

Kacper Kulczycki

Dynamika kwantowa naładowanej cząstki w jednowymiarowym pudle, w zmiennym polu elektrycznym

Warszawa 2007

Spis treści:

<u>Spis treści</u>	<u>1</u>
<u>Wstęp</u>	<u>2</u>
<u>Teoria</u>	<u>2</u>
<u>Algorytm</u>	<u>2</u>
<u>Symulacje</u>	<u>4</u>
<u>Wyniki</u>	<u>8</u>
<u>Wnioski</u>	<u>75</u>
<u>Dodatek 1</u>	<u>75</u>
<u>Dodatek 2</u>	<u>82</u>
<u>Dodatek 2a</u>	<u>82</u>
<u>Dodatek 2b</u>	<u>83</u>
<u>Dodatek 2c</u>	<u>84</u>
<u>Dodatek 2d</u>	<u>86</u>
<u>Dodatek 2e</u>	<u>92</u>
<u><i>Dodatek 2eI</i></u>	<u>93</u>
<u><i>Dodatek 2eII</i></u>	<u>94</u>
<u>Dodatek 3</u>	<u>95</u>
<u>Dodatek 4</u>	<u>95</u>
<u>Dodatek 4a</u>	<u>95</u>
<u>Dodatek 4b</u>	<u>96</u>
<u>Dodatek 5</u>	<u>96</u>
<u>Dodatek 5a</u>	<u>96</u>
<u>Dodatek 5b</u>	<u>96</u>
<u>Dodatek 5c</u>	<u>96</u>
<u>Dodatek 5d</u>	<u>97</u>
<u>Dodatek 5e</u>	<u>97</u>
<u>Dodatek 5f</u>	<u>97</u>
<u>Dodatek 6</u>	<u>97</u>
<u>Dodatek 7</u>	<u>98</u>
<u>Dodatek 8</u>	<u>98</u>

Wstęp

Ćwiczenie polegało na symulacji kwantowej cząstki, opisywanej zależnym od czasu, równaniem Schrödingera (jednowymiarowym).

Wymagało to napisania odpowiedniego programu, całkującego numerycznie równanie Schrödingera, oraz przeprowadzenia przy jego pomocy symulacji.

Teoria

Model wykorzystywany w ćwiczeniu opisuje jednowymiarowy elektron, zamknięty w pudle potencjału o ściankach, o nieskończenie wielkiej energii. Efekty relatywistyczne, jak i spin elektronu zostały w nim zaniedbane. Elektron oddziaływał z polem elektrycznym, o sinusoidalnie zmiennej w czasie amplitudzie. Dla ułatwienia konstrukcji algorytmu, wygodne jest przejście do zmiennych bezwymiarowych, w których operator hamiltona ma postać:

$$\hat{H}(x, \tau) = -\frac{1}{2} \frac{\partial^2}{\partial x^2} + \kappa \left(x - \frac{1}{2}\right) \sin(\omega \tau).$$

A równanie Schrödingera przybiera postać:

$$i \frac{\partial \Psi(x, \tau)}{\partial \tau} = \hat{H}(x, \tau) \Psi(x, \tau).$$

Oczywiście, w tych równaniach obowiązują dodatkowo następujące własności funkcji falowej, wynikające z warunków brzegowych i normalizacyjnych:

$$\Psi(0, \tau) = \Psi(1, \tau) = 0, \quad x \in (0; 1),$$

$$\int_0^1 \Psi(x, \tau)^* \Psi(x, \tau) dx = 1.$$

Stąd łatwo zauważyć że charakterystyczne wielkości będą miały postać:

- gęstość prawdopodobieństwa:

$$\rho(x, \tau) = \Psi(x, \tau)^* \Psi(x, \tau),$$

- średnie położenie:

$$\langle x \rangle(\tau) = \int_0^1 \Psi(x, \tau)^* x \Psi(x, \tau) dx,$$

- energia (wyrażona w znormalizowanych jednostkach):

$$\langle \epsilon \rangle(\tau) = \int_0^1 \Psi(x, \tau)^* \hat{H}(x, \tau) \Psi(x, \tau) dx.$$

Energie stanów stacjonarnych w tych jednostkach przyjmują natomiast wartości:

$$\epsilon_n = \frac{n^2 \pi^2}{2}.$$

A odpowiadające im funkcje falowe:

$$\Psi_n(x, \tau) = \exp(-i \epsilon_n \tau) \sqrt{2} \sin(n \pi x).$$

Algorytm

Aby umożliwić stworzenie algorytmu numerycznego, konieczna jest dyskretyzacja przestrzeni położenia:

$$k \in \overline{0, N} \Rightarrow \Delta x = \frac{1}{N} \Rightarrow x_k = k \Delta x.$$

Kolejnym zabiegiem jest rozbitcie zespolonej funkcji falowej w reprezentacji położenia, na część urojoną i rzeczywistą, oraz zapisanie ich wartości w odpowiednich tablicach, zgodnych z wprowadzoną dyskretyzacją przestrzeni położenia:

$$\Psi(x_k, \tau) = i \operatorname{Im} \Psi_k + \operatorname{Re} \Psi_k = i \Psi_k^I + \Psi_k^R, \quad k \in \overline{0, N}.$$

Inicjalizacyjne wartości funkcji falowej, powinny mieć postać odpowiadającą stanom stacjonarnym:

$$\Psi_k^R = \sqrt{2} \sin(n\pi x_k), \quad \Psi_k^I = 0, \quad k \in \overline{0, N}.$$

W algorytmie numerycznym, w hamiltonianie, operator pochodnej drugiego stopnia działający na funkcję falową, można przybliżyć ilorazem różnicowym:

$$-\frac{1}{2} \frac{\partial^2}{\partial x^2} \Psi(x_k) \rightarrow -\frac{1}{2} \frac{\Psi_{k+1} + \Psi_{k-1} - 2\Psi_k}{\Delta x^2}.$$

Hamiltonian przyjmuje więc postać:

$$H_k^R = \hat{H} \Psi^R(x_k) = -\frac{1}{2} \frac{\Psi_{k+1}^R + \Psi_{k-1}^R - 2\Psi_k^R}{\Delta x^2} + \kappa(x_k - \frac{1}{2}) \Psi_k^R \sin(\omega\tau),$$

$$H_k^I = \hat{H} \Psi^I(x_k) = -\frac{1}{2} \frac{\Psi_{k+1}^I + \Psi_{k-1}^I - 2\Psi_k^I}{\Delta x^2} + \kappa(x_k - \frac{1}{2}) \Psi_k^I \sin(\omega\tau).$$

Ale ze względu na warunki brzegowe, wzory te obowiązują tylko w środku pudła, czyli dla:

$$k \in \overline{1, N-1},$$

natomiast w pozostałych, brzegowych punktach, hamiltonian się zeruje:

$$H_k^I = H_k^R = 0 \quad k \in \{0, N\}.$$

Równanie Schrödingera można zapisać następująco:

$$\frac{d \Psi_k^R(\tau)}{d\tau} = \text{Re}(\hat{H} \Psi_k(\tau)) = \text{Re}(\hat{H}(\Psi_k^R(\tau) + i \Psi_k^I(\tau))) = \hat{H} \Psi_k^I(\tau) = H_k^I(\tau),$$

$$\frac{d \Psi_k^I(\tau)}{d\tau} = \text{Im}(\hat{H} \Psi_k(\tau)) = \text{Im}(\hat{H}(\Psi_k^R(\tau) + i \Psi_k^I(\tau))) = -\hat{H} \Psi_k^R(\tau) = -H_k^R(\tau).$$

Z kolei, całkowanie tych równań, przeprowadzane metodą drugiego rzędu (z krokiem czasowym $\Delta\tau$) sprowadza się, do obliczenia następujących wielkości:

$$\Psi_k^R(\tau + \frac{\Delta\tau}{2}) = \Psi_k^R(\tau) + H_k^I(\tau) \frac{\Delta\tau}{2} \Rightarrow H_k^R(\tau + \frac{\Delta\tau}{2}),$$

$$\Psi_k^I(\tau + \Delta\tau) = \Psi_k^I(\tau) - H_k^R(\tau + \frac{\Delta\tau}{2}) \Delta\tau \Rightarrow H_k^I(\tau + \Delta\tau),$$

$$\Psi_k^R(\tau + \Delta\tau) = \Psi_k^R(\tau + \frac{\Delta\tau}{2}) + H_k^I(\tau + \Delta\tau) \frac{\Delta\tau}{2}.$$

Jak widać, obliczenia można wykonywać tylko dla „wewnętrznych” punktów. Wynika to z tego, że ze względu na warunki brzegowe funkcja falowa jak i hamiltonian, zerują się w punktach brzegowych.

W każdym kroku charakterystyczne wielkości będą miały więc postać:

- norma:

$$N(\tau) = \Delta x \sum_{k=1}^{N-1} ((\Psi_k^R(\tau))^2 + (\Psi_k^I(\tau))^2),$$

- gęstość prawdopodobieństwa w danym punkcie:

$$\rho_k(\tau) = (\Psi_k^R(\tau))^2 + (\Psi_k^I(\tau))^2,$$

- średnie położenie:

$$\langle x \rangle(\tau) = \Delta x \sum_{k=1}^{N-1} ((\Psi_k^R(\tau))^2 + (\Psi_k^I(\tau))^2),$$

- średnia energia:

$$\epsilon(\tau) = \Delta x \sum_{k=1}^{N-1} (\Psi_k^R(\tau) H_k^R(\tau) + \Psi_k^I(\tau) H_k^I(\tau)).$$

Symulacje

Test poprawności programu:

Test 1:

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.0001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=0$ (znormalizowana częstość pola elektrycznego),
 $\kappa=0$ (znormalizowana amplituda pola elektrycznego),
 $T=8000$ (czas trwania symulacji, w ilości kroków),
 $T_d=20$ (ilość kroków, co którą dane zapisywane były do pliku).

Test 2:

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000005$ (znormalizowany krok czasowy całkowania),
 $n=4$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=0$ (znormalizowana częstość pola elektrycznego),
 $\kappa=0$ (znormalizowana amplituda pola elektrycznego),
 $T=4000$ (czas trwania symulacji, w ilości kroków),
 $T_d=20$ (ilość kroków, co którą dane zapisywane były do pliku).

Test 3:

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=9$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=0$ (znormalizowana częstość pola elektrycznego),
 $\kappa=0$ (znormalizowana amplituda pola elektrycznego),
 $T=8000$ (czas trwania symulacji, w ilości kroków),
 $T_d=20$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacje przejść między poziomami energetycznymi, pod wpływem pola elektrycznego:

Symulacja 1 (spodziewane przejście 1→2):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=14.8044066 \cong 3\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=10000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 2 (brak spodziewanego przejścia):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=19.7392088 \cong 4\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=100000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 3 (spodziewane przejście 2→3):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=2$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=24.674011\cong 5\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=10000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 4 (spodziewane przejście 3→4):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=3$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=34.5436154\cong 7\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=10000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 5 (spodziewane przejście 4→5):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=4$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=44.41321981\cong 9\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=10000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 6 (spodziewane przejście 1→3):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=39.4784176\cong 8\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=120000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 7 (spodziewane przejście 1→5):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=118.4352528\cong 24\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=120000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 8 (spodziewane przejście 1→4):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=74.02203301\cong 15\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=140000000$ (czas trwania symulacji, w ilości kroków),

$T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 9 (spodziewane przejście 1→5):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega = 118.4352528 \approx 24\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=1$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 10 (spodziewane przejście 1→3):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=4$ (stan początkowy cząstki – poziom energetyczny),
 $\omega = 39.4784176 \approx 8\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=7$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 11 (brak spodziewanego przejścia):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega = 59.21762641 \approx 12\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=7$ (znormalizowana amplituda pola elektrycznego),
 $T=100000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 12 (spodziewane przejście 1→3):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=4$ (stan początkowy cząstki – poziom energetyczny),
 $\omega = 39.4784176 \approx 8\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=7$ (znormalizowana amplituda pola elektrycznego),
 $T=100000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 13 (brak spodziewanego przejścia):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega = 29.6088132 \approx 6\pi^2/2$ (znormalizowana częstość pola elektrycznego),
 $\kappa=7$ (znormalizowana amplituda pola elektrycznego),
 $T=100000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 14 (brak spodziewanego przejścia):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega = 78.95683521 \approx 16\pi^2/2$ (znormalizowana częstość pola elektrycznego),

$\kappa=7$ (znormalizowana amplituda pola elektrycznego),
 $T=10000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacje przejść między poziomami 1 i 2, pod wpływem pola elektrycznego, badanie okolic częstotliwości rezonansowej ($\omega_{\text{rez}}=14.8044066 \approx 3\pi^2/2$):

Symulacja 1 (90% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=13.32396594$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 2 (92% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=13.62005407$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 3 (94% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=13.9161422$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 4 (98% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=14.50831847$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 5 (100% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta \tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=14.8044066$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 6 (102% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=15.10049473$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 7 (104% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=15.39658286$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 8 (106% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=15.692671$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 9 (108% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=15.98875913$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Symulacja 10 (110% ω_{rez}):

$N=100$ (liczba elementów na które dzielone jest pudło),
 $\Delta\tau=0.000001$ (znormalizowany krok czasowy całkowania),
 $n=1$ (stan początkowy cząstki – poziom energetyczny),
 $\omega=16.28484726$ (znormalizowana częstość pola elektrycznego),
 $\kappa=5$ (znormalizowana amplituda pola elektrycznego),
 $T=20000000$ (czas trwania symulacji, w ilości kroków),
 $T_d=10000$ (ilość kroków, co którą dane zapisywane były do pliku).

Wyniki

W pierwszej, testowej serii symulacji uzyskałem następujące wyniki:

Test 1, wielkości charakterystyczne przedstawiają wykresy na rysunku fig.1.:

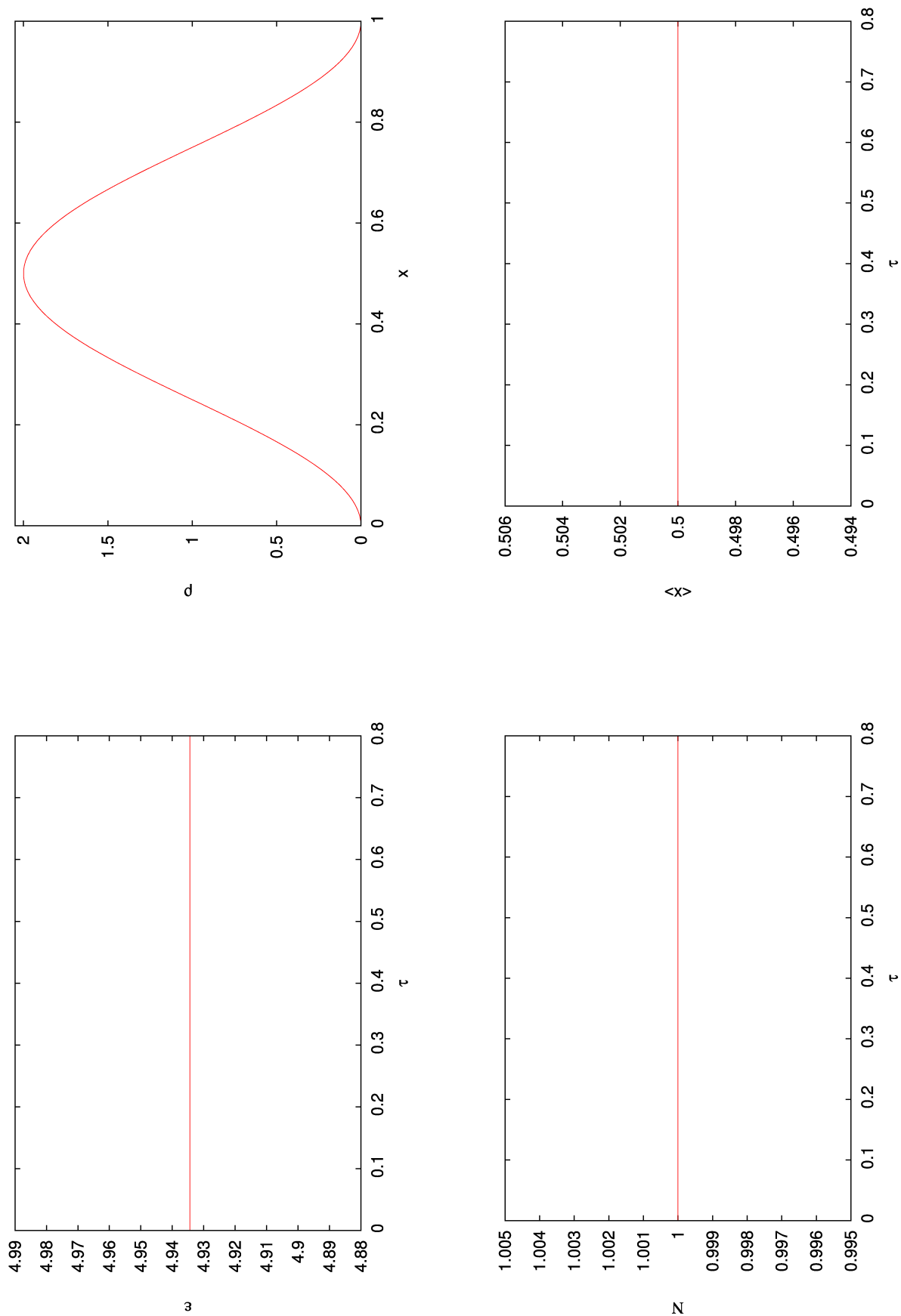


fig.1. Wykresy wielkości charakterystycznych, uzyskanych podczas pierwszej, testowej symulacji.

Test 2, wielkości charakterystyczne przedstawiają wykresy na rysunku fig.2.:

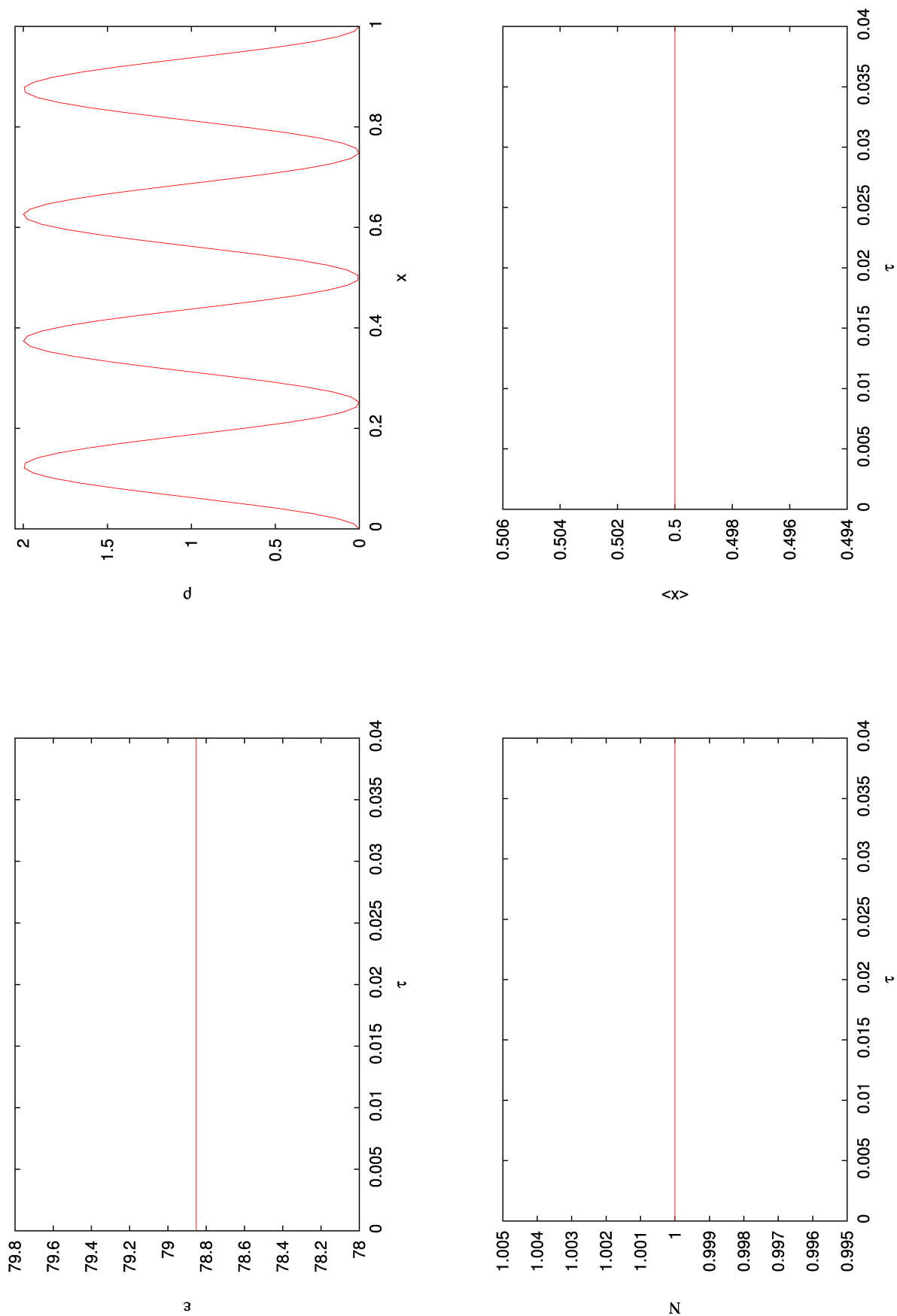


fig.2. Wykresy wielkości charakterystycznych, uzyskanych podczas drugiej, testowej symulacji.

Test 3, wielkości charakterystyczne przedstawiają wykresy na rysunku fig.3.:

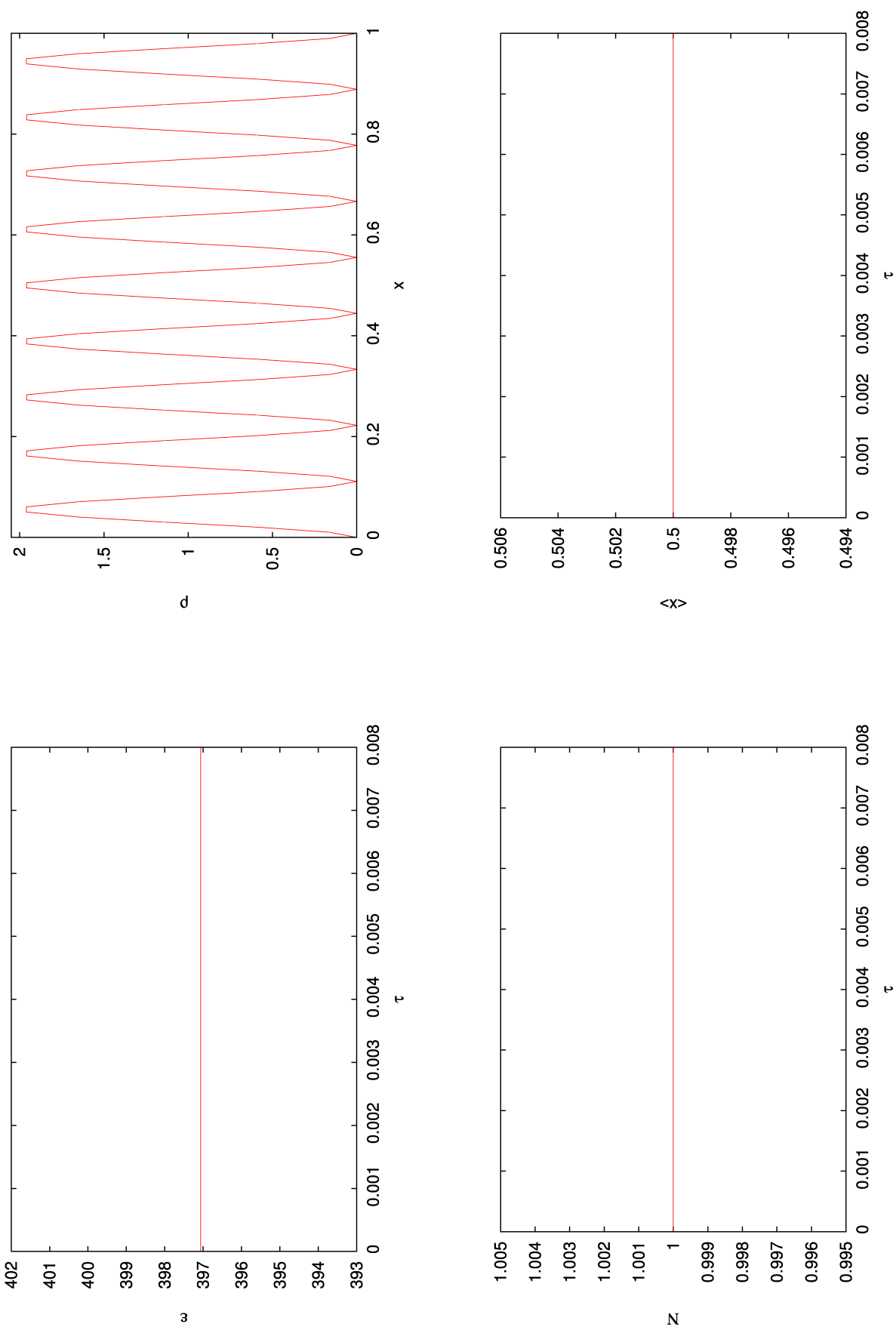


fig.3. Wykresy wielkości charakterystycznych, uzyskanych podczas trzeciej, testowej symulacji.

Jak widać, wielkości charakterystyczne uzyskały oczekiwane wartości, a algorytm zachowywał się stabilnie. Ze względu na to, że w teście trzecim (symulacji stanu stacjonarnego, dziewiątego poziomu energetycznego), konieczne było użycie czasowego kroku całkowania $\Delta\tau=0.000001$. Tej wartości, tego parametru, użyłem we wszystkich kolejnych symulacjach.

W drugiej serii symulacji (przejść między poziomami) uzyskałem następujące wyniki:

Seria 1:

Maksymalna wartość cząstki energii:

19.7584

Cztery wartości najbardziej zbliżone do maksimum:

19.7583

19.7566

19.7565

19.7562

Minimalna wartość cząstki energii:

4.91602

Cztery wartości najbardziej zbliżone do minimum:

4.91625

4.91686

4.91802

4.91836

Średnia wartość cząstki energii:

12.33721

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

12.4092

12.4044

12.3172

12.3153

Podczas symulacji obserwowane było, oczekiwane przejście między poziomami 1 i 2.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.4. oraz fig.5.:

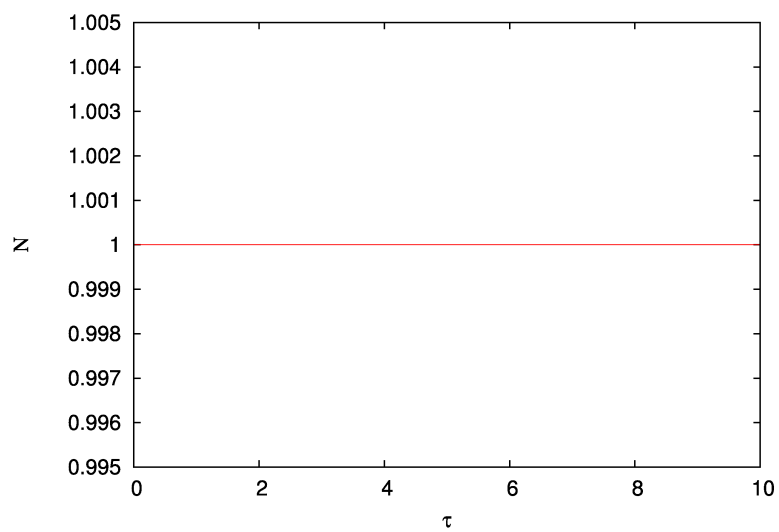
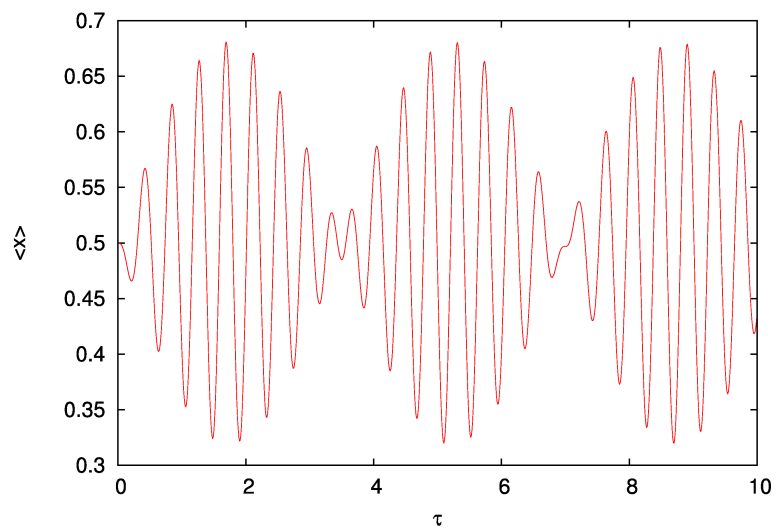
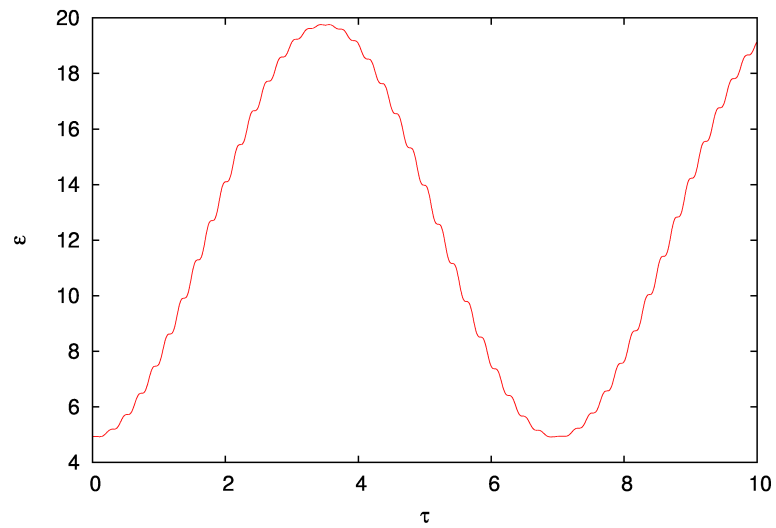


fig.4. Wykresy wielkości charakterystycznych, uzyskanych podczas pierwszej symulacji przejścia.

