Solution of Load-Flow Problem using Fuzzy Linear Regression Approach

Adnan S. Borisly

Electrical & Electronic Department
Telecommunication Institute

A. K. Al-Othman

Electrical Engineering Department
College of Technological Studies

Public Authority for Applied Education and Training (PAAET), KUWAIT

`*Abstract:* - This paper presents a new approach to solve the load-flow problem using Tanaka's Fuzzy Linear Regression formulation (FLR). The load-flow model is formulated as a fuzzy linear optimization problem, where the objective is to minimize the sum of the spread of the states, subject to double inequality constraints on each pre-specified active and reactive power to guarantee that the original membership is included in the state membership. Linear programming is employed to obtain the middle and the symmetric spread for every state variable. The estimated middle corresponds to the value of the state. While the symmetric spreads in the membership functions of the state variables represents the uncertainty (vagueness) around the state.

The proposed formulation has been applied to various test systems. The outcome is very encouraging and proves that proposed (FLR) is very applicable and shows reliability, accuracy in solving the power-flow problem.

Key-Words: - Load-Flow, Newton-Raphson and Fuzzy Linear Regression (FLR)

1 Introduction

The power flow study of an electric power system is also known as “load-flow” study. In essence, this study involves the calculation of line loading given the generation and demand level. Ward and Hall [1] are frequently credited for being the first to formulate the load-flow problem. This problem has been studied widely and solved with the help of various numerical iterative methods such as Gauss-Seidel and Newton-Raphson [2-4].

Evolutionary algorithms have the ability to combat the above drawbacks. As an optimization technique, genetic algorithms [5, 6] and PSO [7, 8] are much less dependent on the start values of the variables in the optimization problem when compared with the widely used Newton-Raphson or mathematical programming techniques such as SQP (Sequential Quadratic Programming). In addition Evolutionary algorithms do not rely on the guidance of the gradient information, such as the Jacobian matrix, hence they are more capable of determining the global optimum solution.

Authors in [9] presented an application of particle swarm optimization (PSO) for solving the load flow problem as an optimization problem. The PSO algorithm has been strengthened using breeding technique similar to that applied in Genetic algorithm (GA). The new suggested algorithm has been applied to two test systems.

Yin and Germay [10] were the first who applied GAs to solve the load-flow problem. Unfortunately results reported were not very near the solution. It was shown in [10] the total mismatch (accuracy) achieved for three runs of GAs of the six-bus test system were 1.0216, 0.5356 and 0.5218. Apparently, these mismatches can only suggest that the solution is quite inaccurate and the problem remains to be solved. These inaccuracies were probably due to the binary representation of candidates, which led to discretization errors. Also, it has been conceded in [10] that those GA solutions can only serve as a guide within the solution search space. The GA solution could then be used as an initial guess for the Newton-Raphson method, which would hopefully converge to the exact solution.

Wong *et al.* introduced, in [11-13], a constrained GA for solving power-flow. This approach was based on a constraint satisfaction technique to force the mismatch of the total power balance equations to zero. By incorporating the concept of dependant variables in the formulation [12] and setting the mismatch to zero, the power injection equations are reorganized to solve for the unknowns (nodal phasor voltages). Using this reformulation of the load-flow equations, Wong *et al.* found that the GA could successfully converge to the correct solution. This study intends to solve the load-flow problem with the well known Tanaka's fuzzy linear regression model.

2 An overview of Tanaka's Fuzzy linear regression

Fuzzy linear regression was introduced by Tanaka et. al [14] in 1982. The general form of Tanaka's formulation is given by:

$$\underset{\sim}{Y} = f(x) = A_0 + A_1 x_1 + A_2 x_2 + ... + A_n x_n = \boldsymbol{Ax} \qquad (1)$$

where $\underset{\sim}{Y}$ is output (dependant fuzzy variable), $\{x_1, x_2, ...x_n\}$ is a non fuzzy set of crisp independent parameters and $\{A_0, A_1, ..., A_n\}$ is a fuzzy set of symmetric members, unknowns, needs to be estimated. Each fuzzy element in that set may be represented by a symmetrical triangular membership function, shown in figure 1, defined by a middle and a spread values, p_i and c_i respectively. The middle is known as the model value and the spread denotes the fuzziness of that model value. The triangular membership function can be expressed as:
Therefore, since $A_i = (p_i, c_i)$, then equation (1) may be rewritten as:

$$\mu_{\underset{\sim}{A}_i}(a_i) = \begin{cases} 1 - \dfrac{|\, p_i - a_i \,|}{c_i} & , \; p_i - c_i \le a_i \le p_i + c_i \\ 0 & , \; otherwise \end{cases} \qquad (2)$$

