Kinetic Model Fitting with Forced Convexification

László Szirmay-Kalos

***Abstract*—During dynamic tomography reconstruction, kinetic models, e.g. the two-tissue compartment model, should be fitted to the reconstructed activity values in each voxel and frame, which is equivalent to the solution of a huge number of general optimization problems. Newton's and Levenberg's methods have quadratic convergence speed close to the optimum, but they may get stuck in local extrema, and their convergence is not guaranteed at all. The failure of convergence of the voxel time activity functions may slow down the convergence of the global reconstruction. To solve this, we propose the dynamic transformation of the target function to shrink those regions of the domain where Newton's method would not go into the direction of the optimum. We specify the conditions of such transformations and provide concrete solutions. We also make the transformation scale or time dependent to allow its gradual elimination when it is not needed anymore.**

I. Introduction

Direct dynamic emission tomography [1], [2], [3] can be used to examine the accumulation of drugs in tissues described by compartment models [4], [5], which express the expected number of decays $\tilde{x}_F(\mathbf{p}_V)$ in *time-frame* F as a non-linear function of unknown parameters $\mathbf{p}_V$ of *voxel* V.

According to the concept of *maximum-likelihood reconstruction* [6], unknown parameters are found to minimize the negative log-likelihood of the current measurement $y_{L,F}$, i.e. the number of detected hits in *LOR* L in all frames F [5], [7], [8], [9]. Such optimization problems can be attacked by stochastic methods that can find global optima, but they are typically slow and less accurate compared to techniques taking advantage of local features, like the gradient or the Hessian of the optimization target. Gradient or Hessian based methods, like the Newton-Raphson or the Levenberg-Marquardt approach, get closer to the minimum if the function is convex and can be well approximated by a paraboloid in the neighborhood of the current estimate. However, they cannot always guarantee that they approach the minimum or converge at all.

In this paper, we investigate this issue, propose a solution and apply it to the compartmental model fitting problem of dynamic tomography. Our proposed transformation also depends on a time or scale parameter that controls the strength of the transformation and allows its gradual removal. Scale dependent modification of the optimization target also showed up in differential equation based image processing algorithms, which inspired our method [10], [11], [12], [13].

First, we consider a one-dimensional motivating example, then the results are generalized to higher dimensions.

Manuscript received December 21, 2019. This work has been supported by OTKA K–124124 and EFOP 4.2.1-16-2017-00021 project together with the European Social Fund (EFOP-3.6.2-16-2017-00013).

L. Szirmay-Kalos is with the Budapest University of Technology and Economics, Department of Control Engineering and Information Technology, Hungary (e-mail: szirmay@iit.bme.hu).

II. Problem statement in one-dimension

Let us consider a one-variate scalar function $f(p)$. We wish to determine the location p^* of its minimum in interval $[p_{\min}, p_{\max}]$. The function has an extremum where its derivative is zero and this extremum is a minimum if the function around the extremum is convex, i.e. its second derivative is positive:

$$f'(p^*) = 0, \quad f''(p^*) > 0.$$

Newton-Raphson iteration approximates optimization target $f(p)$ by a second order parabolic function $\hat{f}(p)$ around current estimate p_n,

$$f(p) \approx \hat{f}(p) = f(p_n) + f'(p_n)(p - p_n) + \frac{f''(p_n)}{2}(p - p_n)^2,$$

and produces the increment d_n between the new guess p_{n+1} and the current guess p_n by finding the extremum of this parabolic approximation

$$\hat{f}'(p_n + d_n) = 0 \implies d_n = -\frac{f'(p_n)}{f''(p_n)}.$$

The parabolic function $\hat{f}(p)$ has a minimum and is convex if $f''(p_n) > 0$ and has a peak and is concave if $f''(p_n) < 0$. Thus, a single iteration step increases $\hat{f}$ if $f''(p_n) < 0$, i.e. when $\hat{f}(p)$ is concave. As we are searching a minimum, this is a wrong direction to go if $\hat{f}$ is sufficiently good approximation of the original function f. Generally, Newton-Raphson iteration likely decreases the function and moves towards the minimum where f is convex, and possibly increases the function and moves away from the minimum where f is non-convex. Thus, regions where f is non-convex are "evil" regions which should be avoided to get a continuously converging iteration.

The method of this paper is based on the recognition that the probability of entering such evil regions can be reduced if these are shrunk. Shrinking regions where f is non-convex requires the modification of the optimization target f. Let us use function combination and investigate the same optimization process for a new function $(s \circ f)(p) = s(f(p))$, where *transformer function* s transforms the values of f. Let us examine the derivatives of the composed function:

$$(s \circ f)' = s' f', \quad (s \circ f)'' = s''(f')^2 + s' f''. \tag{1}$$

If $s > 0$ and $s' \neq 0$, i.e. s is positive and monotone, then the transformed function has the same minima as the original function, thus the location of the optimum is not changed.

