On Dirac wave function in 1D

Ulrich Mutze [*]

June 9, 2020

1 Dirac equation in one-dimensional discrete space

The following series of pictures studies Lorentz transformations ('boosts') for the Dirac electron in a version of the theory in which space is one-dimensional and discrete.

The following text describes what exactly is shown in the pictures.

Each of the pictures shows a electron wave function with two complex-valued spinor components. In one spatial dimensions the standard four spinor components decouple into two pairs and we are grateful for the reduction of the state space dimension by a factor 2 brought about by ignoring one of the couples. To formulate the Dirac equation we need two 2 by 2 matrices

$$\alpha = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} , \quad \beta = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{1}$$

Units are chosen such that $\hbar$, c, and e (not needed here) have the numerical value 1. To be completely explicit with respect to dimensions and kinds of physical quantities let us state that we use a system with a single fundamental dimension, namely *length* $\mathbf{L}$. Some important dimensions (not necessarily fundamental ones) than are: Power $= \mathbf{L}^{-2}$, Time = Mass = Energy = Momentum $= \mathbf{L}^{-1}$, Velocity = Angular Momentum $= \mathbf{L}^0$, Length $= \mathbf{L}^1$, Area = Force $= \mathbf{L}^2$, Volume = Acceleration $= \mathbf{L}^3$. Any physical quantity can the be characterized by a real number (the numerical value), or a set of real numbers (e.g. if the quantity is a tensor), after a unit of length has been chosen. The dimension of a quantity describes the rule by which the numerical value of the quantity changes if we change our unit of length.

Although computer programs allow to work with dimensions, they run faster and need less storage space if they work with pure numbers. So there are reasons for agreeing on a unit of length. Here we choose as unit length the length of an interval in one-dimensional space on which the Dirac wave functions 'live' and which, therefore, will be referred to as 'the biotope'. According to our method of discretization the wave functions are defined on a finite equidistant chain of points which are obtained if we divide the interval $\left[-\frac{1}{2}, \frac{1}{2}\right]$

---

[*]www.ulrichmutze.de

into n congruent parts, each of length $\delta := 1/n$ and take the midpoints $x_i := -\frac{1}{2} + (i - \frac{1}{2})\delta$ of these intervals. Of course, a wave function ψ is a $\mathbb{C}^2$-valued function on this lattice, which we denote by $\mathcal{B}_n$, where $\mathcal{B}$ stands for 'biotope'. The value of ψ at the i-th lattice point x_i will be denoted $\psi(i)$. The scalar product of wave functions ψ and φ is defined as

$$\langle\,\psi\,|\,\varphi\,\rangle := \langle\,\psi(1)\,|\,\varphi(1)\,\rangle \,+\, \ldots \,+\, \langle\,\psi(n)\,|\,\varphi(n)\,\rangle\,, \tag{2}$$

where the scalar product in $\mathbb{C}^2$ is understood. The wave functions thus form in a natural manner a Hilbert space of dimension $d := 2n$ and observables are represented as Hermitian d by d matrices. The basic elements of quantum theory on our discrete biotope are the position operator X and the momentum operator P. Their definition is as follows:

$$(X\psi)_\mu(j) = x_j\,\psi_\mu(j)\,, \quad (P\psi)_\mu(j) = \frac{\psi_\mu(j+1) - \psi_\mu(j-1)}{2\,\mathrm{i}\,\delta} \tag{3}$$

As to the meaning of the out-of range indexes 0 and $n+1$ we assume periodic continuation of ψ i.e. $\psi(0) := \psi(n)$ and $\psi(n+1) := \psi(1)$. With the matrices from (1) we define the Hamiltonian H and the generator N of pure Lorentz transformations as

$$H = \alpha\,P + \beta\,m\,, \quad N = (XH + HX)/2\,, \tag{4}$$

It may be convenient to write out the Hamiltonian in all detail:

$$(H\psi)_\mu(j) = \alpha_{\mu\nu}\frac{\psi_\nu(j+1) - \psi_\nu(j-1)}{2\,\mathrm{i}\,\delta} + m\beta_{\mu\nu}\psi_\nu(j)\,, \tag{5}$$

where sum convention over $\{1, 2\}$ for the Greek indexes is understood.

The Figures 2 to 39 where created in a single run of my C++ program *dirac* with subprogram selector $= 5$ in file dirac.ini. It thus processed the data cointained in file app5.ini. In this case we had $n = 8000$ and all these operators were effectively Hermitian matrices of size 1600 by 1600. The technical means were an 12 years old 2.08 GHz dual core laptop computer compiling via g++-9 under 64 bit Ubuntu 18.04.4 LTS. The total execution time was 141,7 seconds. The largest single subtask was the spectral representation of N (a 1600 x 1600 matrix as stated above) based on the function Eigen::SelfAdjointEigenSolver. This subtask took 54.5 seconds.

The continuum theory of the electron associates the Compton wavelength $\frac{h}{mc}$ and the reduced Compton wavelength $\frac{\hbar}{mc}$ with the mass m of the electron. In the present note we consider the term Compton wavelength as an abbreviation of 'reduced Compton wavelength' and — according to our convention on units — write it as $\frac{1}{m}$.

We always choose the electron as the particle to consider. Its Compton wavelength λ_e is known to be approximately $0.386\,\mathrm{pm}$. Let us now consider the situation that the length l of our biotope is given in real space e.g. as a channel in some semiconductor device. Let its length in normal units be 1 nm. What are the values of m in the Dirac Hamiltonian and how large one will reasonable choose n? We set $\xi := l/\lambda_\mathrm{e}$. This is a dimensionless quantity so it can be computed in any system of units. Especially in our system in which $l = 1$ and $\lambda_\mathrm{e} = 1/m$ so that $\xi = m$. The number n of discretization

points has to be chosen such that for each Compton wavelength in the biotope there are 6 to 10 discretization points. I denote this parameter as *spw* (samples per wave). Then we have $n = spw\,\xi = spw\,m$. The rules for keeping n as low as possible are to let the biotope cover only those parts of space which will be actually 'visited' by the wave function during the simulation and to choose *spw* not unnecessarily large (e.g. not larger than 20). For *spw* too close to 1 or even below 1 the dynamics behaves strange: If one boosts a smooth peaked wave packet it will not move under time evolution but will hold its place — only slightly changing shape in an oscillatory manner. If the intention is to approximate the wave function dynamics of continuum theory one simply has to increase the number *spw*.

The operators introduced so far — in particular H and N — are rather complicated functions. To get an insight into their 'inner working' it is often instructive to visualize their action on simple wave functions. Bell-shaped curves of *Gaussian* type

$$\mathrm{bell}(x, x_0, \sigma) := \exp\big(-\frac{1}{2}((x - x_0)/\sigma)^2\big) \tag{6}$$

are particularly useful and are well established for this purpose. Although a Gaussian, for more than 4σ away from the peak seems to vanish in usual graphical resolution, for the present purpose it is important to have bell-shaped curves which vanish exactly outside a known adjustable interval. To avoid numerical artifacts smooth curves are preferable. Such curves are well-known. The following construction is taken with modifications from [1], Theorem 2.1 of Chapter 10.2.

$$\mathrm{wall}(x) := \begin{cases} \exp(-1.25/x) & x > 0 \\ 0 & x \leq 0 \end{cases}, \tag{7}$$

$$\mathrm{peak}_{\mathrm{lean}}(x, x_0, \sigma) := \mathrm{wall}(x_0 + 4\sigma - x)\,\mathrm{wall}(x - x_0 + 4\sigma), \tag{8}$$

$$\mathrm{peak}(x, x_0, \sigma) := \mathrm{peak}_{\mathrm{lean}}(x, x_0, \sigma)/\mathrm{peak}_{\mathrm{lean}}(x_0, x_0, \sigma). \tag{9}$$

As things are arranged the carrier of $x \mapsto \mathrm{peak}(x, x_0, \sigma)$ is the interval $[x_0 - 4\sigma, x_0 + 4\sigma]$ although in normal graphical resolution even the smaller interval $[x_0 - 3\sigma, x_0 + 3\sigma]$ looks like a carrier. Since the graphics of wave functions show always the squares of the quantum mechanical amplitudes (as will be defined soon) Fig 1 shows also the squared functions.
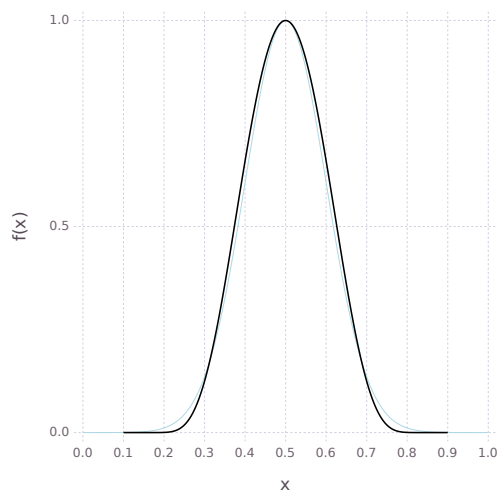
(a) showing function value
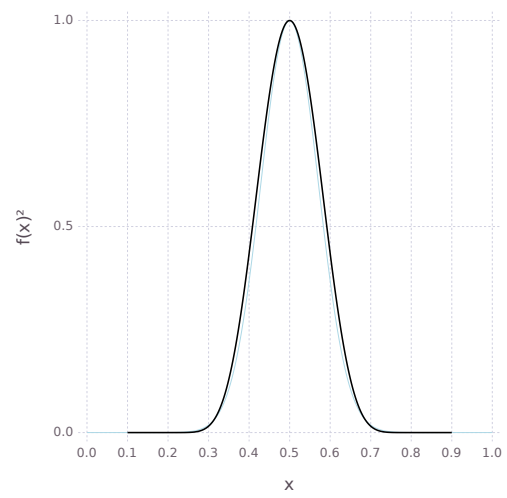
(b) showing square of function value

Figure 1: Functions peak$(\bullet, 0.5, 0.1)$ (black) and bell$(\bullet, 0.5, 0.1)$ (light blue). Function peak is shown only on its carrier interval [0.1, 0.9].

2 Boost series creating delta needles

In this section we try to visualize the action of the boost operation $\exp(-\mathrm{i}\,\rho N)$ by applying it for various values of the rapidity parameter ρ on three bell-shaped states. The number of discretization points is chosen as $n = 800$ which resulted in a total computation time of 4.4 minutes for generating the whole series of figures. This computation was no longer carried out by a program written in Julia but with a C++ program that makes heavy use of the linear algebra library 'Eigen'. The electron mass was chosen as $m = \frac{n}{20} = 40$ leading to $\lambda_{\mathrm{e}} = 1/40 = 20\,\delta$ (recall $\delta := 1/n$) so that the length 1 of the biotope equals the 40-fold of λ_{e} and thus measures $40 \cdot 0.386\,\mathrm{pm} = 15.4\,\mathrm{pm}$.