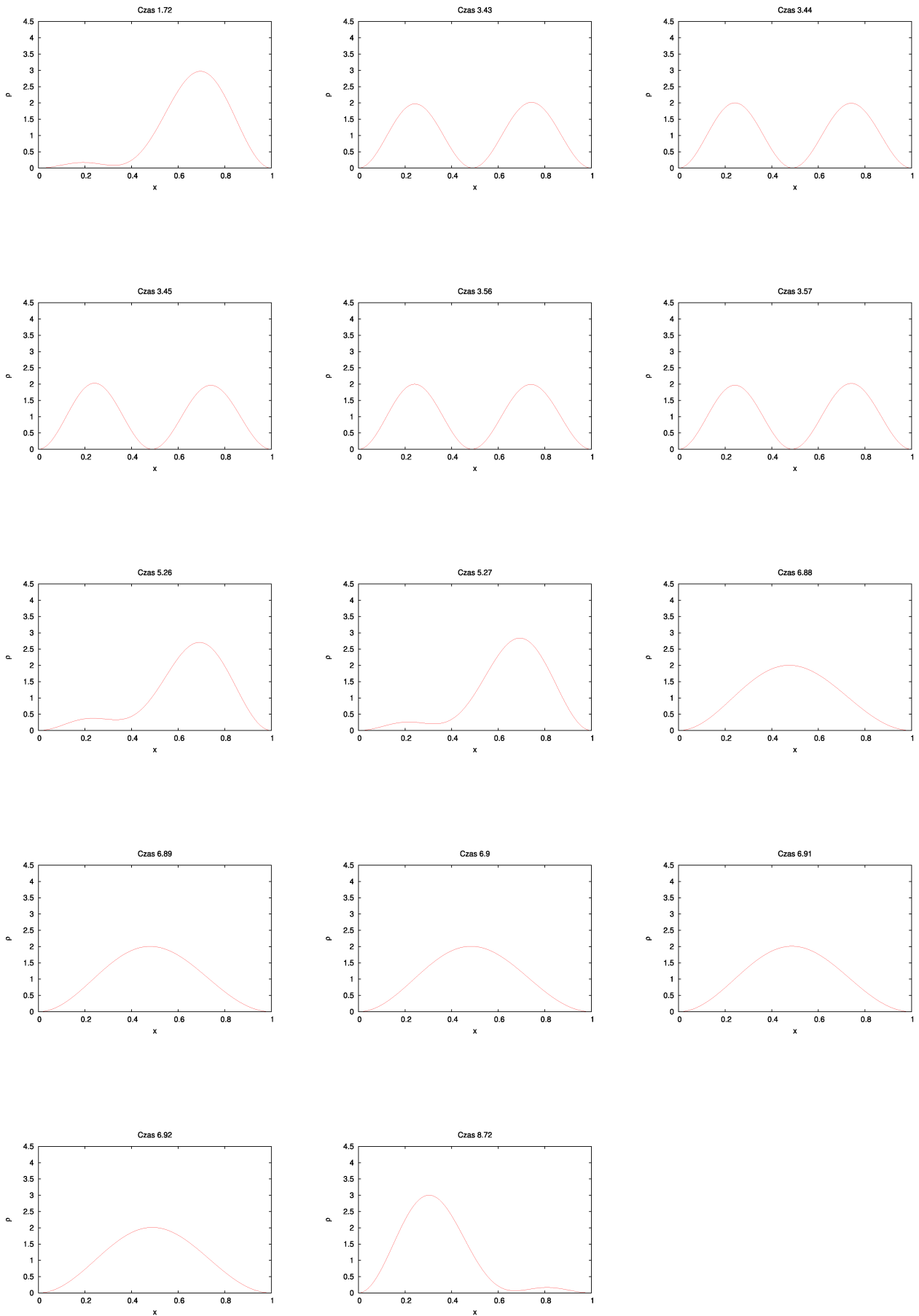


fig.5. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas pierwszej symulacji przejścia.

Seria 2:

Maksymalna wartość cząstki energii:

38.8736

Cztery wartości najbardziej zbliżone do maksimum:

38.8732

38.8731

38.8726

38.8723

Minimalna wartość cząstki energii:

4.91016

Cztery wartości najbardziej zbliżone do minimum:

4.91041

4.91158

4.91263

4.91447

Średnia wartość cząstki energii:

21.89188

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

21.8949

21.8939

21.8783

21.876

Podczas symulacji obserwowane było, **nie oczekiwane przejście, między poziomami 1 i 3**. W przypadku tej symulacji, częstość drgań pola jest połową częstości rezonansowej dla takiego przejścia. Efekt przejścia, jest dość nieoczekiwany, chociaż można go uzasadnić teoretycznym **modelem przejścia dwufotonowego**. Dodatkowo, należy zauważyć że **przejście** do trzeciego poziomu **było niepełne**. **Energia poziomu trzeciego**, w jednostkach zredukowanych, osiąga bowiem wartość **ok. 44.4**. Natomiast wartość **maksymalna energii**, uzyskana w tej symulacji wyniosła zaledwie **ok. 38.8**. Chociaż **ewolucja gęstości prawdopodobieństwa** wskazuje na **uzyskanie rozkładu bardzo bliskiego**, odpowiadającemu **trzeciemu poziomowi energetycznemu**.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.6. oraz fig.7.:

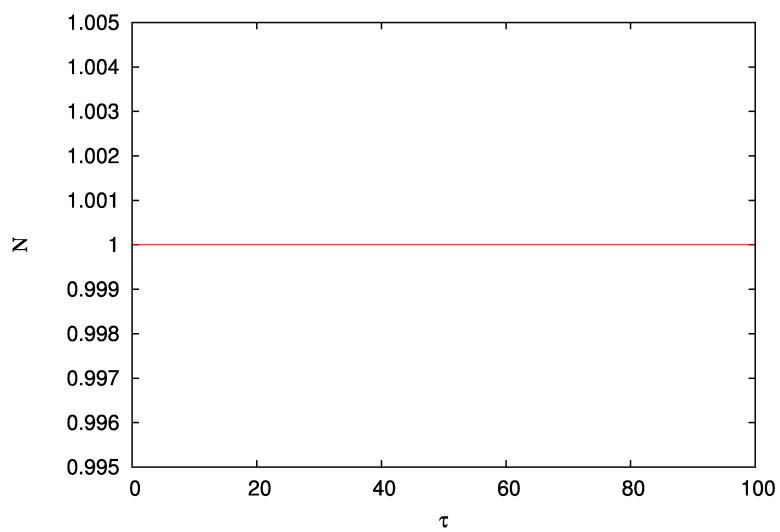
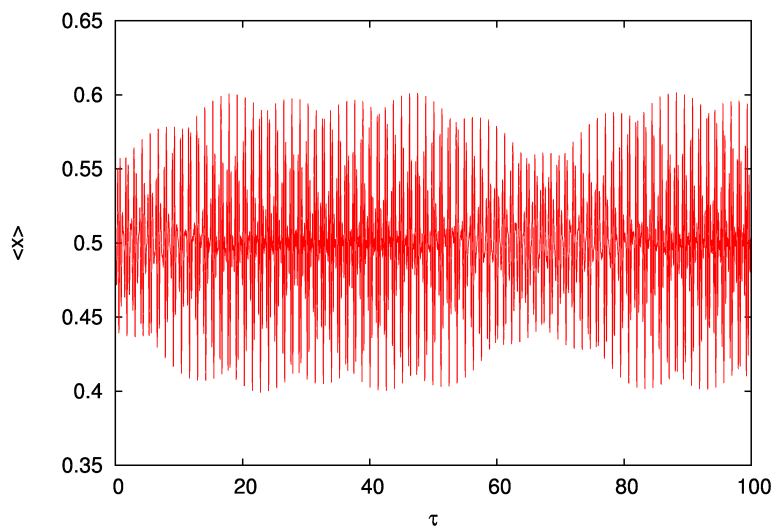
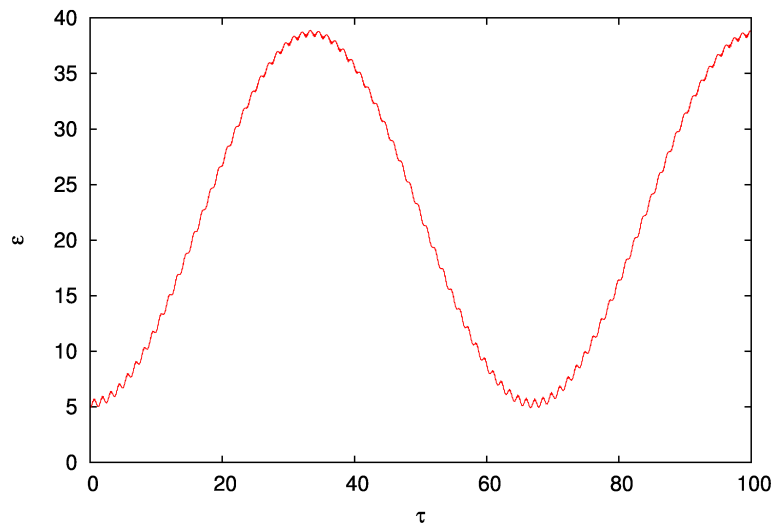


fig.6. Wykresy wielkości charakterystycznych, uzyskanych podczas drugiej symulacji przejścia.

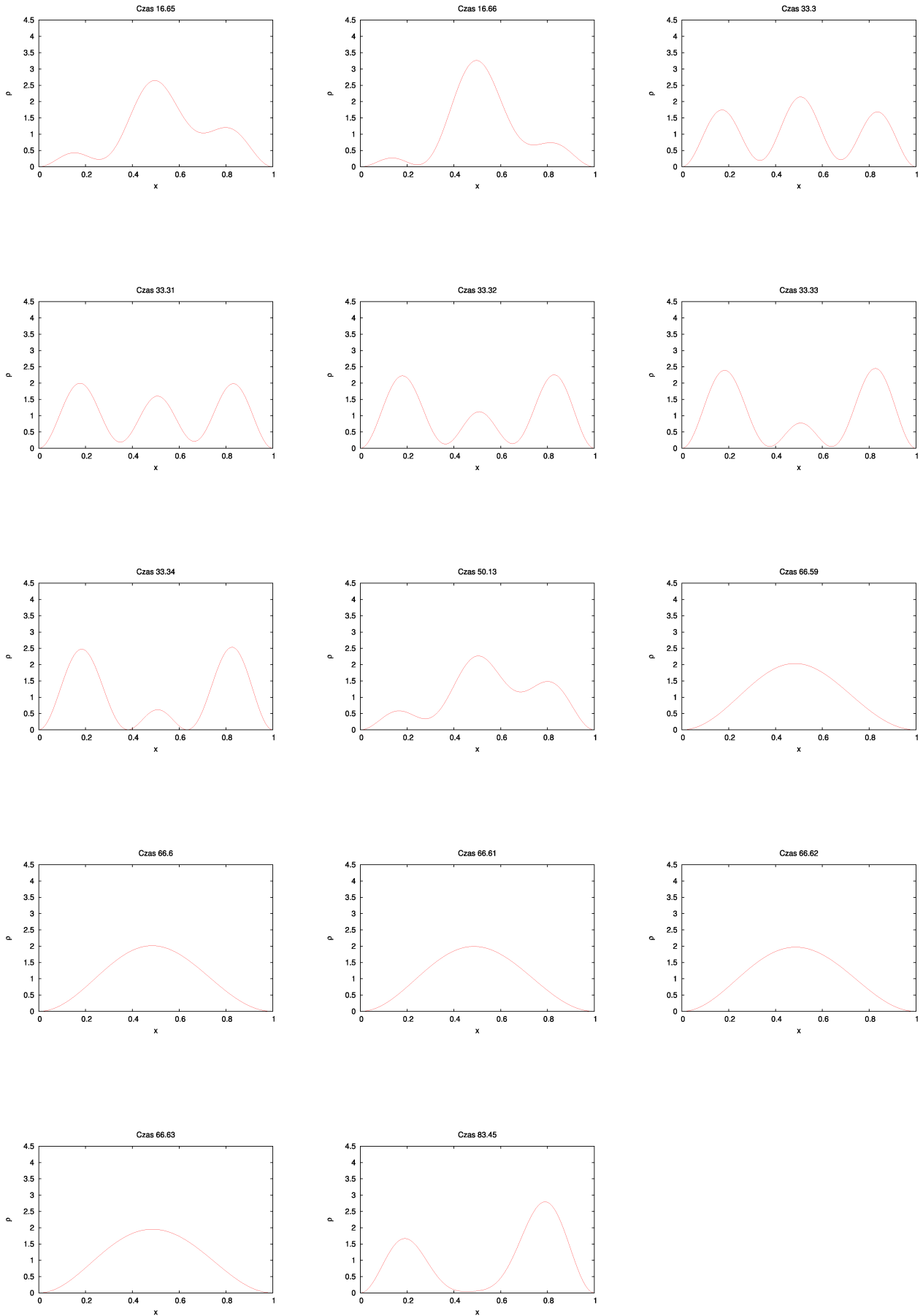


fig.7. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas drugiej symulacji przejścia.

Seria 3:

Maksymalna wartość cząstki energii:

44.3129

Cztery wartości najbardziej zbliżone do maksimum:

44.3128

44.3094

44.3084

44.3071

Minimalna wartość cząstki energii:

19.5017

Cztery wartości najbardziej zbliżone do minimum:

19.5062

19.5066

19.5192

19.5207

Średnia wartość cząstki energii:

31.9073

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

31.9964

31.9585

31.9011

31.8985

Podczas symulacji obserwowane było, oczekiwane przejście między poziomami 2 i 3.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.8. oraz fig.9.:

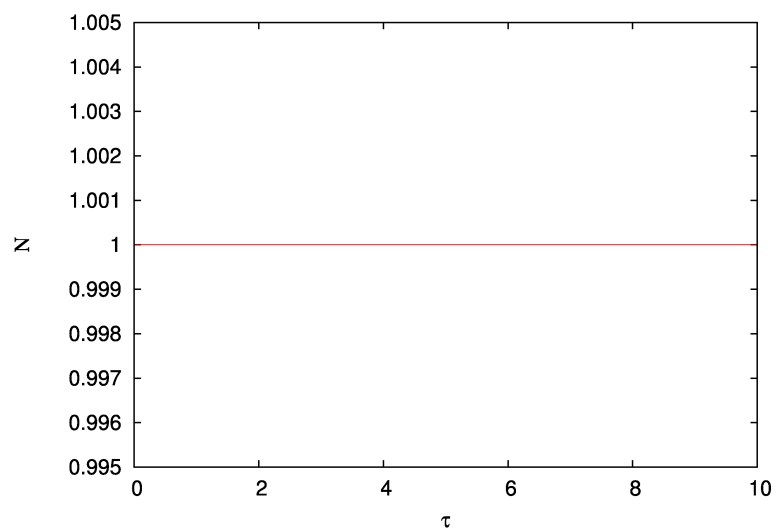
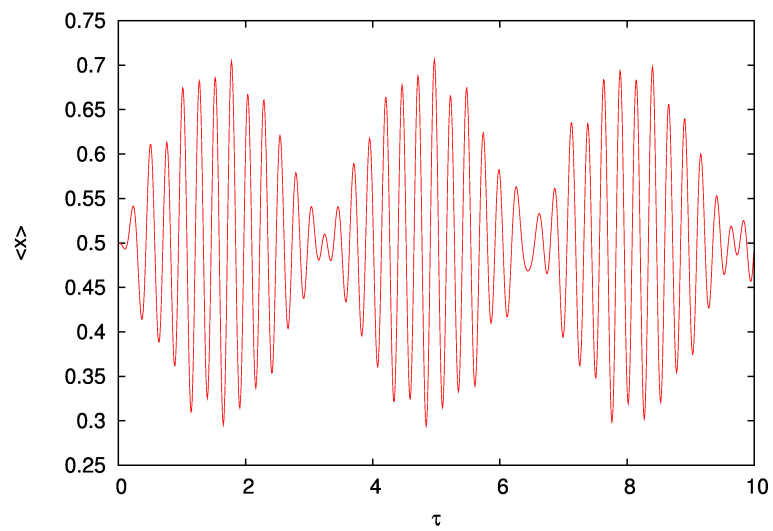
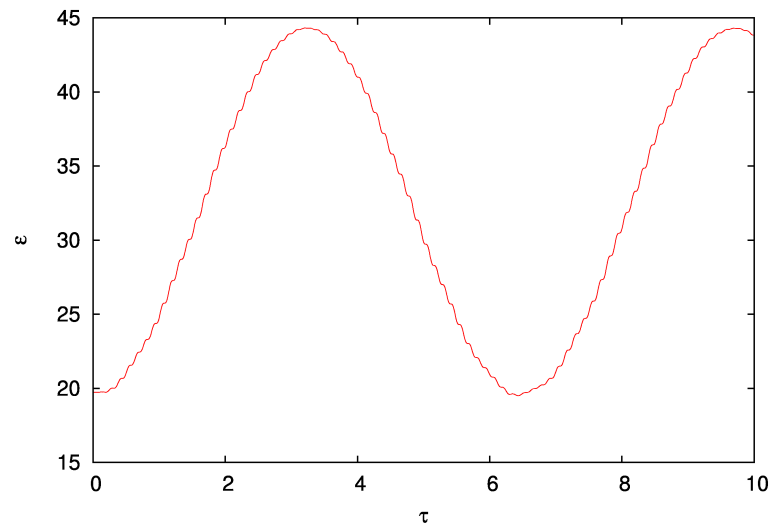


fig.8. Wykresy wielkości charakterystycznych, uzyskanych podczas trzeciej symulacji przejścia.

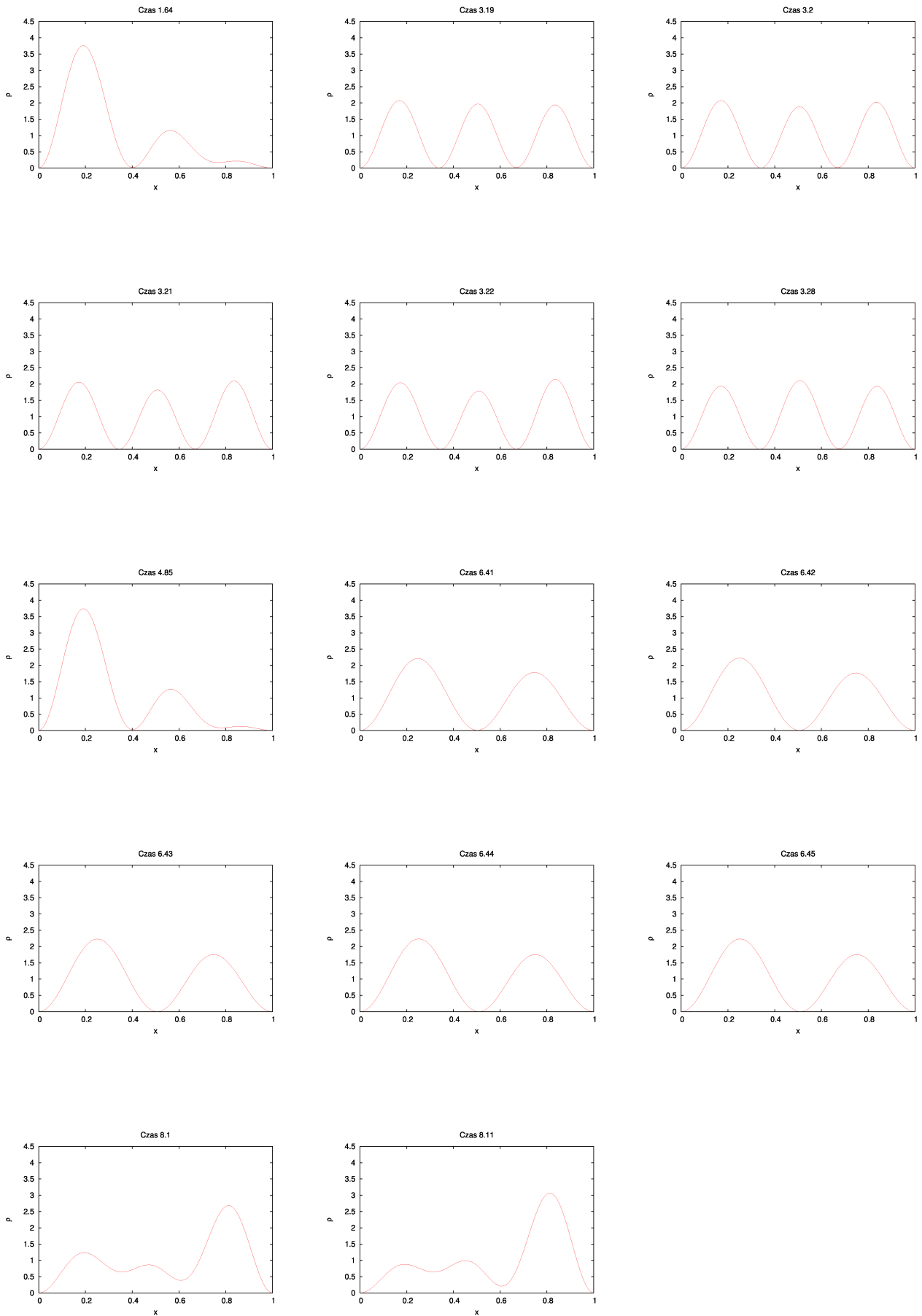


fig.9. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas trzeciej symulacji przejścia.

Seria 4:

Maksymalna wartość cząstki energii:

78.6473

Cztery wartości najbardziej zbliżone do maksimum:

78.6463

78.6458

78.6435

78.6408

Minimalna wartość cząstki energii:

43.9369

Cztery wartości najbardziej zbliżone do minimum:

43.9478

43.9512

43.9813

43.9862

Średnia wartość cząstki energii:

61.2921

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

61.3884

61.3057

61.2346

61.1639

Podczas symulacji obserwowane było, oczekiwane przejście między poziomami 3 i 4.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.10. oraz fig.11.:

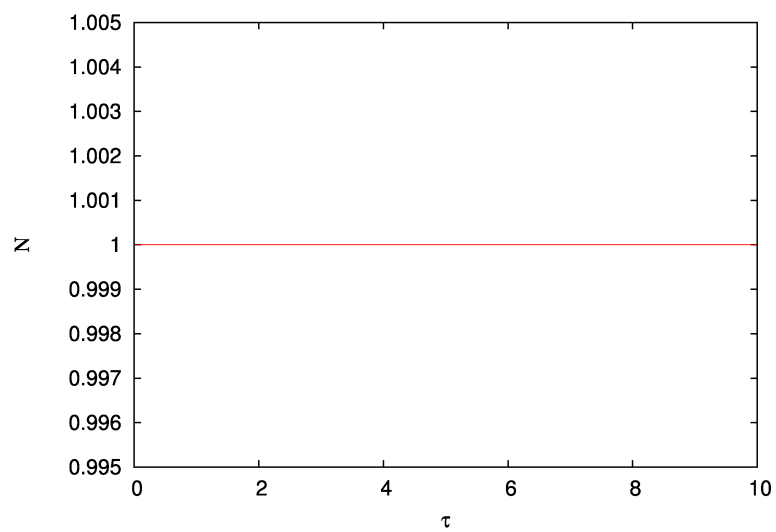
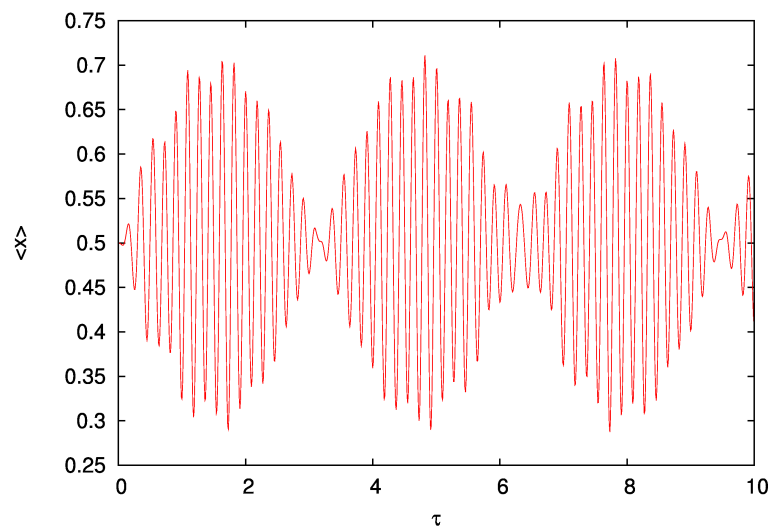
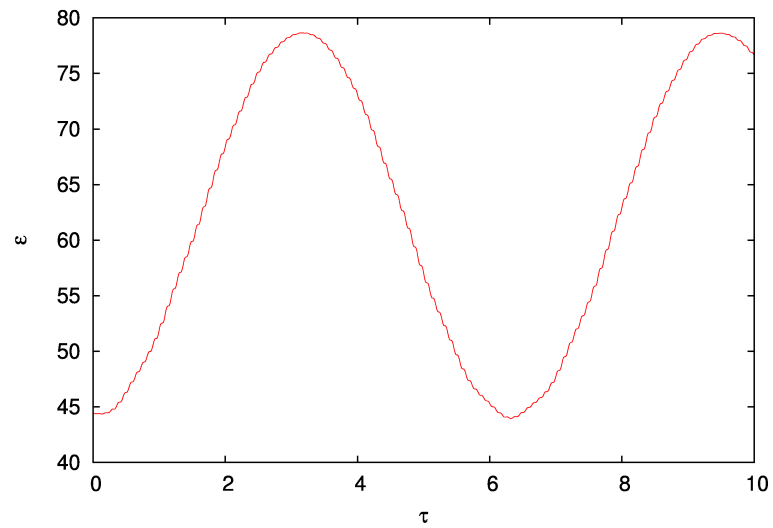


fig.10. Wykresy wielkości charakterystycznych, uzyskanych podczas czwartej symulacji przejścia.

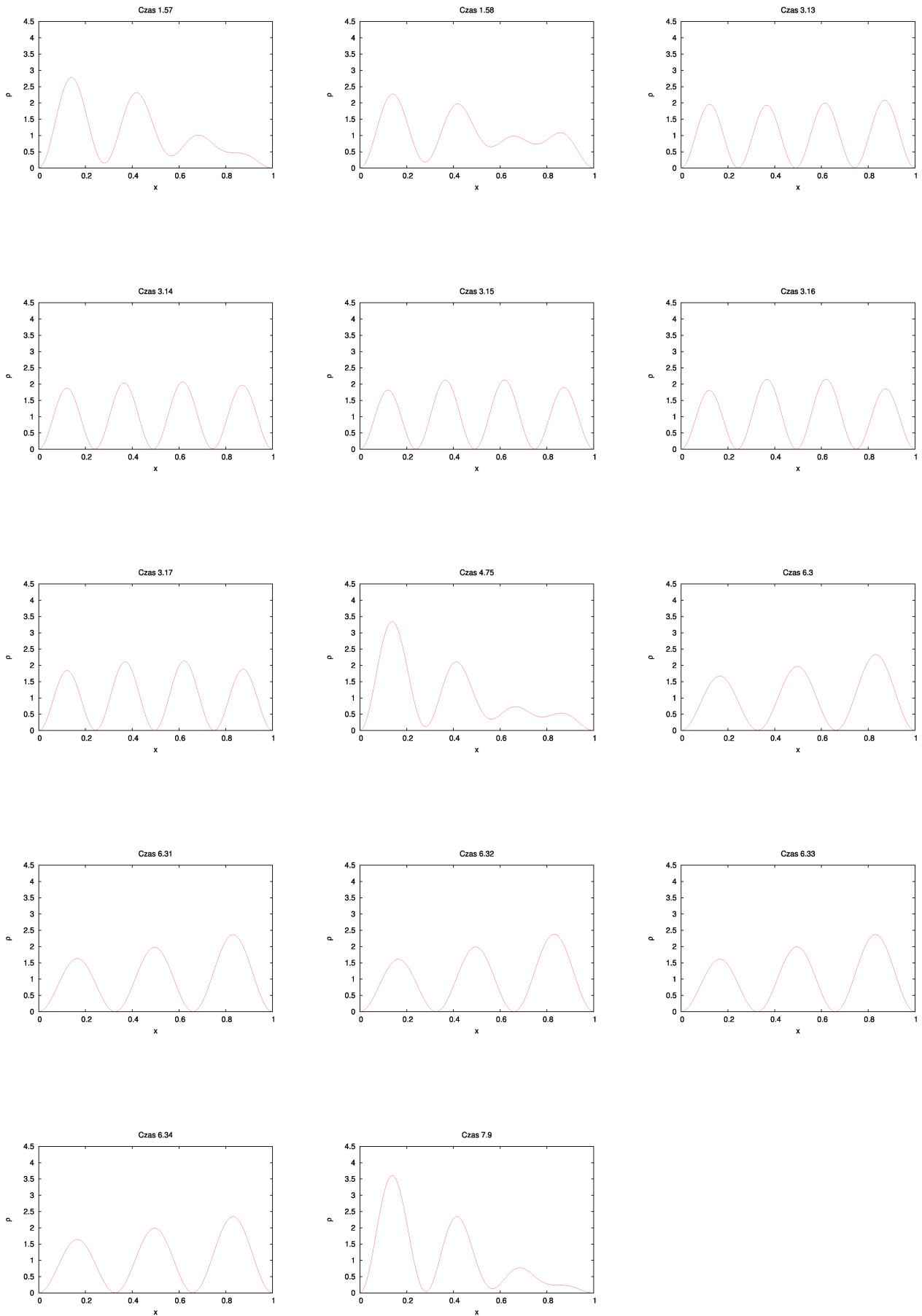


fig.11. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas czwartej symulacji przejścia.

Seria 5:

Maksymalna wartość cząstki energii:

122.267

Cztery wartości najbardziej zbliżone do maksimum:

122.256

122.243

122.238

122.234

Minimalna wartość cząstki energii:

78.3119

Cztery wartości najbardziej zbliżone do minimum:

78.3316

78.3334

78.3588

78.3753

Średnia wartość cząstki energii:

100.28945

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

100.47

100.397

100.082

100.074

Podczas symulacji obserwowane było, oczekiwane przejście między poziomami 4 i 5.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.12. oraz fig.13.:

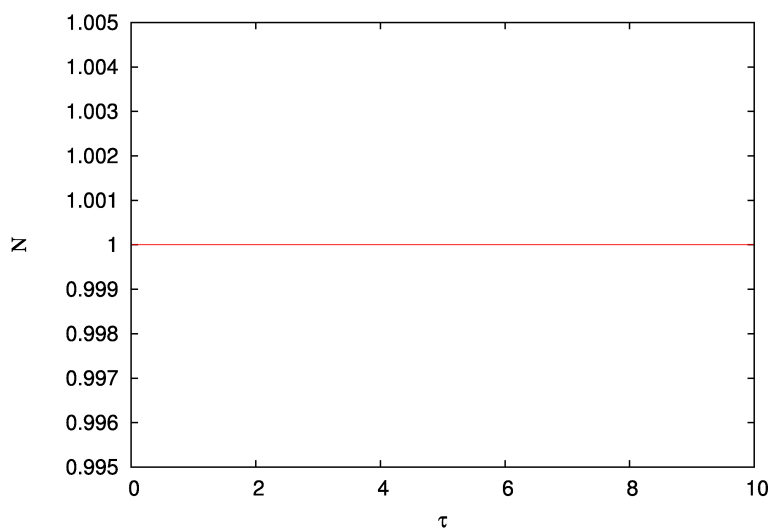
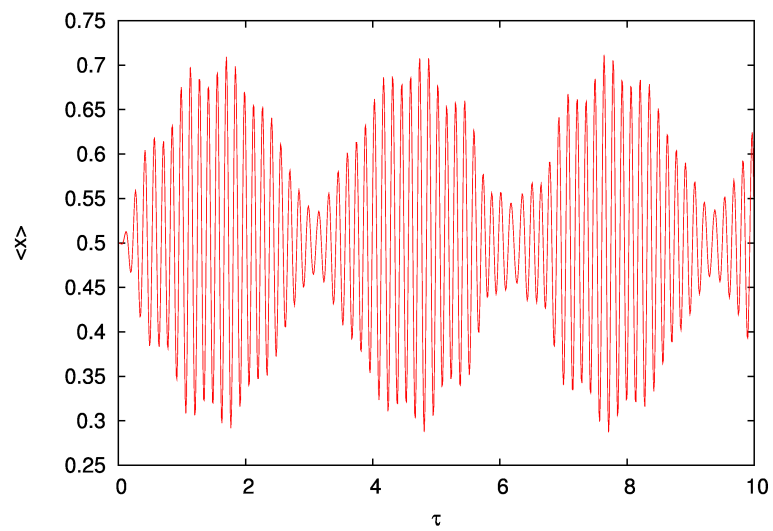
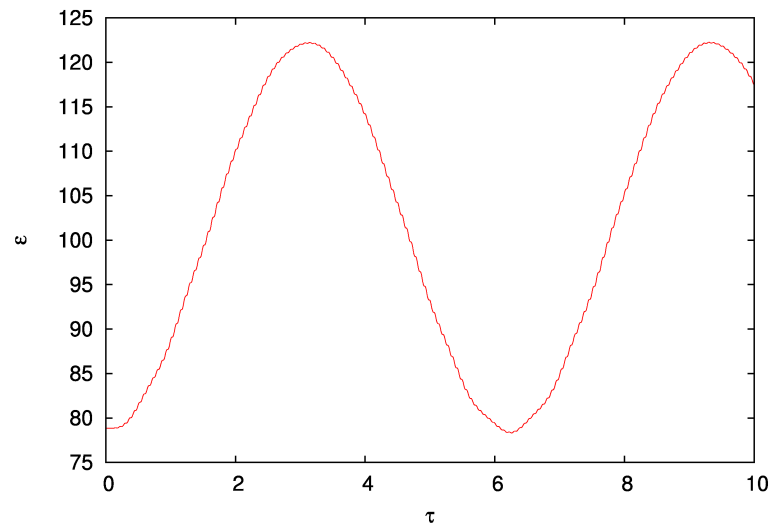


fig.12. Wykresy wielkości charakterystycznych, uzyskanych podczas piątej symulacji przejścia.

