

# How to use generateTemplate

By KGWG DAS team

Hyung Won Lee, Chunglee Kim, Yeong-Bok Bae, Jeongcho Kim

Date 11 Dec. 2017

# Content

- A single waveform generation function
- How to use it
- Example run
- Sample output

# A single waveform generation function

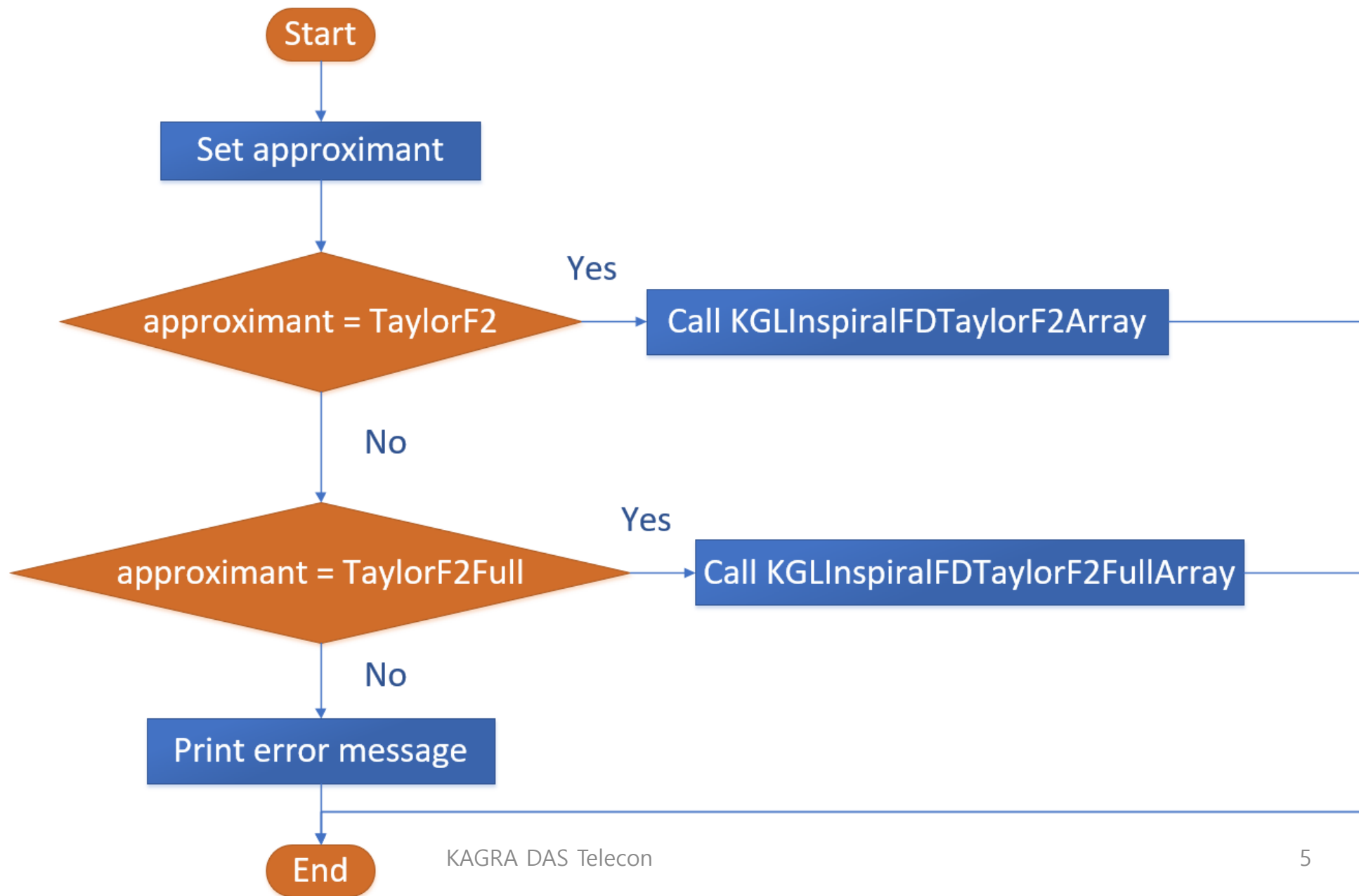
- **`/kagali/waveform/src/KGLWaveforms.c`**
- ```
void KGLDefaultFDWaveformArray( //begin{proto}
    KGLStatus *status,    /**< [in, out] kgl status pointer */
    KGLWaveformModelParams * params, /**< [in] waveform model parameters to print */
    double complex *hp, /**< [out] calculated h_plus value array */
    double complex *hc, /**< [out] calculated h_cross value array */
    double *f, /**< [in] the given frequency value array */
    int n_start, /**< [in] start index of frequency values */
    int n_end, /**< [in] end index of frequency values */
    int N, /**< [in] number of frequency values */
    KGLFDWaveform func /**< [in] actual waveform function pointer for a single frequency */
) //end{proto}
```