Therefore, since $A_i = (p_i, c_i)$, then equation (1) may be rewritten as:

$$\underset{\sim}{Y} = f(x) = (p_0, c_0) + (p_1, c_1)x_1 + ... + (p_n, c_n)x_n \qquad (3)$$

The membership function of output $\underset{\sim}{Y}$ may be given by:

$$\mu_{\underset{\sim}{Y}}(y) = \begin{cases} \max(\min([\mu_{\underset{\sim}{A}_i}(a_i)]) & , \{a \,|\, y = f(x, a)\} \ne \varnothing \\ 0 & , \; otherwise \end{cases} \qquad (4)$$

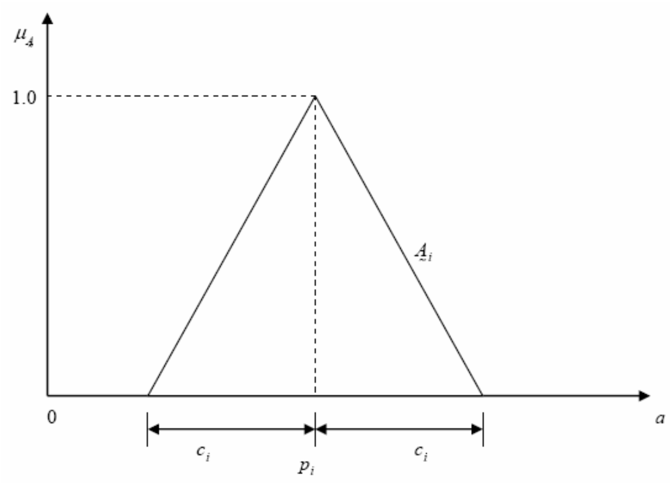

Figure 1 membership function of fuzzy coefficient $\underset{\sim}{A}$

Now, by substituting equation (3) in (4), the output membership function is given as:

$$\mu_{\underset{\sim}{Y}}(y) = \begin{cases} 1 - \dfrac{\left| y - \sum_{i=1}^{n} p_i x_i \right|}{\sum_{i=1}^{n} c_i |x_i|} & , x_i \ne 0 \\ 1 & , \; x_i = 0, \; y_i = 0 \\ 0 & , \; x_i = 0, \; y_i \ne 0 \end{cases} \qquad (5)$$

The output membership function is depicted in figure 2. From regression point of view, equations (1-5) may be applied to m samples where the output can be either non-fuzzy, (certain or exact), in which no assumption of ambiguity is associated with the output or fuzzy (uncertain), where uncertainty in the out is involved due to human judgment or meters impression [15]. In this study both non fuzzy and fuzzy output will be considered.

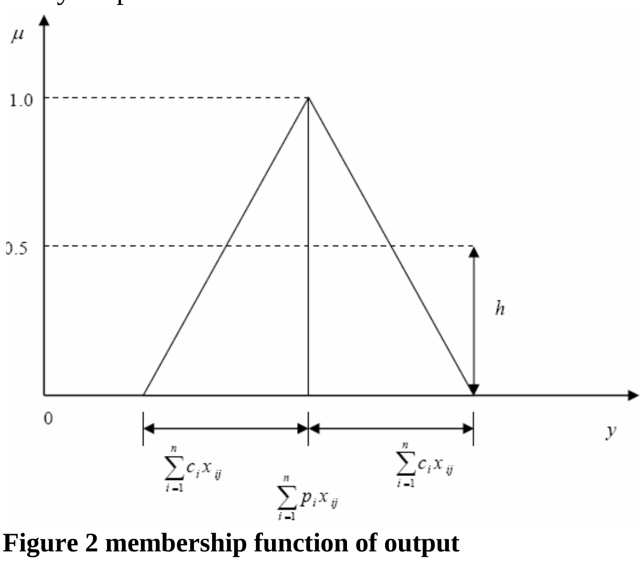

Figure 2 membership function of output

2.1 Non- fuzzy output model [14]:

In this model, Tanaka converted regression model into a linear programming problem [14]. In this case the objective is to solve for the best parameters, i.e. A^*, such that the fuzzy output set is associated with a membership value grater than h as in ;

$$\mu_{Y_j}(y_j) \geq h, \qquad j = 1,....,m \qquad (6)$$

where $h \in [0,1]$ is the degree of the fuzziness and is normally defined by the user.

