Deformation Analysis with Feature Voting

Omer Bar, Gilad Even-Tzur

Technion, Geo-Information and Geodesy, Haifa 3200003, Israel, (omerbar@campus.technion.ac.il; eventzur@technion.ac.il)

Key words: *GNSS; monitor network; deformation; feature voting*

ABSTRACT

Deformation analysis of GNSS network is usually computed using precise coordinates of the monitoring network points. Coordinates change over time construct a velocity field, which is used to estimate fault model parameters. Estimation process of coordinates is affected by several factors such as measurement errors, datum definition and the measurements datum defect. Points defining the monitoring datum have position accuracies which can increase inaccuracies in velocity estimations and the datum defect could cause biases and instability in computing the velocity field. This research proposes an algorithm of estimating geometric fault parameters using feature voting – addressing changes over time in GNSS vectors. The algorithm selects best solution for specific data-sets using minimal squared-disclosure between data and a tested value set of the fault model parameters. We concentrate on geometric fault models which rely solely on geometry between fault-line and monitoring network points. Geometric fault models are ill-conditioned, combined with low-frequency nature data - numerical instability rises. Vectors were computed with scientific processing software (Bernese), with consistent processing parameters at all epochs. Additionally, several numerical processes were adopted to transform the low-frequency data into usable datasets. Test cases were based upon 8 northern sites in the Israel's continuous operating permanent stations. Simulative data was created from a true solved epoch; then variety of epoch data-sets were introduced to the algorithm to compute pre-defined geometric fault model parameters. Test cases show that simulative data, without and with noises, introduced to the algorithm is suited for estimating model parameters properly.

I. INTRODUCTION

For the last twenty years researchers use precise GNSS networks to monitor deformation the earth's crust. Investigating the position of control points over time yield insights into the type and nature of crustal movements and help evaluating the deformation model. These monitoring networks consist of control points with precise coordinates measured over time, computed and adjusted independently per epoch. These precise coordinates as a function of time produce a precise velocity field which is used while estimating fault model parameters, using least square adjustment (Even-Tzur, 2001; Vogel and Van Mierlo, 1994; Ostrovsky, 2005; Nocquet *et al.*, 2002).

Precise coordinates are very useful in estimating deformation, they are easily determined by different GNSS techniques, over small or vast areas. Changes in coordinates over time, directly derive the relative deformation between a set of control points internally and relatively to a specific datum. Scientific GNSS processing software, such as GAMIT (Herring *et al.*, 2018) and Bernese (Dach *et al.*, 2015), are frequently operated to compute precise coordinates from GNSS measurements. These scientific software handles a vast amount of error-factors to precisely estimate coordinates. These error-factors are based on known physical phenomena models, and their residuals are being introduced into the computational process.

Precise coordinates and their Variance-Covariance matrix, are affected of numerous error-factors, such as: Measurement-Datum defect, gross measurement errors and random errors. Estimating coordinates of points embeds the Measurement-Datum defect, although it might cause biases in the estimation process. Monitoring network control points and their precise coordinates endure these biases as well - hence there is a need for a stable monitoring-datum. Sometimes obtaining a stable monitoring-datum is not possible to define, for not detecting a subset of monitoring network points that remain relatively fixed over time under a pre-decided significance level. Additionally, errors are gained and inaccuracies rise while relying on un-modeled and un-deducted movements of the monitoring-datum points (Even-Tzur, 2013; Papo and Perlmuter, 1981).

Fault models define surface movements caused by tectonic movement (Cohen, 1999). The models consider numerous factors that causes ground movements, such as – pressure, temperature, soil types and their fraction rates, etc. Geometrical fault models describe the geometric relativity between the fault-line and a point position from it. This relation, for example, can be expressed using some attributes, such as - perpendicular distance from the fault, lock and slip depths, fault slip velocity etc.

These geometrical fault-models are nonlinear ill-conditioned equations. Ill-condition equations show big errors in output while input consist of small errors. This is a numerical nonstable form of equations, which needs treatment for a stable process of computation (Tikhonov and Arsenin, 1977).

II. FEATURE VOTING TECHNIQUE

A. *Feature Voting*

Feature voting (Boiangiu and Radu, 2003) methodology determines model parameters values while reversing the roles between input data and estimated parameters. For a pre-defined model, for each parameter, numerous values are spanned; each value-set input is tested with the input dataset – if they are compatible (such as residual is under a certain threshold) – a vote is listed for that specific value-set. Usually, tracking the voting is carried out via a voting-matrix, each dimension is logically connected to a parameter and each index to a specific pre-defined value.