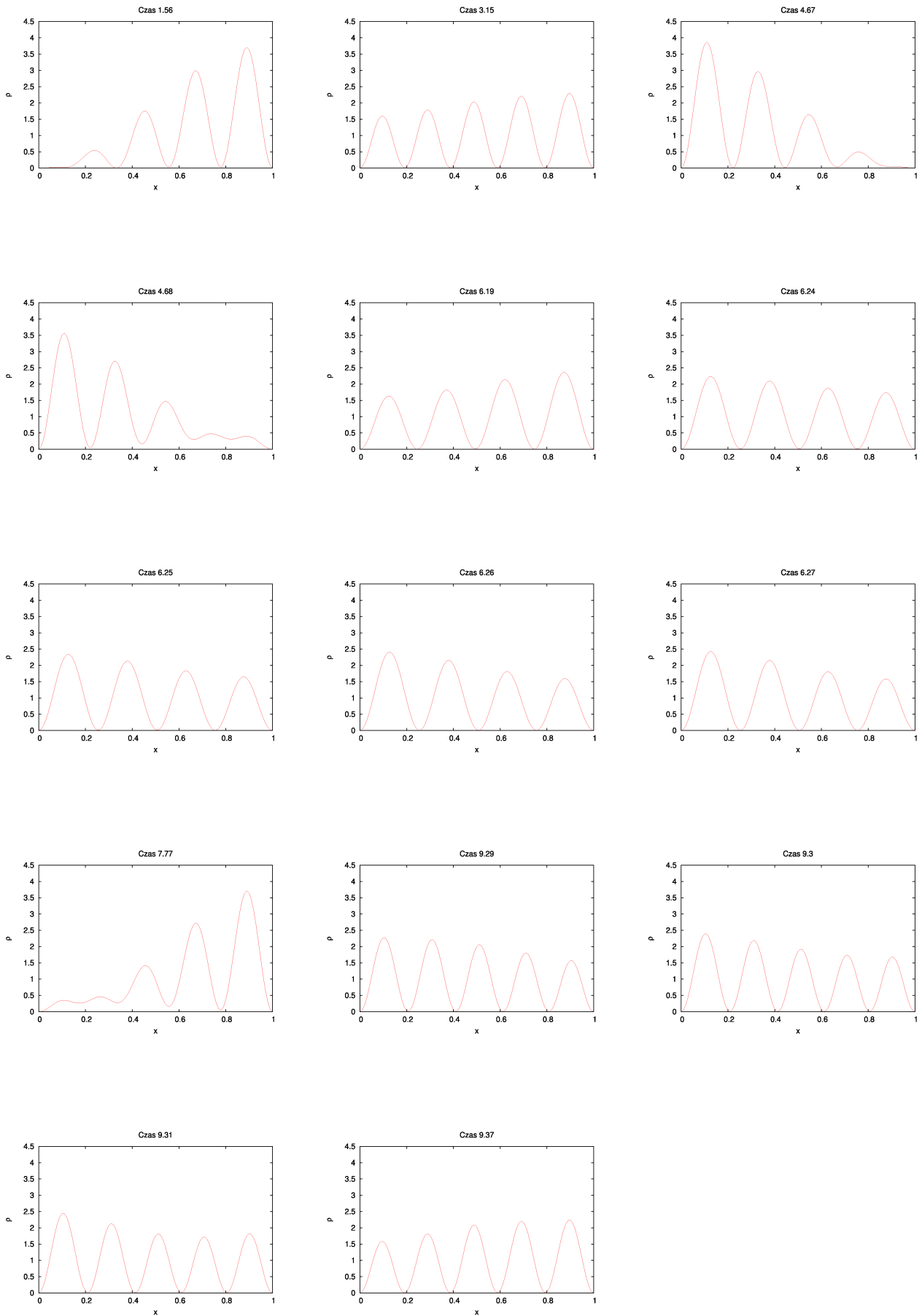


fig.13. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas piątej symulacji przejścia.

Seria 6:

Maksymalna wartość cząstki energii:

5.01977

Cztery wartości najbardziej zbliżone do maksimum:

5.01974

5.01971

5.0197

5.01966

Minimalna wartość cząstki energii:

4.90867

Cztery wartości najbardziej zbliżone do minimum:

4.90869

4.90871

4.90872

4.90874

Średnia wartość cząstki energii:

4.96422

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

4.96424

4.96423

4.96421

4.9642

Podczas symulacji **nie zostało zaobserwowane**, zgodnie z oczekiwaniami **przejście między poziomami 1 i 3**.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.14. oraz fig.15.:

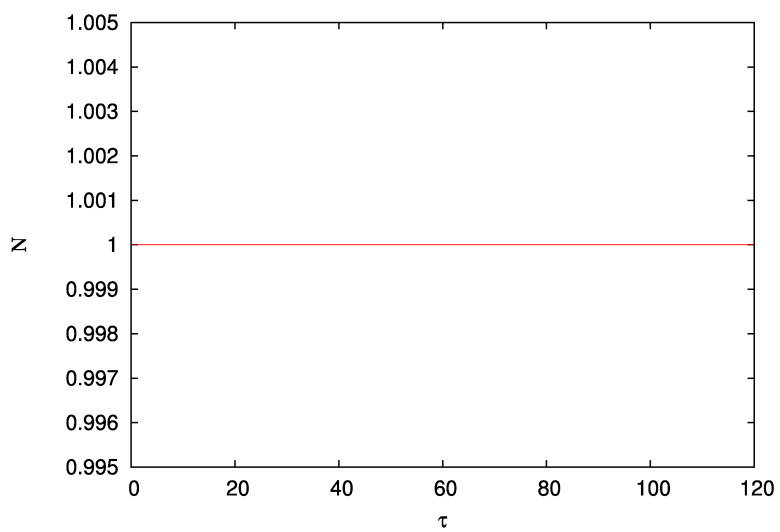
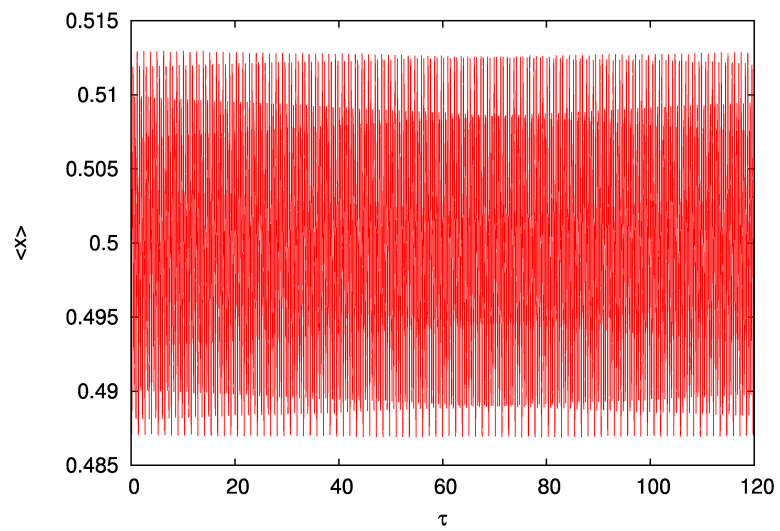
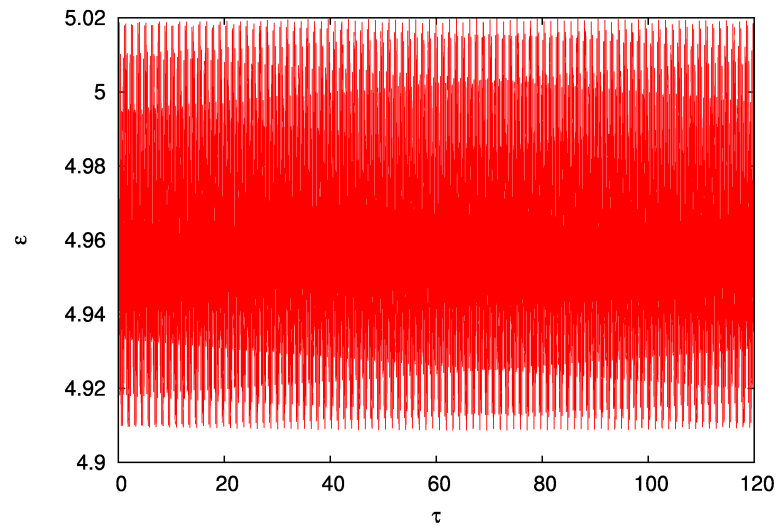


fig.14. Wykresy wielkości charakterystycznych, uzyskanych podczas szóstej symulacji przejścia.

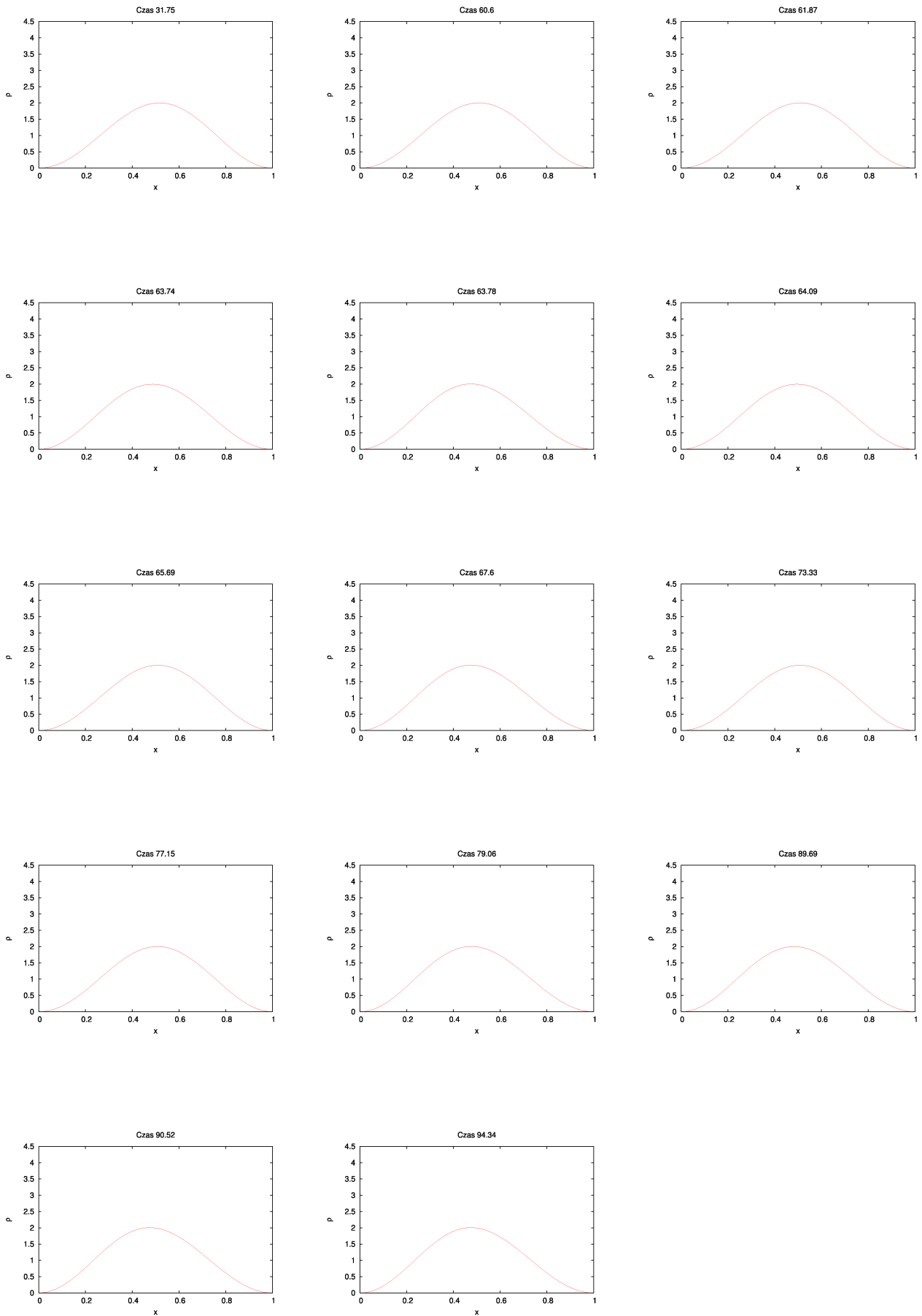


fig.15. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas szóstej symulacji przejścia.

Seria 7:

Maksymalna wartość cząstki energii:

4.95157

Cztery wartości najbardziej zbliżone do maksimum:

4.95157

4.95155

4.95152

4.9515

Minimalna wartość cząstki energii:

4.92289

Cztery wartości najbardziej zbliżone do minimum:

4.92292

4.92293

4.92297

4.92298

Średnia wartość cząstki energii:

4.93723

Cztery wartości najbardziej zbliżone do cząstki:

4.93725

4.93724

4.93722

4.93721

Podczas symulacji **nie zostało zaobserwowane**, zgodnie z oczekiwaniami **przejście między poziomami 1 i 5**.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.16. oraz fig.17.:

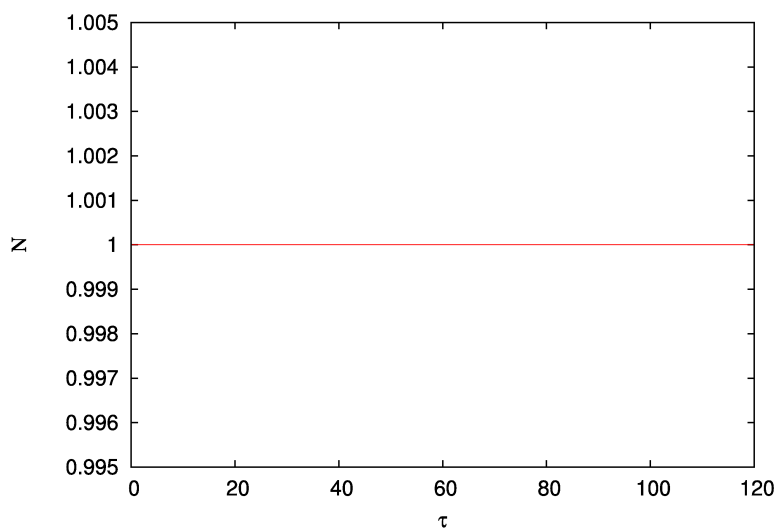
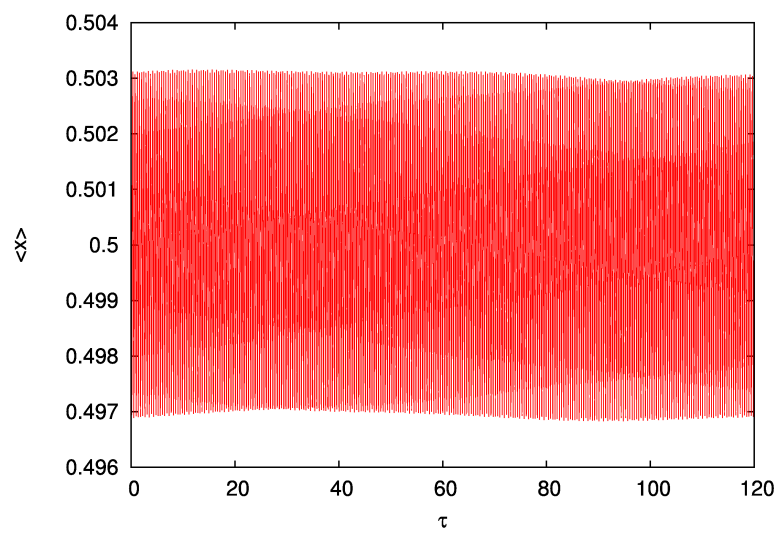
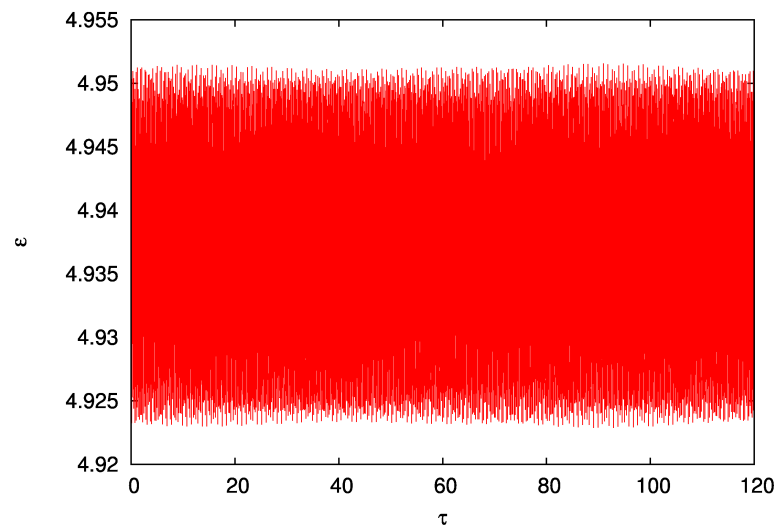


fig.16. Wykresy wielkości charakterystycznych, uzyskanych podczas siódmej symulacji przejścia.

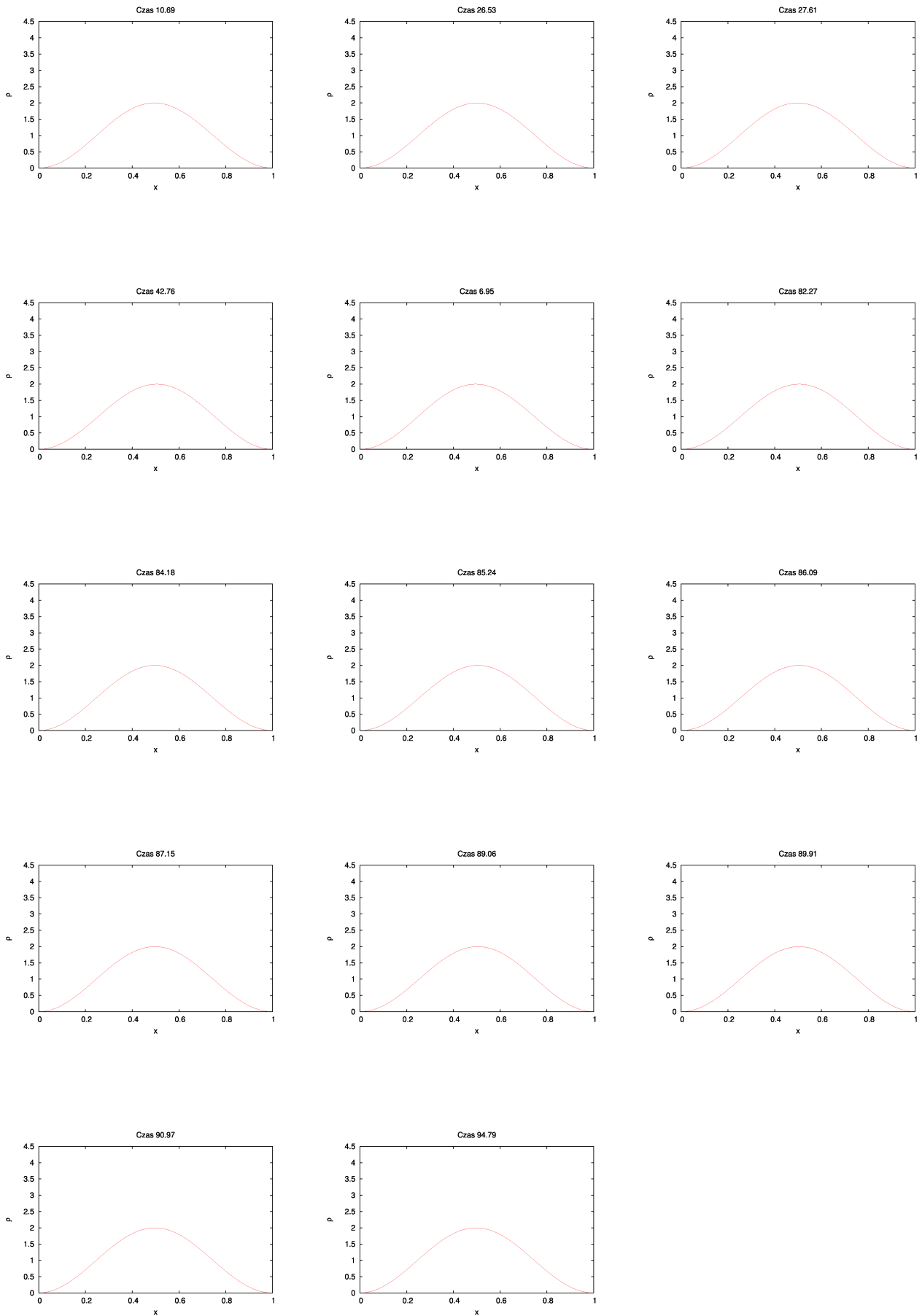


fig.17. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas siódmej symulacji przejścia.

Seria 8:

Maksymalna wartość cząstki energii:

29.5445

Cztery wartości najbardziej zbliżone do maksimum:

29.5428

29.5411

29.541

29.5385

Minimalna wartość cząstki energii:

4.91945

Cztery wartości najbardziej zbliżone do minimum:

4.91964

4.92053

4.92167

4.92195

Średnia wartość cząstki energii:

17.231975

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

17.2421

17.2376

17.2317

17.2303

Podczas symulacji **nie zostało zaobserwowane**, zgodnie z oczekiwaniami **przejście między poziomami 1 i 4**.
Chociaż gęstość prawdopodobieństwa zbliżała się do rozkładu, odpowiadającego trzeciemu oraz czwartemu poziomowi energetycznemu, to jednak cząstka nie przeszła do żadnego innego, niż podstawowy, stabilnego stanu.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.18. oraz fig.19.:

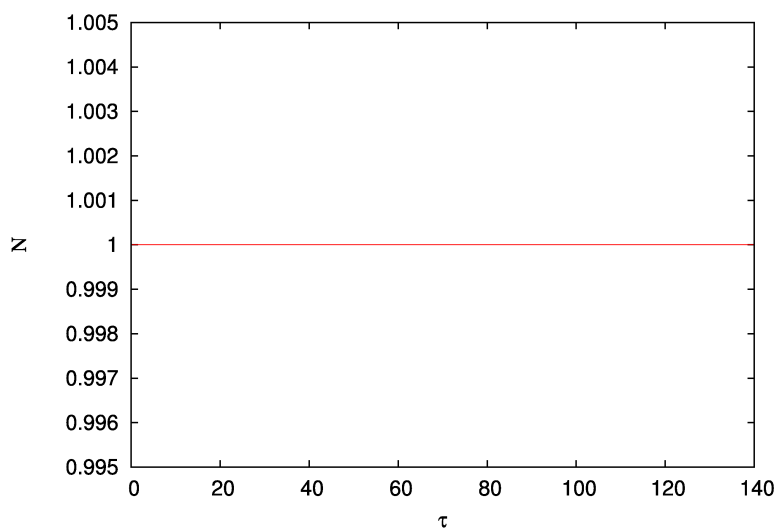
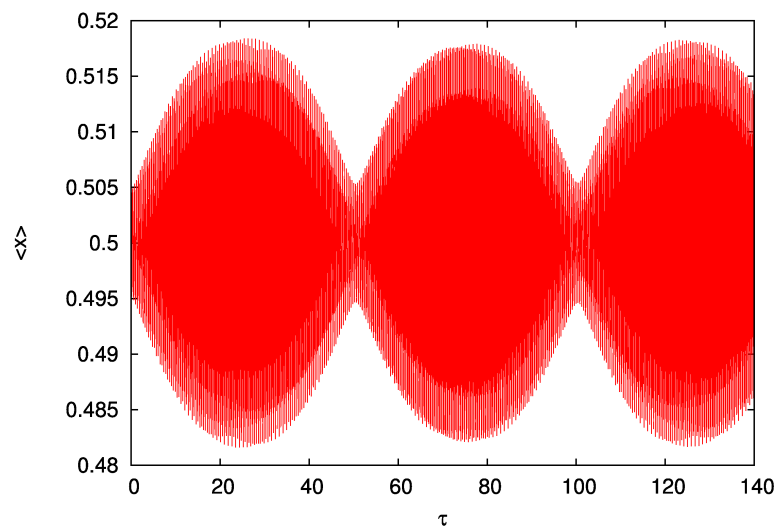
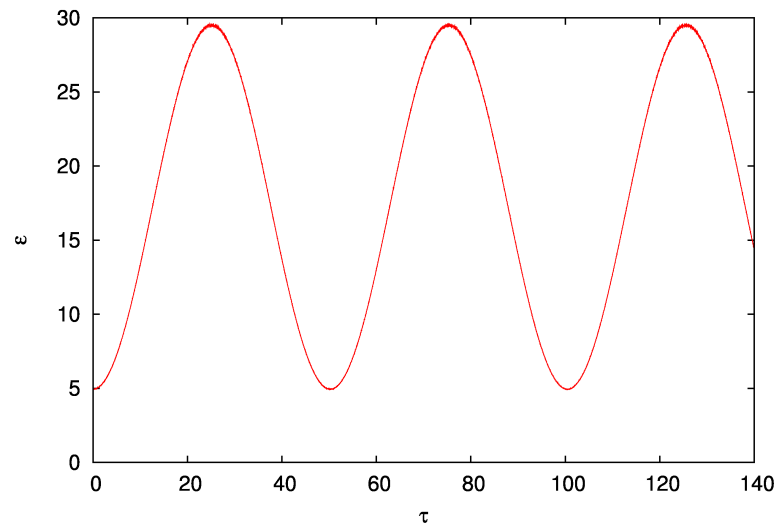


fig.18. Wykresy wielkości charakterystycznych, uzyskanych podczas ósmej symulacji przejścia.

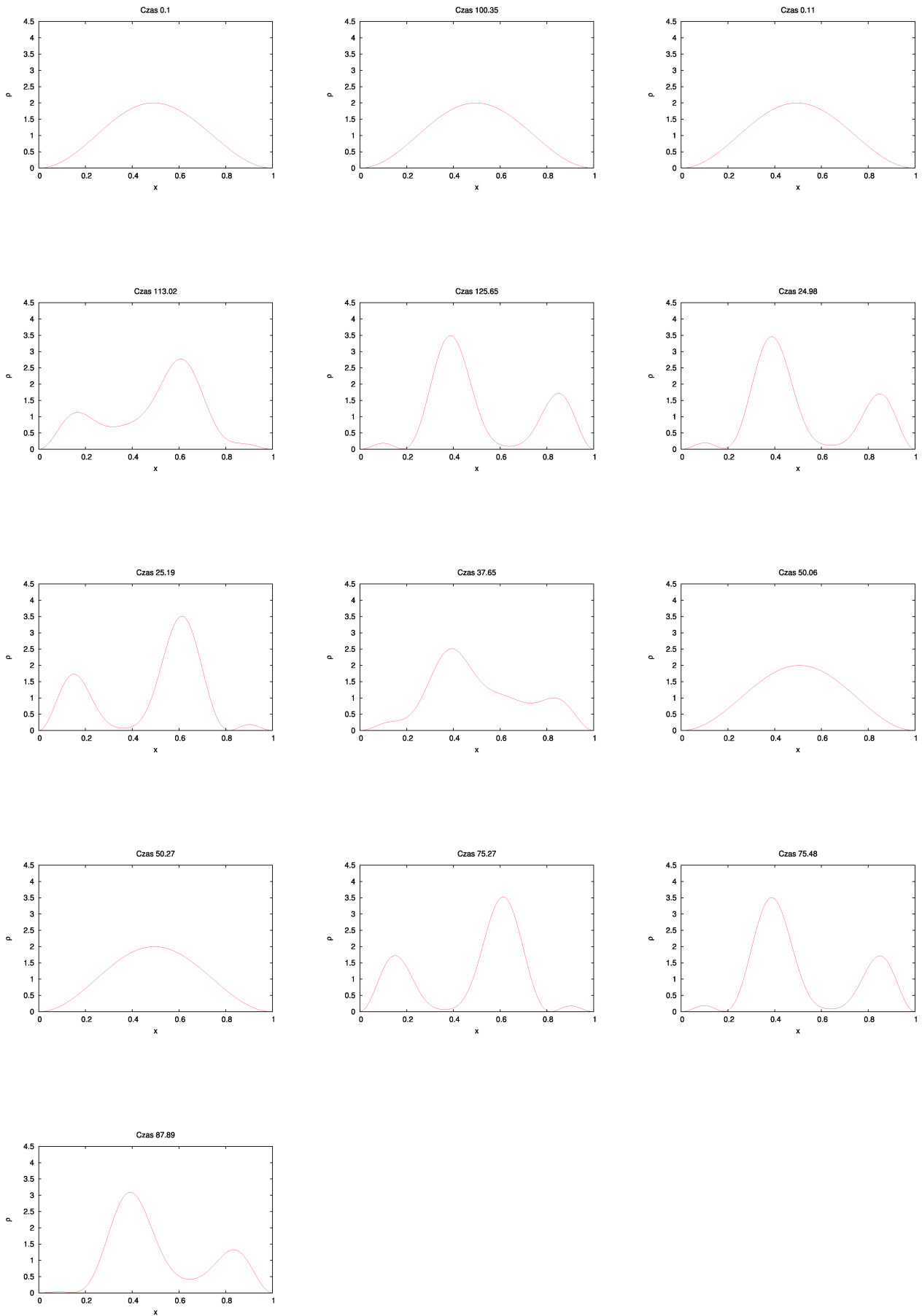


fig.19. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas ósmej symulacji przejścia.