Therefore, with equation (6) as a condition, the main objective is to find the fuzzy coefficients that minimize the spread of all fuzzy output for all data set. Note that the fuzziness in the output is due to fuzziness assumed in the system structure A^*.

Thus, given non-fuzzy data (y_i, x_i), the fuzzy parameters $\boldsymbol{A}^* = (\boldsymbol{p}, \boldsymbol{c})$ may be solve for by the linear programming formulation as:

$$F_{non-fuzzy} = \min(\sum_{j=1}^{m}\sum_{i=1}^{n} c_i x_{ij}) \qquad (7)$$

Subject to:

$$y_j \geq \sum_{i=1}^{n} p_i x_{ij} - (1-h)\sum_{i=1}^{n} c_i x_{ij} \qquad (8)$$

$$y_j \leq \sum_{i=1}^{n} p_i x_{ij} + (1-h)\sum_{i=1}^{n} c_i x_{ij} \qquad (9)$$

Note that from in (8) and (9), $\sum_{i=1}^{n} p_i x_{ij}$, defines the middle value and $\sum_{i=1}^{n} c_i x_{ij}$ defines the sympatric spread to the left, constraint (8), and to the right, constraint (9), as illustrated in figure 2. As can be seen from the figure 2, as the degree of fuzziness, h, increases the spread, c_i, increases and therefore the uncertainty associated with the p_i would increase [16].

3 The Proposed Formulation

The load-flow rectangular formulation can be described as follows. Consider a network with total number of *N* nodes (buses). At any bus *i*, the nodal active $P_i(V, \theta)$ and reactive $Q_i(V, \theta)$ are given by:

$$P_i(V,\theta) = \sum_{j=1}^{N} |V_i||V_j|\left(G_{ij}\cos(\theta_i - \theta_j) + B_{ij}\sin(\theta_i - \theta_j)\right) \qquad (10)$$

$$Q_i(V,\theta) = \sum_{j=1}^{N} |V_i||V_j|\left(G_{ij}\sin(\theta_i - \theta_j) - B_{ij}\cos(\theta_i - \theta_j)\right) \qquad (11)$$

Where

$|V_j|$ Voltage magnitude at bus *j*.

θ_j Voltage angle at bus *j*.

G_{ij}, B_{ij} The (i,j)th element of the admittance matrix Y-bus.

At each node the voltage magnitude and the voltage angle are unknowns and must be calculated, except the slack bus where is known $\theta_{slack} = 0$. Suppose that bus number 1 is the slack bus, therefore the unknown state vector may be defined as:

$$\underline{x} = \begin{bmatrix} \underline{\theta} \\ |\underline{V}| \end{bmatrix} = \begin{bmatrix} \theta_2 \\ \theta_3 \\ \vdots \\ \theta_N \\ |V_1| \\ |V_2| \\ \vdots \\ |V_N| \end{bmatrix} \qquad (12)$$

It is essential to determine the values of the unknowns in equation (12) such that:

$$P_i(\underline{x}) - P_i^{net} = 0 \qquad (13)$$

and

$$Q_i(\underline{x}) - Q_i^{net} = 0 \qquad (14)$$

where P_i^{net} and Q_i^{net} are the prespecified active and reactive power levels of bus i. Thus, mismatch in equations (13) and (14) must be driven to zero (ideally).