Hough transformation, as a common tool for line-extraction from images in Computer Vision, utilizes feature voting technique (Duda and Hart, 1972). Input data are pixels of detected points-of-interest, and its output are the lines detected from them. A line has only two parameters (even after their transformation to the image representation) hence the voting-matrix has only two dimensions. Analyzing all the maxima instances in the voting-matrix, yields the lines detected in the process.

This paper describes detection and analysis of deformation model parameters methodology, while not relying on precise coordinates. Precise vectors (with a time tag) spanning the monitoring network are the methods input, while previous knowledge, such as fault line azimuth, horizontal slip velocity, locking depth etc. - can be added, but not required. Positions are used with a metric precision, only to establish a geometry between input vectors and the fault.

The suggested algorithm compute fault-model parameters adopting feature-voting technique (Fernandes, and Oliveira, 2008), input are vectors computed in several epochs within the monitoring network, with their error estimates. The vectors are computed using a scientific GNSS post-processing software, campaigns processed utilizes same error-models and processing parameters for all epochs; this to ensure same vector solution (or processing type) at all epochs – that sets a single "solution datum" for the vectors. Using vectors from different "solution datums" will cause biases in the parameters values estimation.

Another difficulty is related to the nature of the input dataset (solved vectors), it is a "low frequency" dataset – *i.e.* a small finite amount of vectors to be processed. partially coping with this issue, all vectors in the monitoring networks are being computed - and not only the independent ones. Furthermore, all possible epoch differences are taken into considerations – which as well enlarges the data size. Hence, more geometrical relations are available for the voting technique.

The vectors error estimates assist in computing the threshold of the matching criteria. For example, an error estimate of 1 cm per direction on all vectors (at all epochs) could yield a velocity of √2 cm over time – or larger. Using this concept, maximal horizontal vector velocity error value can be determined and utilized as the minimal span between tested velocity values.

B. *Modifications for Geometric Fault Models*

Feature voting for a geometrical fault model is ambiguous, model equations are point oriented - adaptations are taken to match them to the vector properties. Fault models, and specifically geometrical fault models, describes points velocity affected by a single fault. A vector is a feature assembled between two points, it is necessary to accommodate the model. Furthermore, internal parameters are adapted to cope with internal relativity between the fault-line and the vector points. Point position is replaced with vertical distance to the fault line, hence azimuth and position of the fault are added as needed model parameters.

For example, Equation 1 describes the model of an Infinite long vertical Strike-Slip fault (left locked fault) (Cohen, 1999). This equation describes points velocity with dependency on a function of its geometrical relativity to the fault-line.

$$v_x = \frac{V}{\pi}\tan^{-1}\left(\frac{l}{LD}\right) \qquad (1)$$

where V = the fault slip velocity in mm/year

l = perpendicular distance between a point and the fault line

LD = Lock depth of the fault-line in kilometers

Perpendicular distance is denoted by the relation between point position and (i) fault-line Azimuth, and (ii) fault locking position.

Fault model equation is modified to meet vectors two nodes – points P_1 and P_2, as described in Equation 2. The modification is set to the numeric differentiation of the velocities computed on vectors nodes $v_x^{P_2}$; $v_x^{P_1}$, with respect to the fault-line parameter value-set, at a single epoch, relying on the original fault model.

$$v_{vector} = v_x^{p_2}(t) - v_x^{p_1}(t) \qquad (2)$$

where $v_x^{P_1}$ = velocity computed on one of the vectors nodes

$v_x^{P_2}$ = velocity computed on the second nodes of the vector

t = specific epoch time

The velocity for the vectors length is derived from the relative velocity between the two vector nodes.

Multiplication of this vector-velocity in the time span produces the estimated length-change over time (by a set of give values for the fault mode parameters). This methodology is introduced with coordinates that are not precise, and yields computation that corresponds with a specific threshold denoted via cross-validation.

All available vectors with two or more epochs are introduced to the voting process, which enlarges the dataset entering the voting process. Additionally, this helps coping with changes in control points within the monitoring network, for example – a control point is abandoned and a new control point is built instead of it.

C. *Maxima Detection*

The voting criteria is described by difference between measured length change in vectors sizes, and the computed change in vector length denoted from a pre-defined fault model and a pre-selected value-set for its parameters. Differences smaller than a selected threshold, denotes a vote to that value-set. All value-sets are being tested for all input vectors, and for each criteria-matching (difference is smaller than the threshold) – the vote is promoted voting-matrix for the selected and tested value-set.

