

Oscylator Van der Polla

Bartosz Zbik

2024-04-28

Jawnie zamieniłem równanie drugiego stopnia

$$\ddot{x} = -x - \mu(x^2 - 1)\dot{x} \quad (1)$$

na układ dwóch równań stopnia pierwszego

$$\begin{cases} \dot{x} = v \\ \dot{v} = -x - \mu(x^2 - 1)v \end{cases} \quad . \quad (2)$$

Przedstawiony poniżej kod głównie służy do generowania rysunków, a cała logika rozwiązania równania znajduje się w definicji `vanderpol!` oraz `findsol`.

Rozważyłem również przypadek $\mu = 0$ nie wymieniony w poleceniu, żeby zobaczyć, że problem redukuje się do oscylatora harmonicznego.

1 Kod

```
using DifferentialEquations, Statistics
using CairoMakie
set_theme!(theme_latexfonts(), fontsize=20)

# define the Van der Pol Oscillator
# in place method
function vanderpol!(du, u, mu, t)
    # u = (x, x')
    du[1] = u[2] # x' = v
    du[2] = -mu * (u[1]^2 - 1) * u[2] - u[1]
end

# Van der Pol Oscillator
# this function is only used to make stream plots
function vanderpol(u, mu)
    # u = (x, x')
    dx = u[2] # x' = v
    dv = -mu * (u[1]^2 - 1) * u[2] - u[1]
    return dx, dv
end

tspan = (0.0, 15.0)
mu_rng = [0, 1 / 32, 1 / 16, 1 / 8, 1 / 4, 1 / 2, 1, 3 / 2, 2, 3, 5, 7, 9]
# example initial conditions [x(t), x'(0)]
u0_rng = [[1.0, 0.0], [0.5, 0.0], [2.5, 0.0], [3.0, 0.0]]

function findsol(mu, initcon)
    tspan = (0.0, 15.0)
```

```

prob = ODEProblem(vanderpol!, initcon, tspan, mu)
sol = solve(prob)
return sol
end

function xplot!(ax, mu, u0)
    sol = findsol(mu, u0)
    lines!(ax, 0 .. 15, t -> sol(t) [1])
    ax.xlabel = "time"
    ax.ylabel = "x"
    ax.title = L"\mu = \$\$(mu), ~u_0 = \$\$(u0)"
end

function phasespaceplot!(ax, mu, u0)
    sol = findsol(mu, u0)
    pt = [Point2(sol(t)...) for t in range(0, 15, length=1000)]
    lines!(ax, pt)
    ax.xlabel = L"x"
    ax.ylabel = L"\dot{x}"
    ax.title = L"\mu = \$\$(mu), ~u_0 = \$\$(u0)"
end

function visualizefield(mu)
    fig = Figure()
    ax = Axis(fig[1, 1])
    xrng = -4 .. 4
    vrng = -4 .. 4
    streamplot!(ax, p -> Point2(vanderpol(p, mu)), xrng, vrng)
    ax.xlabel = L"x"
    ax.ylabel = L"\dot{x}"
    ax.title = L"Stream Plot for $\mu = \$\$(mu)$"
    return fig
end

function visualize4x(mu, u0rng)
    fig = Figure()

    ax1 = Axis(fig[1, 1])
    ax2 = Axis(fig[1, 2])
    ax3 = Axis(fig[2, 1])
    ax4 = Axis(fig[2, 2])

    allax = [ax1, ax2, ax3, ax4]

    for i in 1:4
        xplot!(allax[i], mu, u0rng[i])
    end

    return fig
end

function visualize4phase(mu, u0rng)
    fig = Figure()

    ax1 = Axis(fig[1, 1])
    ax2 = Axis(fig[1, 2])
    ax3 = Axis(fig[2, 1])
    ax4 = Axis(fig[2, 2])
    allax = [ax1, ax2, ax3, ax4]

    for i in 1:4
        phasespaceplot!(allax[i], mu, u0rng[i])
    end

    return fig
end

# This loop makes all plots and saves them
for i in eachindex(mu_rng)
    save("out/phase_$(lpad(i, 2, '0')).pdf", visualize4phase(mu_rng[i], u0_rng))
    save("out/xfromt_$(lpad(i, 2, '0')).pdf", visualize4x(mu_rng[i], u0_rng))
    save("out/stream_$(lpad(i, 2, '0')).png", visualizefield(mu_rng[i]))
end

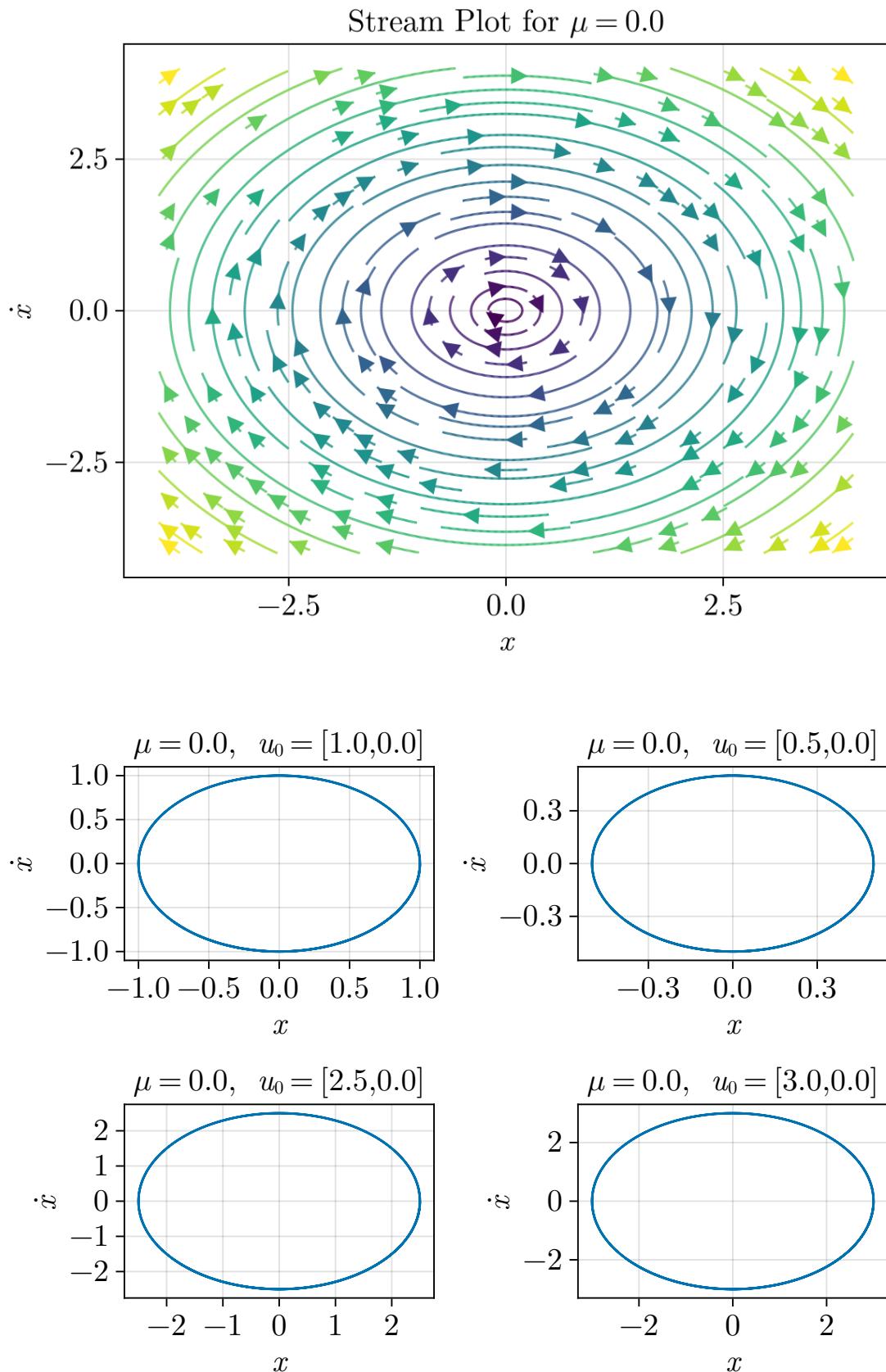
# generates LaTeX code which includes all graphics
open("putplots.tex", "w") do io
    for i in eachindex(mu_rng)
        println(io, "\n%\n% plots for mu = $(mu_rng[i])\n%%%%%%%%%%%%%%")
    end

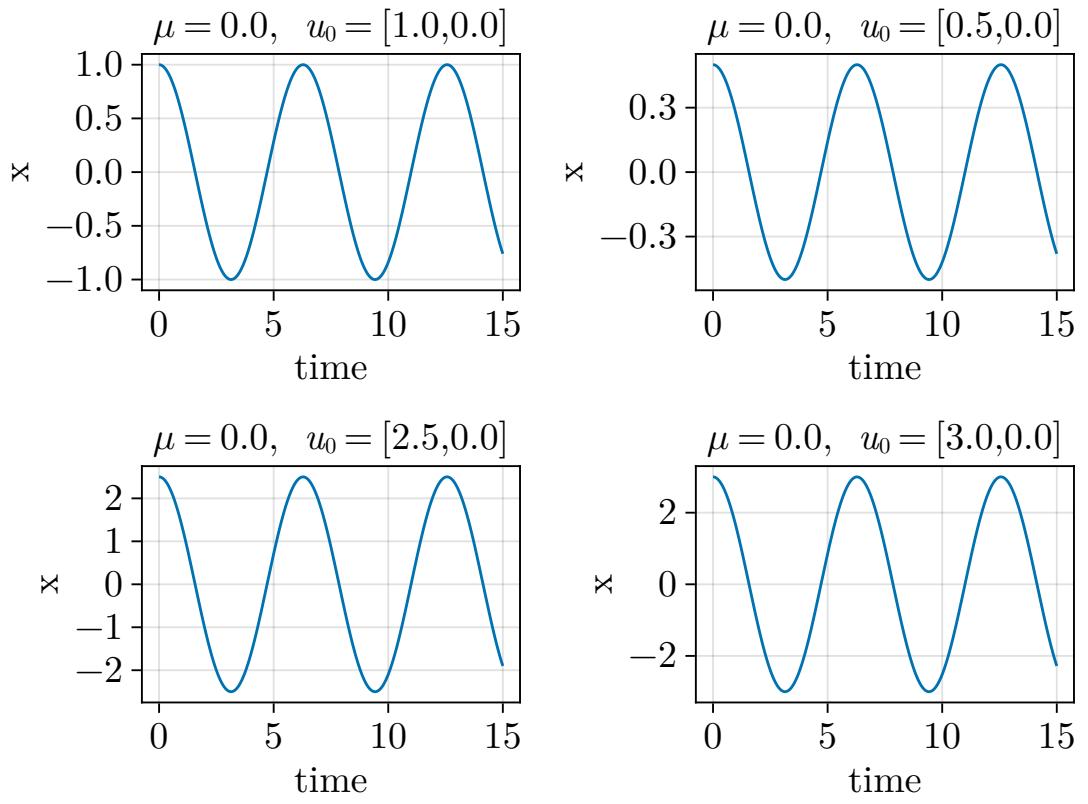
```

```
    println(io, "\\\subsection{Przypadek \$\\mu = \$(mu_rng[i])\$}")
    println(io, "\\\includegraphics[width=\\textwidth]{out/stream_\$(lpad(i,
        2, '0')).png}\\n")
    println(io, "\\\includegraphics[width=\\textwidth]{out/phase_\$(lpad(i,
        2, '0')).pdf}\\n")
    println(io, "\\\includegraphics[width=\\textwidth]{out/xfromt_\$(lpad(i,
        2, '0')).pdf}\\n")
    println(io, "\\clearpage")
end
end
```

