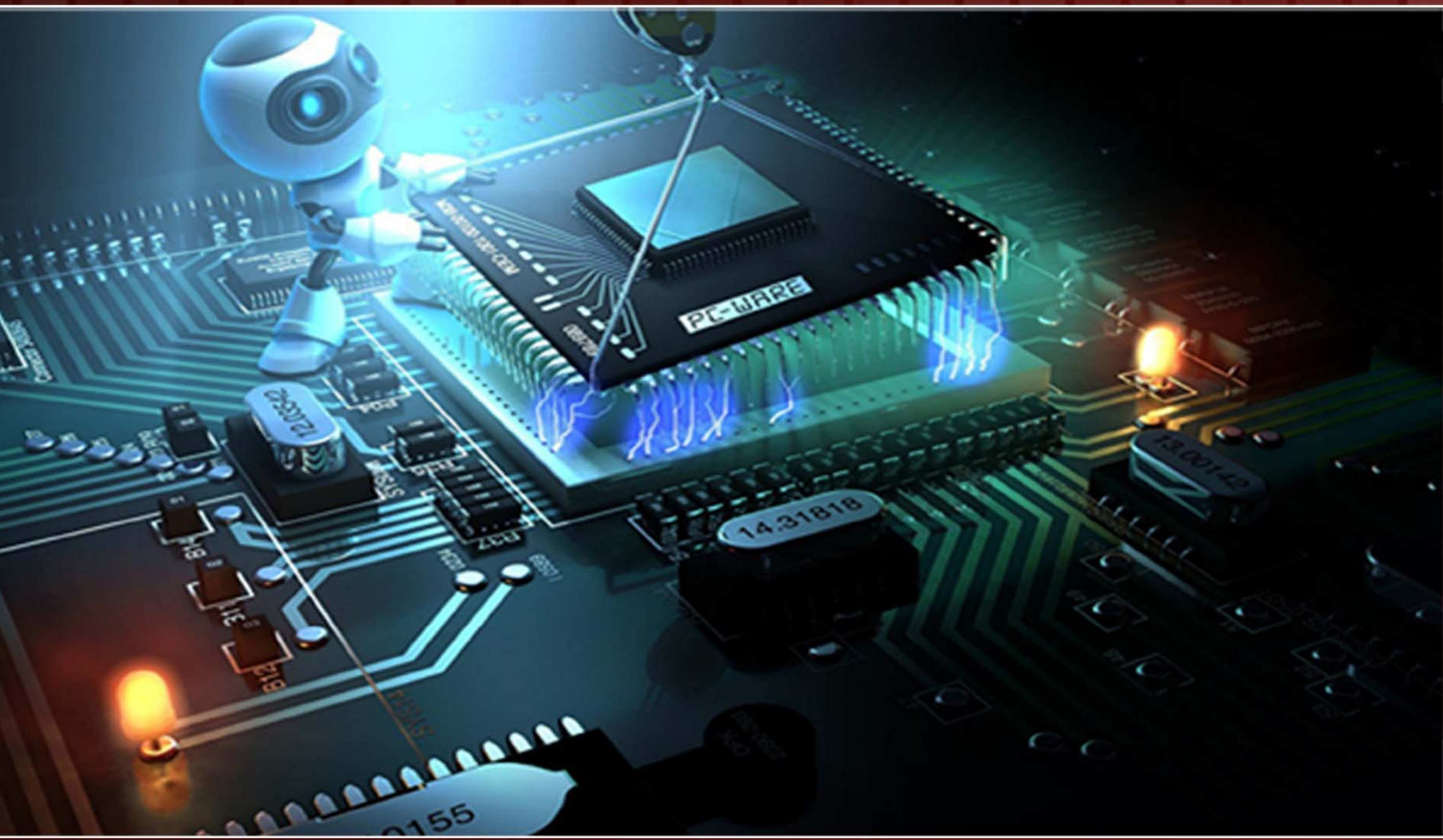
ISSN: 2054-7390

Volume 7 No 1, February 2019

Transactions on Machine Learning and Artificial Intelligence



Society for Science and Education
UNITED KINGDOM

TABLE OF CONTENTS

EDITORIAL ADVISORY BOARD

DISCLAIMER

SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM

TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE

Volume 7 No. 1

ISSN 2054-7390

Tripartite Entanglement for Qubits and Qudit in Double Photoionization of Xenon Atom

M. Chakraborty[1] and S.Sen[2]

[1]*Department of Physics, Asansol Girls' College, Asansol -713304, India*

[2]*Department of Physics; Triveni devi Bhalotia College, Raniganj- 713347, India*

bminakshi@yahoo.com; sandip_sn@yahoo.com

ABSTRACT

Quantum entanglement holds the key to an information processing revolution. In this article, we study the entanglement properties of tripartite states of two electronic qubits and ionic qudit,without observing spin orbit interaction (SOI), produced by single-step double photoionization from Xeon atom following the absorption of a single photon. The dimension of the Hilbert space of the qudit depends upon the electronic state of the residual photoion Xe^{2+}. In absence of SOI, Russel-Salunders coupling (L-S coupling) is applicable. As the estimations of entanglement, we consider Peres-Horodecki condition and negativity. In case of L-S coupling, all the properties of a qubit-qudit system can be predicted merely with the knowledge of the spins of the target atom, the residual photoion, emitted electrons and state of polarization of the incident photons.

Keywords – entanglement, quantum information, qubit, qudit, Peres-Horodecki condition, negativity

1 Introduction

Quantum entanglement is a key prediction of quantum mechanics and one of the resources needed in quantum information (QI) processing [1, 2]. In analogy to the classical information, QI also needs bits, called qubits. But, a bit in QI is a quantum system which has, at least, one observable requiring two-, or higher- , dimensional space for its characterization. A two-dimensional quantum system (e.g., a spin-$\frac{1}{2}$ particle) is called in QI as a quantum bit or qubit [3, 4].In general, a d-dimensional quantum system (with d>2) is called a qudit [5].

Availability of two or more qubits with entanglement is an essential ingredient for any quantum information related studies. Quantum entanglement is a nonlocal property that allows a set of qubit to express higher correlation that is not possible in classical systems.

Research into quantum entanglement was started in 1935 by A. Einstein, B. Polodolsky and N. Rosen describing the EPR paradox [6]. The flaw in EPR arguments was discovered by J. S. Bell [7], who proved that the principle of locality used in EPR paradox, was not consistent with the hidden variables interpretation of quantum theory. Bell's theorem conversely provides one of the possible methods to test whether two or more particles form an entangled state. The strong correlations among entangled

DOI: 10.14738/tmlai.71.5711
Publication Date: 20th January, 2019
URL: http://dx.doi.org/10.14738/tmlai.71.5711

particles can be used as resources of quantum cryptography [8], quantum teleportation [9] and quantum computation [10].

It is known that quantum correlation in qudit or multipartite system is stronger than in the bipartite system. The investigations performed hitherto have already shown that entanglement among more than two particles is not merely an extension of its bipartite counterpart but has instead several new and different properties which are more advantageous, as example, in super dense coding, quantum cloning, teleportation [11]. Koike *el al.* [12] have already demonstrated experimentally "1→2 quantum telecloning" for optical coherent states. To study QI at more fundamental levels, the multipartile entangled states, in addition, are also needed. It has been suggested [13] that the use of qudit can reduce the number of qubits by a function of $\log_2 d$.

In section **2**, we briefly describe the density operator (DO) and states for DPI of an atom. This operator corresponds to the case when the ionizing electromagnetic radiation is in a pure state of polarization and the target atom is in its ground state before DPI. In section **3**, we study the entanglement in DPI for qubits and qudit system. A quantitative application of entangled properties in DPI of xenon is presented in section **4**. Finally section **5** contains the conclusion part.

