Polo del Conocimiento

Pol. Con. (Edición núm. 70) Vol. 7, No 7
Julio 2022, pp. 1713-1734
ISSN: 2550 - 682X
DOI: 10.23857/pc.v7i7

Calibración de concentración de la señal de pet-scan a través de un modelo sokoloff 3k

Calibrating of concentration of the signal from pet-scan through a 3k sokoloff model

Calibração da concentração do sinal do pet-scan através de um modelo 3k sokoloff

Alex Eduardo Pozo Valdiviezo [I]
eduardo.pozo@espoch.edu.ec
https://orcid.org/0000-0003-0480-5669

Correspondencia: eduardo.pozo@espoch.edu.ec

Ciencias Técnicas y Aplicadas
Artículo de Investigación

* **Recibido:** 23 de mayo de 2022 ***Aceptado:** 12 de junio de 2022 * **Publicado:** 24 de julio de 2022

I. Facultad de Ciencias, Escuela Superior Politécnica de Chimborazo (ESPOCH), Riobamba, Ecuador.

Resumen

Este documento describe el uso de métodos de mínimos cuadrados no lineales y de Lagrangian aumentado para ilustrar su uso en la calibración de los parámetros del modelo Sokoloff 3K para encontrar la concentración óptima de la señal de un PET-Scan a lo largo del tiempo. Así como la descripción de métodos numéricos para EDOs como el método explícito de Euler, RK2 y RK4 para aproximar las soluciones del sistema a partir del modelo 3K Sokoloff y las soluciones del problema adjunto asociado al método Lagrangiano aumentado.

Palabras Clave: 3K Sokoloff Model; Problema de mínimos cuadrados no lineal; Método lagrangiano aumentado.

Abstract

This document describes the use of nonlinear least squares and augmented Lagrangian methods to illustrate their use in calibrating the parameters of the Sokoloff 3K model to find the optimal concentration of the signal from a PET-Scan over time. As well as the description of numerical methods for EDOs such as the explicit Euler method, RK2 and RK4 to approximate the solutions of the system from the 3K Sokoloff model and the solutions of the adjoint problem associated with the augmented Lagrangian method.

Keywords: 3K Sokoloff Model; Nonlinear least squares problem; Augmented Lagrangian Method.

Resumo

Este documento descreve o uso de métodos de mínimos quadrados não lineares e métodos Lagrangianos aumentados para ilustrar seu uso na calibração dos parâmetros do modelo Sokoloff 3K para encontrar a concentração ideal do sinal de um PET-Scan ao longo do tempo. Assim como a descrição de métodos numéricos para EDOs como o método de Euler explícito, RK2 e RK4 para aproximar as soluções do sistema a partir do modelo 3K de Sokoloff e as soluções do problema adjunto associado ao método Lagrangiano aumentado.

Palavras-chave: Modelo 3K Sokoloff; Problema não linear dos mínimos quadrados; Método Lagrangiano Aumentado.

Introducción

This project aims to estimate through a 3K Sokoloff model using methods such as nonlinear least squares and augmented Lagrangian to complete this project. The system describing the PETs is described by the Sokoloff model.

The processing of the data has several subdivisions, such as reading the data, splitting the data set to compare the Euler, RK2 and RK4 schemes and testing to obtain the concentration of the signal from the PET-Scan.

The concentration of the signal coming from the PET-Scan depends on the plasma concentrations, free concentrations and bound concentrations mentioned in Sokoloff's 3K model. We tried to solve this problem using the methods mentioned above, which allows us to fit the curve obtained from the Sokoloff model with the data.

ABOUT THE DATASET–

The dataset is composed of two files, a file with a sample of the $\overline{C}_{pet}(t)$ values for all the voxels affected by the tumor, the different measurement instants $t_1, t_2, \dots, t_{N_t-1=t_f}$, as well as a list of some complete 3D images (the whole thorax) i.e. for example for the instant t_n the table $(\overline{C}_{pet}(t_n))_{i,j,k}; \forall i, j, k,$ and a list of the instants corresponding to the full 3D images. While the other file contains the values of $\overline{C}_{plasma}(t)$, identical for all vectors and the different measurement instants $t_0, t_1, \dots, t_{N_t-1} = t_f$..

APPLICATION TO THE 3K SOKOLOFF MODEL–

Once a patient has done this examination, it is possible to do a pharmacokinetic analysis that aims to study the propagation of the tracer after its injection: how it was absorbed, how it was distributed in the body, how it was metabolized and finally how it was eliminated.

From the measurements obtained after a PET-scan, i.e. the C_{pet} signal obtained in each voxel, and the knowledge of the arterial plasma concentration of the unmetabolized tracer, named "input function" and noted C_{plasma} (considered as homegene within the feeder vessel), it is possible to model the functioning of the metabolism using a compartmental model.

Figure 1: Vriens, D., Visser, E. P., de Geus-Oei, L. F., & Oyen, W. J. (2010). Methodological considerations in quantification of oncological FDG PET studies. European journal of nuclear medicine and molecular imaging, 37(7), 1408–1425.

