

極座標ステージとエクストルーダの同期に関する考察

青木 翔平

June 16, 2016

1 線速度について

デカルト座標系での線速度 v は極座標で以下のように表される．

$$\begin{aligned} |v| &= \sqrt{\dot{x}^2 + \dot{y}^2} \\ &= \sqrt{\left(\frac{d}{dt}r \cos \theta\right)^2 + \left(\frac{d}{dt}r \sin \theta\right)^2} \\ &= \sqrt{\dot{r}^2 + r^2\dot{\theta}^2} \end{aligned} \quad (1)$$

いま，角速度と半径方向の速度を制御して線速度一定にすることを考える． $v = k$ （線速度一定）とすると，

$$\dot{r}^2 + r^2\dot{\theta}^2 = k^2 \quad (2)$$

$$\dot{r}^2 = k^2 - r^2\dot{\theta}^2 \quad (3)$$

微小区間で考えると

$$\begin{aligned} \left(\frac{\Delta r}{\Delta t}\right)^2 &= k^2 - r^2 \left(\frac{\Delta \theta}{\Delta t}\right)^2 \\ |\Delta r| &= \sqrt{(k\Delta t)^2 - (r\Delta \theta)^2} \end{aligned} \quad (4)$$

いっぽう，NC コードにおいて直線移動命令 G1 が与えられたとき，CNC コントローラではブレンハムのアルゴリズム [Figure 1] にしたがって補間位置 $\{x_1, x_2, \dots, x_n\}, \{y_1, y_2, \dots, y_n\}$ が計算される．また，計算された $\{x_1, x_2, \dots, x_n\}, \{y_1, y_2, \dots, y_n\}$ は以下の関係式を用いて $\{r_1, r_2, \dots, r_n\}, \{\theta_1, \theta_2, \dots, \theta_n\}$ へと座標変換される．

$$\begin{cases} r_i = \sqrt{x_i^2 + y_i^2} \\ \theta_i = \tan^{-1}(y_i/x_i) \end{cases}$$

すなわち，差分形式とすれば

$$\begin{cases} \Delta r = r_2 - r_1 = \sqrt{x_2^2 + y_2^2} - \sqrt{x_1^2 + y_1^2} \\ \Delta \theta = \theta_2 - \theta_1 = \tan^{-1}(y_2/x_2) - \tan^{-1}(y_1/x_1) \end{cases} \quad (5)$$

一般に，(5) で指定される $\Delta r, \Delta \theta$ が (4) 式を満たすとは限らないため¹，任意の目的地の座標が指定されたときに線速度を一定にすることはできない．

```

dx ← x1 - x0
dy ← y1 - y0
D ← 2 * dy - dx
plot(x0, y0)
y ← y0
for x from x0 + 1 to x1 do
  if D > 0 then
    y ← y + 1
    plot(x, y)
    D ← D + (2 * dy - 2 * dx)
  else
    plot(x, y)
    D ← D + (2 * dy)
  end if
end for

```

Figure 1: Bresenham's algorithm

2 エクストルーダの制御について

ステージの変位と射出量の関係について考える．

デカルト座標系において，移動時間 Δt と線速度 $F[\text{mm/min}]$ の関係は，以下のように表される．

$$\Delta t = \frac{\sqrt{(\Delta x)^2 + (\Delta y)^2}}{F} \cdot 60 \cdot 1000 \quad (6)$$

エクストルーダ（スクリュー）の射出量を ΔE_{screw} ，スクリューの回転数を $SNW[\text{rpm}]$ とおけば，以下の式

¹ r の自由度が残る．

が成り立つ．

$$\begin{aligned}\Delta E_{screw} &= \beta \cdot (SNW) \cdot \Delta t \\ \therefore SNW &= \frac{1}{\beta} \frac{\Delta E_{screw}}{\Delta t} = \beta' \frac{F \cdot \Delta E_{screw}}{\sqrt{(\Delta x)^2 + (\Delta y)^2}}\end{aligned}\quad (7)$$