Let us define the state $\psi = (\psi_1, \psi_2)$ on which we let the operators $\exp(-\mathrm{i}\,\rho N)$ act. Here, the writing of a state as a pair refers to the two components of the values of the wave function, so that ψ_1 and ψ_2 are complex valued functions. The main feature is given by the function peak introduced in (9):

$$\psi_\mu(j) := \mathrm{peak}(x_j, x_0, \sigma)\, a_\mu \, \exp(\mathrm{i}\,\varphi_\mu)\,, \quad j \in \{1, \ldots, 800\}\,, \quad \mu \in \{1, 2\} \qquad (10)$$

and the parameters which enter are given for the computer run under consideration by

$$x_0 = 0\,, \quad \sigma = \frac{1}{32}\,, \quad a_1 = 1\,, \quad a_2 = 0.9\,, \quad \varphi_1 = 0\,, \quad \varphi_2 = \pi/6\,. \qquad (11)$$

The support of ψ has a length $8\sigma = \frac{8}{32} = \frac{1}{4}$ and thus measures $200\delta = 10\lambda_{\mathrm{e}}$. This is a moderate level of localization. Therefore the positive energy component ψ_{p} and and the negative energy component ψ_{n} of ψ both are not much wider than ψ in graphical resolution.

That ψ_1 and ψ_2 are given a small difference in amplitude and phase helps to prevent the curves representing $\mathrm{Re}\,(\psi_1)^2$, $\mathrm{Im}\,(\psi_1)^2$, $\mathrm{Re}\,(\psi_2)^2$, $\mathrm{Im}\,(\psi_2)^2$ from occluding each other. The colors associated with these quantities are green, red, blue, orange respectively. The quantity $\mathrm{Re}\,(\psi_1)^2 + \mathrm{Im}\,(\psi_1)^2 + \mathrm{Re}\,(\psi_2)^2 + \mathrm{Im}\,(\psi_2)^2$ is shown as a black curve.

The states that were computed are these:

$$\{\exp(-\mathrm{i}\,\rho N)\psi : \rho \in \{0, \pm 0.2, \pm 0.4, \ldots, \pm 1.6, \pm 1.8\}\}\,. \qquad (12)$$

For each computed state there were created two graphics: one with linear scale on the y-axis and one with logarithmic scale. The logarithmic graphics are indispensable if one wants to see how the support (carrier) of simulated wave function behaves relative to the exact carrier which in the graphics is marked by a representation (not to scale) of its indicator function (in gray color). This representation is missing on the graphics belonging to $|\rho| > 1.2$ since here the carrier extends over the whole biotope. The exact formula for the image $[a, b]_\rho$ of interval $[a, b]$ under a boost with rapidity ρ is

$$[a, b]_\rho = [\,\exp(|\rho|\,\mathrm{sign}(a))\,a,\ \exp(|\rho|\,\mathrm{sign}(b))\,b\,]\,. \qquad (13)$$

In the present series of images the needles become higher and higher for $\rho \longrightarrow -\infty$. Replacing the initial state by its time reversed version creates that behavior for $\rho \longrightarrow \infty$.

One should notice that the pictures can be strongly magnified by the functionality of the PDF format and the PDF viewer. Their information content exceeds by far that of a normal page view.
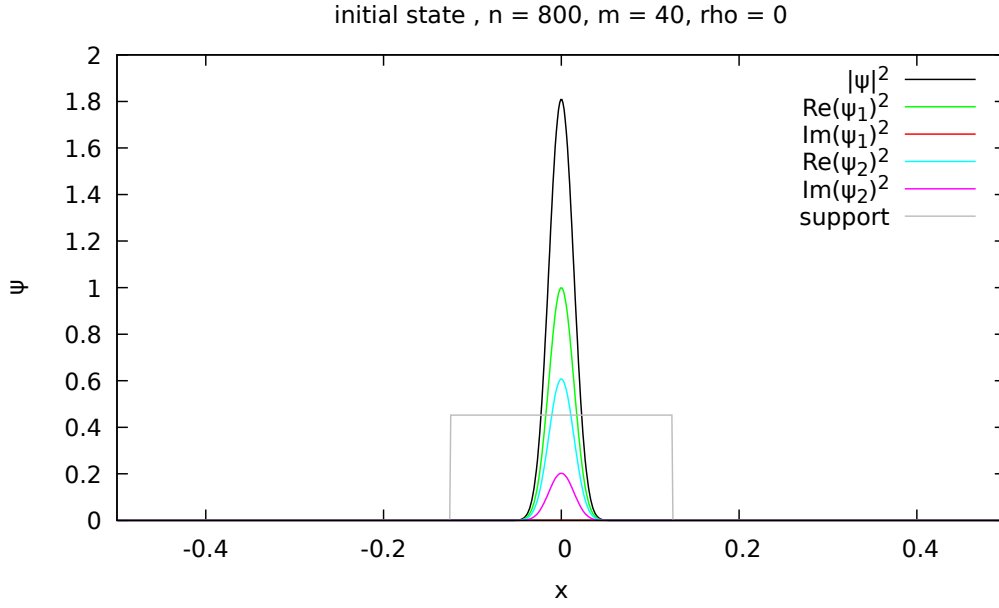
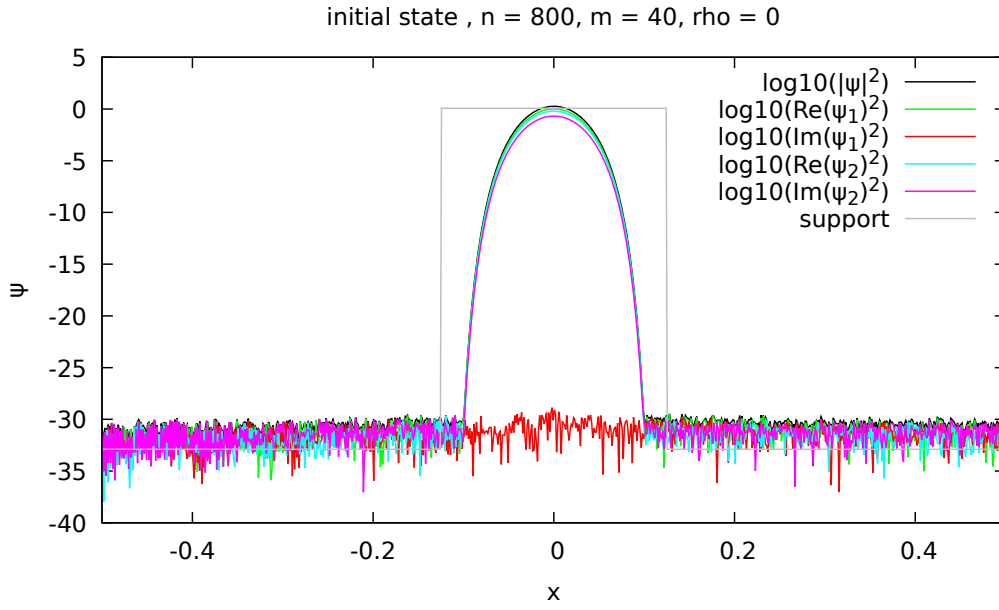Figure 2: State at rest, initial state of a boost series.

Figure 3: State at rest, initial state of a boost series. Log scale.
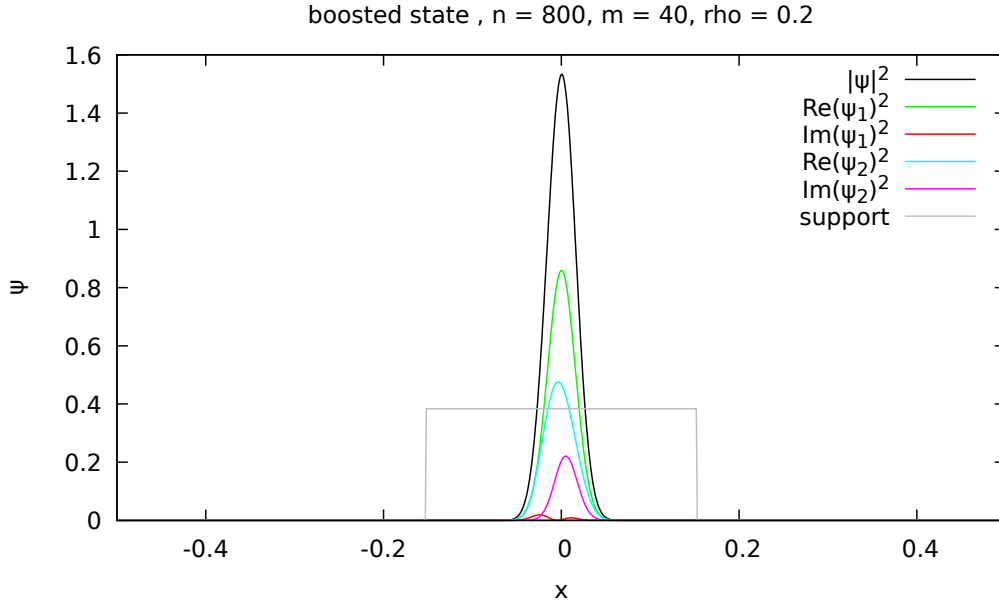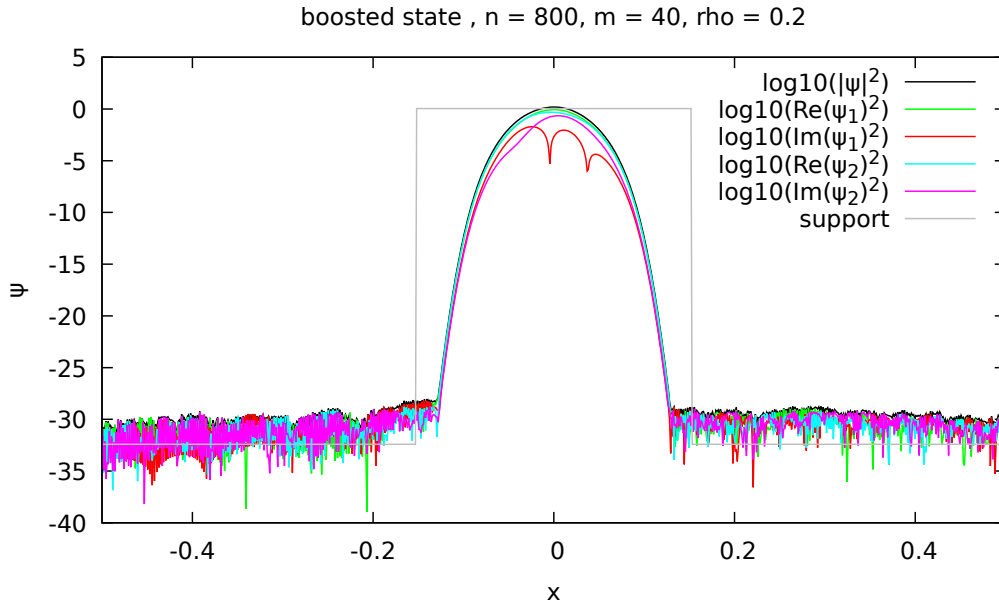
6

Figure 4: Boosted state $\rho = 0.2$.