2 Preliminaries

2.1 Density Operators:

Let us represent by e_1 and e_2 the two freely moving electrons whose entanglement properties we want to investigate. The propagation vector of the *i* (=1, 2)-th electron is $\vec{k}_i=(k_i,\theta_i,\phi_i)$ such that its kinetic energy is given by $\varepsilon_i=\hbar^2 k_i^2/2m$. Also, $\mu_i(=\pm 1/2)$ represents the projection of the spin angular momentum of *i*-th electron along its spin quantization direction $\hat{u}_i=(\alpha_i,\beta_i)$ .These two electrons form an integral part of an atom A are assumed to be simultaneously ejected from it following the absorption of a single photon. If A^{2+} denotes the residual dication, then our process can schematically be represented by

$$h\nu_r\,(|l_r|=1, m_r)+A\,|0\rangle \rightarrow A^{2+}\,|f\rangle + e_1(\vec{k}_1;\mu_1\hat{u}_1)+e_2(\vec{k}_2;\mu_2\hat{u}_2). \tag{1}$$

Here $E_r=h\nu_r$ andb $|l_r|$ =1are, respectively, the energy and the angular momentum of the absorbed photon.Let us denote by $\rho_0=|0\rangle\langle 0|$ and $\rho_r=|m_r\rangle\langle m_r|$ the respective density operators of the unpolarized atom A before DPI and of the ionizing radiation. This means the density operator of the combined (atom + photon) system of Eq. (1) is given by the direct product

$$\rho_i=\rho_0\otimes\rho_r. \tag{2}$$

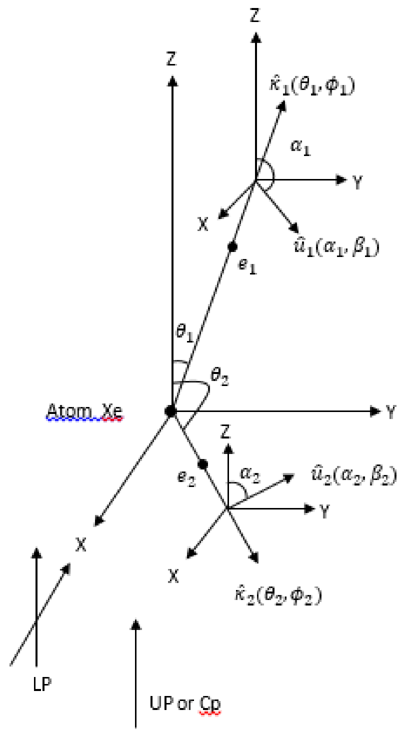


Figure 1. Two electrons are emitted simultaneously after photoabsorption.

Let us denote by F_p the photoionization operator in the $E1$ approximation. Then the density operator of the combined ($A^{2+}+e_1+e_2$) system in equation (1) after DPI becomes

$$\rho_f = K_p F_p \rho_i F_p^+ . \tag{3}$$

Here, $K_p = 3\pi(e^2/\alpha_0 E_r)^2$ with α_0 the dimensionless fine structure constant [14].

2.2 Criteria and measures for entanglement and mixedness of a state

Only recently, quantum entanglement is recognized as resources with various applications such as quantum teleportation and quantum key distribution in the field of quantum information and quantum computation. So, the task of characterizing and quantifying entanglement has emerged as one of the prominent challenge of quantum information theory. The quantum entanglement for multipartite states poses even a greater challenge. The condition for separability given by Peres [15] and Horodecki family [16] is that the partial transpose (PT), with respect to either of the two particles, of its DM remains positive. We have applied Peres-Horodecki positive partial transpose (PPT) condition in order to characterization of entanglement. As a measure of the degree of entanglement we consider negativity [4, 17]

As a measure we consider the *negativity*, which is an additive and operational measure of entanglement. Additivity is a very desirable property that can reduce computation of entanglement. It can also be generalized to multipartite and higher dimensional entanglement. The negativity of a state [18] indicates to which extent a state violates the positive partial transpose separability criterion. The negativity is twice [16] the absolute value of the sum of negative eigenvalues:

$$N = 2\max(0, -\lambda_{neg}) \tag{4a}$$

where λ_{neg} is the sum of the negative eigenvalues of $\rho_{AB}^{T_B}$, is the partial transpose of ρ_{AB}.

For higher dimensional mixed states, the negativity [19], however, can be defined as

$$N = \frac{\left\|\rho_{AB}^{T_B}\right\|_1 - 1}{d-1} \tag{4b}$$

for the states in a $d \otimes d'$ $(d \le d')$ quantized system and $\|.\|_1$ is the trace norm.

3 Entanglement between Two Electronic Qubits and an Ionic Qudit for DPI

3.1 Density matrix

Here, we calculate the DM for the angle- and spin-resolved DPI of an atom without considering SOI (L-S coupling) into account in either of the bound electronic states of A and A^{2+} or in the continua of the two photoelectrons (e_1, e_2) ejected in the process (1). As LS-coupling is applicable here, the total orbital angular momenta ($\vec{L}_0, \vec{L}_f$) and the spin angular momenta ($\vec{S}_0, \vec{S}_f$) of A and A^{2+} are the conserved quantities. If, the orbital angular momentum of the photoelectron e_1 is $\vec{l}_1$ and that of e_2 is $\vec{l}_2$ with their spin angular momenta $(\frac{1}{2})_1$ and $(\frac{1}{2})_2$ respectively, we then have

$$\vec{L}_0 + \vec{l}_r = \vec{L}_f + \vec{l}\,(= \vec{l}_1 + \vec{l}_2)$$

and

$$\vec{S}_0 = \vec{S}_f + \vec{s}_t\left[= (\frac{1}{2})_1 + (\frac{1}{2})_2\right].$$

Here we use the symbols $M_{L_0}, M_{S_0}, M_{L_f}$ and M_{S_f} to represent the respective projections of $\vec{L}_0, \vec{S}_0, \vec{L}_f$ and $\vec{S}_f$ along the polar axis of the space frame. In equation (1), the bound electronic state of atom A is $|0\rangle \equiv |L_0 S_0 M_{L_0} M_{S_0}\rangle$ and that of the dication A^{2+} is $|f\rangle \equiv |L_f S_f M_{L_f} M_{S_f}\rangle$. The density operator (2) for the combined (atom + photon) system can be written as

$$\rho_i = \frac{1}{(2L_0+1)(2S_0+1)} \sum_{M_{L_0} M_{S_0}} |0; 1m_r\rangle\langle 0; 1m_r|, \tag{5}$$

where we have defined $|0; 1m_r\rangle \equiv |0\rangle|1m_r\rangle$.

In order to calculate the DM for the ($A^{2+}+e_1+e_2$) system in process (1), we need to calculate the matrix elements ρ_i and ρ_f. Following the procedures given in reference [20] the matrix elements in the present case are given by

$$\left\langle f;\vec{k}_1,\mu_1\hat{u}_1;\vec{k}_2,\mu_2\hat{u}_2\middle|\rho_f\middle|f;\vec{k}_1,\mu'_1\hat{u}_1;\vec{k}_2,\mu'_2\hat{u}_2\right\rangle = \frac{K_p}{(2L_0+1)(2S_0+1)}\sum_{M_{L_0}M_{S_0}}\left\langle f;\vec{k}_1,\mu_1\hat{u}_1;\vec{k}_2,\mu_2\hat{u}_2\middle|F_p\middle|0;1m_r\right\rangle \times \left\langle f;\vec{k}_1,\mu'_1\hat{u}_1;\vec{k}_2,\mu'_2\hat{u}_2\middle|F_p\middle|0;1m_r\right\rangle^* \tag{6}$$

The DM of the tripartite system for angle- and spin-resolved DPI process (1) in the absence of SOI can be written in the following form:

$$\left\langle f;\vec{k}_1,\mu_1\hat{u}_1;\vec{k}_2,\mu_2\hat{u}_2\middle|\rho_f\middle|f;\vec{k}_1,\mu'_1\hat{u}_1;\vec{k}_2,\mu'_2\hat{u}_2\right\rangle = \frac{d^3\sigma(m_r)}{d\varepsilon_1 d\hat{k}_1 d\hat{k}_2} \times \sigma(S_0;S_f;\hat{u}_1,\hat{u}_2)_{M_{S_f}\mu_1\mu_2;M'_{S_f}\mu'_1\mu'_2}. \tag{7}$$

The first term i.e., the triple differential cross section (TDCS i.e., $d^3\sigma(m_r)/d\varepsilon_1 d\hat{k}_1 d\hat{k}_2$) on the right-hand side of (7) depends upon the orbital angular momenta of *A* and A^{2+}, phase shifts, energies ($\varepsilon_1,\varepsilon_2$) and the directions ($\hat{k}_1,\hat{k}_2$) of the emitted electrons (e_1, e_2), the state of polarization (m_r) of the ionizing radiation and the photoionization dynamics. It does not include spins of the photoelectrons or the target atom or the residual dication. Thus, the TDCS $d^3\sigma(m_r)/d\varepsilon_1 d\hat{k}_1 d\hat{k}_2$ in the DM (7) describes angular correlation between the photoelectrons and the residual ion. Its value is always positive. Here,

$$\frac{d^3\sigma(m_r)}{d\varepsilon_1 d\hat{k}_1 d\hat{k}_2} = (-1)^{m_r+L_0+L_f}\frac{K_p}{4\pi(2L_0+1)}\sum_{\substack{l_1 l_2 l \\ l'_1 l'_2 l' L L' L_r}}\sum_{L_1 L_2 M}(-1)^{l'_1+l'_2+l'}(2L_r+1) \times\sqrt{(2L_1+1)(2L_2+1)}\begin{pmatrix} l_1 & l'_1 & L_1 \\ 0 & 0 & 0\end{pmatrix}\begin{pmatrix} l_2 & l'_2 & L_2 \\ 0 & 0 & 0\end{pmatrix}\begin{pmatrix} 1 & 1 & L_r \\ m_r & -m_r & 0\end{pmatrix}\begin{pmatrix} L_1 & L_2 & L_r \\ M & -M & 0\end{pmatrix}\begin{Bmatrix} 1 & 1 & L_r \\ L' & L & L_0\end{Bmatrix} \times\begin{Bmatrix} l & l' & L_r \\ L' & L & L_f\end{Bmatrix}\begin{Bmatrix} l_1 & l_2 & l \\ l'_1 & l'_2 & l' \\ L_1 & L_2 & L_r\end{Bmatrix} Y_{L_1}^{M}(\hat{k}_1)Y_{L_2}^{-M}(\hat{k}_2)\langle L_f l|F(L)|L_0 1\rangle\langle L_f l'|F(L')|L_0 1\rangle^*. \tag{8}$$

