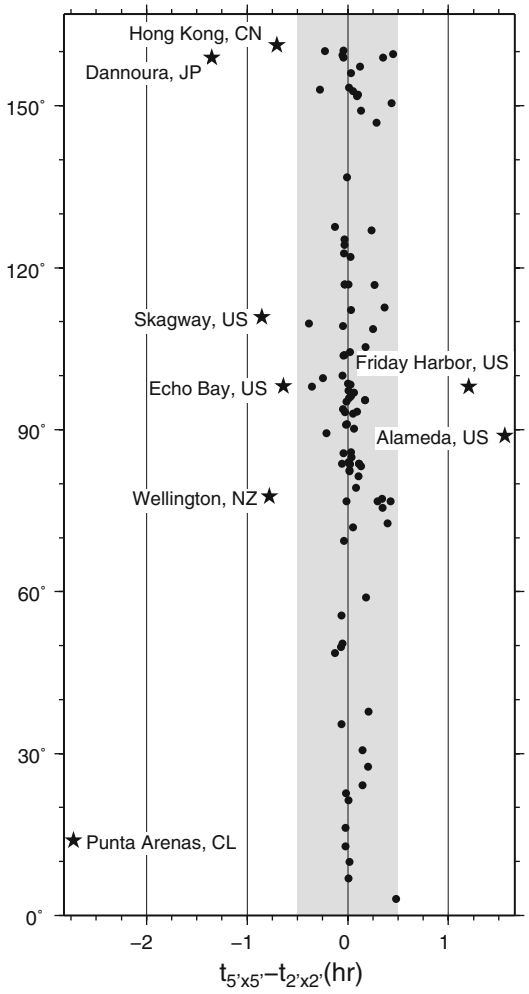
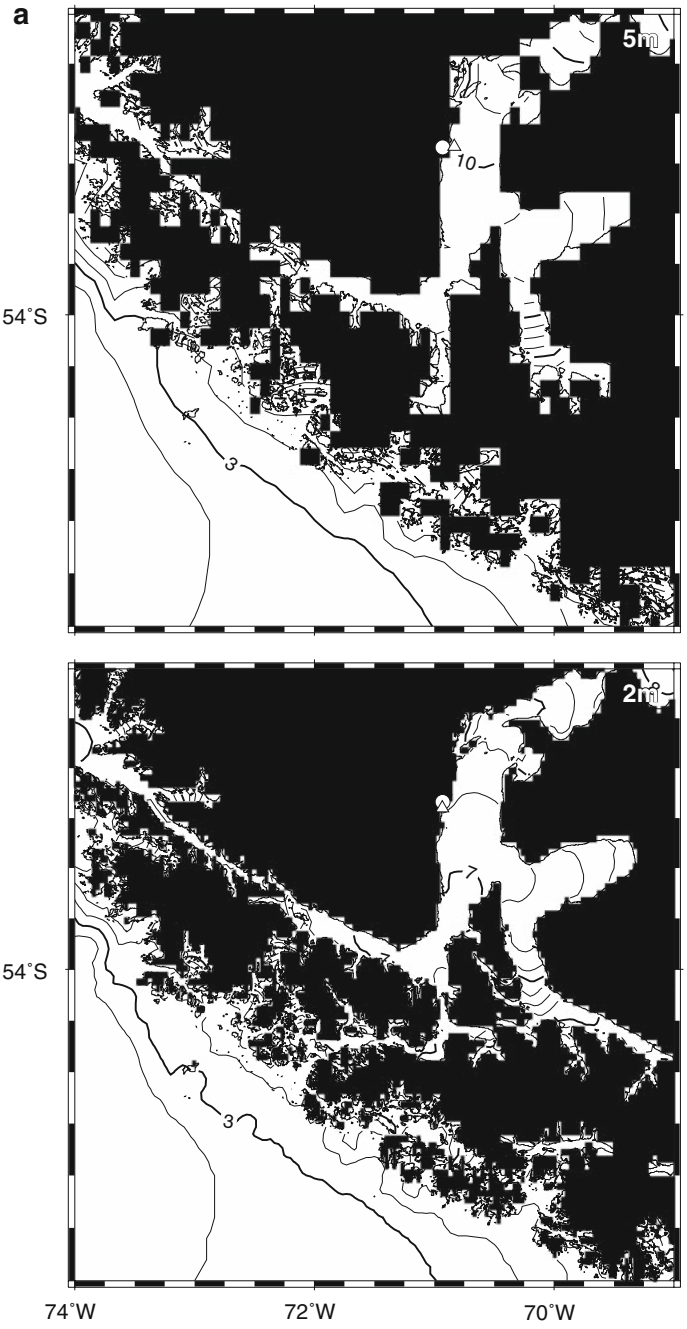


