

Costas Arrays to Order 1030

IEEE DataPort Database

DOI: <http://dx.doi.org/10.21227/H21P42>

Instructions

James K Beard

ORCID [0000-0003-4430-8704](https://orcid.org/0000-0003-4430-8704)

Instructions for downloading and setting up a searchable database of Costas arrays to order 1030

Instructions for the use of the Costas array database analysis and extraction utility

Release 03, February 2019

Table of Contents

1	Files in the Database	5
1.1	Data Files	5
1.2	Database Utility Executable	6
1.3	Other files	7
1.4	Install Only What you Need	7
2	Installation of the Database	8
3	Installation and Use of the Costas Array Extraction and Analysis Utility	10
3.1	Version	10
3.2	Downloading and Installation	10
3.3	The Application-Maintained Configuration File	11
4	Using the Utility	12
4.1	Starting the Utility	12
4.2	The Opening Dialog	15
4.3	The Background Scrolling Text Window	15
4.4	The Selection and Analysis Modes	16
4.4.1	Program Configuration	17
4.4.1.1	Location of Database	17
4.4.1.2	Program Output Files Location	17
4.4.1.3	Clear CSV Files from Output Folder Before Run	17
4.4.1.4	Overwrite or Append Old Files	17
4.4.1.5	Generated Database Only Checkbox	18
4.4.1.6	Index from 0 to N-1 Checkbox	18
4.4.2	Filtering Parameters	21
4.4.2.1	Order of Costas Arrays (Selection Mode only)	21
4.4.2.2	Starting and Ending Orders of Costas Arrays (Analysis Mode only)	21
4.4.2.3	Generator Method Combo Box	21
4.4.2.4	Costas Array Database Type Combo Box	22
4.4.3	Selection by Bounding the Number of Ones in a User-Selected Rectangle of the DAF	24
4.4.3.1	No Filtering on Count of Ones in the DAF	24
4.4.3.2	Bound the Number of Ones in Central 3X3 (4 to not filter)	24
4.4.3.3	Bound the Number of Ones in Central 5X5 (16 to not filter)	25
4.4.3.4	Bound the Number of Ones in Arbitrary User-Selected Rectangle	25
4.4.3.5	Setting the Bound on the Count of Ones	25
4.4.3.6	Setting Row and Column Limits for DAF Region	25
4.4.4	Filtering by Maximum Frequency Hop Radio Buttons	25
4.5	Computation Options for All Database Access Modes	26

4.5.1	Provide DAF Checkbox.....	26
4.5.2	Provide Generator Polynomial in $GF(q)$ Checkbox	26
4.5.3	Order of Galois Field q	26
4.5.4	Computation Options Available in Analysis Mode Only	26
4.5.4.1	Write CSV File of Database File Sizes Checkbox.....	26
4.5.4.2	Output CSV of Counts of Ones in Central 3X3 Square Checkbox.....	27
4.5.4.3	Re-Compute Minimum Peak Frequency Hop Database Checkbox	27
4.5.4.4	Output DAF Max and Min CSV Checkbox.....	27
4.6	<i>The Help Screen Mode</i>	27
4.7	<i>Checking for Newer Version</i>	27
4.8	<i>Future Modes</i>	27
4.9	<i>Minimum Frequency Hop Analysis File</i>	27
4.10	<i>Future Analysis Files</i>	28
5	Troubleshooting	28
5.1	<i>Program Does Not Start</i>	28
5.2	<i>I/O Error in open_database_files</i>	28
5.3	<i>Minor Program Glitches</i>	28
5.4	<i>Request Features and Improvements</i>	28

Table of Figures

Figure 1. Total Costas Arrays Up To a Given Order	6
Figure 2. Total Unpacked Size of Database in Bytes versus Maximum Order	8
Figure 3. Installed Database Directory Structure	9
Figure 4. File Manager View of Database Root Folder	9
Figure 5. Bit Zipper with CA_Database_v3.zip Loaded	10
Figure 6. Folder with contents of CA_Database_v3.zip WWWWW	11
Figure 7. License Dialog	12
Figure 8. IEEE Dataport Citation window including IEEE and BibTeX formats	13
Figure 9. Box Including "Cite" Button	14
Figure 10. Opening Dialog	15
Figure 11. The Database Folder Selection Dialog	17
Figure 12. The Selection Mode Configuration Dialog	19
Figure 13. The Analysis Mode Dialog	20
Figure 14. Generator Method Combo Box	21
Figure 15. Costas Array Type Combo Box	22
Figure 16. G-Symmetric Costas Array of Order 10	23
Figure 17. Order 13 G-Symmetric Costas Array	23
Figure 18. Order 5 G-Symmetric Costas Array	23

List of Tables

Table 1. Files in Database	5
Table 2. Database Archive Files	9
Table 3. Generator Methods	21

1 Files in the Database

1.1 Data Files

This IEEE Dataport database contains Costas arrays for orders from 3 through 1030. For each order, either five (folder\Searches) or nine (folder\Generated) files are included for each order. These files are summarized in Table 1.

Table 1. Files in Database

File name	Purpose	Contents
Sorted_Costas_arrays_N=<order>.txt	Comprehensive list	All CAs
Costas_essense_N=<order>.txt	Shortest list	One Costas array for each set of four or eight
Costas_symmetrical_N=<order>.txt	List of all symmetrical CAs	All symmetrical CAs
Costas_G-Symmetrical_N=<order>.txt	Glide-symmetrical list	Glide-symmetrical CAs
Costas_log_N=<order>.txt	Summary information	Data for extraction utility
Costas_essense_N=<order>_methods.txt	Generator information	Hex 32-bit masks
Costas_G-Symmetrical_N=<order>_methods.txt	Generator information	Hex 32-bit masks
Costas_symmetrical_N=<order>_methods.txt	Generator information	Hex 32-bit masks
Sorted_Costas_arrays_N=<order>_methods.txt	Generator information	Hex 32-bit masks

The generators in reference [2] are used for orders 3 through 1030 to generate the files that are listed in Table 1. Certain unpublished aggressive extensions are included in the software used to generate the database; no one has come forward with a Costas array that is not in the database (other than “The Last Costas Array” of order 27 in [3]) since the first version, to order 200, was published in 2006 [7]. All these files must be placed in the folder \Generated for the database utility to work seamlessly for all orders.

For orders that have been comprehensively searched, the files are placed in folder \Searches; included are orders up to 29, which have comprehensive searches in the peer-reviewed literature^{1,2,3,4,5,6}. Since the “*N=<order>_methods.txt” files are produced by the generator program and have no relevance when the Costas arrays are found by searching, there are no files of the form “*_methods.txt” in the \Searches folder.

Here, we represent Costas arrays of order N as a set of N integers. If these integers are considered as elements of a row vector, each integer is the row index of a 1 in the corresponding column of a matrix, all other elements being zero; the resulting matrix is a Costas array. This representation is not unique; the notation or convention of this sort determines the actual permutation matrix orientation. Different conventions will result in Costas array matrices that differ from other conventions as the transpose or in having the rows or columns reversed, or a combination of these transformations.

Since rotation, transposition, or reversing the order of the rows or columns of a Costas array results in another Costas array, sets of Costas arrays related by these transformations are eight Costas arrays if

there is no symmetry about the main diagonal or antidiagonal, or four Costas arrays if such symmetry does exist. We call these *congruence sets* because Costas arrays in a given such set are congruent on transformations of transposition and rotation or reversal of rows or columns.

The number of available Costas arrays as a function of order is shown in Figure 1. curves are shown for all Costas arrays, and for generated-only. The equation fit is

$$\langle \text{Total number of Costas arrays up to a given Order} \rangle \approx 0.154 \cdot (\text{Order})^{2.82}$$



Figure 1. Total Costas Arrays Up To a Given Order

1.2 Database Utility Executable

There is a database utility in the Scripts section of this DataPort entry as the file Read_CA_Database_v3.zip. This is a ZIP file containing a Windows GUI program as a 64-bit binary, accompanied by several DLLs that provide support for Qt5 widgets, this PDF file, and CA_Database_Default_Options.txt, a sticky options default file. If your computer does not have a compatible version of Qt5 installed, place these DLLs in the same folder as the EXE file. Installation is presented in Section 3.

This utility is capable of reading the entire database or any portion of it that is available to the user through Windows, as determined by user inputs on a pop-up dialog. It has several modes; the user invokes a particular mode by selection on a pop-up dialog. The main application is driven by dialog boxes but is also provided with a scrolling text window.

Modes are provided for simple database extraction according to user-specified criteria (order, symmetry, etc.), database analysis, and a Help capability that requires Internet access.

The Help utility brings up the current page on this utility on the author's web site; a dedicated Help page for this utility will be used in the future if this PDF is deemed to need a more detailed supplement.

1.3 Other files

The file CA_Database_Information_Files.zip contains hugely helpful files, such as a spreadsheet that provides enumeration data on searched and generated lists including breakdowns by generator. In addition, a small text file produced as an information aid by the generator program provides a summary of the generators. A PDF of the CISS paper that introduced the first version of this database, to order 200⁷, is included.