# What it does

- Calculate a FD waveform for given approximant and frequency range
- $f$  : frequency array to be calculated for  $f[0]$  to  $f[N_h-1]$   
actual waveform is calculated from  $f[n_{start}]$  to  $f[n_{end}]$
- $params$  : `KGLWaveformModelParams` pointer containing all necessary information to calculate waveform
- $N$  : total size of array for  $hp$ , and  $hc(=n_{end}-n_{start}+1)$
- $hp, hc$  : complex double array for output waveform

# Algorithm for KGLDefaultFDWaveformArray

approximant=params->variables[KGLWAVEFORM\_MODEL\_PARAMS\_approximant].value.int\_value



```

/**
 * KGLDefaultFDWaveformArray
 *
 * default waveform function prototype, a single function for all waveforms
 */
void KGLDefaultFDWaveformArray( //begin{proto}
    KGLStatus *status,    /**< [in, out] kgl status pointer */
    KGLWaveformModelParams * params, /**< [in] waveform model parameters to print */
    double complex *hp, /**< [out] calculated h_plus value array */
    double complex *hc, /**< [out] calculated h_cross value array */
    double *f, /**< [in] the given frequency value array */
    int n_start, /**< [in] start index of frequency values */
    int n_end, /**< [in] end index of frequency values */
    int N, /**< [in] number of frequency values */
    KGLFDWaveform func /**< [in] actual waveform function pinter for a single frequency */
) //end{proto}
{
    KGLApproximants approximant;
    switch((approximant=params->variables[KGLWAVEFORM_MODEL_PARAMS_approximant].value.int_value))
    {
        case TaylorF2:
            KGLInspiralFDTaylorF2Array(status, params, hp, hc, f, n_start, n_end, N, KGLInspiralFDTaylorF2);
            break;
        case TaylorF2Full:
            KGLInspiralFDTaylorF2FullArray(status, params, hp, hc, f, n_start, n_end, N, KGLInspiralFDTaylorF2Full);
            break;
        default:
            fprintf(stderr, "[ERROR-WAVEFORM]Approximant %d is not supported.\n", approximant);
            fprintf(stderr, "                If this is a new waveform, then add a new number for this in KGLApproximant
s.h.in ");
            fprintf(stderr, "and add appropriate approximant name in KGLApproximants.c\n");
            break;
    }
}

```

# General procedure to use it

1. Include headers
2. Set appropriate values for KGLWaveformModelParams
3. Allocate frequency array and assign values
4. Allocate arrays for hp and hc
5. Call KGLDefaultFDWaveformArray function
6. Save result (hp and hc) or do something you want