Figure 10
Sensitivity of predicted travel time due to bathymetry grid resolution. Differences in travel times (solid dots) calculated from 5×5 and 2×2 arc minute grids are shown at all stations reporting arrivals for the 1960 Chile event. I thoroughly investigated the causes of the largest discrepancies (named stations; solid stars) which all were related to geometry changes for shallow water pathways near the station (see text).

(Fig. 7a). I then ran the travel-time calculations on the 2×2 arc minute grid for the different point sources. The various travel-time delays were found by computing the mean Δt for each solution. Figure 7b shows the prediction delays versus the distance between reported epicenter and point source used. The delay is gradually reduced with distance and appears to approach asymptotically a ∼10-minute level (for distances > 250 km). This distance corresponds to the outer boundary of the uplifted


a
5m
10
3
2m
7
3
54˚S
54˚S
74˚W
72˚W
70˚W

◀ Figure 11
Comparison of bathymetry grid and predicted travel times for the 2×2 min (lower) and 5×5 min (upper) grids for the 1960 Chile event. Open circle is reported station location whereas open triangle is nearest node located in the water. (a) Punta Arenas, Chile, is located in the Strait of Magellan sheltered by the Chilean Archipelago. The different bathymetry resolutions result in different pathways and a shallower average depth. (b) Alameda, California, US in the San Francisco Bay. In the courser 5-min-grid the bay entrance is closed off, forcing the station to be relocated all the way to the Pacific coast and shortening the predicted time. (c) Dannoura, Japan is similar to Alameda, as the relocated station falls on the Japan Sea coast instead of in the Seto Inland Sea to the east of the artificial barrier blocking the Kanmon Strait.

blocks (hachured areas in Fig. 7a) determined from a joint inversion of geodetic data and tsunami waveforms (JOHNSON *et al.*, 1996). The remaining $\sim$10 minute delay most likely reflects the non-point-source nature of the disturbance as well as other causes such as inaccurate bathymetry at reporting stations and inability to identify the arrival of the first wave.

Figure 8 presents a summary and histograms of Δt implied by the data in Figure 6 and augmented by the data for the 1964 tsunami after correcting for the inferred 2.92-hour bias. Figure 8a shows how Δt varies with epicenter-station separation. We clearly see late arrivals (positive Δt) increase for tsunami waves that traveled long distances, while prediction errors ($\Delta t < 0$) are most prevalent for stations not too distant from the tsunami nucleation area. Two different forms of analysis were pursued: (1) Figure 8b gives the standard histogram of the Δt distribution in terms of departures from the predicted value; (2) Figure 8c shows the same departure as a percentage relative to the predicted travel time. This approach was undertaken to show how the misfit varied with travel time. We note that the former quantity appears more normally distributed than the latter, nonetheless both have long tails, suggesting nonparametric statistics should be used to characterize the distributions. Whereas the mean and standard deviation of Δt are 19 and 131 minutes, respectively, the median and the median absolute deviation (MAD) are only -0.3 and 33 minutes, respectively. Clearly, the presence of late arrivals skews the mean away from the expected zero, which is well represented by the median. The percentages also are vulnerable to large scatter due to the normalization by small travel times; I find a median percentage of -0.2 and a MAD of 15%. These robust values represent typical uncertainties and exclude the few extreme cases.

3.4. Runup and Predicted Travel Times

One of many concerns for agencies responsible for issuing warnings is the possibility of overestimating travel times to some stations, such as would have been the case if the 1964 Alaska tsunami travel-time predictions were to be taken at face value. In comparing reported runups to both predicted and reported travel times I note: (1) The largest runups are associated with stations very close to the epicenter. For people in proximal regions of large earthquakes the best defense is to leave the coastal region and seek safety inland while there is still time. (2) Runups at stations with poor correlation between reported