Seria 9:

Maksymalna wartość cząstki energii:

4.93508

Cztery wartości najbardziej zbliżone do maksimum:

4.93507

4.93506

4.93505

4.93504

Minimalna wartość cząstki energii:

4.93394

Cztery wartości najbardziej zbliżone do minimum:

4.93395

4.93396

4.93397

4.93398

Średnia wartość cząstki energii:

4.93451

Cztery wartości najbardziej zbliżone do cząstki:

4.93453

4.93452

4.9345

4.93449

Podczas symulacji **nie zostało zaobserwowane**, zgodnie z oczekiwaniami **przejście między poziomami 1 i 5**.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.20. oraz fig.21.:

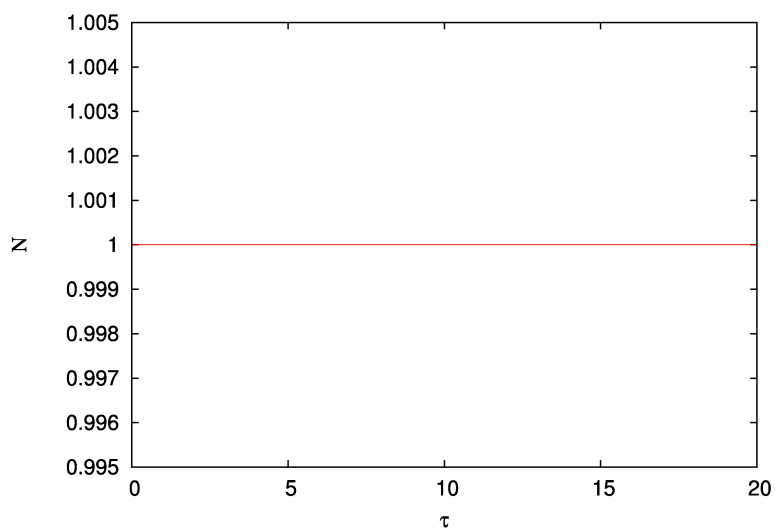
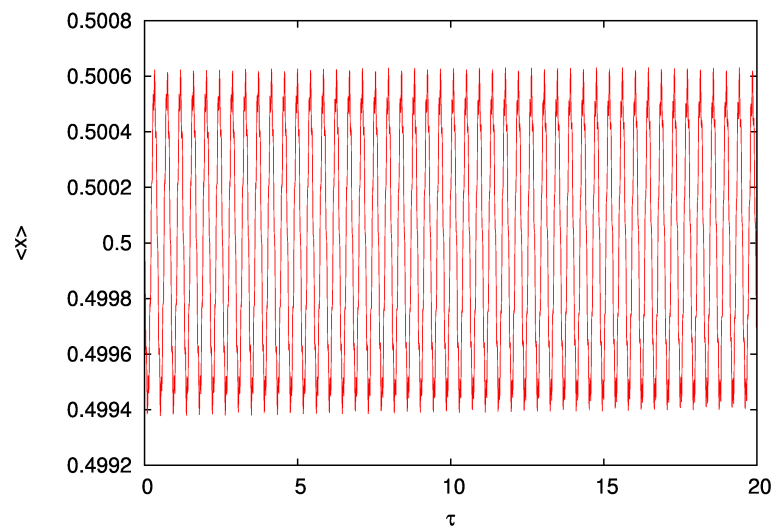
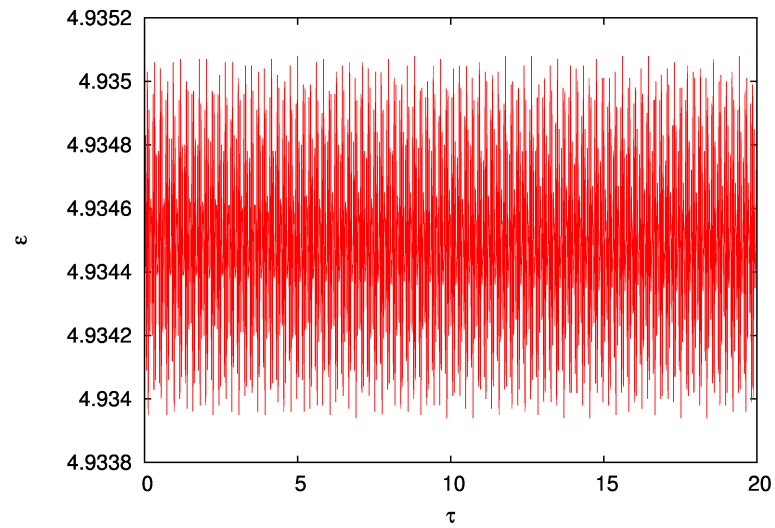


fig.20. Wykresy wielkości charakterystycznych, uzyskanych podczas dziewiątej symulacji przejścia.

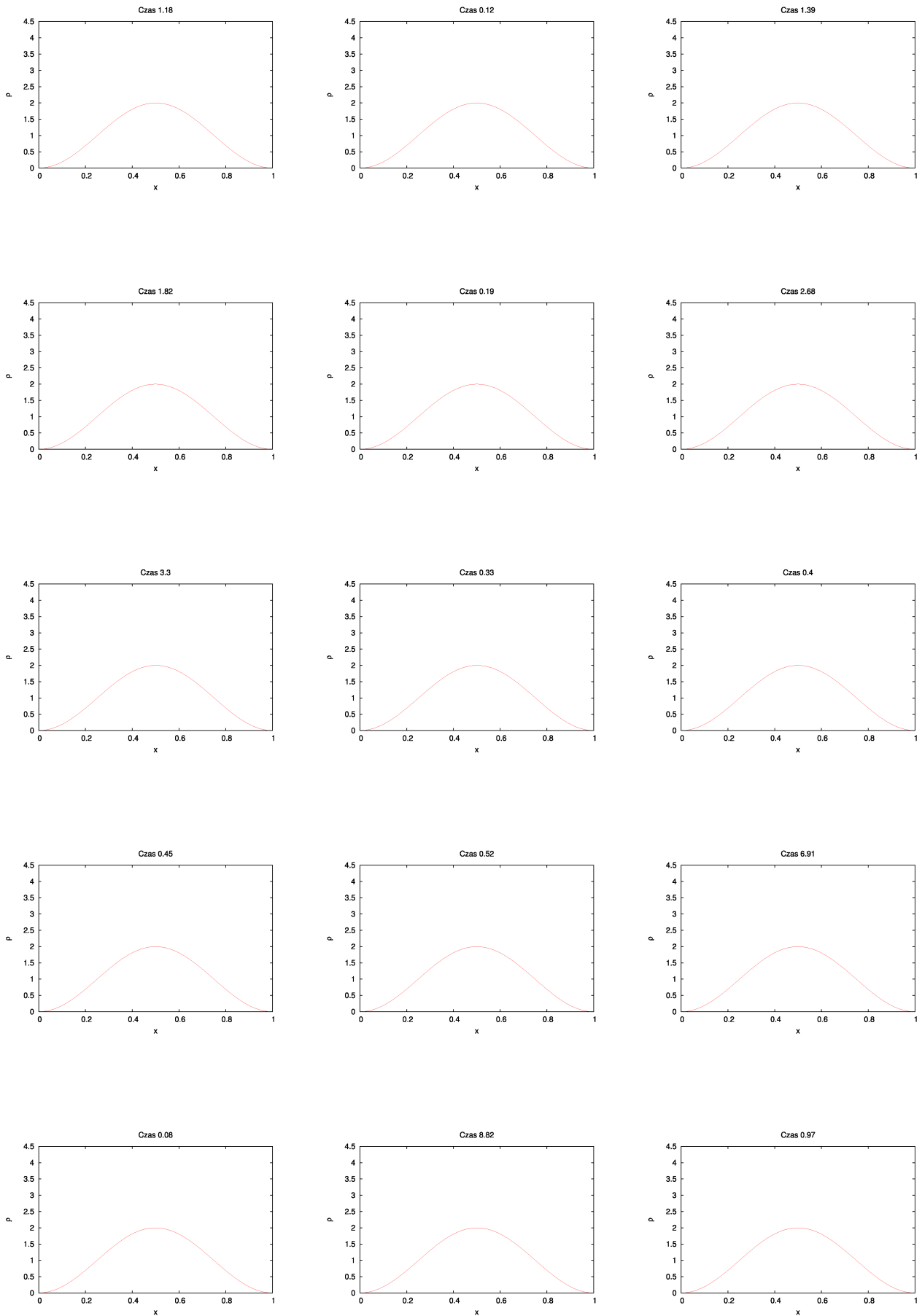


fig.21. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas dziewiętej symulacji przejścia.

Seria 10:

Maksymalna wartość cząstki energii:

5.09996

Cztery wartości najbardziej zbliżone do maksimum:

5.09985

5.09971

5.0997

5.09969

Minimalna wartość cząstki energii:

4.88576

Cztery wartości najbardziej zbliżone do minimum:

4.88591

4.88601

4.88608

4.88612

Średnia wartość cząstki energii:

4.99286

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

4.99314

4.99307

4.99278

4.99267

Podczas symulacji **nie zostało zaobserwowane**, zgodnie z oczekiwaniami **przejście między poziomami 1 i 3**.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.22. oraz fig.23.:

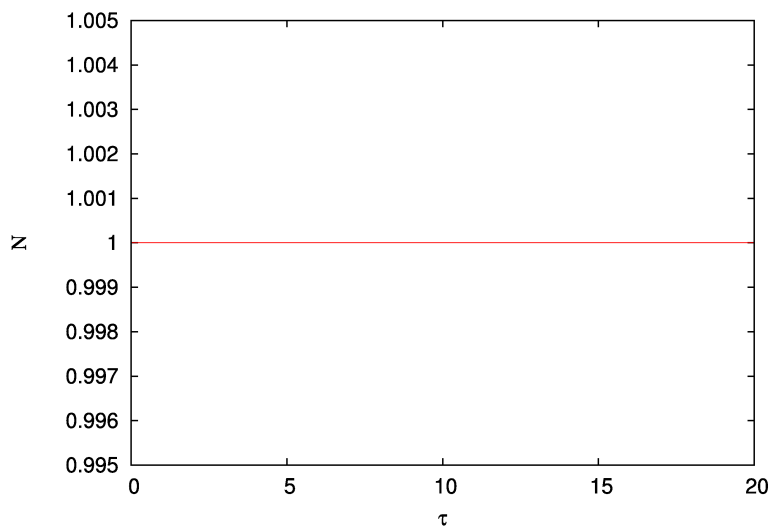
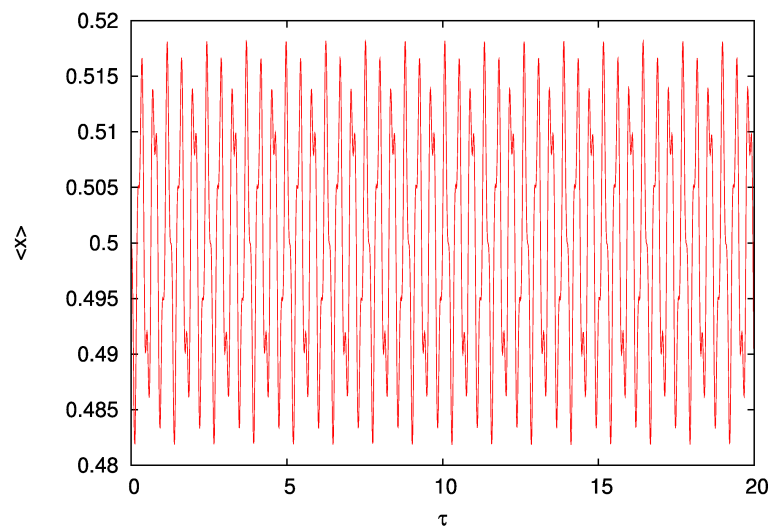
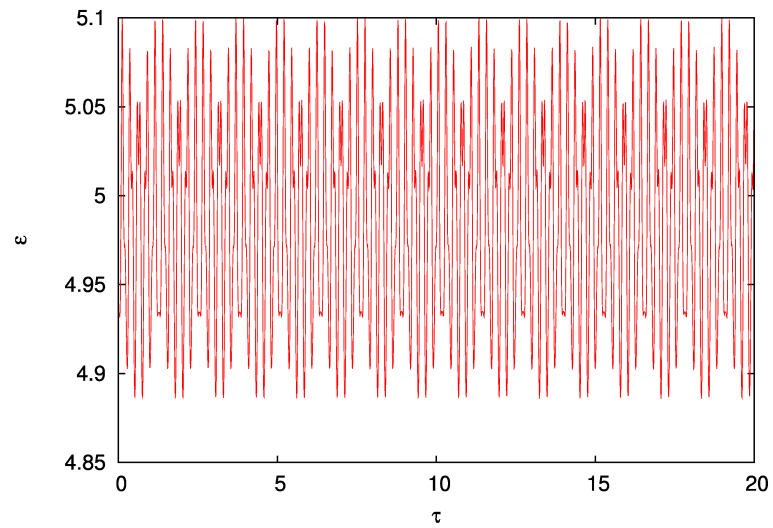


fig.22. Wykresy wielkości charakterystycznych, uzyskanych podczas dziesiątej symulacji przejścia.

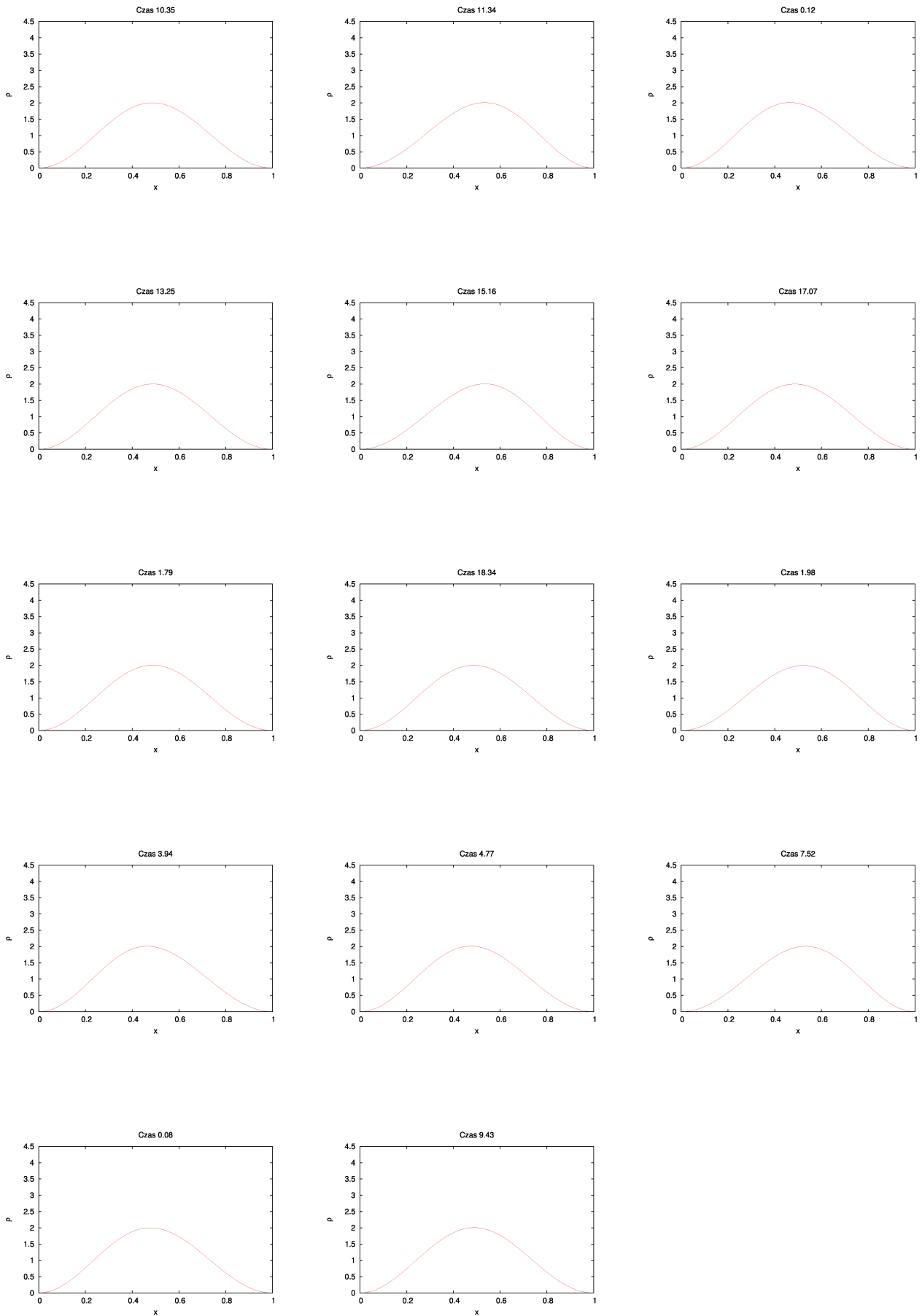


fig.23. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas dziesiątej symulacji przejścia.

Seria 11:

Maksymalna wartość cząstki energii:

5.05339

Cztery wartości najbardziej zbliżone do maksimum:

5.05317

5.05292

5.05289

5.05275

Minimalna wartość cząstki energii:

4.89739

Cztery wartości najbardziej zbliżone do minimum:

4.89748

4.89751

4.89763

4.89774

Średnia wartość cząstki energii:

4.97539

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

4.97544

4.97542

4.97538

4.97537

Podczas symulacji, zgodnie z przewidywaniami nie było obserwowane żadne przejście.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.24. oraz fig.25.:

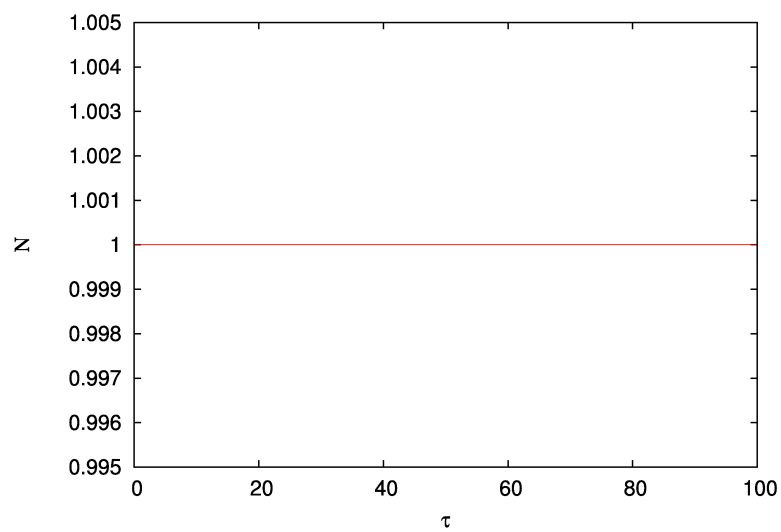
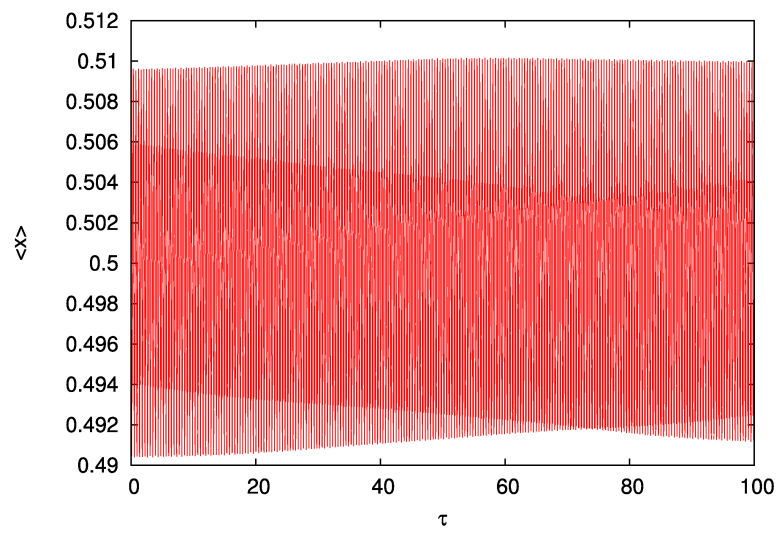
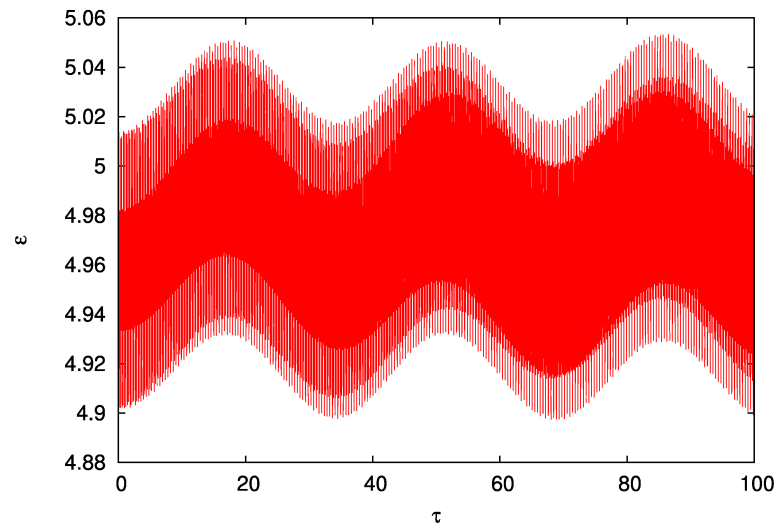


fig.24. Wykresy wielkości charakterystycznych, uzyskanych podczas jedenastej symulacji przejścia.

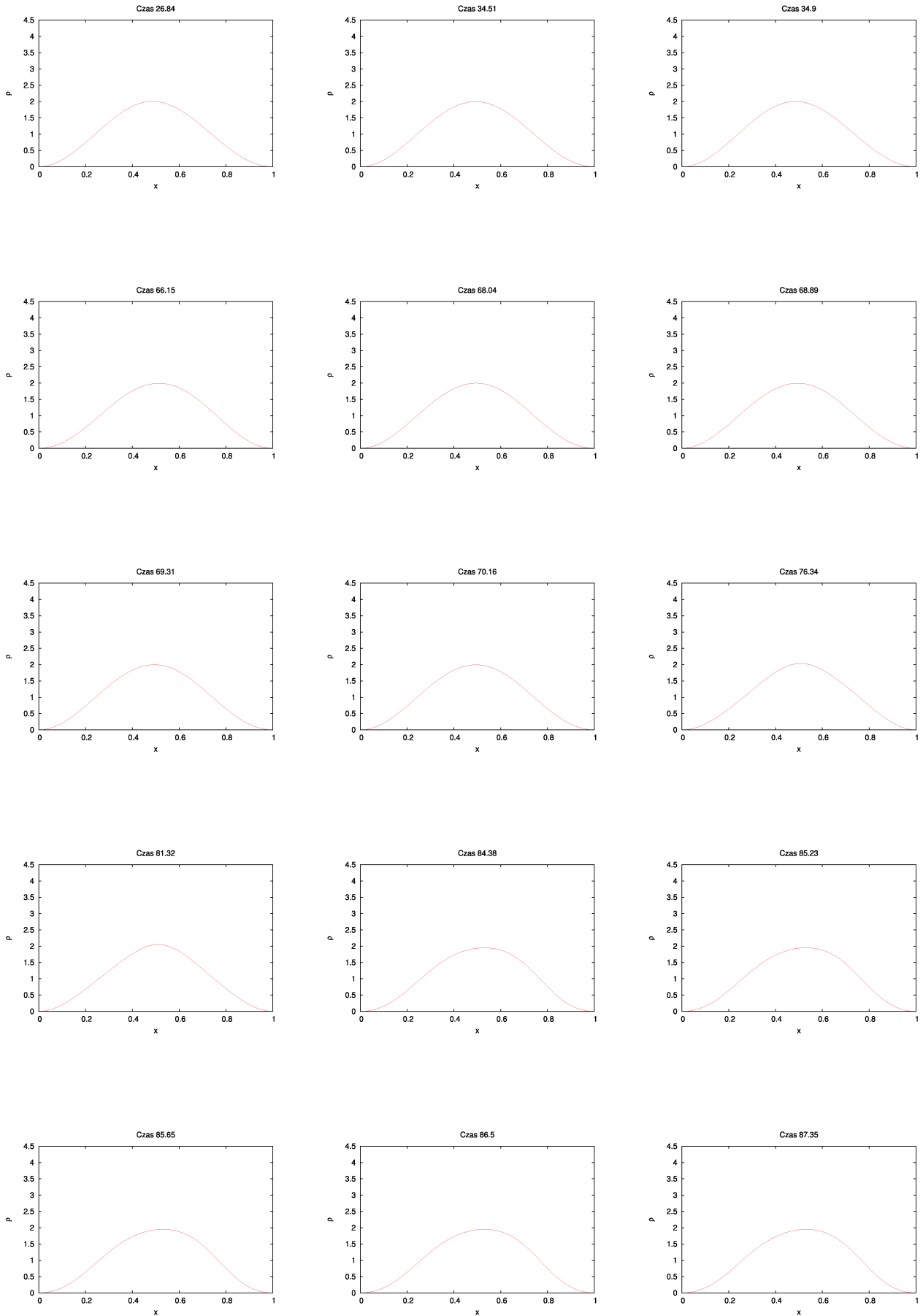


fig.25. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas jedenastej symulacji przejścia.

Seria 12:

Maksymalna wartość cząstki energii:

5.09971

Cztery wartości najbardziej zbliżone do maksimum:

5.0997

5.09969

5.09966

5.09954

Minimalna wartość cząstki energii:

4.88608

Cztery wartości najbardziej zbliżone do minimum:

4.88612

4.88615

4.88624

4.88638

Średnia wartość cząstki energii:

4.992895

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

4.99346

4.99307

4.99278

4.99267

Podczas symulacji **nie zostało zaobserwowane**, zgodnie z oczekiwaniami **przejście między poziomami 1 i 3**.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.26. oraz fig.27.:

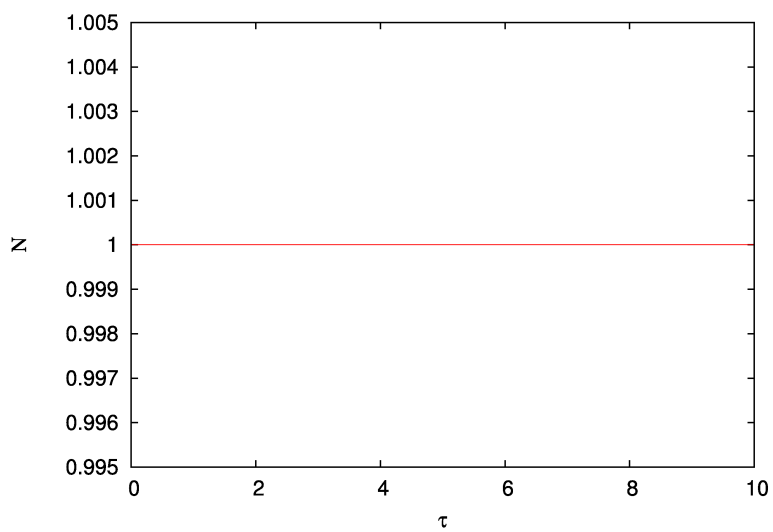
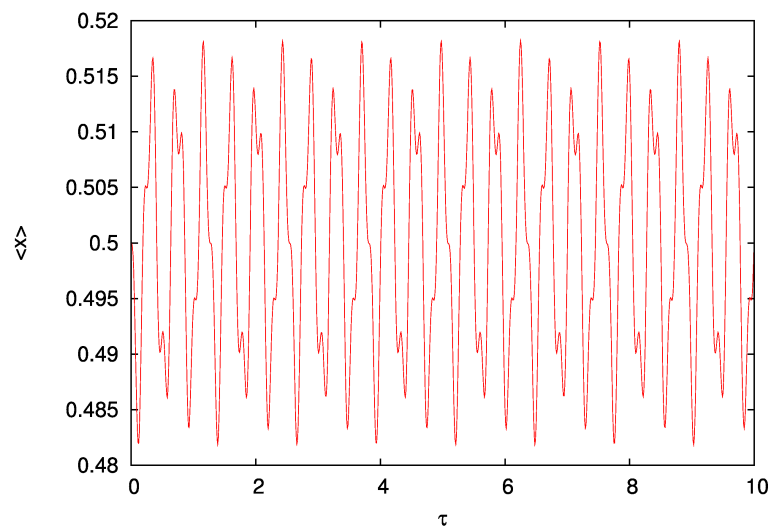
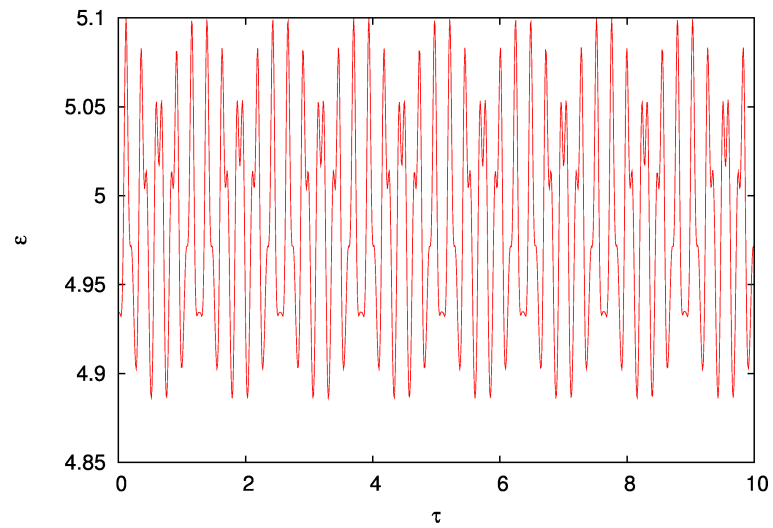


fig.26. Wykresy wielkości charakterystycznych, uzyskanych podczas dwunastej symulacji przejścia.

