Bartosz Włoch 249473

Patrycja Zdradzisz 249428

Sprawozdanie 3

Wieloetapowe Procesy Decyzyjne

1. Wstęp

Laboratorium dotyczyło funkcji lapunowa oraz badania stabilności układu za jej pomocą.

2. Co oznacza, że punkt równowagi jest stabilny? Co oznacza, że punkt równowagi jest stabilny asymptotycznie?

Na podstawie informacji przedstawionych podczas laboratorium wynika, że punkt równowagi jest stabilny gdy trajektorie dla danych punków początkowych nie wychodzą poza obszar okręgów o średnicy ε oraz o średnicy δ. Jeśli punkt równowagi jest stabilny asymptotycznie oznacza że trajektoria jego punktu początkowego zmierza do punktu początkowego Xe.

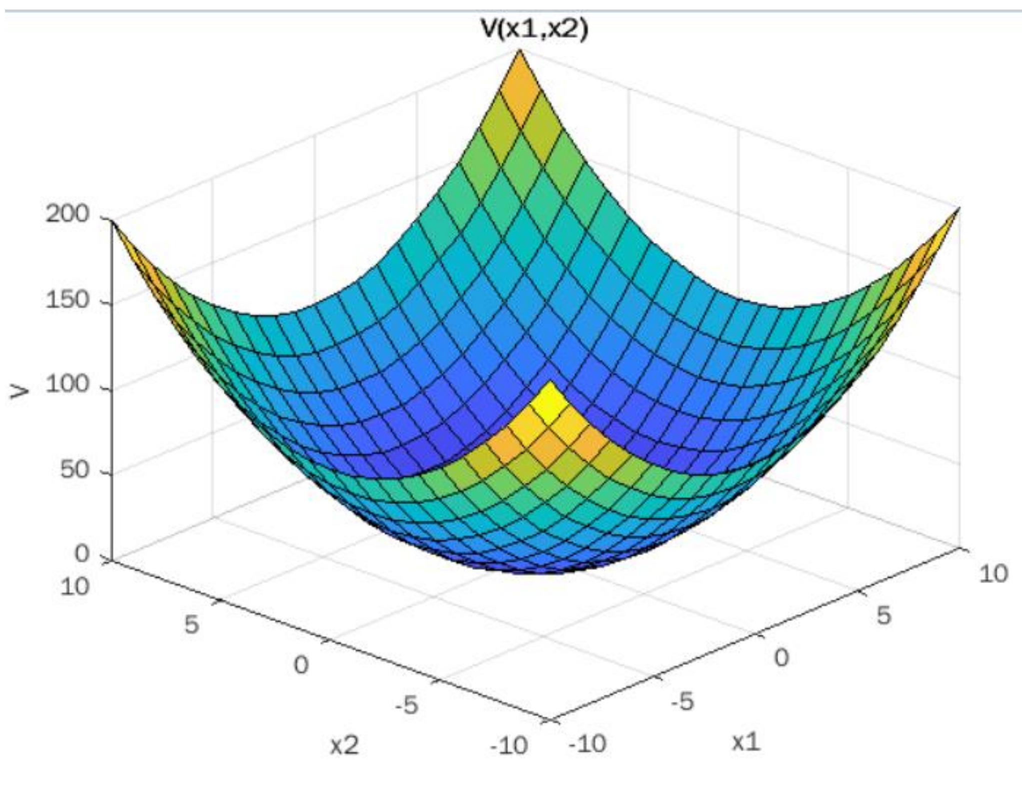
3. Podstawowe własności funkcji Lapunowa

```
[x,y]=meshgrid([-10:0.1:10],[-10:0.1:10]);
V=x.^2+y.^2;

surf(x,y,V)
shading interp;
```

1 kod

(Na rysunku 2 x=x1 y=x2)