We aim at shrinking regions where $s \circ f$ is concave, i.e. where $(s \circ f)'' = s''(f')^2 + s' f'' < 0$, and expand regions where $s \circ f$ is convex by the appropriate selection of s. On the one hand, we should guarantee that where f'' is positive, $(s \circ f)''$ is also positive because we wish to preserve the already convex regions. Examining Eq. 1, $(s \circ f)''$ is surely positive for

978-1-7281-4164-0/19/$31.00

positive f'' if both s' and s'' are positive. On the other hand, shrinking happens when there are points where f'' is negative, but $(s \circ f)''$ becomes positive, which requires s'' be positive. Summarizing,

$$s > 0, \quad s' > 0, \quad s'' > 0, \tag{2}$$

i.e. s is an increasing and convex function. It is enough to guarantee these properties in search domain $[p_{\min}, p_{\max}]$, which contains only non-negative numbers in our target applications.

Let us consider the Newton-Raphson iteration scheme for $(s \circ f)$ instead of f. The increment is

$$d_n = -\frac{(s \circ f)'(p_n)}{(s \circ f)''(p_n)} = -\frac{f'(p_n)}{f''(p_n) + \frac{s''(f)}{s'(f)}(f'(p_n))^2}.$$

In addition to Eq. 2, we can impose further requirements. For example, if we multiply target function f by a positive number, its shape and location of minima do not change, thus there is no reason to modify the increment. This imposes another requirement on transformer function s. The increment should be *independent of multiplying f by a constant*, which happens when $s''(f)/s'(f)$ is inversely proportional to the scaler applied to f. This option leads to the following transformer s:

$$\frac{s''(f)}{s'(f)} = \frac{t}{f} \quad \Longrightarrow \quad s'(f) = f^t \tag{3}$$

where t is a *scale parameter*. Considering that s must be an increasing and convex function, t should be greater than 0. With this choice, the increment is

$$d_n = -\frac{f'(p_n)}{f''(p_n) + \frac{t}{f(p_n)}(f'(p_n))^2}.$$

On the other hand, the increment may be chosen to be *independent of constant addition*, which can be satisfied by making s''/s' constant since in other factors only the derivative of f shows up, which is not modified by a constant addition. The resulting scheme is

$$\frac{s''(f)}{s'(f)} = t \quad \Longrightarrow \quad s'(f) = e^{tf}, \tag{4}$$

With this choice, the increment becomes

$$d_n = -\frac{f'(p_n)}{f''(p_n) + t(f'(p_n))^2}.$$

We use the scaling independent method (Eq. 3) in our experiments, the two approaches do not differ much under typical circumstances.

A. Demonstration with an analytic function

In order to demonstrate the transformation and its effect, we take a simple example function

$$f(p) = \cos(p) + \frac{p}{3} + 1$$

in the domain of $[0, 10]$. This function and its first and second derivatives are shown by Fig. 1. The function has three local minima in the specified domain, at the left boundary $p = 0$, at $p = 2.8$, which is also the global minimum, and at $p = 9.1$, which is a local, but not global minimum. Other extrema are maxima, including $p = 0.3$, $p = 6.4$, and at the right boundary $p = 10$. Newton-Raphson iteration is started at a random value in the $[1, 6]$ interval, which is in between the two local maxima, thus optimization should find the global minimum if it continuously approaching it. However, Newton-Raphson iteration does not guarantee such continuous approach toward the optimum, it may get farther away, may reach points of higher target value, and may even leave the domain. When iteration leaves the $[0, 10]$ domain or the target function value increases, the current position is randomly re-selected from full domain [0,10].