2 Wizualizacja wyników

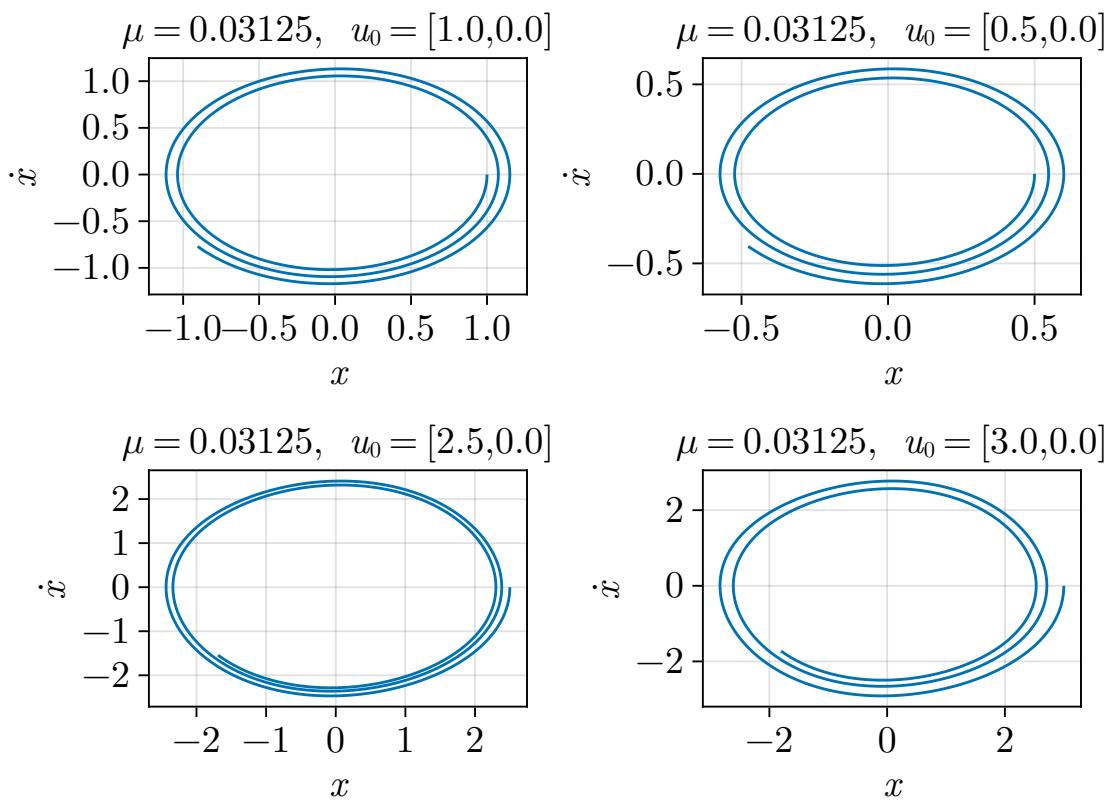
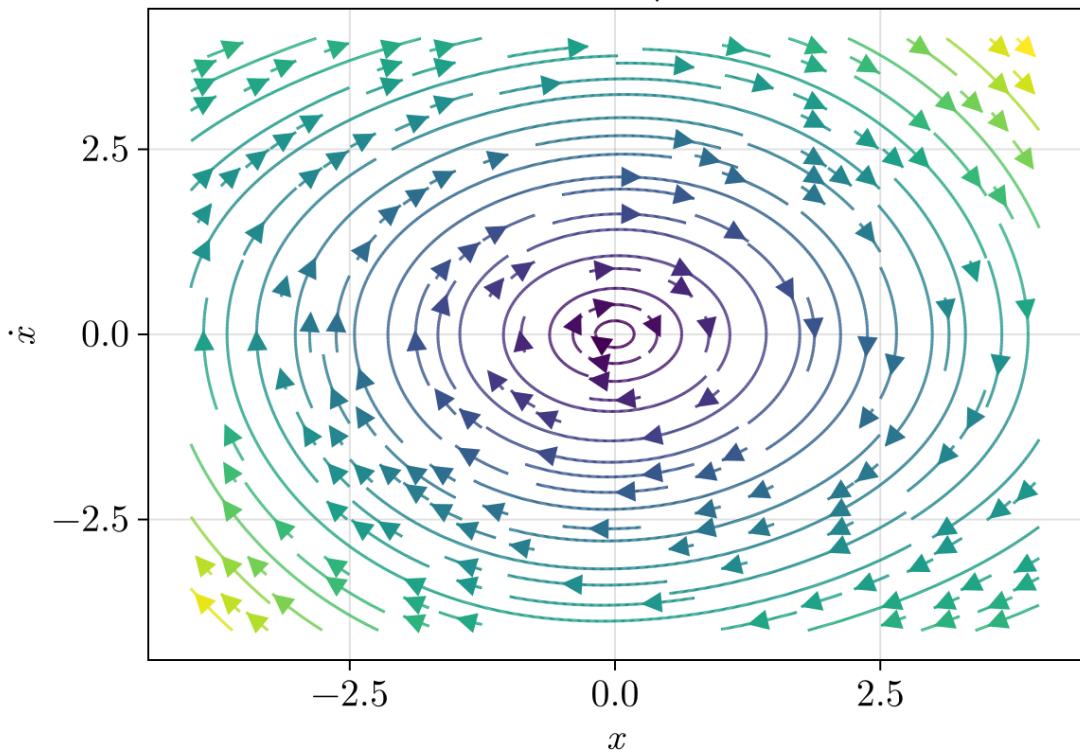
2.1 Przypadek $\mu = 0.0$