An auxiliary file, FrHop_LUB_Database.zip provides a list of the LUB (minimum across the set for each order) of differences in integers in the representation of Costas arrays in the database. In the convention given in Section 1 above, if each entry in the Costas array vector represents a time increment in a frequency-hop waveform, and the row index is used for the frequency for that time interval, the difference between successive row indices is proportional to the frequency hop between time intervals. This information may be useful in selection of Costas arrays when PLL-based waveform generators are used.

Additional files that provide different analysis aids for different applications, including mathematical developments, will be added as appropriate.

1.4 Install Only What you Need

The minimum files that you will need to do most extraction and analysis of Costas arrays are these:

- Read_Database_00.zip, 9,047,818 bytes; software utility to read and analyze the database
- CA_Database_Information_Files.zip, 707,300 bytes; enumeration and other information files
- Costas_folder_Searches.zip, 2,417,331 bytes; database for folder \Searches to order 29
- CA_Database_3-100.zip, 8,929,136 bytes; database for folder \Generated to order 100

This is a total of 21.1 MB, easily downloaded in a few seconds. The database extraction and analysis utility will work with any size database automatically. Unpacking database files for orders over about 300 results in unpacked database sizes about three times larger than the ZIP file. Downloading orders beyond 100 and unpacking them into the \Generated folder results in total unpacked database size increase according to about the fourth power of the order, as shown in Figure 2.

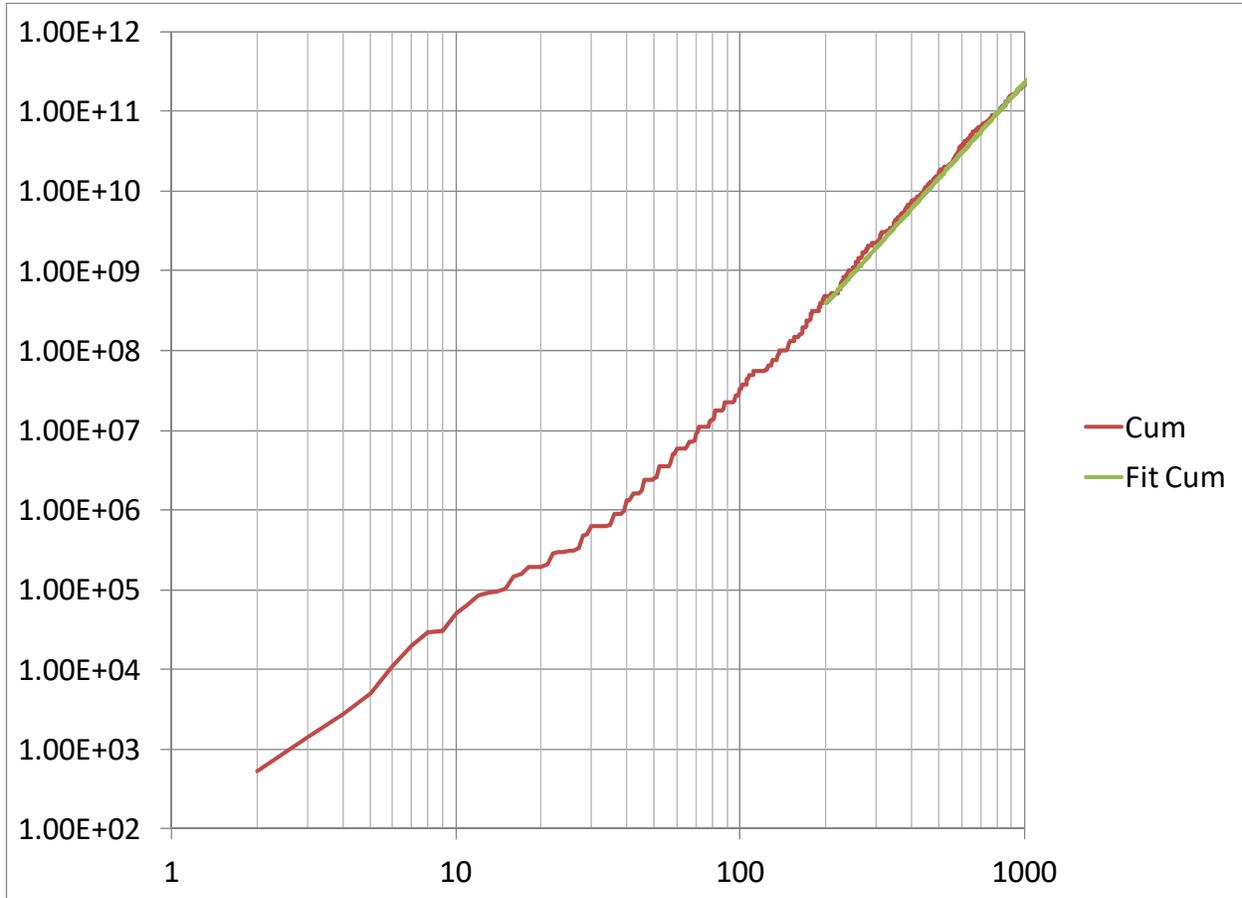


Figure 2. Total Unpacked Size of Database in Bytes versus Maximum Order

The equation for the “Fit Cum” line in Figure 2 that closely models the cumulative database size for orders greater than 200 is

$$\langle \text{Cumulative Size} \rangle \approx 0.28512 \cdot (\text{MaxOrder})^{3.9667}$$

The total file size exceeds 1 GB for orders 255 and over. If your work with Costas arrays requires working directly with orders up to a given order, and the enumeration data in CA_Database_Information_Files.zip is sufficient, then you need only the four files suggested above. Download any more that you need for direct access to Costas arrays of the order you need for additional files for the \Generated folder.

2 Installation of the Database

The Costas array database is contained in the files in Table 2. Extract the files and place in the folders according to Figure 3. When you are done, the result will look like Figure 4. Note that the enumeration spreadsheet file has been placed in the database root folder for convenience. The database extraction and analysis program uses data in these folders and ignores files in the database root folder.

Table 2. Database Archive Files

Downloaded File	Files	Store in
Costas_folder_Searches.zip	First 5 rows of Table 1	CA_Database\Searches
CA_Database_3-100.zip	All rows of Table 1	CA_Database\Generated
CA_Database_101-200.zip (optional)	All rows of Table 1	CA_Database\Generated
CA_Database_201-300.zip (optional)	All rows of Table 1	CA_Database\Generated
...	All rows of Table 1	CA_Database\Generated

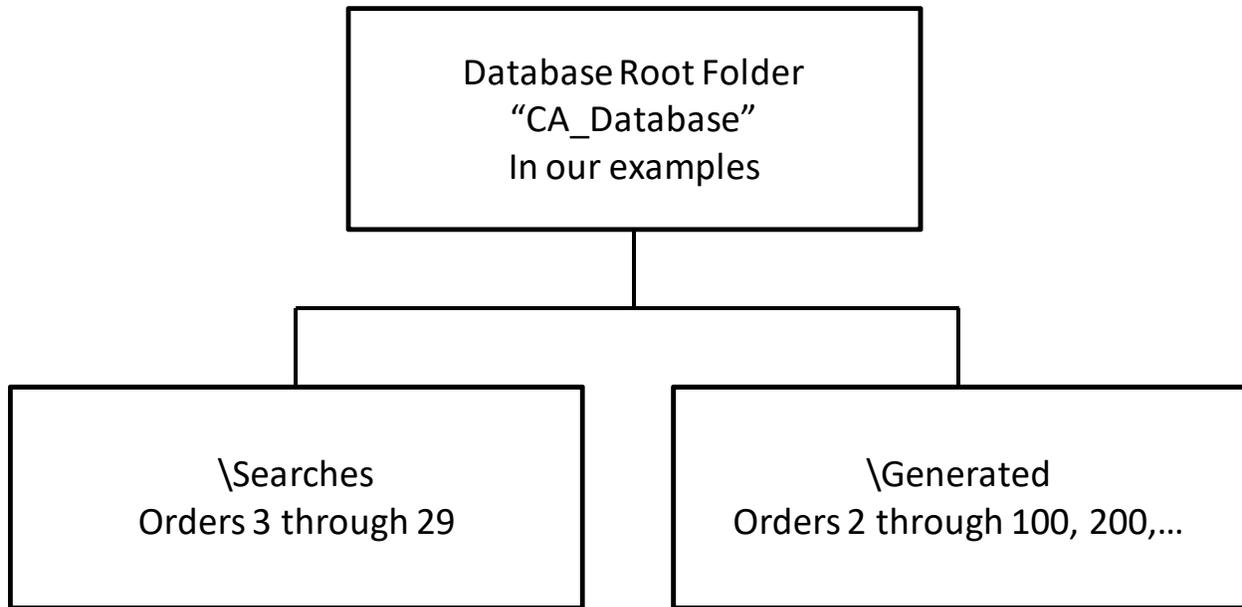


Figure 3. Installed Database Directory Structure

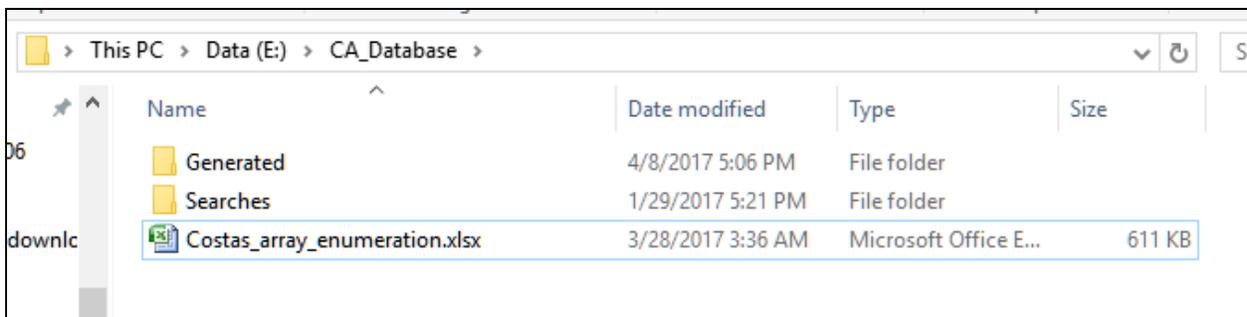


Figure 4. File Manager View of Database Root Folder

The database consists of only the data files shown here. The Windows registry or other system capabilities are not involved beyond system defaults, such as association of files with names ending in .txt with applications that read or write text files. It is recommended that you place the database in a folder of its own at or near the root folder of a local drive to simplify setup; see Section 4.4.1.1.