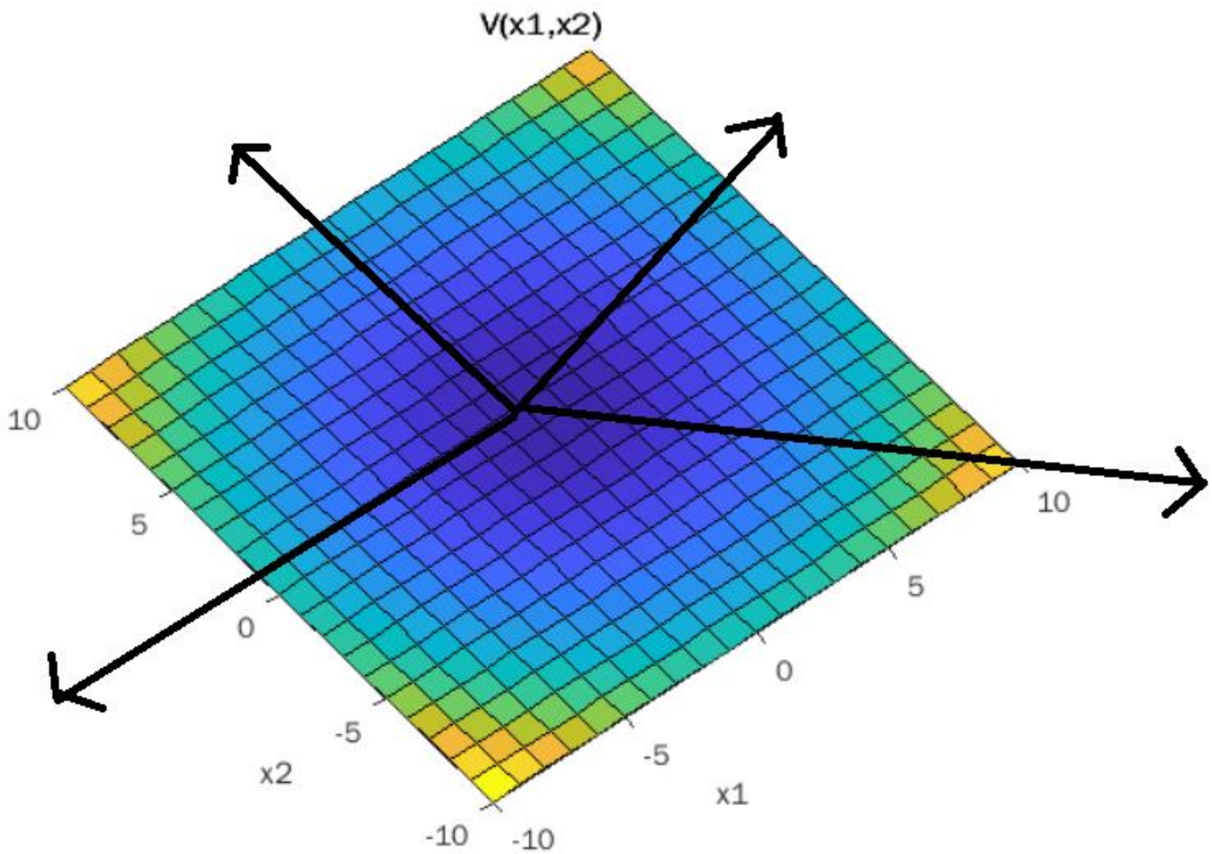
2 funkcja

Funkcja jest skalarna ponieważ w wyniku otrzymujemy konkretną liczbę, dodatkowo funkcja jest też wypukła i nieujemna co widać na wizualizacji gdzie można zaobserwować ramiona skierowane do góry oraz wszystkie wartości są większe od zera.