Due to nonlinearity in equations (10) and (11), an iterative root finding scheme such as Netwon-Raphson (N-R) must be adopted to solving nonlinear equations. By selecting a number of equations from (10) and (11) equal to the number to the number of unknowns in $\underline{x}$, a matrix valued function f($\underline{x}$) as in:

$$f(\underline{x}) = \begin{bmatrix} p_2(\underline{x}) \\ p_3(\underline{x}) \\ \vdots \\ p_N(\underline{x}) \\ Q_1(\underline{x}) \\ Q_2(\underline{x}) \\ \vdots \\ Q_N(\underline{x}) \end{bmatrix} \qquad (15)$$

Successive approach of $\underline{x}$ toward the solution is handled as in:

$$\underline{x}^{k+1} = \underline{x}^{k} + \Delta\underline{x} \qquad (16)$$

k denotes the current iteration number of N-R. The non-linear power system model is linearized around some operating point $\underline{x}^o$ using Taylor series expansion, retaining the first two terms and ignoring higher order terms. This leads to the following relationship where Δx may be then found by solving the set of linear equations:

$$J(\underline{x}^k).\Delta x = -\Delta g(\underline{x}^k) \qquad (17)$$

where $J(\underline{x}^k) = \partial f(\underline{x}^k)/\partial \underline{x}$ is the Jacobian square matrix evaluated at $\underline{x}^k$ and $\Delta g(\underline{x})$ is the active and reactive power mismatches and is computed as

$$\Delta g(\underline{x}) = \begin{bmatrix} p_2(\underline{x}) - p_2^{net} \\ p_3(\underline{x}) - p_3^{net} \\ \vdots \\ p_N(\underline{x}) - p_N^{net} \\ Q_1(\underline{x}) - Q_1^{net} \\ Q_2(\underline{x}) - Q_2^{net} \\ \vdots \\ Q_N(\underline{x}) - Q_N^{net} \end{bmatrix} \qquad (18)$$

The main contribution in this paper is to employ Tanaka's fuzzy linear regression formulation to solve for Δx in the linearized model of equation (17).

The linearized power system in equation (17) the J^{th} measurement can be rewritten as:

$$\Delta g_j = \Delta x_1 J_{j1} + \Delta x_2 J_{j2} + ... + \Delta x_n J_{jn} \qquad (19)$$

If we define the change in the system state variables, $\Delta\underline{x}$, to be fuzzy member having a middle and a spread values, p_i and c_i respectively. Then, equation (19) can be expressed as:

$$\Delta g_j = (p_1,c_1)J_{j1} + (p_2,c_2)J_{j2} + ... + (p_n,c_n)J_{jn} \qquad (20)$$

Note that the modal value p_i (i.e. the middle) for a given unknown represents the value of the change in the system state variables, Δx_i, at the current iteration of the linearized model. The spread c_i on the other hand, which is symmetric, correspond to the incremental confidence interval of that state variable. Therefore, $\Delta\underline{x}$ can be defined:

$$\Delta\underline{x} \equiv \left[(p_1,c_1),(p_2,c_2),....(p_n,c_n)\right] \qquad (21)$$

Tanaka's fuzzy linear regression model, eqs (7)-(9), is modified in order to be used to solve for Δx. In this linear fuzzy formulation, the optimal state estimate vector $\underline{x}$ may be determined by minimize the sum of the spread of all state variables, in this case the change in state variables, subject to a number of constraint representing the rows of the Jacobian can be expressed as:

$$F = \min\left(\sum_{j=1}^{m}\sum_{i=1}^{n} c_i J_{ij}\right) \qquad (22)$$

Subject to:

$$y_j \ge \sum_{i=1}^{n} p_i J_{ij} - (1-h)\sum_{i=1}^{n} c_i J_{ij} \qquad (23)$$

$$y_j \le \sum_{i=1}^{n} p_i J_{ij} + (1-h)\sum_{i=1}^{n} c_i J_{ij} \qquad (24)$$

where h is the degree of the fuzziness and is specified by the decision maker. The above model is a linear programming model and they can be solved by any linear programming package.

Upon choosing an appropriate initial guess $\underline{x}^o$, an arbitrary initial guess of considered state variables, N-R should iterate until the stopping criterion is reached. Thus the non-linear power-flow problem is solved and eventually the states (voltage magnitudes and phase angles) are computed by the fuzzy linear formulation.

4 Implementation of case studies

This section presents some typical results obtained by applying the proposed algorithms to the four-bus system form [3], six-bus test system form [17], IEEE 30-bus, IEEE 39-bus, IEEE 57-bus and IEEE 118-bus test network data form [18]. A set of MATLAB™ files has been developed to facilitate the computation of all state variables to illustrate the concepts. The fuzzy LP problems have been solved by the function *linprog()* incorporated in the MATLAB™ optimization toolbox [19].

