A Concise User Guide for QWSIM

November 20, 2014

Contents

1 Before Starting

- This simulator was tested in both Linux and Mac OSX with gcc compiler. Due to the similarity of both systems, this users guide focusses on Mac OSX only.

- For Mac OSX users, Xcode must be installed from the App Store before getting started.

2 Folder "qwsim" and Command Line

1. Download the zip file "qwsim.zip".

2. Decompress "qwsim.zip".

3. Put the decompressed folder "qwsim" on Desktop. "qwsim" content are depicted in the following figure.
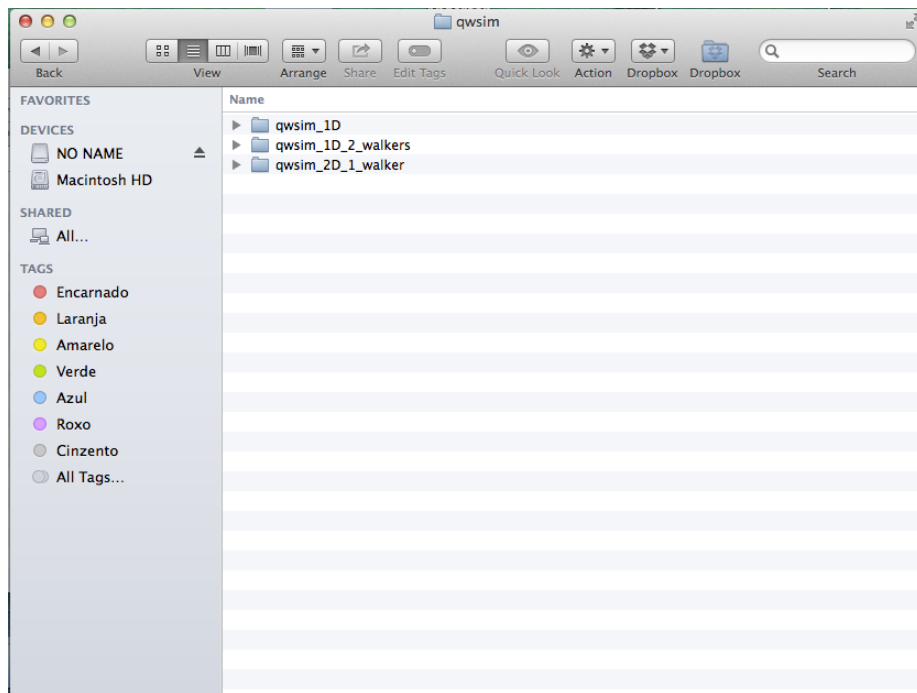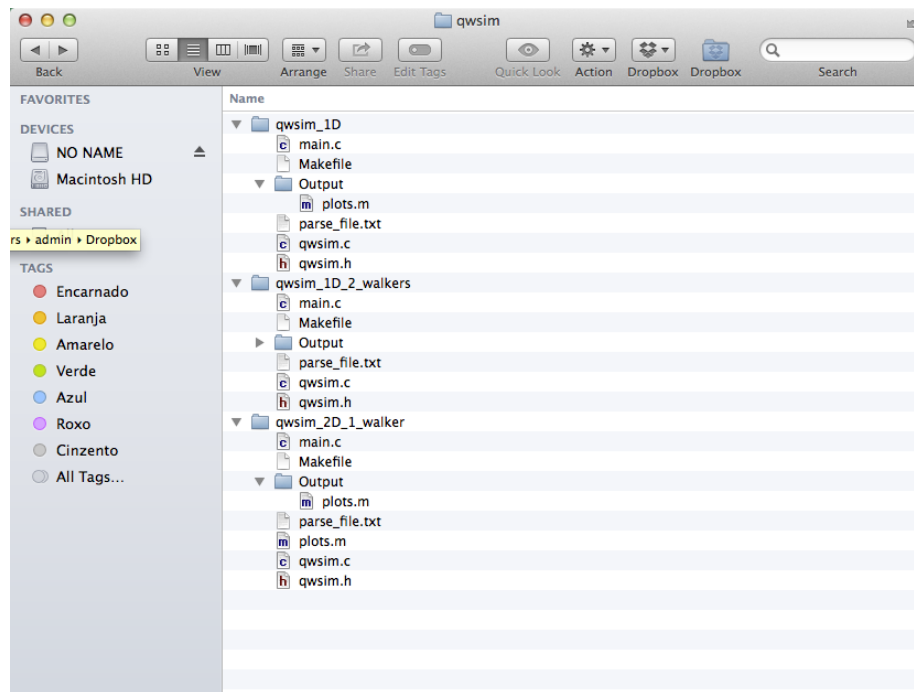

Figure 1: Content of folder "qwsim"

Figure 2: All the contents of the simulator.

4. Open command line/terminal;
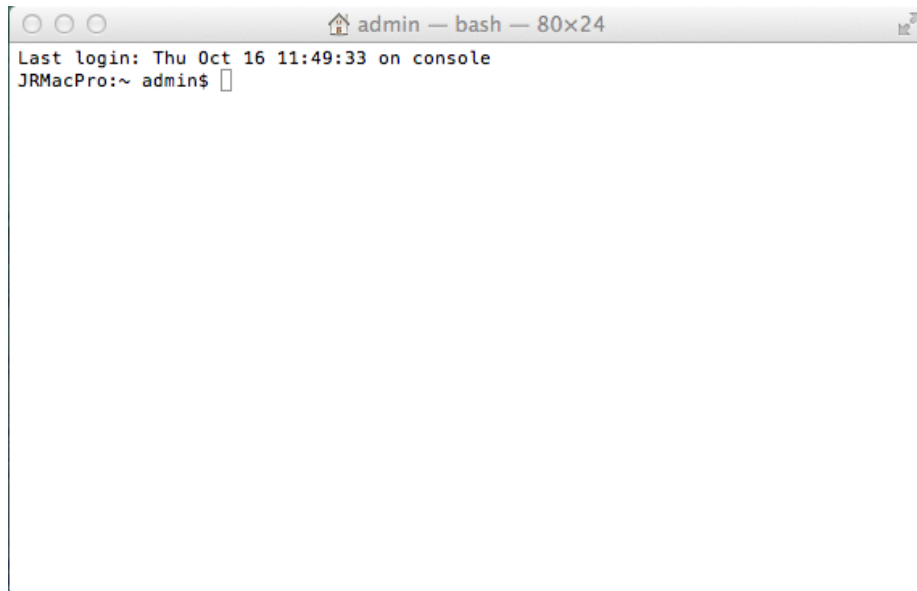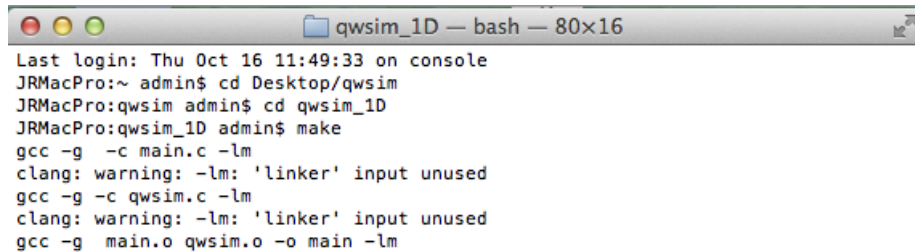
Figure 3: terminal/Command line

3 Compiling the programs

1. To compile the program for 1-particle quantum walk on the line write the following commands in the command line:

   (a) "cd Desktop/qwsim" and press Enter;

   (b) "cd qwsim_1D" and press Enter;

   (c) "make" and press Enter;

Figure 4: Command line instructions to compile 1-particle quantum walk program.

The generated files are "main.o", "qwsim.o" and "main" and are placed in folder "qwsim_1D".

Figure 5: Files generated after compilation of 1-particle quantum walk program.

2. To compile the program for 2-particle quantum walk on the line write in the command line

5

(a) "cd Desktop/qwsim" and press Enter;

(b) "cd qwsim_1D_2_walkers" and press Enter;

(c) "make" and press Enter;

Figure 6: Command line instructions to compile 2-particle quantum walk program.