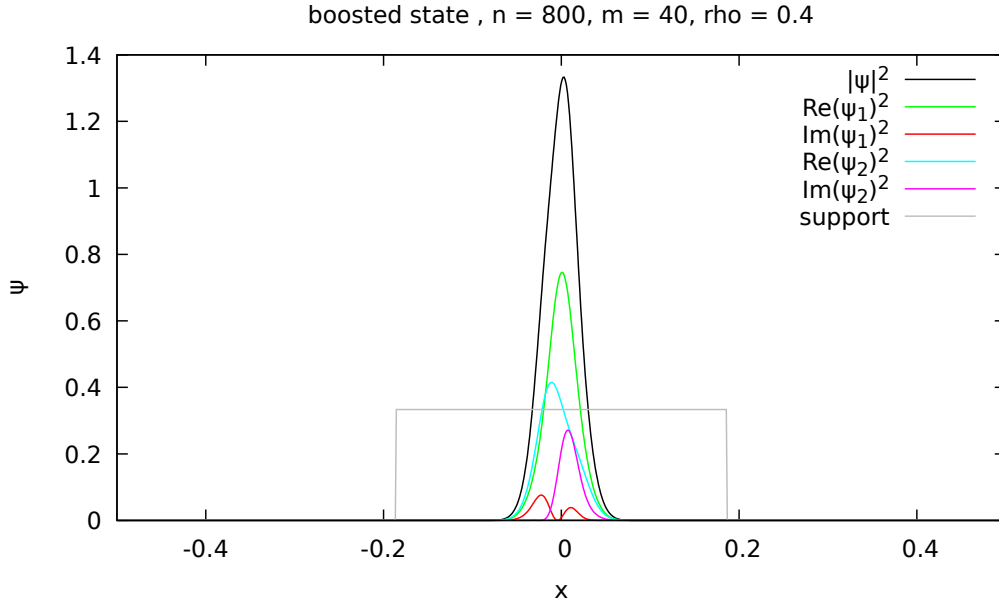Figure 5: Boosted state $\rho = 0.2$. Log scale.
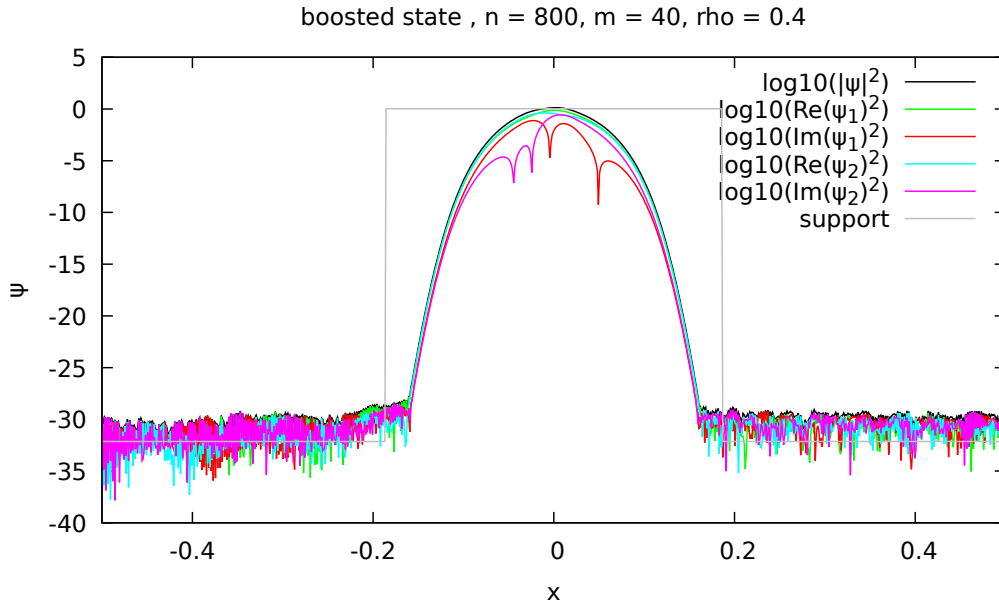
Figure 6: Boosted state $\rho = 0.4$.
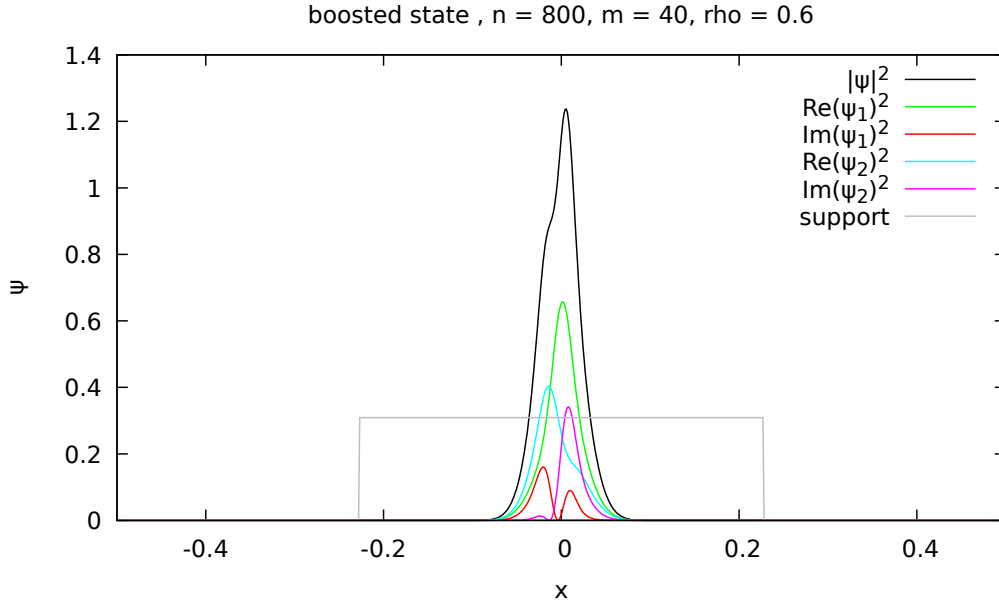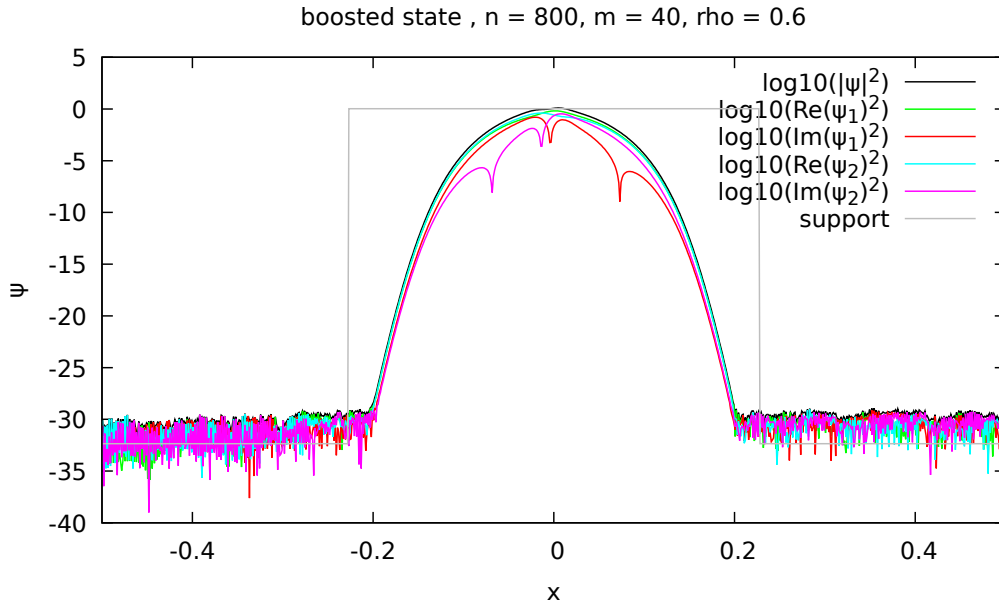

Figure 7: Boosted state $\rho = 0.4$. Log scale.