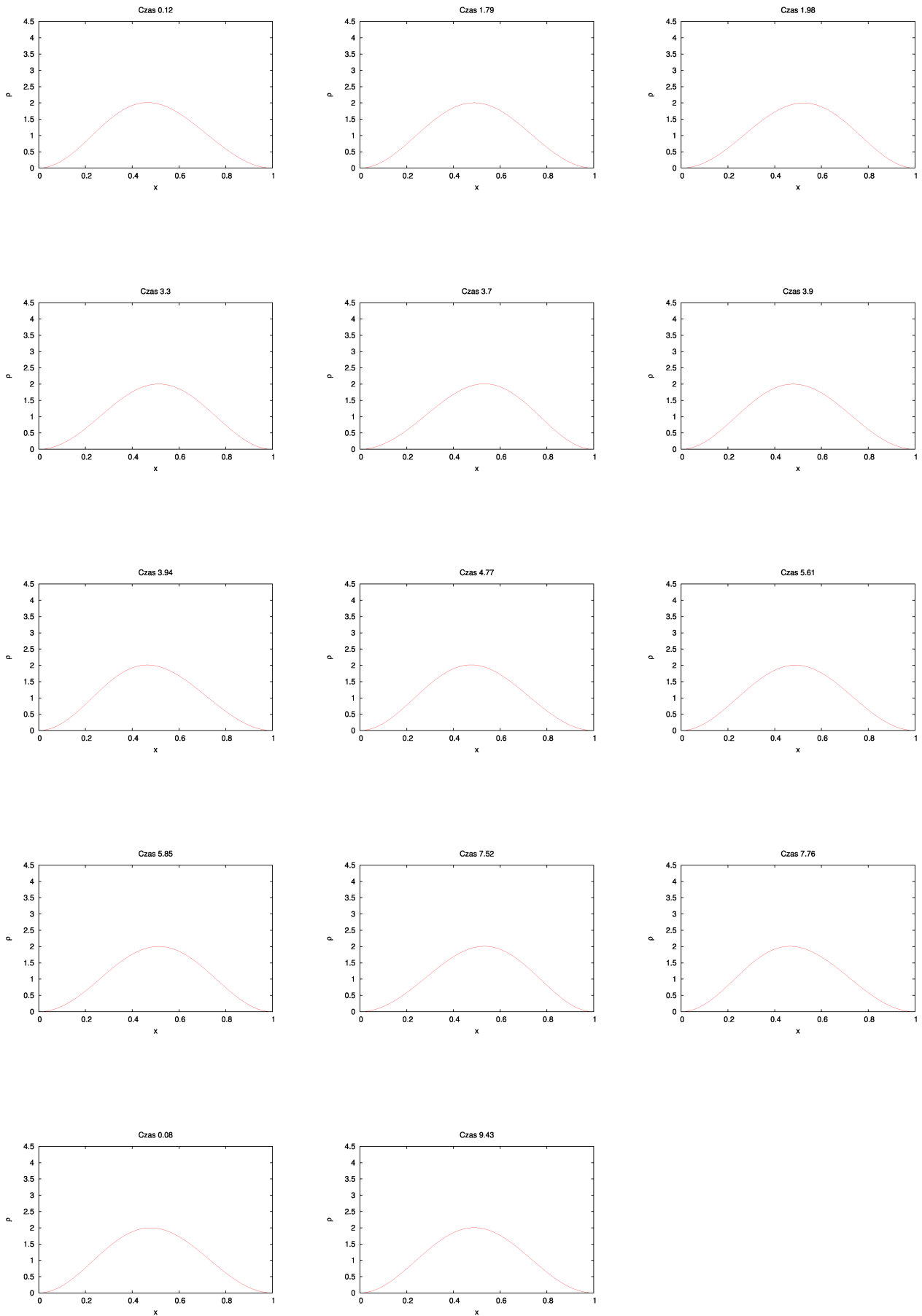


fig.27. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas dwunastej symulacji przejścia.

Seria 13:

Maksymalna wartość cząstki energii:

5.30257

Cztery wartości najbardziej zbliżone do maksimum:

5.301

5.30043

5.30035

5.30015

Minimalna wartość cząstki energii:

4.87716

Cztery wartości najbardziej zbliżone do minimum:

4.87774

4.87778

4.87837

4.88

Średnia wartość cząstki energii:

5.089865

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

5.08989

5.08987

5.08983

5.08982

Podczas symulacji, zgodnie z przewidywaniami nie było obserwowane żadne przejście.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.28. oraz fig.29.:

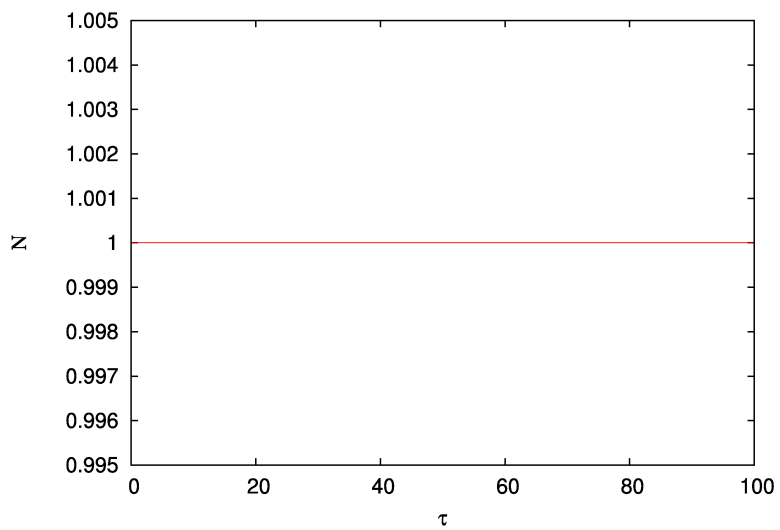
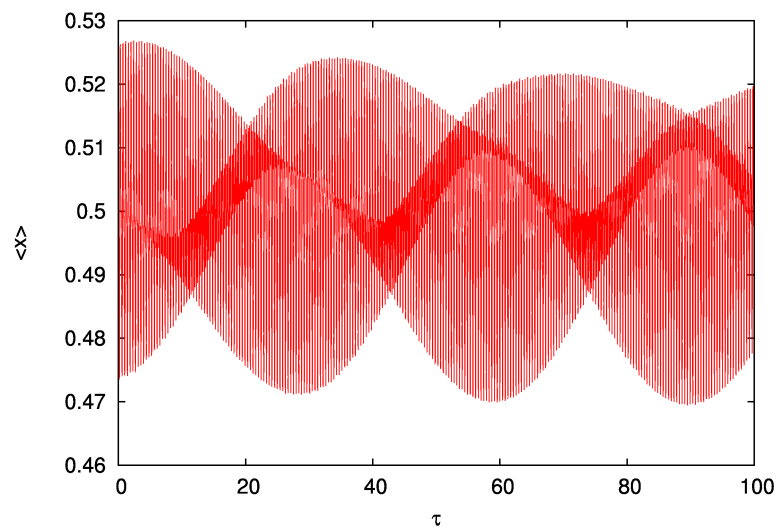
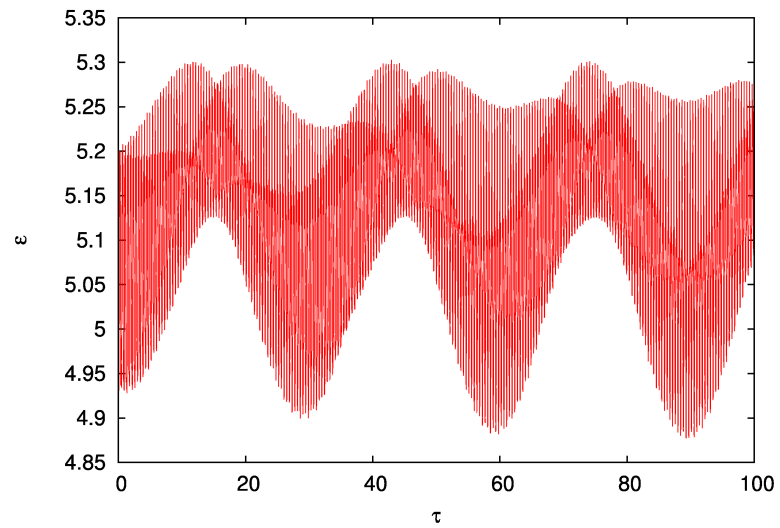


fig.28. Wykresy wielkości charakterystycznych, uzyskanych podczas trzynastej symulacji przejścia.

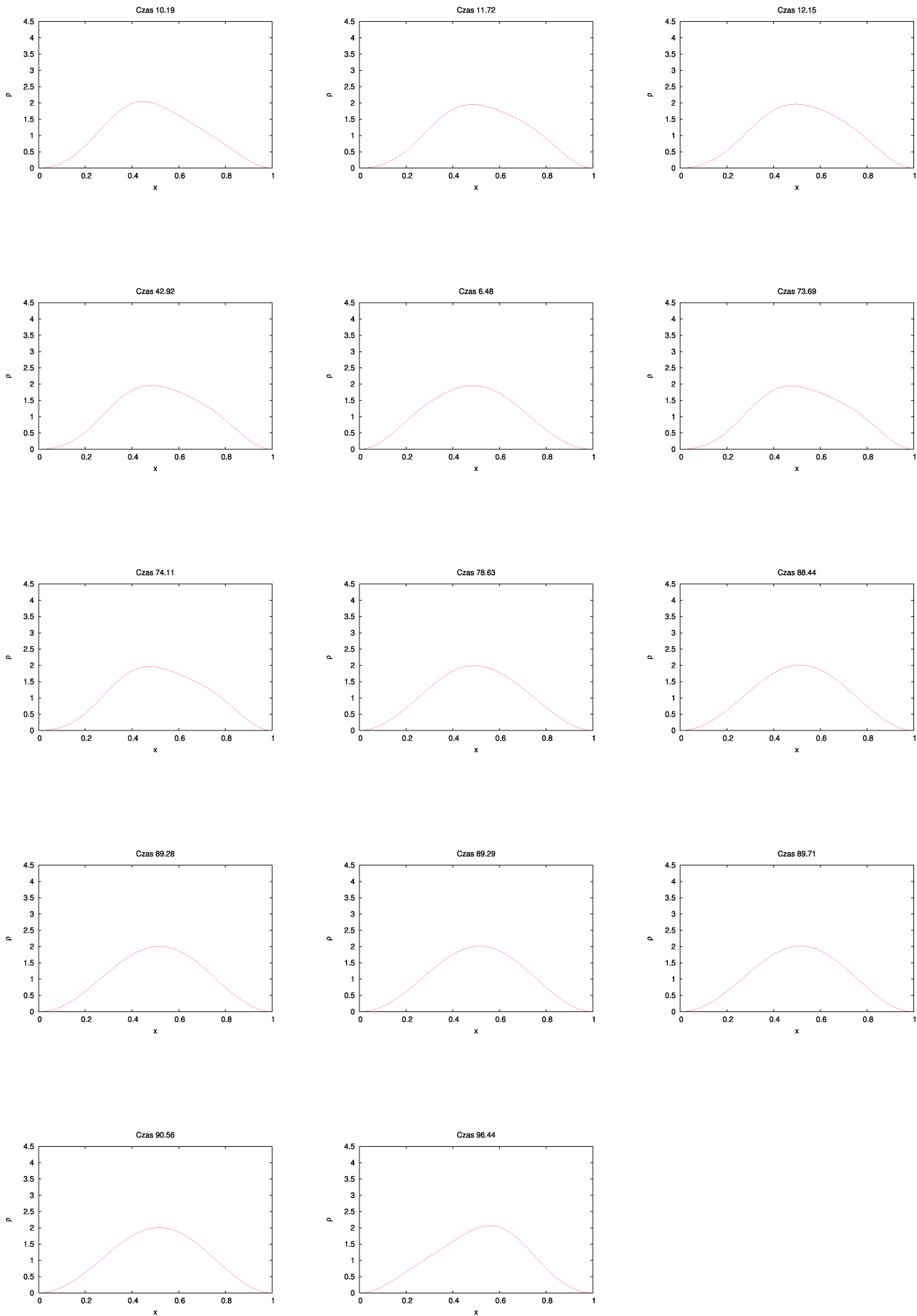


fig.29. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas trzynastej symulacji przejścia.

Seria 14:

Maksymalna wartość cząstki energii:

5.02317

Cztery wartości najbardziej zbliżone do maksimum:

5.02236

5.02158

5.02137

5.02101

Minimalna wartość cząstki energii:

4.90328

Cztery wartości najbardziej zbliżone do minimum:

4.90379

4.90444

4.90446

4.90463

Średnia wartość cząstki energii:

4.963225

(nie wystąpiła w żadnym z zapisanych plików)

Cztery wartości najbardziej zbliżone do cząstki:

4.96351

4.96342

4.96317

4.96302

Podczas symulacji, zgodnie z przewidywaniami nie było obserwowane żadne przejście.

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.30. oraz fig.31.:

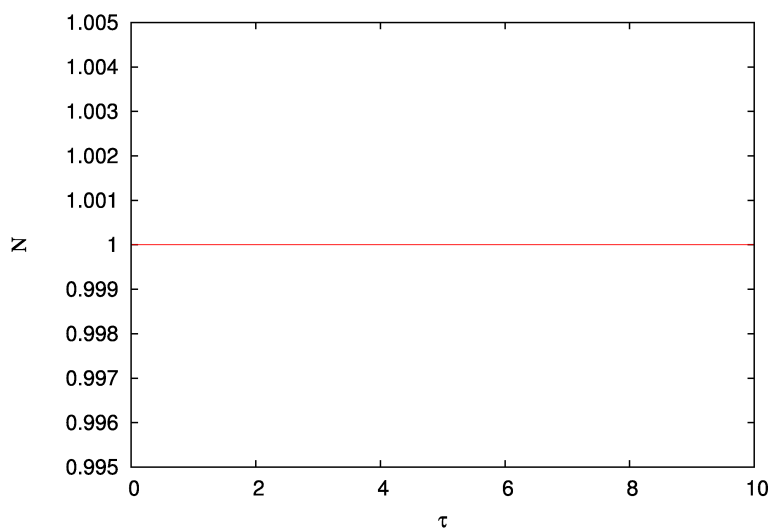
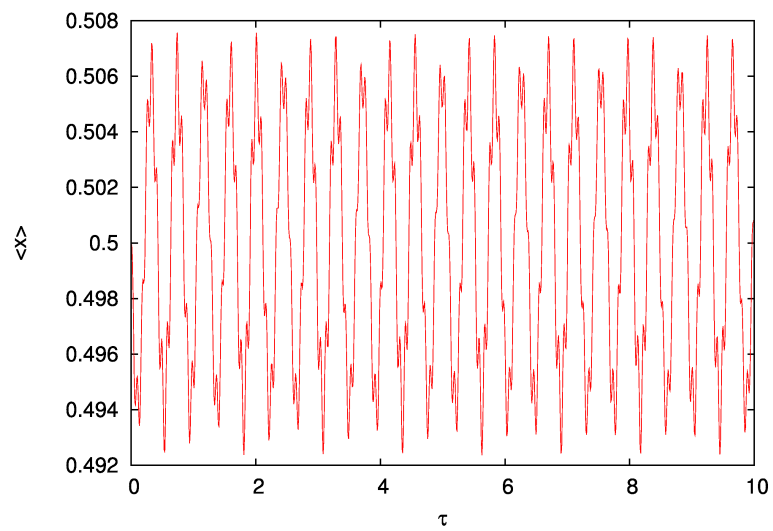
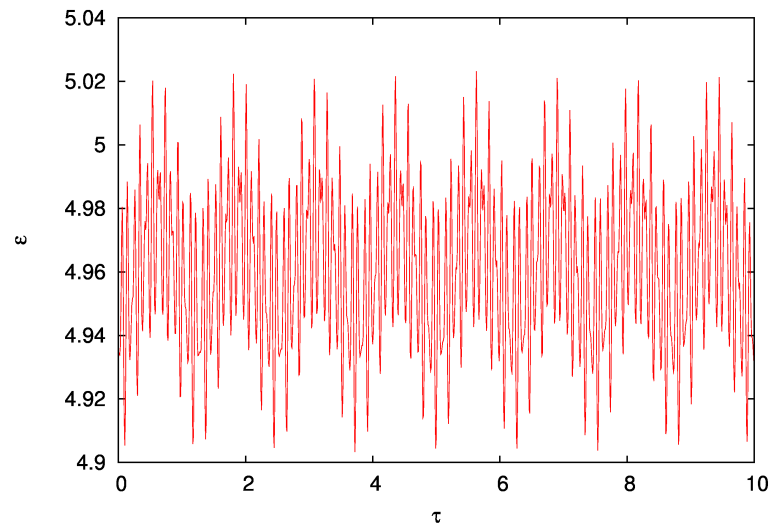


fig.30. Wykresy wielkości charakterystycznych, uzyskanych podczas czternastej symulacji przejścia.

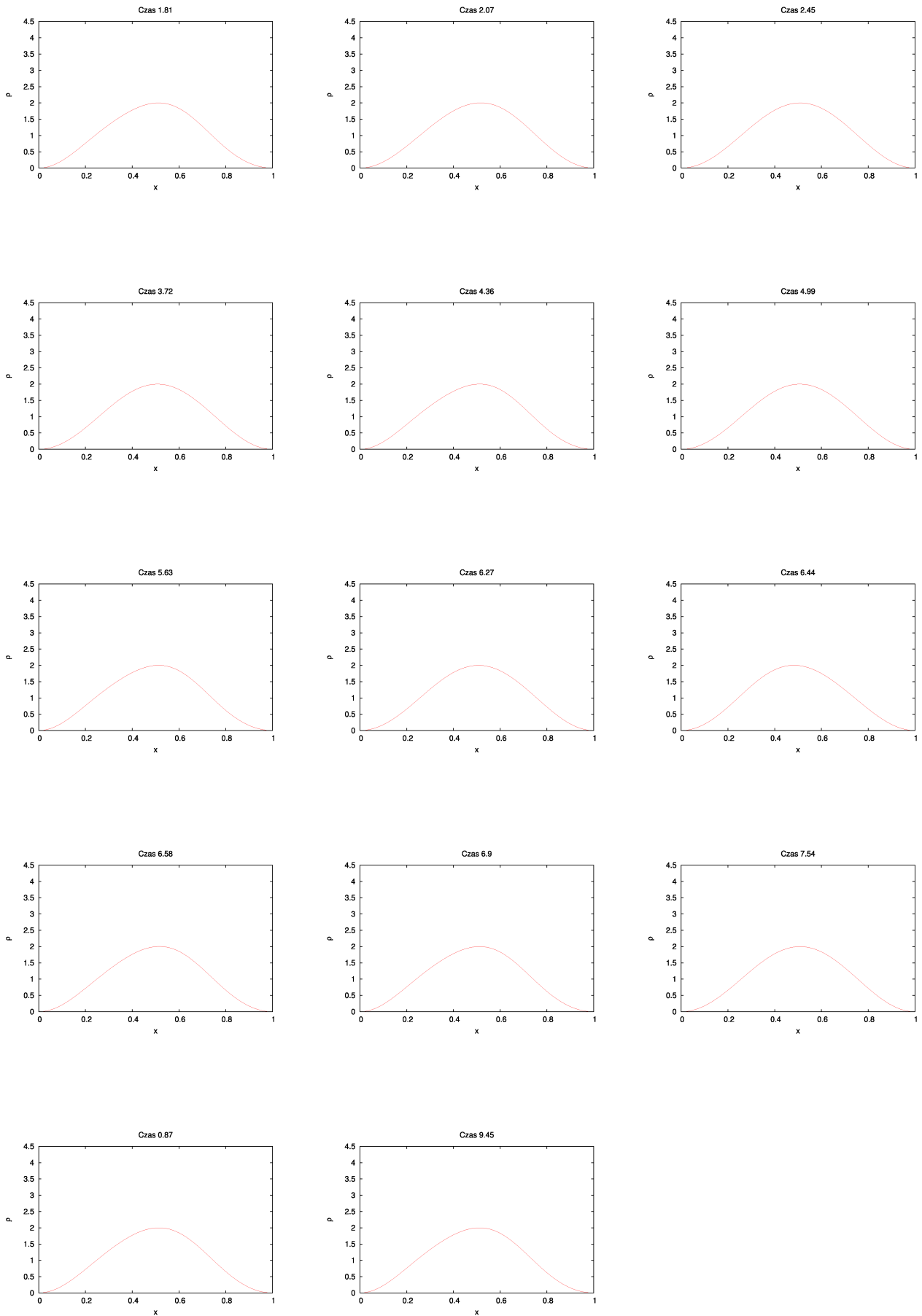


fig.31. Wykresy gęstości prawdopodobieństwa, w różnych chwilach czasu, ilustrujące ewolucję czasową tej wielkości, uzyskaną podczas czternastej symulacji przejścia.

Jak widać, model najlepiej opisuje przejścia pomiędzy sąsiednimi poziomami energetycznymi. Dla przejść pomiędzy odleglejszymi poziomami, pojawiają się niestety odstępstwa.

W trzeciej serii symulacji (badanie okolic rezonansu) uzyskałem następujące wyniki:

Seria 1:

Dziesięć największych wartości energii cząstki :

8.58717

8.58499

8.58382

8.58207

8.58151

8.57846

8.57844

8.57491

8.57467

8.57016

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.32.:

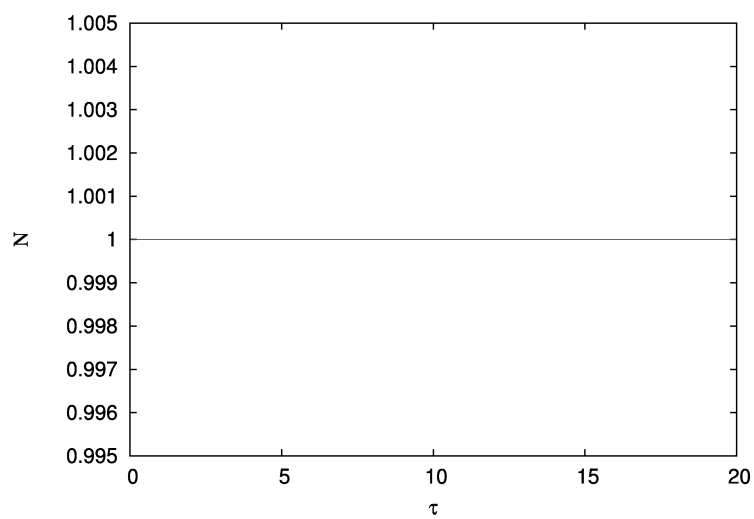
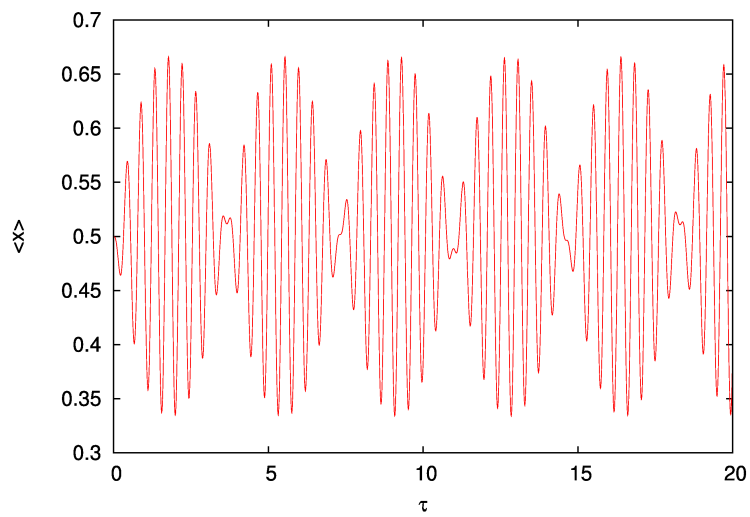
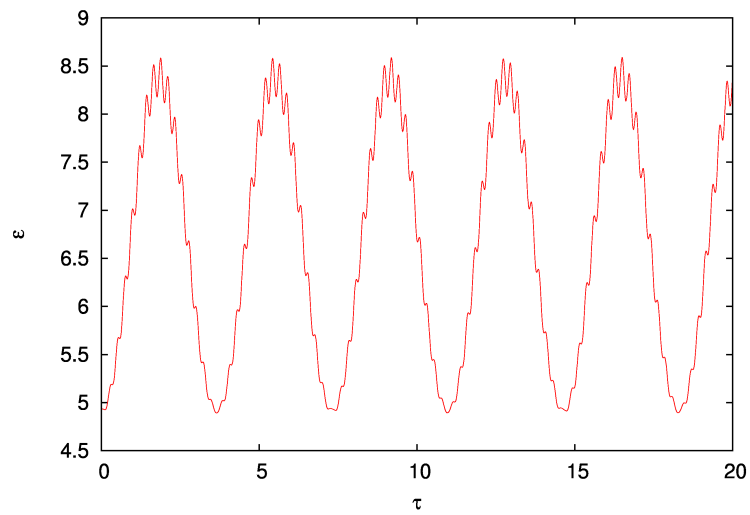


fig.32. Wykresy wielkości charakterystycznych, uzyskanych podczas pierwszej symulacji w okolicach częstotliwości rezonansowej ($90\% \omega_{rez}$).

Seria 2:

Dziesięć największych wartości energii cząstki:

10.0135

10.0134

10.0114

10.0113

10.0109

10.0108

10.0095

10.0057

10.0051

10.0043

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.33.

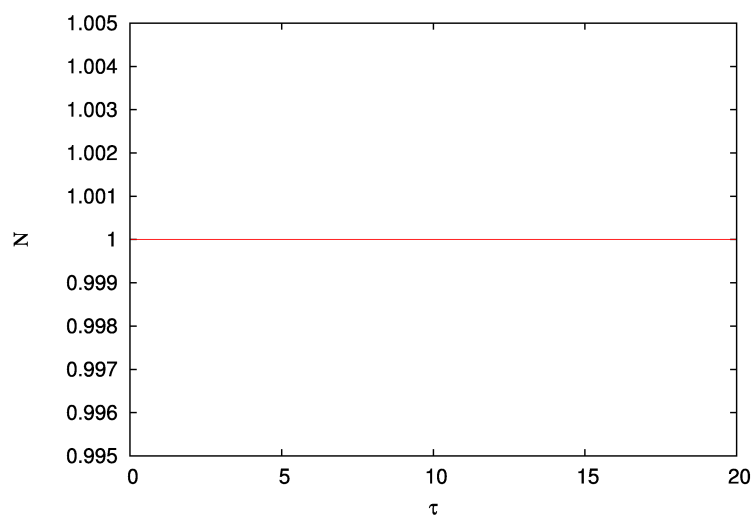
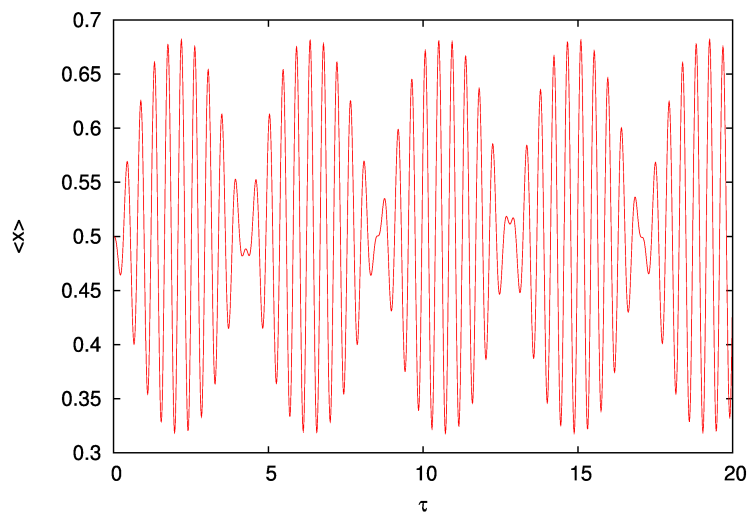
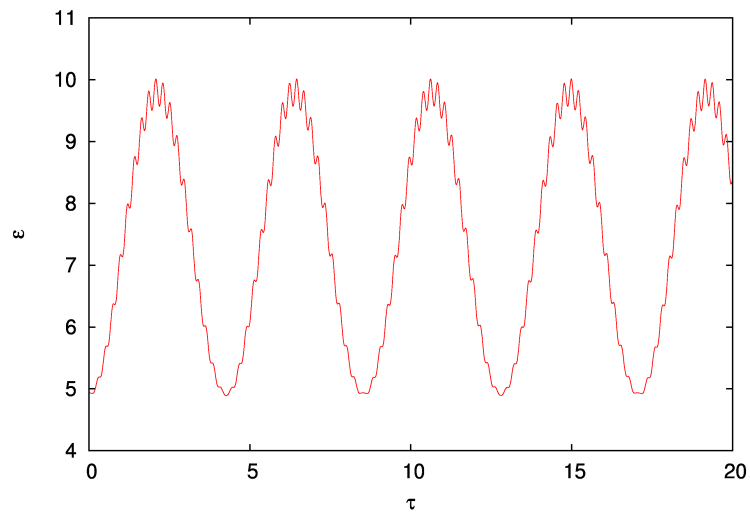


fig.33. Wykresy wielkości charakterystycznych, uzyskanych podczas drugiej symulacji w okolicach częstotliwości rezonansowej ($92\% \omega_{rez}$).

Seria 3:

Dziesięć największych wartości energii cząstki:

12.1441
12.1397
12.1396
12.1393
12.1295
12.1291
12.1241
12.1236
12.1206
12.1203

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.34.

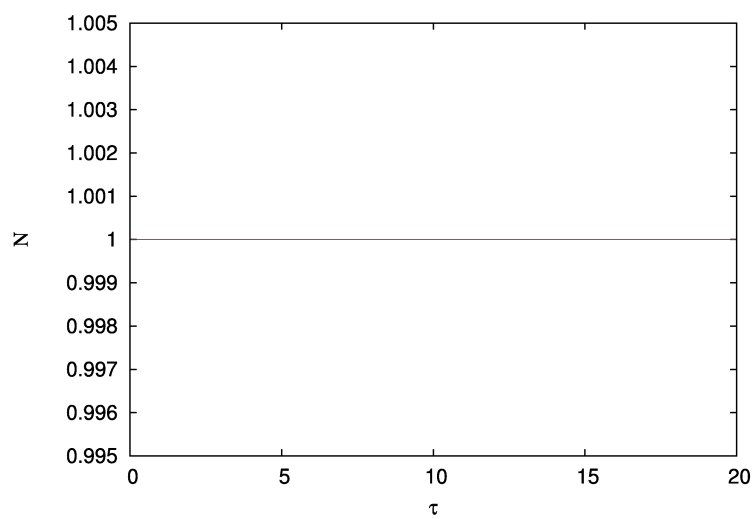
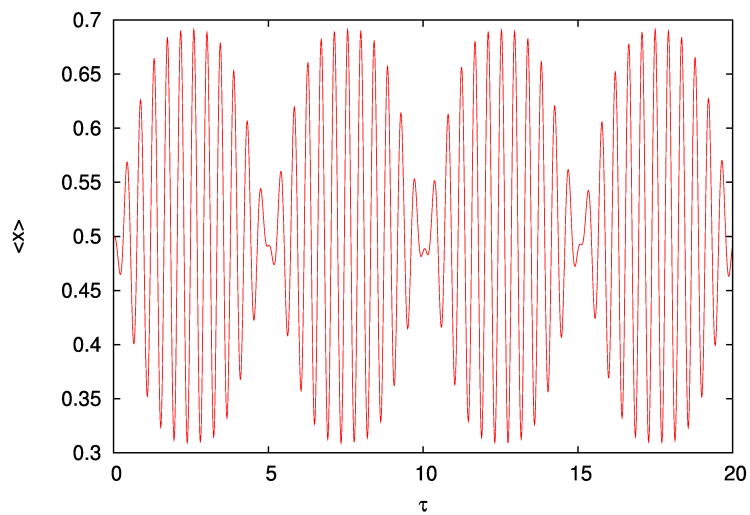
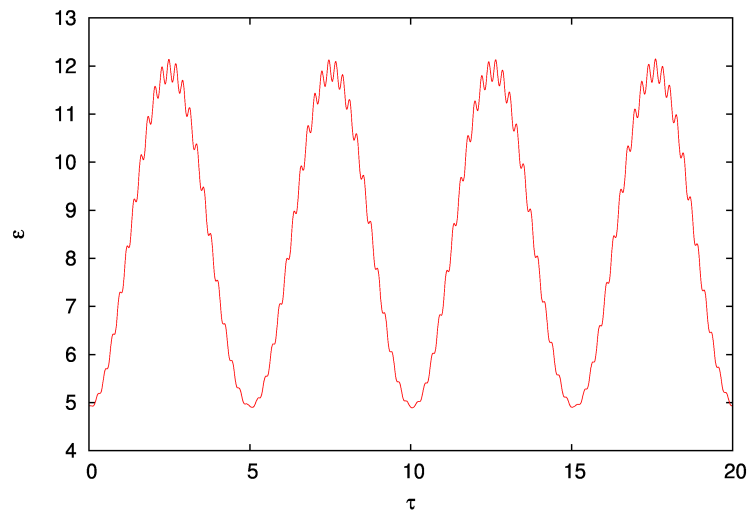


fig.34. Wykresy wielkości charakterystycznych, uzyskanych podczas trzeciej symulacji w okolicach częstości rezonansowej ($94\% \omega_{rez}$).