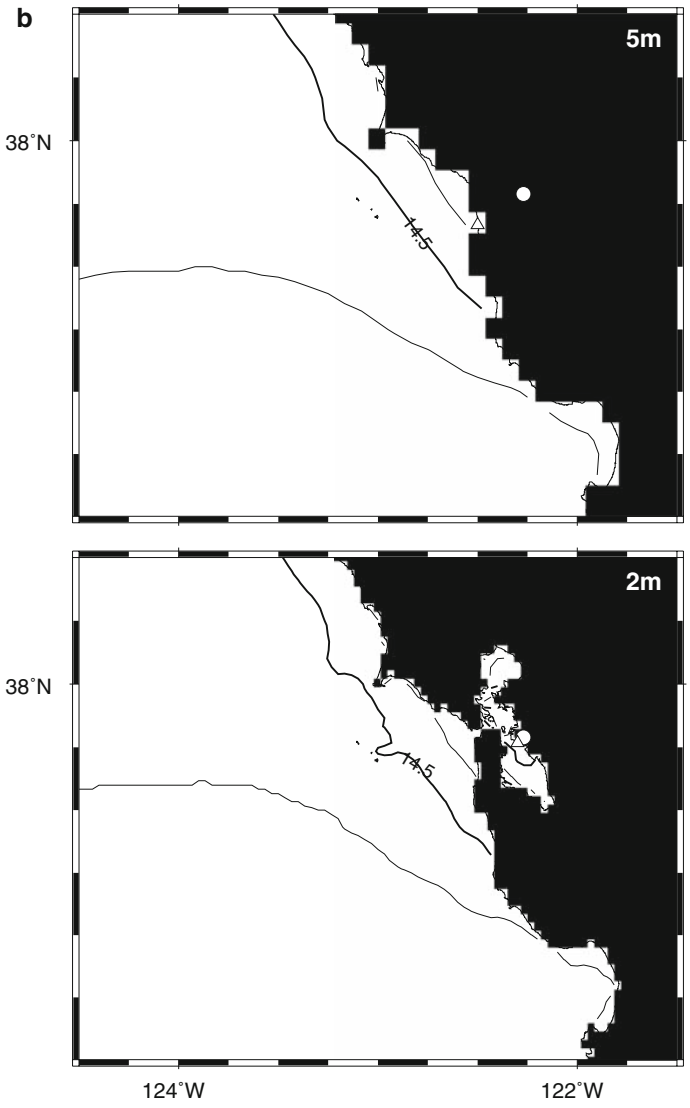


Figure 11
Contd.

and predicted travel times are insignificant. This is highlighted in Figure 9, which displays runups for all 1476 records; it is clear that, in general, the largest values have very small Δt. However, we note that the largest runup (> 35 m) has a disturbingly large prediction delay of 1.6 hours (as do some other runups in the 5–10 m range). This particular observation comes from Scotch Cap on Unimak Island, Alaska following the April 1, 1946 tsunami that originated on the slope to the south of Unimak Island. This

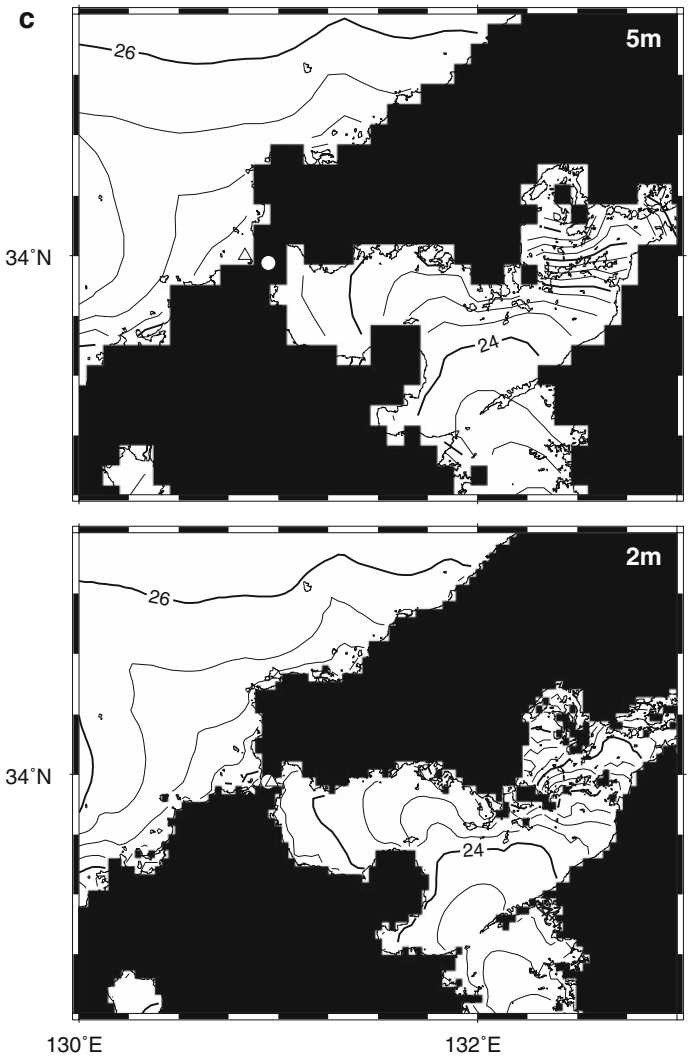


Figure 11
Contd.