$$\frac{\partial}{\partial x_1}({x_1}^2 + {x_2}^2) = 2x_1$$

$$\frac{\partial}{\partial x_2}({x_1}^2 + {x_2}^2) = 2x_2$$

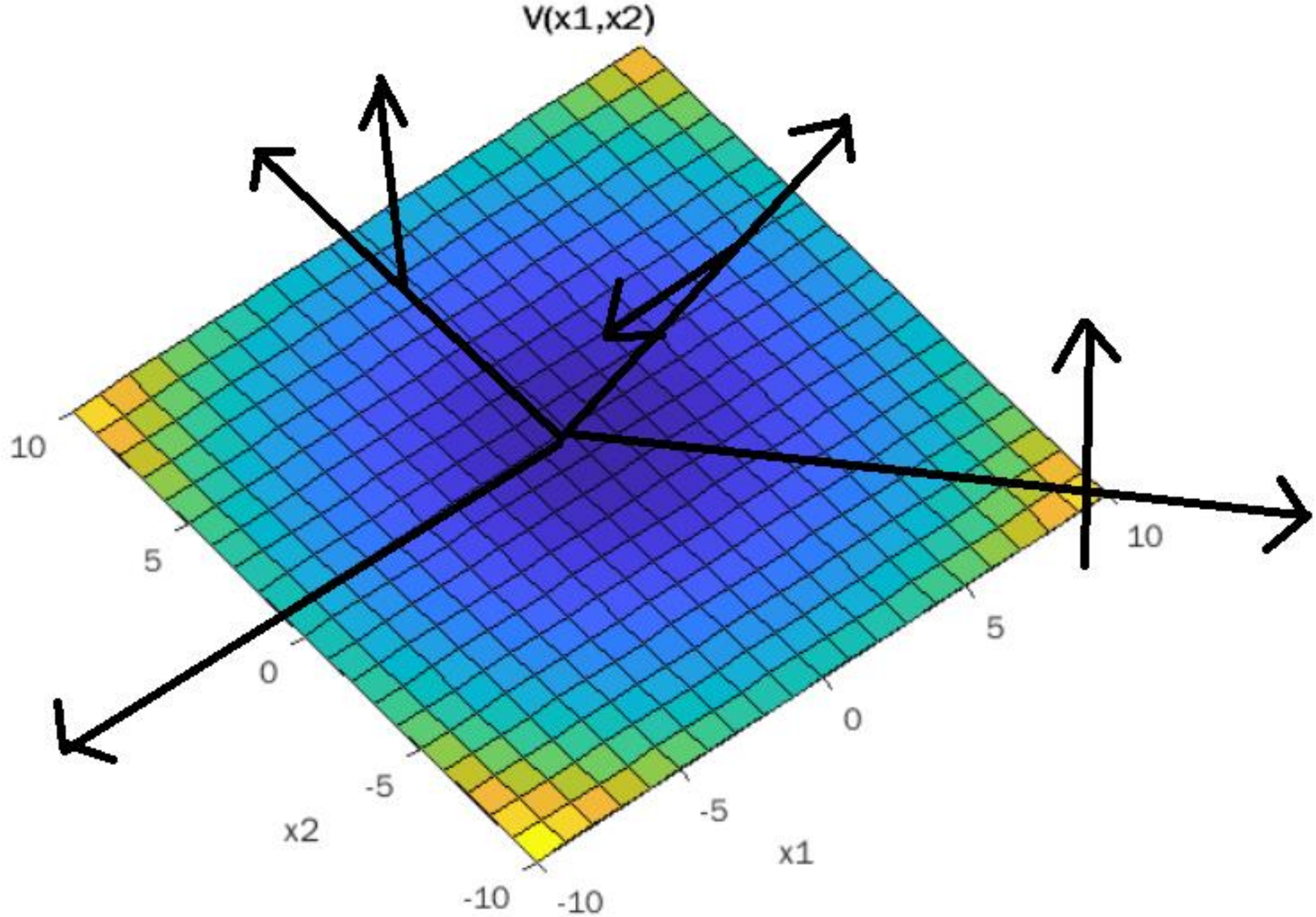
Na podstawie wyliczonych pochodnych cząstkowych można zauważyć oraz że funkcja posiada ciągłą pochodną



3 funkcja

Na otrzymanym wykresie dorysowano kilka przykładowych kierunków zmian funkcji, czyli wektory $\nabla V(x_1, x_2)$

4. Orientacja zmian funkcji Laputowa $\nabla V(x_1, x_2)$ względem funkcji $f(x)$



4 funkcja

Na rysunku funkcji Lapunowa $\nabla V(x_1, x_2)$ dorysowano klika możliwych położeń funkcji f(x) w tym skierowaną na zewnątrz, do środka oraz styczną do V. Strzałka skierowana do środka odpowiada za stabilność układu.

5. Stabilność w sensie Lapunowa

```
function stab = Stable(V,V_der, x)

    assume(x, 'real');
    assumeAlso(x ~= 0);
    V_pom = subs(V,x,0);
    lap = (V_pom == 0) && isAlways(V > 0);
    V_der0 = isAlways(V_der ~= 0);

    assume(x, 'real');
    V_der1 = isAlways(V_der <= 0);
    V_der2 = isAlways(V_der < 0);

    if lap && V_der2
        stab = 'asymptotically stable';
    elseif lap && V_der1 && V_der0
        stab = 'asymptotically stable';
    elseif lap && V_der1
        stab = 'stable';
    else
        stab = 'not stable';
    end

end
```

5 kod

W kodzie widać ze zostały rozparzone 4 przypadki

Przypadek 1- stabilny

$V(0) = 0$ i $V(x) > 0$ dla $x = 0$ oraz $\dot{V}(x) \leq 0$

Przypadek 2 – stabilny asymptotycznie

(2) $\dot{V}(x) < 0$,

Przypadek 3 – stabilny asymptotycznie

(2) $\dot{V}(x) \leq 0$ oraz $\dot{V}(x) = 0$ tylko dla $x = x^*$,

Przypadek 4 – niestabilny

Jeśli nie spełnia żadnego z powyższych.

```
syms x;
V=x^2;
V_der=-2*x^2;
x_der=-x;

Stable(V,V_der,x)
Calc(V,x_der)
```

6 wywołanie funkcji

6. Propozycja implementacji przedstawionych reguł logicznych

Program wywoływany (rysunek 6) jest jako funkcja w specjalnie stworzonym do tego skrypcie. Definiowane tam są $V(x)$ i $\dot{V}(x)$ za pomocą Symbolic Math w matlabie.

```
function deriv = Calc(V, x_der)
    deriv = diff(V) * x_der;
end
```

7 Wywołanie funkcji