Figure 8: Boosted state $\rho = 0.6$.
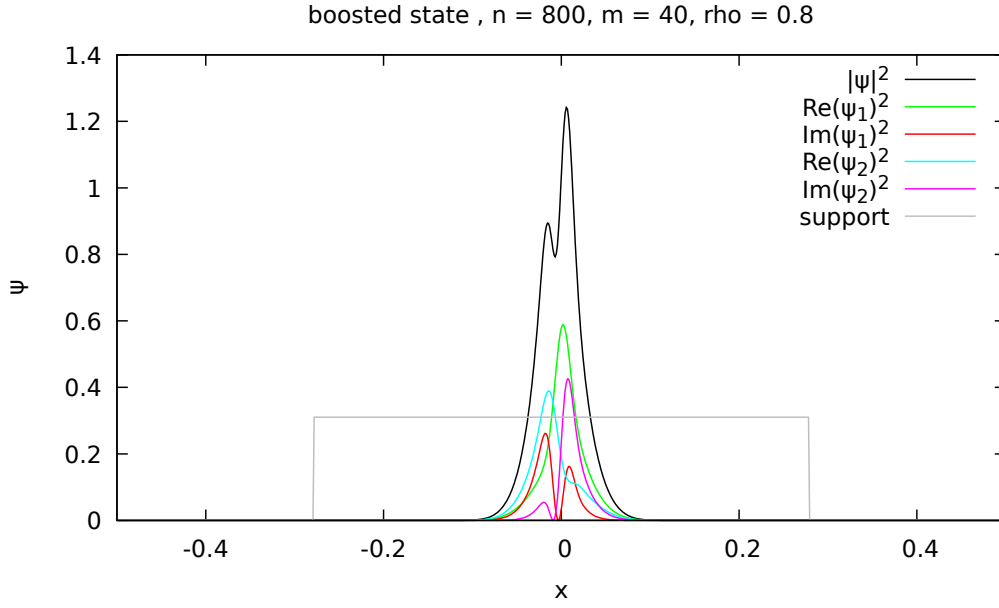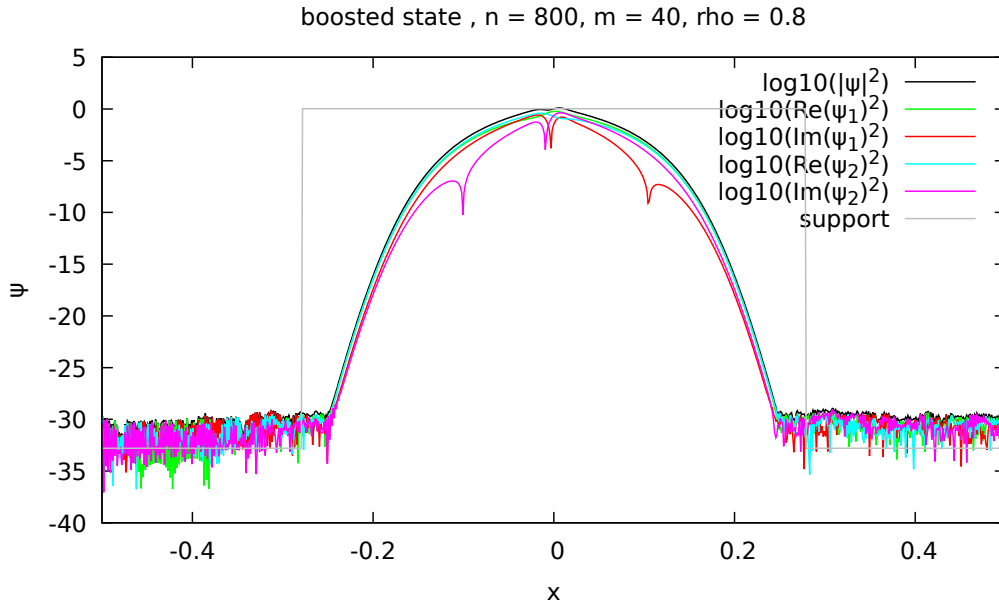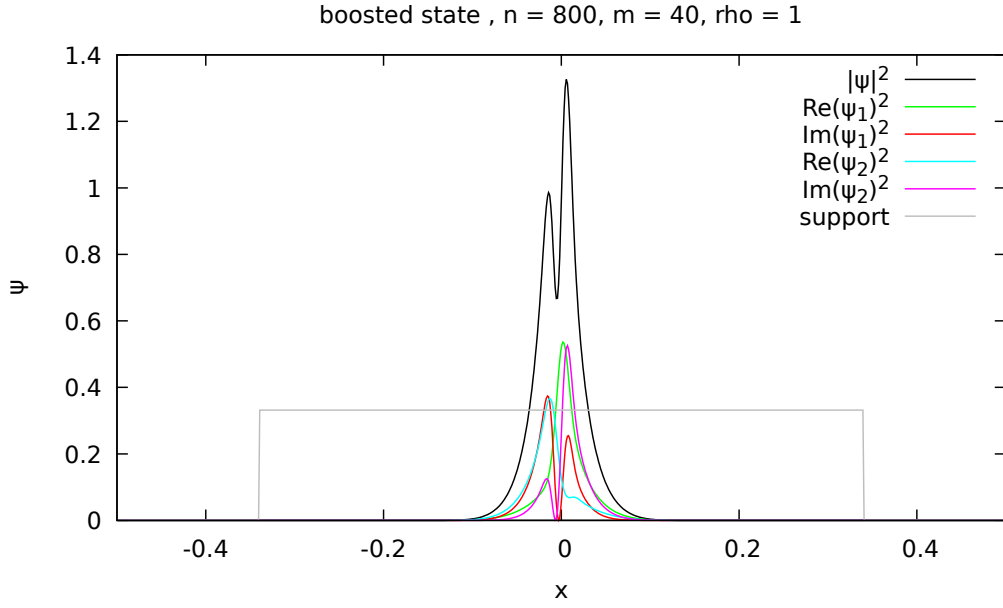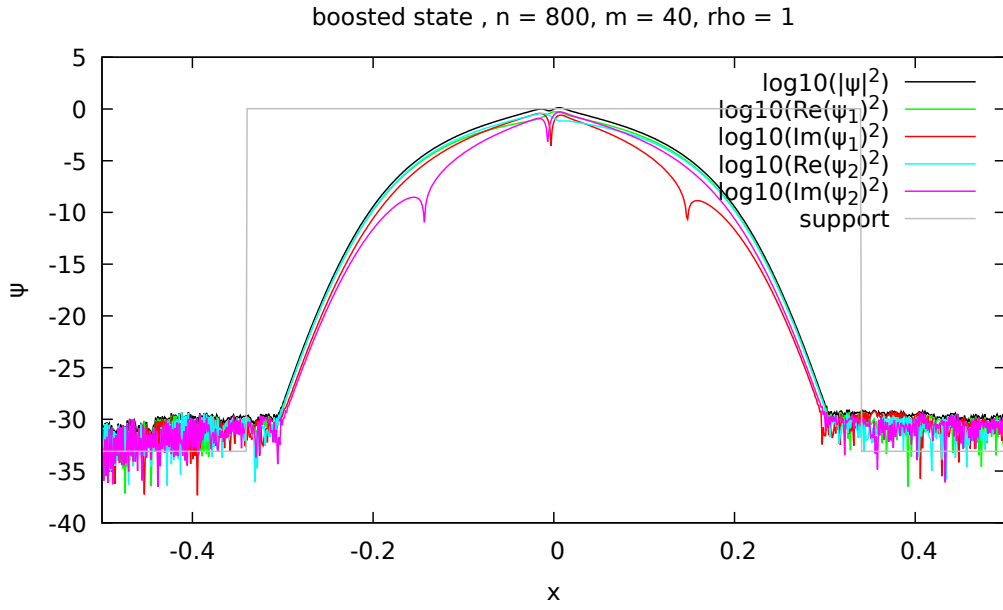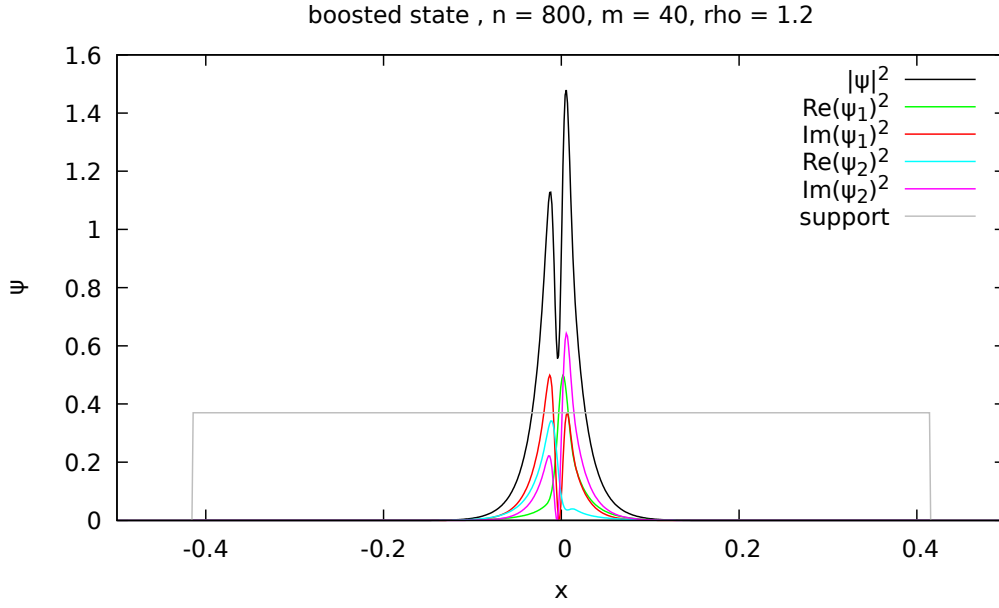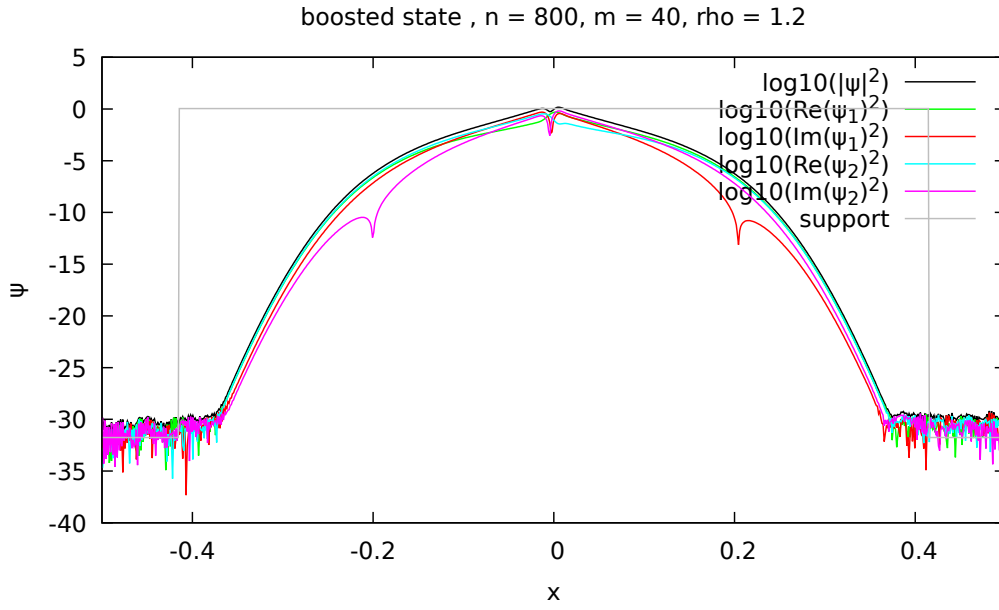