The CTT MATLAB toolbox

A practical user's guide

Behrouz Mehdinejadiani

Department of Water Science and Engineering, Faculty of Agriculture, University of Kurdistan, Sanandaj, Iran

Correspondence: b.mehdinejad@uok.ac.ir

1. What is the CTT MATLAB toolbox?

The CTT is a collection of MATLAB m-files to inversely estimate parameters of continuous time random walk-truncated power law (CTRW-TPL) model using teaching learning-based optimization (TLBO) algorithm. For detailed information about the CTRW-TPL model and the TLBO algorithm, the reader is referred to relevant literatures (e.g., Dentz et al., 2003; Berkowitz et al., 2006; Cortis and Berkowitz, 2004 and 2005; Rao et al., 2011; Rao, 2019). The CTT runs on MATLAB R2017a and higher versions of MATLAB; it has not been tested on earlier versions of MATLAB.

2. Introduction

The CTRW-TPL model has been successfully applied to simulate the contaminant transport in porous and fractured media (e.g., Cortis and Berkowitz, 2004; Gao et al., 2009; Rubin et al., 2012; Wang and Cardenas, 2014; Edery et al., 2016; Chen et al., 2017; Hu et al., 2020; Li et al., 2022; Zhou et al., 2024, among many others). The CTRW-TPL model has four input parameters: normalized transport velocity (v_{φ}), normalized dispersion coefficient (D_{φ}), power law exponent (β), and time scale of t_2, which have to be inversely identified. Cortis and Berkowitz (2005) presented a MATLAB toolbox to inversely estimate the CTRW parameters. Then, Cortis et al.

(2020) expanded and completed the toolbox. The main disadvantage of the toolbox is its sensitivity to initial guesses for the parameters. The CTT is an inverse model based on the TLBO algorithm to estimate the parameters of the CTRW-TPL model. The required scripts of the CTT were designed in MATLAB R2017a. As above-mentioned, the CTT can be run on MATLAB R2017a and the higher versions of MATLAB.

3. MATLAB m-files of the CTT

The MATLAB m-files of the CTT, located in CTT folder, include CTRW_TPL_fit_continuous.m, CTRW_TPL_fit_instantaneous.m, Experimental_data.m, F_CTRW_TPL_continuous.m, F_CTRW_TPL_instantaneous.m, MainCode.m, and TLBO.m. The CTRW_TPL_fit_continuous.m and CTRW_TPL_fit_instantaneous.m calculate the objective function value for continuous and instantaneous injections, respectively. The F_CTRW_TPL_continuous.m and F_CTRW_TPL_instantaneous.m simulate the breakthrough curves (BTCs) for continuous and instantaneous injections, respectively. The Experimental_data.m contains the experimental data and the tracer test characteristics. The TLBO.m executes the TLBO algorithm. The MainCode.m is the main m-file executing the CTT inverse model by connecting the mentioned m-files together.

4. How to use the CTT

First of all, unzip and open the CTT folder. Then, follow the bellow steps:

Step 1: Enter the experimental data in the Experimental_data.m. Note that the experimental data usually contain the tracer relative concentration values versus time values and the travel distance of the tracer. Enter the time values in t_meas matrix and the concentration values in Cr_meas matrix. Also, enter the travel distance value in L variable.

Step 2: Adjust minimum and maximum values of the CTRW-TPL parameters in the TLBO.m according to previous studies or your knowledge about the case study. Note that VarMin matrix is related to the minimum values of the parameters, while VarMax matrix is related to the maximum values of the parameters.

Step 3: Open the MainCode.m and click on the run button. It is observed that the following expression appears in Command Window of MATLAB:

"Maximum number of iterations:"

"Maximum number of iterations" is equal to $iter_{max}$, that the user can set its value according to his/her experience. After entering $iter_{max}$ value, the following expression appears in Command Window of MATLAB:

"Population size:"

"Population size" is equal to N_{pop} that the user can set its value according to his/her experience. After entering N_{pop} value, the following expression appears in Command Window of MATLAB:

"If injection is continuous, type c and if injection is instantaneous, type i:"

This expression is related to the injection type. After entering "c" or "i", the CTT looks for the optimum values of the CTRW-TPL parameters automatically using the TLBO algorithm. During the running the code, the user can observe the best value of the objective function at each iteration in Command Window of MATLAB. Finally, the code prints the optimal values of the CTRW-TPL parameters and the *RMSE* value related to the optimal values of the CTRW-TPL parameters. Also, the CTT displays the experimental and fitted BTCs and the variations of the objective function value versus iteration graphically. It is necessary to mention that due to the stochastic nature of the TLBO algorithm, the CTT has to be executed ten times for each case study, and, finally, the average of the optimal values of the CTRW-TPL parameters obtained in the ten runs has to be reported as the final optimal parameters.

5. One-dimensional forward modeling

In addition to the MATLAB m-files of the CTT, the required MATLAB m-files were designed for the one-dimensional forward modeling of the solute transport using the CTRW-TPL model. The MATLAB m-files of the forward modeling include two groups of files. The first group, which is located in Forward_Group_1 folder, is used for the case where the tracer injection of is continuous. The second group, which is located in Forward_Group_2 folder, is used for the case where the tracer injection of is instantaneous.