Seria 4:

Dziesięć największych wartości energii cząstki:

18.2288
18.2276
18.2252
18.2244
18.2165
18.2126
18.2123
18.2097
18.2085
18.1976

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.35.

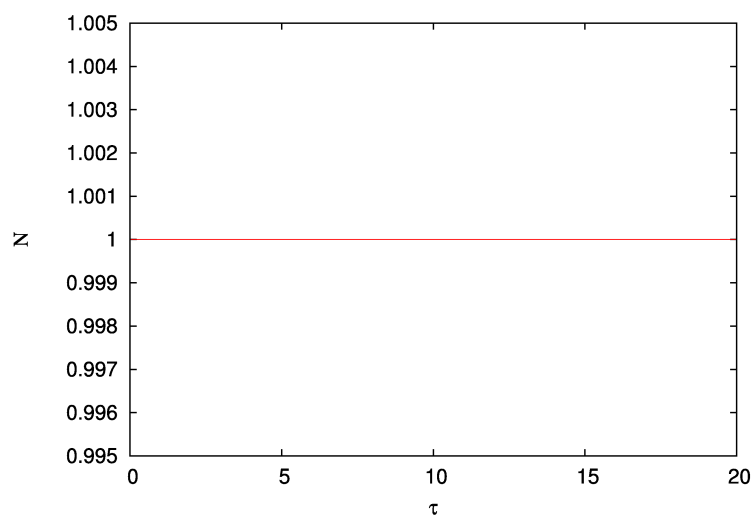
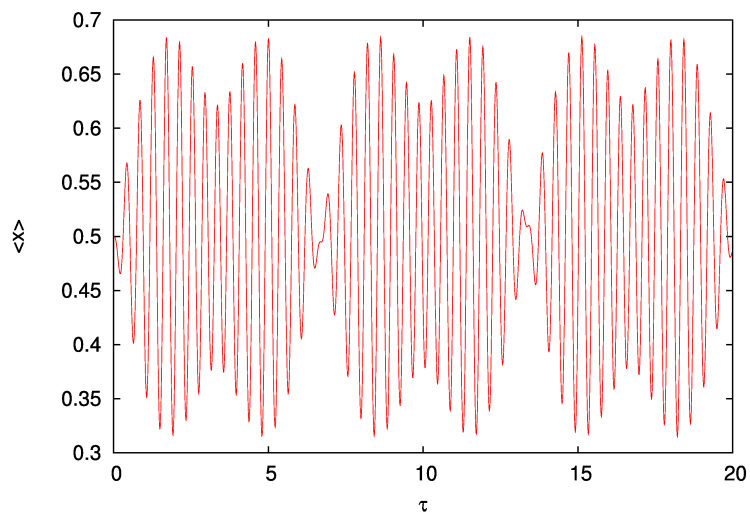
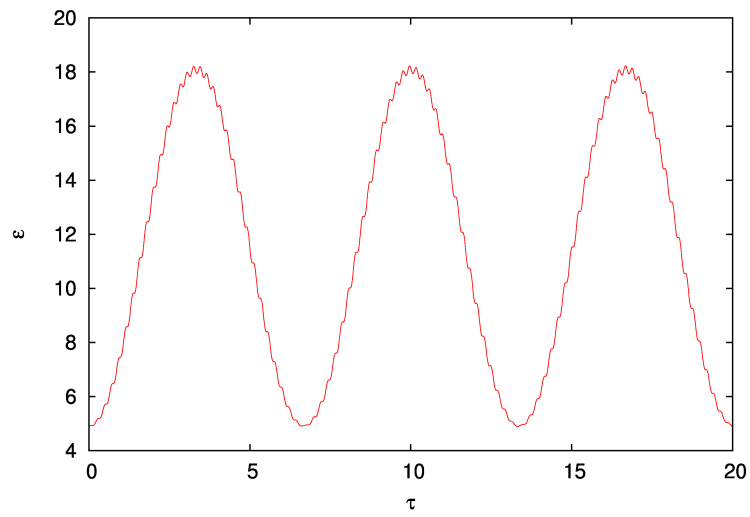


fig.35. Wykresy wielkości charakterystycznych, uzyskanych podczas czwartej symulacji w okolicach częstotliwości rezonansowej ($98\% \omega_{rez}$).

Seria 5:

Dziesięć największych wartości energii cząstki:

19.7641
19.7634
19.7623
19.7622
19.7621
19.7601
19.7597
19.7591
19.7585
19.7584

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.36.

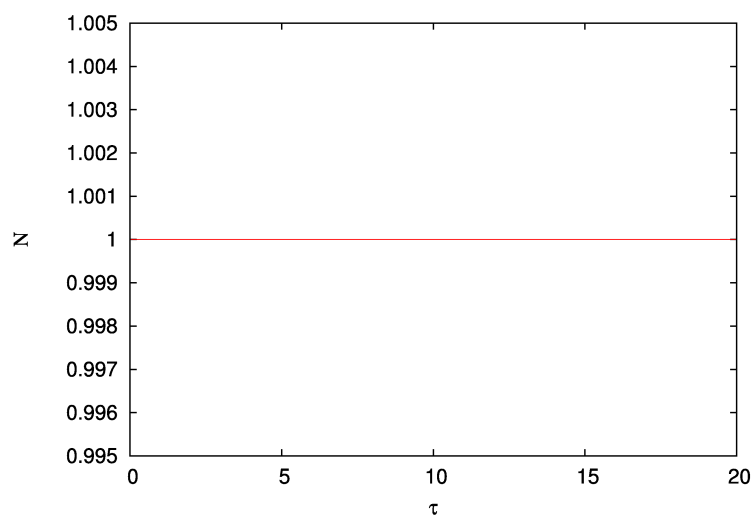
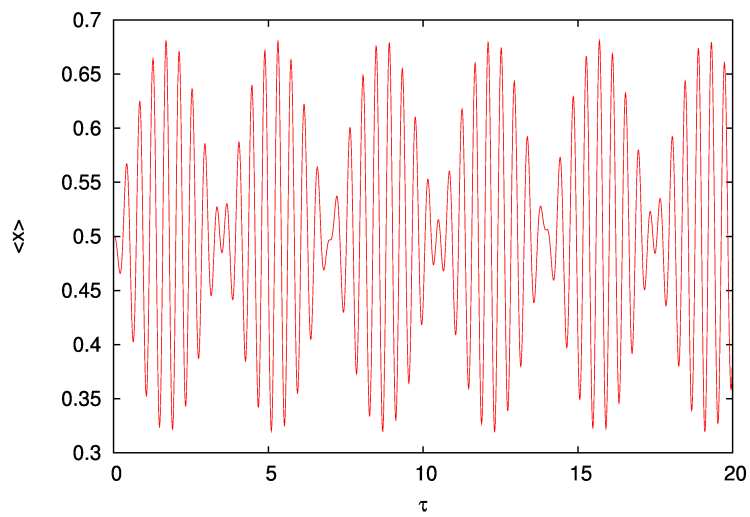
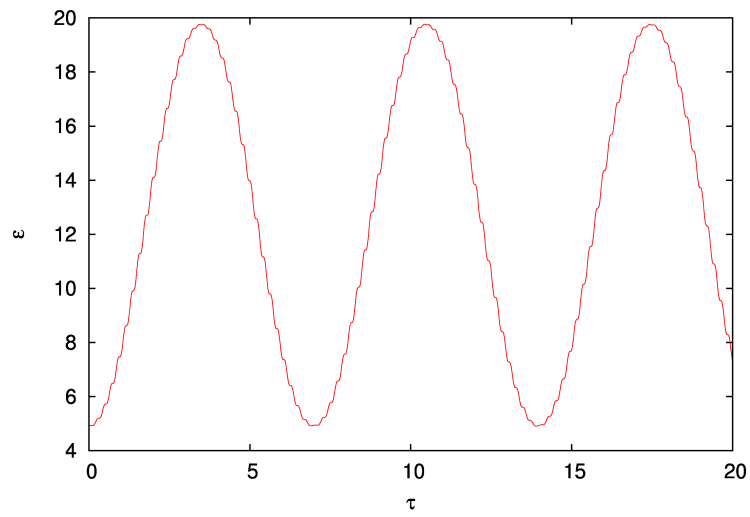


fig.36. Wykresy wielkości charakterystycznych, uzyskanych podczas piątej symulacji w okolicach częstotliwości rezonansowej ($100\% \omega_{rez}$).

Seria 6:

Dziesięć największych wartości energii cząstki:

18.6209

18.6135

18.6098

18.6095

18.6049

18.6046

18.5999

18.5998

18.5993

18.5963

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.37.

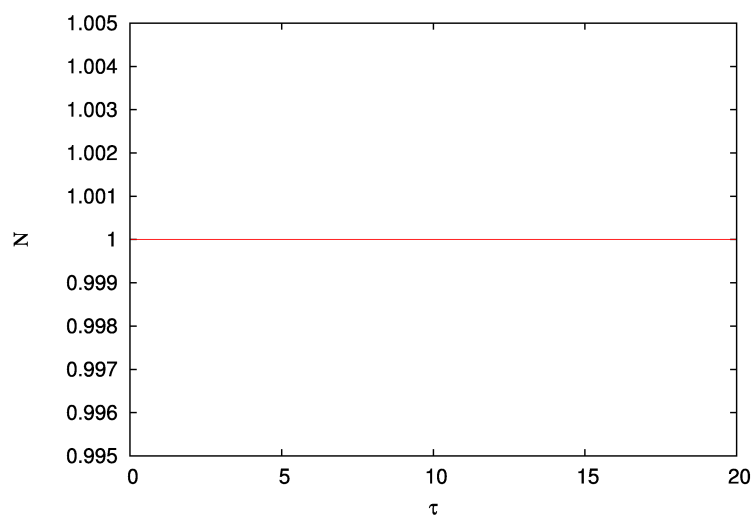
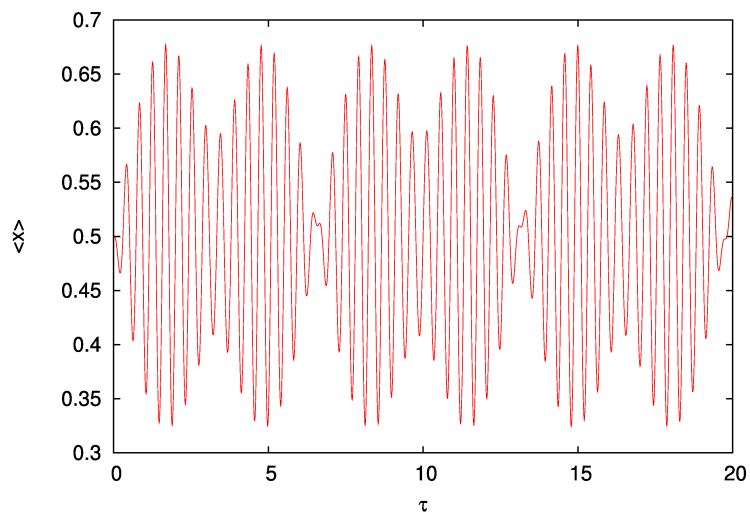
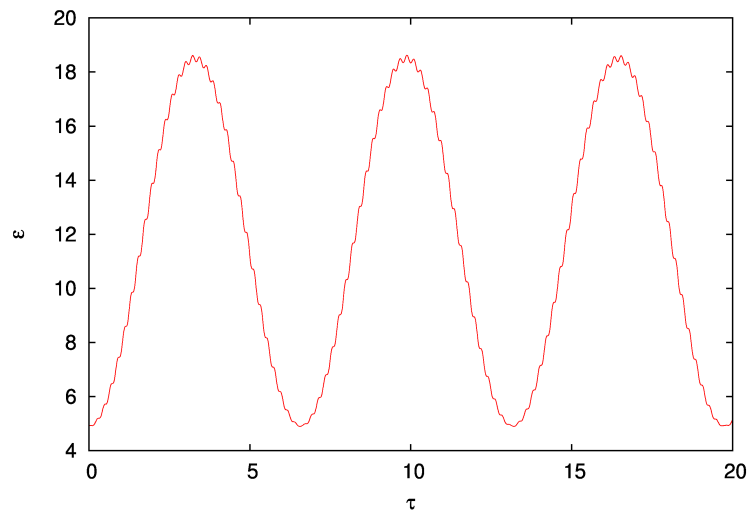


fig.37. Wykresy wielkości charakterystycznych, uzyskanych podczas szóstej symulacji w okolicach częstotliwości rezonansowej ($102\% \omega_{rez}$).

Seria 7:

Dziesięć największych wartości energii cząstki:

15.8288

15.8271

15.8268

15.8233

15.8167

15.8084

15.8006

15.7971

15.7883

15.7788

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.38.

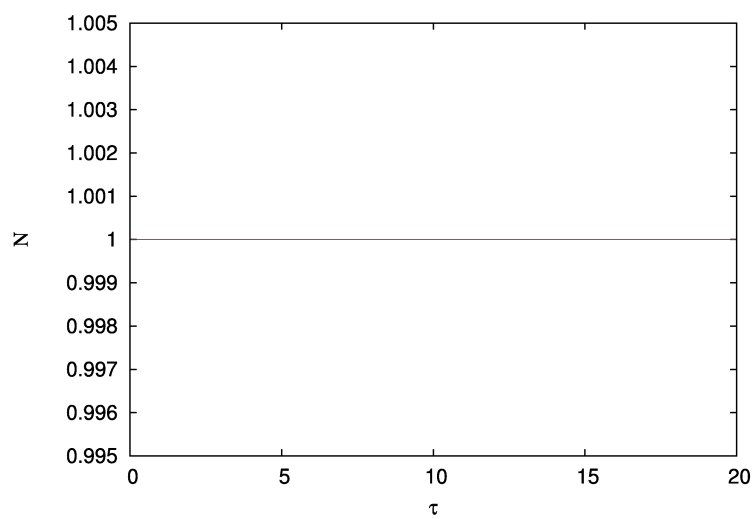
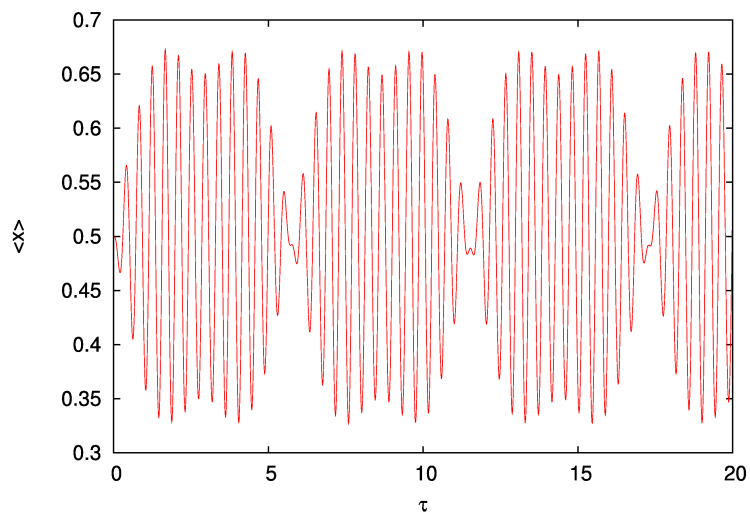
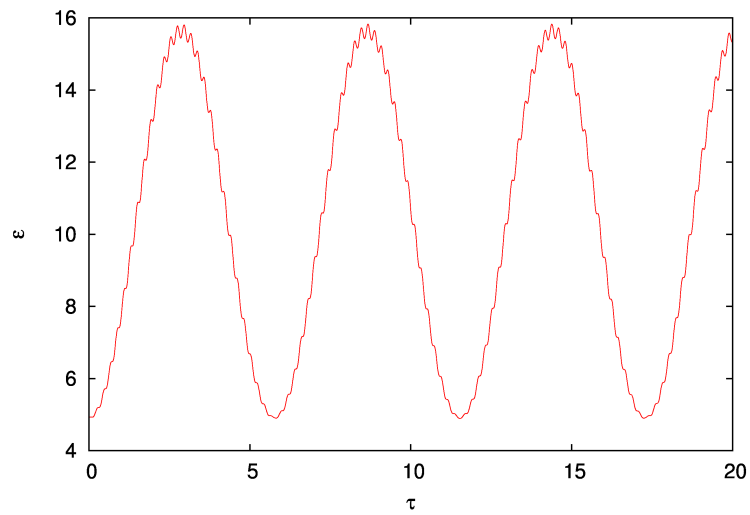


fig.38. Wykresy wielkości charakterystycznych, uzyskanych podczas siódmej symulacji w okolicach częstoty rezonansowej ($104\% \omega_{rez}$).

Seria 8:

Dziesięć największych wartości energii cząstki:

13.1065

13.1031

13.1025

13.1019

13.1002

13.0976

13.0964

13.0949

13.0899

13.0886

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.39.

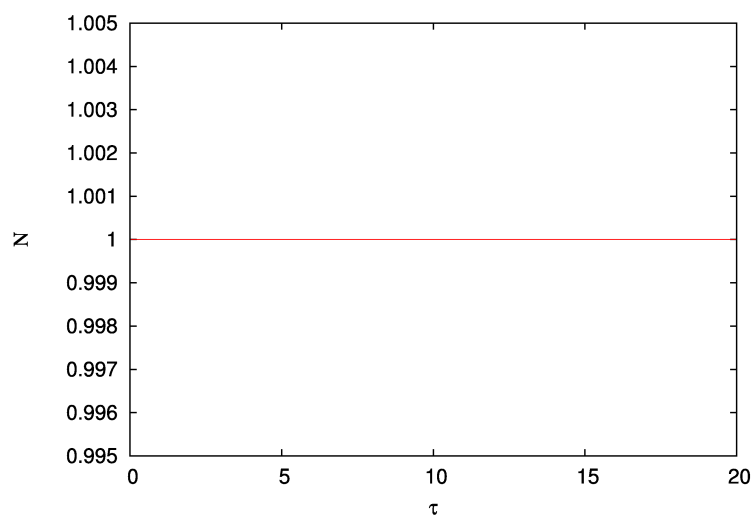
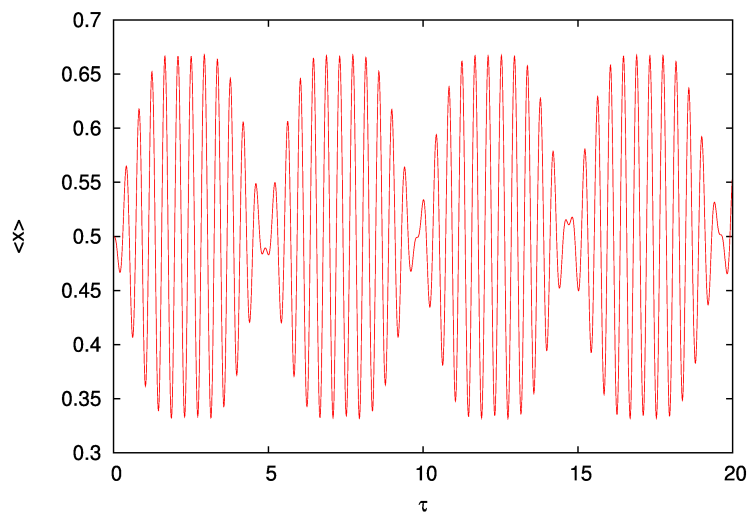
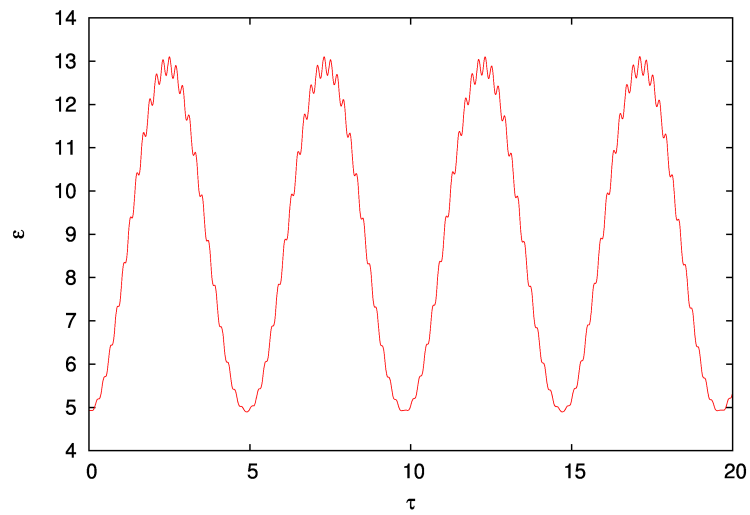


fig.39. Wykresy wielkości charakterystycznych, uzyskanych podczas ósmej symulacji w okolicach częstotliwości rezonansowej ($106\% \omega_{rez}$).

Seria 9:

Dziesięć największych wartości energii cząstki:

11.0546

11.0507

11.0498

11.0489

11.0427

11.0411

11.0359

11.0321

11.0303

11.0296

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.40.

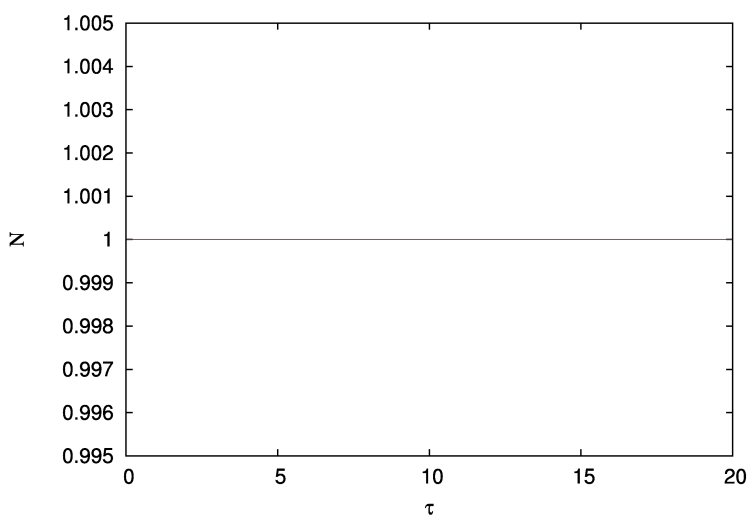
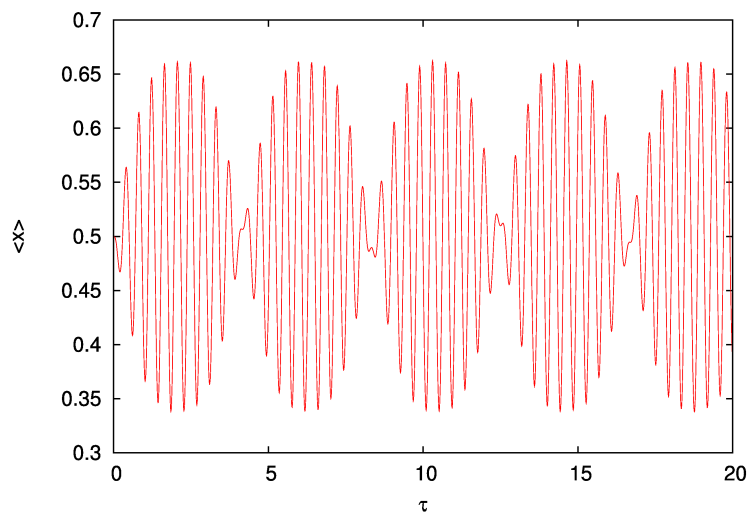
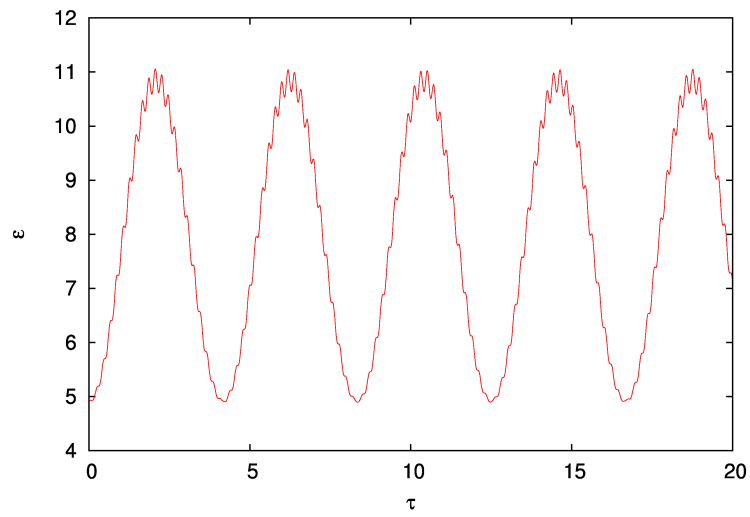


fig.40. Wykresy wielkości charakterystycznych, uzyskanych podczas dziewiątej symulacji w okolicach częstotliwości rezonansowej ($108\% \omega_{rez}$).

Seria 10:

Dziesięć największych wartości energii cząstki:

9.60945

9.60743

9.60272

9.59863

9.59741

9.59654

9.59652

9.59094

9.58235

9.58217

Wielkości charakterystyczne przedstawiają wykresy na rysunku fig.41.

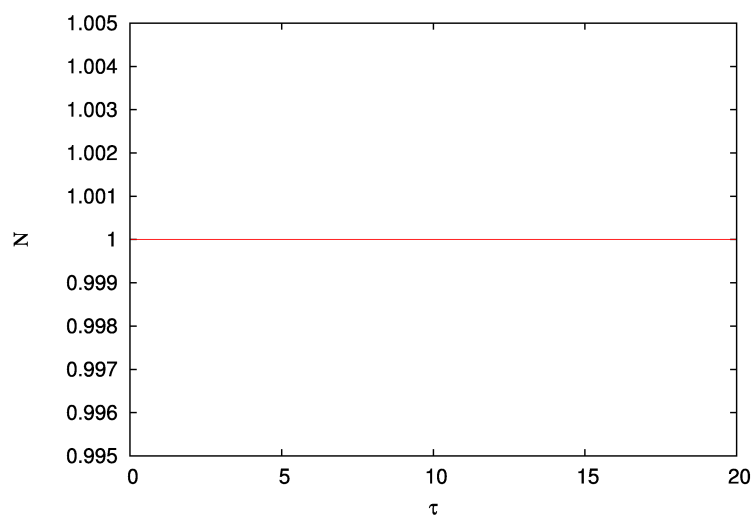
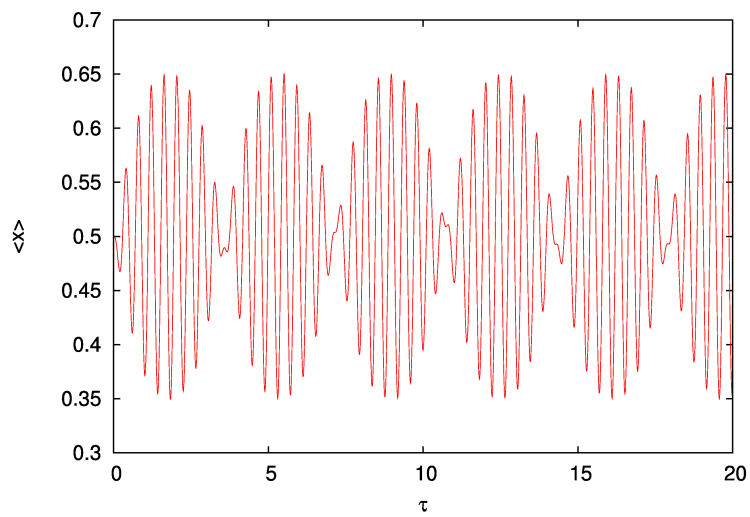
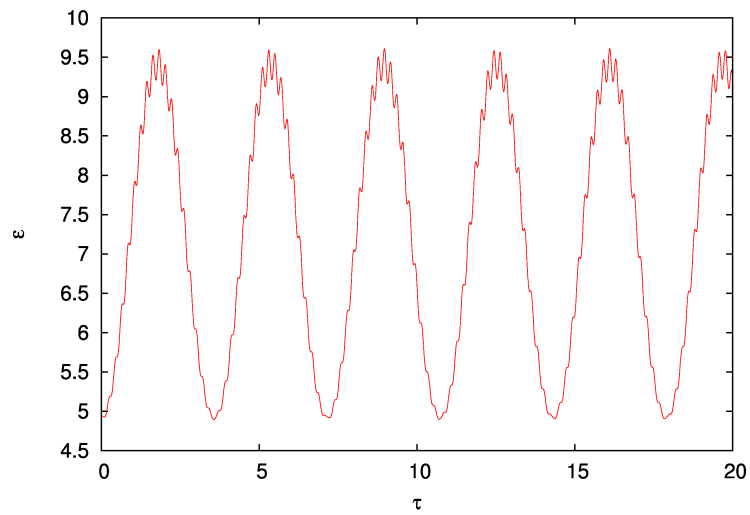


fig.41. Wykresy wielkości charakterystycznych, uzyskanych podczas dziesiątej symulacji w okolicach częstotliwości rezonansowej ($110\% \omega_{rez}$).

Wykres zależności największych wartości energii cząstki, od procenta częstości rezonansowej, przedstawia rysunek fig.42.

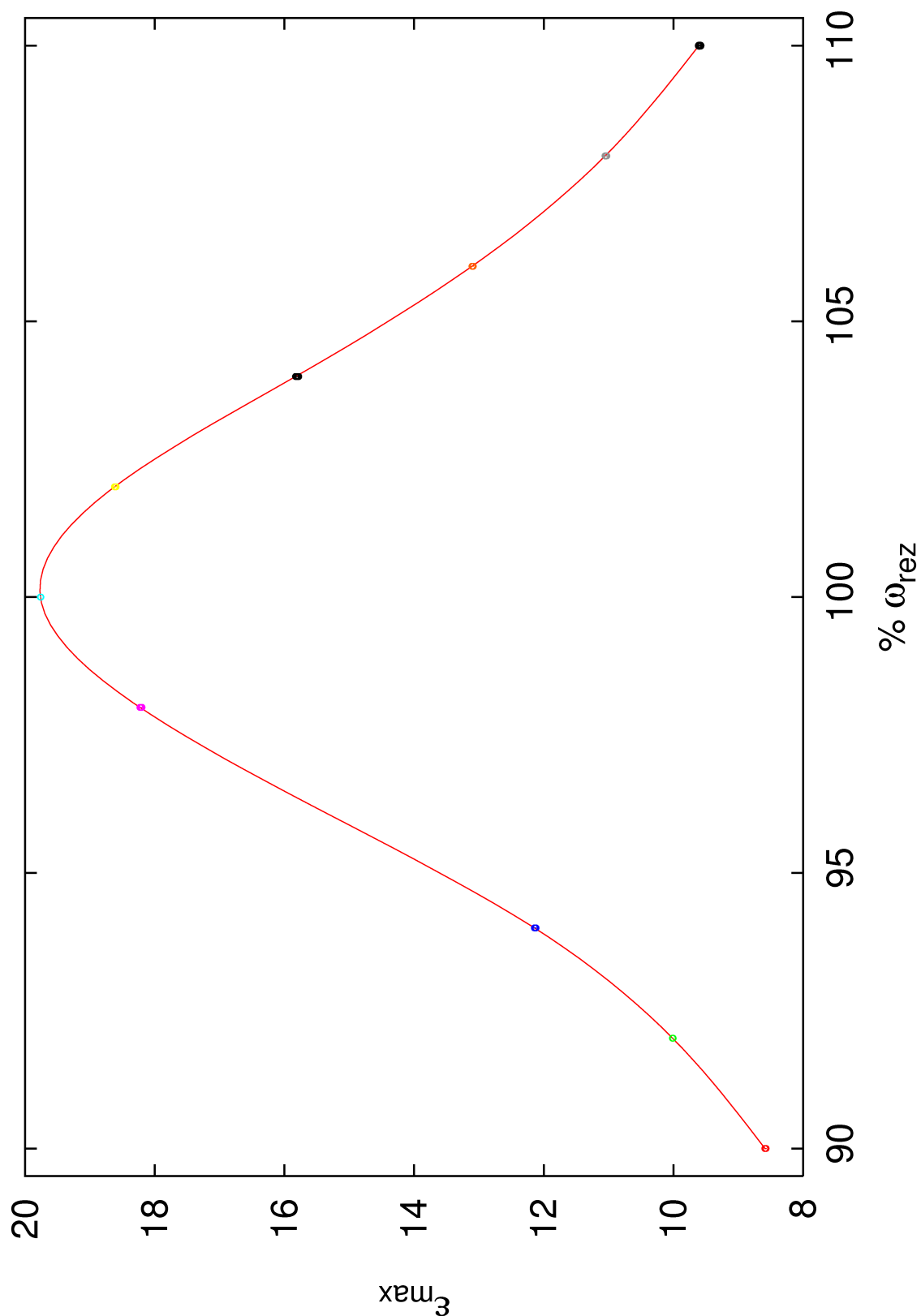


fig.42. Wykres zależności największych wartości energii cząstki, w funkcji procenta częstości rezonansowej; linia ciągła – interpolacja (cubic spline) łącząca wartości maksymalne.