ISSN: 1790-5117

ISBN: 978-960-6766-34-3

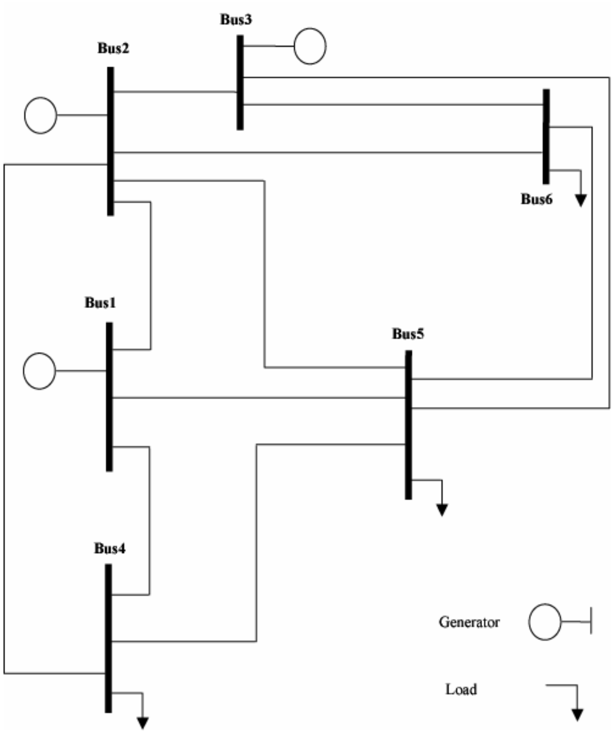

Figure 3 Single Line digram of the six bus test system

Table 1 present typical results obtained by the proposed fuzzy linear formulation, when applied to the six-bus network form [17] and shown in figure 3. For validation purposes, load-flow solution of the same test system has been carried by traditional technique where inversion of Jacobian is obtained and multiplied by $\Delta g(\underline{x})$ to compute the update vector $\Delta \underline{x}$.

TABLE 1: NORMAL SOLUTION OF SIX-BUS SYSTEM

State	Traditional Technique	Fuzzy-LP (*h=0*)	Fuzzy-LP (*h=0.5*)
θ_1	**0**	**0**	**0**
θ_2	**-3.6712**	**-3.6712**	**-3.6774**
θ_3	**-4.2733**	**-4.2733**	**-4.2820**
θ_4	**-4.1958**	**-4.1958**	**-4.2069**
θ_5	**-5.2764**	**-5.2764**	**-5.2908**
θ_6	**-5.9475**	**-5.9475**	**-5.9641**
$\lvert V_1\rvert$	**1.0500**	**1.0500**	**1.0480**
$\lvert V_2\rvert$	**1.0500**	**1.0500**	**1.0480**
$\lvert V_3\rvert$	**1.0700**	**1.0700**	**1.0680**
$\lvert V_4\rvert$	**0.9894**	**0.9894**	**0.9872**
$\lvert V_5\rvert$	**0.9854**	**0.9854**	**0.9833**
$\lvert V_6\rvert$	**1.0044**	**1.0044**	**1.0022**

With the fuzziness level h = 0, the proposed fuzzy LP technique and the traditional technique produce the same load-flow solution, which is expected the fuzzy LP technique would produce crisp solution with h = 0 (a crisp estimate occurs when its corresponding spread or width is 0). However, discrepancies are obvious due to the fairly significant noise level associated with higher level of fuzziness h=0.5.

In this test the fuzzy LP and Newton Raphson algorithm was found to perform reliably, with convergence occurring in 3 iterations. This is consistent with the behavior of the Newton Raphson process in solving other types of power system load-flow problems. Furthermore, the execution time was found to be 1.0514 sec., see table 2**.** Table 2, shows the CPU time and number of iterations required by both fuzzy LP technique and the traditional technique when applied to various test systems.