3 Installation and Use of the Costas Array Extraction and Analysis Utility

3.1 Version

The version of the utility documented here is 3.0. The version is part of the initial scrolling text in the base application text window and is presented other places as the program executes. You can check for a later version from the run startup option dialog as documented in Section 4.2 below. This documentation will be updated as new versions are released.

3.2 Downloading and Installation

The file is CA_Database_v3.zip, in the Scripts area of the IEEE DataPort page on this database. Note that another copy of CA_Database_v3.zip is in the Files area as a work-around for an error in downloading from the Files area.

There are a number of ZIP archive utilities. The screen of Bit Zipper when CA_Database_v3.zip is loaded, is shown in Figure 5 below.

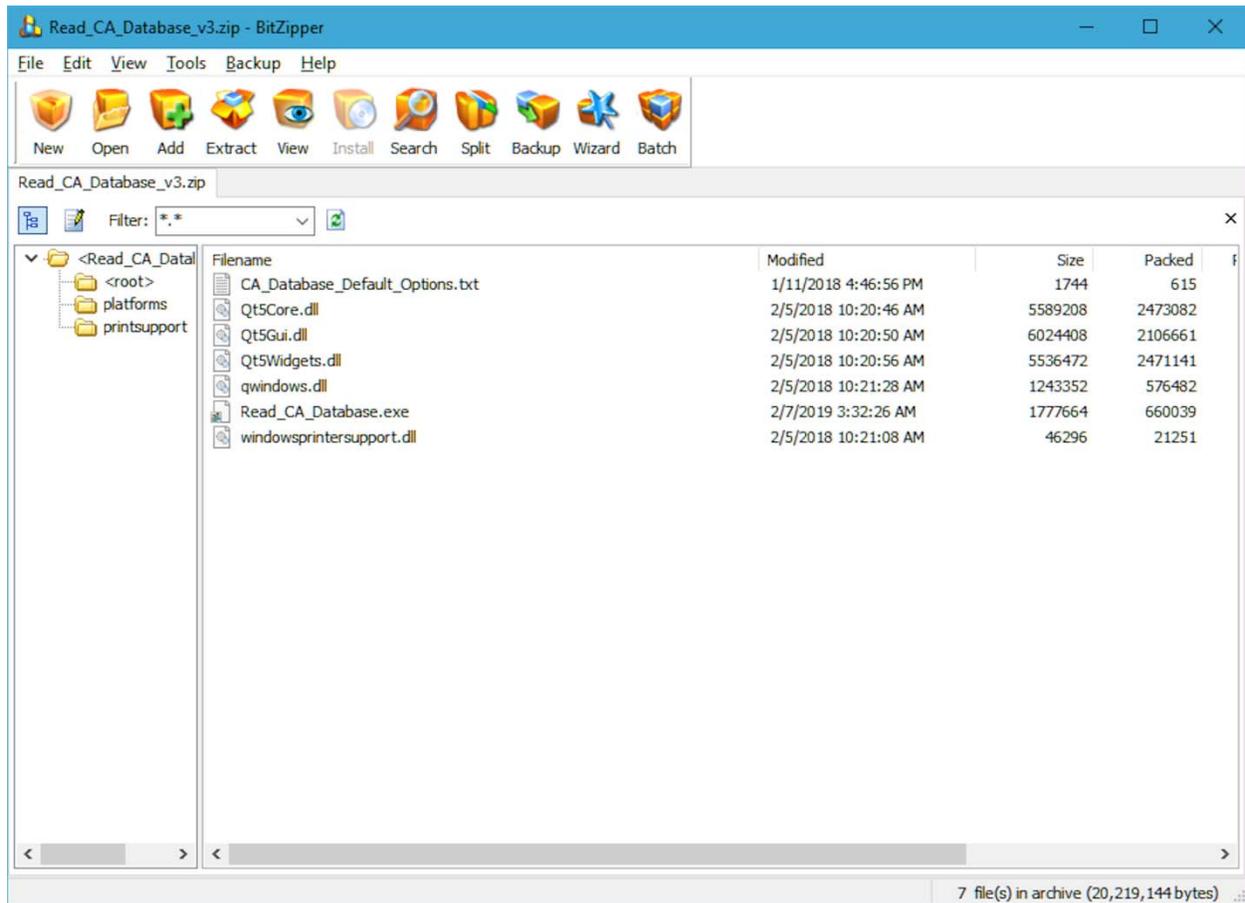


Figure 5. Bit Zipper with CA_Database_v3.zip Loaded

Extract the files to any folder, preferably one that is empty (except possibly for CA_Database_v3.zip itself). The folder should be as shown in Figure 6 below. The folder .\platforms contains qwindows.dll and the folder .\printersupport contains windowsprintersupport.dll. These and the DLLs of the form

Qt5*.dll are Qt DLLs that support Qt widgets, which are used extensively by the application, Read_CA_Database.exe. The text file supports sticky options (see below) and may be viewed but and a derivative file are maintained by the application and should not be edited. If the program gets an error in reading either of these files (and the first time it is run), it will replace them with a default and can cause inconvenience by forcing the user to restore all the sticky options that the program would ordinarily maintain as the defaults for some options whenever it executes.

: PC > Data (E:) > CA_IEEE > Sample

Name	Date modified	Type	Size
platforms	2/9/2019 11:07 AM	File folder	
printsupport	2/9/2019 11:07 AM	File folder	
CA_Database_Default_Options.txt	1/11/2018 4:46 PM	Text Document	2 KB
Costas Arrays to Order 1030 INSTRUCTIO...	2/7/2019 6:46 PM	Adobe Acrobat D...	1,109 KB
Qt5Core.dll	2/5/2018 10:20 AM	Application extens...	5,459 KB
Qt5Gui.dll	2/5/2018 10:20 AM	Application extens...	5,884 KB
Qt5PrintSupport.dll	2/5/2018 10:20 AM	Application extens...	316 KB
Qt5Widgets.dll	2/5/2018 10:20 AM	Application extens...	5,407 KB
Read_CA_Database.exe	2/9/2019 10:09 AM	Application	1,736 KB
Read_CA_Database_v3.zip	2/9/2019 11:06 AM	BitZipper ZIP Arch...	9,149 KB

Figure 6. Folder with contents of CA_Database_v3.zip WWWWW

The two ZIP archives Costas_Searched_N=27 and Costas_Searched_N=28 are for the convenience of users with old downloads who may feel that they need these files in \Searches and should be ignored, and can safely be deleted. Note that there are no Costas arrays of order 28 or 29 that are not included in the \Generated database. They may safely be deleted and will not be included in future releases.

The original download, Read_CA_Database_v3.zip, can be deleted once proper execution of Read_CA_Database.exe is verified.

The registry is not used in installation or execution of this program. You can completely delete this application by deleting the folder with the EXE file and the Qt DLLs.

You may now place shortcuts to Read_CA_Database.exe in your user interface. This can be done by dragging Read_CA_Database.exe to the Start menu quick-start area or the taskbar, or to most dock applications. If you use a conventional Windows shortcut file, the starting folder should be left blank so that the program starts in the folder with the EXE file; this is because the application-maintained configuration file must be in the folder in which the application is running.

3.3 The Application-Maintained Configuration File

A number of options, once set, are usually not changed again, or at least not changed often, for a particular user. Examples are location of the database files, path for the output files, whether the utility is being used in a Windows or Linux environment (currently Linux and Mac versions are not provided),

etc. These options are stored as a text file, CA_Database_Options.txt, after each database mode is executed, and this file is loaded by each database mode as the defaults for some options.

If CA_Database_Options.txt is not present, or if the program encounters a read error in loading this file, the file CA_Database_Default_Options.txt is copied to CA_Database_Options.txt. If there is a read error for CA_Database_Default_Options.txt, it is replaced by the program, but the user may edit this program to change the default options.

4 Using the Utility

4.1 Starting the Utility

When first executing Read_CA_Database.exe, the license file dialog pops up. This window is shown as Figure 7 below. If Cancel is clicked, no action is possible other than to close the window and try again. You must agree to the license by clicking OK to run the utility.