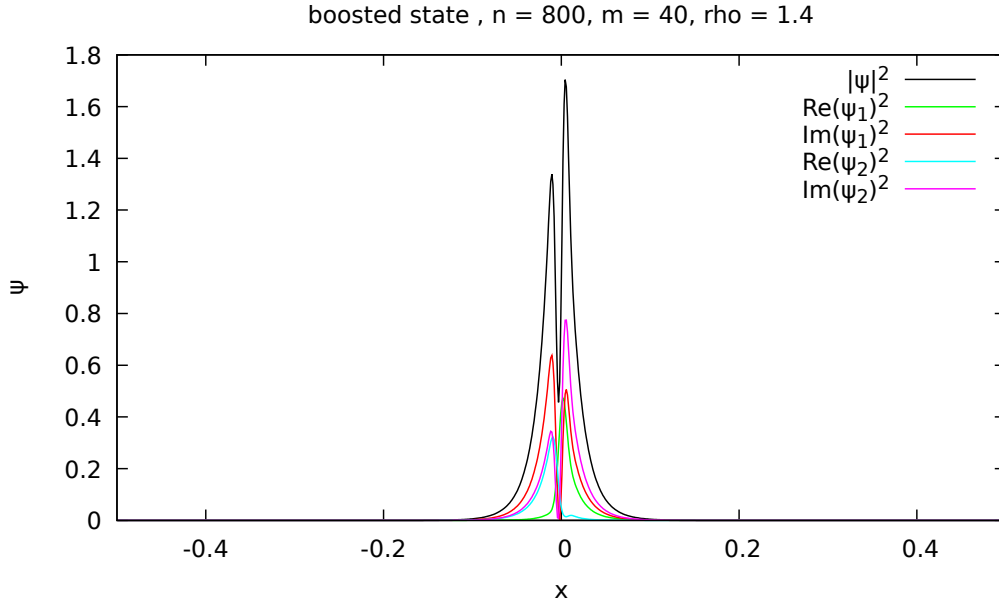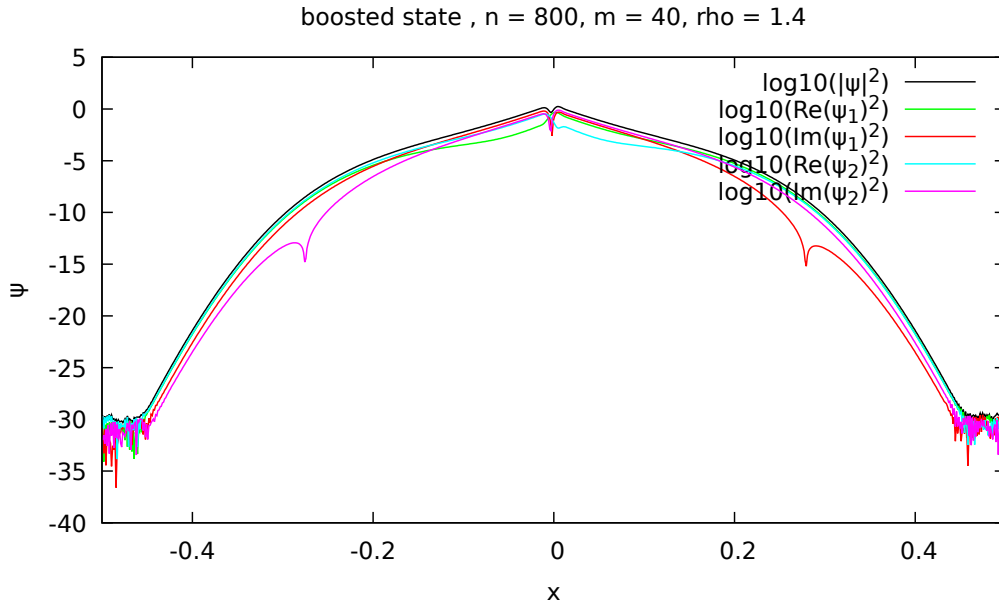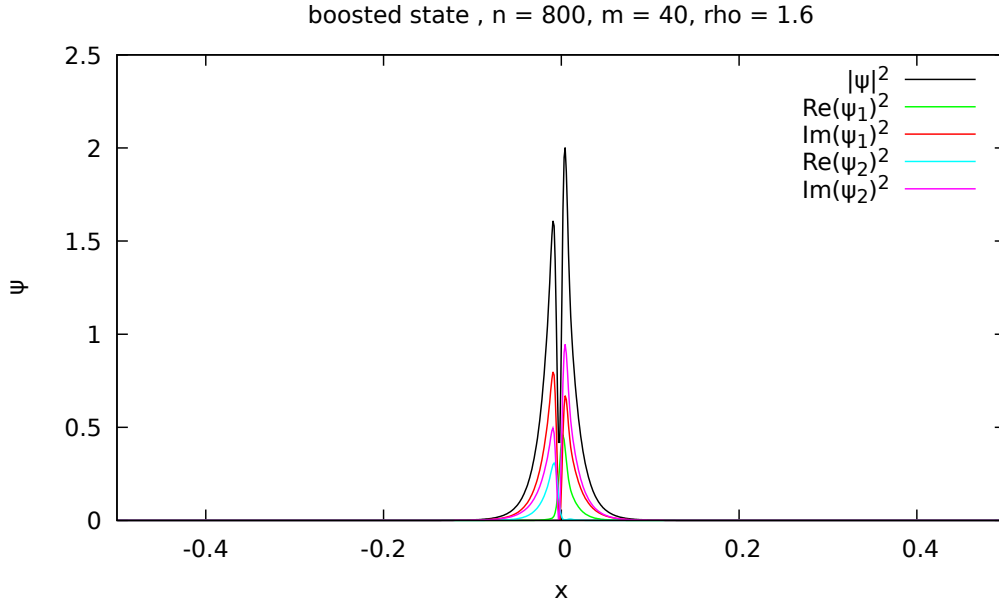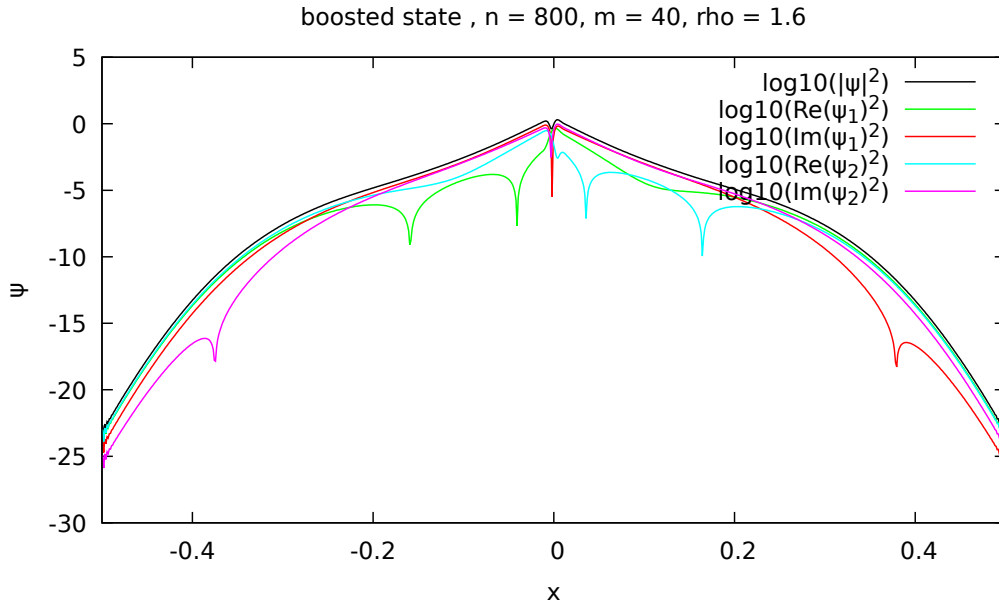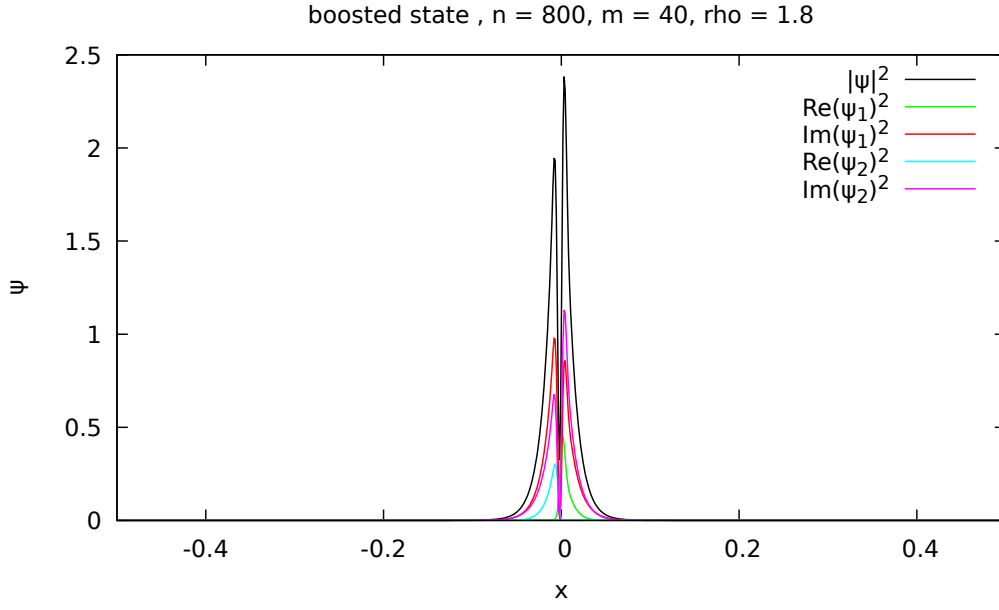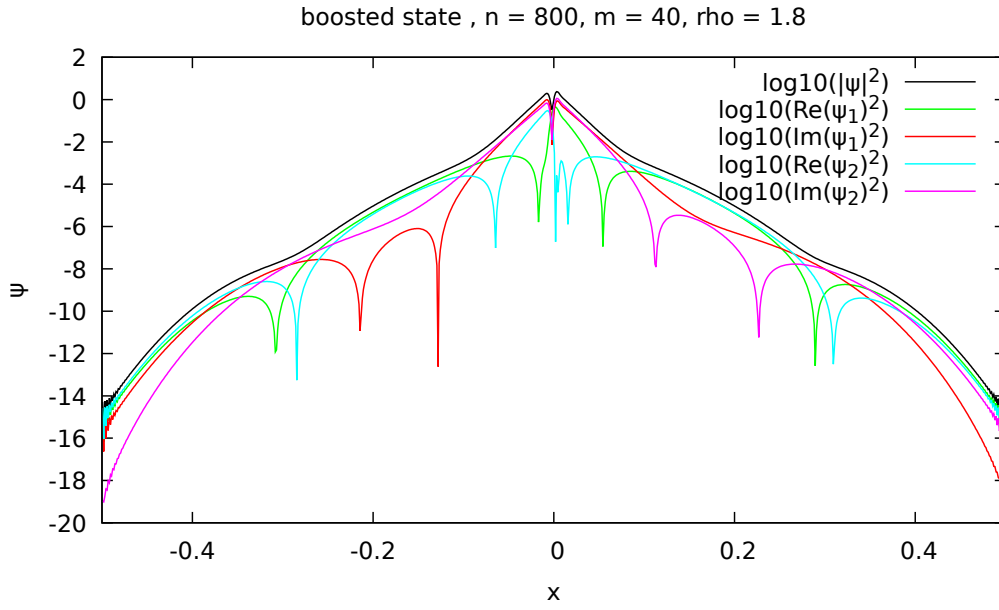