

Київський національний університет імені Т. Шевченка
Факультет комп'ютерних наук та кібернетики

Моделювання систем
Лабораторна робота №2
Звіт

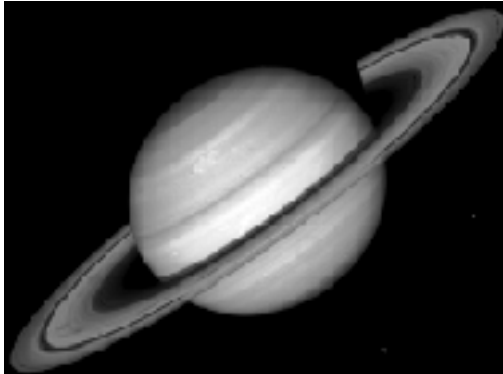
Виконав: студент групи ІПС-31
Бартіш Андрій

Київ-2021

Варіант 2

1. Початкові дані:

X1



Y2



2. Знаходження Z функції:

```
function returnable_value = Z(A, A_pseudo)
    returnable_value = eye(size(A_pseudo, 1)) - A_pseudo * A;
endfunction
```

3. Знаходження псевдооберненої матриці використовуючи формулу Гревіля:

```
function returnable_value = Grevil(A)
    is_swap = 0;
    if (size(A, 1) > size(A, 2))
        is_swap = 1;
        A = A';
    endif

    current_vector = A(1, :);
    A_pseudo = 0;
    current_vector_scalar = current_vector' * current_vector;
    if (current_vector_scalar == 0)
        A_pseudo = current_vector;
    else
        A_pseudo = current_vector / current_vector_scalar;
    endif

    A_iter = current_vector';
    for i = 2 : size(A, 1)
        current_vector = A(i, :);
        Z_A = Z(A_iter, A_pseudo);
        A_iter = [A_iter; current_vector'];
        denom_Z = current_vector' * Z_A * current_vector;
        if (denom_Z > 0.000001)
            A_pseudo = [(A_pseudo - (Z_A * current_vector * current_vector' * A_pseudo) / denom_Z), (Z_A * current_vector) / denom_Z];
        else
            R_A = A_pseudo * A_pseudo';
            denom_R = 1 + current_vector' * R_A * current_vector;
        end
    end
```

```

        A_pseudo = [(A_pseudo - (R_A * current_vector * current_vector' * A_pseudo) / denom_R), (R_A * current_vector) / denom_R];
    endif
endfor

if (is_swap)
    A_pseudo = A_pseudo';
endif
returnable_value = A_pseudo;
endfunction

```

4. Знаходження J функції:

```

function returnable_value = J (A_current, A_next)
    returnable_value = max(max((A_current - A_next) .^ 2));
endfunction

```

5. Знаходження псевдооберненої матриці використовуючи формулу Мура-Пенроуза:

```

function returnable_value = MurPenroze (A)
    is_swap = 0;
    if (size(A,1) > size(A,2))
        is_swap = 1;
        A = A';
    endif

    epsilon = 1e-8;
    infinity = 1e9;
    A_pseudo_current = infinity * ones(size(A))';
    A_pseudo_next = -infinity * ones(size(A))';
    delta = 10.0;
    while (J(A_pseudo_current, A_pseudo_next) > epsilon)
        A_pseudo_current = A_pseudo_next;
        A_pseudo_next = A' * inv(A * A' + (delta) * eye(size(A, 1)));
        delta /= 2.0;
    endwhile

    if (is_swap)
        A_pseudo_next = A_pseudo_next';
    endif
    returnable_value = A_pseudo_next;
endfunction

```

6. Знаходимо лінійний оператор переходу вхідного сигналу у вихідний для Гревіля:

```

X_pseudo = Grevil(X);
A = Y * X_pseudo + rand(size(Y, 1), size(X, 1)) * Z(X_pseudo, X);

```

7. Аналогічно для Мура-Пенроуза

```

X_pseudo = MurPenroze(X);
A = Y * X_pseudo + rand(size(Y, 1), size(X, 1)) * Z(X_pseudo, X);

```