# Headers needed

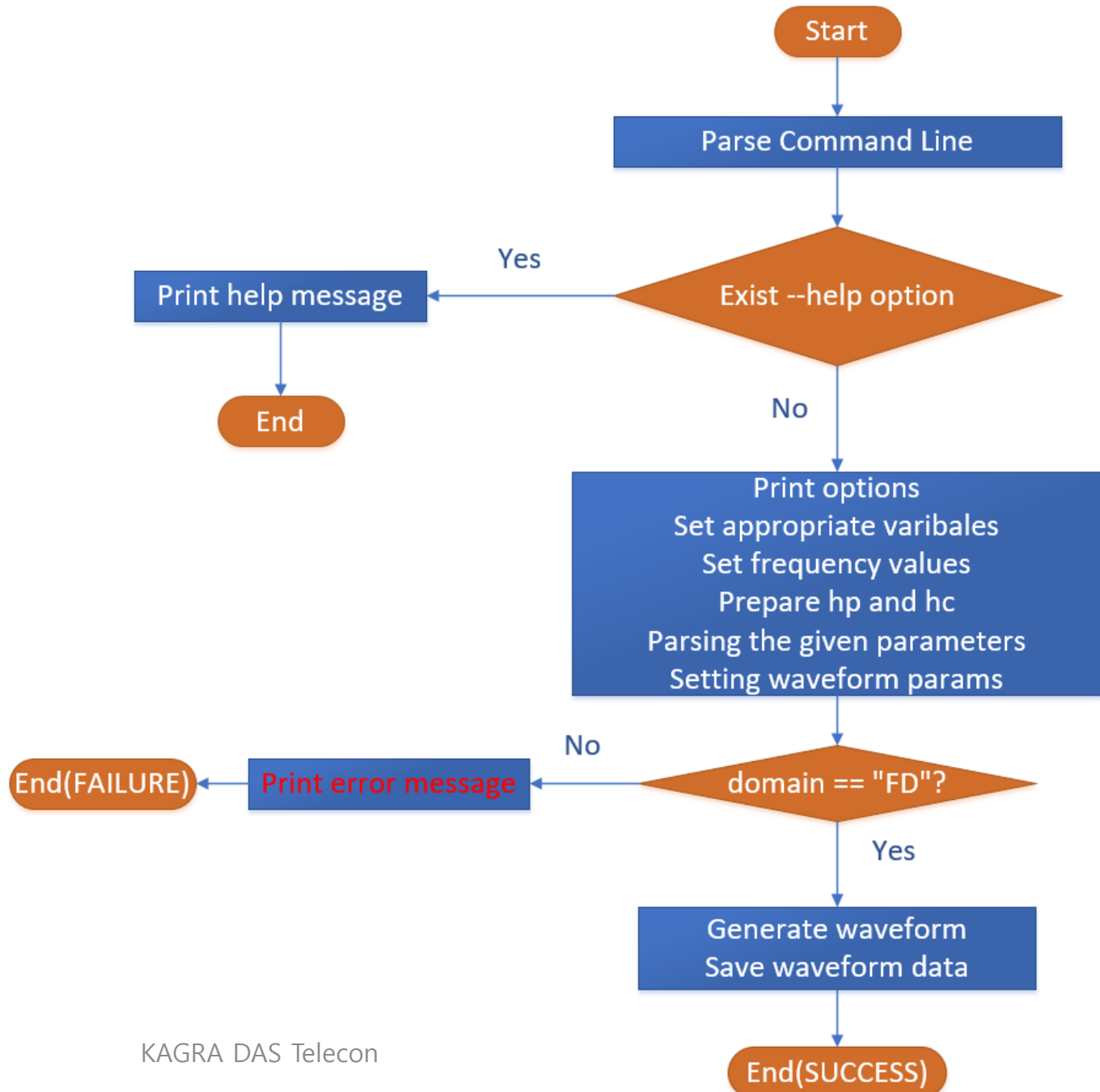
```
#include <sys/types.h>
#include <unistd.h>
#include <stdio.h>
#include <string.h>
#include <complex.h>
#include <gsl/gsl_rng.h>
#include <kagali/KGLStdlib.h>
#include <kagali/KGLProcessParamsTable.h>
#include <kagali/KGLCommandUtil.h>
#include <kagali/KGLApproximants.h>
#include <kagali/KGLWaveformModelParams.h>
#include <kagali/KGLWaveforms.h>
#include <kagali/KGLInference.h>
#include <kagali/KGLDetectors.h>
#include <kagali/KGLInferenceIFOData.h>
#include <kagali/KGLInferenceRunState.h>
#include <kagali/KGLInferenceReadData.h>
#include <kagali/KGLInferenceMCMC.h>
```



# generateTemplate application

- Location : kagali-v0r4a/kagaliapps/cbc/mpisrc
- Purpose : show how to generate a given waveform
- Branch : mcmc