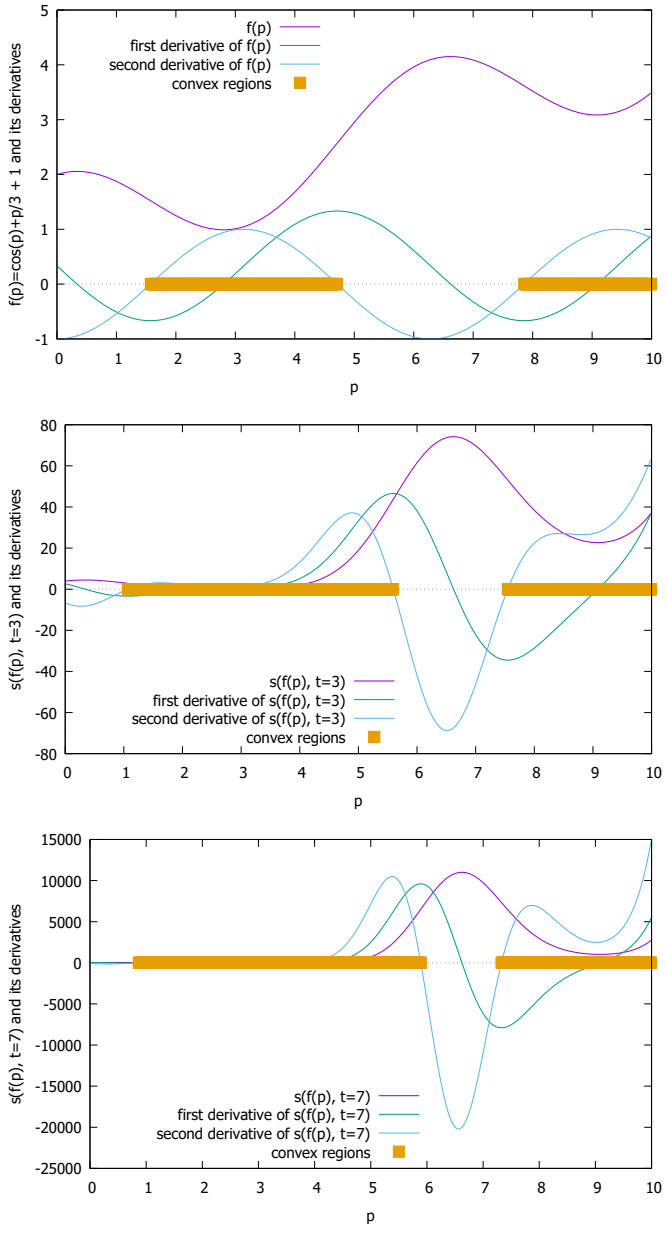


Fig. 1. Example function $f(p) = \cos(p) + \frac{p}{3} + 1$ for minimum search and its transformed versions $s(f(p), t)$ with $t = 3$ and $t = 7$, respectively. The optimum is at $p = 2.8$ in all cases. We also depicted the convex regions with yellow stripes where the Newton-Raphson method is safely converging.

We repeated the same experiment with two versions of

transformer function $s'(f) = f^t$, when scale parameter t is set to 3 and 7, respectively. The effect of the target function transformation is shown by the lower plots of Fig. 1. Note the increase of the convex regions as t gets larger values.

The errors of the target function at the current location and at the optimum point are computed in different iteration steps, and these error estimates are averaged for a large, i.e. 10^4 number of runs of the randomly initialized optimization. The error curve is shown by Fig. 2. As a larger scale parameter means stronger shrinking of the concave regions, the probability of going into a wrong direction is smaller, and thus the probability of converging to a non-global minimum also gets smaller if the initial guess is in the attraction zone of the global minimum. Converging to non-global minimum prohibits the error from further reduction after the first 5-10 iteration steps. From this point of view, it is worth selecting a large scale parameter t, because it shrinks concave regions stronger. However, large scale parameters may have disadvantages as well. Recall that Newton's method finds the next guess using a local quadratic approximation, which means that the method converges quickly if the quadratic approximation is accurate and higher order derivatives can be ignored. By increasing t, higher order derivatives can be amplified and increment $|d_n|$ is reduced when the function is convex, thus the convergence slows down in safe regions. This is why the error of $t = 9$ is larger during the first few iteration steps and becomes better only later on.

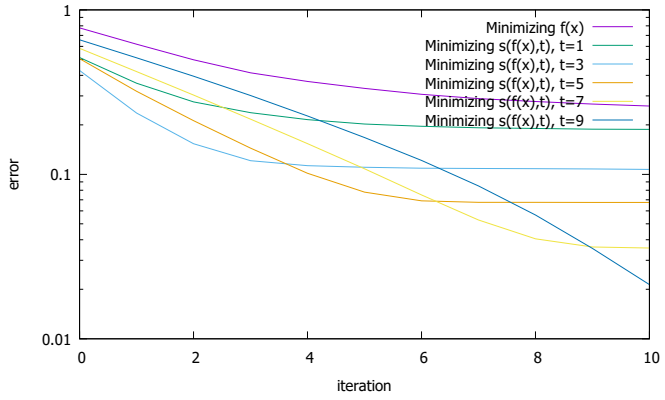


Fig. 2. Error of the Newton-Raphson iteration applied to f and $s(f(p), t)$.

The disadvantage of the initial iteration speed can be reduced by the adaptive control of scale parameter t. We start with an initial value $t = t_0$, and t is halved when the error gets smaller and thus the proposal is accepted, and doubled in case of the error gets larger (Fig. 3).

III. Multi-variate case

The proposed method can be generalized to multi-variate targets $f(\mathbf{p})$ where variable $\mathbf{p}$ is a vector of multiple elements. In higher dimensions, at point $\mathbf{p}$ the target function has an extremum if its gradient is zero. The extremum is a minimum if the Hessian is positive definite and a maximum if it is negative definite. If the Hessian is neither positive definite nor negative definite at a zero gradient point, then it is a saddle point.