Let consider the following two-compartments model proposed by Sokoloff in 1978(1 with $k_4 = 0$):

$$\begin{cases} C_{free}{}'(t) = k_1 \overline{C}_{plasma}(t) - (k_2 + k_3)C_{free}(t) \\ \\ C_{bound}{}'(t) = k_3 C_{free}(t) \end{cases} \qquad (1)$$

Our aim is to have a fair description of the concentration of the signal coming from the PET-Scan

$$C_{pet}(t) = (1 - V_b)(C_{free}(t) + C_{bound}(t)) + V_b \overline{C}_{plasma}(t). \qquad (2)$$

where $C_{free}(t)$ represents the concentration of "free" tracer $C_{bound}(t)$ is the concentration of "bound" tracer, V_b is the blood volume fraction, k_1 and k_2 are the transmembrane passage rates (for each direction), and k_3 is the cytoplasmic phosphorylation rate.

To deal with this equation, one may first for the differential system proposed in equations (1), the target of work must be driven through several natural achievements:

- Numerical Solver of the differential system using Euler Explicit, Runge-Kutta order 2 and Runge-Kutta order 4.
- Quantify the numerical error for each chosen numerical solver.
- Appreciate the parameter sensitivity in term of affluence on the solution.
- Calibrate the model using a descent Gradient Algorithm without any constraint on the admissible solution domain.
- Calibrate the model using continuous Lagrangian, to incorporate some constraints on our optimization problem

METHODS USED–

NUMERICAL SOLVER–

For this particular system,analytic solution searching could be rubbish , thus we discretized the time variable into multiples of some small basic time increment Δt; whence time now takes the discrete values $0, \Delta t, 2\Delta t, \ldots, n\Delta t$. We replace the derivative by a finite difference which approximates the derivative for small Δt.

There are several ways to carry this out:

$$\text{Backwardfinite} - \text{difference:} \frac{dy(t)}{dt} \approx \frac{y(t)-y(t-\Delta t)}{\Delta t)} \quad (3)$$

$$\text{Centeredfinite} - \text{difference:} \frac{dy(t)}{dt} \approx \frac{y(t+\Delta t)-y(t-\Delta t)}{2\Delta t)} \quad (4)$$

$$\text{Forwardfinite} - \text{difference:} \frac{dy(t)}{dt} \approx \frac{y(t+\Delta t)-y(t)}{\Delta t)} \quad (5)$$

The error made in these approximations goes to 0 as $\Delta t \to 0$: It goes faster however to 0 in the central difference approximation than in forward or backward difference. We can see this doing a Taylor expansion:

$$y(t+\Delta t) = y(t) + y'(t)\Delta t + y''(t)\frac{\Delta t}{2!} + y'''(t)\frac{(\Delta t)^3}{3!} + \cdots \qquad (6)$$

It follows that if we denote the error by ε so that:

$$\varepsilon = \text{DifferenceApproximation} - \frac{dy(t)}{dt} \qquad (7)$$

then for the forward and backward error, $\varepsilon = \mathcal{O}(\Delta t)$ but $\varepsilon = \mathcal{O}((\Delta t)^2)$ for the centered one, we say that the forward and backward differences are *first order accurate* approximation.

We can use the difference approximation to obtain a numerical scheme for solving the ODE. Suppose first we wish to solve our system in forward time positive with an initial condition $y(0) = y_0$ given. If we use the forward difference to approximate the derivative, then we obtain the *explicit Euler Scheme*:

$$y(t+\Delta t) = y(t) + \Delta t f(t, y(t)). \quad (8)$$

Setting $y^n \sim y(n\Delta t)$ then yields the recurrence relation:

$$y^{n+1} = y^n + \Delta t f(n\Delta t, y^n), \quad n = 0,1,\ldots,N. \qquad (9)$$

The numerical algorithm is explicit in the sense that once we have computed y^n then y^{n+1} is easily computed using the previous relationship.

We runned our solver for a specific non optimal set of parameter and we plotted the numerical solution vs the empirical mean observation.

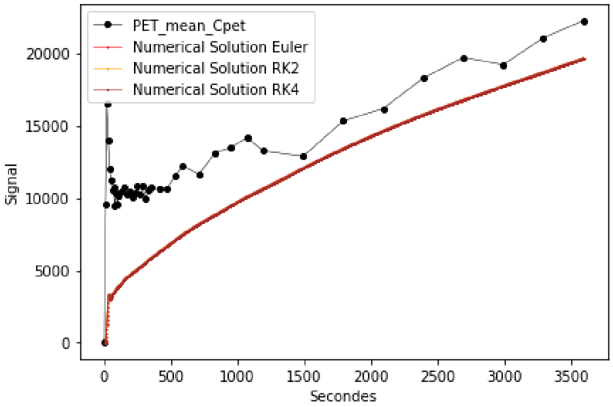


Figure 2: Average $\overline{C}_{pet}$ over time and the results obtained with the three methods

CONVERGENCE OF THE THE SCHEME AND ACCURACY ERROR–

Shape of our solution highlight our lack of fitting the curve for our chosen set of parameters , we will calibrate the model in a second phasis, but before doing so we will try for find the best numerical solution that reduce the error ε independently of the choice of our parameters.

To deal with that, we created a reference solution with the smallest time slice step, in aim to be closer to the continuous solution.

We retrieve for all possible time the reference solution coming from the smallest grid using a natural linear interpolation in aim to scale into this function bigger time steps.

We also provided a grid generator,grid batch function that allows to retrieve a bunch of grid for different format of time steps Δt.