tsunami is unusual in that it is generally assigned a relatively small magnitude (e.g., PACHECO and SYKES, 1992), yet its tsunami magnitude is 9.3 (ABE, 1979) and it produced very large runups focused in a narrow beam normal to the strike of the trench (FRYER *et al.*, 2004). A recent revision to the Scotch Cap runup even raises the value to 42 m (OKAL *et al.*, 2003), and a reanalysis of long-period seismographs suggests the magnitude was probably closer to 8.5 (LOPEZ and OKAL, 2006). Several studies have determined

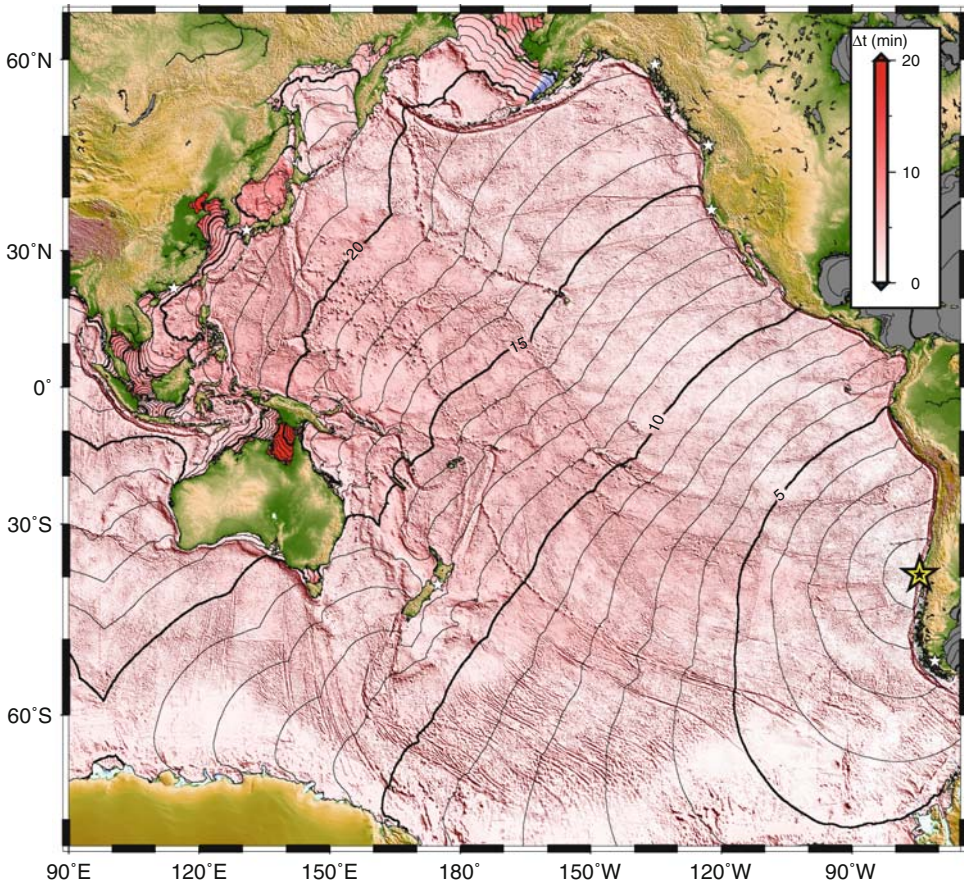


Figure 12
Color-coded differences between travel-time predictions for 2×2 and 5×5 arc minute bathymetry, with superimposed hourly travel-time- contours. Yellow star is epicenter location. The largest differences occur in shallow coastal areas such as between Australia/Papua New Guinea and the Yellow Sea between China and Korea. For > 97% of the Pacific nodes the difference in predicted travel time is less than 5 minutes. White stars denote locations of 8 outliers in Figure 10.

approximate fault plane solutions from the distribution of aftershocks (e.g., JOHNSON and SATAKE, 1997; LOPEZ and OKAL, 2006); hence the point source epicenter solution employed herein to obtain travel times may likely be inadequate in this case as well. However, from the map inset we can determine the main cause of the large Δt: While the NOAA runup data base correctly reports the Scotch Cap observed travel time (48 minutes) and runup, it incorrectly lists as location the coordinates of a point on the north side of Unimak Island, near Cape Mordvinof. Using the Scotch Cap coordinates yields a revised predicted travel time of 53 minutes and an improved Δt of only 5 minutes.