# Algorithm



# Options of generateTemplate application

```
char help[] ="\n\
-----\n\
--- Template waveform generation Parameters ----\n\
-----\n\
(--domain TD/FD)          domain of template waveform time(TD) or frequency(FD) domain[FD]\n\
(--approximant APPROX)    approximant name[TaylorF2]\n\
(--ampOrder AMPORDER)    amplitude order of PN correction[-1]\n\
(--phaseOrder PHASEORDER) phase order of PN correction[-1]\n\
(--f_start F_START)      start frequency[0]\n\
(--f_end F_END)          end frequency[2048]\n\
(--delta_f DELTA_F)      frequency bin size for FD waveform in Hz[1/64]\n\
(--srate SRATE)          sampling rate for TD waveform in Hz[4096]\n\
(--outfile file)         name for output file name[waveform.dat]\n\
(--m1 m1)                mass of the first object in solar mass[1.4]\n\
(--m2 m2)                mass of the second object in solar mass[1.4]\n\
(--distance d)           distance of the source in Mpc[10]\n\
(--eccentricity ECC)      eccentricity of the source at a reference frequency[0]\n\
(--eccOrder ECCORDER)    eccentricity phase correction order of PN correction[-1]\n\
(--f_ecc F_ECC)          reference frequency eccentricity phase correction[10]\n\
(--lambda1 LAMBDA1)      tidal parameter of mass 1 of the source[0]\n\
(--lambda2 LAMBDA2)      tidal parameter of mass 2 of the source[0]\n\
\n";
```

One can see this option by generateTemplate --help

# Part of main code of generateTemplate application

```
KGLWaveformModelParams modelParams;
double complex *hp;
double complex *hc;
int N_h = 0;
N_h = (int)((f_end + 0.5*delta_f)/delta_f) + 1; /* from zero frequency */
f = (double *)malloc(sizeof(double)*N_h);
fprintf(stdout, "f_start = %f, f_end = %f, N = %d\n", f_start, f_end, N);
for(int i=0; i<N_h; i++)
    f[i] = i*delta_f;
fprintf(stdout, "n_start = %d, n_end = %d, N_h = %d\n", n_start, n_end, N_h);
hp = (double complex *)malloc(sizeof(double complex)*N_h);
hc = (double complex *)malloc(sizeof(double complex)*N_h);
modelParams.variables[KGLWAVEFORM_MODEL_PARAMS_approximant].value.int_value = approximant;
KGLSetComponentMasses(status, &modelParams, m1, m2);
KGLSetDistance(status, &modelParams, distance);
if(!strncmp(domain, "FD", 2))
{
    KGLDefaultFDWaveformArray(status, &modelParams, hp, hc, f, n_start, n_end, N, NULL);
    SaveFDWaveform(hp, hc, n_start, n_end, delta_f, filename);
}
else
{
    fprintf(stdout, "TD waveform is not implemented yet.\n");
}
free(hp);
free(hc);
free(f);
```

Prepare variables

Set model params

Calculate waveform and save

# Parse command line options

```
int main(int argc, char *argv[])
{
    KGLStatus *status = NULL;

    status = KGLCreateStatus();
    KGLAbortIfError(status);

    KGLProcessParamsTable *cmdLine = NULL;
    KGLParseCommandLine(status, argc, argv, &cmdLine);

    KGLProcessParamsPair *param_pair = NULL;
    KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--help");
    if(param_pair)
    {
        fprintf(stdout, "%s", help);
        exit(1);
    }
    fprintf(stdout, " ===== Template waveform generation for KAGALI =====\n");
    KGLPrintProcessParamsTable(status, cmdLine);
}
```

# Set appropriate variables 1

```
char domain[] = "FD";
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--domain");
if(param_pair)
{
    strcpy(domain, param_pair->value);
}
KGLApproximants approximant = TaylorF2;
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--approximant");
if(param_pair)
{
    approximant = KGLGetApproximantByName(status, param_pair->value);
    if(approximant < 0) /* not existing approximant */
        approximant = TaylorF2;
}
double f_start = 0.0;
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--f_start");
if(param_pair)
{
    f_start = atof(param_pair->value);
}
double f_end = 2048.0;
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--f_end");
if(param_pair)
{
    f_end = atof(param_pair->value);
}
double delta_f = 1.0/64.0;
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--delta_f");
if(param_pair)
{
    delta_f = atof(param_pair->value);
}
double *f = NULL;
int N = (int)((f_end - f_start + 0.5*delta_f)/delta_f) + 1;
int n_start = (int)((f_start + 0.5*delta_f)/delta_f);
int n_end = (int)((f_end + 0.5*delta_f)/delta_f);
```

## Set appropriate variables 2

```
char filename[256]={0};
strncpy(filename, "waveform.dat", 12);
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--outfile");
if(param_pair)
{
    strncpy(filename, param_pair->value, strlen(param_pair->value));
}
double m1 = 1.4;
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--m1");
if(param_pair)
{
    m1 = atof(param_pair->value);
}
double m2 = 1.4;
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--m2");
if(param_pair)
{
    m2 = atof(param_pair->value);
}
double distance = 10.0;
KGLGetProcessParamsPair(status, cmdLine, &param_pair, "--distance");
if(param_pair)
{
    distance = atof(param_pair->value);
}
```