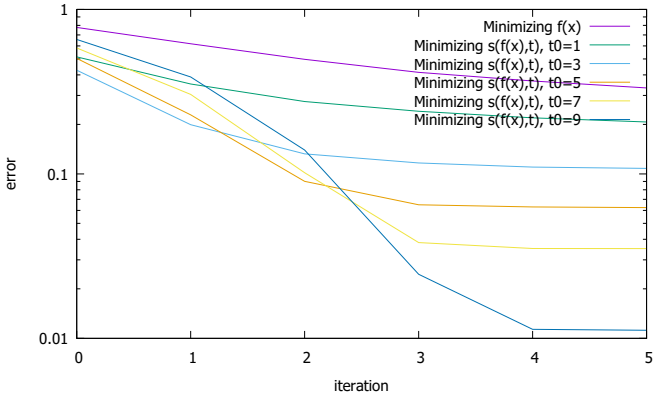


Fig. 3. Error of the Newton-Raphson iteration applied to f and $s(f(p), t)$. Scale parameter t starts at t_0 and is doubled when the next proposal is worse and halved when the next proposal has a smaller error.

Let us consider the second order approximation $\hat{f}$ of f around current estimate $\mathbf{p}^{(n)}$

$$f(\mathbf{p}) \approx \hat{f}(\mathbf{p}) = f(\mathbf{p}^{(n)}) + \mathbf{d}^T \cdot \mathbf{g} + \frac{1}{2}\mathbf{d}^T \cdot \mathbf{H} \cdot \mathbf{d},$$

where $\mathbf{d} = \mathbf{p} - \mathbf{p}^{(n)}$ is the increment, vector $\mathbf{g} = \nabla f$ is the gradient, and matrix $\mathbf{H}$ is the Hessian at $\mathbf{p}^{(n)}$:

$$\mathbf{g}_P = \frac{\partial f}{\partial \mathbf{p}_P}(\mathbf{p}^{(n)}), \quad \mathbf{H}_{PQ} = \frac{\partial^2 f}{\partial \mathbf{p}_P \partial \mathbf{p}_Q}(\mathbf{p}^{(n)}).$$

In the extremum of approximation $\hat{f}$, its gradient must be zero:

$$\nabla \hat{f}(\mathbf{p}) = \mathbf{g} + \mathbf{H} \cdot \mathbf{d} = 0.$$

Thus, to obtain increment $\mathbf{d}$, the following system of linear equations must be solved:

$$\mathbf{H} \cdot \mathbf{d} = -\mathbf{g}.$$

Executing the same steps for $s(f(\mathbf{p}))$, we obtain a slightly modified system of linear equations for increment $\mathbf{d}$:

$$\left(\mathbf{H} + \frac{s''(f)}{s'(f)}\mathbf{g} \cdot \mathbf{g}^T\right) \cdot \mathbf{d} = -\mathbf{g}.$$

Substituting the particular form of the scaling independent transformer $s'(f) = f^t$, we get

$$\left(\mathbf{H} + \frac{t}{f}\mathbf{g} \cdot \mathbf{g}^T\right) \cdot \mathbf{d} = -\mathbf{g}. \tag{5}$$

Note that $\frac{t}{f}\mathbf{g} \cdot \mathbf{g}^T$ is a positive definite matrix if f and t are positive and $\mathbf{g}$ is not zero. Let us examine the effect of the matrix modification on the Newton-Raphson iteration. This iteration has a fixed point where the increment is zero, i.e. $\mathbf{d} = 0$. In such cases gradient $\mathbf{g}$ is also zero, thus the modified iteration can converge only to the extrema of f similarly to the original iteration. On the other hand, when t is large, the added matrix term dominates the original Hessian. This added term is a positive definite matrix with the exception of points where the gradient and consequently the added term are zero. Thus, we can convexify the target function almost everywhere except for the original extrema and their arbitrary small neighborhood.

IV. Application in direct dynamic PET reconstruction

Dynamic tomography algorithms usually iterate two steps, the first executes forward and back projections in each frame, the second fits parameters $\mathbf{p}_V$ of the radiotracer concentration function of pre-defined form $\mathcal{K}(\mathbf{p}_V, t)$ in each voxel V. Thus, the number of fitting algorithm executions is equal to the product of the number of ML-EM iterations and voxels. Considering *compartmental models*, the algebraic form reflects the solution of the differential equation of radiotracer exchange between compartments [14], [15].