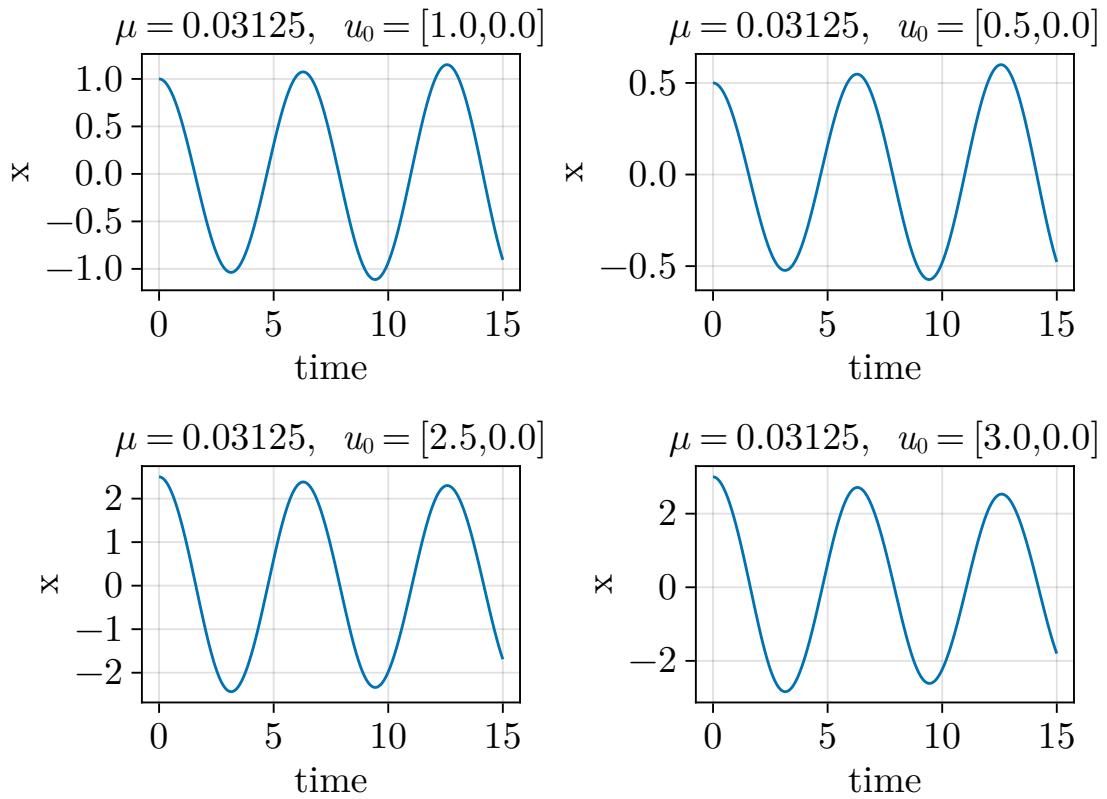




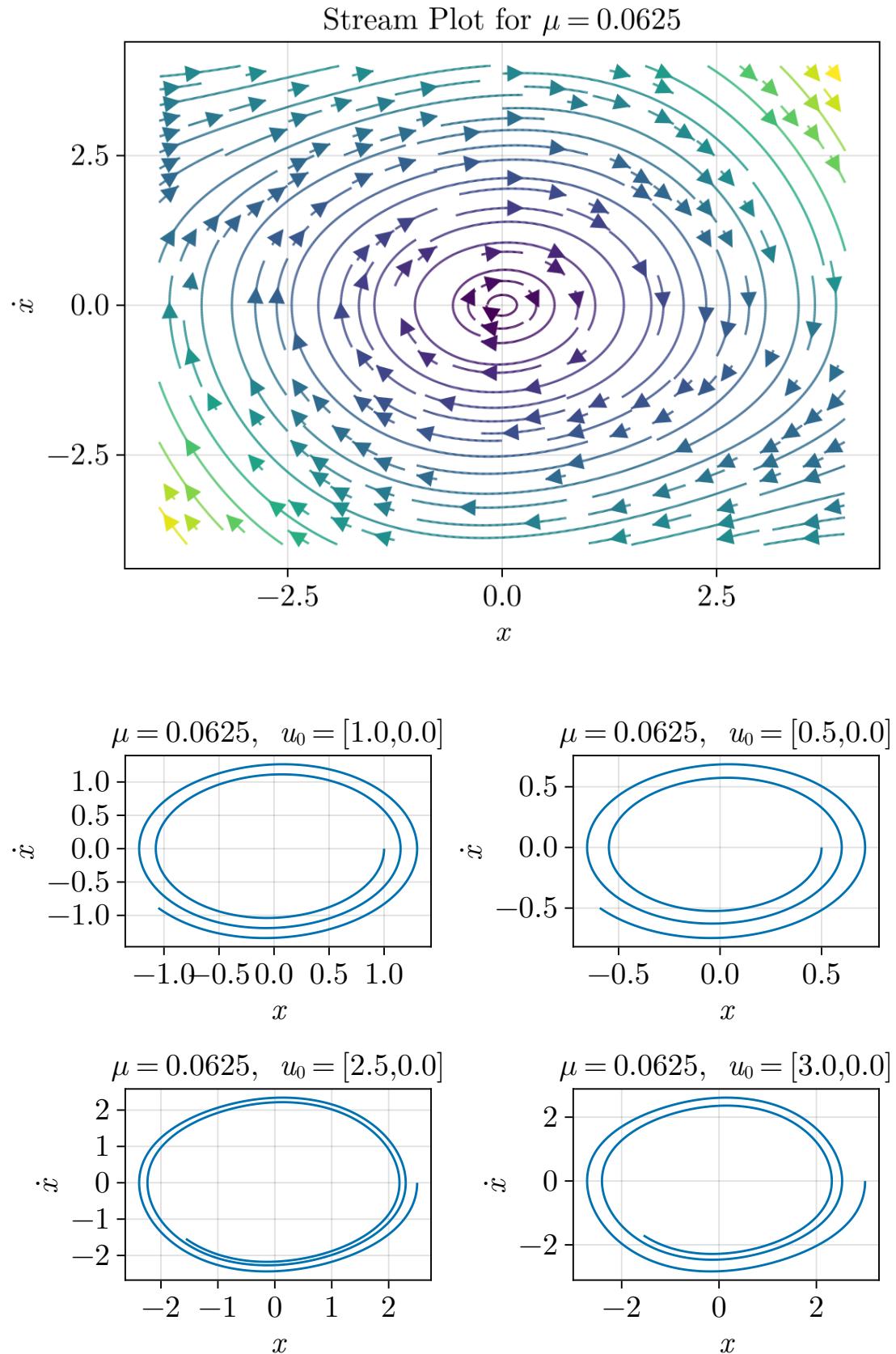
2.2 Przypadek $\mu = 0.03125$

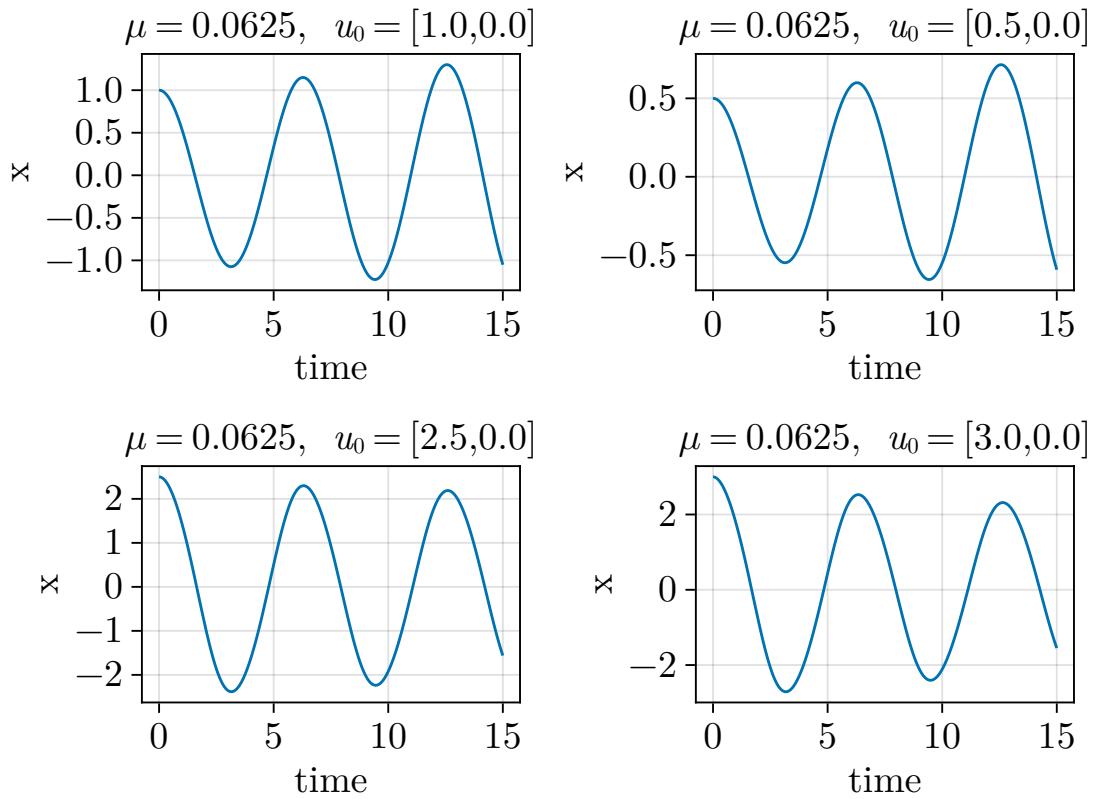
Stream Plot for $\mu = 0.03125$



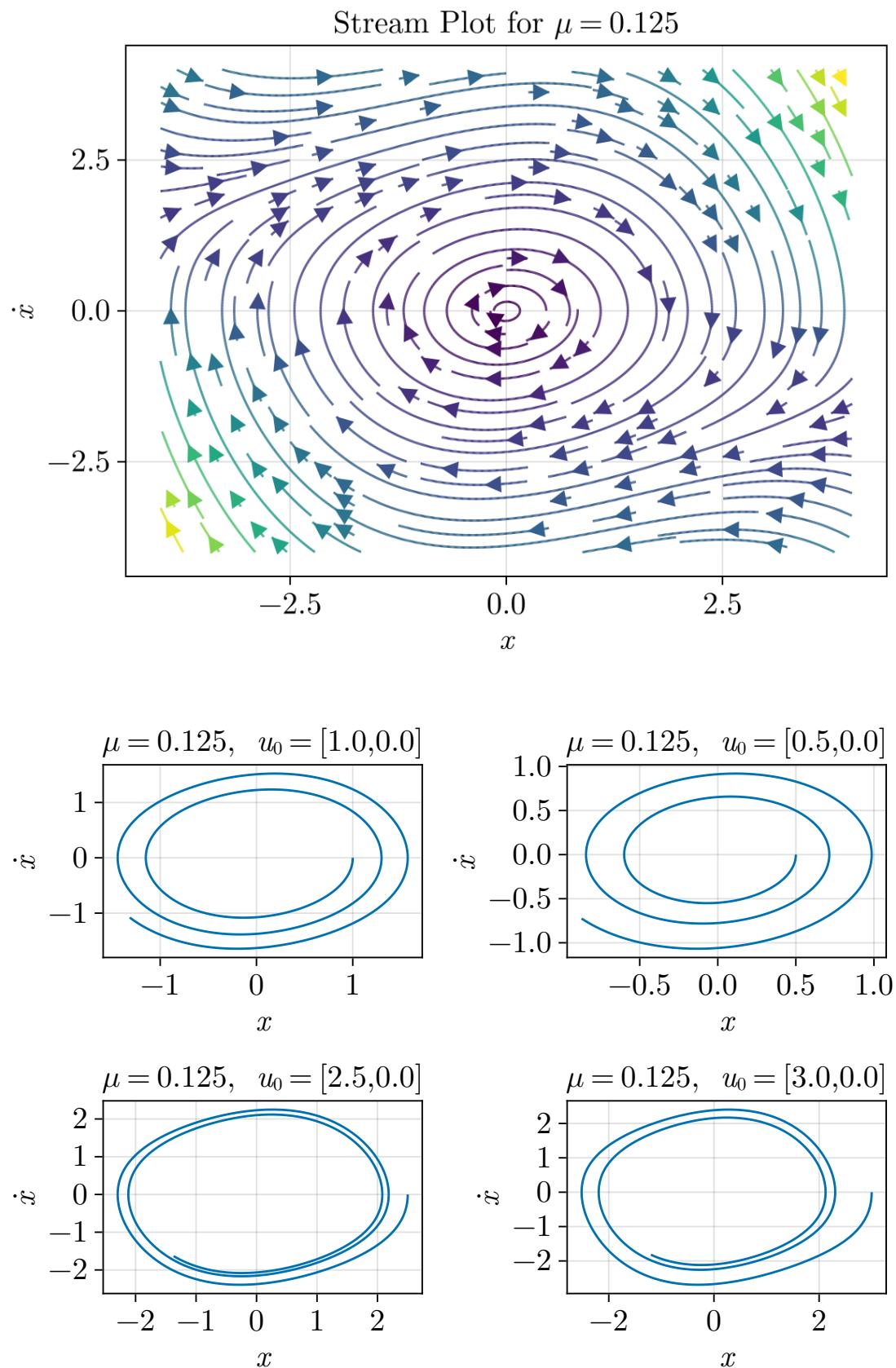


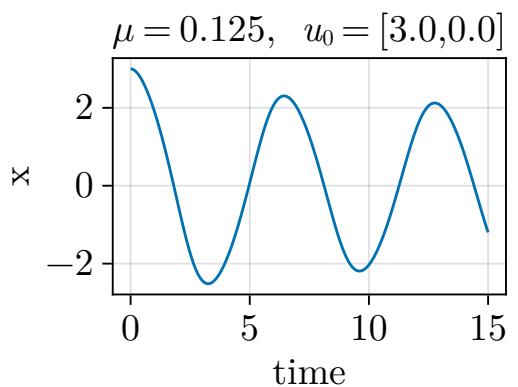
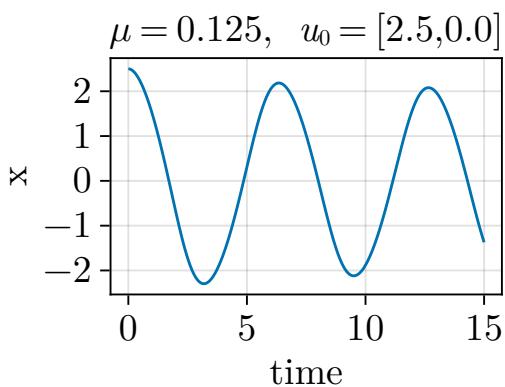
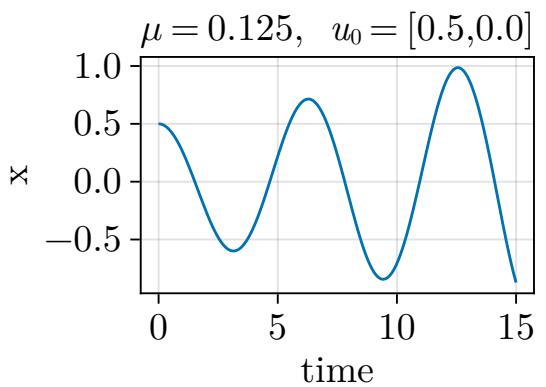
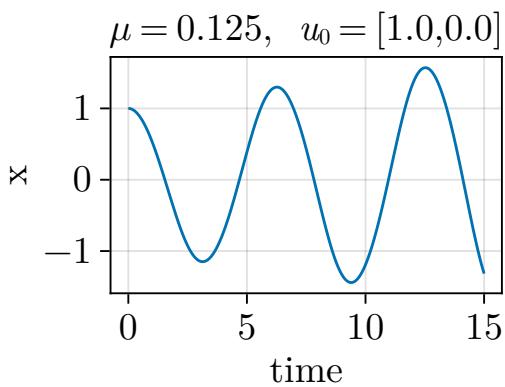
2.3 Przypadek $\mu = 0.0625$