The second term (i.e., $\sigma(S_0;S_f;\hat{u}_1,\hat{u}_2)_{M_f\mu_1\mu_2;M'_f\mu'_1\mu'_2}$) is the spin-correlation matrix. It completely determines the properties of the Coulombic entanglement among (e_1, e_2 and A^{2+}). It can be written as

$$\sigma(S_0;S_f;\hat{u}_1,\hat{u}_2)_{M_{S_f}\mu_1\mu_2;M'_{S_f}\mu'_1\mu'_2} = (-1)^{\mu_1+\mu'_2+M'_{S_f}}(2S_f+1)\sum_{sQm_1m_2n}(-1)^{Q-n}(2s+1)(2Q+1) \begin{pmatrix} \frac{1}{2} & \frac{1}{2} & s \\ \mu_1 & -\mu'_1 & m_1\end{pmatrix}\begin{pmatrix} \frac{1}{2} & \frac{1}{2} & s \\ \mu_2 & -\mu'_2 & m_2\end{pmatrix}\begin{pmatrix} S_f & S_f & Q \\ M_{S_f} & -M'_{S_f} & n\end{pmatrix}\begin{pmatrix} s & s & Q \\ m_1 & m_2 & n\end{pmatrix}\begin{Bmatrix} \frac{1}{2} & \frac{1}{2} & s \\ \frac{1}{2} & \frac{1}{2} & s \\ S_f & S_f & Q\end{Bmatrix} D^{s*}_{m_1 n}(\omega_1)D^{s*}_{m_2 -n}(\omega_2), \tag{9}$$

where *D*s are the rotational harmonics [21] with $\omega_1(\alpha_1,\beta_1,0)$ and $\omega_2(\alpha_2,\beta_2,0)$ the Euler angles which rotate the axis of the space-frame into the spin-polarization directions $\hat{u}_1$ and $\hat{u}_2$ (Fig. 1), respectively.

In expression (9) the allowed values for each of μ_1, μ'_1, , μ_2 and μ'_2 are $\pm\frac{1}{2}$, the spin magnetic quantum numbers M_{S_f} and M'_{S_f} of the photoion A^{2+} can take $(2S_f+1)$ values. So, DM (9) of size $[2(2S_f+1) \times 2(2S_f+1)]$ represents spin correlations between the electronic qubits (e_1 ,e_2) and d[= $(2S_f+1)$]-dimensional ionic qudit A^{2+}. Hence, the dimensionality d of the qudit can be chosen according to the appropriate spins S_0 and S_f of the atomic target *A* and photoion A^{2+}, respectively.

The $d^3\sigma(m_r)/d\varepsilon_1 d\hat{k}_1 d\hat{k}_2$ term in (7) represents the angular correlation of two qubits-qudit system and always has a positive value. Therefore, unless stated otherwise, we write the DM (7) as

$$\left\langle f;\vec{k}_1,\mu_1\hat{u}_1;\vec{k}_2,\mu_2\hat{u}_2\right|\rho_f\left|f;\vec{k}_1,\mu'_1\hat{u}_1;\vec{k}_2,\mu'_2\hat{u}_2\right\rangle=\sigma(S_0;S_f;\hat{u}_1,\hat{u}_2)_{M_{S_f}\mu_1\mu_2;M_{S_f}\mu'_1\mu'_2} \quad (10)$$

In order to study the entanglement properties for DPI in (1), we need to calculate partial transpose (PT) [14, 15] of the DM (10) with respect to either of three subsystems, i.e., photoelectrons (e_1, e_2) and photoion A^{2+}. These are given by

$$\sigma^{T_{A^{2+}}}=\sigma(S_0;S_f;\hat{u}_1,\hat{u}_2)_{M'_{S_f}\mu_1\mu_2;M_{S_f}\mu'_1\mu'_2}, \quad (11a)$$

$$\sigma^{T_{e_1}}=\sigma(S_0;S_f;\hat{u}_1,\hat{u}_2)_{M_{S_f}\mu'_1\mu_2;M'_{S_f}\mu_1\mu'_2} \quad (11b)$$

and

$$\sigma^{T_{e_2}}=\sigma(S_0;S_f;\hat{u}_1,\hat{u}_2)_{M_{S_f}\mu_1\mu'_2;M'_{S_f}\mu'_1\mu_2}. \quad (11c)$$

3.2 Tripartite Entanglement

We have also studied tripartite entanglement properties between ionic qudit and electronic qubit system. The tripartite entanglement can be converted into two stand forms [4. 11] namely $\left|\mathrm{GHZ}\right\rangle$ and $\left|\mathrm{W}\right\rangle$ states. We use $\left|\mathrm{W}\right\rangle$ state as it is more suitable for mixed state entanglement and can be written as

$$\left|\mathrm{W}\right\rangle=\frac{1}{\sqrt{3}}\left(\left|001\right\rangle+\left|010\right\rangle+\left|100\right\rangle\right). \quad (12)$$

We calculate tripartite entanglement using equations (10), (11) and (12) for the geometry where each of the photo electrons is polarized in X-Y plane $(\beta_1 \text{ and } \beta_2=\pi/2)$ of the space frame.

(i) We have first studied the case when $[S_0=S_f-1]$ with $S_f\geq 1$. Without specifying the spin S_f we are not able to diagonalize the DM. We have calculated the DM (10) and its partial transpose using equation (11) for each of the $S_f=(1,\frac{3}{2},2,\frac{5}{2},3)$. For quantifying entanglement we have used negativity as

$$N[\sigma(S_0=S_f-1;S_f\geq 1)]=\frac{\sqrt{6}}{2S_f+1}\text{ with } S_f\geq 1. \quad (13)$$

We can see from Eq. (13) that the tripartite system is partially entangled for all values of $S_f\geq 1$ and the value of negativity ranges from 0 to $\sqrt{2/3}$, decreasing with the increasing value S_f, the spin of ionic qudit A^{2+}.

(ii) Next we study the case for $S_0=S_f+1$ with $S_f\geq 1$. We have calculated the DM (10) and its partial transpose using equation (11) for each value of the $S_f=(1,\frac{3}{2},2,\frac{5}{2},3)$. In this case we have seen that none of the eigenvalues of the partial transpose (11) is negative. So, the states for

$[S_0 = S_f + 1]$ with $S_f \geq 1$ have positive partial transpose (PPT). So, the states for $[S_0 = S_f + 1]$ with $S_f \geq 1$ may possess bound entanglement if $4(2S_f + 1) > 6$, i.e. $S_f \geq 1$ and obeys the inequality [4, 13]

$$\mathrm{Rank}[\sigma(S_0 = S_f + 1; S_f \geq 1)] \geq \mathrm{Max}\{\mathrm{Rank}[\sigma(e_1)], \mathrm{Rank}[\sigma(e_2)], \mathrm{Rank}[\sigma(A^{+2})]\} \qquad (14)$$

In this case, $\mathrm{Rank}[\sigma(S_0 = S_f + 1; S_f \geq 1)]$ is $4(2S_f + 1)$, $\mathrm{Rank}[\sigma(e_1)]$ is 2, $\mathrm{Rank}[\sigma(e_2)]$ is 2 and $\mathrm{Rank}[\sigma(A^{+2})]$ is $(2S_f + 1)$. So, the states for $[S_0 = S_f + 1]$ with $S_f \geq 1$ obey the condition (14) and hence possess bound entanglement.