Jak widać na powyższym rysunku, maksima energii układają się w krzywą dzwonową typu lorentzowskiego.

Wnioski

Model dobrze odwzorowuje zachowanie cząstki w jednowymiarowej studni potencjału, oddziałującej ze zmiennym polem elektrycznym. Niestety model działa poprawnie, głównie dla przejść między sąsiednimi poziomami energetycznymi. Być może, poprawę uzyskiwanych wyników można uzyskać, przez odpowiednie dobranie parametrów startowych, niestety mnie nie udało się odnaleźć takich.

Dodatek 1

Kod źródłowy programu (qd.cpp):

```
// model kwantowej, naladowanej czastki w pudle, w zmiennym polu elektrycznym
// (przestrzen jednowymiarowa)
// by kmkulcz@tempac.okwf.fuw.edu.pl
// (c) Kacper Kulczycki 2006
//*****
#include <iostream>
#include <fstream>
#include <iomanip>
#include <cmath>
#include <cstdlib>
#include <string>
using namespace std;
int n=0;//poziom energetyczny czastki
long double tau=0.0;//znormalizowany czas
long double delta_tau=-0.01;//znormalizowana dlugosc kroku czasowego
long double omega=-0.01;//znormalizowana czestosc drgan pola elektrycznego
long double kappa=-0.01;//znormalizowana amplituda pola elektrycznego
int tsym=-1;//czas symulacji
int fcout=-1;//czestosc zrzucania danych na standardowe wyjście
int ffile=-1; //czestosc zrzucania danych do pliku
long double Norm=0.0; // zmienna pomocnicza do obliczenia stalej normalizacyjnej
long double X=0.0; // zmienna pomocnicza do obliczenia sredniego polozenia
long double E=0.0; // zmienna pomocnicza do obliczenia sredniej energii czastki
char t='x';//typ pliku wynikowego z danymi
int i, j, x, y; //zmienne pomocnicze indeksowania
//*****
// struktura danych
typedef struct Data
{
    int N; // ilosc punktow
    long double delta_x; // wartosc odleglosci miedzy kolejnymi wybranymi punktami
    long double *x; // wartosci zmiennej przestrzennej, w wybranych punktach
    long double *H_r; // wartosci hamiltonianu dla poszczegolnych punktow - cz. rzeczywista
    long double *H_i; // wartosci hamiltonianu dla poszczegolnych punktow - cz. zespolona
    long double *Psi_r; // wartosci funkcji falowej dla poszczegolnych punktow - cz.
    rzeczywista
    long double *Psi_i; // wartosci funkcji falowej dla poszczegolnych punktow - cz.
    zespolona
    long double *ro; //wartosci gestosci prawdopodobienstwa zanalezienia czastki w
    poszczegolnych punktach
} Data;
//*****
// wkladanie argumentow do struktury
Data *NewArg(int NN)
{
    Data *Data1;
    Data1 = (Data *)malloc (sizeof(Data));
    Data1->delta_x=1.0/NN;
    Data1->N=NN+1;
    Data1->x = new long double[NN+1];
    Data1->H_r = new long double[NN+1];
    Data1->H_i = new long double[NN+1];
    Data1->Psi_r = new long double[NN+1];
```

```

Data1->Psi_i = new long double[NN+1];
Data1->ro = new long double[NN+1];
return(Data1);
}
//*****
//trick na zamiane int na char (od 0 do 9)
char Conv(int nbr)
{
    char dum;
    switch (nbr)
    {
        case 1:
            dum='1';
            break;
        case 2:
            dum='2';
            break;
        case 3:
            dum='3';
            break;
        case 4:
            dum='4';
            break;
        case 5:
            dum='5';
            break;
        case 6:
            dum='6';
            break;
        case 7:
            dum='7';
            break;
        case 8:
            dum='8';
            break;
        case 9:
            dum='9';
            break;
        case 0:
            dum='0';
            break;
    }
    return dum;
}
//*****
// tworzenie poczatkowych wartosci
void Initial(Data* D)
{
    for(i=0 ; i<(D->N) ; i++) //wartosci polozen
    {
        D->x[i]=(D->delta_x)*i;
    }
    for(i=0 ; i<(D->N) ; i++) // funkcja falowa i gestosc prawdopodobienstwa
    {
        D->Psi_r[i]=sin(n*M_PI*(D->x[i]))*M_SQRT2;
        D->Psi_i[i]=0.0;
        D->ro[i]=(D->Psi_r[i])*(D->Psi_r[i]);
    }
    D->H_r[0]=0.0; //hamiltonian
    D->H_i[0]=0.0;
    D->H_r[D->N]=0.0;
    D->H_i[D->N]=0.0;
    for(i=1 ; i<(D->N)-1 ; i++)
    {
        D->H_r[i]=(-0.5)*((D->Psi_r[i+1])+(D->Psi_r[i-1])-2.0*(D->Psi_r[i]))/((D->delta_x)*(D->delta_x));
        D->H_i[i]=(-0.5)*((D->Psi_i[i+1])+(D->Psi_i[i-1])-2.0*(D->Psi_i[i]))/((D->delta_x)*(D->delta_x));
    }
}

```

```

}
//*****
// krok ewolucji
void Evol_step(Data* D)
{
    for(i=0 ; i < (D->N); i++) //psi_r(tau+delta_tau/2)
    {
        D->Psi_r[i]+=(D->H_i[i])*delta_tau/2.0;
    }
    for(i=1 ; i < ((D->N)-1) ; i++) // H_r(tau+delta_tau/2)
    {
        D->H_r[i]=(-0.5)*((D->Psi_r[i+1])+(D->Psi_r[i-1]))-2.0*(D->Psi_r[i]))/((D-
>delta_x)*(D->delta_x))+kappa*(D->x[i]-0.5)*(D->Psi_r[i])*sin(omega*tau);
    }
    for(i=0 ; i<(D->N) ; i++)// psi_i(tau+delta_tau)
    {
        D->Psi_i[i]-=(D->H_r[i])*delta_tau;
    }
    for(i=1 ; i < ((D->N)-1) ; i++) // H_i(tau+delta_tau)
    {
        D->H_i[i]=(-0.5)*((D->Psi_i[i+1])+(D->Psi_i[i-1]))-2.0*(D->Psi_i[i]))/((D-
>delta_x)*(D->delta_x))+kappa*(D->x[i]-0.5)*(D->Psi_i[i])*sin(omega*tau);
    }
    for(i=0 ; i < (D->N) ; i++) //psi_r(tau+delta_tau)
    {
        D->Psi_r[i]+=(D->H_i[i])*delta_tau/2.0;
    }
    for(i=1 ; i < ((D->N)-1) ; i++) // H_r(tau+delta_tau)
    {
        D->H_r[i]=(-0.5)*((D->Psi_r[i+1])+(D->Psi_r[i-1]))-2.0*(D->Psi_r[i]))/((D-
>delta_x)*(D->delta_x))+kappa*(D->x[i]-0.5)*(D->Psi_r[i])*sin(omega*tau);
    }
}
//*****
// program modelujacy naladowana czastke w studni, w polu elektrycznym
// (dynamika kwantowa w jednym wymiarze)
int main()
{
    // "interface"

    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
    cout<<'\\n';
    for(i=0 ; i<15 ; i++)
    {
        cout<<" ";
    }
    cout<<"program modelujacy kwantowa, naladowana czastke\\n";
    for(i=0 ; i<11 ; i++)
    {
        cout<<" ";
    }
    cout<<"w jednowymiarowym pudle oraz zmiennym polu elektrycznym\\n";
    for(i=0 ; i<19 ; i++)
    {
        cout<<" ";
    }
    cout<<" by kmkulcz@tempac.okwf.fuw.edu.pl\\n";
    for(i=0 ; i<23 ; i++)
    {
        cout<<" ";
    }
    cout<<" (c) Kacper Kulczycki 2006\\n";
    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
}

```

```

    }
    cout<<'\n';

    // czesc glowna

    int* Nn= new int;//ilosc elementow na ktore dzielone jest pudlo
    *Nn=0;
    while(*Nn<1)
    {
        cout<<"Podaj ilosc czesci, na ktore bedzie podzielone pudlo:\n";
        cin>>*Nn;
    }
    while (!(delta_tau>0.0))
    {
        cout<<"Podaj znormalizowana dlugosc kroku czasowego (okolo 0.000001):\n";
        cin>>delta_tau;
    }
    while(n<1)
    {
        cout<<"Podaj poziom energetyczny zajmowany przez czastke, na poczatku:\n";
        cin>>n;
    }
    while(omega<0.0)
    {
        cout<<"Podaj znormalizowana czestosc pola elektrycznego:\n";
        cin>>omega;
    }
    while(kappa<0.0)
    {
        cout<<"Podaj znormalizowana amplituda pola elektrycznego:\n";
        cin>>kappa;
    }
    while(tsym<1)
    {
        cout<<"Podaj dlugosc czasu symulacji (w ilosci krokow):\n";
        cin>>tsym;
    }
    while(fcout<0)
    {
        cout<<"Podaj co ile krokow, dane maja byc wypisywane na ekran (w ilosci krokow):\n";
        cin>>fcout;
    }
    while(ffile<0)
    {
        cout<<"Podaj co ile krokow, dane maja byc zapisywane do pliku (w ilosci krokow):\n";
        cin>>ffile;
    }
    while(t!='g'&&t!='G'&&t!='m'&&t!='M')
    {
        cout<<"Podaj format danych w pliku:\n\tg,G - gnuplot\n\tm,M - matlab\n[g/G/m/M]";
        cin>>t;
    }
    switch(t)
    {
        case 'g':
            {
                for(i=0 ; i<80 ; i++)
                {
                    cout<<"-";
                }
                cout<<"\nDane beda znajdowac sie w kolejnych plikach: ro.dat.[0-9][0-9]..., gdzie
                liczba\inna koncu, odpowiadac bedzie numerowi zapisywanego kroku, dodatkowo
                zostanie\ntworzony plik dat.script, ktory umozliwi sekwencyjne wczytywanie i
                wyswietlanie\ntych plikow, oprócz tego powstana pliki z chwilowymi wartosciami:
                sredniej\nenergii ukladu: E.dat, sredniego polozenia czastki X.dat, stalej
                normalizacyjnej\Norm.dat;\n";
                for(i=0 ; i<80 ; i++)
                {
                    cout<<"-";
                }
            }
    }

```

```

    }
    cout<<'\n';
};
break;
case 'G':
{
    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
    cout<<"\nDane beda znajdowac sie w kolejnych plikach: ro.dat.[0-9][0-9]..., gdzie
liczba\inna koncu, odpowiadac bedzie numerowi zapisywanego kroku, dodatkowo
zostanie\ntworzony plik dat.script, ktory umozliwi sekwencyjne wczytywanie i
wyswietlanie\ntych plikow, oprócz tego powstana pliki z chwilowymi wartosciami:
sredniej\neenergii ukladu: E.dat, sredniego polozenia czastki X.dat, stalej
normalizacyjnej\nNorm.dat;\n";
    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
    cout<<'\n';
};
break;
case 'm':
{
    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
    cout<<"\nDane beda znajdowac sie w kolejnych plikach: ro.dat.[0-9][0-9]..., gdzie
liczba\inna koncu, odpowiadac bedzie numerowi zapisywanego kroku, dodatkowo
zostanie\ntworzony plik datscript.m, ktory umozliwi sekwencyjne wczytywanie i
wyswietlanie\ntych plikow, oprócz tego powstana pliki z chwilowymi wartosciami:
sredniej\neenergii ukladu: E.dat, sredniego polozenia czastki X.dat, stalej
normalizacyjnej\nNorm.dat;\n";
    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
    cout<<'\n';
};
break;
case 'M':
{
    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
    cout<<"\nDane beda znajdowac sie w kolejnych plikach: ro.dat.[0-9][0-9]..., gdzie
liczba\inna koncu, odpowiadac bedzie numerowi zapisywanego kroku, dodatkowo
zostanie\ntworzony plik datscript.m, ktory umozliwi sekwencyjne wczytywanie i
wyswietlanie\ntych plikow, oprócz tego powstana pliki z chwilowymi wartosciami:
sredniej\neenergii ukladu: E.dat, sredniego polozenia czastki X.dat, stalej
normalizacyjnej\nNorm.dat;\n";
    for(i=0 ; i<80 ; i++)
    {
        cout<<"-";
    }
    cout<<'\n';
};
break;
}
Data* data;
data=NewArg(*Nn);
delete Nn;
ofstream strm;
Initial(data);
for(i=0 ; i<(data->N) ; i++) // obliczanie zmiennych pomocniczych
{

```



```

    Norm+=(data->Psi_r[i])*(data->Psi_r[i])+(data->Psi_i[i])*(data->Psi_i[i]);
    X+=(data->x[i])*((data->Psi_r[i])*(data->Psi_r[i])+(data->Psi_i[i])*(data-
>Psi_i[i]));
    E+=(data->Psi_r[i])*(data->H_r[i])+(data->Psi_i[i])*(data->H_i[i]);
    data->ro[i]=(data->Psi_r[i])*(data->Psi_r[i])+(data->Psi_i[i])*(data->Psi_i[i]);
}
Norm*=data->delta_x;
X*=data->delta_x;
E*=data->delta_x;

//czy pisac do pliku?

if(ffile>0)
{
    strm.open("Norm.dat",ios::trunc);
    strm<<tau<<'\\t'<<Norm<<'\\n';
    strm.close();
    strm.open("X.dat",ios::trunc);
    strm<<tau<<'\\t'<<X<<'\\n';
    strm.close();
    strm.open("E.dat",ios::trunc);
    strm<<tau<<'\\t'<<E<<'\\n';
    strm.close();
    strm.open("ro.dat.0",ios::trunc);
    for(i=0; i<(data->N); i++)
    {
        strm<<data->x[i]<<'\\t'<<data->ro[i]<<'\\n';
    }
    strm.close();
    if(t=='g' || t=='G')
    {
        strm.open("dat.script",ios::trunc);
        strm<<"set xlabel '\\x'\\nset ylabel '\\ro'\\nset yrange [0:4.5]\\nplot \\\"ro.dat.0\\\"
notitle with lines\\npause 1\\n";
        strm.close();
    }
    else
    {
        strm.open("datscript.m",ios::trunc);
        strm<<"data0=zeros(\"<<data-
>N<<\",2);\\ndata0=load('ro.dat.0');\\nplot(data0(:,1),data0(:,2),'-r');\\npause(0.5);\\n";
        strm.close();
    }
}
//czy pisac?
if(fcout>0)
{
    cout<<"początkowa wartość stałej normalizacyjnej: "<<Norm<<'\\n';
    cout<<"początkowa wartość średniego położenia cząstki: "<<X<<'\\n';
    cout<<"początkowa średnia wartość energii cząstki: "<<E<<'\\n';
    X=data->x[0];
    Norm=data->ro[0];
    for(i=1; i<(data->N); i++)
    {
        if(data->ro[i]>Norm)
        {
            X=data->x[i];
            Norm=data->ro[i];
        }
    }
    cout<<"początkowe położenie o największej gęstości prawdopodobieństwa: "<<X<<'\\n';
    cout<<"początkowa największa wartość gęstości prawdopodobieństwa: "<<Norm<<'\\n';
}
Norm=0.0;
X=0.0;
E=0.0;

//część symulacyjna

```

```

for(y=1;y<tsym+1 ;y++)
{
    tau=delta_tau*y;
    Evol_step(data); // obliczenie psi i hamiltonianu po danym kroku
    for(i=0 ; i<(data->N) ; i++) // obliczanie zmienych pomocniczych
    {
        Norm+=(data->Psi_r[i])*(data->Psi_r[i])+(data->Psi_i[i])*(data->Psi_i[i]);
        X+=(data->x[i])*((data->Psi_r[i])*(data->Psi_r[i])+(data->Psi_i[i])*(data->Psi_i[i]));
    }
    E+=(data->Psi_r[i])*(data->H_r[i])+(data->Psi_i[i])*(data->H_i[i]);
    data->ro[i]=(data->Psi_r[i])*(data->Psi_r[i])+(data->Psi_i[i])*(data->Psi_i[i]);
}
Norm*=data->delta_x;
X*=data->delta_x;
E*=data->delta_x;

//czy pisac do pliku?

if(ffile>0&&y%ffile==0)
{
    strm.open("Norm.dat",ios::app);
    strm<<tau<<'\\t'<<Norm<<'\\n';
    strm.close();
    strm.open("X.dat",ios::app);
    strm<<tau<<'\\t'<<X<<'\\n';
    strm.close();
    strm.open("E.dat",ios::app);
    strm<<tau<<'\\t'<<E<<'\\n';
    strm.close();
    string nazwa("ro.dat.");
    int Nn=y;
    x=y;
    for(j=(int)(floor(log10(y*1.0))+1;j>0;j--)
    {
        Nn=x;
        for(i=0;i<j-1;i++)
        {
            Nn/=10;
        }
        for(i=0;i<j-1;i++)
        {
            Nn*=10;
        }
        x-=Nn;
        for(i=0;i<j-1;i++)
        {
            Nn/=10;
        }
        nazwa.resize(nazwa.size()+1,Conv(Nn));
    }
    strm.open(nazwa.data(),ios::app);
    for(i=0; i<data->N; i++)
    {
        strm<<data->x[i]<<'\\t'<<data->ro[i]<<'\\n';
    }
    strm.close();
    if(t=='g' || t=='G')
    {
        strm.open("dat.script",ios::app);
        strm<<"plot \"ro.dat.\"<<y<<"\" notitle with lines\npause 1\n";
        strm.close();
    }
    else
    {
        strm.open("datscript.m",ios::app);
        strm<<"data"<<y<<"=zeros("<<data->N<<" ,2);\\n"<<"data"<<y<<"=load('ro.dat.'"<<y<<"');\\n\\nplot(data"<<y<<"(:,1),data"<<y<<" ,'-r');\\npause(0.5);\\n";
        strm.close();
    }
}

```

```

    }
}

// czy pisac ?

if(fcout>0&&y%fcout==0)
{
    cout<<"wartosc stalej normalizacyjnej po "<<y<<" krokach: "<<Norm<<'\n';
    cout<<"wartosc sredniego polozenia czastki po "<<y<<" krokach: "<<X<<'\n';
    cout<<"srednia wartosc energii czastki po "<<y<<" krokach: "<<E<<'\n';
    X=data->x[0];
    Norm=data->ro[0];
    for(i=1; i<(data->N); i++)
    {
        if(data->ro[i]>Norm)
        {
            X=data->x[i];
            Norm=data->ro[i];
        }
    }
    cout<<"polozenie o najwiekszej gestosci prawdopodobienstwa po "<<y<<" krokach:
"<<X<<'\n';
    cout<<"najwieksza wartosc gestosci prawdopodobienstwa po "<<y<<" krokach:
"<<Norm<<'\n';
}
Norm=0.0;
X=0.0;
E=0.0;
}

return 0;
}

```

Dodatek 2

Skrypty shellowe do obróbki danych z symulacji:

Dodatek 2a

Skrypt przetwarzający dane z symulacji testowych (skrypt_1.sh):

```

#!/bin/sh

# (c) Kacper Kulczycki 2007

# Skrypt generujący wykresy podstawowych wielkości, generowanych
# podczas testowych symulacji.

# Script generates plots of basic quantities, generated during test
# simulations.

LICZNIK=1
echo "# Tworzenie plików skryptów w katalogach"
while [ -e qd"$LICZNIK" ]
do
    echo "# skrypt.[1-7] -> qd$LICZNIK"
    cp -f skrypt.[1-7] qd"$LICZNIK"
    LICZNIK=`expr $LICZNIK + 1`
done
cd qd1
LICZNIK=`expr $LICZNIK - 1`
echo "# Przechodzenie do katalogów"
while [ $LICZNIK -gt 0 ]
do
    cd "../qd$LICZNIK"

```

```

echo "# katalog qd$LICZNIK"
cat skrypt.3 | gnuplot
echo E"$LICZNIK".t.ps
mv E.ps ../qd_wykresy/E"$LICZNIK".t.ps
cat skrypt.4 | gnuplot
echo X"$LICZNIK".t.ps
mv X.ps ../qd_wykresy/X"$LICZNIK".t.ps
cat skrypt.5 | gnuplot
echo N"$LICZNIK".t.ps
mv N.ps ../qd_wykresy/N"$LICZNIK".t.ps
cat skrypt.6 | gnuplot
echo RO"$LICZNIK".t.ps
mv RO.ps ../qd_wykresy/RO"$LICZNIK".t.ps
KATALOG="$PWD"
cd ../qd_wykresy
psmerge -ot"$LICZNIK".ps RO"$LICZNIK".t.ps X"$LICZNIK".t.ps E"$LICZNIK".t.ps
N"$LICZNIK".t.ps
psnup -4 -q -pa4 -m1.0cm t"$LICZNIK".ps > test"$LICZNIK".ps
cd ..
./ps2png qd_wykresy/test"$LICZNIK".ps
ps2epsi qd_wykresy/test"$LICZNIK".ps
mv qd_wykresy/test"$LICZNIK".png qd_wykresy/ilustracje/png/
mv ./test"$LICZNIK".epsi qd_wykresy/ilustracje/eps/test"$LICZNIK".eps
mv qd_wykresy/test"$LICZNIK".ps qd_wykresy/ilustracje/ps
mv qd_wykresy/*.ps qd_wykresy/ps
cd "$KATALOG"
LICZNIK=`expr $LICZNIK - 1`
done

```

Dodatek 2b

Skrypt przetwarzający dane z symulacji przejść między poziomami (skrypt_2.sh):

```

#!/bin/sh

# (c) Kacper Kulczycki 2007

# Skrypt generujący wykresy podstawowych wielkości, generowanych
# podczas symulacji przejść.

# Script generates plots of basic quantities, generated during
# simulations of passes.

LICZNIK=1
echo "# Tworzenie plików skryptów w katalogach"
while [ -e qd_"$LICZNIK" ]
do
  echo "# skrypt.[1-7] -> qd_$LICZNIK"
  cp -f skrypt.[1-7] qd_"$LICZNIK"
  LICZNIK=`expr $LICZNIK + 1`
done
cd qd_1
LICZNIK=`expr $LICZNIK - 1`
echo "# Przechodzenie do katalogów"
while [ $LICZNIK -gt 0 ]
do
  cd "../qd_$LICZNIK"
  echo "# katalog ./qd_$LICZNIK"
  cat skrypt.3 | gnuplot
  echo E"$LICZNIK".r.ps
  mv E.ps ../qd_wykresy/E"$LICZNIK".r.ps
  cat skrypt.4 | gnuplot
  echo X"$LICZNIK".r.ps
  mv X.ps ../qd_wykresy/X"$LICZNIK".r.ps
  cat skrypt.5 | gnuplot
  echo N"$LICZNIK".r.ps
  mv N.ps ../qd_wykresy/N"$LICZNIK".r.ps

```

```

KATALOG="$PWD"
cd ../qd_wykresy
psmerge -or"$LICZNIK".ps E"$LICZNIK"r.ps X"$LICZNIK"r.ps N"$LICZNIK"r.ps
psnup -3 -q -pA4 -m1.0cm r"$LICZNIK".ps | pstops -b -q "1:0U(1w,1h)" | psresize -pA4>
przejscie"$LICZNIK".ps
cd ..
./ps2png qd_wykresy/przejscie"$LICZNIK".ps
ps2epsi qd_wykresy/przejscie"$LICZNIK".ps
mv qd_wykresy/przejscie"$LICZNIK".png qd_wykresy/ilustracje/png/
mv ./przejscie"$LICZNIK".epsi qd_wykresy/ilustracje/eps/przejscie"$LICZNIK".eps
mv qd_wykresy/przejscie"$LICZNIK".ps qd_wykresy/ilustracje/ps
mv qd_wykresy/*.ps qd_wykresy/ps
cd "$KATALOG"
LICZNIK=`expr $LICZNIK - 1`
done

```

Dodatek 2c

Skrypt przetwarzający dane z symulacji okolic rezonansu (skrypt_3.sh):

```

#!/bin/sh

# (c) Kacper Kulczycki 2007

# Skrypt generujący wykresy podstawowych wielkości, generowanych
# podczas symulacji okolic rezonansu.

# Script generates plots of basic quantities, generated during
# simulations of surroundings of resonance.

echo "# Tworzenie plików skryptów w katalogach"
for LICZNIK in 90 92 94 98 100 102 104 106 108 110
do
    echo "# skrypt.[1-7] -> qd-$LICZNIK"
    cp -f skrypt.[1-7] qd-"$LICZNIK"
done
cd qd_1
INDEX=$NULL
echo "# Przechodzenie do katalogów"
for LICZNIK in 90 92 94 98 100 102 104 106 108 110
do
    cd ../qd-"$LICZNIK"
    echo "# katalog qd-$LICZNIK"
    if [ -e Eb.max.dat ]
    then
        rm Eb.max.dat
    fi
    cat skrypt.3 | gnuplot
    echo E"$LICZNIK"b.ps
    mv E.ps ../qd_wykresy/E"$LICZNIK"b.ps
    cat skrypt.4 | gnuplot
    echo X"$LICZNIK"b.ps
    mv X.ps ../qd_wykresy/X"$LICZNIK"b.ps
    cat skrypt.5 | gnuplot
    echo N"$LICZNIK"b.ps
    mv N.ps ../qd_wykresy/N"$LICZNIK"b.ps
    KATALOG="$PWD"
    cd ../qd_wykresy
    psmerge -ob"$LICZNIK".ps E"$LICZNIK"b.ps X"$LICZNIK"b.ps N"$LICZNIK"b.ps
    psnup -3 -q -pA4 -m1.0cm b"$LICZNIK".ps | pstops -b -q "1:0U(1w,1h)" | psresize -pA4 >
    rezonans"$LICZNIK"proc.ps
    cd ..
    ./ps2png qd_wykresy/rezonans"$LICZNIK"proc.ps
    ps2epsi qd_wykresy/rezonans"$LICZNIK"proc.ps
    mv qd_wykresy/rezonans"$LICZNIK"proc.png qd_wykresy/ilustracje/png/
    mv ./rezonans"$LICZNIK"proc.epsi qd_wykresy/ilustracje/eps/rezonans"$LICZNIK"proc.eps
    mv qd_wykresy/rezonans"$LICZNIK"proc.ps qd_wykresy/ilustracje/ps

```

```

mv qd_wykresy/*.ps qd_wykresy/ps
cd "$KATALOG"

# wyszukiwanie dziesieciu najwiekszych wartosci energii

# finding ten greatest values of mean energy

echo "# najwieksze wartosci sredniej energii:"
echo -e "# czas\tenergia"
LICZNIK2=$NULL
for NUMBER in 1 2 3 4 5 6 7 8 9 10
do

# TEMP najwieksza wartosc calkowita

# TEMP - greatest integer value

TEMP=`cut -f 2 E.dat | sort -nrk 1 -t "." | cut -d "." -f 1 | head -n 1`

# NUMBER wartosc zmiennoprzecinkowa NUMBER-owa w kolejnosci

# NUMBER - NUMBERth floating point value

NUMBER=`cut -f 2 E.dat | tr '.' a | grep "$TEMP"a | sort -nurk 2 -t "a" | head -n
"$NUMBER" |tail -n 1`

# w LICZNIK2 linie zawierajace liczbe NUMBER w polu energii

# LICZNIK2 - lines containing number NUMBER in energy field

TEMP=`cat E.dat | tr '.' a | sed s/'a'/'\./' | grep "$NUMBER"| tr a '.'`
NUMBER=`echo $NUMBER | tr a '.'`
LICZNIK2=`echo -e "$LICZNIK2\n$TEMP"`
echo -e "$LICZNIK2\t$NUMBER">>Eb.max.dat
done

# usuwanie zbednych linii

# deleting unnecessarily lines

TEMP=`echo "$LICZNIK2" | wc -l`
TEMP=`expr $TEMP - 1`
LICZNIK2=`echo "$LICZNIK2" | tail -n "$TEMP"|sort -rt "." -k 3`
echo "$LICZNIK2"
done
cd ..
MAXIMA=$NULL
if [ -e Eb.max.dat ]
then
rm Eb.max.dat
fi
for LICZNIK in 90 92 94 98 100 102 104 106 108 110
do
TEMP=`cat qd-"$LICZNIK"/Eb.max.dat | head -n 1`
echo "$TEMP">>Eb.max.dat
done
NUMBER=`echo -e "set autoscale\nset xr [89.5:110.5]\nset xlabel '% {/Symbol
w}_{rez}'\nset ylabel '{/Symbol e}_{max}'\nplot 'Eb.max.dat' notitle smooth cspline with
lines 1 2, 'qd-90/Eb.max.dat' notitle with points 1 6, 'qd-92/Eb.max.dat' notitle with
points 2 6, 'qd-94/Eb.max.dat' notitle with points 3 6, 'qd-98/Eb.max.dat' notitle with
points 4 6, 'qd-100/Eb.max.dat' notitle with points 5 6, 'qd-102/Eb.max.dat' notitle with
points 6 6, 'qd-104/Eb.max.dat' notitle with points 7 6, 'qd-106/Eb.max.dat' notitle with
points 8 6, 'qd-108/Eb.max.dat' notitle with points 9 6, 'qd-110/Eb.max.dat' notitle with
points -1 6\n"`
TEMP=`cat skrypt.1`
NUMBER=`echo -e "$NUMBER\n$TEMP"`
NUMBER=`echo -e "$NUMBER\nset output 'Eb.max.ps'\n"`
TEMP=`cat skrypt.2`
NUMBER=`echo -e "$NUMBER\n$TEMP"`