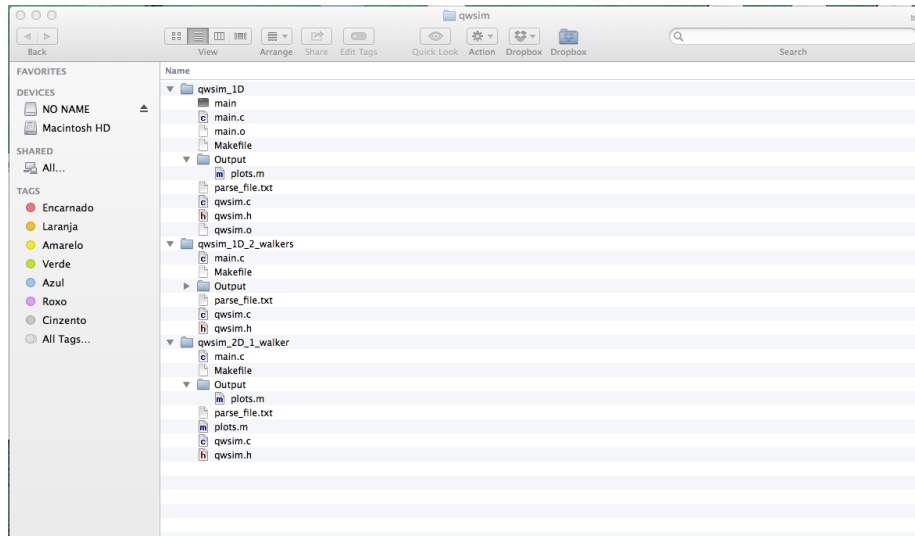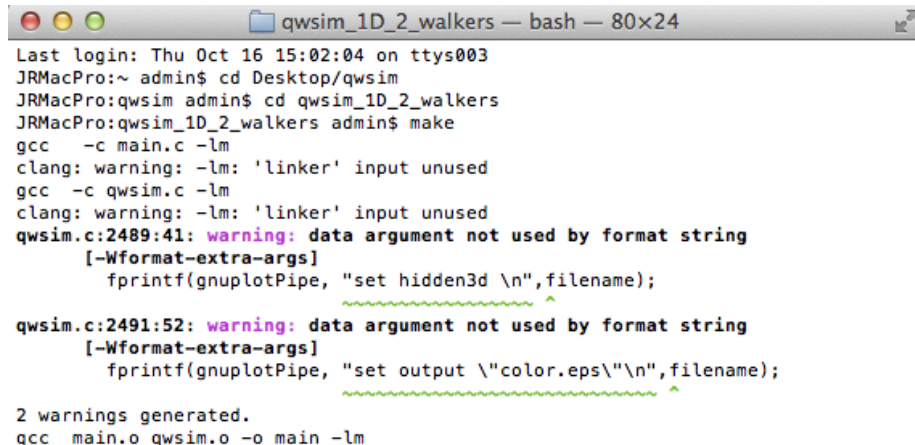The generated files are "main.o", "qwsim.o" and "main" and are placed in folder "qwsim_1D_2_walkers".

Figure 7: Files generated after compilation of 2-particle quantum walk program.

3. To compile the program for 1-particle quantum walk on a 2D lattice write in the command line

   (a) "cd Desktop/qwsim" and press Enter;

   (b) "cd qwsim_2D_1_walker" and press Enter;

   (c) "make" and press Enter;

```
Last login: Thu Oct 16 15:19:19 on ttys000
JRMacPro:~ admin$ cd Desktop/qwsim
JRMacPro:qwsim admin$ cd qwsim_2D_1_walker
JRMacPro:qwsim_2D_1_walker admin$ make
gcc   -c main.c -lm
clang: warning: -lm: 'linker' input unused
gcc   -c qwsim.c -lm
clang: warning: -lm: 'linker' input unused
qwsim.c:2154:40: warning: data argument not used by format string
      [-Wformat-extra-args]
        fprintf(gnuplotPipe, "set hidden3d\n",filename);
                             ~~~~~~~~~~~~~~~~~ ^
qwsim.c:2156:52: warning: data argument not used by format string
      [-Wformat-extra-args]
        fprintf(gnuplotPipe, "set output \"color.eps\"\n",filename);
                             ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~ ^
2 warnings generated.
gcc   main.o qwsim.o -o main -lm
```

Figure 8: Command line instructions to compile 1-particle quantum walk program on a 2D lattice.

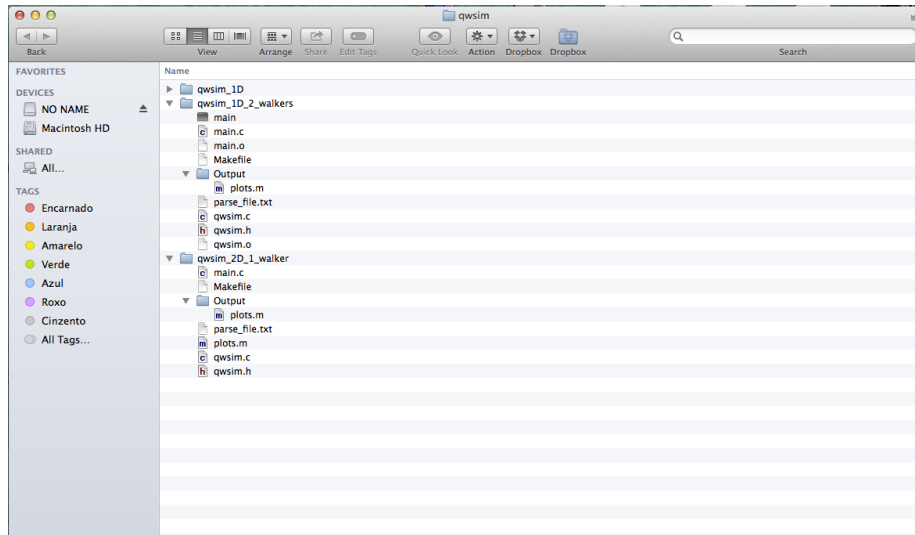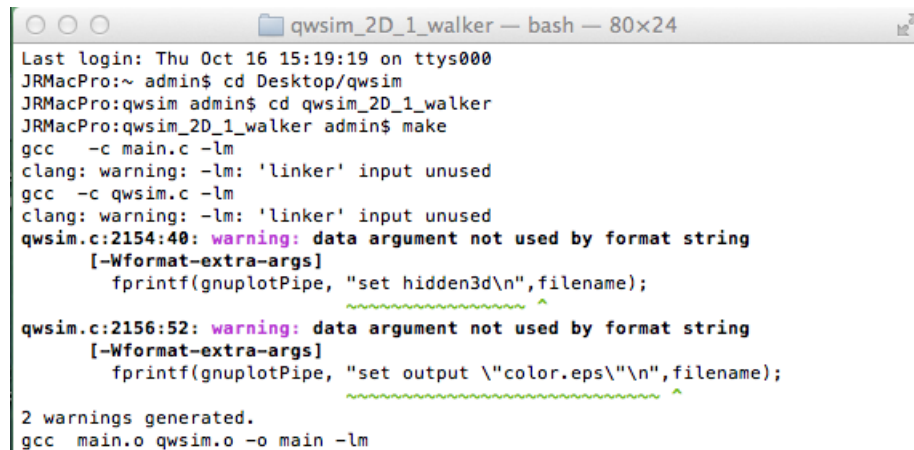The generated files are "main.o", "qwsim.o" and "main" and are placed in folder "qwsim_2D_1_walker".
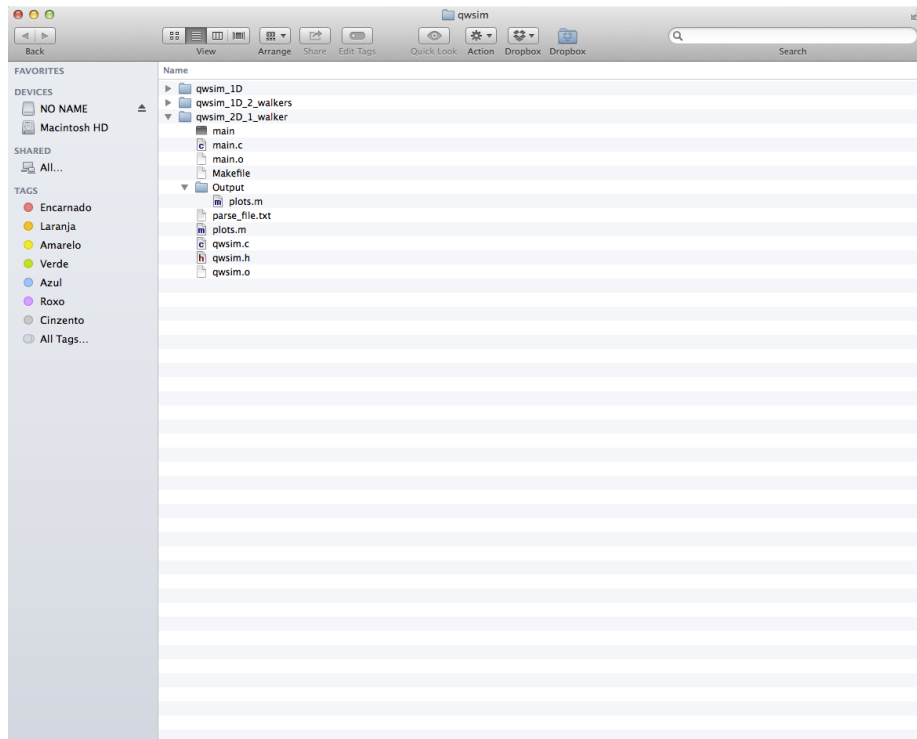
Figure 9: Files generated after compilation of 1-particle quantum walk program on a 2D lattice.