TABLE 2: CPU & EXECUTION TIME

System	**Fuzzy LP**		**Trad. Technique**	
	# Itr	**CPU time**	**# Itr**	**CPU time**
4-bus	5	0.3153 **sec**	3	0.11000 **sec**
six-bus	3	1.0514 **sec**	3	0.28000 **sec**
30-bus	4	1.1498 **sec**	4	0.23000 **sec**
39-bus	4	1.2976 **sec**	4	0.27158 **sec**
57-bus	3	1.7819 **sec**	3	0.26113 **sec**
118-bus	3	2.8449 **sec**	3	0.36153 **sec**

5 Conclusion

An analysis of uncertainty in power system state This paper introduces a new approach based on Tanaka's Fuzzy linear regression technique to solve the load-flow problem. The problem is formulated as a constrained optimization problem. Linear programming and N-R has been employed to solve the load-flow problem. The proposed fuzzy linear regression technique has bee applied to various test systems. Results show that the proposed method was accurate, reliable, and in efficient method for solving the load-flow problem.

ISSN: 1790-5117

ISBN: 978-960-6766-34-3

References:

[1] J. B. Ward and H. W. Hale, "Digital computer solution of power-flow problems," *Trans. AIEE*, vol. PAS-75, pp. 938-404, 1956.

[2] A. R. Bergen and V. Vittal, *Power systems analysis*, 2nd ed. Upper Saddle River, N.J.: Prentice Hall, 2000.

[3] J. J. Grainger and W. D. Stevenson, *Power system analysis*. New York: McGraw-Hill, 1994.

[4] J. D. Glover and M. S. Sarma, *Power system analysis and design*, 2nd ed. Boston: PWS Pub., 1994.

[5] D. E. Goldberg, *Genetic algorithms in search, optimization, and machine learning*. Reading, Mass. ; Harlow: Addison-Wesley, 1989.

[6] Z. Michalewicz, *Genetic algorithms + data structures = evolution programs*, 3rd rev. and extended ed. Berlin ; New York: Springer-Verlag, 1996.

[7] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," presented at Conf. on Neural Networks, Perth,Australia, pp. 1942-1948, 1995.

[8] V. Miranda and N. Fonseca, "EPSO - evolutionary particle swarm optimization, a new algorithm with applications in power systems," presented at Transmission and Distribution Conference and Exhibition, pp. 745 -750, 2002.

[9] A. A. El-Dib, H. K. M. Youssef, M. M. El-Metwally, and Z. Osman, "Load flow solution using hybrid particle swarm optimization," presented at International Conference on Electrical, Electronic and Computer Engineering, ICEEC, pp. 742-746, 2004.

[10] X. D. Yin and N. Germay, "Investigation on Solving the Load Flow Problem by Genetic Algorithms," *Elect. Power Systems Res*, vol. 22, pp. 151-163, 1991.

[11] K. P. Wong and A. Li, "Solving the load-flow problem using genetic algorithm," *Evolutionary Computation, 1995., IEEE International Conference on*, vol. 1, pp. 103, 29 Nov.-1 Dec. 1995.

[12] K. P. Wong, A. Li, and M. Y. Law, "Development of constrained-genetic-algorithm load-flow method," *Generation, Transmission and Distribution, IEE Proceedings-*, vol. 144, pp. 91 - 99, March 1997.

[13] K. P. Wong, A. Li, and T. M. Y. Law, "Advanced constrained genetic algorithm load flow method," *Generation, Transmission and Distribution, IEE Proceedings-*, vol. 146, pp. 609 -616, Nov. 1999.

[14] H. Tanaka, S. Uejima, and K. Asai, "Fuzzy linear regression model," presented at Int. Congr. on Applied Systems Research and Cybernetics, Acapulco, Mexico, pp. 2933-2938, 1980.

[15] T. Ross, *Fuzzy Logic with Engineering Applications*, 2nd ed: John Wiley & Sons, Ltd, April 2005.

[16] H. Moskowitz and K. Kim, "On assessing the H value in fuzzy linear regression," *Fuzzy Sets and Systems*, vol. 58, pp. 303-327, 1993.

[17] A. J. Wood and B. F. Wollenberg, *Power generation, operation, and control*, 2nd ed. New York ; Chichester: John Wiley & Sons, 1996.

[18] Y. Wallach, *Calculations and programs for power system networks*. Englewood Cliffs, N.J.: Prentice-Hall, 1986.

[19] J. S. Lai and F. Z. Peng, "Multilevel Converter-A New Breed of Power Converters," *IEEE Trans. Ind. Appl.*, vol. 32, pp. 509-517, 1996.

ISSN: 1790-5117

ISBN: 978-960-6766-34-3