# Allocate memories

```
KGLWaveformModelParams modelParams;  
double complex *hp;  
double complex *hc;  
int N_h = 0;  
N_h = (int)((f_end + 0.5*delta_f)/delta_f) + 1; /* from zero frequency */  
f = (double *)malloc(sizeof(double)*N_h);  
fprintf(stdout, "f_start = %f, f_end = %f, N = %d\n", f_start, f_end, N);  
for(int i=0; i<N_h; i++)  
    f[i] = i*delta_f;  
fprintf(stdout, "n_start = %d, n_end = %d, N_h = %d\n", n_start, n_end, N_h);  
hp = (double complex *)malloc(sizeof(double complex)*N_h);  
hc = (double complex *)malloc(sizeof(double complex)*N_h);  
modelParams.variables[KGLWAVEFORM_MODEL_PARAMS_approximant].value.int_value = approximant;  
KGLSetComponentMasses(status, &modelParams, m1, m2);  
KGLSetDistance(status, &modelParams, distance);
```

KGLWaveformModelParams structure has values in SI unit

KGLSetComponentMasses : set masses given in solar unit and set also chirp mass and eta

KGLSetDistance : set distance given in Mpc unit

defined in `/kagali/waveform/src/KGLWaveformModelParams.c`



## Generate waveform and save

```
if(!strncmp(domain, "FD", 2))
{
    KGLDefaultFDWaveformArray(status, &modelParams, hp, hc, f, n_start, n_end, N, NULL);
    SaveFDWaveform(hp, hc, n_start, n_end, delta_f, filename);
}
else
{
    fprintf(stdout, "TD waveform is not implemented yet.\n");
}
free(hp);
free(hc);
free(f);

return 0;
```

# Example run

- run.sh script

```
#!/bin/bash
masterdir=/home/hwlee/projects/KGL/kagali/build/mcmc
source $masterdir/etc/kagali-user-env.sh
$masterdir/bin/generateTemplate --approximanr TaylorF2 --m1 1.4 --m2 1.4 --f_start 10.0
```