Naive observation but for the reference grid i.e 2^{14+1} nodes, we have a perfect match of our all numerical method:

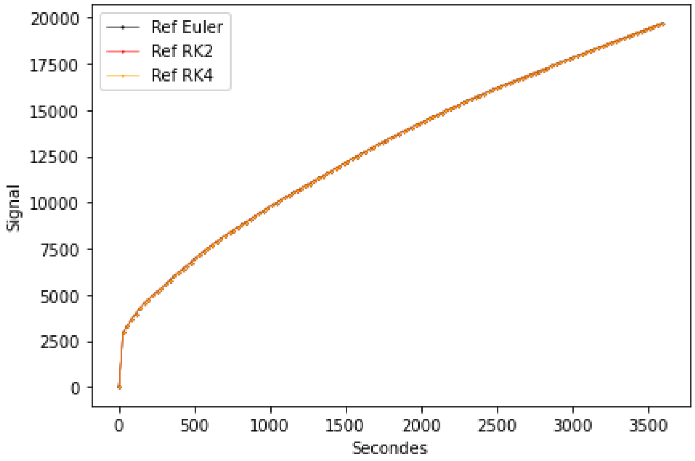


Figure 3: Results obtained with the three schemes

As illustration, we here for the biggest time grid the numerical solution compared to the reference one.

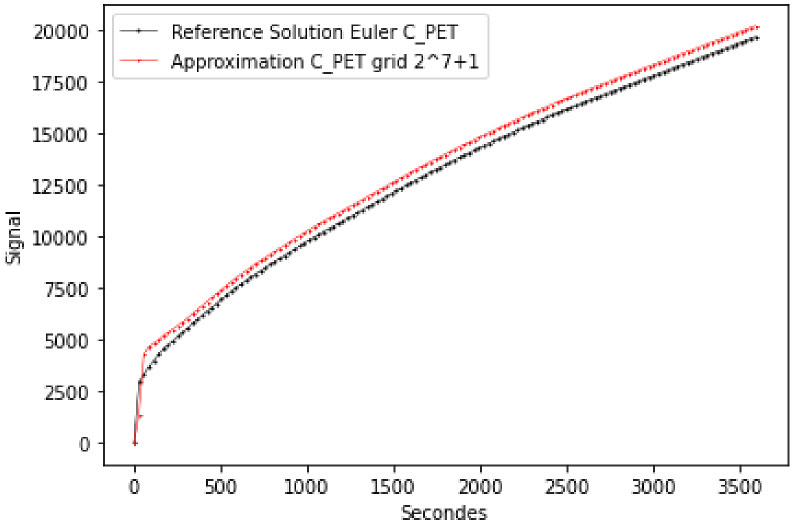


Figure 4: Results obtained with the three schemes using a reference solution

Numerical error is clearly adapted , we will try to illustrate for all the numerical scheme the order of the numerical error operated within grid choice.

To perform such a procedure, we take a grid batch and a numerical scheme in the input and return the tuple $(\log(\Delta t), \log E)$ where E is the most important local error for given a choice of grid.

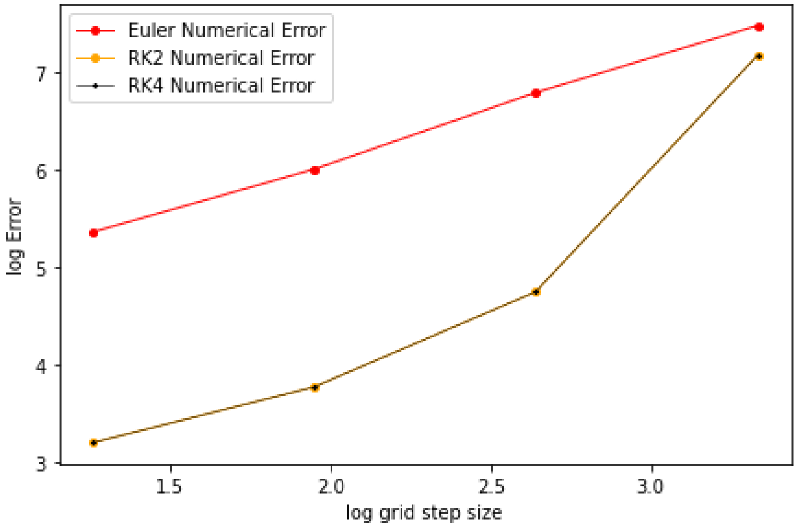


Figure 5: Curves $\ln(E)$ vs. $\ln(\Delta t)$ for each of the schemes

Our selection would be done in favor of Runge-Kutta 2 as there no improvement of Runge-Kutta 4 (Numerical error is the same than RK2 and complexity costs are more important) as we could expect theoretically, in aim to find the order of each method, we computed the slope using a linear regression.

We obtained the following results:

- Euler Method Slope $= 1$, the method is first order accurate as Taylor expansion could testify
- RK2 Method Slope $= 2$, the method is second order accurate as Taylor expansion could testify.
- RK4 Method Slope $= 2$, the method is second order accurate and Taylor expansion would give us order 4.

Parameters identification and sensitivities

As our choice in term of numerical error control has been done in favor of RK2, we will now aboard the calibration problematic but as first guess we will try to have a intuition on our parameter set sensitivity.

To deal with that let choose a couple of parameter and make a naive range variation of them and compute the solution for these precise choices and compare them to the empirical mean.

This manual method is done in respect within the unit figure variation for each parameter.

For k_1 we pass the interval [0,0.002] with step of 0.002 and we appreciate the marginal sensitivity.