4 Examples of Configurations

The following subsections presents the input files for the examples in the folder
"usersguide_and_examples".

4.1 Examples of Configurations for Particle Quantum Walk on the Line

4.1.1 folder ex1

- a line of size 201 (-100 to 100);

- a number of steps 100;

- circular boundary condition;

- initial state $\frac{1}{\sqrt{50}}(5\left|-5\right\rangle\left|L\right\rangle + i5\left|5\right\rangle\left|R\right\rangle)$;

- hadamard coin operator in every position;

N 100

Steps 100

points 2

i -5 5

c 0 1

real_value 5 0

complex_value 0 5

index_brokenlink_imp 0.0

fixed_broken_links 0

I_0

coin 0

varying_coin_each_step 0

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

index_coin_imp 0

fixed_coins 0

i

theta_0

theta_1

zeta_0

zeta_0

xi_0

xi_1

ref 0

dim_absorb 0

i

c

gnu_plot 0

_____

4.1.2   folder ex2

- a line of size 201 (-100 to 100);

- a number of steps 100;

- circular boundary condition;

- initial state $\frac{1}{\sqrt{50}}(5\,|-5\rangle\,|L\rangle + i5\,|5\rangle\,|R\rangle)$;

- varying coin operator in every position of parameters $\theta \in [0.70, 0.80], \zeta \in [0.70, 0.80], \xi \in [0.20, 0.30]$; The chosen Hadamard coin will have no effect when this mode is used;

```
N  100

Steps  100

points  2

i  −5  5

c  0  1

real_value  5  0

complex_value  0  5

index_brokenlink_imp  0.0

fixed_broken_links  0

I_0

coin  0

varying_coin_each_step 1

theta_0 0.70

theta_1 0.80

zeta_0 0.70

zeta_1 0.80

xi_0 0.20

xi_1 0.30

index_coin_imp  0

fixed_coins  0

i

theta_0

theta_1

zeta_0
```

zeta_0

xi_0

xi_1

ref 0

dim_absorb 0

i

c

gnu_plot 0

4.1.3 folder ex3

- a line of size 201 (-100 to 100);

- a number of steps 100;

- circular boundary condition;

- initial state $\frac{1}{\sqrt{50}}(5\left|-5\right\rangle\left|L\right\rangle + i5\left|5\right\rangle\left|R\right\rangle)$;

- varying coin operator in every position of parameters $\theta \in [0.70, 0.80], \zeta \in [0.70, 0.80], \xi \in [0.20, 0.30]$; The chosen Hadamard coin will have no effect when this mode is used;

- fixed coin operators are defined in positions 9 with parameters $\theta \in [0, 0.75], \zeta \in [0, 0.75], \xi \in [0.25, 0.25]$ and -5 with parameters $\theta \in [0.75, 1.23], \zeta \in [0.75, 1.23], \xi \in [0.75, 0.95]$. These coin operators will be used only in those positions, leaving the others with the varying coin operator defined for all other nodes;

```
N 100

Steps 100

points 2

i −5 5

c 0 1

real_value 5 0

complex_value 0 5

index_brokenlink_imp 0.0

fixed_broken_links 0

I_0

coin 0

varying_coin_each_step 1

theta_0 0.70

theta_1 0.80

zeta_0 0.70

zeta_1 0.80

xi_0 0.20

xi_1 0.30

index_coin_imp 0
```

fixed_coins 2

i 9 -5

theta_0 0 0.75

theta_1 0.75 1.23

zeta_0 0 0.75

zeta_1 0.75 1.23

xi_0 0.25 0.75

xi_1 0.25 0.95

ref 0

dim_absorb 0

i

c

gnu_plot 0

_____

4.1.4 folder ex4

- a line of size 201 (-100 to 100);

- a number of steps 100;

- circular boundary condition;

- initial state $\frac{1}{\sqrt{50}}(5\left|-5\right\rangle\left|L\right\rangle + i5\left|5\right\rangle\left|R\right\rangle)$;

- varying coin operator in every position of parameters $\theta \in [0.70, 0.80], \zeta \in [0.70, 0.80], \xi \in [0.20, 0.30]$; The chosen Hadamard coin will have no effect when this mode is used;

- fixed coin operators are defined in positions 9 with parameters $\theta \in [0, 0.75], \zeta \in [0, 0.75], \xi \in [0.25, 0.25]$ and -5 with parameters $\theta \in [0.75, 1.23], \zeta \in [0.75, 1.23], \xi \in [0.75, 0.95]$. These coin operators will be used only in those positions, leaving the others with the varying coin operator defined for all other nodes.

- Probability of a coin to be chosen with parameters $\theta, \zeta, \xi \in [0, \pi/2]$, in each position at each step, is set to 0.5. These coin operators has precedence to the afore defined coin operators;

```
N  100

Steps  100

points  2

i  −5  5

c  0  1

real_value  5  0

complex_value  0  5

index_brokenlink_imp  0.0

fixed_broken_links  0

I_0

coin  0

varying_coin_each_step  1

theta_0  0.70  0.20

theta_1  0.80  1.30

zeta_0  0.70  0.20

zeta_1  0.80  1.30

xi_0  0.20  0.90
```

```
xi_1  0.30  1

index_coin_imp 0.5

fixed_coins  2

i  9  −5

theta_0  0  0.75

theta_1  0.75  1.23

zeta_0  0  0.75

zeta_1  0.75  1.23

xi_0  0.25  0.75

xi_1  0.25  0.95

ref  0

dim_absorb  0

i

c

gnu_plot  0
```

4.1.5   folder ex5

- a line of size 201 (-100 to 100);

- a number of steps 100;

- circular boundary condition;

- initial state $\frac{1}{\sqrt{50}}(5\left|-5\right\rangle\left|L\right\rangle + i5\left|5\right\rangle\left|R\right\rangle)$;

- varying coin operator in every position of parameters $\theta \in [0.70, 0.80], \zeta \in [0.70, 0.80], \xi \in [0.20, 0.30]$; The chosen Hadamard coin will have no effect when this mode is used;

- fixed coin operators are defined in positions 9 with parameters $\theta \in [0, 0.75], \zeta \in [0, 0.75], \xi \in [0.25, 0.25]$ and -5 with parameters $\theta \in [0.75, 1.23], \zeta \in [0.75, 1.23], \xi \in [0.75, 0.95]$. These coin operators will be used only in those positions, leaving the others with the varying coin operator defined for all other nodes.

