

倒立振子シミュレータ Rev1

トランジスタ技術 2019年7月号 DVD 付録の「倒立振子シミュレータ」に、修正点が見つかりました。詳細は以下の通りです。

【対象プログラム名】

(旧) SolveRiccatiEquation.py

(新) SolveRiccatiEquation_rev1.py

【修正箇所】

(旧) 374 行目

```
x = numpy.dot(A_BK, x)
```

(新) 374 行目

```
x = numpy.dot(Ax, x) + numpy.dot(Bx, Vin)
```

【解説】

修正前のプログラムでは、次状態 \mathbf{x}_{k+1} を次のように計算していました (374 行目)。ただし次式の“ K ”は状態フィードバック・ゲインで、プログラム中では“Gain”という変数としています。

$$\begin{aligned}\mathbf{x}_{k+1} &= A_x \mathbf{x}_k + B_x (K \mathbf{x}_k) \\ &= (A_x + B_x K) \mathbf{x}_k\end{aligned}$$

上式の“ $K \mathbf{x}_k$ ”はモータに印加する電圧“ V_{in} ”に相当します。プログラム中ではこの値が 3.3 V (絶対値) を超えた場合に、強制的に“ $V_{in} = 3.3$ ” (あるいは“ $V_{in} = -3.3$ ”) と書き換えています。しかし、次状態 \mathbf{x}_{k+1} を求める式は上式のとおりでした。これでは、常に理想的な“ $K \mathbf{x}_k$ ”という電圧 (3.3 V 以上になる可能性もある) をモータに印加することになってしまいます。

そこで、新しいプログラムでは次状態 \mathbf{x}_{k+1} を計算する式 (374 行目) を次のように修正しました。

$$\mathbf{x}_{k+1} = A_x \mathbf{x}_k + B_x V_{in}$$

これで、 V_{in} がリミッタ処理によって 3.3 V に制限されている場合でも、現実のシステムに近い結果を算出できます。なお、モータ電圧のオフセット分は考慮していません。これは「倒立振子が立つか否か」という問題に対してクリティカルではないためです。もしオフセットについてもシミュレーションに含めたい場合は、 V_{in} の正負に応じて場合分けをし、オフセット電圧を加算する (負の場合は減算する) ようにしてください。