

# Optymalna strategia podawania leku

Tomasz Kanas

21 maja 2020

## 1 Sformułowanie problemu

Celem pracy jest znalezienie strategii podawania leku, przy leczeniu nowotworu, pozwalającej osiągnąć możliwie największą skuteczność terapii. W tym celu skorzystamy z modelu przedstawionego w pracy [tu wypada zacytować wyjściową pracę]. Model ten przedstawia rozwój nowotworu w czasie w zależności od dawki leku za pomocą równania różniczkowego:

$$\begin{aligned}V_1'(t) &= \lambda_1 V_1 F\left(\frac{V_1 + \alpha_{12} V_2}{K}\right) - \beta_1 V_1 g(t) \\V_2'(t) &= \lambda_2 V_2 F\left(\frac{V_2 + \alpha_{21} V_1}{K}\right) - \beta_2 V_2 g(t) \\K'(t) &= -\mu K + (V_1 + V_2) - d(V_1 + V_2)^{2/3} K - \beta K g(t) \\V_1(0) &= 20, \quad V_2(0) = 280, \quad K(0) = 650\end{aligned}\tag{1}$$

Gdzie  $F(x) = -\ln(x)$ , oraz stałe mają wartości:

$t \in [t_0, T]$	$\alpha_{12} = ??$ (przyjąłem 0.1)	$\beta_1 = 0.15$	$\lambda_1 = 0.192$
$t_0 = 0$	$\alpha_{21} = ??$ (przyjąłem 0.15)	$\beta_2 = 0.1$	$\lambda_2 = 0.192$
$T = 200$		$\beta = 0.05$	$\mu = 0$
$0 \leq g(t) \leq g_{\max} = 3$			$d = 0.00873$

Funkcja  $V_1(t)$  modeluje liczbę komórek guza podatnych na lek w momencie  $t$ ,  $V_2(t)$  liczbę komórek guza odpornych na lek, a  $K(t)$  jest parametrem nazwanym w pracy “uczynnieniem”. Praca podaje też funkcjonal celu, który pozwala porównać różne przebiegi leczenia i tym samym stwierdzić która strategia podawania leku daje lepsze rezultaty. W tym przypadku im niższa wartość funkcji tym lepszy przebieg leczenia:

$$J(V_1, V_2, K) = \int_0^T V_1(t) + V_2(t) dt + \omega \int_0^T G\left(\frac{V_2(t) - V_1(t)}{\epsilon}\right) dt\tag{2}$$

$$G(x) = \frac{1 + \tanh(x)}{2} \quad \omega = ?? \text{ (przyjąłem 1000)}, \quad \epsilon = 0.01$$

Parametrem równania różniczkowego (1) jest funkcja  $g : [t_0, T] \rightarrow [0, g_{\max}]$ . Jeśli dla danej funkcji  $g$  równanie (1) posiada rozwiązanie, to możemy przypisać jej wartość funkcjonału  $J$  dla tego rozwiązania. Oznacza to, że możemy przyjąć  $J(g) := J(V_1, V_2, K)$ , gdzie  $V_1, V_2, K$  spełniają równanie (1). Dzięki temu naszym celem staje się znalezienie takiej funkcji  $g$  dla której istnieje rozwiązanie równania (1) oraz wartość  $J(g)$  jest minimalna. Problem ten w literaturze nazywa się problemem optymalnego sterowania, a funkcję  $g$  sterowaniem.

## 1.1 Problem przybliżony

Analityczne rozwiązywanie problemu optymalnego sterowania rzadko kiedy jest możliwe, a nawet gdy jest możliwe, jest trudne. Z tego powodu zdecydujemy się na szukanie rozwiązania przybliżonego.

Najpierw ograniczmy problem do problemu optymalizacji skończenie wymiarowej. W tym celu wprowadźmy dyskretyzację czasu, poprzez ustalenie siatki dyskretyzacji, czyli ciągu punktów w czasie:

$$0 = t_0 < t_1 < \dots < t_{n-1} < t_n = T \quad (3)$$

oraz wartości sterowania w tych punktach

$$g_i = g(t_i) \quad (4)$$

Możemy teraz przybliżać sterowanie za pomocą funkcji stałej na przedziałach  $[t_i, t_{i+1})$ , lub splajnem z węzłami w punktach  $t_i$ . Mając tak przybliżoną funkcję sterowania możemy numerycznie przybliżyć rozwiązanie równania różniczkowego (1), oraz wartość całek występujących w funkcji celu (2). Otrzymujemy w ten sposób przybliżoną funkcję celu  $\hat{J} : [0, g_{\max}]^n \rightarrow \mathbb{R}$  i tym samym problem przybliżony:

$$\begin{aligned} \min_{g_1, \dots, g_n} \hat{J}(g_1, \dots, g_n) \text{ z ograniczeniami} \\ \forall_{i \in \{1, \dots, n\}} 0 \leq g_i \leq g_{\max} \end{aligned} \quad (5)$$