4 Example for Tripartite Entanglement In DPI For Xenon

As an application for the case $|S_f - S_0|$=0 in in xenon atom. The DPI process in the ground electronic state of xenon atom is the following:

$$h\nu_r + Xe(4d^{10}5s^2 5p^{6\ 3}P_0) \rightarrow Xe^{2+}(4d^9 5s^1 5p^{6\ 1}P_0) + e_1(\vec{k}_1; \mu_1 \hat{u}_1) + e_2(\vec{k}_2; \mu_2 \hat{u}_2). \qquad (15)$$

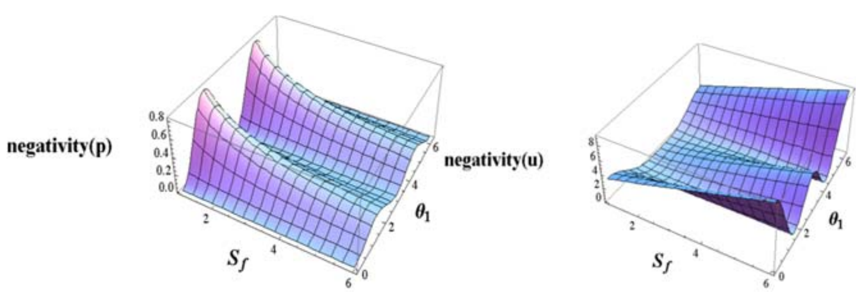
We use the values of TDCS of Xenon given in Ref. [22] for photon energy 1eV for the geometry $\theta_{12} = 180^0$,where θ_{12} is the angle between two ejected electrons. The TDCS are

$$\left.\frac{d^3\sigma(\theta_{12} = 180^0)}{d\varepsilon_1 d\hat{k}_1 d\hat{k}_2}\right|_p = 4a^2\, \mathrm{E}^{n-\frac{3}{2}} \sin^2\theta_1 \qquad (16a)$$

And

$$\left.\frac{d^3\sigma(\theta_{12} = 180^0)}{d\varepsilon_1 d\hat{k}_1 d\hat{k}_2}\right|_u = 4a^2\, \mathrm{E}^{n-\frac{3}{2}} (1 + \cos^2\theta_1) \qquad (16b)$$

for polarized and unpolarized photons respectively. The variation of Negativity entropy with respect to the direction of ejection and spin polarization of the photoelectrons is shown below



(a) (b)

Figure 2. Variation of Negativity with respect to S_f and θ_f

From figure 2 we see that depending on direction of ejection of photoelectrons and state of polarization of incident photons for $\left|S_f - S_0\right|$=0 in xenon atom are partially entangled and unentangled and magnitude of negativity is also effected by TDCS.

5 Conclusion

Entanglement between a pair of particles is a very delicate thing and is easily destroyed. Currently, many scientists are working on making stronger systems where entanglement is stronger and lasts longer to use for quantum computation and information more easily. Photons are, probably the fastest and best carriers of information but they are not suitable for storing information for a long period of time [23]. Ions (positive or negative) and electrons are stable, easily detectable and capable of storing information for a long time. Electrons are the lightest particles next to photon therefore able to travel with speeds comparable with light. Furthermore, electrons have since long been used to carry information in the form of electric signals and /or for storing it for a long time. It was already suggested [24] that electron spin can be used for quantum computation [1].

DPI is the most natural processes for simultaneously producing two electrons in continuum in a single step. It is the most direct manifestation of electron- electron correlation in an atom, because had the independent particle model been valid, two electrons would not have emerged simultaneously following the absorption of a single photon. Thus, simultaneous ejection of two electrons from an atom, after the absorption of a single photon, could take place due to the existence of correlation effects between them. Here we have tried to show that the DPI process is a powerful tool for investigations of tripartite entanglement between two electronic qubits and an ionic qudit. The entanglement is quantified by Peres-Horodecki's NPT condition and negativity. These bipartite and tripartite systems may be pure or mixed and may possess free or bound entanglement[25]. They may be totally entangled, partially entangled or separable depending on spin states of target *A* and residual dication A^{2+} as well as of the directions spin quantization and ejection of the photoelectrons [26].

A quantitative case is studied for DPI in Xenon atom and has been shown that the TDCS effects entanglement. For Xenon atom we have studied the states for $\left|S_f - S_0\right|$=0 and have shown that depending on the direction of ejection of photoelectrons and state of polarizations, the states are totally entangled, partially entangled and unentangled.

REFERENCES

[1] *Quantum Computation and Quantum Information*, M.A. Nielsen and I. L. Chuang (Cambridge University Press, Cambridge, England, 2000).

[2] *Quantum Information: An Introduction to basic Theoretical Concepts and Experiments,* edited by G. Alber, T. Berth, M. Horodecki, P. Horodecki, R. Horodecki, M. Rötteler, H. Weinfurter, R. Werner and A. Zeilinger (Spinger, Berlin, 2000) .

[3] B. Shumacher, *Phys. Rev. A* **51**, 2738 (1995).

[4] M. Horodecki, P. Horodecki and R. Horodecki, Rev. Mod. Phys. **81**, 865 (2009).

[5] P. Rungta et al., in Directions in Quantum Optics: A Collection of Papers Dedicated to the Memory of Dan Walls, edited by H. J. Carmichae et al., (Spinger, Berlin, 2001), pp. 149-164.

[6] A. Einstein, B. Polodolsky and N. Rosen, Phy. Rev. **47,** 777 (1935).

[7] J. S. Bell, Physics **1,** 195 (1965).

[8] A. Ekert, Phys. Rev. Lett. **67**, 661 (1991).

[9] C. H. Bannett, C. Brassard, R. Jozsa, A. Peres and W. K. Wooters, Phys. Rev. Lett. **70**, 1895 (1993).

[10] A. Ekert and. R. Jozsa, Rev. Mod. Phys. **68**, 733 (1998).

[11] S. Parida and N. Chandra, Phys. Rev. A **86,** 062302 (2012).

[12] S. Koike, H. Takahashi, H. Yonezawa, N. Takei, S. L. Braunstein, T. Aoki and A. Furusawa, Phys. Rev. Lett. **96,** 060504 (2006).

[13] A. Galindo and M.A. Martin-Delgado, Rev. Mod. Phys.**74**, 347 (2002).

[14] C.H. Bennett, D. P. DiVincenzo, J. Smolin, and W. K. Wootters, Phys. Rev. A 54, 3824 (1996).

[15] A. Peres, Phys. Rev. Lett. **77**, 1413 (1996).

[16] M. Horodecki, P. Horodecki and R. Horodecki, Phys. Rev. A. **223,** 1, (1996).

[17] T.C. Wei, K. Nemoto, P. M. Goldbart, P. G. Kwiat, W. J. Munro and F. Verstraete, Phys. Rev. A **67**, 022110 (2003).

[18] S. Ishizaka and T. Hiroshima, Phys. Rev. A 62, 022310 (2000).

[19] W. K. Wootters, Phys. Rev. Lett. **80**, 2245 (1998).

[20] N. Chandra and M. Chakraborty, J. Phys. B: At. Mol. Opt. Phys. **35,** 2219 (2002).

[21] *Angular Momentum in Quantum Mechanics,* A. R. Edmonds (Princeton University press, Princeton, 1974).

[22] A Huetz, P. Selles, D. Waymel and J. Mazeau, J. Phys. B **24** ,1917 (1991).

[23] *Quantum Entanglement in Electron Optics*, N. Chandra and R. Ghosh, (Springer-Verlag, Berlin, 2013).

[24] D. Loss, D. P. DiVicenzo, Phys. Rev. A **57**, 120 (1998).

[25] M. Chakraborty and S. Sen, Journal of Electron Spectroscopy and Related Phenomena,**203**, 60 (2015).

[26] M. Chakraborty, Indian Journal of Science and Technology, **10**, 27 (2017).

SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
Volume 7 No. 1
ISSN 2054-7390

Design of Weather Monitoring Sensors and Soil Humidity in Agriculture Using Internet of Things (IoT)

Yoga Alif Kurnia Utama[1], Yonatan Widianto[2], Yulius Hari[3], Muhammad Habiburrahman[4]
Engineering Faculty, Widya Kartika University[1,2,3,4]
yoga.alif@widyakartika.ac.id[1], yonatan@widyakartika.ac.id[2], yulius.hari.s@widyakartika.ac.id[3], habibsantri@gmail.com[4]

ABSTRACT

One of the developments in internet technology at the moment is the Internet of things (IoT). IoT is a concept that aims to expand the benefits of internet connectivity that is connected continuously and can monitor a system or object. As for capabilities such as data sharing, remote control, including monitoring a system. Because Indonesia is an agricultural country, this study tries to make a monitoring of a smart farmer system that helps people monitor plants on their farms. This system will monitor weather parameters such as temperature and humidity conditions, as well as soil moisture conditions. If soil moisture is reduced due to heat, then water will automatically be pumped and flow to the plant. Farm owners can also monitor the condition of their plants in real time remotely through the Blynk application media, Facebook website, and Twitter. Based on the test results, the system can measure the environmental conditions of plants, can connect to the internet and be able to send data from sensor measurements to the Blynk application, Facebook website and Twitter. With the development of these systems and devices, it is hoped that the farmers or the community can monitor the environmental conditions of the plants

Keywords: Internet of things (IoT), Smart Farmer, Weather Sensors and Soil Moisture, Microcontroller**.**

1 Preliminary

The rapid development of science and technology has had a great impact on human life. In the development of technology, electronics and computers, effectiveness and efficiency are the references in the use and use of technology so that it can achieve optimal results. To be able to realize this, a device and a system are needed that can process data accurately. Along with the increasing needs of the community in the fields of agriculture and plantations, a system that is expected to facilitate and improve effectiveness in farming or gardening is needed. One of them is currently developing rapidly is the internet of things (IoT). The main challenge in IoT is to facilitate the exchange of data between the physical world (hardware) and the world of information such as sending data obtained from electronic equipment through an interface between the user and the equipment**.**

The sensor collects physical raw data from real-time checking and converting it into understandable format machines so as to facilitate the exchange of data between various forms of data formats (Thing) [3]. IoT appears as a big issue on the Internet, it is expected that billions of things (hardware) or objects will be equipped with various types of sensors that can connect to the internet through networks and

DOI: 10.14738/tmlai.71.5613
Publication Date: 29th January, 2019
URL: http://dx.doi.org/10.14738/tmlai.71.5613

technology support, such as embedded sensors and actualization, in some cases real-time data flow will automatically be generated by connected things and sensors. Of all the activities in IoT, that is to collect the correct raw data in an efficient way, but more important is to analyze and process raw data into more valuable information [4]. In the field of agriculture and plantations where the concept of the internet of things is expected to reduce human work in several ways such as irrigation, temperature detection, controlling humidity which was originally done simply and manually, can be replaced by more thorough systems and devices.