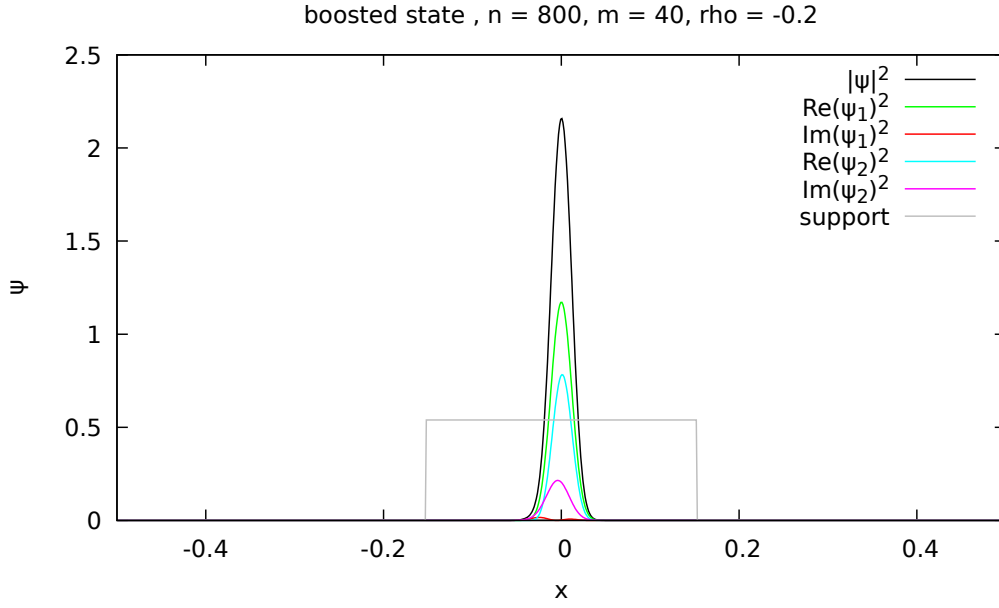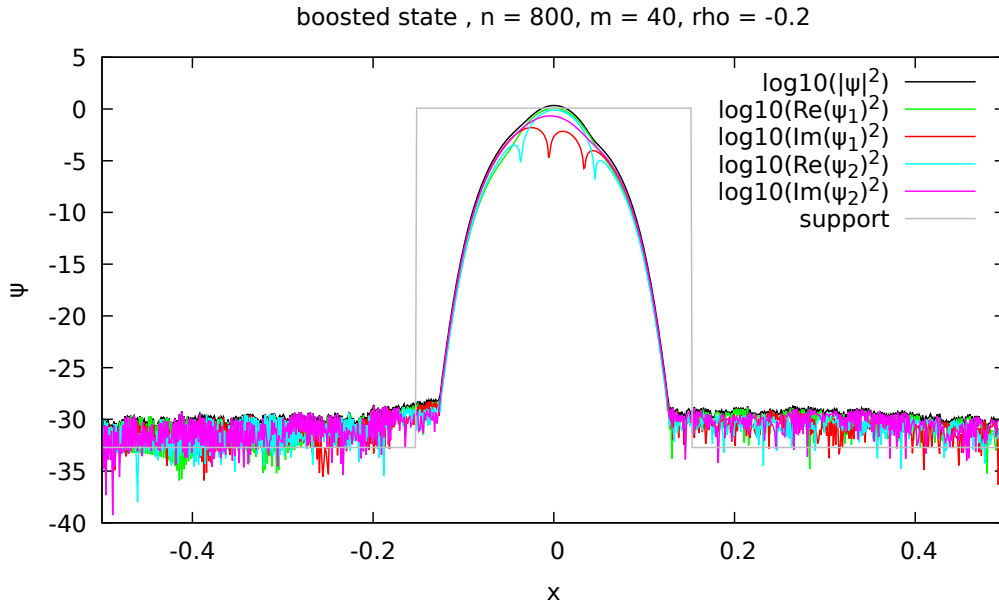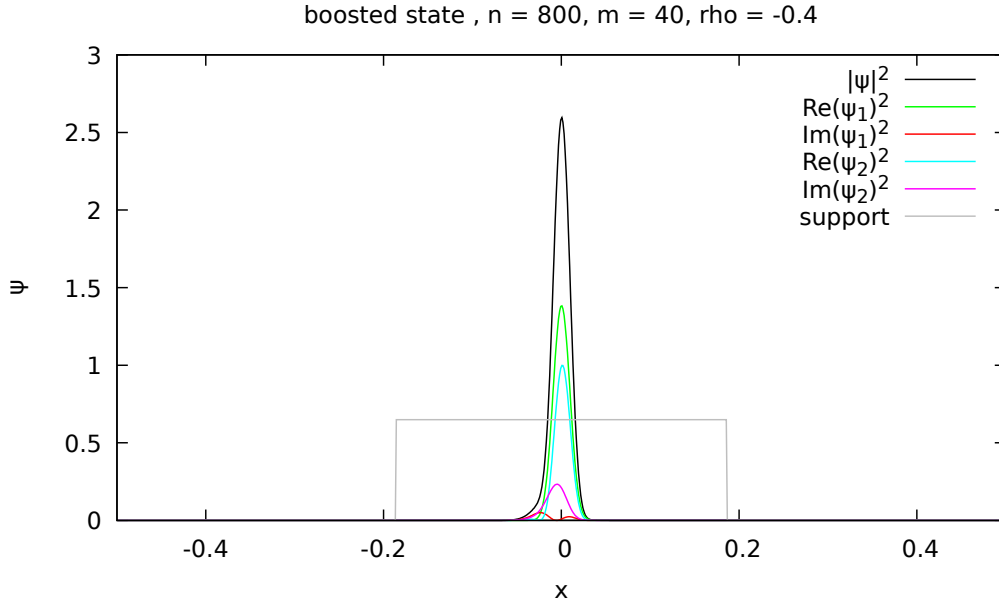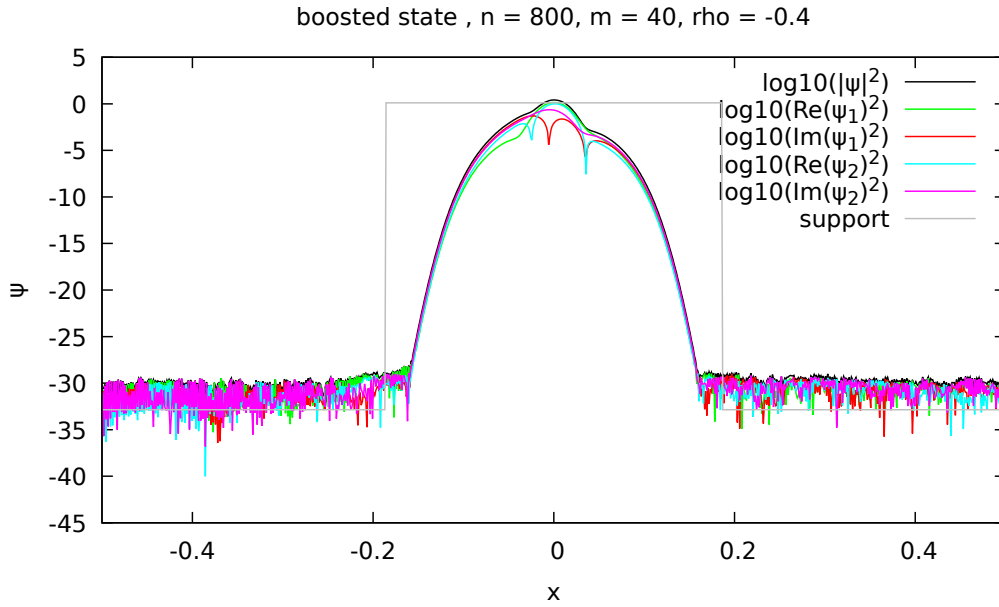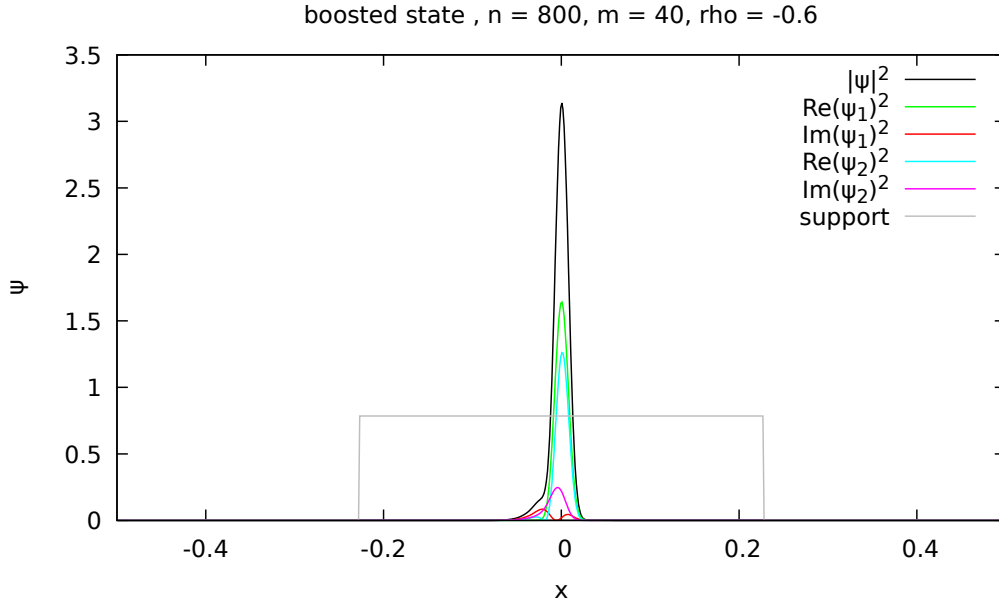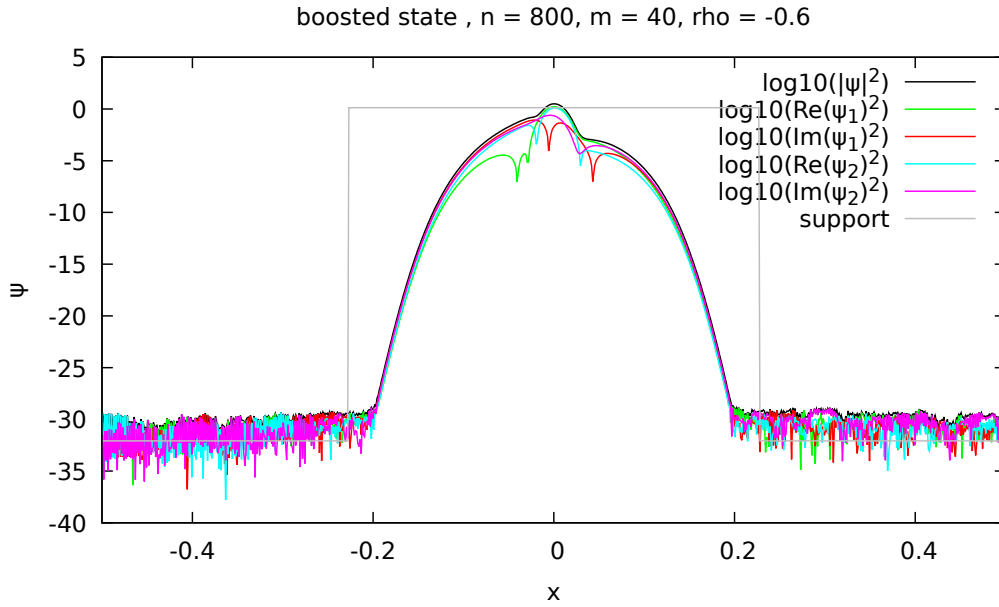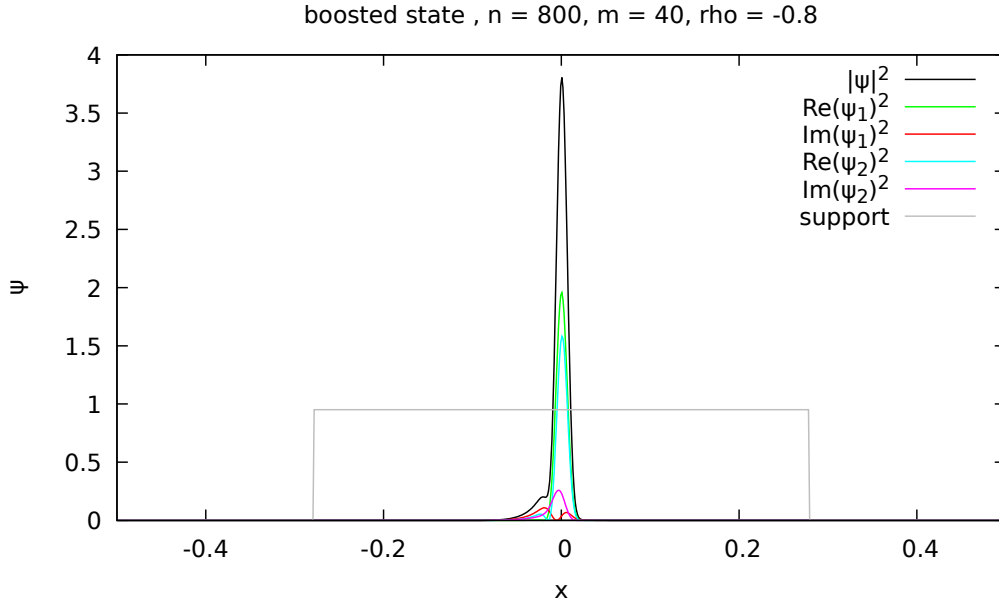