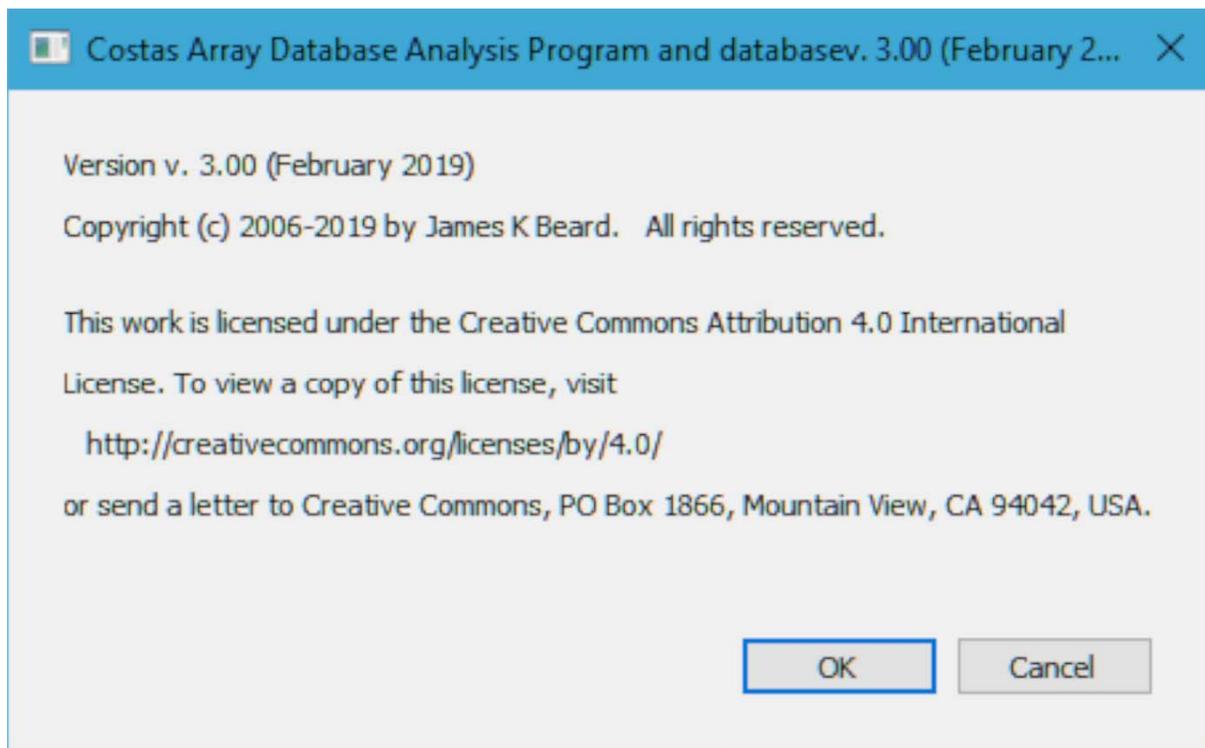


Figure 7. License Dialog

Note that this is the Simplified Attribution 4.0 International (CC BY 4.0), available from Creative Commons at <https://creativecommons.org/licenses/by/4.0/>. This license allow use and extension of the work, including incorporation of the work into other work, including free and commercial software, with the only requirement being attribution, and you may not claim ownership to the database or restrict access or use of the database by others. “The work” means the database files and other files in the database including the enumeration spreadsheet and this Instructions file, and the program binary Read_CA_Database.exe. Examples of attribution:

[1] James K Beard, "Costas arrays and enumeration to order 1030", IEEE Dataport, 2017. [Online]. Available: <http://dx.doi.org/10.21227/H21P42>. Accessed: <date you downloaded>.

BiBTeX:

```
@misc{h21p42-17,  
doi = {10.21227/H21P42},  
url = {http://dx.doi.org/10.21227/H21P42},  
author = { James K Beard },  
publisher = {IEEE Dataport},  
title = {Costas arrays and enumeration to order 1030},  
year = {2017} }
```

Note that the DOI link will immediately take you to the database pages in IEEE Dataport, which may be accessed by anyone with an IEEE logon or a free IEEE Dataport account. On that page, there is a box below the AWS S3/Download/Scripts box as shown as Figure 9. Pressing the Cite button brings up the citation formatting window shown here as **Error! Reference source not found.** Clicking the tabs brings up clipboard-copyable examples that reference this database in all standard publication formats, including IEEE, BibTeX, RIS, APA, Harvard, MLA, Vancouver and Chicago.

The two examples given above are the IEEE format and the BibTeX format; note that the [bare_jrnl.tex](#) template by Michael Shell accepts the label @misc<label>,... rather than @data<label>,... as given in the Cite dialog. This template loads [IEEEtran.cls](#).

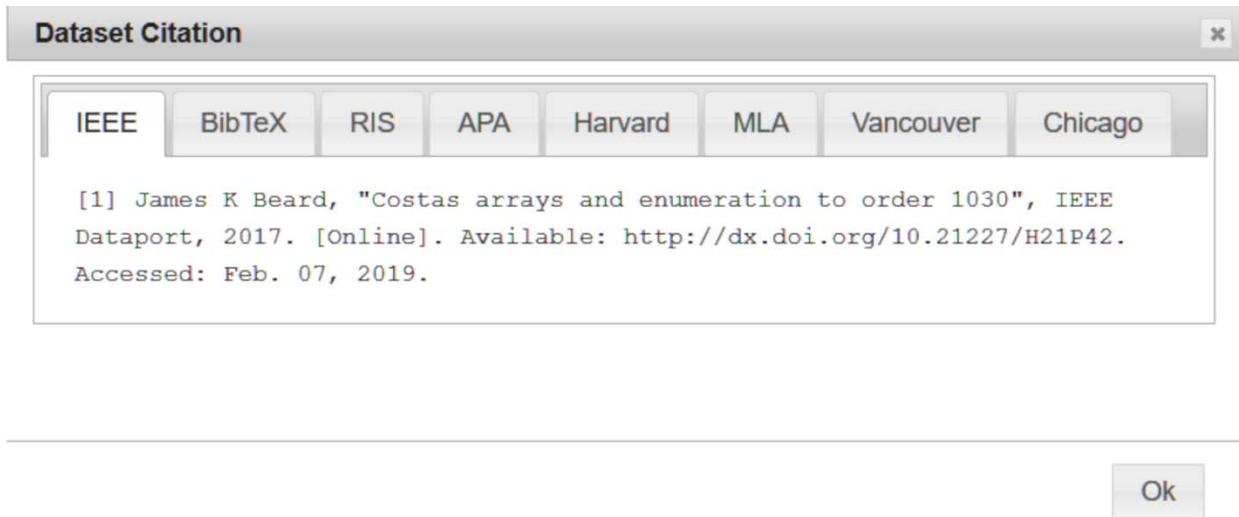


Figure 8. IEEE Dataport Citation window including IEEE and BibTeX formats

COSTAS ARRAYS AND ENUMERATION TO ORDER 1030

Citation Author(s): [James K Beard](#)

Submitted by: [James Beard](#)

Last updated: Sat, 06/16/2018 - 23:18

DOI: [10.21227/H21P42](https://doi.org/10.21227/H21P42)

Data Format: ZIP(binary txt and csv files)

Links: [JK Beard's web site page](#),
[JK Beard's Costas array files](#),
[Wikipedia: Costas array](#),
[Online Encyclopedia of Integer Sequences](#)

License: [Creative Commons Attribution](#) 

Dataset Views:  479

[Share / Embed](#)[Cite](#)

Figure 9. Box Including "Cite" Button

4.2 The Opening Dialog

Clicking OK on the license dialog opens the mode selection dialog shown in Figure 10. The text widow behind the dialog includes provenance, license, and referencing information.