Several maxima can be found in the voting matrix - due to the interval of the parameters values and due to the ill-condition state of the fault model. All votes over a certain significant level (such as vote numeric value is more than 85% of the input vectors), undergo cross-validation computation. The cross-validation is the total squared residuals for each detected maxima value-set, related to all input vectors. For example – in a case of a single fault computation - a maxima value-set is selected with respect to the minimum cross-validation from all detected optional maxima's.

D. *Coping with Low Frequency Data*

Input data-set is of low-frequency nature, and the ill-conditioned equation-system is unstable numerically. Enlarging the data-set is possible by replicating input vectors using random noise perturbation with an average of zero – as utilized in Monte-Carlo simulation simulation (Raychaudhuri, 2008). Adding more data with the same trend of the original data, with zero bias and a zero-average random noise, emphasizes the votes for most appropriate value-set and assists in the cross-validation process. Nonetheless, all input is being normalized, to assist overcoming the ill-conditioned state of the fault model.

E. *Initial Values*

Initial values can be adopted from other researchers works. Nonetheless, several parameters can be approximated using input data – such as horizontal slip rate, by maximum difference of homological-vectors lengths divided by their time-span. Other parameters should be spread with respect to pre-known physical attributes – such as, local maximal crust depth for maximal locking depth in a locked fault model.

III. TEST CASES

Test cases relate to northern area of Israel with eight sites from the Israeli continuous operation reference stations network, assembling the deformation monitoring network for the simulations - gathering a set of 28 vectors. This area, as can be seen in Figure 1, is greatly affected by the Dead-Sea Fault Transform (Reinking *et al.*, 2011). The paper will describe simulations for left Infinite long vertical Strike-Slip fault model. Several simulations resemble the Dead-Sea Fault parameters in the northern Israeli area near the Golan Heights.

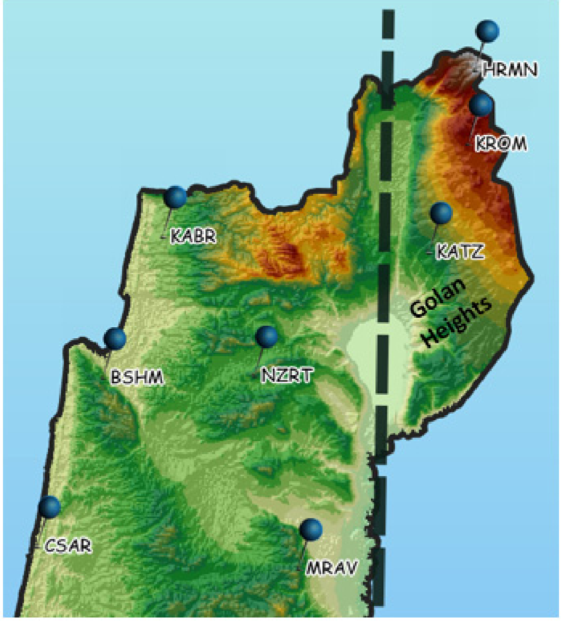

Figure 1. Northern Israel with Dead Sea fault simulative fault line (dashed grey) and Survey of Israel CORS sites (blue push-pins).

For the simulations - single epoch was computed with a processing software, and rest of needed measurements epochs were created artificially from it.

A. *Simulative Cases*

Synthetic epochs were created by computing the location change to each point, in each epoch, related to the first epoch points coordinates (that was computed using a GNSS processing software) – relying on pre-selected fault model and parameter-values. Since no precise coordinates are in use, each vector was computed a length changes by the position of its end-nodes (point) with respect to the fault-line.

Simulative data cases concluded several stages:

1) Two additional epochs without noise added.
2) Three additional epochs without noise added.
3) Two additional epochs with noise added.
4) Three additional epochs with noise added.

The noise added to these test cases is a random normal-distribution and has perturbation nature relatively to the input dataset.

The feature-voting methodology presented returned outputs similar in ratio to the pre-defined fault parameters. Computation process is based on ill-condition equations, hence any tiny truncation and round-off error in the computational process inside the memory - affects the process results. Thus, there is low percentage of absolute accuracy in computed fault parameter values.

Tables 1 and 2 describes the input and outputs of the four cases of the simulations.