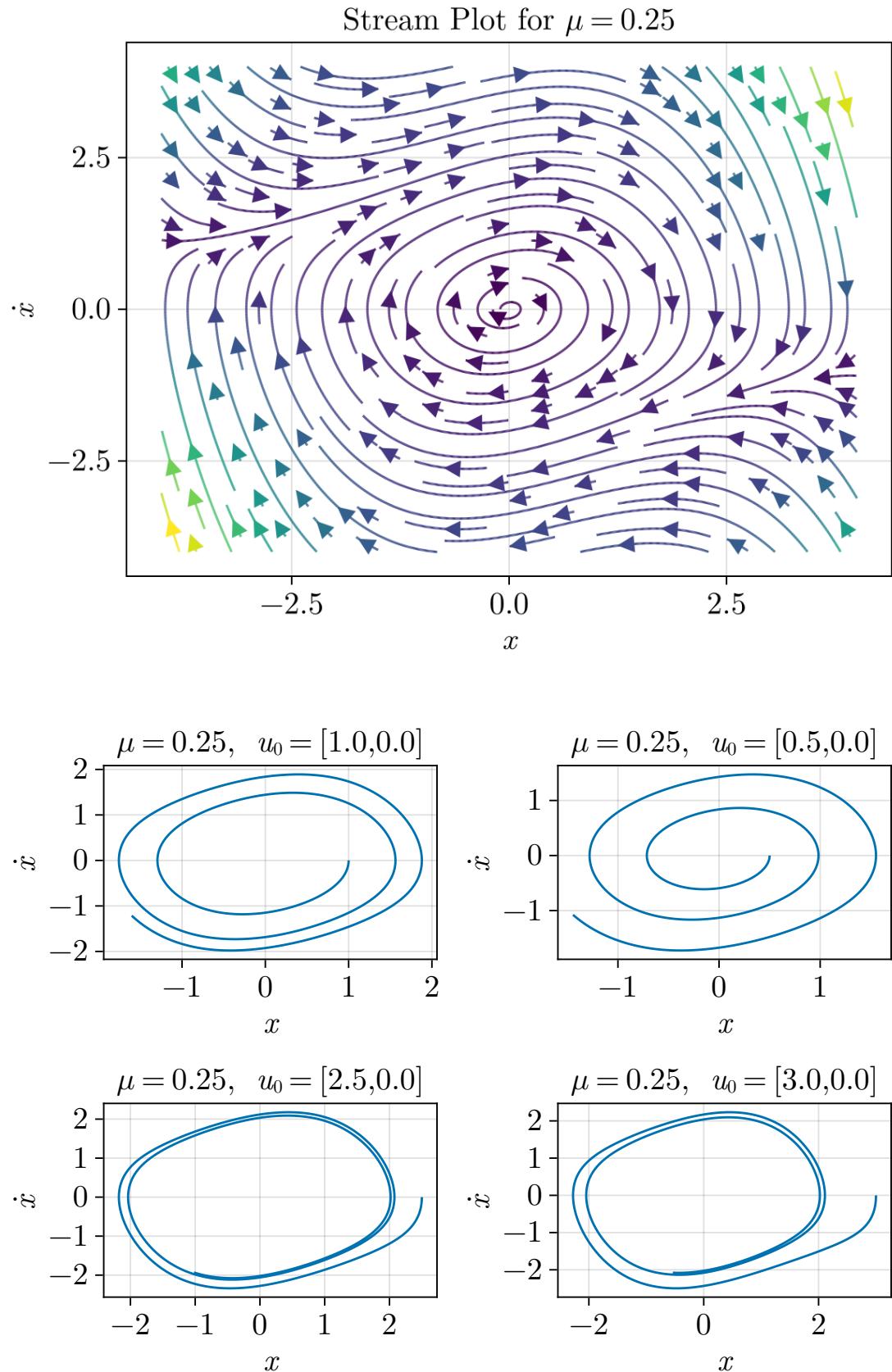


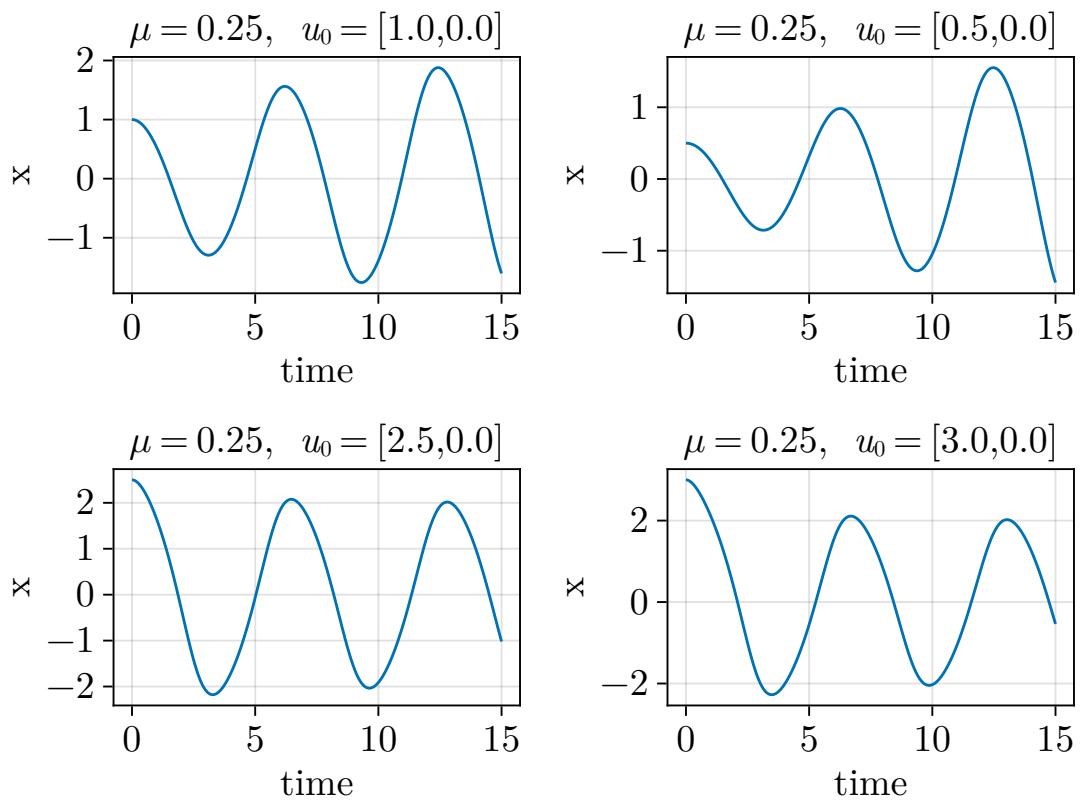
2.4 Przypadek $\mu = 0.125$



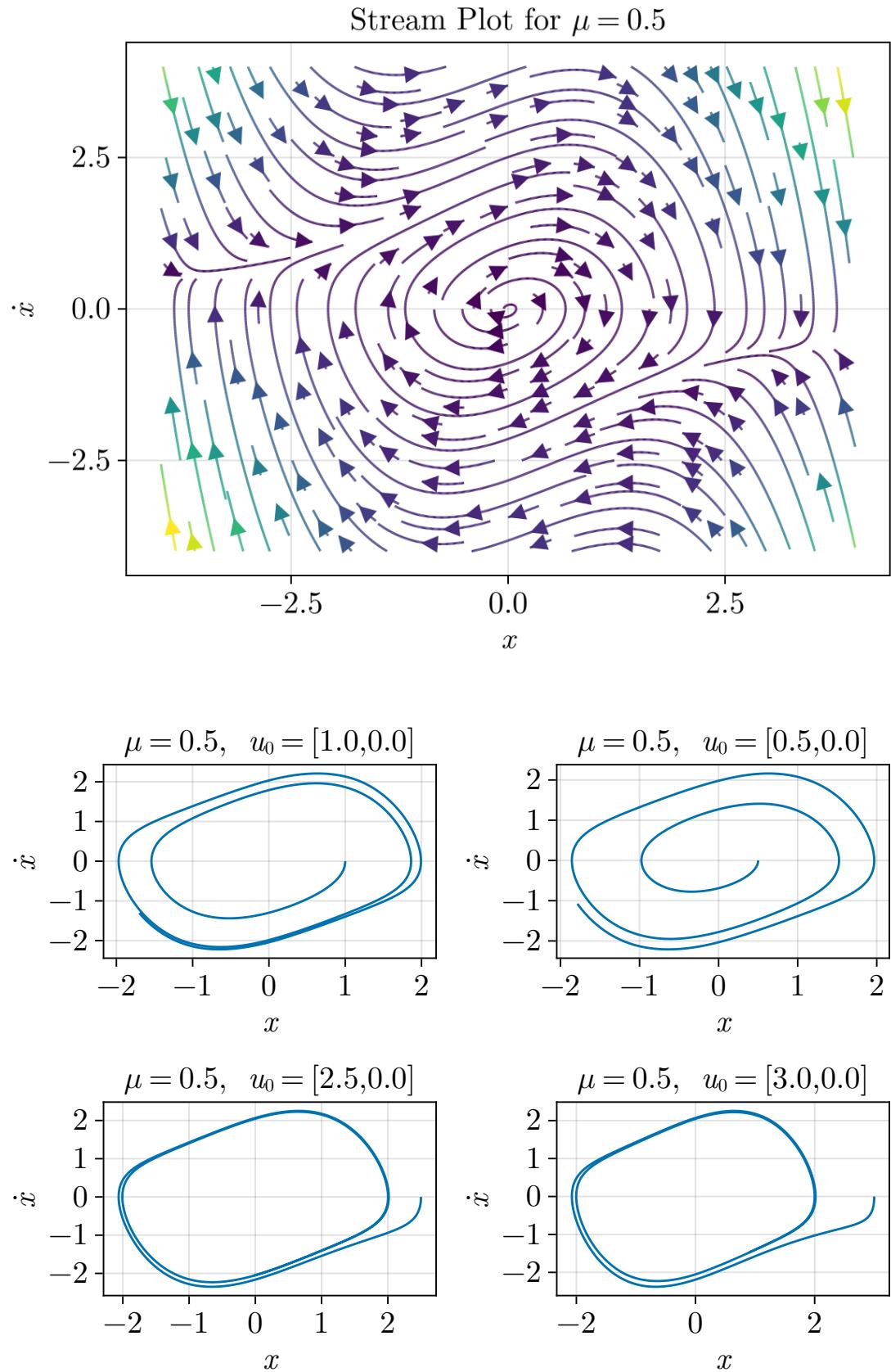


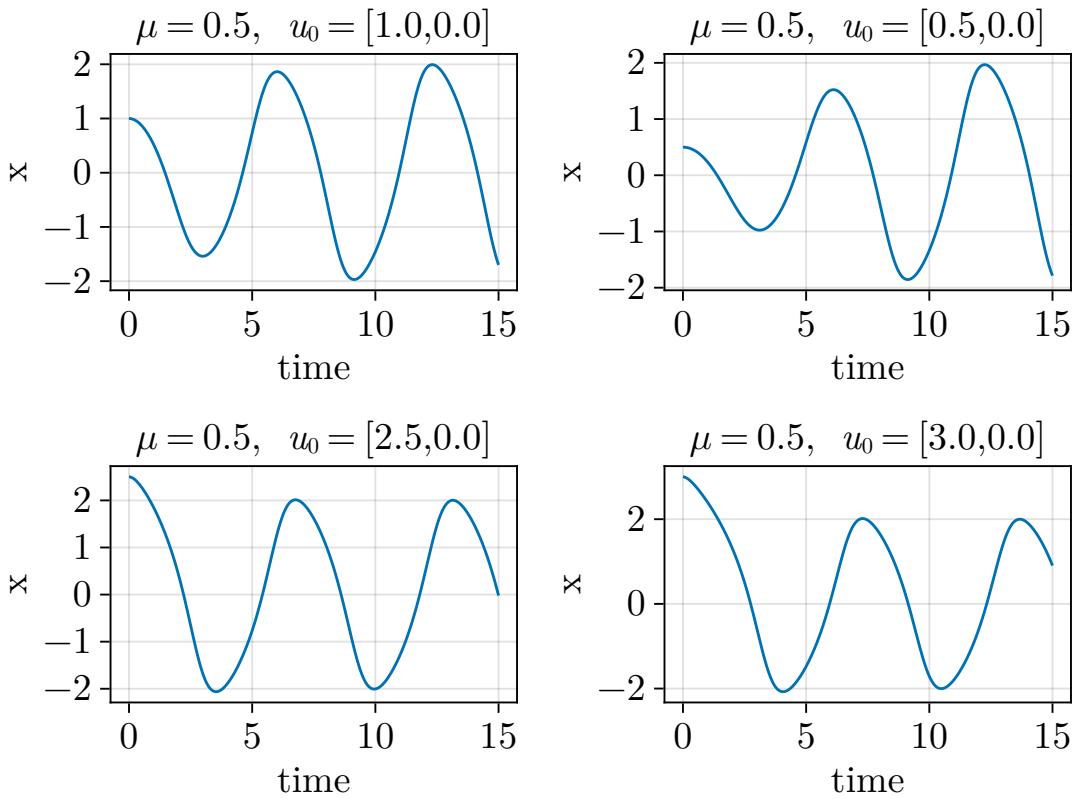