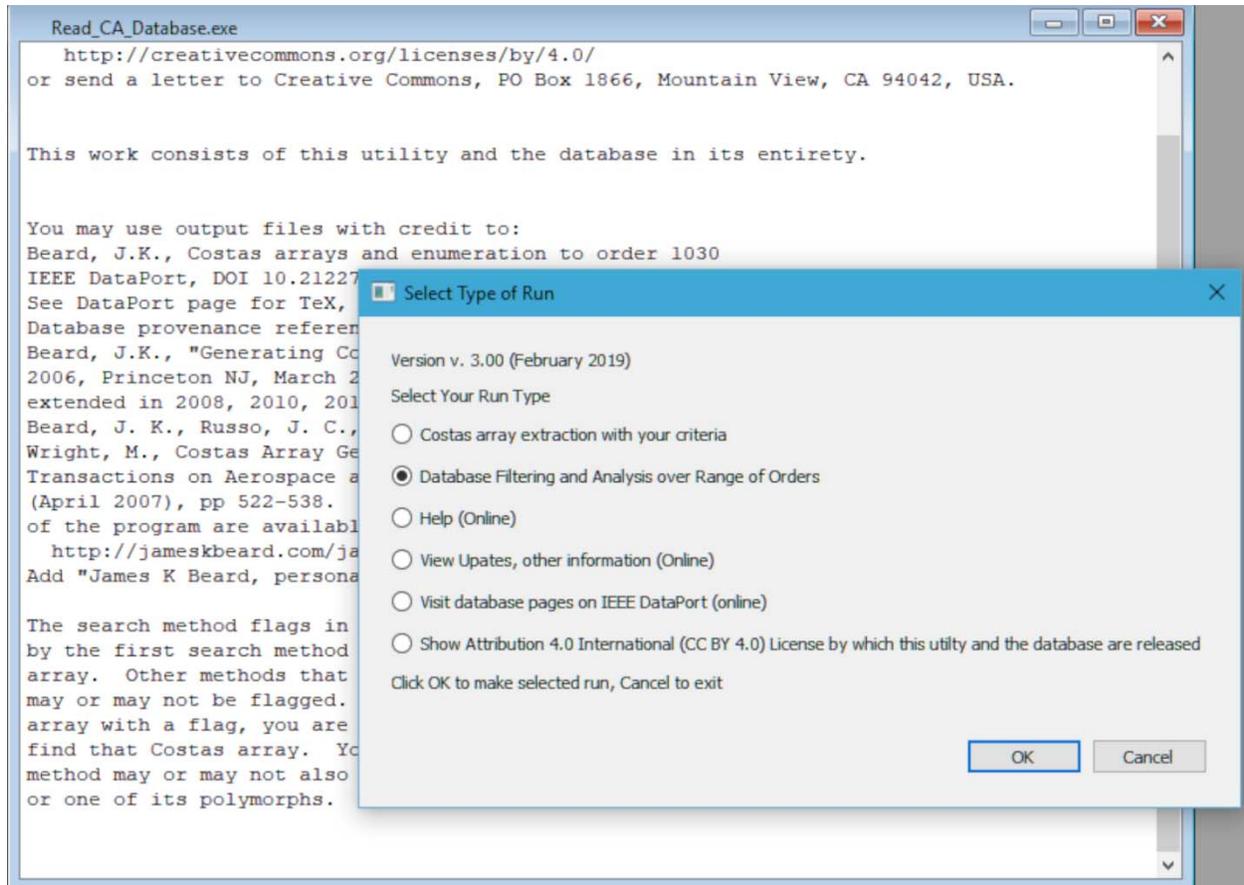


Figure 10. Opening Dialog

The first two dialog selections offer user options for running the utility. The remaining selections open your default browser to web pages and require Internet access to function properly.

4.3 The Background Scrolling Text Window

The application itself runs in the main application window with the application title, and a scrolling text window runs in this window, as shown in Figure 10. Closing the application window closes the application. Closing the text window is possible but we advise against that because status, information, and heartbeat information are posted to the scrolling text window. Nearly all information in the scrolling text window is also saved in an output file, Costas_Array_Database_Output.txt. Although the Costas arrays that are selected by user-defined filtering in the application are written to a CSV file as output, the scrolling text window allows preview of the first 20 terms in any Costas array selected for output.

4.4 The Selection and Analysis Modes

Invoking the Selection mode by clicking the first radio button in the opening dialog, then OK, brings up the mode configuration dialog shown in Figure 12. Invoking the Analysis mode by clicking the second radio button in the opening dialog, then OK, brings up the mode configuration dialog shown in Figure 13. The dialogs are similar, each having three sections, Program Configuration, Extraction, and Computation. Configuration options determine how the program operates; these options are not changed often, and are “sticky” options that default to the values used in the most recent run. Extraction options provide several ways that you can filter the Costas arrays to be extracted or used in computations. Computation options are operations that are to be performed on Costas arrays that meet the filtering criteria in the Extraction section. The overall dialogs seem similar, with the differences being

- The Selection mode accepts only one Costas array order, whereas the Analysis mode accepts a starting and ending order, and repeats its operations over a range of orders.
- There are a lot more options available in the Computation section of the Analysis mode.

In the following sections, we will walk through the options in each section of these dialogs.

4.4.1 Program Configuration

4.4.1.1 Location of Database

The Windows Explorer folder selection dialog is invoked by clicking the “...” to the right of the first two options in the dialogs of Figure 12 or Figure 13. Use of this makes typing in the path of the database folder unnecessary.

The database itself may be in a folder on your system drive, a folder on another hard disk or USB hard disk, on a USB thumb drive, a CD-ROM or data DVD, or a network drive or other location that is attached and available to applications as a drive letter. The database root folder must be known to the application; this is why this is the first option in each mode that accesses the database. The database root folder must contain the subfolders \Searches and \Generated; see Figure 3 for a file manager view of the database root folder; note the extra file in the root database folder – files in this folder are ignored by the utility. The minimum workable contents of the database folder are as shown in Figure 11.

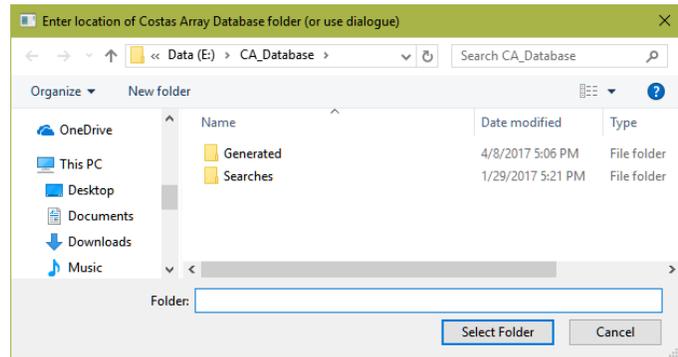


Figure 11. The Database Folder Selection Dialog

4.4.1.2 Program Output Files Location

This option, new with this release of the Costas Array Database Extraction and Analysis Utility, allows you to put the output of the program anywhere you have write permission. The window selection dialog, as shown for the database folder dialog in Figure 11, begins with the folder where the Read_CA_Database.exe file resides, and a convenient option is to open a subfolder there and to make subfolders of that folder for output files from runs of the utility. But you can use the folder selection dialog to place your output files anywhere your computer has access and that you have write permission.

4.4.1.3 Clear CSV Files from Output Folder Before Run

Checking this box will use a system command to delete all CSV files from the folder selected as described in Section 4.4.1.2 before running the program. This will momentarily open a PowerShell window while the system command is running. If you are clearing a folder with a vast amount of data (many megabytes or even in the gigabyte range), this operation can take several seconds. Unless you have extracted Costas array files that you want to delete to avoid confusing new output, leave this box unchecked.

4.4.1.4 Overwrite or Append Old Files

If you have a run that is a logical extension of an existing run, as when you want to extend the range of orders that are searched, uncheck this box and your new run will append its data to existing files where appropriate. Checking this box will overwrite files of the same name with new runs.

4.4.1.5 Generated Database Only Checkbox

The database is in two folders, \Generated and \Searches. The \Searches database contains databases on orders 2 through 29. These orders have been subjected to exhaustive searches in the peer-reviewed literature^{1,2,3,4,5,6,7}. For orders 8 through 27, the exhaustive searches find Costas arrays that are not produced by the generators⁶. For this reason, to make all known Costas arrays available to the program, the recommended state is to use the \Searches database for orders 27 and below. Although orders 28 and 29 have been exhaustively searched, there are none that are not produced by the generators^{4,5}. The \Generated database includes information about which generator produces each Costas array, so that database is preferred when it has all known Costas arrays for a given order, and is always used for orders 28 and higher.

When only the Costas arrays that are produced by the generators are desired, or when generator information is essential, the user may check this box and the data in \Generated which includes the information on which generator produces each particular Costas array.

4.4.1.6 Index from 0 to N-1 Checkbox

The program produces comma-separated values (CSV) files for use as input to other programs or loading by spreadsheet programs, and for output to the scrolling text window. The user or software that uses these files may prefer that the Costas array row vector be indexed from 0 to N-1, and this is done when this box is checked. Default when this box is not checked is to index from 1 to N.