For example, a possible algebraic form for the two-tissue compartment model is:

$$\mathcal{K}(\mathbf{p}, t) = f_v C_w(t) + \left(\epsilon(t) \sum_{i=1}^{n} a_i e^{-\alpha_i t}\right) * C_p(t), \quad (6)$$

where $f_v \in [0, 1]$ is the unknown *fraction of blood* and C_W is the known or separately measured *total blood concentration function*, $\epsilon(t)$ is the Heaviside step function, and kinetic parameters K_1, k_2, k_3, k_4 and macro parameters, including *Uptake rate* K_i and *Volume distribution* V_D, can also be determined from fitted parameters a_1, a_2, α_1, α_2:

$$K_1 = \frac{a_1 + a_2}{1 - f_v}, \quad (7)$$

$$k_2 = \frac{a_1\alpha_1 + a_2\alpha_2}{a_1 + a_2}, \quad (8)$$

$$k_3 = \frac{a_1 a_2 (\alpha_1 - \alpha_2)^2}{(a_1 + a_2)(a_1\alpha_1 + a_2\alpha_2)}, \quad (9)$$

$$k_4 = \frac{\alpha_1\alpha_2}{k_2}, \quad (10)$$

$$K_i = \frac{K_1 k_3}{k_2 + k_3}, \quad (11)$$

$$V_D = \frac{K_1}{k_2}\left(1 + \frac{k_3}{k_4}\right). \quad (12)$$

Radiotracer isotopes randomly decay generating events in the detectors of the tomograph. An event is identified with the pair of involved detectors, called *LOR*, and the detection time. The measurement time is partitioned into intervals (t_F, t_{F+1}) called *frames*.

Using concentration function $\mathcal{K}(\mathbf{p}_V, t)$ for voxel V represented by parameter vector $\mathbf{p}_V$, the expected number of decays of this voxel in frame F is

$$\tilde{x}_F(\mathbf{p}_V) = \int_{t_F}^{t_{F+1}} \mathcal{K}(\mathbf{p}_V, t) e^{-\lambda t} \mathrm{d}t. \quad (13)$$

Parameter λ defines the decay rate of the radiotracer.

We use the maximum-likelihood estimator to obtain unknown voxel parameters $\mathbf{p}_V$, i.e. determine that parameter set which maximizes the likelihood of the measured events. This leads to the following non-linear equations for every voxel [5], [8]:

$$f_P(\mathbf{p}_V) = \sum_F \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_{V,P}} \left(\frac{x_{V,F}}{\tilde{x}_F(\mathbf{p}_V)} - 1\right) = 0. \quad (14)$$

This equation contains the expected number of decays $\tilde{x}_F$ in frame F of voxel V as defined by Eq. (13), its derivatives with respect to the unknown voxel parameters, and its forward and back projection with system matrix $\mathbf{A}_{L,V}$:

$$x_{V,F} = \tilde{x}_F(\mathbf{p}_V) \cdot \frac{\sum_L \mathbf{A}_{L,V} \frac{y_{L,F}}{\sum_{V'} \mathbf{A}_{L,V'} \tilde{x}_F(\mathbf{p}_{V'})}}{\sum_L \mathbf{A}_{L,V}}.$$

Note that $x_{V,F}$ depends on the parameter vectors of all voxels. Taking $x_{V,F}$ from the previous iteration [5], [8], which is similar to the one-step-late strategy, equations of different voxels become independent and we solve a curve fitting separately in each voxel. Because of this, we omit subscript V in the equations from now on.

We can realize that the non-linear equations (Eq. (14)) for different elements of the parameter vector form the gradient of the following scalar function, which looks like a Kullback-Leibler divergence:

$$\frac{\partial \tilde{x}_F}{\partial \mathbf{p}_P}\left(\frac{x_F}{\tilde{x}_F(\mathbf{p})} - 1\right) = \frac{\partial}{\partial \mathbf{p}_P}\left(\tilde{x}_F(\mathbf{p}) - x_F \log(\tilde{x}_F(\mathbf{p}))\right).$$

The drawback of this option is that the Kullback-Leibler divergence is not a distance measure, and it does not give zero for a perfect fit, but can be either positive or negative. Thus, we use the difference of the Kullback-Leibler divergences of the current and the perfect fits:

$$f_{KL}(\mathbf{p}) = \sum_F \tilde{x}_F(\mathbf{p}) - x_F \log(\tilde{x}_F(\mathbf{p})) - x_F + x_F \log(x_F). \quad (15)$$

To find the optimum, the Newton-Raphson iteration solves the following linear system for step $\mathbf{d}$:

$$\mathbf{H} \cdot \mathbf{d} = \mathbf{g} \quad (16)$$

where

$$\mathbf{H}_{P,Q} = \sum_F \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_P} \frac{x_F}{\tilde{x}_F^2(\mathbf{p})} \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_Q} - \frac{\partial^2 \tilde{x}_F}{\partial \mathbf{p}_P \partial \mathbf{p}_Q}\left(\frac{x_F}{\tilde{x}_F(\mathbf{p})} - 1\right),$$

$$\mathbf{g}_P = \sum_F \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_P}\left(\frac{x_F}{\tilde{x}_F(\mathbf{p})} - 1\right). \quad (17)$$