前回，スライサーからの $\Delta E = E_2 - E_1$ を ΔE_{screw} と等しいとみなして計算を行ったが，これは誤りである．なぜならば，スライサーが出力した E には極座標の位置による速度変化が考慮されていないからである．したがって，スライサーからのエクストルーダの値は無視する（実際の実装ではエクストルーダの値は射出の On/Off 判定のために使用している）

極座標におけるステージの線速度を $F_{polar}(t)$ とすれば，微分を後退差分表示することで以下の式を得る．

$$\begin{aligned}F_{polar}(t_k) &= \left. \frac{dS}{dt} \right|_{t=tk} \\ &= \left. \sqrt{\dot{r}^2 + r^2 \dot{\theta}^2} \right|_{t=tk} \\ &= \sqrt{\left(\frac{r_k - r_{k-1}}{\Delta t} \right)^2 + \left(\frac{r_k + r_{k-1}}{2} \right)^2 \left(\frac{\theta_k - \theta_{k-1}}{\Delta t} \right)^2} \\ \text{ステージの変位とスクリューの変位は比例するので} \\ \int_{t_1}^{t_2} SNW(t) dt &= \alpha \int_{t_1}^{t_2} F_{polar}(t) dt \\ \therefore SNW|_{t=t_k} &= \alpha F_{polar}(t_k) \\ &= \frac{\alpha}{\Delta t} \sqrt{(r_k - r_{k-1})^2 + \left(\frac{r_k + r_{k-1}}{2} \right)^2 (\theta_k - \theta_{k-1})^2} \\ &= \frac{\alpha \cdot 60 \cdot 1000 \cdot \sqrt{(\Delta x)^2 + (\Delta y)^2}}{F} \cdot \sqrt{(r_k - r_{k-1})^2 + \left(\frac{r_k + r_{k-1}}{2} \right)^2 (\theta_k - \theta_{k-1})^2} \\ &= \frac{\alpha'}{F} \sqrt{(r_k - r_{k-1})^2 + \left(\frac{r_k + r_{k-1}}{2} \right)^2 (\theta_k - \theta_{k-1})^2} \\ &\quad \cdot \sqrt{(r_k \cos \theta_k - r_{k-1} \cos \theta_{k-1})^2 + (r_k \sin \theta_k - r_{k-1} \sin \theta_{k-1})^2}\end{aligned}\quad (8)$$

(8) 式にしたがってスクリュー回転数を制御することで，線速度が変化する状況においてもスクリューの射出量を線速度に追従させることができる．

3 予想される問題点

3.1 半径の近似精度

(8) 式に表される r_k の計算において、 r_k と r_{k-1} の平均値を用いているが、 r が急激に増減する場合にこの部分の誤差の影響が大きくなる。一部の特殊な形状 例えば鋭利なヒトデ型 を造形する際に問題が懸念されるが、これは与えられる G-code の半径方向の刻み ($= \Delta r$) が小さい場合、実際は問題にならないと考えられる。

4 試験結果をうけた制御式の変更

(8) 式にしたがって実際にプリント試験を行ったところ、角速度の変動が速すぎて射出量が追従できない、あるいは角速度が位置によってうまく変動しないなどの不具合があった。微分の誤差が効いているものと思われる。

したがって制御量の修正を行った。修正後の関係式を以下に示す。

$$SNW = \gamma \frac{F \cdot r \cdot \Delta E}{\sqrt{(\Delta x)^2 + (\Delta y)^2}} \quad (9)$$

(9) 式は原点より遠い場所ほどスクリー回転数を高めるというアドホックな方法であるが、直感的にもわかりやすく、またうまくいく方法であった。

5 議論

現時点で判明している問題点として、

1. そもそも吐出側を一定にしたい
2. 原点での切り返し問題
3. エクストルーダの On/Off の遅延

がある。以下でそれぞれについて考察する。

5.1 吐出側を一定にする

現在の計画では G-code に応じて吐出側、すなわちスクリーの角速度を追従させることを目的としているが、これには以下の欠点がある。

- 回転数値が指令されてから実際にスクリーの回転数に変更されるまでに遅延がある
- スクリーの回転数に変更されたあと、射出量が追従するまでに遅延がある

- 吐出量の変動に応じて冷却を調節する必要があるが、冷却はリニアに効くわけではない

したがって、ステージの線速度を制御したほうがよいという結論になるが、前章で述べたように極座標において線速度を一定にすることは理論的に不可能である。

では、逆にエクストルーダが一定速度で回転しているとして、これにステージ側の線速度を追従させることは可能であろうか。極座標系での速度は r と θ を用いて (2) 式で与えられるが、 v を与えた場合に Δr と $\Delta \theta$ は一意に決まるわけではない。加えて前章での経験では、平方根および乗算演算に起因する数値誤差が懸念されるので、直接的に r と θ を決定することは避けたい。