Indonesia is one of the countries in the equator. Because it is crossed by the equator, the climate in Indonesia includes a tropical climate. This tropical climate makes agriculture one of the most superior commodities in Indonesia. Therefore it is not surprising that the majority of the population works in agriculture, fisheries, and plantations. The community involved in this field, especially agriculture and plantations, in the processing and maintenance of almost all plants manually starting from irrigation, pest eradication, providing fertilizers and medicines, while to know the quality of the soil and nutrients needed by the plant itself cannot be known to farmers. Therefore, the research tries to help the community, especially farmers, by making a device consisting of weather sensors and soil moisture in order to help farmers monitor their agricultural results in real time because the sensor data is sent and displayed on the blynk application on mobile phones. Android, and can be checked through the website, Facebook and Twitter.

2 Literature Review

Internet of things (IoT) is a concept that aims to expand the benefits of continuously connected internet connectivity. As for capabilities such as data sharing, remote control, etc., including objects in the real world. For example, food, electronics, any collection of equipment including living things that are all connected to local and global networks through embedded and always active sensors, the term internet of things was originally conceived by Kevin Ashton in 1999**.**

The main challenge in IoT is to facilitate the exchange of data between the physical world (hardware) and the information world. Like how to process data obtained from electronic equipment through an interface between the user and the equipment. The sensor collects physical raw data from real-time scenarios and converts it into understandable formatting machines so that it will be easily exchanged between various forms of data formats (Thing) [3].

IoT appears as a big issue in the world of communication and information, it is expected that billions of physical things or objects will be equipped with various types of sensors that can be connected to the internet through networks and technology support, such as embedded sensors and actualization, in some cases real data flow -time will be automatically generated by connected things and sensors. Of all the activities in IoT, that is to collect the correct raw data in an efficient way, but more important is to analyze and process raw data into more valuable information [4].

Smart Farmer is a prototype that will be made with electronic components consisting of several sensors needed in the agricultural field and the NodeMcu microcontroller. This device can connect with users through internet connectivity and transmit data from these environmental conditions, the concept is included in the concept of the internet of things.

Data obtained from the sensor will be sent periodically and read on the Blynk application and posted on Facebook and Twitter sites. Apart from reading data from sensors, there will be a condition where a moisture sensor checks soil moisture, if soil moisture is lacking, or the soil conditions are too dry from the specified standard it will be checked by the system and will pump water to sufficient water content or soil moisture in the plant.

In making this smart farmer, the required components include microcontroller NodeMcu, Digital Humidity And Temperature 11 (DHT11) sensor, and soil moisture sensor (soil moisture sensor), the advantage of using NodeMcu is the wifi module integrated in its components, making it very easy sending data to the server via the internet network, besides that there is a Blynk application that will be used by the author to monitor plant environment data, this application is an Android smartphone application that can be configured as needed.

Smart farmer is included in the information technology section where generally there are input, process and output flows, as in the diagram below. In this case, what acts as an input is the sensor, the DHT11 sensor, and the soil moisture sensor, then the processing part is the microcontroller NodeMcu, then the output is the data display on the Blynk application and posting the status on the Twitter website and Facebook.

3 Method

The device made in this study refers to the block diagram in Figure 1 below. This tool consists of a weather sensor in the form of a DHT 11 sensor that will measure temperature and humidity. In addition, there is a soil moisture sensor that will measure the level of soil moisture on agricultural land. These sensor data will determine how long the water pump will turn on to irrigate plants and nutrients. The water pump is turned on by using a relay so the water pump can be turned off and turned on automatically. In reading the sensor and controlling the actuator in the form of a water pump, a Nodemcu microcontroller is needed. This microcontroller was chosen because it can be directly connected to the internet network. The internet network is used to send sensor data and actuator status to an android application called Blynk and several websites namely Facebook and Twitter.

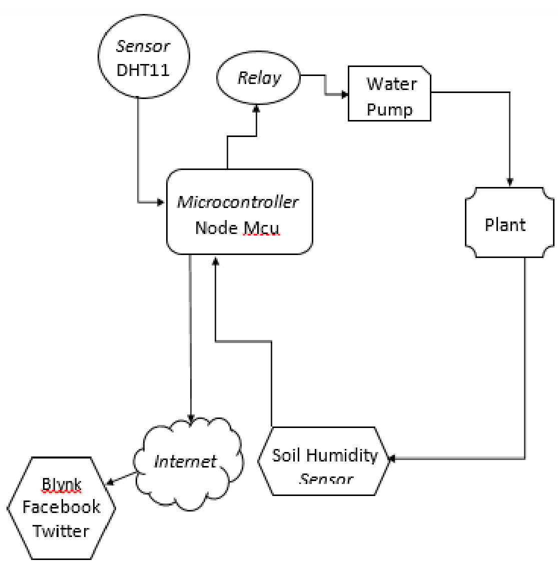


Figure 1. Block Tool Diagram

In designing a tool with the appropriate function shown in the block diagram in Figure 1, several sequences are needed. The series is described as follows:

1. DHT 11 Sensor Series

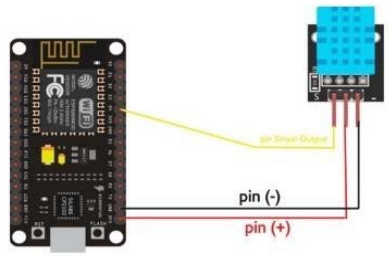


Figure 2. DHT11 and NodeMcu Sensor Circuits

DHT 11 sensor has 3 pins, pin (+) connected to 3v3 NodeMcu power output, then pin (-) connected to GND (Ground) NodeMcu, finally, S pin on DHT11 is connected to pin D4 on NodeMcu which functions as a receiver of value data from sensor DHT11.

2. Sensor Soil Moisture Series

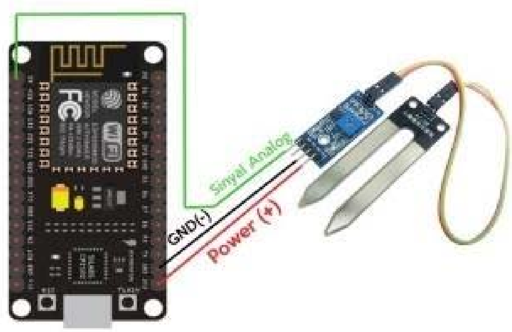


Figure 3. NodeMcu circuit with Sensor Soil Moisture

Soil moisture sensor is used to measure the humidity on the ground, the circuit as shown above.

3. Pump Circuits

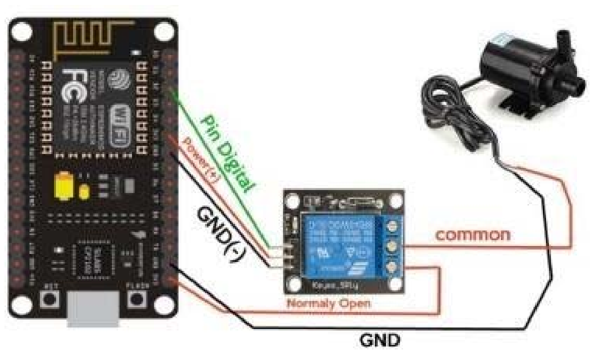


Figure 4. NodeMcu circuit with relay and pump

The circuit aims to activate and deactivate the pump, Relay is used to adjust the condition of ON and OFF of the pump. Writing program code using the LUA language, NodeMcu also supports writing in C, the same as used on Arduino. To run the series, some programming syntax must be made. Some of the syntax in the program that has been made is described as follows:

1. Device Declaration

In this first step, we define the Blynk application via #define BLYNK_PRINT Serial by writing #include <BlynkSimpleEsp8266.h>, then the Wifi module to communicate data over the internet. The writing of the line of code is as follows.

```
//Blynk dan ESP
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
```