# Sample run output

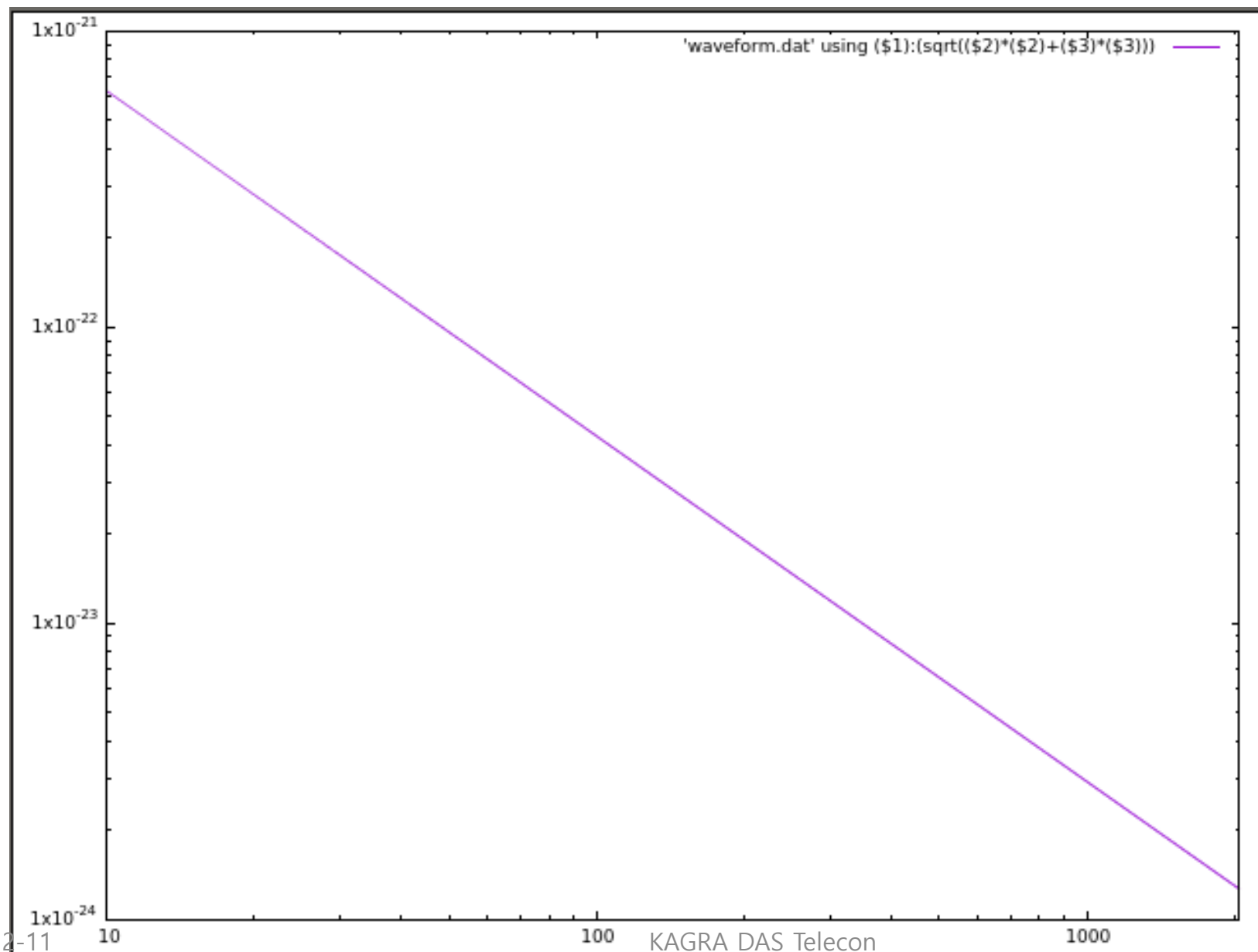
```
hwlee@kaeri:~/projects/KGL/kagali/test_run$ ./run.sh
===== Template waveform generation for KAGALI =====
KGLProcessParamsTable :
  Program : /home/hwlee/projects/KGL/kagali/build/mcmc/bin/generateTemplate, Process Id : 26726
  Pairs :
    Parameter Name : --approximanr, Parameter Type : 0, Value : TaylorF2
    Parameter Name : --m1, Parameter Type : 0, Value : 1.4
    Parameter Name : --m2, Parameter Type : 0, Value : 1.4
    Parameter Name : --f_start, Parameter Type : 0, Value : 10.0
f_start = 10.000000, f_end = 2048.000000, N = 130433
n_start = 640, n_end = 131072, N_h = 131073
```