ここで発想を変えて、射出量に応じた線速度から直接 r と θ の制御量を導くのではなく、なるべくこれに近い線速度でステージが動くように本体を制御することを考える。具体的には、以下のように問題を単純化する。

仮定 1: 半径方向の速度変化は比較的小さく、回転方向の運動が主である

この仮定をおけば (2) 式は $v = r\omega$ と簡素化できるから、半径に反比例するようにステージ側の回転テーブルの回転数を調節すれば、エクストルーダのスクリー回転数に追従することができる。

仮定 1 の妥当性として、かりに r の変動量 \dot{r} が大きかったとしてもそれは瞬間的なものであり、スクリー回転数上昇後の樹脂吐出量の応答が低速であることを踏まえれば、この変動成分は無視できると可能性が高い。 r の絶対量は仮定後の式でも吸収できるから、問題はないと考えられる。

5.2 原点での切り返し問題

現在のファームウェアの実装では原点で進行方向を切返すために、原点近傍での出力に継ぎ目が見られる。これを避けるためにはファームウェアのロジックを変更する必要がある。

原点での切り返しの問題になるかどうかは造形物の形状に依存する。例えばコルセットをステージ中央に配置したときのプリントでは問題になっていない。まっすぐな円筒なども問題は発生しないが、折れ曲がった指先が原点にかかるような配置を行うと原点での切り返しが発生する。

5.3 エクストルーダの On/Off の遅延

スクリー回転数にゼロを指定するとピンが作動して射出が止められるが、このピンが再度開いた時に樹脂が流れ出るまでに遅延が発生する。これを解決する一つの方

法は G-code 指令を先読みすることでこの遅延分を見込んだ開閉を行うことである。

これの実現方法には (1) G-code を先読みする方法と、(2) エクストルーダのピン開閉時にステージ側のモーターを停めてしまう方法の 2 通りが考えられる。組み込みの実装としては後者のほうが簡易であったので、これを採用した。

A 極座標における Jerk について

前章で述べたように、極座標系において一般に線速度は一定にすることができない。 \dot{r} と $\dot{\theta}$ の振る舞いはファームウェア内で加速度を設定することで操作できるが、この際の瞬発的な加速度の増加（加加速度 \dot{a} ）を Jerk と呼んでいる。

いま、極座標で加速度制御を行ったときに線速度のプロファイルがどのような様子を描くかに興味がある。極座標系での線速度は以下の式 (10) から計算できる。半径方向の速度 \dot{r} と角速度 $\dot{\theta}$ を適当な台形プロファイルで仮定して数値計算した結果を Figure 2 に示す。図中の右で示される v が極座標系における線速度である。

$$|v| = \sqrt{\dot{r}^2 + r^2 \dot{\theta}^2} \quad (10)$$

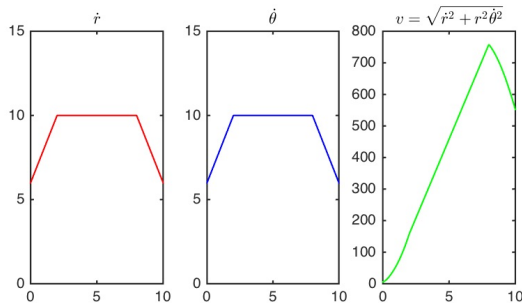


Figure 2: Velocity profile

Figure 2 から、 \dot{r} と $\dot{\theta}$ を台形型の速度プロファイルで制御したとしても、線速度は台形型とならないことがわかる。これはデカルト座標系との違いである。

References

- [1] 伊理正夫, 藤野和建『数値計算の常識』, 共立出版 (1985)