- Probability of a coin to be chosen with parameters $\theta, \zeta, \xi \in [0, \pi/2]$, in each position at each step, is set to 0.5. These coin operators has precedence to the afore defined coin operators.

- 2 Fixed broken links are set between nodes 9 and 10 and between -5 and -4.

```
N 100

Steps 100

points 2

i −5 5

c 0 1

real_value 5 0

complex_value 0 5

index_brokenlink_imp 0.0

fixed_broken_links 2

I_0 -50 50

coin 0

varying_coin_each_step 1

theta_0 0.70 0.20

theta_1 0.80 1.30
```

17

```
zeta_0  0.70  0.20
zeta_1  0.80  1.30
xi_0  0.20  0.90
xi_1  0.30  1
index_coin_imp  0.5
fixed_coins  2
i  9  −5
theta_0  0  0.75
theta_1  0.75  1.23
zeta_0  0  0.75
zeta_1  0.75  1.23
xi_0  0.25  0.75
xi_1  0.25  0.95
ref  0
dim_absorb  0
i
c
gnu_plot  0
```

4.1.6 folder ex6

- a line of size 201 (-100 to 100);

- a number of steps 100;

- circular boundary condition;

- initial state $\frac{1}{\sqrt{50}}(5\,|-5\rangle\,|L\rangle + i5\,|5\rangle\,|R\rangle)$;

- varying coin operator in every position of parameters $\theta \in [0.70, 0.80], \zeta \in [0.70, 0.80], \xi \in [0.20, 0.30]$; The chosen Hadamard coin will have no effect when this mode is used;

- fixed coin operators are defined in positions 9 with parameters $\theta \in [0, 0.75], \zeta \in [0, 0.75], \xi \in [0.25, 0.25]$ and -5 with parameters $\theta \in [0.75, 1.23], \zeta \in [0.75, 1.23], \xi \in [0.75, 0.95]$. These coin operators will be used only in those positions, leaving the others with the varying coin operator defined for all other nodes;

- Probability of a coin to be chosen with parameters $\theta, \zeta, \xi \in [0, \pi/2]$, in each position at each step, is set to 0.5. These coin operators has precedence to the afore defined coin operators;

- 2 Fixed broken links are set between nodes 9 and 10 and between -5 and -4;

- Random broken links are set independently of the fixed broken links, with value 0.5;

N 100

Steps 100

points 2

i −5 5

c 0 1

real_value 5 0

complex_value 0 5

index_brokenlink_imp 0.5

fixed_broken_links 2

I_0 −50 50

coin 0

varying_coin_each_step 1

theta_0 0.70

```
theta_1  0.80

zeta_0  0.70

zeta_1  0.80

xi_0  0.20

xi_1  0.30

index_coin_imp  0.5

fixed_coins  2

i  9  −5

theta_0  0  0.75

theta_1  0.75  1.23

zeta_0  0  0.75

zeta_1  0.75  1.23

xi_0  0.25  0.75

xi_1  0.25  0.95

ref  0

dim_absorb  0

i

c

gnu_plot  0
```

4.2 Examples of Configuration for 2 particles quantum walk on the line

4.2.1 Folder ex1

- a line of size 61 (-30 to 30);

- a number of steps 30;

- reflecting boundary condition;

- initial state $\frac{1}{\sqrt{50}}(5\ket{-5}\ket{L} + i5\ket{5}\ket{R})$;

M 30

steps 30

points 1

i 0

j 0

c 0

real_value 1

complex_value 0

different_lines 0

index_brokenlink_imp_1 0

fixed_broken_link_1 0

I_0

coin1 0

varying_coin_each_step_1 0

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

index_coin_imp_1 0

fixed_coin_1 0

K

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

rf1 1

index_brokenlink_imp_2 0

fixed_broken_link_2 0

J_0

coin2 0

varying_coin_each_step_2 0

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

index_coin_imp_2 0

fixed_coin_2 0

K

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

rf2 1

dim 0

i_0

j_0

m_options 0

vn_x 1

vn_y 1

vn_m_info 1

q_discord 1

```
ent_form  1
gnu_plot  0
```

4.2.2   Folder ex2

- a line of size 201 (-100 to 100);

- a number of steps 100;

- reflecting boundary condition;

- initial state $|0, 0\rangle |LL\rangle$;

- Measure at position $(15, 0)$ with option 1;

```
M 30
steps 30
points 1
i 0
j 0
c 0
real_value 1
complex_value 0
different_lines 0
index_brokenlink_imp_1 0
fixed_broken_link_1 0
I_0
coin1 0
varying_coin_each_step_1 0
theta_0
theta_1
zeta_0
zeta_1
xi_0
xi_1
index_coin_imp_1 0
fixed_coin_1 0
K
theta_0
theta_1
zeta_0
```

zeta_1

xi_0

xi_1

rf1 1

index_brokenlink_imp_2 0

fixed_broken_link_2 0

J_0

coin2 0

varying_coin_each_step_2 0

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

index_coin_imp_2 0

fixed_coin_2 0

K

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

rf2 1

dim 1

i_0 15

j_0 0

m_options 1

vn_x 1

vn_y 1

vn_m_info 1

q_discord 1

ent_form 1

gnu_plot 0

26

4.2.3   Folder ex3

- two lines of size 61 (-30 to 30);

- a number of steps 30;

- Two particles in different lines;

- reflecting boundary condition;

- initial state $|0,0\rangle\,|LL\rangle$;

- varying coin operator in every position of parameters $\theta \in [0.70, 0.80], \zeta \in [0.70, 0.80], \xi \in [0.20, 0.30]$ for the second walker. The chosen Hadamard coin will have no effect when this mode is used;

```
M 30

steps  30

points  1

i  0

j  0

c  0

real_value  1

complex_value  0

different_lines  1

index_brokenlink_imp_1  0

fixed_broken_link_1  0

I_0

coin1  0

varying_coin_each_step_1  0

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

index_coin_imp_1  0

fixed_coin_1  0

K

theta_0
```

theta_1

zeta_0

zeta_1

xi_0

xi_1

rf1 1

index_brokenlink_imp_2 0

fixed_broken_link_2 0

J_0

coin2 0

varying_coin_each_step_2 1

theta_0 0.70 0.20

theta_1 0.80 1.30

zeta_0 0.70 0.20

zeta_1 0.80 1.30

xi_0 0.20 0.90

xi_1 0.30 1

index_coin_imp_2 0

fixed_coin_2 0

K

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

rf2 1

dim 0

i_0

j_0

m_options 0

vn_x 1

vn_y 1

vn_m_info 1

```
q_discord  1
ent_form  1
gnu_plot  0
```

4.2.4   Folder ex4

- two lines of size 61 (-30 to 30);

- a number of steps 30;

- circular boundary condition;

- initial state $|0,0\rangle\,|LL\rangle$;

- varying coin operator in every position of parameters $\theta \in [0.70, 0.80], \zeta \in [0.70, 0.80], \xi \in [0.20, 0.30]$ for the second walker. The chosen Hadamard coin will have no effect when this mode is used;