Jest to problem optymalizacji nieliniowej z ograniczeniami. Rozwiązanie tego problemu jest poza zakresem tej pracy i zadowolimy się tu faktem, że istnieją metody pozwalające uzyskać przybliżone rozwiązanie tego problemu, oraz implementacje tych metod.

To podejście do numerycznego problemu optymalnego sterowania jest zaproponowane w [1] i nazywa się sekwencyjnym (ang. “sequential”).

## 1.2 Plan rozwiązania

Aby obliczyć wynik problemu przybliżonego skorzystamy ze środowiska MatLab/Octave wraz z dostarczonym z nim optymalizatorem problemu optymalizacji nieliniowej z ograniczeniami (odpowiednio FMINICON w MatLab, NONLIN\_MIN w Octave). W tym celu zaimplementujemy przejście od problemu optymalnego sterowania. Będziemy też musieli znaleźć odpowiednią siatkę dyskretyzacji i punkt startowy dla optymalizatora.

## 2 Implementacja

Optymalizator NONLIN\_MIN w Octave posiada prosty interfejs, jest to funkcja której należy dostarczyć funkcję do zoptymalizowania, punkt startowy, ograniczenia oraz ewentualne opcje dodatkowe.

Zauważmy, że znalezienie dobrego wyniku będzie wymagało zapewne wielokrotnego wywołania naszej funkcji celu przez optymalizator, więc zależy nam aby funkcja celu liczyła się możliwie szybko. Lepsza wydajnościowo implementacja pozwoli też na większe zagęszczenie siatki dyskretyzacji i tym samym wzrost dokładności aproksymacji.

Jedną z praktyk pozwalającą poprawić wydajność programów w środowiskach MatLab i Octave jest tak zwana wektoryzacja. Polega ona na zastępowaniu pętli operacjami na wektorach. Korzysta to z faktu, że wiele funkcji w tych środowiskach można wywołać z wektorem parametrów, zamiast pojedynczego parametru i zwracają one wtedy wektor wyników, oraz wykonują się znacznie szybciej niż gdyby wywołać je wielokrotnie w pętli. Z tego powodu będziemy korzystać z tej techniki gdzie to tylko możliwe.

Aby sprowadzić problem optymalnego sterowania do problemu optymalizacji nieliniowej musimy przybliżyć rozwiązanie równania różniczkowego (1) a następnie za pomocą uzyskanego rozwiązania przybliżyć wartość funkcji celu (2).

## 2.1 Równanie różniczkowe

Do przybliżenia równania (1) skorzystamy z dostarczonej w Octave implementacji metody Dormanda-Prince'a 4-tego rzędu ze zmiennym krokiem. Jest to popularna metoda pozwalająca uzyskać wysoką dokładność, ponadto umożliwia przekazanie wektora punktów w czasie w których chcemy otrzymać przybliżenie wyniku. Pozwala to rozwiązywać równanie w sposób zwektoryzowany i okaże się bardzo wygodne przy implementacji funkcji celu.

## 2.2 Funkcja celu

Aby obliczyć wartość funkcji celu użyjemy dostarczonej w Octave implementacji adaptacyjnej kwadratury Gaussa-Konroda. Osiąga ona dobrą dokładność, ewaluując odcałkowywaną funkcję w dość niewielu punktach. Zakłada też, że odcałkowywana funkcja jest zwektoryzowana, co wykorzystamy do poprawy wydajności rozwiązania.

Pewną niedogodnością jest fakt, że nie możemy nic zakładać o kolejności argumentów w wektorze który kwadratura przekazuje odcałkowywanej funkcji, natomiast funkcja rozwiązująca równanie różniczkowe wymaga by punkty były jej przekazane w kolejności rosnącej. Musimy więc posortować wektor argumentów przed przekazaniem ich do funkcji liczącej rozwiązanie równania różniczkowego, oraz przywrócić pierwotny porządek dla wyników tych obliczeń.

## Literatura

- [1] M. Diehl. *Numerical Optimal Control*. 2011.