```

```

echo "$NUMBER" | gnuplot
echo Eb.max.ps
psresize -pA4 Eb.max.ps ./qd_wykresy/Eb.max.ps
./ps2png qd_wykresy/Eb.max.ps
ps2epsi qd_wykresy/Eb.max.ps
mv qd_wykresy/Eb.max.png qd_wykresy/ilustracje/png/Erez.max.png
mv ./Eb.max.epsi qd_wykresy/ilustracje/eps/Erez.max.eps
mv qd_wykresy/Eb.max.ps qd_wykresy/ilustracje/ps/Erez.max.ps

```

Dodatek 2d

Skrypt przetwarzający dane o ewolucji gęstości prawdopodobieństwa w symulacjach przejść (skrypt.spect.sh):

```

#!/bin/sh

# (c) Kacper Kulczycki

# Skrypt tworzący tablice wykresów gęstości prawdopodobieństwa dla
# różnych wartości energii układu (maksymalnej, zbliżonych do
# maksymalnej, średniej – jeśli wystąpiła, zbliżonych do średniej,
# zbliżonych do minimalnej i minimalnej).

# Script generates table of plots, of density of probability for
# various values of energy, of the system (maximal, nearing maximal,
# mean – if it was, nearing mean, nearing minimal and minimal).

NUMBER="1"
cd qd_1
while [ -e ../qd_"$NUMBER" ]
do
echo "# Praca w katalogu qd_$NUMBER "
cd ../qd_"$NUMBER"

# przetwarzanie skryptu dla gnuplota produkowanego przez program qd

# processing script for gnuplot generated with qd program

cat dat.script |grep -v pause | grep -v "set yrange" |cut -d '"' -f 2 > dat.script.2
TEMP2=`cat dat.script.2| wc -l`
TEMP=`cat dat.script.2 | cut -d '.' -f 3`
LICZNIK="$NULL"
for DUMM in $TEMP
do
DUMM=`echo "scale=9; $DUMM * 0.000001" | bc`
MAXMIN=`echo $DUMM | cut -f 1 -d '.'`
if test -z `echo "$MAXMIN"`
then
DUMM=`echo 0"$DUMM"`
fi
DUMM=`echo :"$DUMM"0| tr '.' a`
LICZNIK=`echo -e "$LICZNIK\n$DUMM"`
done
echo "$LICZNIK" | tail -n "$TEMP2" > dat.script.2
cat dat.script |grep -v pause | grep -v "set yrange" | cut -d '"' -f 2 | paste
dat.script.2 - > dat.script.sort

# w pliku dat.script.sort czasy i nazwy plików

# filenames and times in dat.script.sort file

rm dat.script.2
echo "# gotowy plik dat.script.sort"

# szukanie maximów i minimów

# looking for maximas nad minimas

```

```

CALOSC=$NULL

# wyszukiwanie pieciu najwiekszych wartosci energii
# finding five greatest values of energy

for LICZNIK in 1 2 3 4 5
do

# w TEMP najwieksza wartosc calkowita
# TEMP - greatest integer value

TEMP=`cut -f 2 E.dat | sort -nrk 1 -t "." | cut -d "." -f 1 | head -n 1`

# w DUMM wartosc zmiennoprzecinkowa LICZNIK-owa w kolejnosci
# DUMM - LICZNIKth floating point value

DUMM=`cut -f 2 E.dat | tr '.' a | sed -e a\:_:|tr '\n' 'b'|tr '_' '\n' | sed s/b// |
sed s/b// | grep :"$TEMP"a | sort -urk 2 -t "a" | head -n "$LICZNIK" | tail -n 1`

# CALOSC - linie zawierajace liczbe DUMM w polu energii
# CALOSC - lines containing number DUMM in energy field

TEMP=`cat E.dat | tr '.' a | sed s/'a'/'\./' | sed s/'\t'/'\t:/' | sed a\:_ | tr '\n'
'b'|tr '_' '\n'| sed s/b// | sed s/b// | grep -m 1 "$DUMM" | tr a '.' | sed s/:'// | sed
s/:'//`
CALOSC=`echo -e "$CALOSC\n$TEMP"`
done

# usuwanie zbednych linii
# deleting unnecessarily lines

TEMP=`echo "$CALOSC" | wc -l`
TEMP=`expr $TEMP - 1`
CALOSC=`echo "$CALOSC" | tail -n "$TEMP"`
MAXIMUM=`echo "$CALOSC" | head -n 1 |cut -f 2`
echo -e "# maksimum:\n$MAXIMUM"
echo -e "# wartosci zblizone do maksimum:\n`echo "$CALOSC" | tail -n "$TEMP"| tail -n 4
|cut -f 2`"
echo "# pliki z najwiekszymi energiami:"
INDEX=`echo "$CALOSC" | tr '[:blank:]' ':' | cut -f 1 -d ':'`
for INDEX2 in $INDEX
do
DUMM=`echo "scale=9; $INDEX2 * 1.0 " | bc`
MAXMIN=`echo $DUMM | cut -f 1 -d '.'`
if test -z `echo "$MAXMIN"`
then
DUMM=`echo 0"$DUMM"`
fi
DUMM=`echo :"$DUMM"0 | tr '.' a`
done

# w LICZNIK numer pliku z dana wartoscia
# LICZNIK - file number with that value

LICZNIK=`cat dat.script.sort | grep "$DUMM" | cut -f 3 -d '.'`

# skrypt.7 wymaga podania numeru w nazwie plotowanego pliku
# skrypt.7 needs giving number from filename

for MAXMIN in $LICZNIK
do
./skrypt.7 "$MAXMIN" "$INDEX2" | gnuplot
echo RO"$NUMBER"_"$MAXMIN".r.ps
done

```



```

mv ROr.ps ../qd_wykresy/RO"$NUMBER_"$MAXMIN"r.ps
done
done
MAXMIN="0"
CALOSC=$NULL
LICZNIK2="0"

# wyszukiwanie pieciu najmniejszych wartosci energii
# finding five least values of energy

for LICZNIK in 1 2 3 4 5
do

# w TEMP najmniejsza wartosc calkowita
# TEMP - least integer value

TEMP=`cut -f 2 E.dat | sort -nk 1 -t "." | cut -d "." -f 1 | head -n 1`

# w DUMM wartosc zmiennoprzecinkowa LICZNIK-owa w kolejnosci
# DUMM - LICZNIKth floating point value

DUMM=`cut -f 2 E.dat | tr '.' a | sed -e a\:_:|tr '\n' 'b'|tr '_' '\n' | sed s/b// |
sed s/b// | grep :"$TEMP"a | sort -uk 2 -t "a" | head -n "$LICZNIK" | tail -n 1 `

# CALOSC - linie zawierajace liczbe DUMM w polu energii
# CALOSC - lines containing number DUMM in energy field

TEMP=`cat E.dat | tr '.' a | sed s/'a'/'\./' | sed s/'\t'/'\t:/' | sed a\:_ | tr '\n'
'b'|tr '_' '\n'| sed s/b// | sed s/b// | grep -m 1 "$DUMM"| tr a '.' | sed s/':'/'// |sed
s/':'/'// `
CALOSC=`echo -e "$CALOSC\n$TEMP"`
done

# usuwanie zbednych linii
# deleting unnecessarily lines

TEMP=`echo "$CALOSC" | wc -l`
TEMP=`expr $TEMP - 1`
CALOSC=`echo "$CALOSC" | tail -n "$TEMP"`
MINIMUM=`echo "$CALOSC" | head -n 1|cut -f 2`
echo -e "# minimum:\n$MINIMUM"
echo -e "# wartosci zblizone do minimum:\n`echo "$CALOSC" | tail -n "$TEMP"| tail -n 4 |
cut -f 2`"
echo "# pliki z najmniejszymi energiami:"
INDEX=`echo "$CALOSC" | tr '[:blank:]' ':' | cut -f 1 -d ':'`
for INDEX2 in $INDEX
do
DUMM=`echo "scale=9; $INDEX2 * 1.0 " | bc`
MAXMIN=`echo $DUMM | cut -f 1 -d '.'`
if test -z `echo "$MAXMIN"`
then
DUMM=`echo 0"$DUMM"`
fi
DUMM=`echo :"$DUMM"0 | tr '.' a`

# w LICZNIK numer pliku z dana wartoscia
# LICZNIK - file number with that value

LICZNIK=`cat dat.script.sort | grep "$DUMM" | cut -f 3 -d '.'`

# skrypt.7 wymaga podania numeru w nazwie plotowanego pliku
# skrypt.7 needs giving number from filename

```

```

for MAXMIN in $LICZNIK
do
./skrypt.7 "$MAXMIN" "$INDEX2" | gnuplot
echo RO"$NUMBER_"$MAXMIN".r.ps
mv ROr.ps ../qd_wykresy/RO"$NUMBER_"$MAXMIN".r.ps
done
done

# szukanie sredniej wartosci energii

# looking for mean

MAXMIN=`echo "scale=9; $MINIMUM / 2 + $MAXIMUM / 2 " | bc`

# usuwanie zer z konca liczby

# deleting zeros from end of number

CALOSC=`echo $MAXMIN |cut -d "0" -f 1`
LICZNIK="0"
for INDEX in 2 3 4 5 6 7 8 9 10 11
do
INDEX2=`echo $MAXMIN | cut -d "0" -f $INDEX`
if test -n "$INDEX2"
then
CALOSC=`echo "$CALOSC$LICZNIK$INDEX2"`
LICZNIK="0"
else
LICZNIK=`echo "$LICZNIK"0`
fi
done
echo -e "# srednia:\n$CALOSC"
echo "# pliki ze srednia energia:"
NUMBER2=`echo :"$CALOSC" | tr '.' a `

# TEMP linie zawierajace liczbe CALOSC w polu energii

# TEMP - lines containing number CALOSC in energy field

TEMP=`cat E.dat | tr '.' a | sed s/'a'/'\./' | sed s/'\t'/'\t:/' | sed a\0_ | tr '\n'
'b' |tr '_' '\n'| sed s/b// | sed s/b// | grep "$NUMBER2"| sort -uk 2 -t 'a' | head -n 1
| tr a '.' | sed s/:'// `

# w INDEX lista wszystkich czasow

# INDEX - list of all times

INDEX=`echo "$TEMP" | tr '[:blank:]' ':' | cut -f 1 -d ':'`
for INDEX2 in $INDEX
do
DUMM=`echo "scale=9; $INDEX2 * 1.0 " | bc`
LICZNIK=`echo $DUMM | cut -f 1 -d '.'`
if test -z `echo "$LICZNIK"`
then
DUMM=`echo 0"$DUMM"`
fi
DUMM=`echo :"$DUMM"0 |tr '.' a`

# w MAXMIN numer pliku z dana wartoscia

# MAXMIN - file number with that value

MAXMIN=`cat dat.script.sort | grep "$DUMM" | cut -f 3 -d '.'`

# skrypt.7 wymaga podania numeru w nazwie plotowanego pliku

# skrypt.7 needs giving number from filename

```

```

for LICZNIK in $MAXMIN
do
./skrypt.7 "$LICZNIK" "$INDEX2" | gnuplot
echo RO"$NUMBER_"$LICZNIK".r.ps
mv ROr.ps ../qd_wykresy/RO"$NUMBER_"$LICZNIK".r.ps
done
done

# wyszukiwanie dwóch najmniejszych/największych wartości energii większych/mniejszych od
średniej

# finding two least/greatest values of energy bigger/smaller than mean

INDEX=$NULL
INDEX2=$NULL
TEMP=`echo "scale=9; $CALOSC * 0.9 " | bc`
TEMP2=`echo "scale=9; $CALOSC * 1.1 " | bc`
MAXMIN=`cut -f 2 E.dat`
for DUMM in $MAXMIN
do
if [ $(echo "scale=9; $DUMM > $CALOSC " | bc) -ne 0 ]
then
INDEX=`echo -e "$INDEX\n$DUMM"`
fi
if [ $(echo "scale=9; $DUMM < $CALOSC " | bc) -ne 0 ]
then
INDEX2=`echo -e "$INDEX2\n$DUMM"`
fi
done
DUMM=`echo "$INDEX"| wc -l`
DUMM=`expr $DUMM - 1`

# w INDEX liczby z gornego otoczenia sredniej

# INDEX - mean upper surrounding numbers

INDEX=`echo "$INDEX"|tail -n "$DUMM"`
DUMM=`echo "$INDEX2"|wc -l`
DUMM=`expr $DUMM - 1`

# w INDEX2 liczby z dolnego otoczenia sredniej

# INDEX - mean lower surrounding numbers

INDEX2=`echo "$INDEX2"|tail -n "$DUMM"`

# znajdowanie liczby najmniejszej/największej i kolejnej większej/mniejszej

# finding smalest/bigest and next greater/least numbers

LICZNIK="$MAXIMUM"
for DUMM in $INDEX
do
if [ $(echo "scale=9; $LICZNIK > $DUMM " | bc) -ne 0 ]
then
LICZNIK="$DUMM"
fi
done
CALOSC=`echo "$INDEX" | tr '.' ':' | grep -v $(echo "$LICZNIK"| tr '.' ':') | tr ':'
','`
TEMP2="$MAXIMUM"
for DUMM in $CALOSC
do
if [ $(echo "scale=9; $TEMP2 > $DUMM " | bc) -ne 0 ]
then
TEMP2="$DUMM"
fi
done

```

```

# w INDEX dwie najblizsze liczby z gornego otoczenia sredniej
# INDEX - two nearest upper surrounding numbers

INDEX=`echo -e "$TEMP2\n$LICZNIK"`
LICZNIK="$MINIMUM"
for DUMM in $INDEX2
do
  if [ $(echo "scale=9; $LICZNIK < $DUMM " | bc) -ne 0 ]
  then
    LICZNIK=$DUMM
  fi
done
CALOSC=`echo "$INDEX2" | tr '.' ':' | grep -v $(echo "$LICZNIK"|tr '.' ':') | tr ':' '.'`
TEMP="$MINIMUM"
for DUMM in $CALOSC
do
  if [ $(echo "scale=9; $TEMP < $DUMM " | bc) -ne 0 ]
  then
    TEMP=$DUMM
  fi
done

# w INDEX2 dwie najblizsze liczby z dolnego otoczenia sredniej
# INDEX2 - two nearest lower surrounding numbers

INDEX2=`echo -e "$LICZNIK\n$TEMP"`

# w MINMAX wartosci zmiennoprzecinkowe z otoczenia sredniej
# MINMAX - mean surrounding floating point values

MAXMIN=`echo -e "$INDEX\n$INDEX2"`
echo "# wartosci najbardziej zblizone do sredniej:"
echo "$MAXMIN"
echo "# pliki z energiami zblizonymi do sredniej:"
CALOSC=$NULL

# w INDEX wartosc zmiennoprzecinkowa z otoczenia sredniej
# INDEX - mean surrounding floating point value

for INDEX in $MAXMIN
do

# w CALOSC linie zawierajace liczbe INDEX w polu energii
# CALOSC - lines containing number INDEX in energy field

NUMBER2=`echo :"$INDEX": | tr '.' a `
TEMP=`cat E.dat | tr '.' a | sed s/'a'/'\./ | sed s/'\t'/'\t:/' | sed a\:_ | tr '\n'
'b' |tr '_' '\n' | sed s/b// | sed s/b// | grep -m 1 "$NUMBER2" | tr a '.' | sed s/:'// `
| sed s/:'// `
CALOSC=`echo -e "$CALOSC\n$TEMP"`
done

# usuwanie zbednych linii
# deleting unnecessarily lines

TEMP=`echo "$CALOSC" | wc -l`
TEMP=`expr $TEMP - 1`
CALOSC=`echo "$CALOSC" | tail -n "$TEMP"`
MAXIMUM=`echo "$CALOSC" | head -n 1 | cut -f 2`
INDEX=`echo "$CALOSC" | tr '[:blank:]' ':' | cut -f 1 -d ':'`

# w INDEX numery "czasow" - lini

```

```

# INDEX - time field values from lines

for INDEX2 in $INDEX
do
  DUMM=`echo "scale=9; $INDEX2 * 1.0 " | bc`
  MAXMIN=`echo $DUMM | cut -f 1 -d '.'`
  if test -z `echo "$MAXMIN"`
  then
    DUMM=`echo 0"$DUMM"`
  fi
  DUMM=`echo :"$DUMM"0 |tr '.' a`

# w TEMP numer pliku z dana wartoscia

# TEMP - file number with that value

  TEMP=`cat dat.script.sort | grep "$DUMM" | cut -f 3 -d '.'`

# skrypt.7 wymaga podania numeru w nazwie plotowanego pliku

# skrypt.7 needs giving number from filename

  for MAXMIN in $TEMP
  do
    ./skrypt.7 "$MAXMIN" "$INDEX2" | gnuplot
    echo RO"$NUMBER_"$MAXMIN".r.ps
    mv ROr.ps ../qd_wykresy/RO"$NUMBER_"$MAXMIN".r.ps
  done
done
LICZNIK="$PWD"
cd ../qd_wykresy
INDEX=`ls -1 RO"$NUMBER_"*.ps`
INDEX2=$NULL
for DUMM in $INDEX
do
  pstops -b -q -pA4 "1:0U(1w,1h)" "$DUMM" "$DUMM".u
  INDEX2=`echo -e "$INDEX2\n$DUMM.u"`
done
DUMM=`echo "$INDEX2" | wc -l`
DUMM=`expr $DUMM - 1`
INDEX=`echo $INDEX2 | tail -n "$DUMM"`
psmerge -or"$NUMBER"evol.ps `echo $INDEX`
psnup -15 -q -pA4 -r -m1.0cm r"$NUMBER"evol.ps ewolucja"$NUMBER".ps
cd ..
./ps2png qd_wykresy/ewolucja"$NUMBER".ps
ps2epsi qd_wykresy/ewolucja"$NUMBER".ps
mv qd_wykresy/ewolucja"$NUMBER".png qd_wykresy/ilustracje/png/
mv ./ewolucja"$NUMBER".epsi qd_wykresy/ilustracje/eps/ewolucja"$NUMBER".eps
mv qd_wykresy/ewolucja"$NUMBER".ps qd_wykresy/ilustracje/ps
mv qd_wykresy/*.ps qd_wykresy/ps
cd "$LICZNIK"
LICZNIK="0"
INDEX="0"
INDEX2="0"
TEMP="0"
TEMP2="0"
NUMBER=`expr $NUMBER + 1 `
done

```

Dodatek 2e

Skrypty przetwarzający dane o ewolucji gęstości prawdopodobieństwa w symulacjach przejść i tworzące z nich animacje w formacie gif:

Dodatek 2e1

Skrypt skrypt.anim.1.sh (wresja 1):

```
#!/bin/sh

# (c) Kacper Kulczycki 2007

# Skrypt generujący animowane gif-y, przy pomocy programu Eddiego
# Kohlera gifsicle, z plików postscriptowych generowanych przez
# gnuplota.

# Script generates animated gifs from postscript, files generated with
# gnuplot, with Eddie's Kohler gifsicle program.

LICZNIK="1"
cd qd_1
while [ -e ../qd_"$LICZNIK" ]
do
  echo "# Praca w katalogu qd_$LICZNIK "
  cd ../qd_"$LICZNIK"

# przetwarzanie skryptu dla gnuplota produkowanego przez program qd
# processing script for gnuplot generated with qd program

  cat E.dat | grep -v pause | grep -v "set xrange" | cut -f 1 > dat.script.2
  NUMBER=`cat dat.script |grep -v pause | grep -v "set xrange" | cut -d '"' -f 2 | cut -d
  "." -f 3 | paste dat.script.2 - | tr '\t' '_'`

# tworzenie plików postscriptowych przez gnuplota (z użyciem skryptu
# skrypt.7)

# generating postscript files with gnuplot (with skrypt.7 script)

  LICZNIK2=$NULL
  for DUMM in $NUMBER
  do
    TEMP=`echo $DUMM | cut -f 2 -d "_"`
    TEMP2=`echo $DUMM | cut -f 1 -d "_"`
    ./skrypt.7 "$TEMP" "$TEMP2" | gnuplot
    echo RO"$LICZNIK_"$TEMP".r.gif
    cd ..

# konwersja *.ps do *.gif (z użyciem skryptu ps2gif)

# conversion *.ps to *.gif (with ps2gif script)

    ./ps2gif ../qd_"$LICZNIK"/ROr.ps
    mv ../qd_"$LICZNIK"/ROr.gif ../qd_wykresy/gif/RO"$LICZNIK_"$TEMP".gif
    LICZNIK2=`echo -e "$LICZNIK2\n"../qd_wykresy/gif/RO"$LICZNIK_"$TEMP".gif`
    cd ../qd_"$LICZNIK"
  done
  DUMM=`echo "$LICZNIK2" | wc -l`
  DUMM=`expr $DUMM - 1`
  NUMBER=`echo "$LICZNIK2" | tail -n "$DUMM"|tr '\n' ' '`

# tworzenie animacji

# generating animation

  gifsicle -V -d20 -k4 -l $NUMBER > ../qd_wykresy/animacje/ewolucja"$LICZNIK".gif
  echo "ewolucja"$LICZNIK".gif"
  LICZNIK=`expr $LICZNIK + 1`
done
```

Dodatek 2eII

Skrypt skrypt.anim.2.sh (wersja 2):

```
#!/bin/sh

# (c) Kacper Kulczycki 2007

# Wersja druga skryptu do generowania animowanych gif-ow z
# postscriptowych wykresow gnuplota, pozwala na tworzenie plikow
# zawierajacych wieksza ilosc klatek niz jego pierwsza wersja, niestety
# pliki sa nieznacznie wieksze od tych generowanych przez pierwsza
# wersje skryptu, a skrypt dziala wolniej.
# !! Skryptu nalezy uzywac po uprzednim wywolaniu skryptu w wersji !!
# !! pierwszej. !!

# The second version of the script generates animated gifs from
# postscript plots generated with gnuplot, it enable creation of files
# containing more frames than it was in first version, unfortunately
# files are a little bit bigger than produced by first version of
# script and script executes slower.
# !! Script should be used after executing it's first version. !!

LICZNIK="1"
cd qd_1
while [ -e ../qd_"$LICZNIK" ]
do
  echo "# Praca w katalogu qd_$LICZNIK "
  cd ../qd_"$LICZNIK"

# przetwarzanie skryptu dla gnuplota produkowanego przez program qd

# processing script for gnuplot generated with qd program

  cat E.dat | grep -v pause | grep -v "set yrange" | cut -f 1 > dat.script.2
  NUMBER=`cat dat.script |grep -v pause | grep -v "set yrange" | cut -d '"' -f 2 | cut -d
"." -f 3 | paste dat.script.2 - | tr '\t' '_'`
  LICZNIK2=$NULL
  for DUMM in $NUMBER
  do
    TEMP=`echo $DUMM | cut -f 2 -d "_"`
    TEMP2=`echo $DUMM | cut -f 1 -d "_"`
    LICZNIK2=`echo -e "$LICZNIK2\n"../qd_wykresy/gif/RO"$LICZNIK"_"$TEMP".r.gif`
  done
  DUMM=`echo "$LICZNIK2" | wc -l`
  DUMM=`expr $DUMM - 1`
  NUMBER=`echo "$LICZNIK2" | tail -n "$DUMM"|tr '\n' ' '`
  touch ../qd_wykresy/animacje/evolution"$LICZNIK".gif
  touch ../qd_wykresy/gif/ewolucja"$LICZNIK".gif

# czesc glowna - tworzenie animacji

# main part - creation of animation

  for LICZNIK2 in $NUMBER
  do
    gifsicle --conserve-memory --multifile -V -d20 -k4 -l
    ../qd_wykresy/animacje/evolution"$LICZNIK".gif $LICZNIK2 >
    ../qd_wykresy/gif/ewolucja"$LICZNIK".gif
    mv ../qd_wykresy/gif/ewolucja"$LICZNIK".gif
    ../qd_wykresy/animacje/evolution"$LICZNIK".gif
  done
  echo "evolution"$LICZNIK".gif"
  LICZNIK=`expr $LICZNIK + 1`
done
```

Dodatek 3

Skrypt shellowy uruchamiający wszystkie skrypty do obróbki danych z symulacji (skrypt.all.sh):

```
#!/bin/sh
# (c) Kacper Kulczycki 2007
# Skrypt uruchamiający wszystkie skrypty przetwarzające dane z symulacji
# Script executes all scripts processing data from simulations

echo "# >>>>skrypt_1.sh<<<<" > qd.log
echo "# >>>>skrypt_1.sh<<<<"
./skrypt_1.sh >> qd.log
echo "# >>>>skrypt_2.sh<<<<" >> qd.log
echo "# >>>>skrypt_2.sh<<<<"
./skrypt_2.sh >> qd.log
echo "# >>>>skrypt_3.sh<<<<" >> qd.log
echo "# >>>>skrypt_3.sh<<<<"
./skrypt_3.sh >> qd.log
echo "# >>>>skrypt.spect.sh<<<<" >> qd.log
echo "# >>>>skrypt.spect.sh<<<<"
./skrypt.spect.sh >> qd.log
echo "# >>>>skrypt.anim.1.sh<<<<" >> qd.log
echo "# >>>>skrypt.anim.1.sh<<<<"
./skrypt.anim.1.sh >> qd.log
# echo "# >>>>skrypt.anim.2.sh<<<<" >> qd.log
# echo "# >>>>skrypt.anim.2.sh<<<<"
# ./skrypt.anim.2.sh >> qd.log
```

Dodatek 4

Skrypty shellowe do obróbki plików wynikowych, powstałych po obróbce danych z symulacji:

Dodatek 4a

Skrypt przetwarzający pliki w formacie ps do formatu png (ps2png):

```
#!/bin/sh
# /usr/local/bin/ps2png

# skrypt skopiowany ze strony:
# http://www.karakas-online.de/myLinuxTips/ps2png.html#FTN.AEN425
# rozdzielczosc pliku wynikowego, zwiekszona wzgledem orgialu

# script taken from page:
# http://www.karakas-online.de/myLinuxTips/ps2png.html#FTN.AEN425
# resolution of output file increased in relation to orginal script

for i in "$@"
do
    i_new=`dirname $i`/`basename $i .ps`.png
    echo convert $i to $i_new
    gs -dNOPAUSE -sDEVICE=png256 -r600x600 -sOutputFile=$i_new -q -dBATCH $i
    # convert -rotate 90 $i_new $i_new
    # convert -resize 50% $i_new $i_new
done
#convert -transparent white test.png test.png
```


Dodatek 4b

Skrypt przetwarzający pliki w formacie ps do formatu gif (ps2gif):

```
#!/bin/sh
# (c) Kacper Kulczycki
# Skrypt konwertujący liste plików postscriptowych podanych jako
# argumenty wywołania, do plików formatu gif, oparty o pomysł skryptu
# ps2png ze strony:
# http://www.karakas-online.de/myLinuxTips/ps2png.html#FTN.AEN425
# Script converts postscript files given as execution parameters to
# files in gif format, based on idea of script ps2png, from page:
# http://www.karakas-online.de/myLinuxTips/ps2png.html#FTN.AEN425

while test -n "$1"
do
  for INDEX in "$1"
  do
    INDEX2=`dirname $INDEX`/`basename $INDEX .ps`.gif
    convert -rotate 90 -density 96x96 $INDEX $INDEX2
  done
  shift
done
```

Dodatek 5

Skrypty dla gnuplota, do tworzenia wykresów z danych, z symulacji:

Dodatek 5a

Skrypt określający parametry tworzonego wykresu w formacie postscript (skrypt.1):

```
set size 1.04, 1.0
set pointsize 0.5
set terminal postscript landscape enhanced color solid lw 1.0 "Helvetica" 20
```

Dodatek 5b

Skrypt tworzący wykres w formacie postscript i przywracający ustawienia domyślne (skrypt.2):

```
replot
set terminal x11
set size 1,1
```

Dodatek 5c

Skrypt tworzący wykres energii od czasu (skrypt.3):

```
set xlabel '{/Symbol t}'
set ylabel '{/Symbol e}'
plot 'E.dat' notitle with lines 1 1
load 'skrypt.1'
set output 'E.ps'
load 'skrypt.2'
```

Dodatek 5d

Skrypt tworzący wykres średniego położenia od czasu (skrypt.4):

```
set xlabel '{/Symbol t}'
set ylabel '<x>'
plot 'X.dat' notitle with lines 1 1
load 'skrypt.1'
set output 'X.ps'
load 'skrypt.2'
```

Dodatek 5e

Skrypt tworzący wykres normy od czasu (skrypt.5):

```
set xlabel '{/Symbol t}'
set ylabel '{/Symbol N}'
set yrange [0.995:1.005]
plot 'Norm.dat' notitle with lines 1 1
load 'skrypt.1'
set output 'N.ps'
load 'skrypt.2'
```

Dodatek 5f

Skrypt tworzący wykres gęstości prawdopodobieństwa od położenia (skrypt.6):

```
set xlabel 'x'
set ylabel '{/Symbol r}'
set yrange [0:2.05]
plot 'ro.dat.0' smooth cspline notitle with lines 1 1
load 'skrypt.1'
set output 'RO.ps'
load 'skrypt.2'
```

Dodatek 6

Skrypt shellowy tworzący skrypt dla gnuplota, do tworzenia wykresu gęstości prawdopodobieństwa od położenia, wpisujący w tytule czas, dla którego tworzony jest wykres. Wymaga podania numeru pliku i wartości czasu:

```
#!/bin/sh

# (c) Kacper Kulczycki 2007

# Skrypt tworzy skrypt dla gnuplota, wymaga dwóch argumentów wywołania
# numeru pliku i czasu

# Script generates script for gnuplot, it needs giving two parameters
# number from filename nad time value

echo "set xlabel 'x'"
echo "set ylabel '{/Symbol r}'"
echo "set yrange [0:4.5]"
echo "set title \"Czas $2\""
echo -n "plot 'ro.dat.'"
echo -n "$1"
echo "' notitle with lines 1 1"
echo "load 'skrypt.1'"
echo "set output 'ROr.ps'"
echo "load 'skrypt.2'"
```

Dodatek 7

Plik Makefile :

```
all: qd

qd: qd.o
    g++ -o qd qd.o

qd.o: qd.cpp
    g++ -c qd.cpp

clean:
    rm -f *.o *~

install:
    ./skrypt.sh
```

Dodatek 8

Skrypt shellowy (wywoływany przez make) tworzący drzewo katalogów, oraz kopiujący pliki egzekucyjne do katalogów (skrypt.sh):

```
#!/bin/sh
for INDEX in 1 2 3
do
mkdir "qd$INDEX"
cp qd "qd$INDEX"
done
for INDEX in 1 2 3 4 5 6 7 8 9 10 11 12 13 14
do
mkdir "qd_$INDEX"
cp qd "qd_$INDEX"
done
for INDEX in 90 92 94 98 100 102 104 106 108 110
do
mkdir "qd-$INDEX"
cp qd "qd-$INDEX"
done
mkdir qd_wykresy qd_wykresy/ilustracje qd_wykresy/ilustracje/eps
qd_wykresy/ilustracje/png qd_wykresy/ilustracje/ps qd_wykresy/animacje qd_wykresy/gif
qd_wykresy/ps
```

Ten opis, oraz animowane pliki gif, obrazujące ewolucje gęstości prawdopodobieństw, można znaleźć pod adresem:

<http://studenci.fuw.edu.pl/~kmkulcz/opisy/mat-komp/qd>

Kod programu, oraz skrypty można znaleźć pod adresem:

<http://studenci.fuw.edu.pl/~kmkulcz/programy/mat-komp/qd/>