

何をしているプログラムなのか

脳を構成している神経細胞の活動をシミュレーションするプログラムです。
単一の神経細胞の活動は、膜電位（細胞膜の内側と外側の電位差） V_m によって表現されます。
膜電位の振る舞いや、複数の神経細胞の間の相互作用（シナプス接続）の振る舞いは主に以下の方程式群によって記述することができます。

$$C_m \frac{dV_m}{dt} = G_L (E_L - V_m) + I_{app}; \text{ if } V_m > V_{th} \text{ then } V_m \mapsto V_{reset}.$$
$$I_{\text{syn}}(t) = g_{\text{syn}}(t) (u(t) - E_{\text{syn}}).$$
$$g_{\text{syn}}(t) = \sum_f \bar{g}_{\text{syn}} \left[1 - e^{-(t-t^{(f)})/\tau_{\text{rise}}} \right] \left[a e^{-(t-t^{(f)})/\tau_{\text{fast}}} + (1-a) e^{-(t-t^{(f)})/\tau_{\text{slow}}} \right] \Theta(t - t^{(f)}),$$

※一部省略して載せています。

お渡ししたプログラムは、設定したパラメータをもとに微分方程式を逐次処理で解き、その結果をCSVファイルに保存するプログラムとなっています。

工夫した点

- ・ 並列処理
実際にシミュレーションを行う際には、様々な初期パラメータでのシミュレーションを行う必要があります。
必要なシミュレーション条件の数は、数**100**通りになる場合もありますので、
CPUの1コアごとに独立したシミュレーションを走らせるようにし、並列処理を行うようにしてあります。
例えば、**160**通りの条件を**8**スレッド**CPU**を用いてシミュレーションする場合、
20通りのシミュレーション時間（+オーバーヘッド）で済むので計算時間を節約できます。
- ・ 拡張性
神経細胞の数や細胞間の接続関係を自由に設定することができ、数百個の神経細胞ネットワークのシミュレーションも行うことができます。
これを実現するため、方程式のパラメータの殆どを二次元配列で扱っています(**Neuron_LIF**クラスの**self.V**など)。
- ・ メモリ節約
少ないメモリでもシミュレーションを実行できるようにしてあります。
神経細胞の数やシミュレーション時間が増えると、メモリ使用量が掛け算で増加していきます。
そのため、確保する配列の長さに制限を設けており、制限以上の時間でのシミュレーションを行う場合は、シミュレーションを分割して同じ配列を使いまわすことで、メモリ使用量を抑えています。