2. DHT Sensor Declaration 11

Defines the variables needed to read the DHT11 sensor, calls module #include "DHT.h", then defines pin #define DHTPIN 2, pin 2 or GPIO 2 equals D4, for the NodeMcu pin schema, #define DHTTYPE DHT11 defines the sensor the DHT11 sensor is used. Following is a fragment of the DHT11 code.

```
//DHT SENSOR
#include "DHT.h"
#define DHTPIN 2
#define DHTTYPE DHT11
#define DHT_INDIKATOR 12 DHT
dht(DHTPIN, DHTTYPE); float
kelembabanUdara = 0; float
suhuUdara = 0;
```

3. Declaration of Humidity Sensor

The code line below defines which pins are used to read soil moisture sensors or soil moisture sensors.

```
//SOIL SENSOR #define soil_pin
A0 int kelembabanTanah = 0;
```

4. Declaration of Pump Relays

The pump functions to irrigate if the soil lacks water, in the code line pin 4 or GPIO 4 means pin D2 to connect the NodeMcu to the relay input pin, then the relay is connected to the pump.

```
//RELAY
//#define POMPA_PIN 12
#define   POMPA_PIN   4   #define
TIME_PUMP_ON  7 boolean status
Pompa = 0;
```

5. Timer Declaration

The timer is used to read data periodically in settings using milliseconds, so the line of code above defines the timer.

```
//TIMER
#include <SimpleTimer.h>
#define BACA_DATA_DHT_SOIL 2
#define KIRIM_DATA_SENSOR 1
#define AUTO_CONTROLL 6
SimpleTimer timer;
```

6. Variable Declarations and Auto Control Values

In the variable below, it is defined that the drought limit is if the dry level is less than 30, and the wet limit is at level 80.

```
//VALUE AUTO CONTROL SOIL SENSOR
#define KERING 30
#define BASAH 80
```

7. Declaration of Wifi Connection

In the code line below, defined access point information that will be addressed as an internet server along with a token or auth code provided by Blynk when registering, to be able to be accessed by the Blynk application.

```
//KONEKSI
char auth[] = "dc18d91e662549149e7c5bf4b5c9781c"; char
ssid[] = "LAB ELEKTRO";
char pass[] = "elektro123";
```

8. Function Setup

In the code line below it functions as the basic setup of the program as needed, there is Serial.begin (9600) to activate the display of the NodeMcu terminal on port 9600. Serial.println ("Smart Farmer"), Serial.println ("... Ready Application") will appear first when activating the terminal, then call the Blynk function from the Blynk module to connect to the internet, in the Blynk.begin code line (auth, ssid, pass) auth, ssid, and pass, dht.begin to activate the dht sensor reading, then define pinMode for soil moisture sensors and DHT sensors as OUTPUT, and finally start process to call the start process function which functions to call a collection of functions that have been joined to the start process function.

```
void setup(){   Serial.begin(9600);
delay(10);
  Serial.println("Smart Farmer");
  Serial.println(".... Aplikasi ready");

  Blynk.begin(auth, ssid, pass);   dht.begin();

  pinMode(soil_pin, OUTPUT);   pinMode(POMPA_PIN,
OUTPUT);
```

```
  startProses();
}
```

9. Function Start Proses

The start process function functions to collect all the functions created manually according to the needs and then join with the timer to update data periodically.

```
void startProses()
{
  timer.setInterval(BACA_DATA_DHT_SOIL*1000,          getDhtSoilData);
timer.setInterval(KIRIM_DATA_SENSOR*1000,           kirimDataSensor);
timer.setInterval(AUTO_CONTROLL*1000, kontrolOtomatis);
}
```

10. The function of the Pump Button from the Application

The above function is useful for activating the pump through the Blynk application. In the application a virtual button is made, where the button is useful to turn on the pump manually, the virtual button is declared with a virtual pin 6, therefore, so that the button on the application can be read in the program, the parameter number 6 is created in the BLYNK_WRITE line (6).

```
BLYNK_WRITE(6) // Pump remote control
{
  int i=param.asInt();
  if (i==1)
  {
    statusPompa = !statusPompa;     kontrolPompa();
  }
}
```

11. Function of reading DHT11 sensors

The function above is used to read the values of DHT11 sensors and soil moisture, to read the air temperature using dht.readTemperature () which is declared with the temperature variable Air, and dht .readHumidity () which is declared with variable humidity. then the statement failed to read the DHT sensor. To send notifications to social media, there is the Blynk.tweet class ("... .ome text"), besides being sent to social media, the information is displayed in the serial monitor provided by the NodeMcu with the command Serial.print ().

```
void getDhtSoilData(){   float lihatSuhu = suhuUdara;   float
lihatKelembaban = kelembabanUdara;   suhuUdara =
dht.readTemperature();   kelembabanUdara =
dht.readHumidity();   if (isnan(suhuUdara) ||
isnan(kelembabanUdara)){     Serial.println("Gagal Membaca
Sensor DHT!");     suhuUdara = lihatSuhu;
    kelembabanUdara = lihatKelembaban; //
digitalWrite(DHT_INDIKATOR,HIGH);     return;
  }else{
    Serial.print("Suhu Udara: ");
```

```
  Serial.println(suhuUdara);
  Serial.print("Kelembaban Udara: ");
  Serial.println(kelembabanUdara);
  Serial.println(" RH");   }
………………………………………………………………
Blynk.tweet(String("Kondisi Tanaman, Kelembaban Tanah: ") + kelembabanTanah + "
RH\n Suhu Udara: " + suhuUdara + " ºC \n Kelembaban Udara: " + kelembabanUdara + " RH");

}
```

12. The function of reading the soil moisture sensor

The DHT11 sensor readings with soil moisture are in the same function, namely the getDhtSoilData () function, the purpose is combined into one function to facilitate sending data to social media, the analogRead (soil_pin) function is useful for reading analog signals, because the humidity sensor used by researchers connected to an analog pin from NodeMcu. There is a map function (humidity, 400, 1024, 0, 100) which is also useful for configuring data readings because the sensors used by researchers read data between 100 to 1024. Serial.println function is used if you want to display text or sentences with new lines added or enter.

```
void getDhtSoilData(){
 -----------------------------------------------------   //Baca Soil
Moisture Sensor.
 kelembabanTanah = 0;
 delay (500);
 kelembabanTanah = analogRead(soil_pin);
 if(isnan(kelembabanTanah)){
  Serial.println("Gagal Membaca Sensor Kelembaban Tanah!");     return;
 }
 kelembabanTanah = map(kelembabanTanah, 400, 1024, 100, 0);
 Serial.print("kelembaban Tanah: ");
 Serial.println(kelembabanTanah);
 Serial.println(" RH");
```

13. Automatic Control Function

The function in the code line above is to check soil moisture.

```
void kontrolOtomatis(){    if
(kelembabanTanah < KERING)
 {
  hidupkanPompa();
 }
}
```

14. Function Turn on the Pump

In the above line of code, it is called and gives a value to the Pump variable status = 1, used as conditioning when pumping water, there is another function, which is Pump control () to turn on the pump which is given delay.


```
void hidupkanPompa(){   statusPompa =
1;   kontrolPompa();   delay
(TIME_PUMP_ON*1000);   statusPompa
= 0;   kontrolPompa();
}
```

15. Function Pump

The code line above is used to turn on the pump by writing digitalWrite (POMPA_PIN, HIGH). The line is checked if the Pump status is equal to 1, the pump is turned on with a value from digitalWrite = HIGH, then sends a notification to the Blynk application using Blynk.notify (), the notification sent to the blynk application will appear as written on the program. In the line of code, there is also Serial.print () which is useful for sending info to the serial monitor, if the Pump status variable is not equal to 1 then digitalWrite = LOW.

```
void kontrolPompa()
{
 if (statusPompa == 1)
 {
  digitalWrite(POMPA_PIN,HIGH);
  Blynk.notify("Peringatan!!, Kelembaban Tanah Berkurang, Pompa Air Dihidupkan");
  Serial.print("\n Peringatan!!, Kelembaban Tanah Berkurang, Pompa Air Dihidupkan");
Blynk.tweet("Peringatan!!, Kelembaban Tanah Berkurang, Pompa Air Dihidupkan");
 }   else
 {
  digitalWrite(POMPA_PIN,LOW);
  Blynk.notify("Kelembaban Tanah Sudah Cukup, Pompa Air Dimatikan!!");
  Serial.print("\n Kelembaban Tanah Sudah Cukup, Pompa Air Dimatikan!! \n");
Blynk.tweet("Kelembaban Tanah Sudah Cukup, Pompa Air Dimatikan!!");
 }
}
```

16. Function Send Sensor Data to Blynk

The Blynk application has a virtual pin to be able to read the value of the sensors used in this device, therefore to declare which pin to use, write the code as above Blynk.virtuaalWrite (0, temperature Air) means virtual pin 0 (V0) to read the air temperature. Next is a line of code to synchronize the sensor data and virtual pin of the Blynk application.

```
void kirimDataSensor()
{
 Blynk.virtualWrite(0, suhuUdara); //virtual pin V0
 Blynk.virtualWrite(1, kelembabanUdara); // virtual pin V1
 Blynk.virtualWrite(2, kelembabanTanah); // virtual pin V2 }
```

17. Function Loop

In writing the program code, there is one function that is useful to run the program repeatedly or continuously, this is the function of the loop, so functions that have been declared with various needs will be running continuously as long as the device is ON. Here is a line of code from the Loop function.

```
void loop(){     Blynk.run();
timer.run();
 }
```

4 Results and Discussion

After the tool is finished, the tool is tested. Testing this tool is divided into several dates on the NodeMcu monitor series, Blynk application and social media. Display of data that is raised, among others, air temperature data, air humidity data, soil moisture data, information about the activities of the pump that is active and is dead. The trial display of sensors on Facebook and Twitter can be seen in Figure 5 and Figure 6 below.