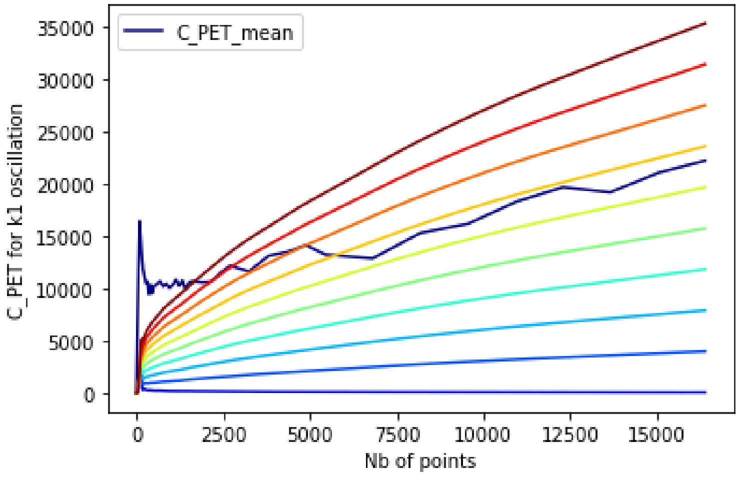


Figure 6: Variations of k_1 by setting the other parameters

We do the same procedure for k_2 we pass the interval [0,0.01] with step of 0.001.

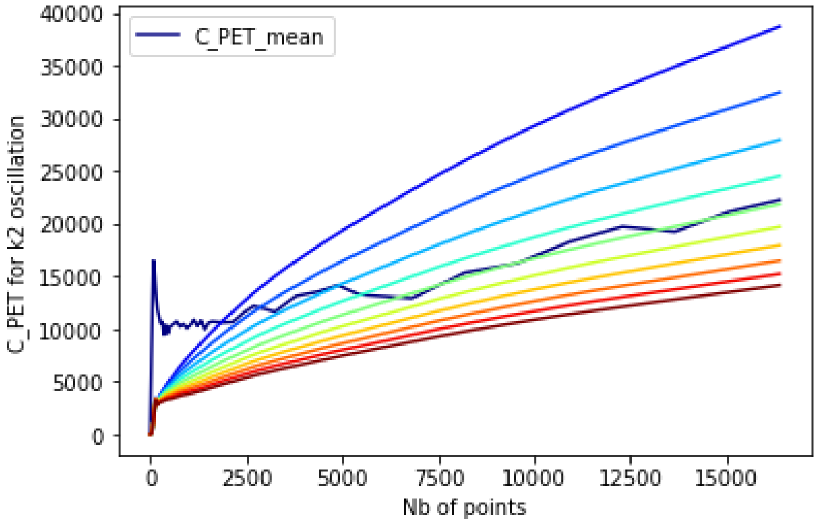


Figure 7: Variations of k_2 by setting the other parameters

We do the same procedure for k_3 we pass the interval [0,0.001] with step of 0.001.

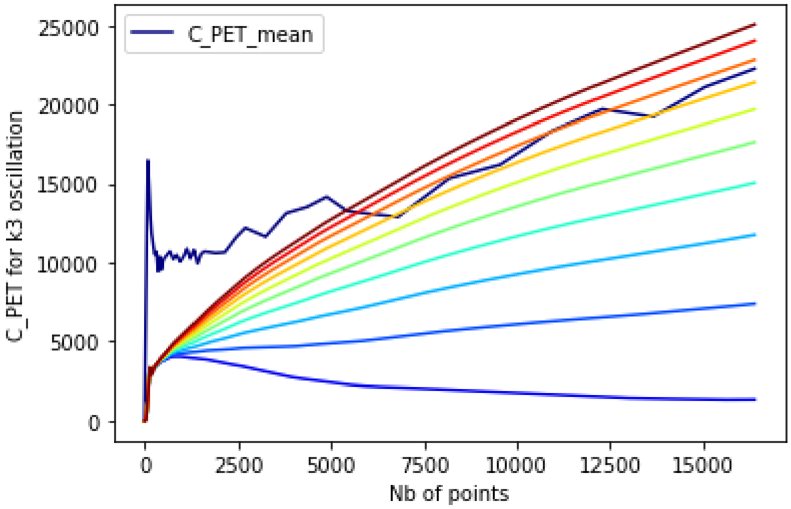


Figure 8: Variations of k_3 by setting the other parameters

We do the same procedure for V_b we pass the interval [0,0.1] with step of 0.01.

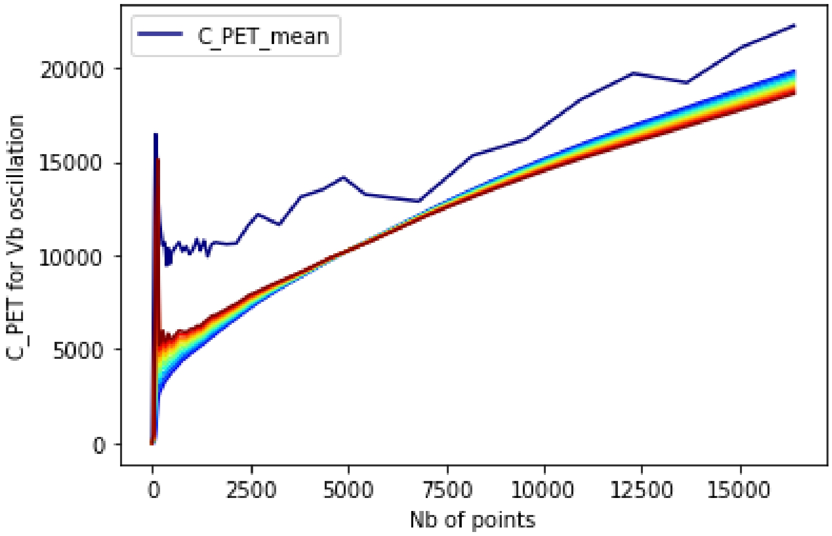


Figure 9: Variations of V_b by setting the other parameters

This clearly give an idea of the functionality of each parameters, the V_b one is clearly a driven to reproduce the initial spike , the other parameter lead to some parallel shift of the exponential shape post spike.