- measure at positions $(15, y)$ for $-30 \leq y \leq 30$. Measure option 1;

```
M 30

steps  30

points  1

i  0

j  0

c  0

real_value  1

complex_value  0

different_lines  1

index_brokenlink_imp_1  0

fixed_broken_link_1  0

I_0

coin1  0

varying_coin_each_step_1  0

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

index_coin_imp_1  0

fixed_coin_1  0

K

theta_0
```

theta_1

zeta_0

zeta_1

xi_0

xi_1

rf1 1

index_brokenlink_imp_2 0

fixed_broken_link_2 0

J_0

coin2 0

varying_coin_each_step_2 1

theta_0 0.70

theta_1 0.80

zeta_0 0.70

zeta_1 0.80

xi_0 0.20

xi_1 0.30

index_coin_imp_2 0

fixed_coin_2 0

K

theta_0

theta_1

zeta_0

zeta_1

xi_0

xi_1

rf2 1

dim 61

i_0 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15
15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15 15

j_0 -30 -29 -28 -27 -26 -25 -24 -23 -22 -21 -20 -19 -18 -17 -16 -15 -14 -13 -12 -11 -10 -9 -8 -7 -6 -5
-4 -3 -2 -1 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30

m_options 1

vn_x 1

vn_y 1

vn_m_info 1

q_discord 1

ent_form 1

gnu_plot 0

4.3 Example for 1 particle quantum walk on the lattice

4.3.1 Folder ex1

- two lines of size 91 (-45 to 45);

- a number of steps 45;

- Reflecting boundary conditions in both directions;

- initial state $\frac{1}{2}(|-30,-30\rangle\,|E\rangle+i\,|-30,-30\rangle\,|N\rangle+|30,30\rangle\,|W\rangle+i\,|30,30\rangle\,|S\rangle$);

- Fixed coins at positions $16 \leq x,y \leq 45$ of parameters $\zeta_1,\theta_1,\xi_1,\zeta_2,\theta_2,\xi_2 \in$
  $[0.392699081698724, 1.17809724509617]$;

N 45

steps 45

points 4

i -30 -30 30 30

j -30 -30 30 30

c 0 2 3 1

real_value 1 0 1 0

complex_value 0 1 0 1

moi 0

moj 0

index_brokenlink_imp 0

fixed_broken_links 0

i

j

c

coin 0

varying_coin_each_step 0

theta_10

theta_11

theta_20

theta_21

zeta_10

zeta_11

zeta_20
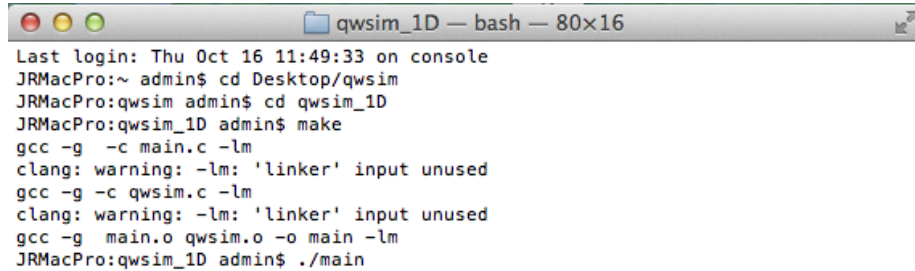
zeta_21

xi_10

xi_11

xi_20

xi_21

index_coin_imp 0

fixed_coins 841

i 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 16 17 18 ...

j 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 16 17 17 17 ...

theta_10 0.392699081698724 0.392699081698724 0.392699081698724 0.392699081698724 0.392 ...

theta_11 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724509617 1.1780972 ...

theta_111.17809724509617 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724 ...

theta_21 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724509617 1.1780972 ...

zeta_10 0.392699081698724 0.392699081698724 0.392699081698724 0.392699081698724 0.3926 ...

zeta_11 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724 ...

zeta_20 0.392699081698724 0.392699081698724 0.392699081698724 0.392699081698724 0.3926 ...

zeta_21 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724 ...

xi_10 0.392699081698724 0.392699081698724 0.392699081698724 0.392699081698724 0.392699 ...

xi_11 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724509617 1.1780972450 ...

xi_20 0.392699081698724 0.392699081698724 0.392699081698724 0.392699081698724 0.392699 ...

xi_21 1.17809724509617 1.17809724509617 1.17809724509617 1.17809724509617 1.1780972450 ...

refI 1

refJ 1

dim 0

i_0

j_0

m_options 0

vn_x 0

vn_y 0

vn_m_info 0

q_discord 0

ent_form 0

gnu_plot 0

5 Running the simulation

1. To run program for 1-particle quantum walk on the line write the following command in the command line:

   (a) "./main" and type enter;

Figure 10: Command line instructions to run 1-particle quantum walk program after compilation.

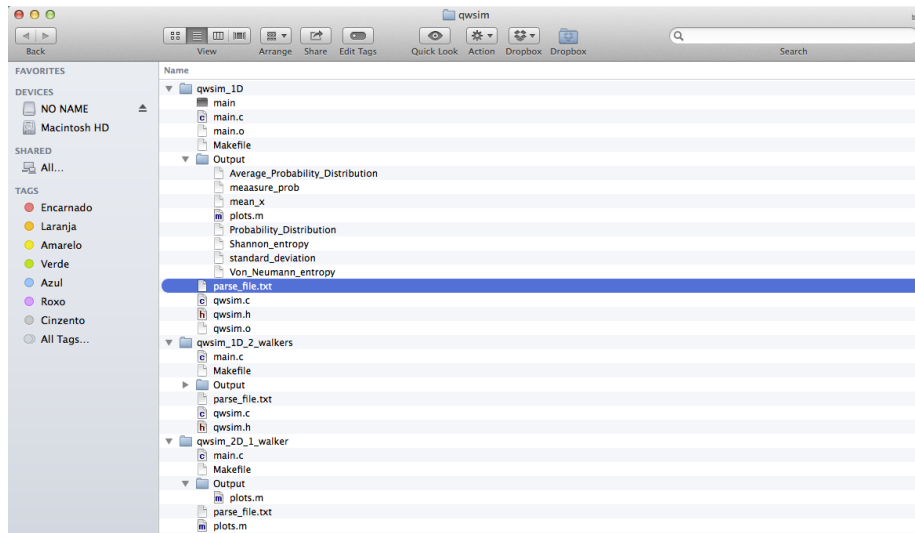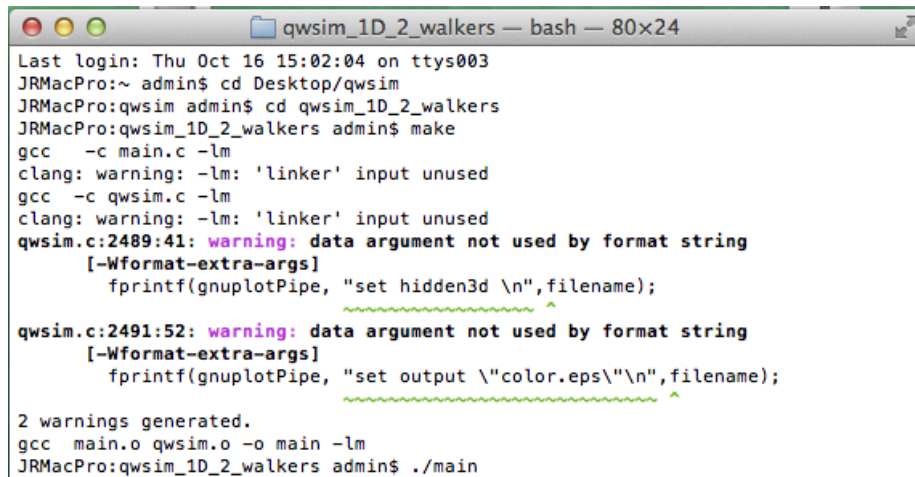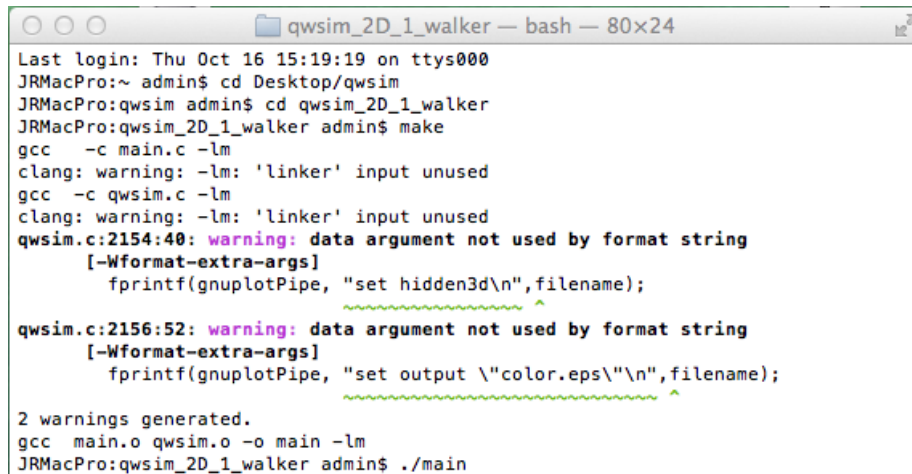The generated files are "main.o", "qwsim.o" and "main" and are placed in folder "qwsim_1D".