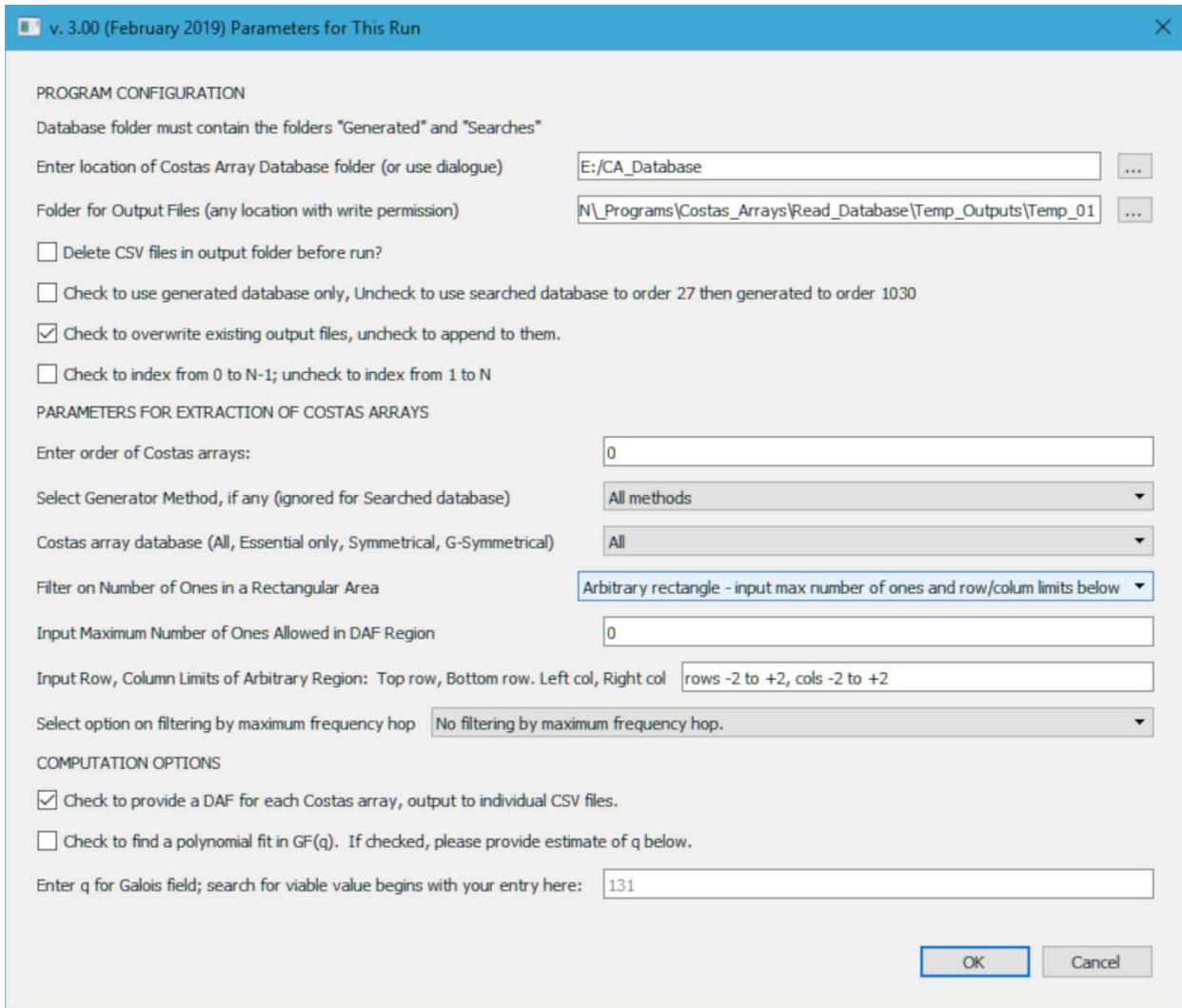


Figure 12. The Selection Mode Configuration Dialog

v. 3.00 (February 2019) Parameters for This Run

PROGRAM CONFIGURATION

Database folder must contain the folders "Generated" and "Searches"

Enter location of Costas Array Database folder (or use dialogue) ...

Folder for Output Files (any location with write permission) ...

Clear CSV files from output folder before running?

Check to use generated database, Uncheck (RECOMMENDED); use searched database to order 27, generated to order 1030

Check to overwrite existing output files, uncheck to append to them.

Check to index from 0 to N-1; uncheck to index from 1 to N

SELECTION AND FILTERING PARAMETERS FOR EXTRACTION OF COSTAS ARRAYS

Enter starting order of Costas arrays:

Enter ending order of Costas arrays:

Select Generator Method, if any (ignored for Searched database)

Costas array database (All, Essential only, Symmetrical, G-Symmetrical)

Filter on Number of Ones in a Rectangular Area

Input Maximum Number of Ones Allowed in DAF Region

Input Row, Column Limits of Arbitrary Region: Top row, Bottom row. Left col, Right col

Select option on filtering by maximum frequency hop

COMPUTATION OPTIONS

Check to provide a DAF for each Costas array, output individual CSV files.

Check to find a polynomial fit in GF(q). If checked, please provide estimate of q below.

If previous box checked, enter q for Galois field; search for viable value begins with your entry here:

Check to write CSV file of counts of different types and database file sizes.

Check to output CSV of counts of ones in central 3X3 square (SLOW for lots of large orders)

Check to output CSV of counts of ones in central 5X5 square (SLOW for lots of large orders)

Check to re-compute minimum peak frequency hop database . Orders taken from database; SLOW for lots of large orders.

Check to output DAF max and min CSV (SLOW for lots of large orders).

OK Cancel

Figure 13. The Analysis Mode Dialog

4.4.2 Filtering Parameters

4.4.2.1 Order of Costas Arrays (Selection Mode only)

In the Selection mode, you enter the single order of Costas arrays that you are investigating here.

4.4.2.2 Starting and Ending Orders of Costas Arrays (Analysis Mode only)

In the Analysis mode, you enter the starting and ending orders of Costas arrays that you wish to investigate here. They may be the same order if you are only looking at one order, but you want the analysis options available in this mode. If the ending order is lower than the starting order, no action is performed.

4.4.2.3 Generator Method Combo Box

The Generator Method combo box allows the user to select only one generator method. About half of the list is shown in the screen shot of Figure 14. The full list is shown in Table 3 from [2]. No filtering by method is done when the default, All methods, is selected. This option is grayed out when data is taken from a searched database and method information is not available.

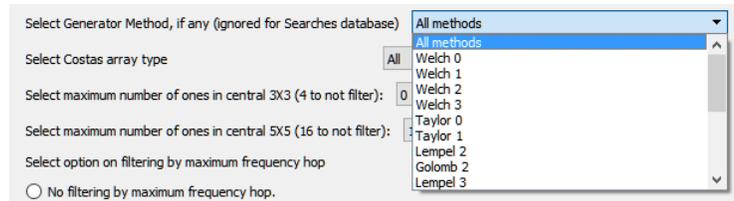


Figure 14. Generator Method Combo Box

Table 3. Generator Methods

Method	Notes
Welch 0	Welch 1, add dots at one of (1, 1), (1, p), (p, 1), or (p, p)
Welch 1	One of two base methods, $i = \alpha^{j-1} \in GF(p)$, p prime
Welch 2	Welch 1, with dot at (1, 1) removed, when it is present
Welch 3	Welch 1 with dots at (1,1),(2,2) removed; these dots are present when $\alpha = 2$
Taylor 0	Lempel-Golomb 2, when $q \equiv -1 \pmod 6$ add dot pair at (1, 1), (q, q) or (1, q), (q, 1)
Taylor 1	Lempel-Golomb 2, add a dot at (1, 1), (1, q), (q, 1), or (q, q)
Lempel 2	One of two base methods, $\alpha^i + \alpha^j \equiv 1 \in GF(q)$
Golomb 2	One of two base methods, $\alpha^i + \beta^j \equiv 1 \in GF(q)$
Lempel-Golomb 3	Lempel 2, Golomb 2, dot at (1,1) removed, works when $\alpha = 2$
Lempel-Golomb 4	Lempel 2, Golomb 2 when $q = 2^k$, dots at (1, 1) and (2, 2) removed
Taylor 4	Lempel 2, dots at (1, 2) and (2, 1) removed
Golomb* 4	Lempel-Golomb 3, $\alpha + \beta = 1$, delete dots at (1, 1), (2, q - 2)
Golomb* 5	Golomb* 4 with additional dot at (q - 2, 2) also removed
Inhom. Add 1	Lempel-Golomb 2 with γ on right-hand side; singularity
Inhom. Sub 1	Inhomogeneous Additive 1, but with subtraction on left-hand side; singularity
Rickard W 0	Welch 1, j incremented mod(p), with added dot
Rickard L-G 1	Lempel-Golomb 2, j incremented mod(q - 1), added dot

The generator program sets a bit in a bitmask at a position that denotes one of the generators in Table 3 when a Costas array is generated and is found not to be a duplicate of a Costas array already in the database. Then, the Costas array is transposed and rotated to find all four or eight Costas arrays in its congruent set and all four are added to the database with the same method bitmap. Careful examination of this scheme reveals that it is possible that Costas arrays are in the database that are generated by methods that are not flagged in the associated methods bitmap.

The generator routine, not released to anyone at present, is capable of performing searches using any combination of methods, and one can be assured that all the Costas arrays found by a particular method may be found in a run focused on that determination by enabling only that method.

The methods in Table 3 all begin with a Welch 1 generator, which generates Costas arrays of order one less than every prime, and the Lempel-Golomb 2 generator, which generates Costas arrays of order two less than any power of a prime $N + 2 = q = p^k$. Note that each uses a Galois field of order q , denoted as $GF(q)$. Positive integer arithmetic modulo a prime p is the simplest example of arithmetic in a Galois field. Several of the generator methods apply only for such fields, i.e. $q = p^1 = p$. In particular, the Welch generator and any of the methods based on it apply only with $k = 1$. The Lempel-Golomb method applies for $k \geq 1$ but the Taylor 0, Lempel-Golomb 3, Taylor 4, Golomb*4 and Golomb*5 all begin with $GF(p)$ where $k = 1$.

The generator program sets bits in a bitmask associated with a generated Costas array that is provided with the database as an ASCII HEX data file associated with each Costas array. Generalizations that add or remove dots sets other bits in the bitmask, but older versions that selected for Welch 1, Lempel 2, or Golomb 2 generators would select all Costas arrays that were derived from these generators. The selected arrays often were not easily traceable to a Welch 1 or Lempel-Golomb 2 generator. This release of the utility now selects only the designated generator type and rejects Costas arrays that were obtained by further manipulation.