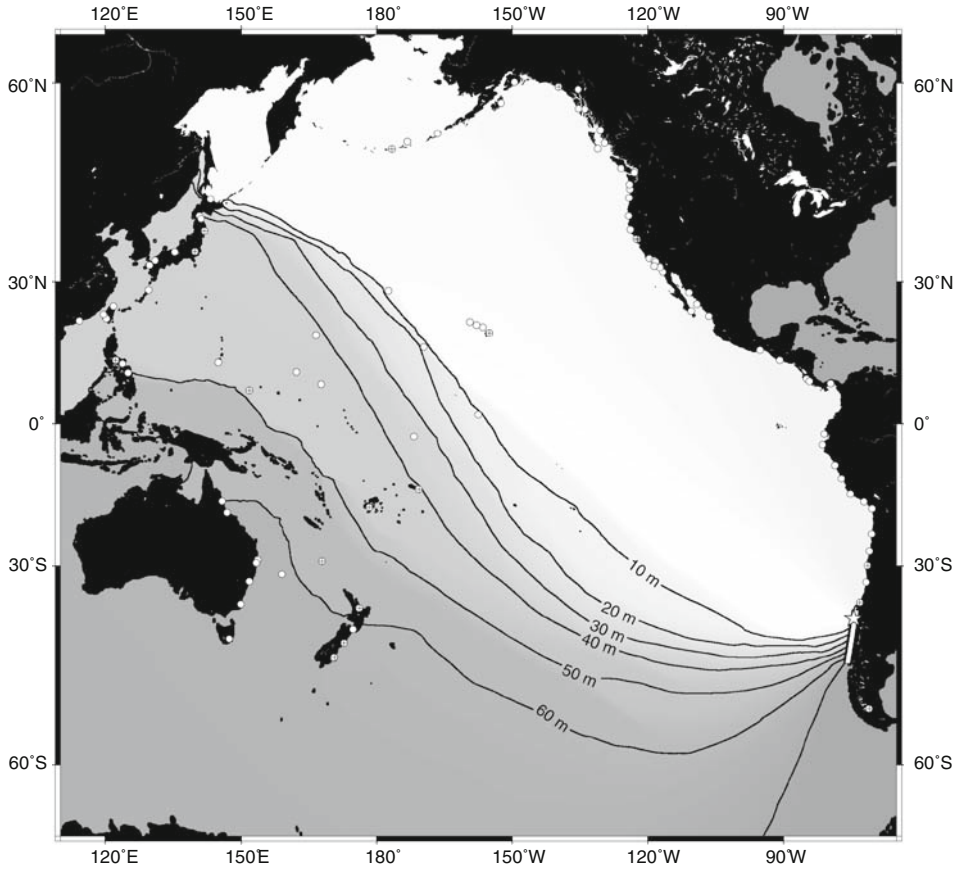


Figure 13
Differences in travel-time predictions for the 1960 Chile tsunami using a close approximation to the actual rupture area (white rectangle) versus the point-source epicenter solution (star). Contours (in minutes) project the effect being limited to the southern and western Pacific (shaded areas). Circles represent recording stations; crossed circles have predicted travel times that are slower than observed by 10 minutes or more.

3.5. Effect of Using Coarser Bathymetry Grids

Regardless of the method used to calculate travel times, any uncertainties in water depth will translate into uncertainties in the predictions. Equation (1) suggests uncertainties in depth are more critical for areas of shallow water where an underestimated depth can give rise to significant travel-time delays. One source of depth uncertainty comes from the preparation of gridded bathymetry. Given the 104 stations that reported observed travel times for the 1960 Chile tsunami, I repeated the travel-time calculation using a coarser, 5×5 arc minute global grid derived by filtering ETOPO2 with a 17-km median filter to avoid aliasing and to reduce the influence of

narrow, shallow features in the derived grid. I then computed the difference in predicted travel-time to the 104 stations and plotted these differences versus the corresponding epicenter-to-station great circle distances (Fig. 10). While the differences have $\sim$ zero mean we find a handful of significant outliers as well as a general variability with standard deviation of $\sim$5 minutes. To determine the source of the larger outliers (stars) I thoroughly investigated each of the bathymetry grids near the eight named stations; here I discuss three representative examples that highlight the typical causes of such discrepancies. The remaining examples have similar albeit less severe characteristics.