Figure 11: Files generated after simulation of 1-particle quantum walk program.

2. To compile the program for 2-particle quantum walk on the line write in the command line.

   (a) "./main" and type enter;

```
Last login: Thu Oct 16 15:02:04 on ttys003
JRMacPro:~ admin$ cd Desktop/qwsim
JRMacPro:qwsim admin$ cd qwsim_1D_2_walkers
JRMacPro:qwsim_1D_2_walkers admin$ make
gcc   -c main.c -lm
clang: warning: -lm: 'linker' input unused
gcc   -c qwsim.c -lm
clang: warning: -lm: 'linker' input unused
qwsim.c:2489:41: warning: data argument not used by format string
      [-Wformat-extra-args]
        fprintf(gnuplotPipe, "set hidden3d \n",filename);
                             ~~~~~~~~~~~~~~~~ ^
qwsim.c:2491:52: warning: data argument not used by format string
      [-Wformat-extra-args]
        fprintf(gnuplotPipe, "set output \"color.eps\"\n",filename);
                             ~~~~~~~~~~~~~~~~~~~~~~~~~~~ ^
2 warnings generated.
gcc  main.o qwsim.o -o main -lm
JRMacPro:qwsim_1D_2_walkers admin$ ./main
```

Figure 12: Command line instructions to compile 2-particle quantum walk program after compilation.

The generated files are "main.o", "qwsim.o" and "main" and are placed in folder "qwsim_1D".
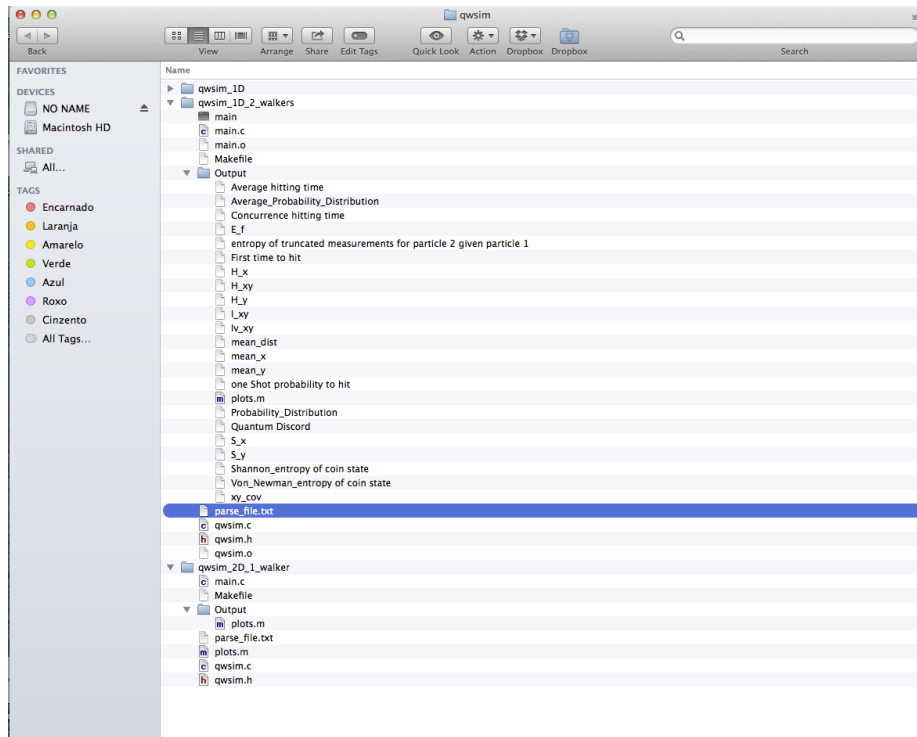
Figure 13: Files generated after simulation of 2-particle quantum walk.

3. To run program for 1-particle quantum walk on the line write the following command in the command line:

   (a) "./main" and type enter;

```
Last login: Thu Oct 16 15:19:19 on ttys000
JRMacPro:~ admin$ cd Desktop/qwsim
JRMacPro:qwsim admin$ cd qwsim_2D_1_walker
JRMacPro:qwsim_2D_1_walker admin$ make
gcc   -c main.c -lm
clang: warning: -lm: 'linker' input unused
gcc   -c qwsim.c -lm
clang: warning: -lm: 'linker' input unused
qwsim.c:2154:40: warning: data argument not used by format string
      [-Wformat-extra-args]
        fprintf(gnuplotPipe, "set hidden3d\n",filename);
                             ~~~~~~~~~~~~~~~~ ^

qwsim.c:2156:52: warning: data argument not used by format string
      [-Wformat-extra-args]
        fprintf(gnuplotPipe, "set output \"color.eps\"\n",filename);
                             ~~~~~~~~~~~~~~~~~~~~~~~~~~~~ ^

2 warnings generated.
gcc  main.o qwsim.o -o main -lm
JRMacPro:qwsim_2D_1_walker admin$ ./main
```

Figure 14: Command line instructions to compile 1-particle quantum walk on a lattice program.

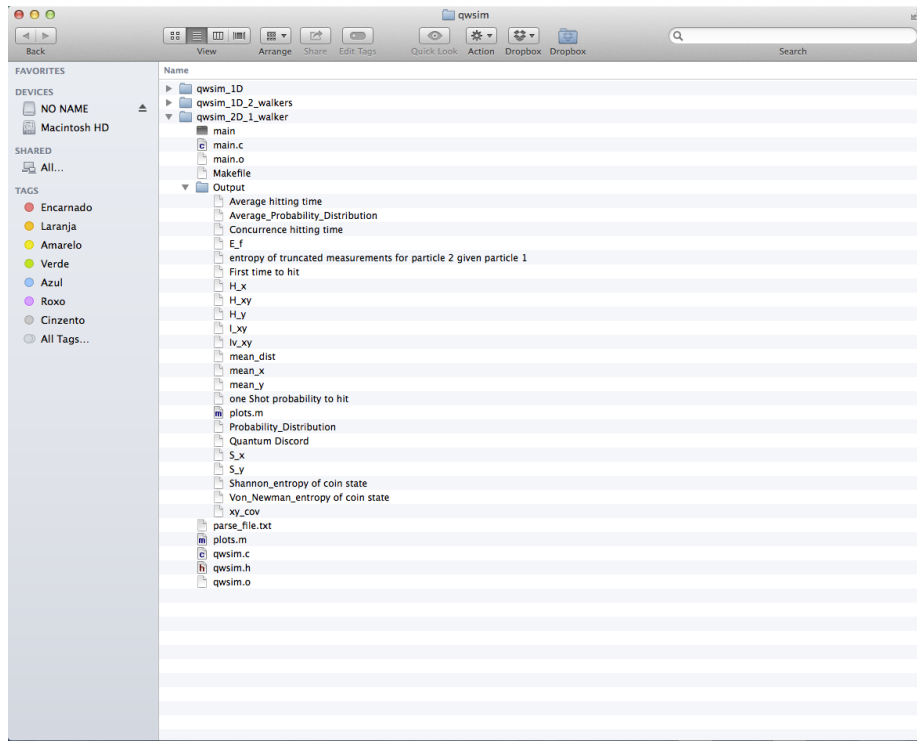The generated files are "main.o", "qwsim.o" and "main" and are placed in folder "qwsim_1D".