Figure 9: Boosted state $\rho = 0.6$. Log scale.

Figure 10: Boosted state $\rho = 0.8$.

Figure 11: Boosted state $\rho = 0.8$. Log scale.

boosted state , n = 800, m = 40, rho = 1

Figure 12: Boosted state $\rho = 1.0$.
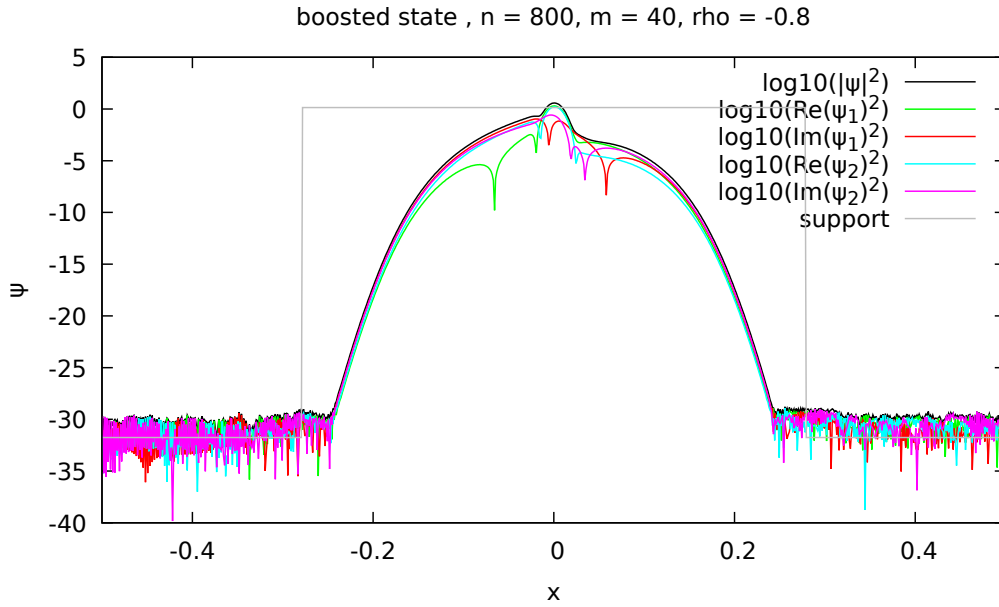
boosted state , n = 800, m = 40, rho = 1

Figure 13: Boosted state $\rho = 1.0$. Log scale.

Figure 14: Boosted state $\rho = 1.2$.

Figure 15: Boosted state $\rho = 1.2$. Log scale.

boosted state , n = 800, m = 40, rho = 1.4

Figure 16: Boosted state $\rho = 1.4$.

boosted state , n = 800, m = 40, rho = 1.4

Figure 17: Boosted state $\rho = 1.4$. Log scale.

13

Figure 18: Boosted state $\rho = 1.6$.

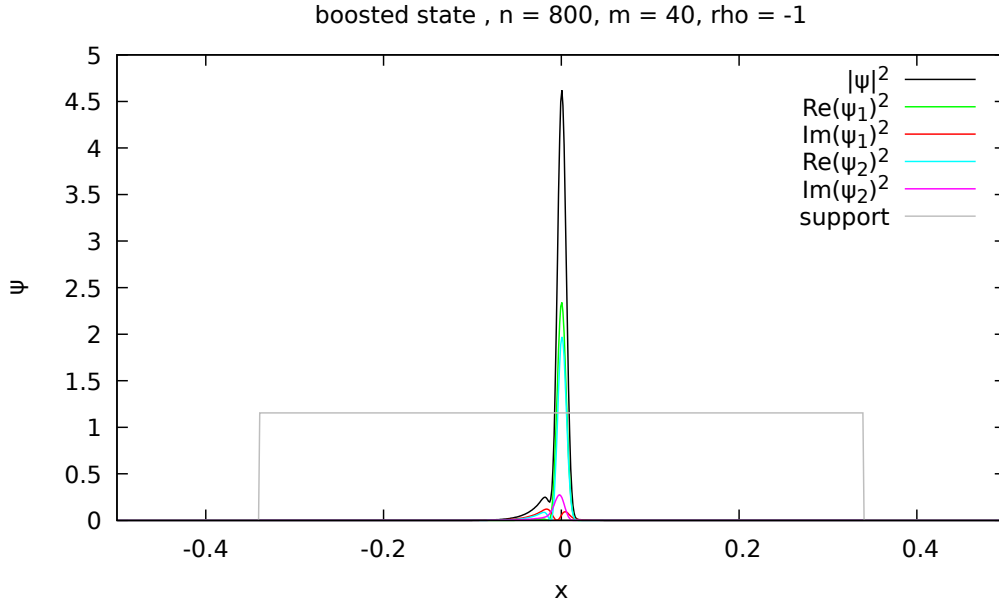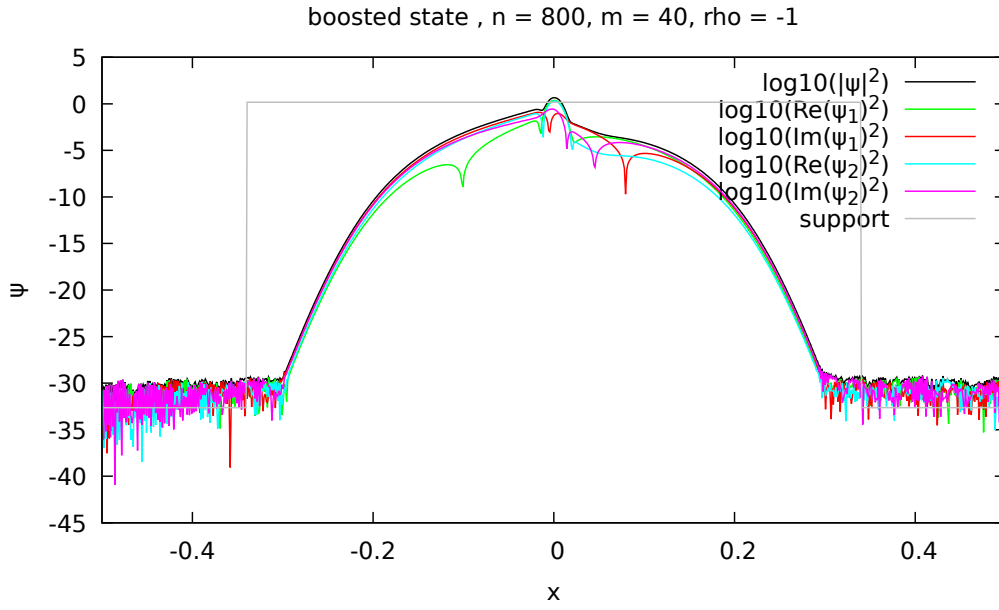
Figure 19: Boosted state $\rho = 1.6$. Log scale.

Figure 20: Boosted state $\rho = 1.8$.

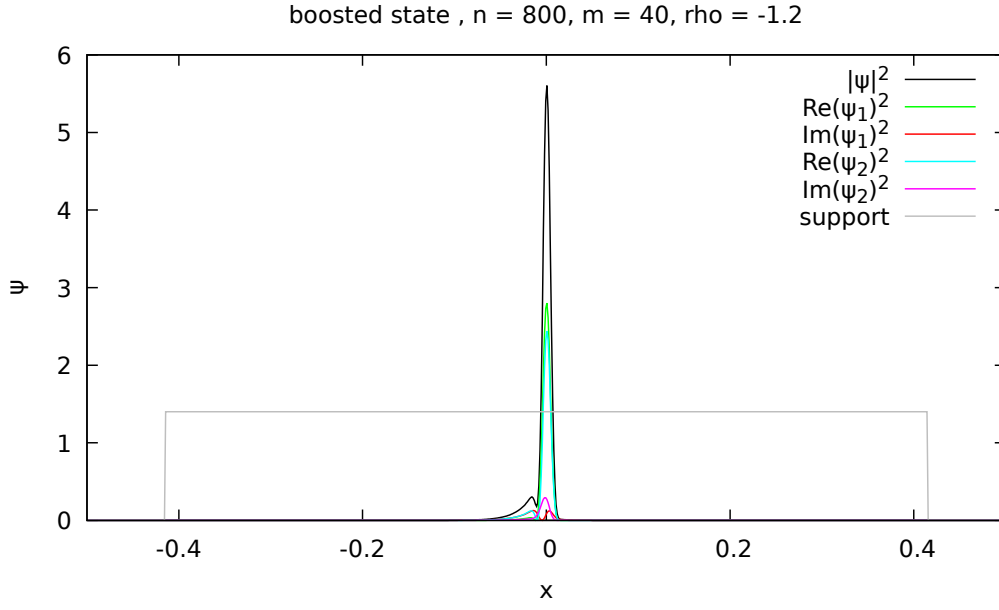Figure 21: Boosted state $\rho = 1.8$. Log scale.

15

Figure 22: Boosted state $\rho = -0.2$.

Figure 23: Boosted state $\rho = -0.2$. Log scale.

16

Figure 24: Boosted state $\rho = -0.4$.

Figure 25: Boosted state $\rho = -0.4$. Log scale.

17

Figure 26: Boosted state $\rho = -0.6$.

Figure 27: Boosted state $\rho = -0.6$. Log scale.
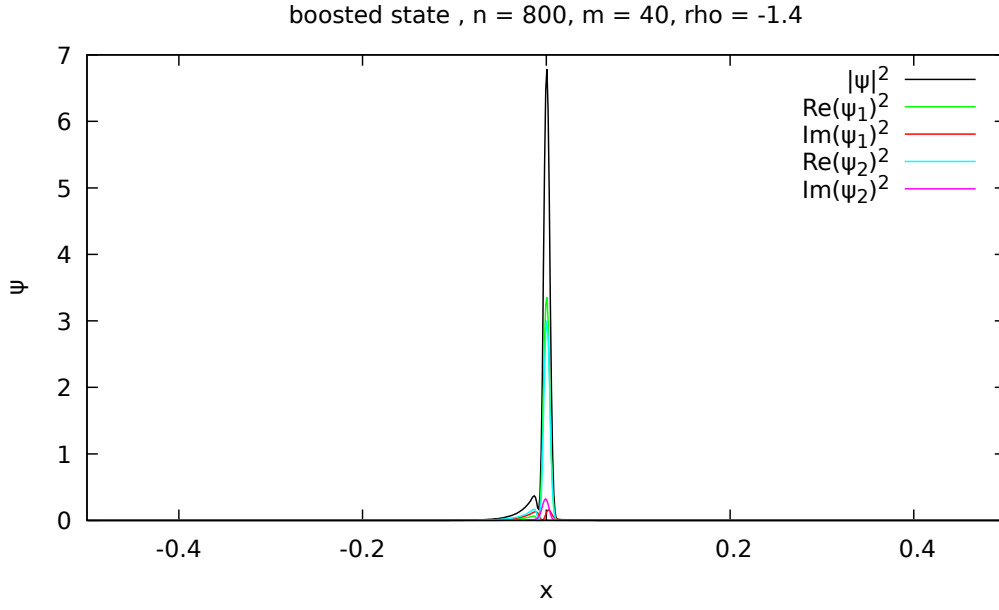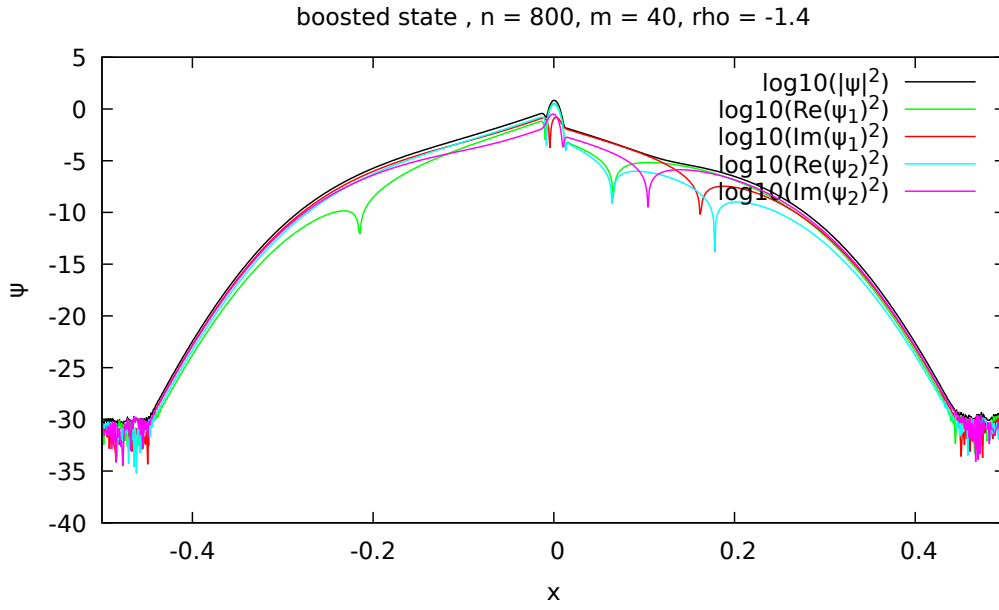
18

Figure 28: Boosted state $\rho = -0.8$.