Table 1. Simulations results: Known Values

#	Az [^{O}]	Ld [m]	V_Hz [mm/year]	E0 [m]	N0 [m]
1	0.3	12,015	8.2	252,200	760,505
2	0.3	12,015	8.2	252,200	760,505
3	0.3	12,015	8.2	252,200	760,505
4	0.3	12,015	8.2	252,200	760,505

Table 2. Simulations results: Computed Values

#	Az [^{O}]	Ld [m]	V_Hz [mm/year]	E0 [m]	N0 [m]
1	-0.87	11,919	8.1	251,770	747,884
2	0	12,324	8.2	251,902	749,902
3	-0.08	11,900	8.1	251,770	747,885
4	0	12,500	8.3	252,419	775,505

B. *Discussion and Conclusions*

This feature-voting process for evaluating fault parameters has advantages such as (a) changes in the control points of the deformation monitoring network doesn't need any special input manipulation, (b) estimation is not based on precise coordinates – it relies on precise relations between them, (c) copes with gross errors in input – they are eliminated in the voting process while searching for maxima, and (d) geodetic datum and monitoring datum are not set – thus, omitting error-estimates of the coordinates which causes biases in the traditional estimation process of model parameter.

Ill-conditioned state of geometric fault model equations, is taken into consideration in the presented algorithm. Several actions are taken to assist in computing an appropriate value-set for the model parameters such as adding perturbed replicas of the data, normalizing the data and fundamentally use a feature voting technique.

This technique cannot detect rigid body motion, such as full body rotation or translation – due to the fact that there are not internal changes in these cases.

The proposed method can be modified to meet other geometric fault models - such as strike-slip fault models with free sleeping or locking in a defined range.

Future research will tackle true data, without constant sites in the input data. Additionally, we will try to describe the estimated error in the output results.

IV. ACKNOWLEDGEMENTS

We thank the Survey of Israel in supporting the research and suppling GNSS measurements for it.

References

Boiangiu, C. A., and Radu, I. (2003). Voting-based image segmentation. *The Proceedings of Journal ISOM*, 7.2, pp. 211-220.

Cohen, S. (1999). Numerical models of crustal deformation in seismic zones. *Advances in Geophysics*, 41, pp. 134-231.

Dach, R., Lutz, S., Walser, P., and Fridez, P. (2015). *Bernese GNSS Software Version 5.2.* Astronomical Institute, University of Bern. Bern: Bern Open Publishing. DOI: 10.7892/boris.72297

Duda, R. O., and Hart, P. E. (1972). Use of the Hough Transformation to Detect Lines and Curves in Pictures.. *Communicatiha ons of the ACM*, 15.1, pp. 11-15

Even-Tzur, G. (2001). Sensitivity analysis of deformation monitoring networks in the north of Israel. *Israel Journal of Earth Sciences*, 50(1).

Even-Tzur, G. (2013). Datum Definition for GPS networks. *Survey Review*, 35(277), pp. 475-486.

Herring, T., King, R., Floyd, M., and McClusky, S. (2018). *GAMIT Reference Manua*l. Massachusetts Institute of Technology. Department of Earth, Atmospheric, and Planetary Sciences.

Fernandes, L. A., and Oliveira, M. M. (2008). Real-time line detection through an improved Hough transform voting scheme. *Pattern recognition*, 41(1), pp. 299-314.

Nocquet, J. M., Calais, E., And Nicolon, P. (2002). Reference frame activity: Combination of National (RGP) and Regional (REGAL) Permanent Networks Solutions with EUREF-EPN and the ITRF2000. *Proceedings of The EUREF 2002 Symposium*, (pp. 398-404).

Ostrovsky, E. (2005). *The G1 geodetic-geodynamic network: results of the G1 GPS surveying campaigns in 1996/1997 and 2001/2002*. Survey of Isreal, Tel-Aviv, Israel.

Papo, H., And Perlmuter, A. (1981). Datum definition by free net adjustment. *Bulletin Geodesique*, 55, pp. 218-226.

Raychaudhuri. (2008). Introduction to Monte Carlo simulation. *WSC '08: Proceedings of the 40th Conference on Winter Simulation*, pp. 91-100. DOI: 10.1109/WSC.2008.4736059.

Reinking, J., Smit-Philipp, H., and Even-Tzur, G. (2011). Surface deformation along the Carmel Fault System, Israel. *Journal of Geodynamics*, 52, pp. 321-331.

Tikhonov, A., and Arsenin, V. (1977). Solutions of Ill-posed Problems. New York: V. H. *Winston and Sons*.

Vogel, M., and Van Mierlo, J. (1994). Deformation analysis of the Kfar-Hanassi network. Haifa: *Perelmuter workshop on dynamic deformation models*, pp. 273-284