Figure 15: Files generated after simulation of 1-particle quantum walk on a lattice.

6 After Simulation

1. The files generated after the simulation of 1-particle quantum walk on a line are:

   - Average_probability_Distribution;
   - measure_prob;
   - mean_x;
   - Probability_Distribution;
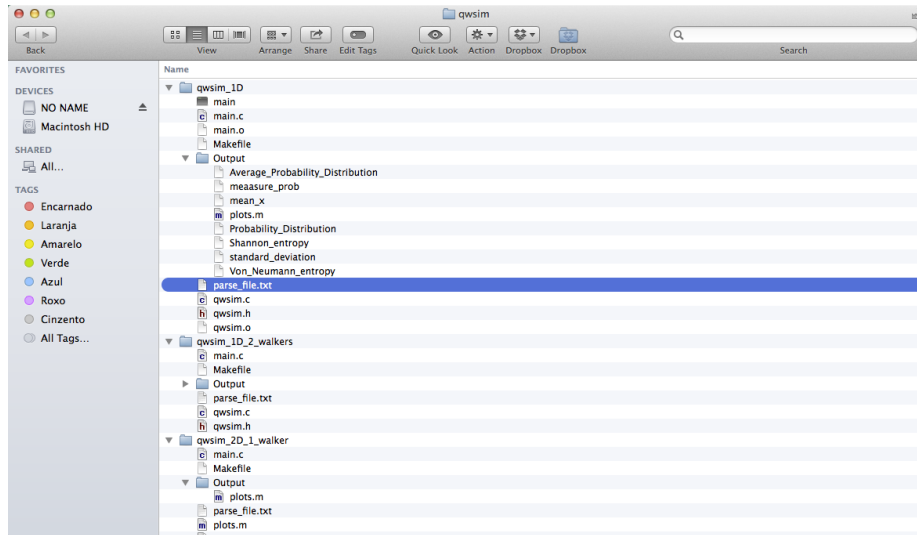   - Shannon_entropy;
   - Von_Neumann_entropy.

Figure 16: Files generated after simulation of 1-particle quantum walk on a line.

2. The files generated after the simulation of 2-particles quantum walk on a line are:

   - Average_hitting_time;
   - Average_probability_Distribution;
   - Concurrence_hitting_time;
   - E_f;
   - entropy of truncated measurements for particle 2 given particle 1;
   - first time to hit;
   - H_x;

40

- H_xy;

- H_y;

- I_xy;

- Iv_xy;

- mean_dist;

- mean_x;

- mean_y;

- one shot probability to hit;

- Probability_Distribution;

- Quantum Discord;

- S_x;

- S_y;

- Shannon_entropy of coin state;

- xy_cov.
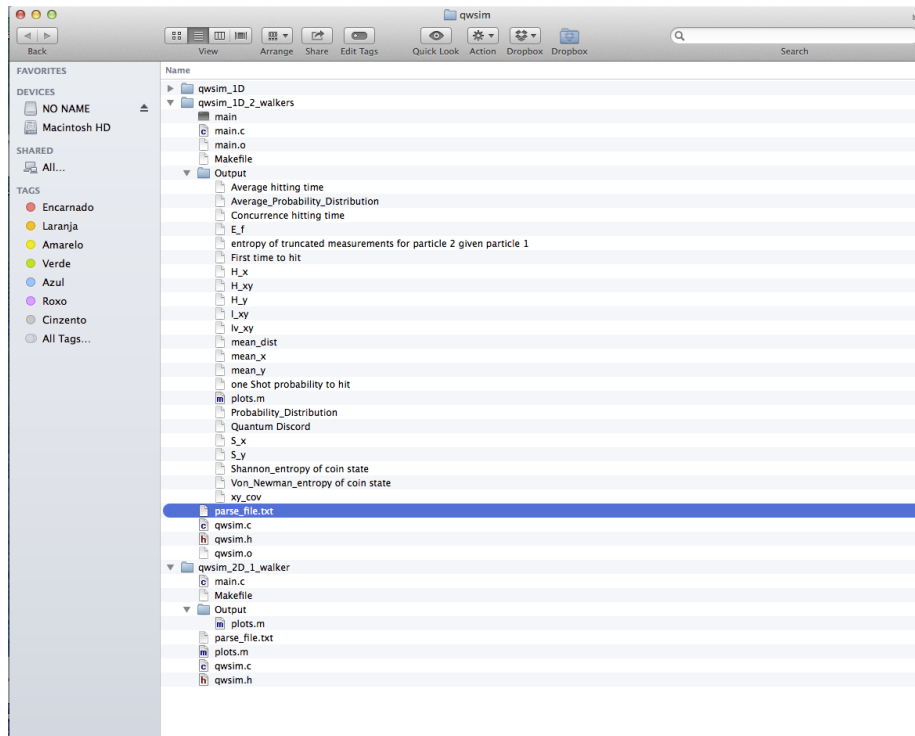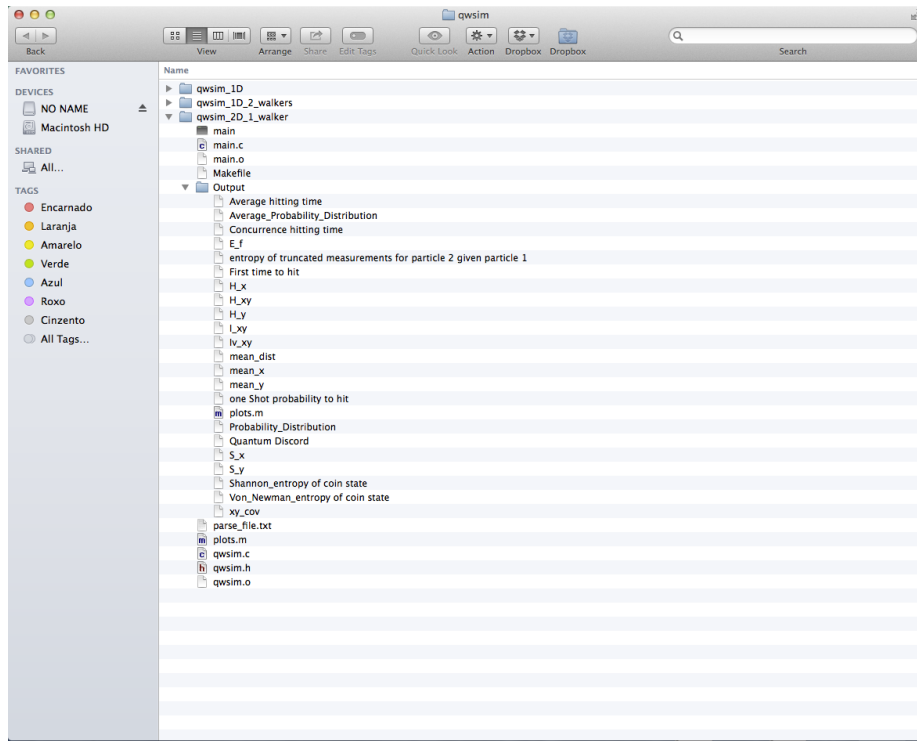

Figure 17: Files generated after simulation of 2-particles quantum walk on a line.

3. The files generated after the simulation of 2-particles quantum walk on a line are:

- Average_hitting_time;
- Average_probability_Distribution;
- Concurrence_hitting_time;
- E_f;
- entropy of truncated measurements for particle 2 given particle 1;
- first time to hit;
- H_x;
- H_xy;
- H_y;
- I_xy;
- Iv_xy;
- mean_dist;
- mean_x;
- mean_y;
- one shot probability to hit;
- Probability_Distribution;
- Quantum Discord;
- S_x;
- S_y;
- Shannon_entropy of coin state;
- xy_cov.