# Sample output(waveform.dat)

hwlee@kaeri: ~/projects/KGL/kagali/test\_run

| #                      | frequency               | hp Real                 | hp Image                | hc Real                 | hc Image |
|------------------------|-------------------------|-------------------------|-------------------------|-------------------------|----------|
| 1.0000000000000000e+01 | -2.8054993604382618e-23 | 6.2655278217875380e-22  | 6.2655278217875380e-22  | 2.8054993604382618e-23  |          |
| 1.0015625000000000e+01 | 5.1020705909913042e-22  | 3.6278622648015156e-22  | 3.6278622648015156e-22  | -5.1020705909913042e-22 |          |
| 1.0031250000000000e+01 | 4.3911762106362892e-22  | -4.4460976247974157e-22 | -4.4460976247974157e-22 | -4.3911762106362892e-22 |          |
| 1.0046875000000000e+01 | -5.3809083256957826e-22 | -3.1550719121425057e-22 | -3.1550719121425057e-22 | 5.3809083256957826e-22  |          |
| 1.0062500000000000e+01 | 7.8341390200821081e-23  | 6.1768995756266618e-22  | 6.1768995756266618e-22  | -7.8341390200821081e-23 |          |
| 1.0078125000000000e+01 | 2.3218538917570505e-22  | -5.7651296413626210e-22 | -5.7651296413626210e-22 | -2.3218538917570505e-22 |          |
| 1.0093750000000000e+01 | -2.9136247691181049e-22 | 5.4771468589641994e-22  | 5.4771468589641994e-22  | 2.9136247691181049e-22  |          |
| 1.0109375000000000e+01 | 1.1949902960912904e-22  | -6.0763216068759869e-22 | -6.0763216068759869e-22 | -1.1949902960912904e-22 |          |
| 1.0125000000000000e+01 | 2.9580517701300432e-22  | 5.4278602979206579e-22  | 5.4278602979206579e-22  | -2.9580517701300432e-22 |          |
| 1.0140625000000000e+01 | -6.1695024682428838e-22 | 1.0833458925195527e-23  | 1.0833458925195527e-23  | 6.1695024682428838e-22  |          |
| 1.0156250000000000e+01 | 4.3005783769718668e-23  | -6.1443477946426500e-22 | -6.1443477946426500e-22 | -4.3005783769718668e-23 |          |
| 1.0171875000000000e+01 | 5.9801324385207044e-22  | -1.4283334555767606e-22 | -1.4283334555767606e-22 | -5.9801324385207044e-22 |          |
| 1.0187500000000000e+01 | 4.9137796369923356e-22  | 3.6771925109796924e-22  | 3.6771925109796924e-22  | -4.9137796369923356e-22 |          |
| 1.0203125000000000e+01 | 2.6030901509019241e-22  | 5.5458489481687678e-22  | 5.5458489481687678e-22  | -2.6030901509019241e-22 |          |
| 1.0218750000000000e+01 | 2.0080477131933903e-22  | 5.7763735751885130e-22  | 5.7763735751885130e-22  | -2.0080477131933903e-22 |          |
| 1.0234375000000000e+01 | 3.5138147677221552e-22  | 4.9918695459922132e-22  | 4.9918695459922132e-22  | -3.5138147677221552e-22 |          |
| 1.0250000000000000e+01 | 5.8255918067301750e-22  | 1.7876576757233014e-22  | 1.7876576757233014e-22  | -5.8255918067301750e-22 |          |
| 1.0265625000000000e+01 | 4.4888194127974322e-22  | -4.1051181908569597e-22 | -4.1051181908569597e-22 | -4.4888194127974322e-22 |          |
| 1.0281250000000000e+01 | -3.3540598384627256e-22 | -5.0616894040317503e-22 | -5.0616894040317503e-22 | 3.3540598384627256e-22  |          |
| 1.0296875000000000e+01 | -4.2044865167784035e-22 | 4.3660387582509148e-22  | 4.3660387582509148e-22  | 4.2044865167784035e-22  |          |
| 1.0312500000000000e+01 | 5.9582488345325182e-22  | 1.0533349037259995e-22  | 1.0533349037259995e-22  | -5.9582488345325182e-22 |          |
| 1.0328125000000000e+01 | -4.2780170589353759e-22 | -4.2637670683218309e-22 | -4.2637670683218309e-22 | 4.2780170589353759e-22  |          |
| 1.0343750000000000e+01 | 3.1346678090336607e-22  | 5.1503924790441821e-22  | 5.1503924790441821e-22  | -3.1346678090336607e-22 |          |

```
gnuplot> set logscale xy
gnuplot> plot 'waveform.dat' using ($1):(sqrt(($2)*($2)+($3)*($3))) with lines
```



# Repository Snapshot

KAGALI-0.4 alpha

Search:  KAGALI-0.4 alpha

Overview Activity Issues New issue Documents Files Repository Code reviews Settings

## Revisions

Revision:

| #        |     | DATE                | AUTHOR              | COMMENT                                                                                                                                                    | CODE REVIEWS      |
|----------|-----|---------------------|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------|
| 66d01a6c | ●   | 12/01/2017 01:28 am | HyungWon Lee        | change 2*KGL_PI to KGL_TWOPi                                                                                                                               | No reviews:Assign |
| 58393c5e | ○ ● | 11/28/2017 01:56 pm | HyungWon Lee        | change the waveform output format                                                                                                                          | No reviews:Assign |
| ac44ef14 | ○ ○ | 11/26/2017 11:29 pm | HyungWon Lee        | add a new approximant TaylorF2Full containing all extra terms                                                                                              | No reviews:Assign |
| a3e07dd8 | ○ ○ | 11/26/2017 11:03 pm | HyungWon Lee        | implement a simple TaylorF2 waveform                                                                                                                       | No reviews:Assign |
| 4590fb17 | ○ ○ | 11/26/2017 02:01 am | HyungWon Lee        | works on generateTemplate program, added some setting functions                                                                                            | No reviews:Assign |
| b44970b9 | ○ ○ | 11/25/2017 01:25 am | HyungWon Lee        | works on generateTemplate program                                                                                                                          | No reviews:Assign |
| 1db51ce0 | ○ ○ | 11/25/2017 12:59 am | HyungWon Lee        | working on generateTemplate program                                                                                                                        | No reviews:Assign |
| ac0bd40c | ○ ○ | 11/24/2017 01:19 pm | HyungWon Lee        | working on generateTemplate program                                                                                                                        | No reviews:Assign |
| 90d0135a | ○ ○ | 11/24/2017 11:46 am | HyungWon Lee        | making template generattion program in progress                                                                                                            | No reviews:Assign |
| b200ca14 | ○ ○ | 11/24/2017 01:19 am | HyungWon Lee        | modify perl script file                                                                                                                                    | No reviews:Assign |
| 9462d2a6 | ○ ○ | 11/24/2017 01:06 am | HyungWon Lee        | fix bugs                                                                                                                                                   | No reviews:Assign |
| da5d1fa9 | ○ ○ | 11/24/2017 12:17 am | HyungWon Lee        | add generateTemplate program                                                                                                                               | No reviews:Assign |
| bb4b7020 | ○ ○ | 11/21/2017 05:03 pm | HyungWon Lee        | add KGLApproximants enumerator                                                                                                                             | No reviews:Assign |
| 5ea67038 | ○ ○ | 11/19/2017 02:19 pm | HyungWon Lee        | add char_value memebr in WaveformModelParamsValue structure                                                                                                | No reviews:Assign |
| b5cfa309 | ○ ○ | 11/17/2017 05:42 pm | Kim Jeongcho        | Merge branch 'mcmc' of <a href="https://vt001.resceu.s.u-tokyo.ac.jp/git/kagali-v0r4a">https://vt001.resceu.s.u-tokyo.ac.jp/git/kagali-v0r4a</a> into mcmc | No reviews:Assign |
| c474f7b5 | ○ ○ | 11/17/2017 05:34 pm | Kim Jeongcho        | add KGLGenerateSimulation structure                                                                                                                        | No reviews:Assign |
| c31e2cd1 | ○ ○ | 08/28/2017 06:02 am | HyungWon Lee        | Merge branch 'mcmc' of <a href="https://vt001.resceu.s.u-tokyo.ac.jp/git/kagali-v0r4a">https://vt001.resceu.s.u-tokyo.ac.jp/git/kagali-v0r4a</a> into mcmc | No reviews:Assign |
| 58373351 | ○ ○ | 08/09/2017 12:42 pm | Oohara Ken-ichi     | Invalid symbolic link KGLInferencetest.h has been removed.                                                                                                 | No reviews:Assign |
| 325e57dd | ○ ○ | 08/07/2017 02:53 pm | Kim Jeongcho        | Merge branch 'master' into mcmc                                                                                                                            | No reviews:Assign |
| 8103d19f | ○ ○ | 06/02/2017 01:01 am | HyungWon Lee        | Merge branch 'master' into mcmc                                                                                                                            | No reviews:Assign |
| 74c80301 | ○ ○ | 04/13/2017 03:23 pm | Yuzurihara Hirotaka | KGLInspiralParameters.c : Added comment.                                                                                                                   | No reviews:Assign |
| e1dd171f | ○ ○ | 04/08/2017 06:10 am | Yuzurihara Hirotaka | cbc/src/KGLInspiralTaylorF2.c : Added assert                                                                                                               | No reviews:Assign |
| 427b2953 | ○ ○ | 04/08/2017 04:33 am | Yuzurihara Hirotaka | KGLEstimatePSD : Fixed typo                                                                                                                                | No reviews:Assign |
| 8e118f93 | ○ ○ | 04/08/2017 04:32 am | Yuzurihara Hirotaka | cbc/src/KGLInspiralTaylorF2 : Fixed typo                                                                                                                   | No reviews:Assign |
| 9052583e | ○ ○ | 04/08/2017 04:31 am | Yuzurihara Hirotaka | cbc/src/KGLInspiralTaylorF2 : Added comment                                                                                                                | No reviews:Assign |

# What has been done

- generateTemplate
  - Make a simple application to generate data for the given waveform

# What to do

- generateTemplate
  - Add options, --strain, --ifo K1/L1/H1/V1, --gps GPS
  - Generate strain data for given template at given observatory and time (Jeongcho will implement)
- inferenceMCMC
  - Parallel tempering MCMC parameter estimation for CBC signal
  - Implementation for likelihood calculation
  - Likelihood function : Young-bok Bae will implement



# Discussions