2.5 Przypadek $\mu = 0.25$



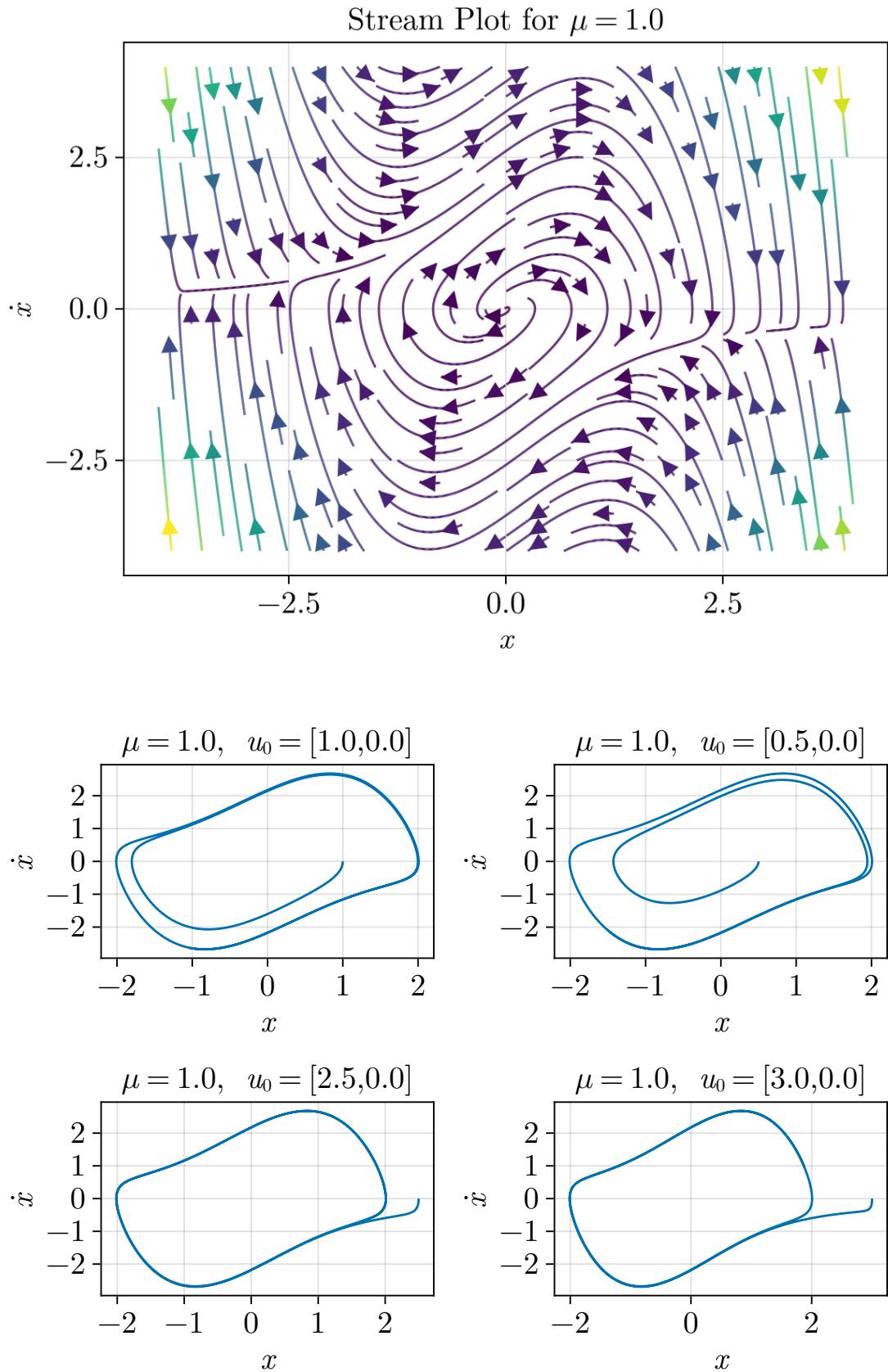


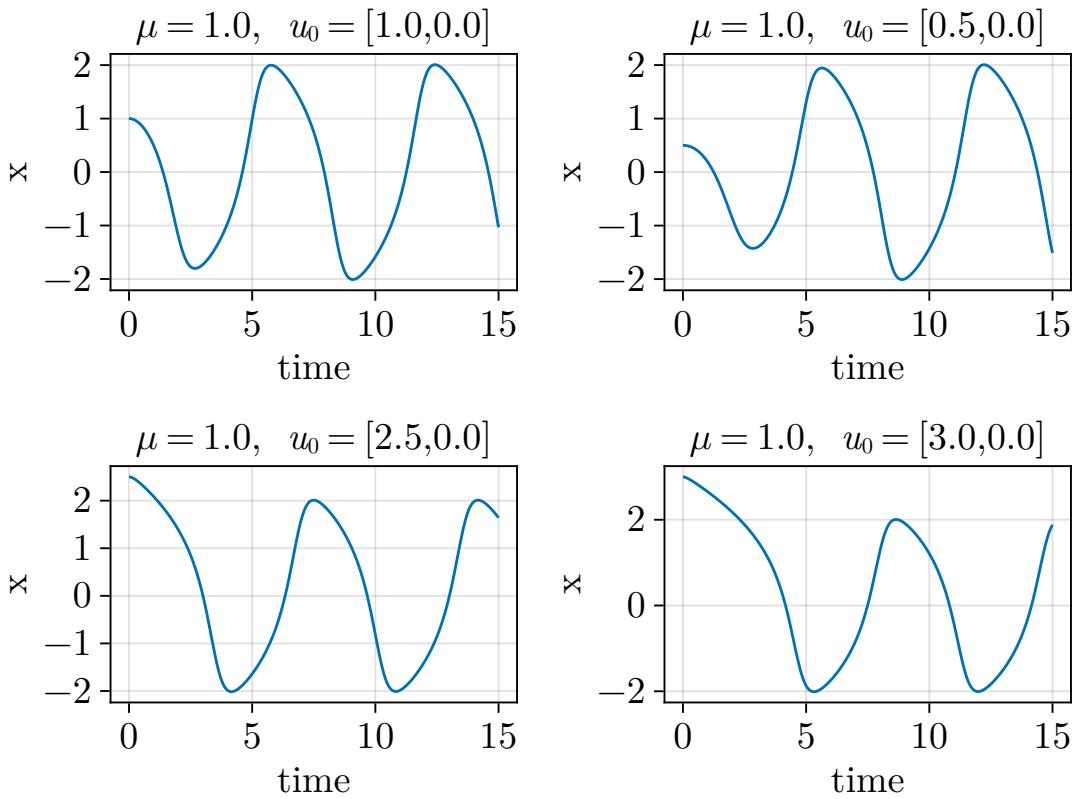
2.6 Przypadek $\mu = 0.5$



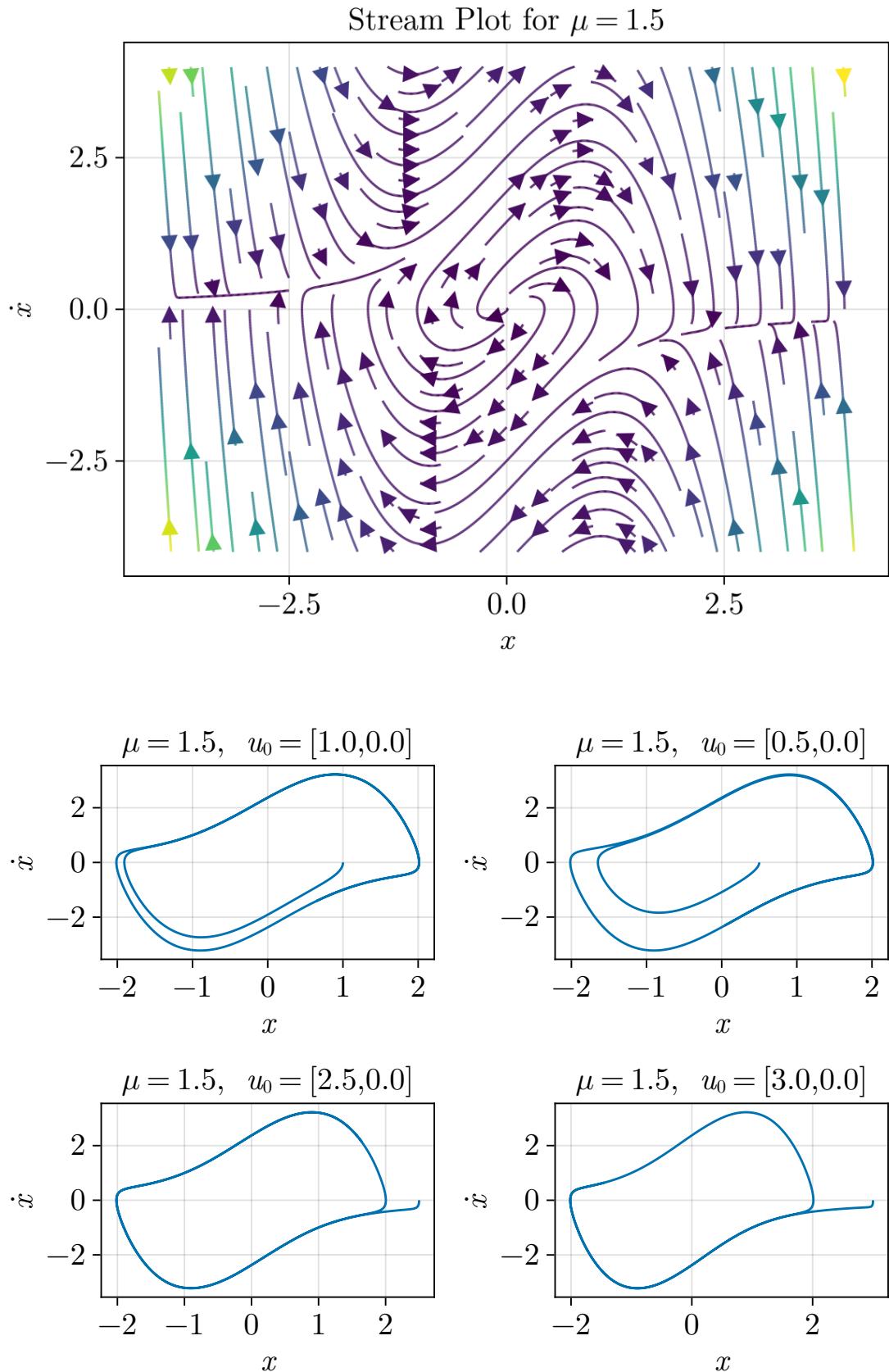