Clearly if we would have a chance to reproduce any initial spike, the optimization procedure must leverage particularly on this V_b.

NON LINEAR LEAST SQUARE PROBLEM SOLVING

Let consider the Energy function of the free parameters vector $u = [k_1, k_2, k_3, V_b]$ the optimization problem is to find (if it exist the solution of this problem):

$$\begin{cases} J(u) = \frac{1}{2}\int_{t_0}^{t_f} |C_{pet\,u}(t) - \overline{C}_{pet}(t)|^2 dt \\ u^* = arg\ min_{u\in\mathbb{R}^4} J(u). \end{cases} \qquad (10)$$

In our case the C_{pet} function is obtained thanks to the numerical resolution of an ODE on the set of points $t_0, t_1, \dots, t_{N_t - 1}$, so we want to minimize:

$$J(u) = \frac{\Delta t}{2}\sum_{i=0}^{N_t-1} (C_{pet\,u}(t_i) - \overline{C}_{pet}(t_i))^2 \quad (11)$$

Clearly the function J is convex, continuous, coercive so inf-compact on a finite dimension space, so there exist a solution u^* to this problem.
We will them use a Gauss-Newton algorithm to find a such root.
To do so, we compute the integral using a Euler scheme, and then the problem is mapped to the following optimization problem:

$$u^* = arg\ min_{u\mathbb{R}^4} \frac{\Delta t}{2} \parallel F(u) \parallel_2^2 \qquad (12)$$

Where the function $F: \mathbb{R}^4 \longrightarrow \mathbb{R}^{N_t}$

$$F(u) = (C_{pet,u}(t_0) - \overline{C}_{pet}(t_0), \dots, C_{pet,u}(t_{N-1}) - \overline{C}_{pet}(t_{N-1})). \quad (13)$$

We compute the gradient using the Gauss-Newton method that is based on the residual norm reduction. Let build the iterative matrix and vectors:

$$A_k = (\nabla F(u^k))^T \nabla F(u^k), \quad (14)$$
$$b_k = -(\nabla F(u^k))^T F(u^k), \quad (15)$$
$$res_k = A_k^{-1} b_k = u^{k+1} - u^k. (16)$$

These relation are obtained from the Normal equation, the iterative matrix suite A_k is non singular symmetric positive. We decide to limit our self to 10 iterations as we converge quiet quickly. We obtained $u^* = [0.00257289, 0.00690342, 0.00191878, 0.10377291]$ This set of parameter tends to be a good first fit of the signal concentration.

AUGMENTED LAGRANGIAN AND ADJOINT PROBLEM–

In this section , we want to add some constraint to solve our optimization problem, we then decide to introduce some penalization in aim to add constraint in our root finding.

We can solve this problem using the penalized minimization Lagrange problem:

$$\begin{cases} J(u) = \int_{t_0}^{t_f} |C_{pet\,u} - \overline{C}_{pet}|^2 w(t) dt + \frac{\lambda}{2} |u - u_{ap}|^2. \\ \inf_u J(u). \end{cases} \quad (17)$$

where $w(t)$ is a function that allows us to modulate the weight we give to each value drawn from the $\overline{C}_{pet}$ data: If we consider that all the values have the same importance we can just take $w(t) = 1$. The The second term in $J(u)$ is interesting when we have an idea of the values of the parameters to optimize. It allows us to stay close to these values.

In order to be able to minimize $J(u)$ we introduce the Lagrangian:

$$L(y, u, p) = \int_{t_0}^{t_f} |C_{pet\,u} - \overline{C}_{pet}|^2 w(t) dt + \frac{\lambda}{2} |u - u_{ap}|^2 + \int_{t_0}^{t_f} p(t)^T (y'(t) - f(y(t), u)) dt. (18)$$

Where

$$y(t) = (C_{free}(t), C_{bound}(t)$$

and

$$C(y(t),u) = C_{pet\,u}(t) = (1 - V_b)(C_{free}(t) - C_{bound}(t)) + V_b\overline{C}_{plasma}(t).$$

Thus

$$J(u) = L(y_u, u, p)$$

for all p and y_u solution of the ODE system solved with parameters u.

Our optimization problem is in fact an optimal control problem with u named control variable, to solve the Lagrange problem, we will use a gradient method.

The idea is build by recurrence a suite $u^k \in U_{ad}$ that would approach u^*

Here is the routine to follow :

- Given u_0, we will choose as first guess the vector used in Part 3.
- We build by recurrence $u^{k+1} = u^k - \rho\nabla J(u^k)$.
- We provide a convergence test of the residual and a stop if the if the maximum number of iterations is reached (we did not compute it as the also tends to converge quickly).

For our problem:

Step 1: Choosing u_0 and u_{ap} we do have some idea as we discovered previously that the most sensitive variable is V_b as this variable allows to reproduce the initial signal spike. We choose u_{ap} equal to our previous finding in Gauss-Newton and we choose a penalty parameter λ that would allow to shift not so strongly on the V_b direction.