Figure 11a shows travel-time grids for the 5×5 minute (upper) and 2×2 minute (lower) bathymetry grids. Black indicates nodes on land. The circle indicates the location of Punta Arenas in southern Chile where the two travel-time grids differ by almost 2 hours. Examination of the nodes quickly reveals differences between the two grid resolutions. While the finer 2×2 minute grid is able to preserve some of the narrow waterways between the numerous islands in the Chilean Archipelago and the Strait of Magellan, the 5×5 minute grid has closed off many of these pathways, resulting in two significantly different paths from epicenter to station. In this extreme case the 5×5 minute grid prediction would be almost 10 hours. Neither solutions come particularly close to the reported travel time of $\sim$4 hours (7.3 vs. 10 hours).

Figure 11b shows the opposite situation occurring at the Alameda tide station inside San Francisco Bay, USA. Here, the narrow inlet (spanned by the Golden Gate Bridge) could not be represented in the coarser grid, resulting in the entire bay being land-locked. In such cases, the travel-time grid must be sampled at the ocean node nearest the tide station (triangle), which in this case is relocated to the Pacific coast. In comparison, the finer grid allows for propagation into the shallow bay, thus resulting in almost one hour longer travel time and a much better fit to the observed travel time.

Finally, Figure 11c displays the travel-time grids near the Japanese station Dannoura (circle). Again, the coarser grid is unable to represent the narrow Kanmon Strait connecting the Japan Sea to the Seto Inland Sea, and when the nearest node in the ocean is selected it falls on the western rather than eastern side of the artificial barrier. Hence, as the waves must propagate around Kyushu to reach the station, we find a delay of almost 2 hours relative to the 2×2-minute grid prediction. Interestingly, the reported travel time to Dannoura is 29.5 hours, which is about 4 hours longer than the 2×2-minute prediction. Based on Figure 2 it would appear that the reported arrival time corresponds to a later arrival; the bathymetry near the station is not as complicated as in the case of Punta Arenas, and hence the 4-hour difference is unlikely to reflect a prediction error.

By resampling the 5×5 minute travel time prediction grid onto a 2-minute grid we may compute the predicted differences for the entire Pacific (Fig. 12). It is noteworthy that the two grids differ by less than 5 minutes at $\sim$97% of the grid nodes. The only significant deviations visible at this scale are differences in the 20–60 minute range for shallow areas between Australia and Papua New Guinea and in the Yellow Sea. The extreme cases in Figure 10 (white stars in Fig. 12) are not typical nodes in this regard, yet

many tide stations are obviously located in shallow water near land and hence are affected locally by the grid resolution effect. We can also see a subtle delay effect due to the denser seamount populations in the Western Pacific. While the 5×5 minute predictions in general are slightly slower than the 2×2 minute predictions, a few areas show the opposite effect (e.g., off Alaska), again reflecting the difference in pathways when narrow waterways are not adequately represented in one of the grids.

4. Discussion

This investigation has found several characteristics of observed and predicted travel times and their statistical distribution and depth-dependency that may be of interest to both warning centers and tsunami researchers. However, given the simple approach used to predict travel times, the observations made in this study are more germane to the near-real-time response to a tsunami in progress when quick and accurate estimates of travel times are required. It is reassuring that the simple predictions based on Eq. (1) and the standard 2×2 minute bathymetric grid are quite consistent with reported travel times for the 127 tsunamis studied here (e.g., Fig. 6). However, there are clear departures from the expected 1:1 correlation and these have been examined in some detail. I have demonstrated that predictions in some cases have considerable delays and determined three main causes for these delays: (1) The inability of the epicenter point source to adequately represent the actual water impact that generated the tsunami, (2) occasional large changes in propagation geometry due to the finite spatial resolution of the global grids, and (3) uncertainties in depth for shallow water regions. Given Eq. (1), all significant bathymetric bias will occur in shallow waters. For any event, these areas are most likely to include the immediate regions surrounding both epicenter and observation points (or warning points). Warning agencies and tsunami researchers should therefore strive to acquire the best available local bathymetric data in all regions that fall in this category.