4.4.2.4 Costas Array Database Type Combo Box

The Costas array type combo box allows filtering for symmetrical Costas arrays, or only searching a minimal database that contains just one of the four or eight members of any given congruent set. This combo box is shown in Figure 15.

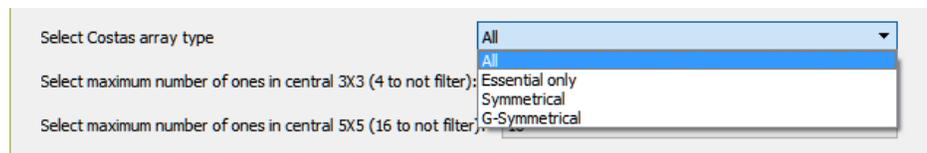


Figure 15. Costas Array Type Combo Box

The list of Essential Costas arrays is found by selection of the one of the four or eight Costas array vectors in each congruent set that begins with the smallest column indices.

The list of Symmetrical Costas arrays includes only those Costas arrays of a given order that are symmetrical about the main diagonal. Note that all Costas arrays generated by the Lempel 2 method, e.g. Lempel-Golomb with $\alpha = \beta$, are symmetrical, as are generalizations that begin with such arrays when adding or deleting dots preserves the symmetry property.

The list of G-symmetrical Costas arrays is a list of glide-symmetric Costas arrays. These are enumerated and given their own section in each database because

Herbert Taylor⁸ noticed that all Costas arrays generated by the Welch 1 method have glide-symmetry, which he defined for Costas arrays as the bottom half of the matrix being congruent with the top half, but with the order of the columns reversed. Figure 16 shows order 10 Costas array (1,2,9,3,5,10,8,4,7,6). Note that the top and bottom five rows are congruent on reversal of columns. Entries in the G-Symmetric database are determined by checking for this property, not by selecting Costas arrays generated by the Welch 1 method.

An interesting logical point is raised for odd order G-symmetric Costas arrays: There will be a middle row, and for G-symmetry to apply the middle row must have a 1 in the center element. But since the Welch 1 generator generates Costas arrays of order one less than a prime, then there will be none generated that are of odd order. But with the database extraction utility we find that there are 16 G-symmetric Costas arrays of order 13; the example (3,11,6,9,1,5,7,12,8,2,13,4,10) is shown as Figure 17, and, yes, the center element is a one. Our enumeration spreadsheet in CA_Database_Information_Files.zip tells us that these are Lempel-Golomb 3 Costas arrays, and, further, that four G-symmetrical Costas arrays of order 5 exist. One of these is shown in Figure 18; our enumeration spreadsheet tells us that this Costas array may be generated by Lempel-Golomb 3 or other methods.

Taylor and others have performed comprehensive searches over orders higher than 29 with the constraint of symmetry and G-symmetry; he and others have reported⁹ that there are no symmetrical or G-symmetrical Costas arrays of order

1									
	1								
		1							
						1			
			1						
								1	
							1		
		1							
				1					

Figure 16. G-Symmetric Costas Array of Order 10

				1								
								1				
1										1		
					1							
	1											
						1						
								1				
		1										1
1										1		
							1					
									1			

Figure 17. Order 13 G-Symmetric Costas Array

1				
			1	
		1		
				1
	1			

Figure 18. Order 5 G-Symmetric Costas Array

32 or 33.

4.4.3 Selection by Bounding the Number of Ones in a User-Selected Rectangle of the DAF

This combo box allows the user to select only Costas arrays that have a maximum number of ones in any rectangular region of the DAF. The sign convention is for row one of the Costas array matrix to be the bottom row and row indices increase as you move up. For frequency hop waveforms, if this agrees with your use of a Costas array, the DAF will correspond to what you observe in the ambiguity function. If you index the Costas array matrix beginning with row one at the top, or your frequency indexing in the waveform is the opposite direction, the DAF produced by the utility will be upside-down relative to what you observe in your ambiguity functions.

There are four selections in the combo box:

- No selection by ones in a DAF region,
- Selection by maximum count of ones in the central 3X3 region,
- Selection by maximum count of ones in the central 5X5 region, and
- Selection by maximum count of ones in a rectangular region specified by the user.

Each of these is explained in more detail in the following paragraphs.

4.4.3.1 No Filtering on Count of Ones in the DAF

With this option selected, filtering is bypassed. The user may filter by generator method and select the type of Costas array by specifying subsets such as symmetrical instead of the full set for each order.

4.4.3.2 Bound the Number of Ones in Central 3X3 (4 to not filter)

In formulating a waveform, selection may be done in part to achieve better performance near the central peak of the ambiguity function¹⁰.

This option and the next offer an opportunity to select Costas arrays with a discrete ambiguity function (DAF) that has not only a specified number of ones near the central peak, which is always equal to N . The DAF is defined from a stationary Costas array matrix as shown in Figure 16, Figure 17 or Figure 18, and an identical Costas array that is moved to the right/left or up/down by a discrete number of rows and columns. By the Costas condition, the count is always zero or one except with zero shifts, when the count is N .

The properties of a convolution matrix make it impossible for any ones to coincide when the number of rows shifted is zero except when the number of columns shifted is also zero, and vice versa. Therefore the maximum number of ones possible in the 3X3 central square of the DAF is $(3-1)^2 = 4$, not $3^2 = 9$, so setting this threshold at 4 disables filtering. The DAF is glide-symmetrical so this option is a combo box with allowable values of 0, 2, and 4.

Note that this option is redundant with that of Section 4.4.4, the principal advantage being that a simple click correctly specifies the maximum and minimum rows and columns to be -1, +1, -1 and +1.

4.4.3.3 Bound the Number of Ones in Central 5X5 (16 to not filter)

This combo box allows the user to filter on the number of ones allowed in the central 3X3 cells of the DAF. The maximum number of ones possible in the 5X5 central square of the DAF is $(5-1)^2 = 16$, not $5^2 = 25$ and again the number of ones must be an even number, so we have a combo box with allowable values of even numbers between zero and 16.

Note that this option is redundant with that of Section 4.4.4, the principal advantage being that a simple click correctly specifies the maximum and minimum rows and columns to be -2, +2, -2 and +2, which is the default for the option of Section 4.4.3.4.

4.4.3.4 Bound the Number of Ones in Arbitrary User-Selected Rectangle

This is the most general option, since it can be centered anywhere in the DAF, can be any size, can be rectangular instead of square, and can specify an odd number of rows or columns. Also, this option adds a row to a table of the number of ones in the specified area, which is produced in the output file DAF_Counts_Profile.csv.

4.4.3.5 Setting the Bound on the Count of Ones

This input need not be set if selection by maximum count of ones is set to No Filtering as in Section 4.4.3.1 but must be specified if selection by count of ones in 3X3, 5X5, or arbitrary user-specified regions are selected. Enter only the numeral zero or contiguous numerals for a positive integer in this field.

4.4.3.6 Setting Row and Column Limits for DAF Region

This input need be set only for selection in an arbitrary region of the DAF is selected, and is the field that is used to specify the region. The default for this input, as shown in Figure 12 and Figure 13, is rows -2 to +2 and columns -2 to +2. This is the central 5X5 square. You can edit this line or type in your own inputs from scratch, with a 128-character limit. The program ignores non-numerical characters between the numbers in this field. A plus sign is optional and may be omitted if desired. The program shows the lowest row or column first, then the second, but the program orders the numbers after input so the only important thing is that the first two numbers indicate the lower and upper row numbers and the second two numbers indicate the leftmost and rightmost column numbers.

4.4.4 Filtering by Maximum Frequency Hop Radio Buttons

Costas arrays are used in frequency-hop schemes where the row index represents a frequency, and, usually, these frequency bins are adjacent and equally spaced. When the user is selecting Costas arrays according to maximum difference between adjacent row indices, the selection minimizes the magnitude of the frequency change required to implement the waveform¹⁰. Although direct digital synthesis effectively makes this frequency change time zero, phase-locked loop (PLL) implementations are often used in small, low-power, or low cost systems such as cell phones, WiFi modems, etc. This radio button allows selection of the Costas arrays of a given order that are the minimum for that order. A second radio button extends the filtering to end-around differences in row index.

4.5 Computation Options for All Database Access Modes

4.5.1 Provide DAF Checkbox

Checking this box provides the DAF an additional CSV file. The default is to provide a DAF for each selected Costas array. Uncheck this box if you only want list of selected Costas arrays. Since each DAF is in its own CSV file, unchecking this box can speed up runs that select large numbers of Costas arrays.

4.5.2 Provide Generator Polynomial in $GF(q)$ Checkbox

Checking this box provides a polynomial in $GF(q)$ that generates the Costas array row indices. The generator is of the form

$$x^{r(i)} = \sum_{s=0}^{N-1} c_s \cdot x^{i-s}$$

where x is any generating element of $GF(q)$ and $r(i)$ is the i^{th} row index of the Costas array vector.