In our case the functional is

$$g(y,u,t) = \frac{1}{2}|C_{pet,u}(t) - \overline{C}_{pet}(t)|^2 w(t).$$

Step 2: Solving the Cauchy problem i.e find a numerical solution of this iterative ODE system:

$$\begin{cases} y'_k = f(t, y_k, u_k) \\ y_k(0) = y_0. \end{cases} \qquad (19)$$

This problem does have a unique solution as the function f is Lipschitz. Then, solve numerically the adjoint ODE using a backward retrograde ODE Euler scheme:

$$\begin{cases} -p'_k = D_y(f(y(t),u))^* p(t) - w(t)(C(y(t),u) - C_{pet}(t))D_y(C(y(t),u)) \\ p(T) = 0 \end{cases} \quad (20)$$

Let $\overline{p}$ be the solution of the adjoint problem then, let $u \in \mathbb{R}^4$, the gradient is:

$$D_u J(u) = D_u L(y,u,p) = \int_0^T w(t)(C(y(t),u) - \overline{C}_{pet}) \cdot D_u C(y(t),u)\, dudt + \lambda(u - u_{ap}) \cdot du - \int_0^T D_u(f(y(t),u))^* p(t) \cdot dudt.$$

Therefore,

$$D_u J(u) = \int_0^T (w(t)(C(y(t),u) - \overline{C}_{pet}(t)))D_u C(y(t),u)^* - D_u(f(y(t),u))^* \overline{p}(t))$$

$$\nabla_y f(t, y_k, u_k) = \begin{bmatrix} -(k_2 + k_3) & 0 \\ k_3 & 0. \end{bmatrix} \quad (21)$$

and

$$\nabla_y g(t, y_k, u_k) = \begin{bmatrix} 1 - V_b \\ 1 - V_b \end{bmatrix}^T (C_{pet,u}(t) - \overline{C}_{pet}(t))w(t). \quad (22)$$

Leading to the following result for the derivative of the Hamiltonian:

$$\begin{bmatrix} -(k_2 + k_3)p_k^{(1)} + k_3 p_k^{(2)} - (C_{pet,u} - \overline{C}_{pet})w(1 - V_b) \\ \\ -(C_{pet,u} - \overline{C}_{pet}(t))w(t)(1 - V_b) \end{bmatrix}^T \quad (23)$$

To solve numerically the ODE proposed by the adjoint we use the Backward Euler scheme:

$$\begin{cases} p^N = p(T) \\ p^{i-1} = p^i - \Delta t\, h(p^i, i\Delta t), \quad i = N, N-1, \dots, 1. \end{cases} \qquad (24)$$

Leading to the following equation for the differential inserting $\overline{p}$ solution of our adjoint retrograde ODE

$$D_uJ(u) \cdot \delta u = \lambda(u - u_{ap}) \cdot \delta u + \int_{t_0}^{t_f} [(C_{pet,u} - \overline{C}_{pet})\, w\, D_uC - \overline{p}\, D_uf] \cdot \delta u\, dt \ (25)$$

with

$$D_uC = [0, 0, 0, -(C_{free} + C_{bound}) + \overline{C}_{plasma}] \qquad (26)$$

and

$$D_uf = \begin{bmatrix} \overline{C}_{plasma} & -C_{free} & -C_{free} & 0 \\ 0 & 0 & -C_{free} & 0 \end{bmatrix}. \qquad (27)$$

So to resume up our **Step 2**:

- Solve our Cauchy problem for y_k.
- Solve our Cauchy problem for p_k.
- Let set $D_uJ(u^k)$.

Step 3: Test if $D_uJ(u^k) = 0$ if so we stop with succeed and u^k is the solution that we are looking for. If this is not the case we continue looping using the relationship:

$$u^{k+1} = u^k - \rho D_uJ(u^k). \qquad (28)$$

Step 4: When the residual is small enough, we stop with succeed else we set $k = k + 1$ and we go back to the **Step 2**. After this process we have the following outputs:

- The suite u^k.
- The suite of gradient $D_uJ(u^k)$.
- The residual norm suite $\| res_k \|$.

- The suite of energy $J(u^k)$.

At this stage let's talk a bit on the weight function chosen and the penalty parameter λ.

We remind that we want to reproduce the initial spike so we need to have some control on the last parameter of our u vector. We also want the Energy function J to be convex, so we chosen a particular function that tends to offset as much as possible our gradient $D_u J$.

So we want that $\int_{t_0}^{t_f} (C_{pet,u}u(t) - \overline{C}_{pet}(t))\, w(t)\, (1 - V_b)\, dt$ must be under control within the initial spike divergence.

We then set a kind of Gauss-Hermite weight function $w(t) = \exp(-t^2)$, we remind that the energy is associated to a weighted least square problem, the parameter estimation is strongly dependent of the weight choice. The weight function is useful to control the local error and particularly adapted within our spike issue. Power decay are also good candidates.

