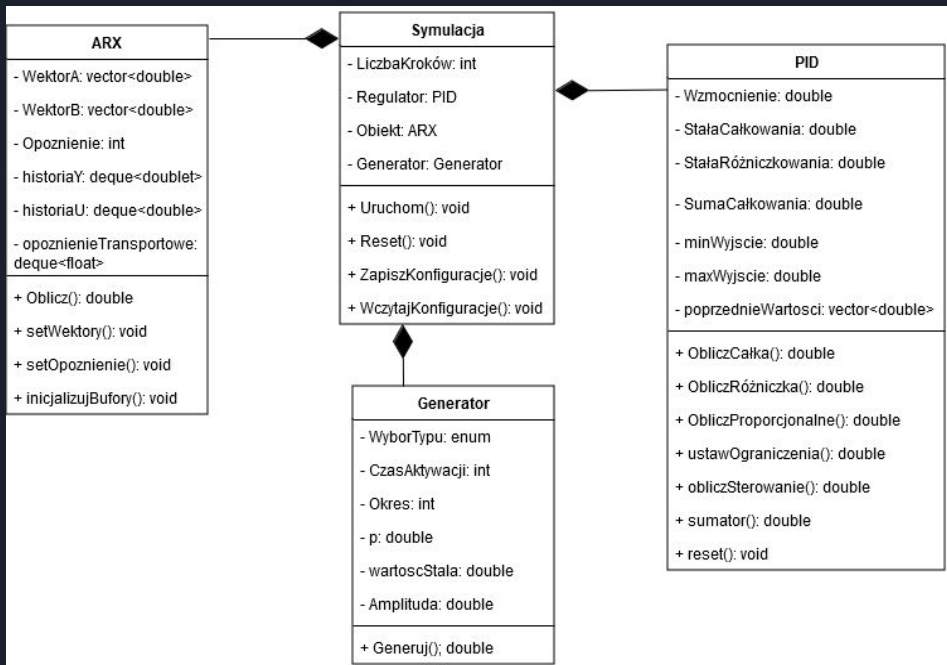


SYMULATOR UKŁADU AUTOMATYCZNEJ REGULACJI

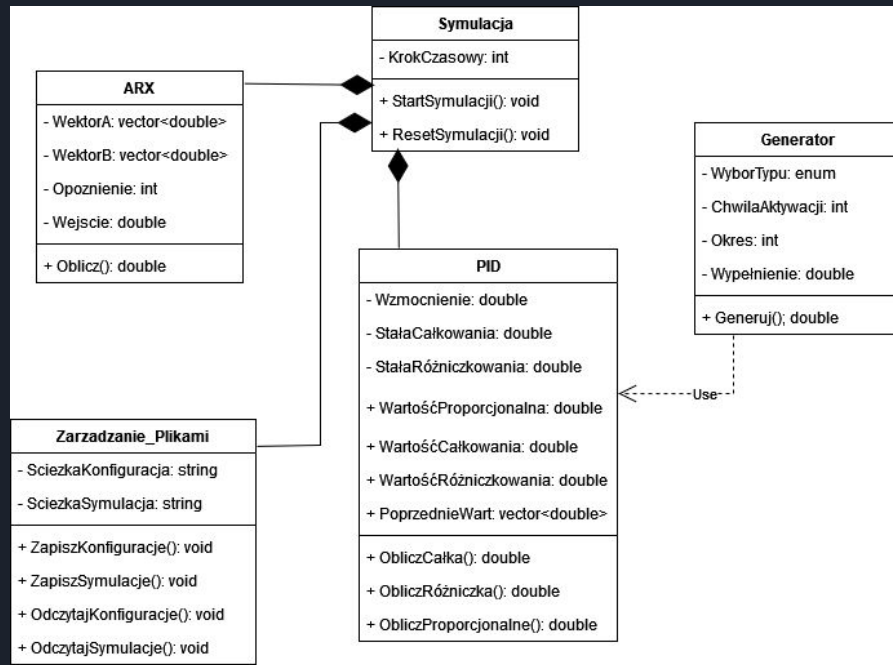
Rafał Cholewa
Artur Albrecht
IPpp sekcja 1

Obecny schemat UML w porównaniu ze starym

Nowy schemat



Stary schemat



```

double ARX::Oblicz(float u)
{
    double mean = 0, stdev = 0.1;

    random_device gen_los;
    mt19937 gen_plos;
    gen_plos.seed(gen_los());

    normal_distribution<float> rozkladGausa(mean, stdev);
    double zaklocenia = rozkladGausa(gen_plos);

    double wyjscie = 0.0f;

    opoznienieTransportowe.push_back(u);

    if (opoznienieTransportowe.size() > opoznienie)
    {
        double opoznione_u = opoznienieTransportowe.front();
        opoznienieTransportowe.pop_front();

        historiaU.push_front(opoznione_u);
        if (historiaU.size() > wektorB.size())
            historiaU.pop_back();
    }

    for (int i = 0; i < wektorA.size(); i++)
        wyjscie -= wektorA[i] * historiaY[i];
    for (int j = 0; j < wektorB.size(); j++)
        wyjscie += wektorB[j] * historiaU[j];

    historiaY.push_front(wyjscie);
    if (historiaY.size() > wektorA.size())
        historiaY.pop_back();

    return wyjscie + zaklocenia;
}

```

Największe problemy ARX

Najwięcej satysfakcji PID

```
double PID::ObliczProporcjonalne(double uchyb)
{
    return wzmacnienie * uchyb;
}

double PID::ObliczCalka(double uchyb)
{
    if (stalaCalkowania)
    {
        sumaCalkowania += uchyb;
        return sumaCalkowania / stalaCalkowania;
    }
    else return 0;
}

double PID::ObliczRozniczka(double uchyb)
{
    double wartoscRozniczujaca = 0.0f;
    if (!poprzednieWartosci.empty())
        wartoscRozniczujaca = stalaRozniczkowania * (uchyb - poprzednieWartosci.back());
    else wartoscRozniczujaca = stalaRozniczkowania * uchyb;

    poprzednieWartosci.push_back(uchyb);

    return wartoscRozniczujaca;
}

double PID::ObliczSterowanie(double uchyb)
{
    double proporcjonalne = ObliczProporcjonalne(uchyb);
    double calka = ObliczCalka(uchyb);
    double rozniczka = ObliczRozniczka(uchyb);

    double sterowanie = proporcjonalne + calka + rozniczka;

    if (sterowanie < minWyjscie) sterowanie = minWyjscie;
    else if (sterowanie > maxWyjscie) sterowanie = maxWyjscie;

    return sterowanie;
}
```