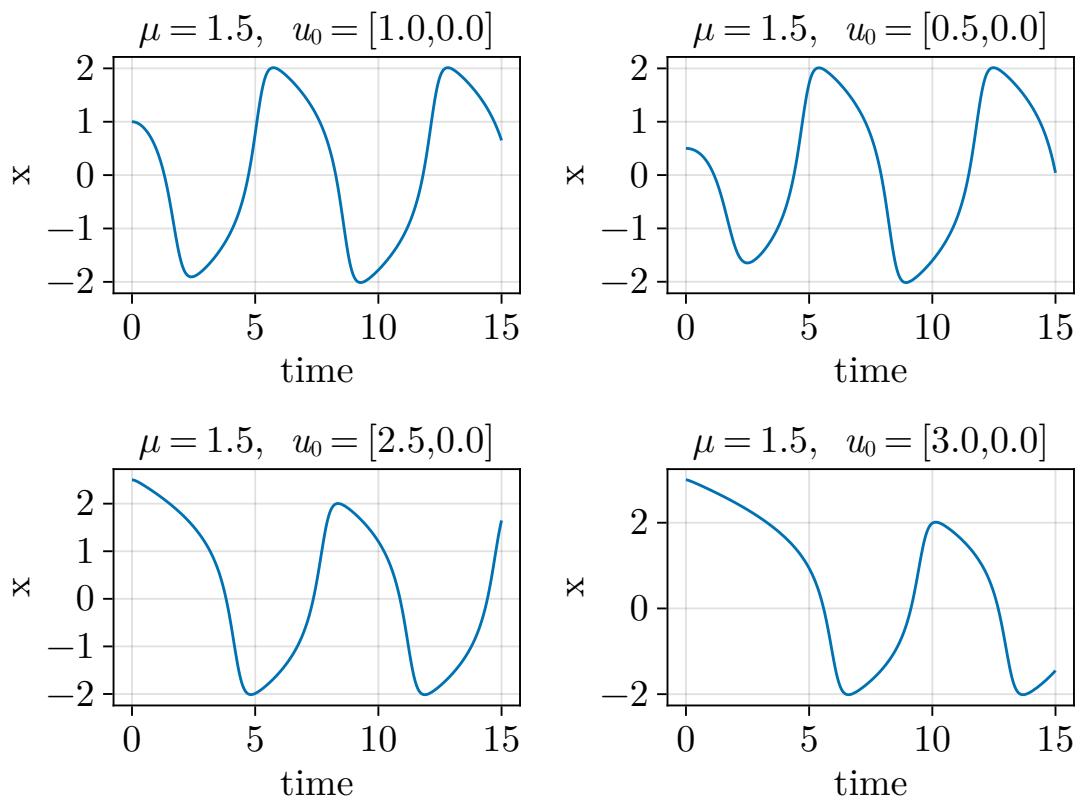
2.7 Przypadek $\mu = 1.0$



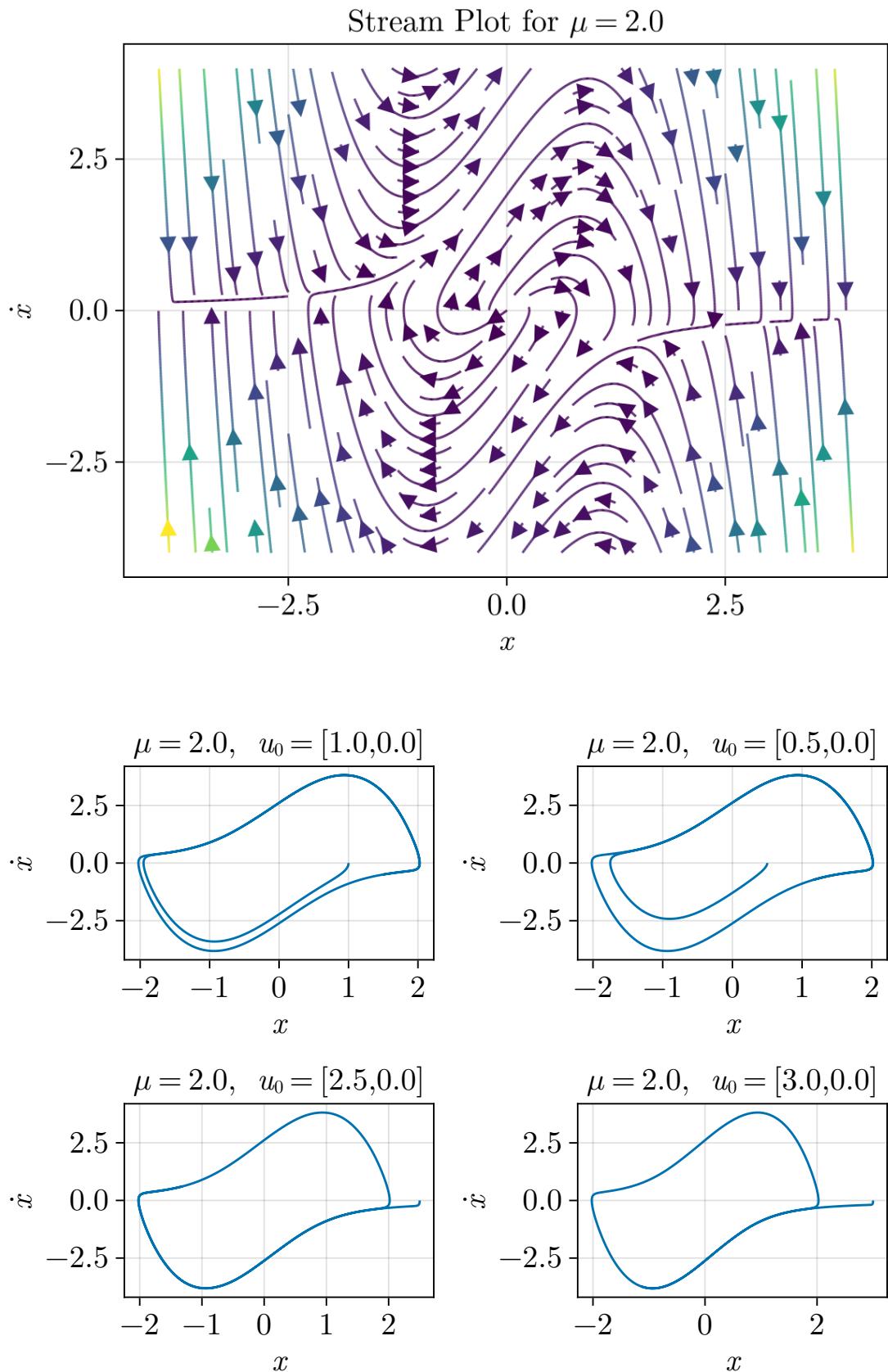


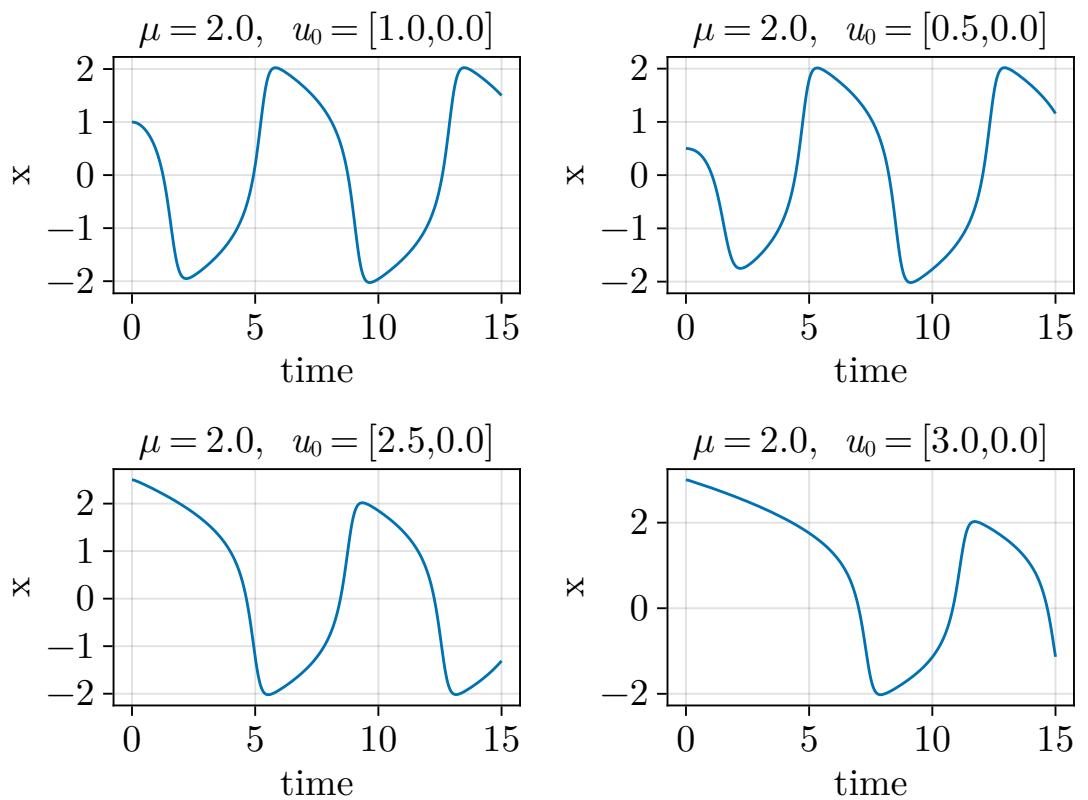
2.8 Przypadek $\mu = 1.5$