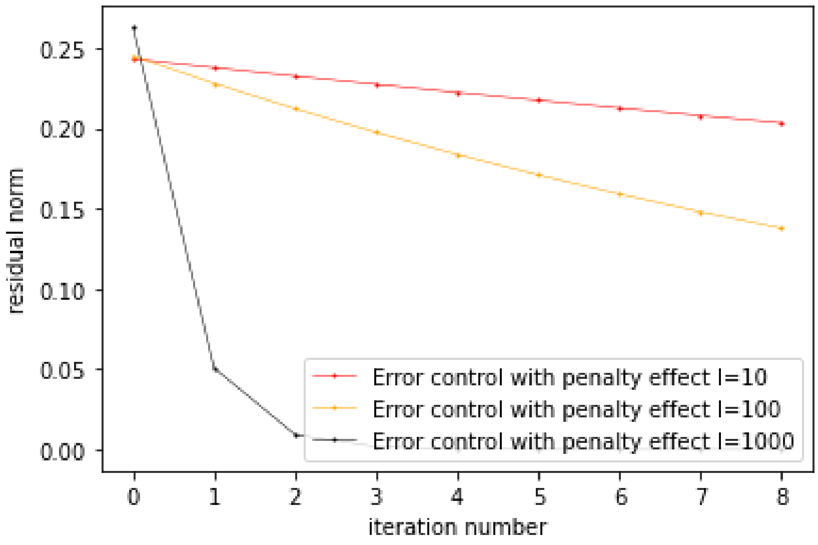


Figure 10: Curves $\| res_k \|$ vs. iteration numbers for three different penalties

The convergence is quiet fast, around 10 iterations are enough but the price to pay is that the total energy is quiet high, so we reached a local minimum for the energy function quickly but this is not a absolute minimum.

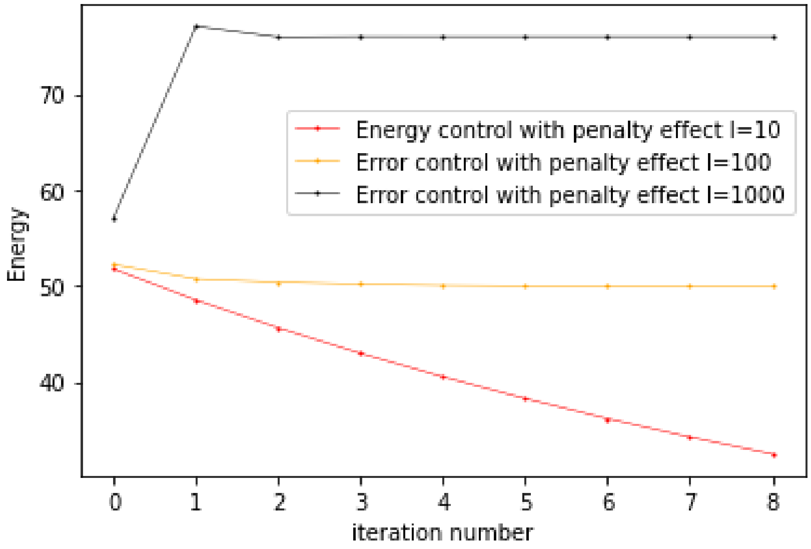


Figure 11: Curves Energy vs. iteration numbers for three different penalties

We are now facing a problem as we reach fast a local minimum for strong value of penalty but our energy function is quiet high, this fact is simply coming from our assumption of taking a fixed value of $\rho = 10^{-3}$.

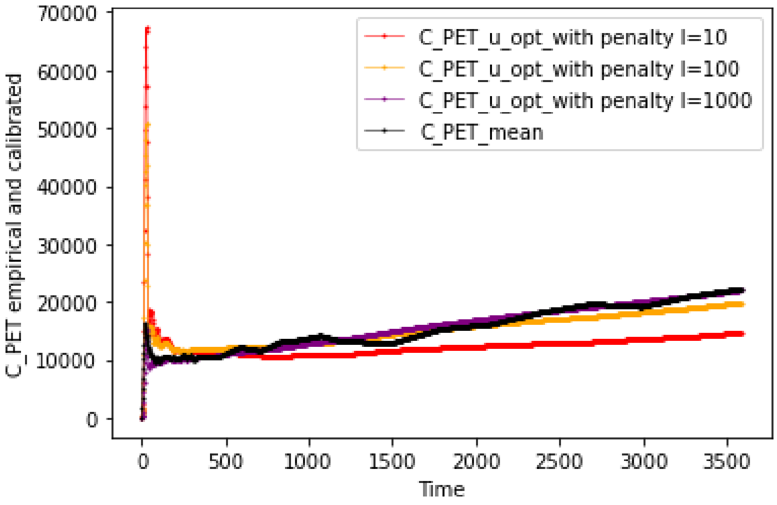


Figure 12: C_{pet} mean adjusted using the augmented Lagrangian method for three different penalties

Instead of using the Cauchy rule to optimize the gradient step parameter, we could decide to change our target vector u_{ap} in aim to reproduce the spike, we remark that there was a trade off between choosing the gradient step and the control of penalty. We decided to set u_{ap} with optimum parameter for all component except for the last one that is the initial spike driver.

As we have chosen weight function that was quiet flatting all the first parameter, the direction of our gradient is carried by the last component, this is why we decided to put strong penalty to have the possibility to explore not far from our V_b parameter intuition.

Choosing $u_{ap} = [0.00257289, 0.00690342, 0.00191878, 0.02]$ leads to a particular good fit within less than 10 iterations.