Figure 29: Boosted state $\rho = -0.8$. Log scale.

Figure 30: Boosted state $\rho = -1.0$.

Figure 31: Boosted state $\rho = -1.0$. Log scale.

20

boosted state , n = 800, m = 40, rho = -1.2

Figure 32: Boosted state $\rho = -1.2$.

boosted state , n = 800, m = 40, rho = -1.2
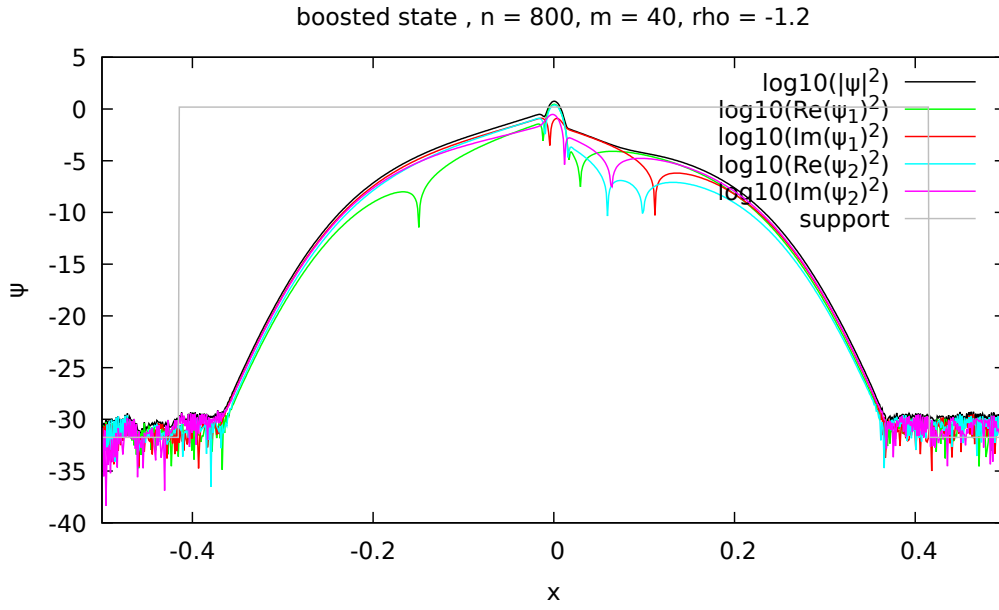
Figure 33: Boosted state $\rho = -1.2$. Log scale.

Figure 34: Boosted state $\rho = -1.4$.

Figure 35: Boosted state $\rho = -1.4$. Log scale.

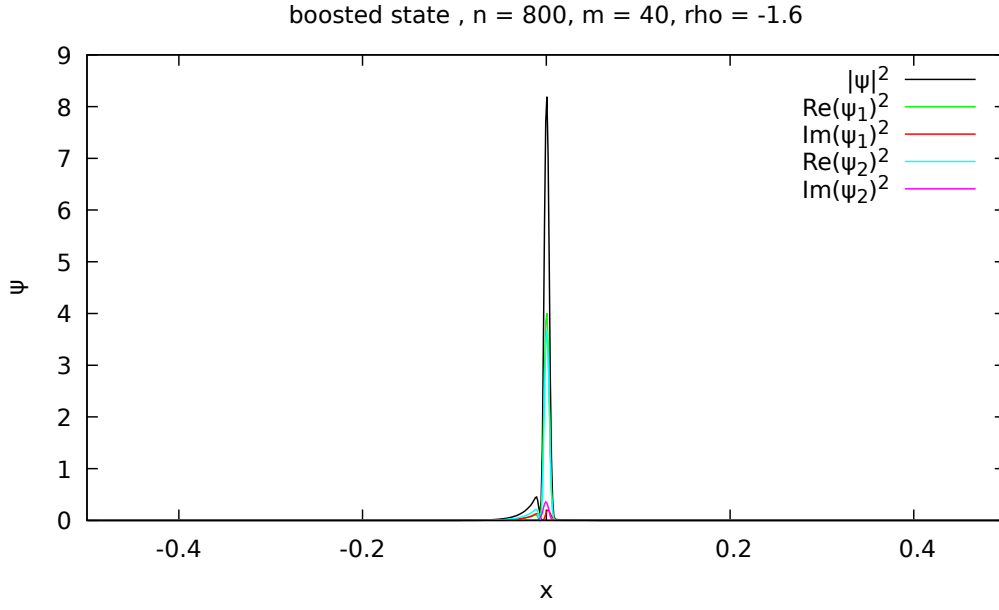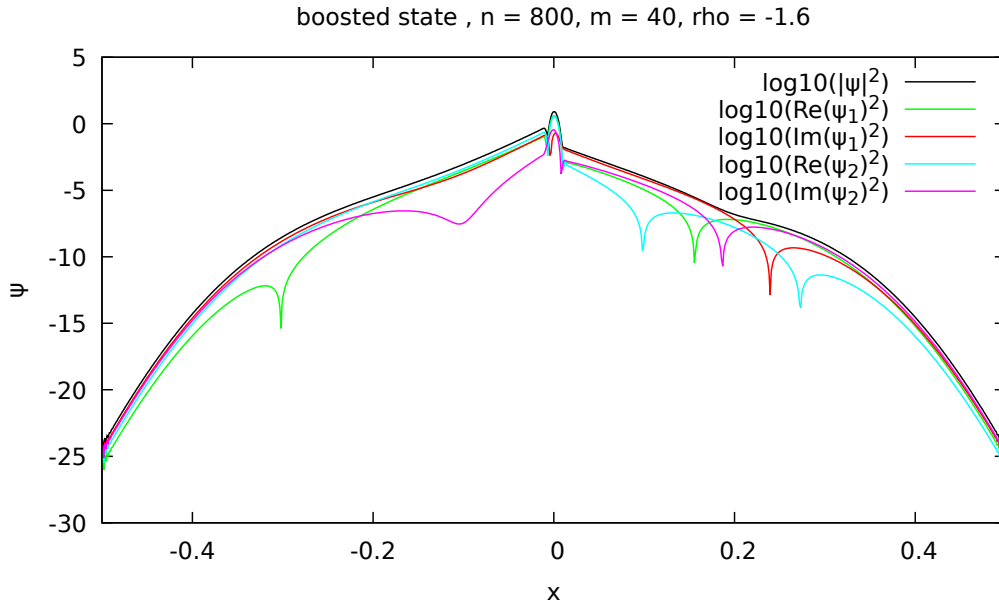Figure 36: Boosted state $\rho = -1.6$.
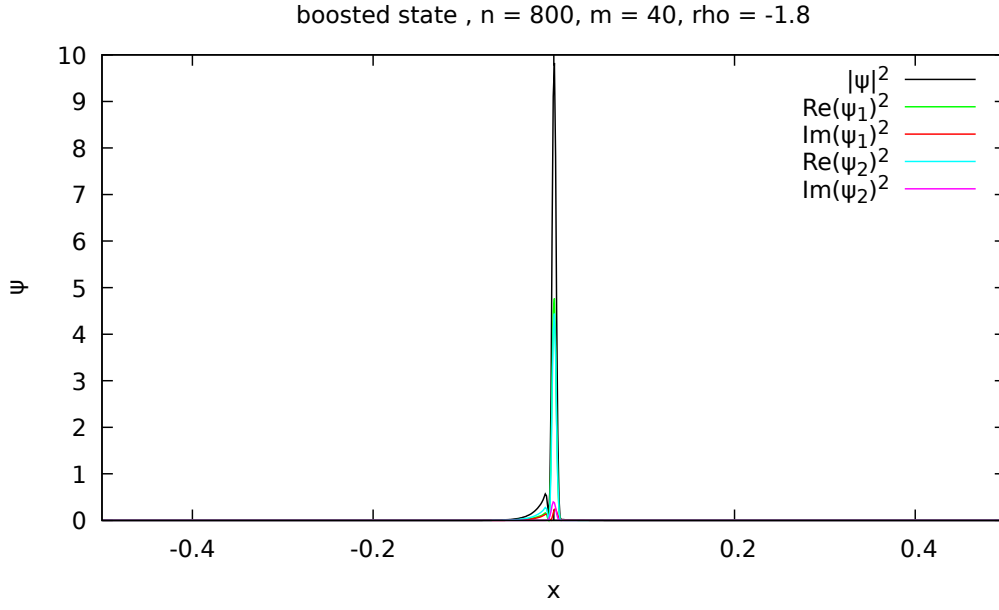

Figure 37: Boosted state $\rho = -1.6$. Log scale.

boosted state , n = 800, m = 40, rho = -1.8

Figure 38: Boosted state $\rho = -1.8$.
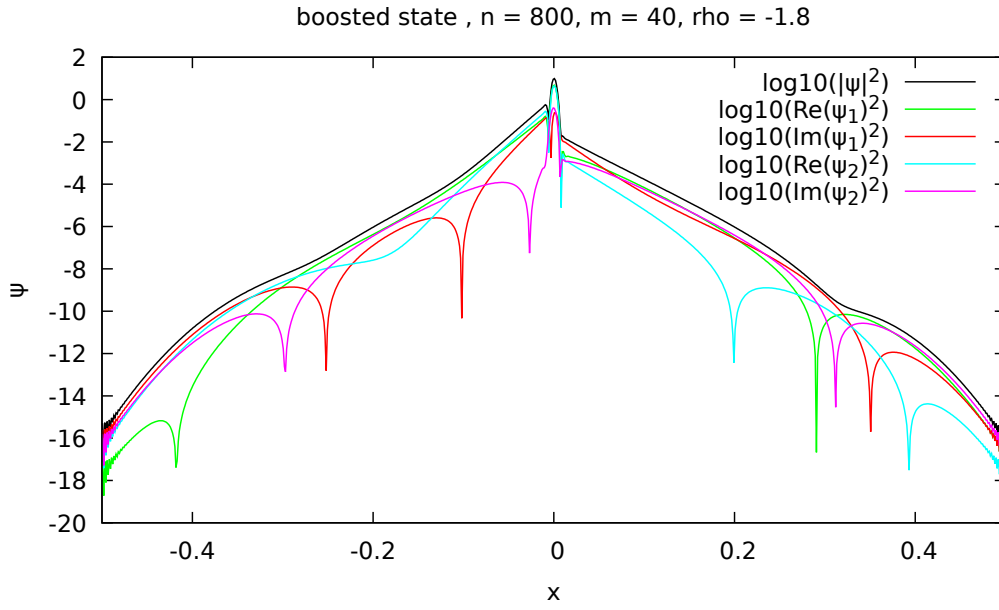

boosted state , n = 800, m = 40, rho = -1.8

Figure 39: Boosted state $\rho = -1.8$. Log scale.

References

[1] Lynn H. Loomis and Shlomo Sternberg: Advanced Calculus, Addison-Wesley 1968

last modification 2020-06-09