1. Trial Date 23 January 2018 Display of Facebook Website

Figure 5. Display of Facebook Website Information January 23.

In the above view, there is info sent from the device, about the current environmental conditions. Data is sent periodically every 2 seconds.

2. Trial January 23 Display Twitter Website.

Figure 6. Display of Twitter Website Information January 23.

5 Conclusion

Based on the implementation of making research ranging from analysis, design to the implementation stage, the author can draw conclusions as follows:

a) This device is very good for use in agriculture because it is very helpful for farmers to monitor and irrigate crops.
b) Farmers can control and monitor remotely because the NodeMcu microcontroller has a wifi device that can connect to the internet, so it can access data freely from anywhere.
c) The Blynk application is very helpful in IoT for creating monitoring applications.
d) In addition to opening the Blynk application, there are also notifications received through social media accounts, making it very easy to monitor.

REFERENCES

[1] Barr, Michael, and Anthony Massa. 2006. Programming Embedded Systems with C and GNU Development Tools. Sebastopol: O'Reilly Media.

[2] Chui, Michael, dkk. 2013. Ten IT-enabled Business Trend for The Decade Ahead. McKinsey Global Institute.

[3] Suresh, P., Daniel, J. V., & Aswathy, R. H. (2014). A state of the art review on the Internet of Things (IoT) History , Technology and fields of deployment.

[4] Wang, C., Daneshmand, M., Dohler, M., Mao, X., Hu, R. Q., & Wang, H. (2013). Guest Editorial- Special issue on the internet of things (IoT): Architecture, protocols, and services. IEEESensors Journal, 13(10), 3505–3508. http://doi.org/10.1109/JSEN.2013.2274906

[5] Utama, Y. A. K., & Hari, Y. (2017, September). Design of PID disturbance observer for temperature control on room heating system. In Electrical Engineering, Computer Science and Informatics (EECSI), 2017 4th International Conference on (pp. 1-6). IEEE.

[6] Hari, Y., & Dewi, L. P. (2018). Forecasting System Approach for Stock Trading with Relative Strength Index and Moving Average Indicator. Journal of Telecommunication, Electronic and Computer Engineering (JTEC), 10(2-3), 25-29.

[7] Hari, Y., Dewi, L. P., Santi, C., & Sembiring, M. J. (2018). Improving Financial Marketing And Accountability With Technology Penetration For The Bag SMEs Community In Gresik, Jawa Timur. Advances in Social Sciences Research Journal, 5(10).

Society for Science and Education United Kingdom
TMLAI Transactions on Machine Learning and Artificial Intelligence
Volume 7 No. 1
ISSN 2054-7390

Novel Artificial Human Optimization Field Algorithms – The Beginning

[1]Hassan M.H. Mustafa; [2]Satish Gajawada
[1]*Faculty of Specified Education, Dept. of Educational Technology,*
Banha University, Egypt
[2]*Alumnus, Indian Institute of Technology Roorkee*
Founder, Artificial Human Optimization Field
prof.dr.hassanmoustafa@gmail.com; gajawadasatish@gmail.com

ABSTRACT

New Artificial Human Optimization (AHO) Field Algorithms can be created from scratch or by adding the concept of Artificial Humans into other existing Optimization Algorithms. Particle Swarm Optimization (PSO) has been very popular for solving complex optimization problems due to its simplicity. In this work, new Artificial Human Optimization Field Algorithms are created by modifying existing PSO algorithms with AHO Field Concepts. These Hybrid PSO Algorithms comes under PSO Field as well as AHO Field. There are Hybrid PSO research articles based on Human Behavior, Human Cognition and Human Thinking etc. But there are no Hybrid PSO articles which based on concepts like Human Disease, Human Kindness and Human Relaxation. This paper proposes new AHO Field algorithms based on these research gaps. Some existing Hybrid PSO algorithms are given a new name in this work so that it will be easy for future AHO researchers to find these novel Artificial Human Optimization Field Algorithms. A total of 6 Artificial Human Optimization Field algorithms titled "Human Safety Particle Swarm Optimization (HuSaPSO)", “Human Kindness Particle Swarm Optimization (HKPSO)", “Human Relaxation Particle Swarm Optimization (HRPSO)", “Multiple Strategy Human Particle Swarm Optimization (MSHPSO)", “Human Thinking Particle Swarm Optimization (HTPSO)" and “Human Disease Particle Swarm Optimization (HDPSO)” are tested by applying these novel algorithms on Ackley, Beale, Bohachevsky, Booth and Three-Hump Camel Benchmark Functions. Results obtained are compared with PSO algorithm.

Keywords: Artificial Humans, Artificial Human Optimization Field, Particle Swarm Optimization, Genetic Algorithms, Hybrid Algorithms, Global Optimization Techniques, Nature Inspired Computing, Bio-Inspired Computing, Artificial Intelligence, Machine Learning

Highlights: 1) World’s First Hybrid PSO algorithm based on Human Kindness is proposed in this paper.

2) World’s First Hybrid PSO algorithm based on Human Relaxation is proposed in this paper

3) World’s First Hybrid PSO algorithm based on Human Disease is proposed in this paper

4) Made corrections to previous work under AHO Field in the Introduction Section of the paper

5) A Novel Section “Interesting Findings in Artificial Human Optimization Field” is present in this article

DOI: 10.14738/tmlai.71.5712
Publication Date: 09th February, 2019
URL: http://dx.doi.org/10.14738/tmlai.71.5712

6) Some existing Hybrid PSO algorithms are given a new name in this paper

1 Introduction

A field is a particular branch of study. Artificial Human Optimization Field (AHO Field) is a latest field. Proposing a new algorithm is different from proposing a new field. Generally researchers propose new algorithms. But for the first time in research industry history, a young researcher proposed a new field through Transactions on Machine Learning and Artificial Intelligence journal paper. Artificial Human Optimization (AHO) is a very recent field which took its birth on December 2016. This work was published in Transactions on Machine Learning and Artificial Intelligence with title “Entrepreneur: Artificial Human Optimization”. Hence this field is less than 2 years old. According to recent articles in AHO literature, there is scope for many PhD’s and PostDoc’s in Artificial Human Optimization Field (AHO Field). Also there exists an ocean of opportunities in Artificial Human Optimization Field. According to article “Entrepreneur: Artificial Human Optimization”, the first article in AHO Field was proposed in 2012 and there exists less than 20 papers in AHO Field. This mistake was corrected and the first article in AHO Field was proposed in 2009 according to article “Artificial Human Optimization – An Introduction”. Again there was a mistake. This mistake was corrected in article “Artificial Human Optimization – An Overview”. The correction was that the first article in AHO Field was proposed in 2006. Again there was a mistake. Finally, according to this paper “Novel Artificial Human Optimization Field Algorithms – The Beginning”, the first paper in AHO Field was proposed in 2003 with title “Society and Civilization: An Optimization Algorithm Based on the Simulation of Social Behavior”. Also there exist more than 30 papers in AHO Field. According to a recent article in AHO Literature, there is scope for millions of research articles in AHO Field [1-12]. Papers [1-12] gives details about Artificial Human Optimization Field, its algorithms and its overview. Papers [13-17] shows Hybrid PSO algorithms which come under Artificial Human Optimization Field.

The rest of the article is organized as follows:

Section 2 shows Particle Swarm Optimization (PSO) algorithm. Section 3 to Section 8 shows "Human Safety Particle Swarm Optimization (HuSaPSO)", “Human Kindness Particle Swarm Optimization (HKPSO)", “Human Relaxation Particle Swarm Optimization (HRPSO)", “Multiple Strategy Human Particle Swarm Optimization (MSHPSO)", “Human Thinking Particle Swarm Optimization (HTPSO)" and “Human Disease Particle Swarm Optimization (HDPSO)” respectively. Interesting Findings in AHO Field are shown in Section 9. Section 10 gives results obtained. Finally, Conclusions are given in Section 11.

2 Particle Swarm Optimization

Particle Swarm Optimization (PSO) was proposed by Kennedy and Eberhart in 1995. PSO is based on Artificial Birds. It has been applied to solve complex optimization problems. Papers [18-24] shows you details related to PSO, its algorithms and its overview.