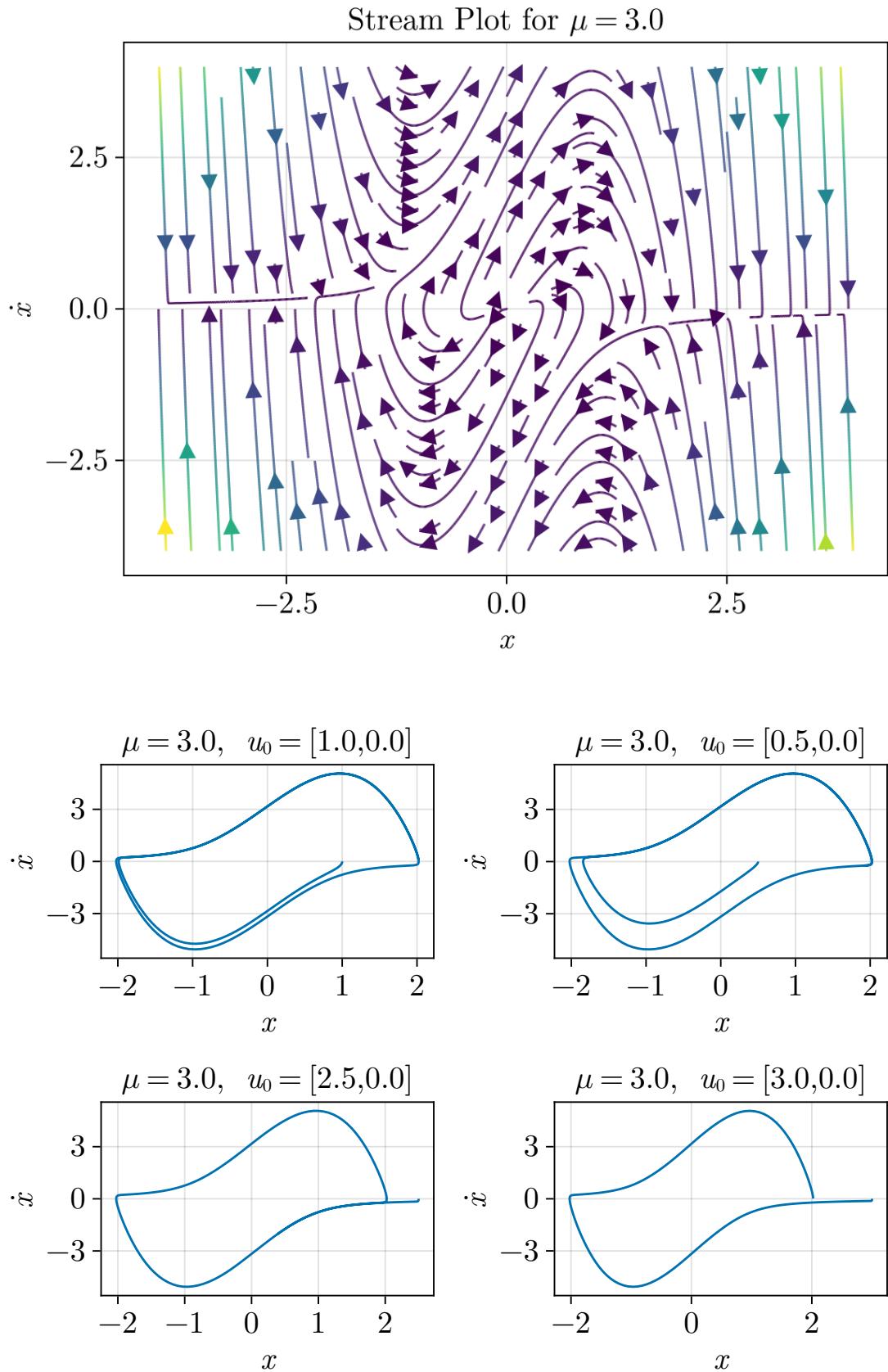


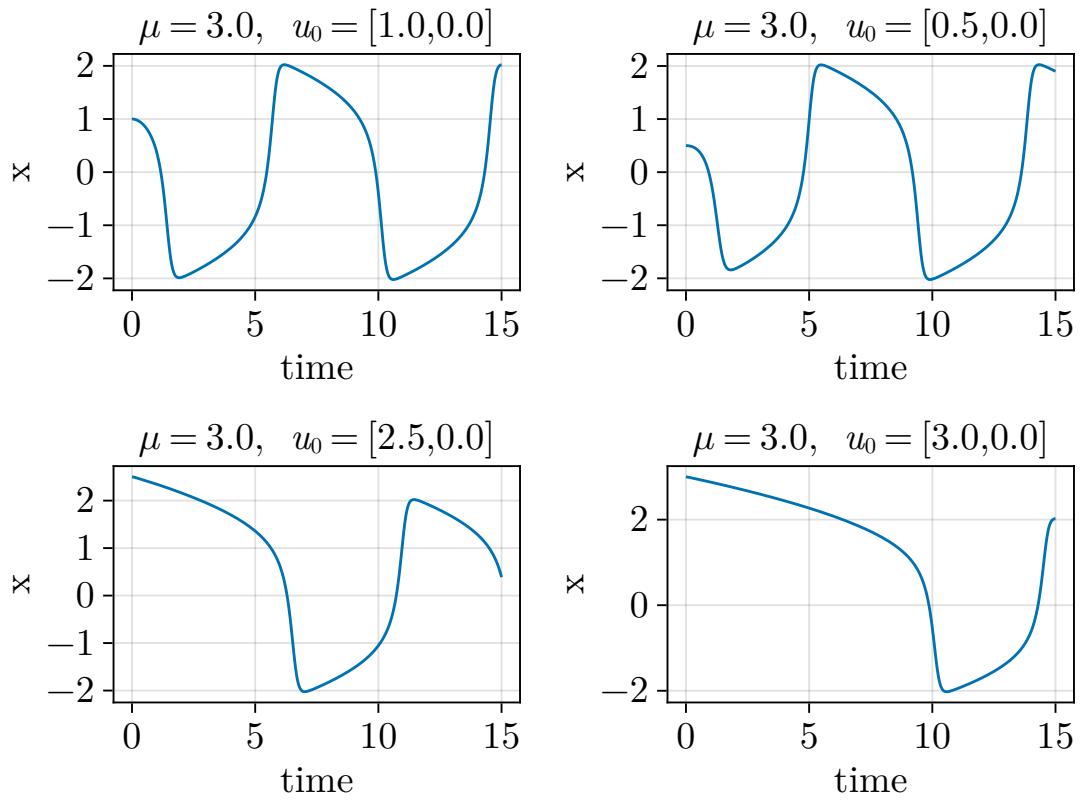
2.9 Przypadek $\mu = 2.0$



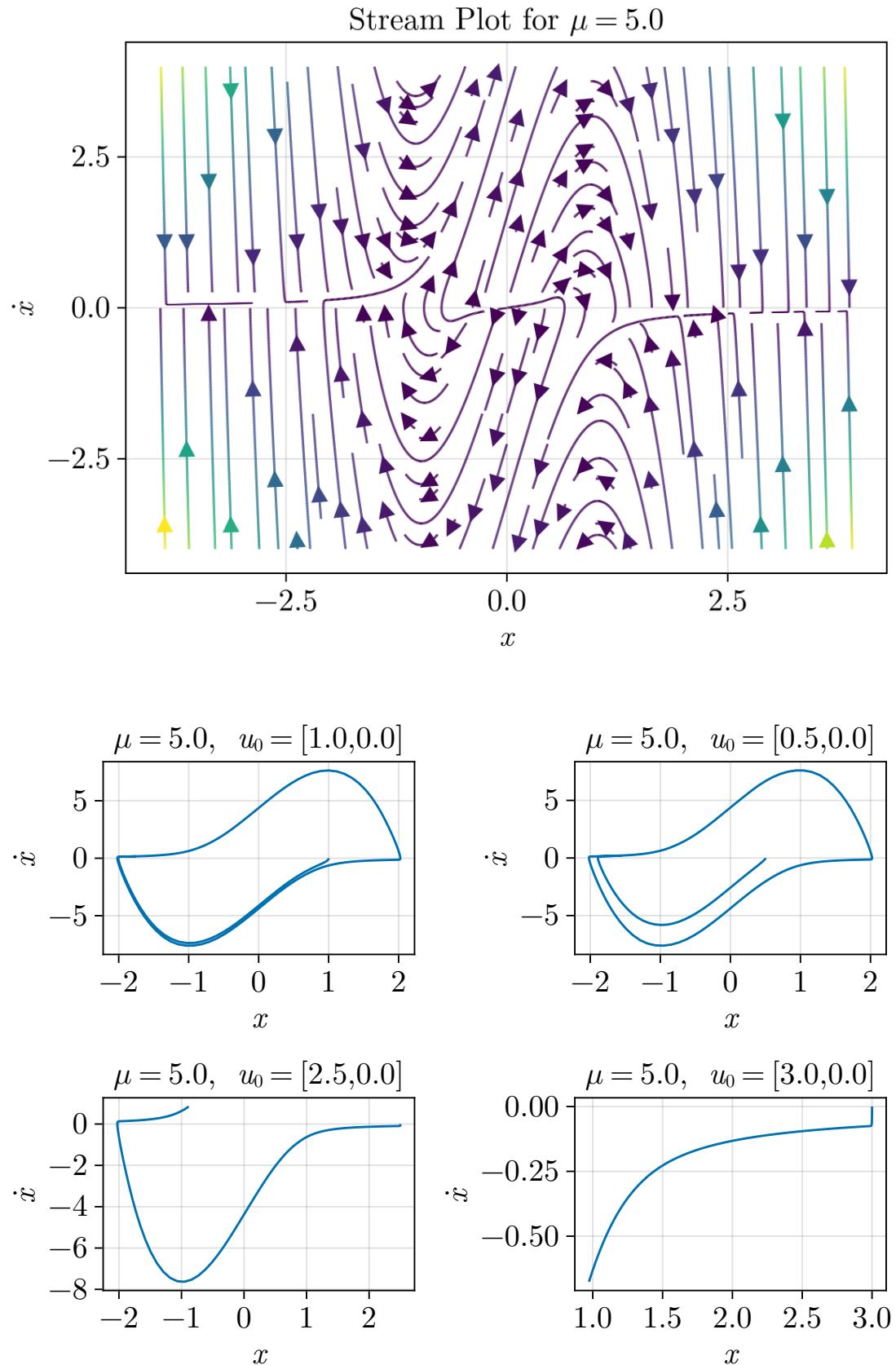


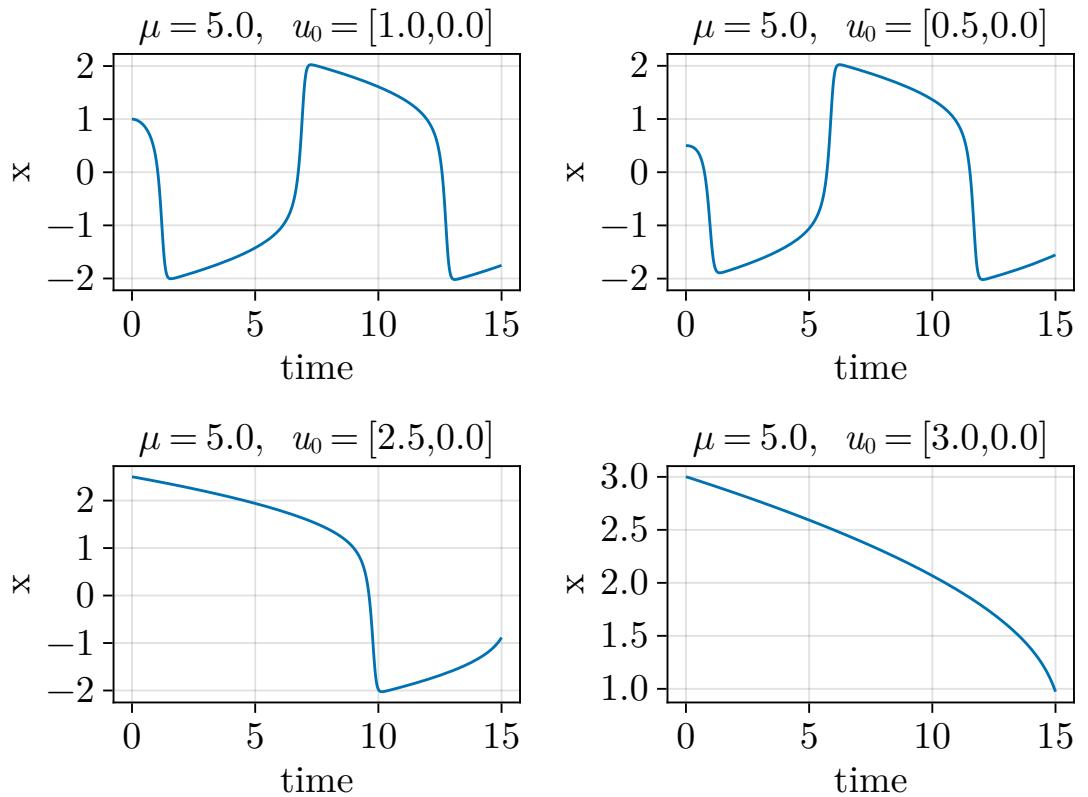