To exemplify the bias that may result from using a point source (i.e., the epicenter) for tsunami travel-time evaluation I contrast the predictions from the 1960 Chile tsunami using two different sources: (1) The reference calculation uses the reported epicenter (Fig. 4) which is what warning centers must use in a real-time warning situation, and (2) the rupture zone identified by PLAFKER and SAVAGE (1970). The latter source region extends over 1000 km southward from the epicenter and hence prediction of travel times south of the epicenter can be expected to differ. Figure 13 shows the difference in predicted travel time (in minutes) between the point- and line-source calculations. As anticipated the largest discrepancies are found to the south of a line from the epicenter to Japan, i.e., the shaded region. In particular, at stations in New Zealand and Australia the difference in predicted travel time is almost 1 hour. Stations that reported an observed travel time shorter than the reference prediction are shown with a crossed circle; the majority of such stations fall in the affected region. Of course, slow propagation in

shallow waters near some stations and failure to detect first arrivals may have obscured the predicted trend to some extent.

In a warning situation the emphasis lies on preparing as accurate estimated times of arrival as possible. It is therefore of great concern that certain combinations of epicenter locations and station placements, when used with Eq. (1) and standard global bathymetric grids, yield travel times that are unacceptably delayed. Should such delayed predictions be presented as accurate they may cause considerable damage directly (by giving wrong information) and indirectly (by reducing the confidence the community has in warning centers). However, large tsunamigenic earthquakes do not occur daily, hence there is ample time between events to lay the groundwork required to avoid such overestimates. Given the rapidity with which travel-time estimates can be obtained, warning centers may explore the effects that epicenter and station location and bathymetry grid quality have on the predicted values. For instance, the finite number of tide stations and warning points could be explored in detail (such as was done in Fig. 11) to determine if the coordinates of the station should be adjusted to avoid particularly shallow areas and if important, but narrow water ways are well represented in the grid. Special processing of the bathymetry may be required to reduce delays and optimize travel-time predictions. Similarly, precalculations of tsunamis from anywhere along the ring of fire could be examined and used to identify regions where point source solutions may be particularly susceptible to error (such as along wide continental margins, e.g., Fig. 7). Since such numerical experiments are not subject to the time-constraints of an emergency response, higher resolution grids (requiring longer calculation times) may be employed in order to map the sensitivity of the predictions to the grid spacing used during emergency operations. Finally, assessment of travel times from model-based forecast systems may be used to address the uncertainties of point-source based solutions (e.g., GREENSLADE and TITOV, 2008). The goal of such efforts would be to enable warning centers to calculate and report reasonable error bounds on any estimated time of arrival released to the public.

5. *Conclusions*

1. Simple long-wave predictions of tsunami travel times calculated from a global grid of bathymetry yields approximate results that correlate highly with $\sim$1500 reported travel times from 127 separate events.
2. Large outliers exist on both sides of the expected trend. Observation times that greatly exceed the simple predictions are most likely later arrivals. In cases when predictions greatly exceed observation times we find that either the reports had clerical errors or there were peculiar circumstances with respect to the geometry of the pathways and their depths near a particular station. Because most stations are located next to land, these conditions do occur in enough places to warrant concern.

3. The largest significant causes of uncertainty for predicted travel-times are the inadequate approximation of the tsunami source by the epicenter point source and the poor characterization of shallow bathymetry near stations and some epicenters. Depending on circumstances, the travel-time delays from these errors sources can be significant (i.e., hours).
4. Numerical simulations of hypothetical tsunamis from any point along subduction zones can be performed and used to delineate areas from which the simple travel-time solutions may be inadequate. Likewise, the examination of the variability of travel times near all stations of points of interest can be used to map which regions need special consideration in a warning situation and to guide special processing of bathymetry to ensure the proper representation of key waterways near stations.

Acknowledgments

Diana Greenslade, Gerard Fryer, editor Kenji Satake and an anonymous reviewer made numerous suggestions that lead to improvements in the text. This is SOEST contribution no. 7606.

References

Abe, K. (1979), *Size of great earthquakes of 1837–1974 inferred from tsunami data*, J. Geophys. Res. *84*, 1561–1568.

Christensen, D. H. and Beck, S. L. (1994), *The rupture process and tectonic implications of the great 1964 Prince William Sound earthquake*, Pure Appl. Geophys. *142*, 29–53.