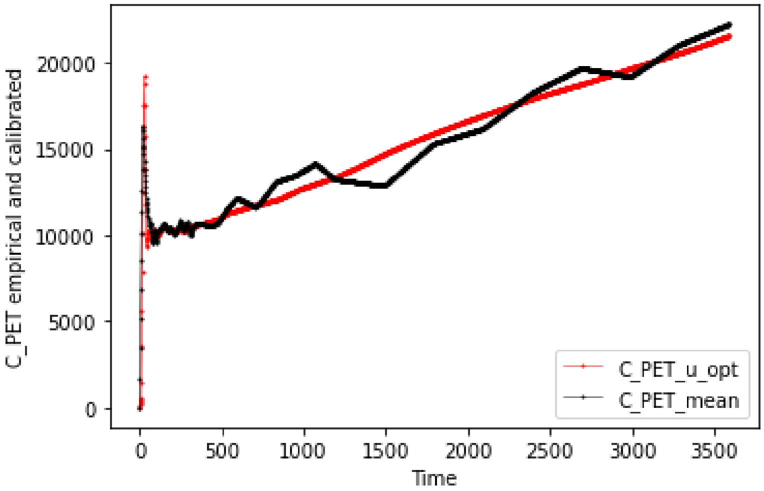
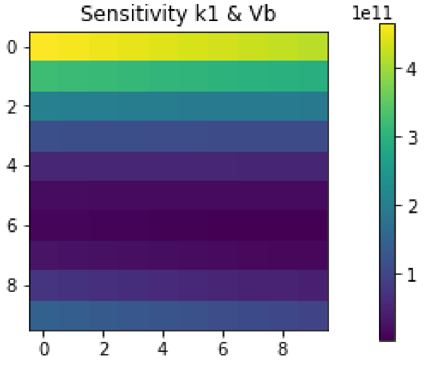


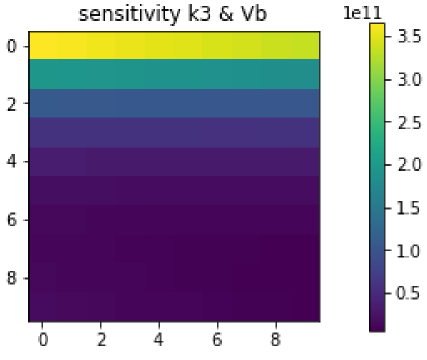
Figure 13: C_{pet} calibrated using the augmented Lagrangian method

Energy is still high due to the high value of the penalty trade off.

This solution obtained seems to be in line with our 2D energy plot:



[Sensitivity of k_1 and V_b]



[Sensitivity of k_3 and V_b]

Vertical axis is the Energy variation for V_b and horizontal axis is for k_1 then k_2, k_3, these maps report us that minimal energy is expected close to 5 units for V_b (1 unit = 0.01), and quiet independently from the other parameters. Our u_{ap} choice has been done in respect of these observations as we do have a idea where we need to look for in term of V_b area.

We can conclude that the leading parameter is the V_b as this is the spike driver and calibration could be quiet independently from the others. Penalized problem leads us to a trade off convexity vs sparcity.

Discussions

As already stated before, we applied Non Linear Square Problem and the Augmented Lagrangian method from which we got several values that can be used in the analysis and prediction, them being the transmembrane passage rates k_1, k_2, the cytoplasmic phosphorylation rate k_3 and the blood volume fraction V_b. These values are different for Non Linear Least Square and Augmented Lagrangian Method and are given in the table below:

Table 1: Non Linear Least Square VS Augmented Lagrangian

$(k_1, k_2, k_3 AND\ V_b)$				
	k_1	k_2	k_3	V_b
Non Linear Least Square	0.00257289	0.00690342	0.00191878	0.10377291
Augmented Lagrangian	0.00257289	0.00690342	0.00191878	0.02

Conclusions

Nonlinear least squares and augmented Lagrangian techniques were used. After having applied both least squares and augmented Lagrangian, we observed that augmented Lagrangian performed better than the nonlinear least squares technique in terms of prediction efficiency and accuracy.

A significant analysis was performed with the help of the penalty control error and the residual norm. In this project, we tried to show the best method for the prediction of free tracer concentration.

Another thing we observed is that the blood volume fraction V_b is very sensitive to changes. The accuracy we have obtained is creditable if we take into account that the augmented Lagrangian method allows the sensitivity of the parameters to be controlled by weight functions. It should also be noted that all the variables in the data set are closely related to the C_{pet} variable.

Acknowledgement

I would like to thank my colleague Jonathan Lévy, who guided me through the project, on how to apply the Quadratic Nonlinear Problem and the Augmented Lagrangian Method, even during method implementation.

Referencias

1. ALLAIRE., (2007). "Numerical analysis and optimization: an introduction to mathematical modelling and numerical
2. simulation." OUP Oxford.
3. BONNANS, GILBERT, LEMARÉCHAL, SAGASTIZÁBAL., (2006) "Numerical
4. optimization: theoretical and practical aspects." Springer Science & Business Media.
5. FORTIN, GLOWINSKI., (2000). "Augmented Lagrangian methods: applications to the numerical solution of boundary-value problems." Elservier.
6. SOKOLOFF, REIVIC, KENNEDY, DES ROSIERS, PATLAK, PETTIGREW, SAKURADA, SHINOHARA., (1977). "The [14c] deoxyglucose method for the measurement of local cerebral glucose utilization: theory, procedure, and normal values in the conscious and anesthetized albino rat 1." Journal of neurochemistry, 28(5):897–916.
7. SUN, YUAN., (2006). "Optimization theory and methods: nonlinear programming." volume 1. Springer Science & Business Media.
8. VRIENS, VISSER, DE GEUS-OEI, OYEN., (2010). "Methodological considerations in quantification of oncological fdg pet studies." European journal of nuclear medicine and molecular imaging, 37(7):1408–1425.