Figure 18: Files generated after simulation of 1-particle quantum walk on a lattice.
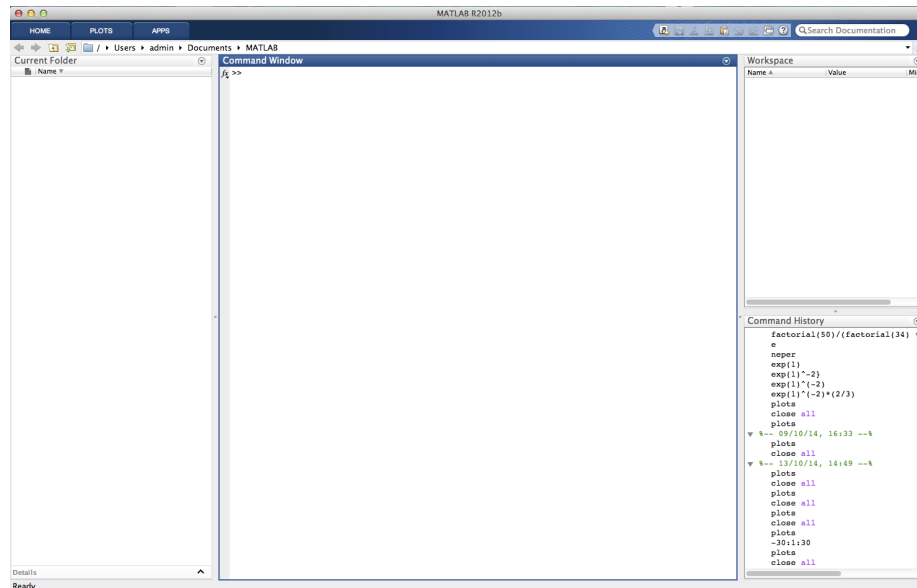
7 Using Matlab

1. Open Matlab;

Figure 19: Command Line of Matlab.

2. In order to plot the results of 1 particle quantum walk on the line do:

   (a) Change the directory to
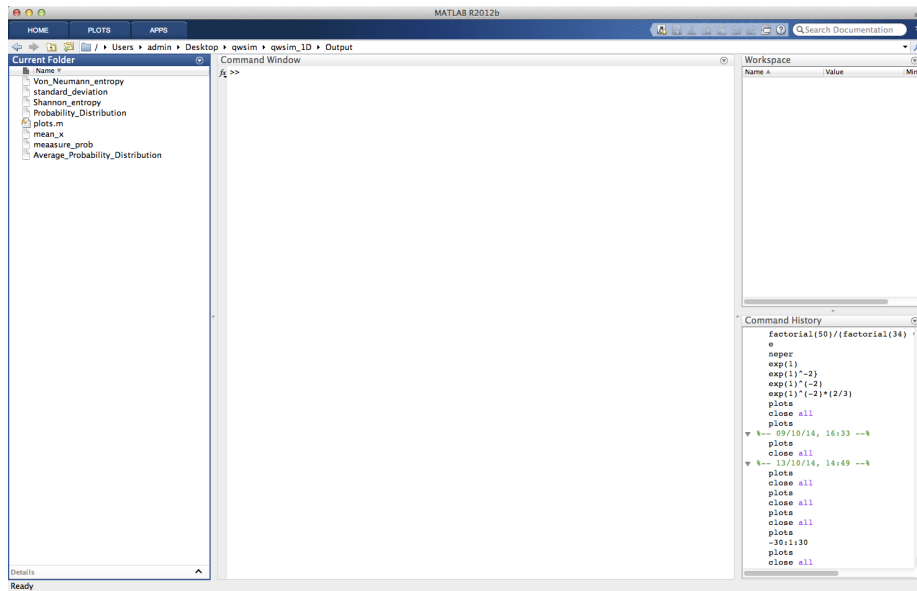   "/users/admin/Desktop/qwsim/qwsim_1D/Output":

Figure 20: Changing directory in Matlab.

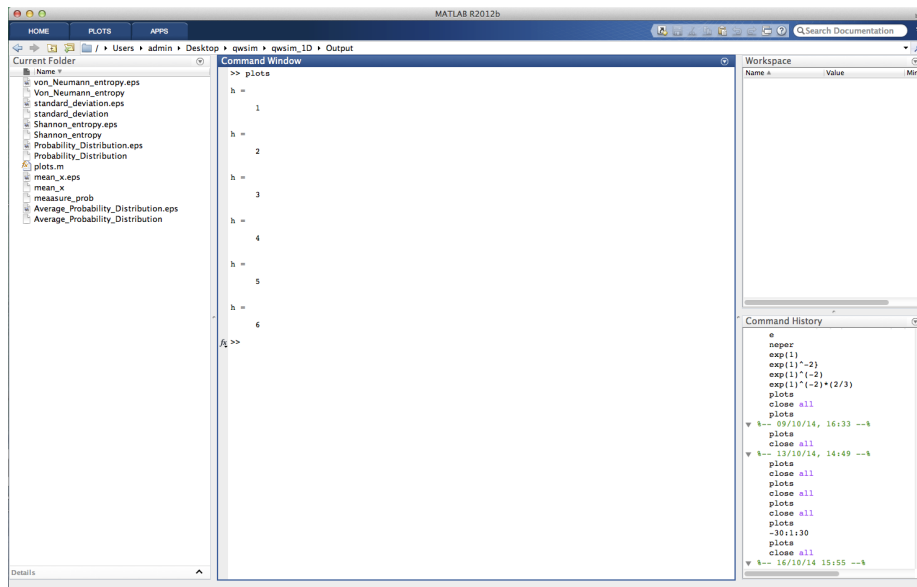(b) Write "plots" and press enter in the Matlab command line;

Figure 21: Plotting the data from the simulation of 1-particle quantum walk on a line.

3. In order to plot the results of 2 particles quantum walk on the line do:

   (a) Change the directory to
       "/users/admin/Desktop/qwsim/qwsim_1D_two_walkers/Output":

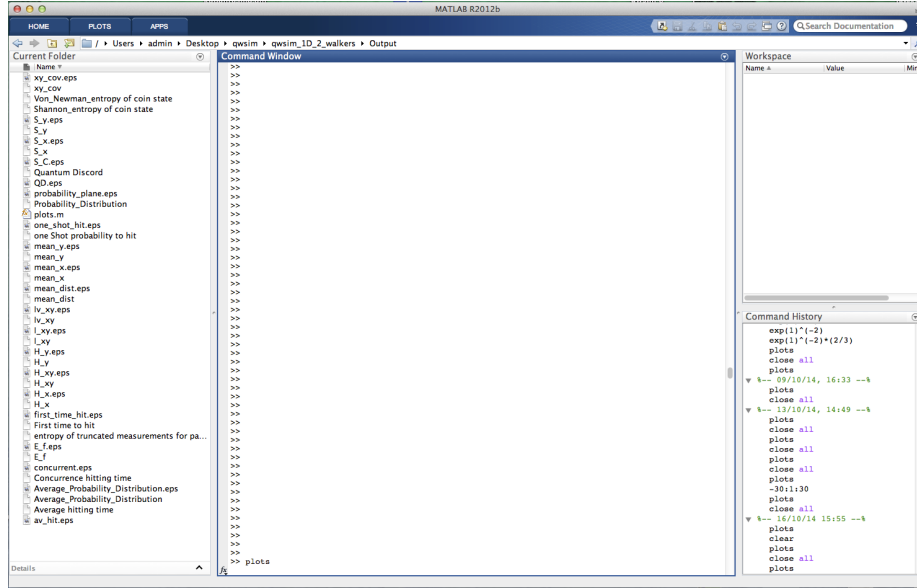   (b) Write "plots" and press enter in the Matlab command line;

Figure 22: Plotting the data from the simulation of 2-particle quantum walk on a line.

4. In order to plot the results of 1 particle quantum walk on a 2D lattice do:

   (a) Change the directory to
       "/users/admin/Desktop/qwsim/qwsim_2D_one_walker/Output":

   (b) Write "plots" and press enter in the Matlab command line;

References

[1] J. Rodrigues, N. Paunkovic, P. Mateus, "A Simulator for Discrete Quantum Walk on Lattice"