To convexify the target function based on Eq. 5, we solve the following modified equation:

$$\left(\mathbf{H} + \frac{t}{f_{KL}} \mathbf{g}^T \cdot \mathbf{g}\right) \cdot \mathbf{d} = \mathbf{g}. \quad (18)$$

In the result section, we use the Levenberg-Marquardt algorithm for comparison, which is a popular choice for fitting. The Levenberg-Marquardt method was proposed for least-square function fitting. However, Eq. (14) is not equivalent to least-square fitting since it would minimize the absolute error between the data points and the fitted function, while Eq. (14) can be interpreted as the minimization of the relative error. Such problems can be attacked by replacing the $1/\tilde{x}_F$ term in Eq. (14) by its Taylor's expansion [8]:

$$\frac{1}{\tilde{x}_F(\mathbf{p}^* + \mathbf{d})} \approx \frac{1}{\tilde{x}_F(\mathbf{p}^*)} + \sum_Q \frac{\partial 1/\tilde{x}_F}{\partial \mathbf{p}_Q} \mathbf{d}_Q \quad (19)$$

$$= \frac{1}{\tilde{x}_F(\mathbf{p}^*)} - \frac{1}{\tilde{x}_F^2(\mathbf{p}^*)} \sum_Q \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_Q} \mathbf{d}_Q$$

where $\mathbf{p}^*$ is the current estimate and $\mathbf{d} = \mathbf{p} - \mathbf{p}^*$ is the offset. With this substitution, we get a linear system of equations for the unknown offset

$$\mathbf{F} \cdot \mathbf{d} = \mathbf{g} \tag{20}$$

where

$$\begin{aligned} \mathbf{F}_{P,Q} &= \sum_F \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_P} \frac{x_F}{\tilde{x}_F^2(\mathbf{p})} \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_Q}, \\ \mathbf{g}_P &= \sum_F \frac{\partial \tilde{x}_F}{\partial \mathbf{p}_P} \left(\frac{x_F}{\tilde{x}_F(\mathbf{p})} - 1 \right). \end{aligned} \tag{21}$$

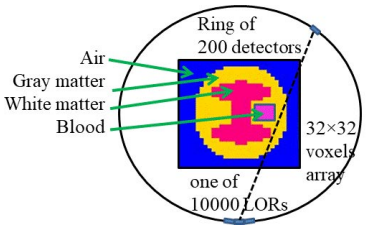
To make the iteration more robust and similar to gradient search when we are farther from the solution, the concept of the Levenberg-Marquardt method can be applied, which emphasizes the diagonal of the matrix with an automatically controlled parameter ν

$$\mathbf{F}^{LM} = \mathbf{F} + \mathrm{diag}(\mathbf{F})\nu$$

where ν is increased if the error gets larger in this subiteration step and decreased otherwise.

V. Results

To examine the proposed methods, we use a brain phantom segmented to white matter, gray matter, blood, and air (Fig. 4).



	k_1	k_2	k_3	k_4	f_v
Air	0	0	0	0	0
Gray	2	1	0.4	0	0
White	1	2	0.3	0	0
Blood	3	0	0.1	0	0

Fig. 4. 2D tomograph model: The detector ring contains 200 detector crystals and each participates in 99 LORs connecting this crystal to crystals being in the opposite half circle. The voxel array to be reconstructed is in the middle of the ring and has 32×32 resolution, i.e. 1024 voxels.

To examine the proposed methods, we use a 2D phantom [16] (Fig. 4) and reconstruct the activity curves of a 2D brain model.

A. Reconstruction of the parameters of Fig. 4

In the first set of experiments, we reconstruct the macroparameter images for the set of microparameters shown in Fig. 4. The total number of hits during the 10 sec long measurement time is 1.1 million. The measurement time is decomposed to $N_F = 30$ uniform frames. We use the two-tissue compartment model for fitting. The blood input function follows the *Feng's model* [17]

$$C_p(t) = \epsilon(t) \left(A_1 t e^{-\beta_1 t} + \sum_{j=2}^{4} A_j \left(e^{-\beta_j t} - e^{-\beta_1 t} \right) \right)$$

with $A_1 = 200, A_2 = 100, A_3 = 50, A_2 = 20, \beta_1 = 1.5, \beta_2 = 0.5, \beta_3 = 0.1, \beta_4 = 1$.

We have executed 10 ML-EM iterations. The proposed method is compared to the original Newton-Raphson iteration and also to the Levenberg-Marquardt method. The reconstructions are depicted by Figure 5 showing the reconstructed macro-parameter images.