Each group contains two files named Experimental_data.m and Forward_CTRW_TPL.m. The Experimental_data.m file is the MATLAB m-file containing the experimental data and the tracer test characteristics, and the Forward_CTRW_TPL.m file calculates the tracer concentration values using the CTRW-TPL model and compares them with the measured concentration values.

To use these scripts, first of all, unzip and open the Forward _Group_1 or Forward _Group_2 folder according to the type of the tracer injection. Then, follow the bellow steps:

Step 1: Enter the experimental data in the Experimental_data.m. Note that the experimental data usually contain the tracer relative concentration values versus time values and the travel distance of the tracer. Enter the time values in the t_meas matrix and the concentration values in the Cr_meas matrix. Also, enter the travel distance value in the L variable.

Step 2: Enter the values of the v_{φ}, D_{φ}, β, and t_2 parameters in v_psi, D_psi, beta, and t2 variables of the Forward_CTRW_TPL.m, respectively. Note the dimensions of the v_psi, D_psi, and t2 variables depend on the dimensions of measurement times in the Experimental_data.m file. The beta variable and relative concentration values are dimensionless.

Step 3: Click on the run button of the Forward_CTRW_TPL.m. After executing the script, *RMSE* value appears in Command Window of MATLAB. Also, a graphical comparison of the measured and calculated concentration values is displayed.

References

Berkowitz, B., Cortis, A., Dentz, M., Scher, H. 2006. Modeling non-Fickian transport in geological formations as a continuous time random walk. Reviews of Geophysics. 44: RG2003. doi:10.1029/2005RG000178.

Chen, Z., Zhan, H., Zhao, G., Huang, Y., Tan, y. 2017. Effect of roughness on conservative solute transport through synthetic rough single fractures. Water, 9: 565. https://doi.org/10.3390/w9090656.

Cortis, A., Berkowitz, B. 2004. Anomalous transport in classical soil and sand columns. Soil Science Society of America Journal, 68:1539–1548. https://doi.org/10.2136/sssaj2004.1539.

Cortis, A., Berkowitz, B. 2005. Computing ‘‘anomalous’’ contaminant transport in porous media: The CTRW MATLAB Toolbox. Groundwater, 947-950. https://doi.org/10.1111/j.1745-6584.2005.00045.x.

Cortis, A., Emmanuel, S., Rubin, S., Willbrand, K., Ben-Zvi, R., . Nissan, A., Berkowitz, B. 2020. The CTRW Matlab toolbox v4.0: A practical user's guide. http://www.weizmann.ac.il/EPS/People/Brian/CTRW.

Dentz, M., Berkowitz, B. 2003. Transport behavior of a passive solute in continuous time random walks and multi-rate mass transfer. Water Resources Research, 39: 1111. https://doi.org/10.1029/2001WR001163.

Edery, Y., Geiger, S., Berkowitz, B. 2016. Structural controls on anomalous transport in fractured porous rock. Water resources Research, 52: 5634–5643. https://doi.org/10.1002/2016WR018942.

Gao, G., Zhan, H., Feng, S.H., Huang, G., Mao, X. 2009. Comparison of alternative models for simulating anomalous solute transport in a large heterogeneous soil column. Journal of Hydrology, 377: 391–404. https://doi.org/10.1016/j.jhydrol.2009.08.036.

Hu, Y., Xu, W., Zhan, L., Ye, Z., Chen, Y. 2020. Non-Fickian solute transport in rough-walled fractures: The effect of contact area. Water, 12: 2049. https://doi.org/10.3390/w12072049.

Li, Y., Bian, J., Wang, Q., Li, T. 2022. Experiment and simulation of non-reactive solute transport in porous media. Groundwater, 60(3): 330-343. https://doi.org/10.1111/gwat.13153.

Rao, R.V., Savsani, V.J., Vakharia, D.P. 2011. Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. Computer-Aided Design, 43: 303–315. https://doi.org/10.1016/j.cad.2010.12.015.

Rao, S.S. 2019. Engineering Optimization: Theory and Practice. John Wiley & Sons, Inc., USA.

Rubin, S., Dror, I., Berkowitz, B. 2012. Experimental and modeling analysis of coupled non-Fickian transport and sorption in natural soils. Journal of Contaminant Hydrology, 132: 28-35. https://doi.org/10.1016/j.jconhyd.2012.02.005.

Wang, L., Cardenas, M.B. 2014. Non-Fickian transport through two-dimensional rough fractures: Assessment and prediction. Water Resources Research, 50: 871–884. https://doi.org/10.1002/2013WR014459.

Zhou, J-Q., Guo, L-G., Jiao, J.J., Jiang, X-Y., Luo, X. 2024. Can geometric parameters enable direct prediction of non-Fickian transport in rock fractures across diverse flow regimes? Journal of Geophysical Research: Solid Earth, 129: e2023JB027695. https://doi.org/10.1029/2023JB027695.