The coefficients are stored as integers $index(c_s)$ so that the coefficients c_s are found by $c_s = x^{index(c_s)}$ and the index of the value of the polynomial is the row index of a one for column i . This allows the use of integers between zero and $q-2$ to represent all the nonzero elements of $GF(q)$ in a CSV file. A negative integer means that the coefficient is zero.

4.5.3 Order of Galois Field q

If the generator polynomial box is checked as in Section 4.5.4.1, you must enter a value for the order q of the Galois field in the text box immediately below the check box. The value of q must be $N+2$ or larger. The program checks to see if the entered value of q is a valid value of the form p^k where p is a prime and k is a positive integer; if not, the value of q is incremented by one and the test repeated; this continues until a valid value of q is found for which $GF(q)$ exists. This ensures that a polynomial in elements of $GF(q)$ that generates the Costas array exists.

This capability was used in some work reported in CISS¹¹. Generalizations of the Lempel-Golomb generator were considered, and this utility does reproduce the generator when it is given a Lempel-Golomb 2 Costas array. Modified Lempel-Golomb inhomogeneous and Rickard Lempel-Golomb may also produce generator polynomials with only a few nonzero coefficients. This capability is available for research.

4.5.4 Computation Options Available in Analysis Mode Only

4.5.4.1 Write CSV File of Database File Sizes Checkbox

Checking this box enables an analysis of the database file sizes. It is very slow for large databases and of limited value, because its principal information is given in Figure 2, but the capability is available to the user. A CSV file is written with counts of different types of Costas arrays and of the file sizes in the database.

4.5.4.2 Output CSV of Counts of Ones in Central 3X3 Square Checkbox

Check this box to provide counts of ones in the central 3X3 square of the DAF.

4.5.4.3 Re-Compute Minimum Peak Frequency Hop Database Checkbox

The minimum frequency hop for each order is hard-wired in the database extraction and analysis utility. Check this box to recomputed it for your database as stored on your computer.

4.5.4.4 Output DAF Max and Min CSV Checkbox

Check this box to output CSV files that represent the maximum and minimum values in each DAF cell. The DAF is a square matrix $(2 \cdot N - 1)$ cells on a side. Orders with large numbers of Costas arrays will produce a minimum DAF matrix that is mostly zeros except for the center cell (which is N) and a maximum DAF matrix that is mostly ones except for the center cell.

4.6 The Help Screen Mode

This option brings up the web page <http://jameskbeard.com/jameskbeard/Files.html#CostasArrays> in your default web browser. Future versions will look in the database root folder for the current version of this file and display it in the PDF viewer of your web browser. This provides a Help capability for offline computers. If it is not found there, this mode will display this file from the IEEE DataPort web site or my web site.

4.7 Checking for Newer Version

This option looks at <http://jameskbeard.com/jameskbeard/Files.html#GUIDATABASE>, the online posting of the database extraction utility on my web site. Future versions will look for it on the IEEE DataPort web site.

4.8 Future Modes

The modes that exist in the database utility were developed and installed to meet the needs of users as these needs arose. If you have a suggestion for a mode that is of interest to users of Costas arrays, please post your suggestion to the IEEE DataPort page for this database.

As the utility has evolved, differences between the Extraction and the Selection and Analysis modes has decreased to the point that the Extraction mode is a subset of the Selection and Analysis mode. This is why the Selection and Analysis mode is the default. Unless user feedback indicates continued implementation of two separate modes, the Extraction mode may be deleted in the next revision.

The selection of 3X3 and 5X5 central regions in Section 4.4.3.2 and Section 4.4.3.3 is a one-click definition of a square central region, but these combo box selections are duplications of the more general section of an arbitrary region as given in Section 4.4.3.4. Unless user feedback indicates that these selections are useful as they are, they will be deleted or modified to perform new functions in future releases.

4.9 Minimum Frequency Hop Analysis File

The minimum frequency hop, both the simple and the end-around computation, are available as a spreadsheet file.

4.10 Future Analysis Files

More analysis files will be uploaded as available. You may submit files or ideas for new analysis files that you believe are of interest to Costas array users.

5 Troubleshooting

5.1 Program Does Not Start

If the program does not start, make sure that you have extracted all the files from Read_CA_Database_v3.zip including the two folders .\platforms and .\printsupport, each of which will contain one small DLL. The only issues found to date with the program starting properly have been traced to missing Qt5 DLLs. The contents of the folder containing the extraction utility should look as in Figure 6. Other files such as CA_Database_Options.txt and Read_CA_Database.exe output.txt will be present once the program has been run.

5.2 I/O Error in open_database_files

This error shows up in a pop-up with the window title “I/O Error in open_database_files” and a message such as

```
LUN 11
File name: D:\\Searches\\Costas_essence_N=5.txt
The system cannot find the path specified
```

The main text window will then show a system I/O error for reading past an end of file and the program will end, leaving the application open, ready for another run.

This error occurs when the database location is not properly set as described in Section 4.4.1.1; note that this need be done only the first time the program is run. Any subset of the database may be present but must include the folder .\Searches containing the database of Costas arrays found by exhaustive search, and at least one set of files in .\Generated.

5.3 Minor Program Glitches

Look for the appearance of files of the form fort.<number> files in the folder containing the Read_CA_Database.exe. These should never appear, as they are due to complex program execution flow disruptions that indicate a bug. If these do appear, please notify me at jkbeard@ieee.org with the version of the program (3.0 for the current release) and the name of the file, in particular the <number> portion as that identifies where the error occurs. The file itself will contain text that should be in an output CSV or text file and is not important in finding and fixing the bug.

5.4 Request Features and Improvements

We have indicated that we are considering simplifications such as using just one mode to combine the functions of the Selection and Analysis modes (see screen shots of the mode dialogs shown in Figure 12 and Figure 13). We also have indicated that we are considering elimination of one-click definitions of central 3X3 and 5X5 areas of the DAF as described in Sections 4.4.3.2 and 4.4.3.3 and using the more general selection of an area of the DAF described in Section 4.4.3.4. If you have an opinion on these

changes, or if you have a suggestion for another simplification or change, we are interested in hearing from you.

The utility is a bit of a “Swiss Army Knife” utility that supports research objectives of a few areas of the literature. If you would like to see a new filtering or selection criteria or other feature like the output of a DAF, please let us know. We are considering presentation of the Costas array matrix in the same format that we now present the DAF, possibly in the same CSV file as the DAF, for example. To do this properly, we need to define the sign convention for Costas array matrices – positive upward or positive downward – which will be set by the user with a check box. If you have an opinion on this, or would like to see another new function in the utility, please let us know at jkbeard@ieee.org.

¹ J. K. Beard and K. Erickson and M. Monteleone and M. Wright and J. C. Russo, "Combinatoric collaboration on Costas arrays and radar applications", Proceedings of IEEE Radar Conference pp 260-265 (2004).

² J. K. Beard and J. C. Russo and K. G. Erickson and M. C. Monteleone and M. T. Wright, *Costas array generation and search methodology*, IEEE Transactions on Aerospace and Electronic Systems, vol. 43 no. 2, April 2007, pp 522-538 (2007).

³ K. Drakakis and S. Rickard and J. K. Beard and R. Caballero and F. Iorio and G. O'Brien and J. Walsh, *Results of the enumeration of Costas arrays of order 27*, IEEE Trans. Inf. Theory, vol.54 pp 4684-4687 (2008).

⁴ K. Drakakis and F. Iorio and S. Rickard, *The enumeration of Costas arrays of order 28*, Proceedings of Information Theory Workshop, pp 1-5, August/September 2010.

⁵ K. Drakakis and F. Iorio and S. Rickard and J. Walsh, *Results of the enumeration of Costas arrays of order 29*, Advances in Mathematics Communications, vol. 5, pp 547-553 (2011).

⁶ Jon C. Russo, Keith G. Erickson, and James K. Beard, *Costas Array Search Technique that Maximizes Backtrack and Symmetry Exploitation*, 44th Annual Conference on Information Sciences and Systems (CISS) 2010, May 13, 2010, pp 1-8, DOI: 10.1109/CISS.2010.5464772

⁷ James K Beard, *Generating Costas Arrays to Order 200*, 2006 40th Annual Conference on Information Sciences and Systems (CISS), March 23, 2006, DOI: 10.1109/CISS.2006.286635

⁸ Herbert Taylor, *Costas Arrays*, Chapter IV.7 in *The CRC Handbook of Combinatorial Designs*, Charles J. Colbourn and Jeffrey H. Dinitz, editors, CRC Press (1996), ISBN 0-8493-8948-8, pp. 256-260; see enumeration table on p. 259.

⁹ Curtis P. Brown, Michal Cenkli, Richard A Games, Joseph J. Rushanan, and Oscar Moreno, *New Enumeration Results for Costas Arrays*, IEEE Symposium on Information Theory, January 1993, p. 405.

¹⁰ Bill Correll and James K Beard, *Selecting Appropriate Costas Arrays for Target Detection*, 2017 IEEE Radar Conference, May 8-12, 2017, paper 5351 in Waveform Design 2, Grand Ballroom 1 (Grand Level)

¹¹ James K Beard, *Costas array generator polynomials in finite fields*, 42nd Annual Conference on Information Sciences and Systems (CISS), March 20, 2008, DOI: 10.1109/CISS.2008.4558709