B. Random system reconstruction

In the second round of experiments, the measured data are obtained with simulation when kinetic microparameters $k_1, \ldots, k_4$ are selected randomly from the [0.02, 2] interval. The measurement time is decomposed to $N_F = 20$ frames. We have reconstructed 35 random models with 10 iterations using the classical Newton-Raphson iteration, Levenberg's method, and the proposed convexification algorithm. We turned off all regularization methods.

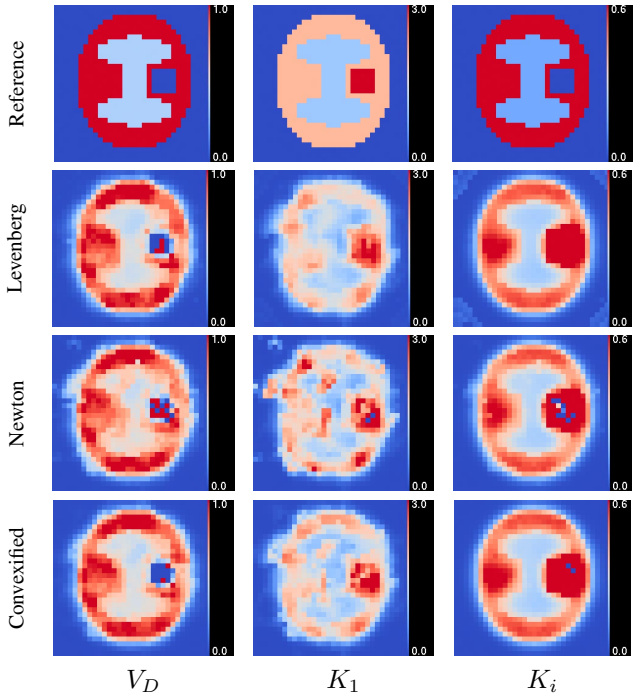


Fig. 5. Reconstructions of kinetic macro parameters Volume distribution V_D, K_1, and K_i.

Table I shows the averages and standard deviations of reconstruction errors for the three compared methods and for different number of frames. The convexified method provides the lowest error values.

Fig. 6 compares the reconstructed and the ground truth macroparameters V_D in white matter and gray matter for the compared three methods. We can observe that the average distance from the ideal line of slope 1 gets smaller, thus the convexified approach provides more robust fitting.

VI. Conclusions

This paper proposed a function transformation to help Newton-Raphson iteration converge to the minimum. The transformation does not modify the locations of the extrema, but shrinks those regions where Newton-Raphson iteration would propose a sample that gets farther from the optimum. We applied the method to emission tomography reconstruction and

TABLE I

RECONSTRUCTION ERRORS OF TEN TOTAL ACTIVITY TESTS RANGING FROM 40 DECAYS PER VOXEL IN AVERAGE TO 400 DECAYS PER VOXEL IN AVERAGE. THE RELATIVE L_2 ERRORS OF THE ACTIVITY CURVES AND THE TOTAL ERROR OF V_D MACROPARAMETER ARE COMPUTED AFTER EACH RECONSTRUCTION AND WE SHOW THE AVERAGE ERRORS TOGETHER WITH THE STANDARD DEVIATIONS.

# of frames	Method	L_2 error	error of V_D
5 frames	Newton	33.9 $\pm$ 2.3	444 $\pm$ 478
	Levenberg	33.2 $\pm$ 3.6	524 $\pm$ 188
	Convexification	32.4 $\pm$ 0.7	238 $\pm$ 17
10 frames	Newton	38.9 $\pm$ 16.5	541 $\pm$ 294
	Levenberg	33.2 $\pm$ 3.5	524 $\pm$ 188
	Convexification	31.1 $\pm$ 0.8	267 $\pm$ 57
40 frames	Newton	46.6 $\pm$ 21.7	680 $\pm$ 383
	Levenberg	33.1 $\pm$ 1.2	333 $\pm$ 96
	Convexification	30.6 $\pm$ 1.2	312 $\pm$ 92

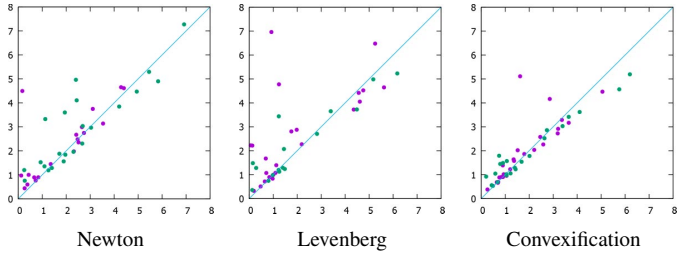


Fig. 6. Reconstructions of V_D kinetic macro parameter. Purple points correspond to white matter reconstructions, green ones to gray matter reconstructions.

have shown results obtained with a 2D phantom. This solution is integrated into the Tera-tomo system.