2.10 Przypadek $\mu = 3.0$



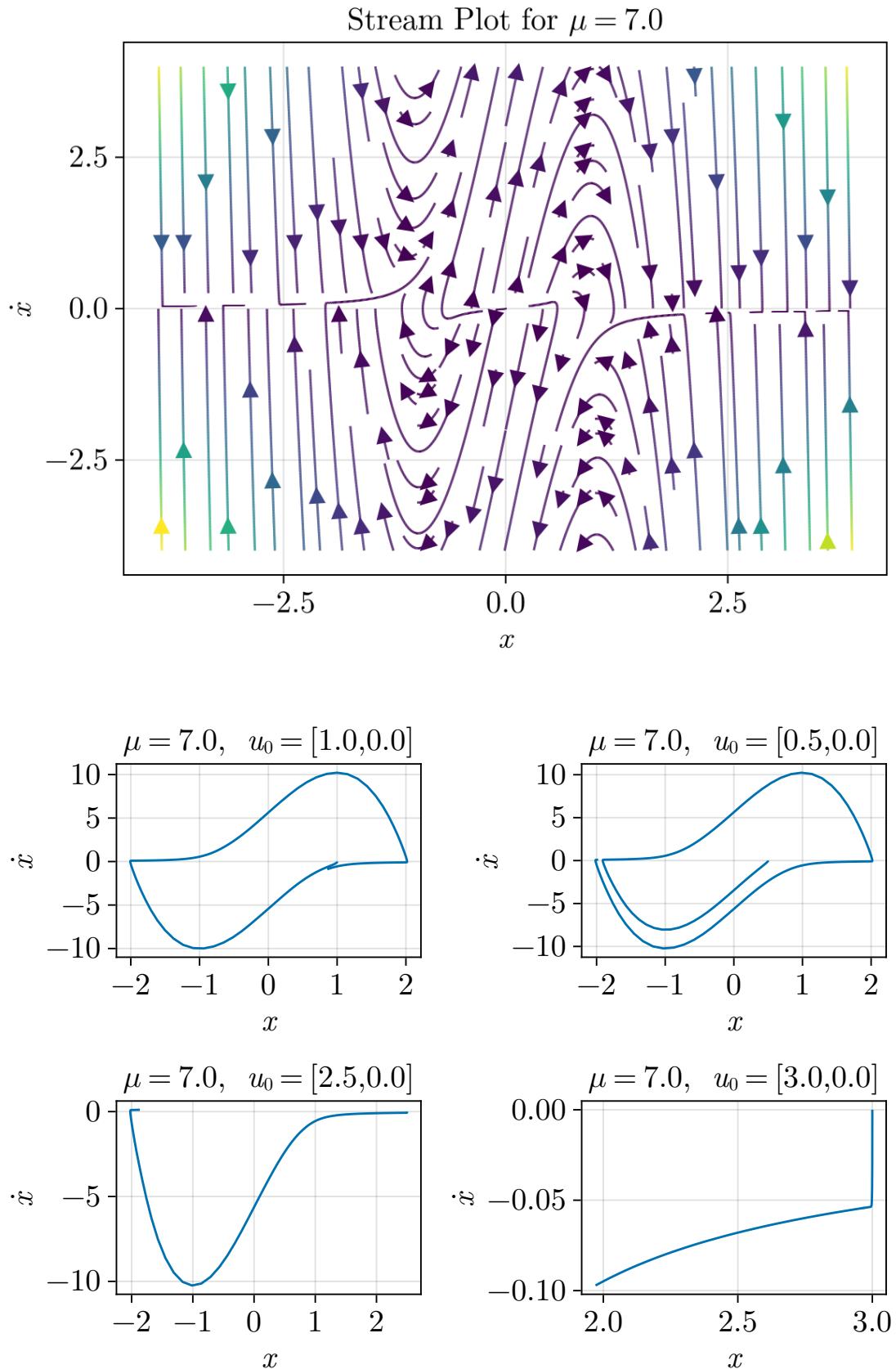


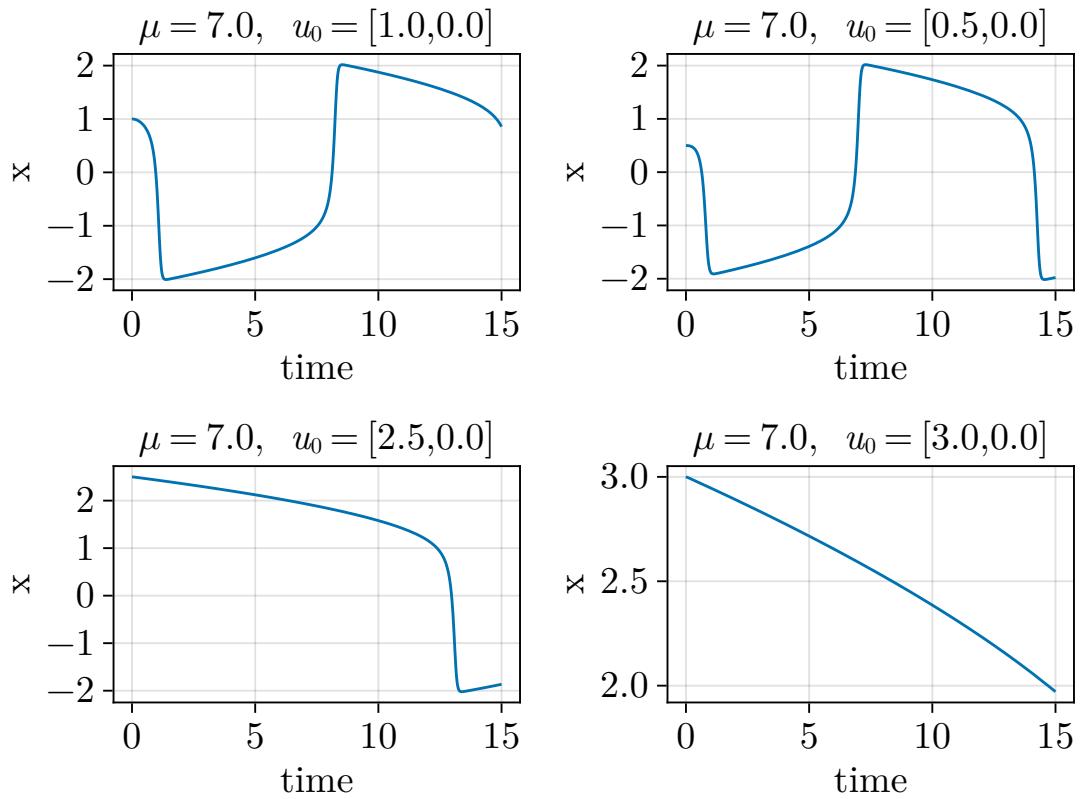
2.11 Przypadek $\mu = 5.0$





2.12 Przypadek $\mu = 7.0$





2.13 Przypadek $\mu = 9.0$