Dudley, W., T*sunami*!, 2nd ed., 380 pp. (Univ. Hawaii Press, Honolulu, 1998).

Fryer, G. J. *et al*. (2004), *Source of the great tsunami of 1 April 1946: A landslide in the upper Aleutian forearc*, Marine Geology *203*(3–4), 201–218.

Geoware (2007), *TTT - A tsunami travel-time calculator*, http://www.geoware-online.com

Greenslade, D. J. M. and Titov, V. V. (2008), *A comparison study of two numerical tsunami forecasting systems*, Pure Appl. Geophys. *165*, 1991–2001.

Hatori, T. (1981) *Tsunami magnitude and source area of the Aleutian-Alaska tsunamis*, Bull. Earthquake. Res. Inst., Univ. Tokyo *56*, 97–110.

Johnson, J.M. *et al*. (1996) *The 1964 Prince William Sound earthquake: Joint inversion of tsunami and geodetic data*, J. Geophys. Res. *101*(B1):523–532.

Johnson, J. M. and Satake, K. (1997), *Estimation of seismic moment and slip distribution of the April 1, 1946, Aleutian tsunami earthquake*, J. Geophys. Res. *102*(B6), 11, 765–711,774.

Kowalik, Z. *et al*. (2005), *Numerical modeling of the global tsunami: Indonesian tsunami of 26 December 2004*, Science of Tsunami Hazards *23*(1), 40–56.

Lopez, A. M. and Okal, E. A. (2006), *A seismological reassessment of the source of the 1946 Aleutian 'tsunami' earthquake*, Geophys. J. Internat. *165*(3), 835–849.

Mader, C.L. (2004), *Numerical Modeling of Water Waves*, 2nd ed., 274 pp. CRC Press, Baton Rouge.

Mei, C. C. (1989), *The Applied Dynamics of Ocean Surface Waves*. (World Scientific, Singapore).

National Geophysical Data Center (2006), *2-minute gridded global relief data (ETOPO2v2)*, U.S. Department of Commerce, National Oceanic and Atmospheric Administration.

National Geophysical Data Center (2007) *NOAA/WDC historical tsunami database at NGDC*, edited, http://www.ngdc.noaa.gov/seg/hazard/tsu_db.shtml.

Okal, E. A. *et al*. (2003), *Near-field survey of the 1946 Aleutian tsunami on Unimak and Sanak Islands*, Bull. Seismol. Soc. Am. *93*(3), 1226–1234.

Pacheco, J. F. and Sykes, L. R. (1992), *Seismic moment catalog of large, shallow earthquakes, 1900–1989*, Bull. Seismol. Soc. Am. *82*, 1306–1349.

Pararas-Carayanis, G. (1967), *A study of the source mechanism of the Alaska earthquake and tsunami of March 27, 1964, I. Water waves*, Pac. Sci. *21*, 301–310.

Plafker, G. and Savage, J. C. (1970), *Mechanism of the Chilean Earthquake of May 21 and 22, 1960*, Geol. Soc. Am. Bull. *81*, 1001–1030.

Ruff, L. and Kanamori, H. (1983), *The rupture process and asperity distribution of three great earthquakes from long-period diffracted P-waves*, Phys. the Earth and Planet. Inter. *31*, 202–230.

Shepard, F. P. *et al*. (1950), *The tsunami of April 1, 946*, Bull. Scripps Inst. Oceanogr. Univ. Calif. *5*, 391–528.

Sherburne, R. W. *et al*. *The hypocenter, origin time, and magnitude of the Prince William Sound earthquake of March 28, 1964*. In *The Prince William Sound, Alaska Earthquake of 1964 and Aftershocks* (ed. L. E. Leipold), pp. 49–69 (US Dep. of Comm., Environ. Sci. Serv. Admin., Washington, D.C. 1969).

Smith, W. H. F. and Sandwell, D. T. (1994), *Bathymetric prediction from dense satellite altimetry and sparse shipboard bathymetry*, J. Geophys. Res. *99*(B11), 21,803–821,824.

Smith, W. H. F. and Sandwell, D. T. (1997), *Global seafloor topography from satellite altimetry and ship depth soundings*, Science *277*(5334), 1956–1962.

Wessel, P. and Smith, W. H. F. (1998), *New, improved version of Generic Mapping Tools released*, EOS Trans., AGU, *79* (47), 579.

(Received January 27, 2008, accepted May 10, 2008)
Published Online First: February 6, 2009