REFERENCES

[1] A. J. Reader and H. Zaidi, "Advances in PET image reconstruction," *PET Clinics*, vol. 2, no. 2, pp. 173–190, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1556859807000193

[2] A. J. Reader and J. Verhaeghe, "4d image reconstruction for emission tomography," *Physics in Medicine and Biology*, vol. 59, no. 22, p. R371, 2014. [Online]. Available: http://stacks.iop.org/0031-9155/59/i=22/a=R371

[3] L. Szirmay-Kalos and A. Kacsó, "Regularizing direct parametric reconstruction for dynamic pet with the method of sieves," in *Molecular Imaging Conference*, ser. MIC '16, 2016, pp. M16D–1.

[4] M. Kamasak, C. Bouman, E. Morris, and K. Sauer, "Direct reconstruction of kinetic parameter images from dynamic pet data," *IEEE Trans Med Imaging*, vol. 24, no. 5, pp. 636–650, 2005.

[5] G. Wang and J. Qi, "An optimization transfer algorithm for nonlinear parametric image reconstruction from dynamic pet data," *IEEE Trans Med Imaging*, vol. 31, no. 10, pp. 1977–1988, 2012.

[6] L. Shepp and Y. Vardi, "Maximum likelihood reconstruction for emission tomography," *IEEE Trans. Med. Imaging*, vol. 1, pp. 113–122, 1982.

[7] G. Wang and J. Qi, "Direct estimation of kinetic parametric images for dynamic pet," *Theranostics*, vol. 3, no. 10, pp. 802–815, 2013.

[8] L. Szirmay-Kalos, A. Kacsó, M. Magdics, and B. Tóth, "Dynamic pet reconstruction on the gpu," *Periodica Polytechnica Electrical Engineering and Computer Science*, vol. 62, no. 4, pp. 134–143, 2018. [Online]. Available: https://pp.bme.hu/eecs/article/view/11739

[9] M. Magdics and et al., "TeraTomo project: a fully 3D GPU based reconstruction code for exploiting the imaging capability of the NanoPET/CT system," in *World Molecular Imaging Congress*, 2010.

[10] P. Perona and J. Malik, "Scale-space and edge detection using anisotropic diffusion," *IEEE Trans on Image Processing*, vol. 12, pp. 629–639, 1990.

[11] S. Osher and R. P. Fedkiw, "Level set methods: An overview and some recent results," *Journal of Computational Physics*, vol. 169, no. 2, pp. 463 – 502, 2001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0021999100966361

[12] F. Catt, P.-L. Lions, J.-M. Morel, and T. Coll, "Image selective smoothing and edge detection by nonlinear diffusion," *SIAM Journal on Numerical Analysis*, vol. 29, no. 1, pp. pp. 182–193, 1992. [Online]. Available: http://www.jstor.org/stable/2158083

[13] P. Chaudhari, A. Oberman, S. Osher, S. Soatto, and G. Carlier, "Deep relaxation: partial differential equations for optimizing deep neural networks," *Research in the Mathematical Sciences*, vol. 5, no. 3, p. 30, Jun 2018. [Online]. Available: https://doi.org/10.1007/s40687-018-0148-y

[14] H. Watabe, Y. Ikoma, Y. Kimura, M. Naganawa, and M. Shidahara, "Pet kinetic analysis—compartmental model," *Annals of Nuclear Medicine*, vol. 20, no. 9, p. 583, Nov 2006. [Online]. Available: http://dx.doi.org/10.1007/BF02984655

[15] J. Yan, B. Planeta-Wilson, and R. Carson, "Direct 4d list mode parametric reconstruction for pet with a novel em algorithm," in *Nuclear Science Symposium Conference Record, 2008. NSS '08. IEEE*, Oct 2008, pp. 3625–3628.

[16] L. Szirmay-Kalos, M. Magdics, B. Tóth, and T. Bükki, "Averaging and Metropolis iterations for positron emission tomography," *IEEE Trans Med Imaging*, vol. 32, no. 3, pp. 589–600, 2013.

[17] D. Feng, S.-C. Huang, and X. Wang, "Models for computer simulation studies of input functions for tracer kinetic modeling with positron emission tomography," *International Journal of Bio-Medical Computing*, vol. 32, no. 2, pp. 95 – 110, 1993. [Online]. Available: http://www.sciencedirect.com/science/article/pii/002071019390049C

[18] L. Szirmay-Kalos, M. Magdics, and B. Tóth, "Multiple importance sampling for PET," *IEEE Trans Med Imaging*, vol. 33, no. 4, pp. 970–978, 2014.