In PSO, first we initialize all particles as shown below. Two variables $pbest_i$ and gbest are maintained. $pbest_i$ is the best fitness value achieved by i^{th} particle so far and gbest is the best fitness value achieved by all particles so far. Lines 4 to 11 in the below text helps in maintaining particle best and global best. Then the velocity is updated by rule shown in line no. 14. Line 15 updates position of i^{th} particle. Line 19 increments the number of iterations and then the control goes back to line 4. This process of a particle moving towards its local best and also moving towards global best of particles is continued until termination criteria will be reached.

Procedure: Particle Swarm Optimization (PSO)

1) Initialize all particles
2) iterations = 0
3) **do**
4) **for** each particle i **do**
5) **If** (f(x_i) < f($pbest_i$)) **then**
6) $pbest_i = x_i$
7) **end if**
8) **if** (f($pbest_i$) < f(gbest)) **then**
9) gbest = $pbest_i$
10) **end if**
11) **end for**
12) **for** each particle i **do**
13) **for** each dimension d **do**
14) $v_{i,d} = w*v_{i,d} +$
C_1*Random(0,1)*($pbest_{i,d} - x_{i,d}$)
\+ C_2*Random(0,1)*($gbest_d - x_{i,d}$)
15) $x_{i,d} = x_{i,d} + v_{i,d}$
17) **end for**
18) **end for**
19) iterations = iterations + 1
20) **while** (termination condition is false)

3 Human Safety Particle Swarm Optimization

In PSO particles move towards local best and global best. Almost all PSO algorithms are based on best location of particles. But there is another strategy which is moving towards the optimal by using worst location of particles. Some algorithms in PSO Field are based on this idea in which worst location of particles also helps in finding optimal solution.

The idea of using worst location of particles in the velocity updating equation was first introduced in [28].

A Novel PSO Algorithm was proposed in [25]. In this algorithm, a coefficient is calculated based on distance of particle to closest best and closest worst particles. This coefficient is used in updating velocity of particle. In [26], velocity is updated using both particles best and particles worst location. This work is extended in [27] where velocity of particle is updated using particles local worst, global worst, particles local best and global best of all particles. The velocity is updated in [11] where particles move towards the local best and global best in even iterations and move away from local worst and global worst in odd iterations.

According to our experience it can be observed that Humans not only learn from his/her own local best and other individuals global best but also learns from his/her own local worst and other individuals global worst. Hence in [28], a new PSO (NPSO) is proposed where optimal solution is found by moving away from local worst location of particle and global worst location of all particles. The algorithm (NPSO) proposed in [28] is based on Artificial Human Optimization Field Concepts because there are Humans who try to be on safe side by moving away from local worst and global worst. Hence NPSO in [28] is given a new name titled “Human Safety Particle Swarm Optimization (HuSaPSO)” in this current paper.

In line no. 14 in below procedure it can be seen that velocity update equation is based on moving away from local worst of particle and global worst of all particles. In NPSO work in [28], researchers haven't used inertia weight while updating velocity but in the below procedure, inertia weight is used. Human Safety Particle Swarm Optimization (HuSaPSO) is shown below:

Procedure: Human Safety Particle Swarm Optimization (HuSaPSO)

1) Initialize all particles
2) iterations = 0
3) **do**
4)   **for** each particle i **do**
5)   **If** (f(x_i) < f($pbest_i$)) **then**
6) $pbest_i = x_i$
7)   **end if**
8)   **if** (f($pbest_i$) < f(gbest)) **then**
9)   gbest = $pbest_i$
10)   **end if**
11)   **end for**
12)   **for** each particle i **do**
13)   **for** each dimension d **do**
14) $v_{i,d} = w*v_{i,d} +$
                C_1*Random(0,1)*($x_{i,d} - pworst_{i,d}$)
              + C_2*Random(0,1)*($x_{i,d} - gworst_d$)
15) $x_{i,d} = x_{i,d} + v_{i,d}$
17)   **end for**
18)   **end for**
19)   iterations = iterations + 1
20) **while** (termination condition is false)

4 Human Kindness Particle Swarm Optimization

There are no Hybrid PSO algorithms based on Human Kindness till date. Human Kindness is modeled by introducing $KindnessFactor_i$ for particle i. This factor is added in the position update equation in line number 15 of the below procedure. The more the KindnessFactor the faster is the movement of particle. In this work, a random number between 0 and 1 is generated and assigned to KindnessFactor of particle. The Proposed Human Kindness Particle Swarm Optimization (HKPSO) is shown below:

Procedure: Human Kindness Particle Swarm Optimization (HKPSO)

1) Initialize all particles
2) iterations = 0
3) **do**
4)   **for** each particle i **do**
5)   **If** (f(x_i) < f($pbest_i$)) **then**
6) $pbest_i = x_i$
7)   **end if**
8)   **if** (f($pbest_i$) < f(gbest)) **then**
9)   gbest = $pbest_i$
10)   **end if**
11)   **end for**

12) **for** each particle i **do**
13) **for** each dimension d **do**
14) $v_{i,d} = w*v_{i,d} + C_1*Random(0,1)*(pbest_{i,d} - x_{i,d}) + C_2*Random(0,1)*(gbest_d - x_{i,d})$
15) $x_{i,d} = x_{i,d} + KindnessFactor_i * v_{i,d}$
17) **end for**
18) **end for**
19) iterations = iterations + 1
20) **while** (termination condition is false)

5 Human Relaxation Particle Swarm Optimization

There are no Hybrid PSO algorithms based on Human Relaxation till date. All particles move in some direction in all iterations. There is nothing like relaxation for a particle. RelaxationProbability is introduced in this paper in an attempt to model Human Relaxation. A random number is generated in the line number 13 in the below procedure. If the random number generated is less than or equal to RelaxationProbability then the particle is said to be on relaxation state and this particle will skip velocity updating and position updating in this particular iteration. On the other hand, if the random number generated is greater than RelaxationProbability, then particle will undergo velocity and position updating just like in normal PSO. Proposed Human Relaxation Particle Swarm Optimization (HRPSO) is shown below:

Procedure: Human Relaxation Particle Swarm Optimization (HRPSO)

1) Initialize all particles
2) Initialize RelaxationProbability
2) iterations = 0
3) **do**
4) **for** each particle i **do**
5) **If** (f(x_i) < f($pbest_i$)) **then**
6) $pbest_i = x_i$
7) **end if**
8) **if** (f($pbest_i$) < f(gbest)) **then**
9) gbest = $pbest_i$
10) **end if**
11) **end for**
12) **for** each particle i **do**
13) **if** Random(0,1) < = RelaxationProbability
14) **continue** // continues to next particle
15) **end if**
16) **for** each dimension d **do**
17) $v_{i,d} = w*v_{i,d} + C_1*Random(0,1)*(pbest_{i,d} - x_{i,d}) + C_2*Random(0,1)*(gbest_d - x_{i,d})$
18) $x_{i,d} = x_{i,d} + v_{i,d}$
19) **end for**
20) **end for**
21) iterations = iterations + 1
22) **while** (termination condition is false)

6 Multiple Strategy Human Particle Swarm Optimization

Hassan Satish Particle Swarm Optimization (HSPSO) proposed in [11] is given a new name titled "Multiple Strategy Human Particle Swarm Optimization (MSHPSO)" in this paper. MSHPSO is obtained by incorporation of Multiple Strategy Human Optimization (MSHO) concepts into Particle Swarm Optimization. In starting and even generations the Artificial Humans move towards the best fitness value. In odd generations Artificial Humans move away from the worst fitness value. In MSHPSO, local worst of particle and global worst of all particles are maintained in addition to local best of particle and global best of all particles. This is shown in lines 4 to 17. In lines 19 to 24 velocity is calculated by moving towards the local best of particle and global best of all particles. In lines 26 to 31 pseudo code for odd generations is shown in below text. In these odd generations particles move away from local worst of particle and also away from global worst of all particles. In line 33, number of iterations is incremented by one. Then control goes back to line number 4. This process of moving towards the best in one generation and moving away from the worst in next generation is continued until termination criteria has been reached. MSHPSO proposed in [11] is shown below:

Procedure: Multiple Strategy Human Particle Swarm Optimization (MSHPSO)

1) Initialize all particles
2) iterations = 0
3) **do**
4) **for** each particle i **do**
5) **If** (f(x_i) < f($pbest_i$)) **then**
6) $pbest_i = x_i$
7) **end if**
8) **if** (f($pbest_i$) < f(gbest)) **then**
9) gbest = $pbest_i$
10) **end if**
11) **If** (f(x_i) > f($pworst_i$)) **then**
12) $pworst_i = x_i$
13) **end if**
14) **if** (f($pworst_i$) > f(gworst)) **then**
15) gworst = $pworst_i$
16) **end if**
17) **end for**
18) **If** ((iterations == 0) || (iterations%2==0)) **then**
// for starting and even iterations
19) **for** each particle i **do**
20) **for** each dimension d **do**
21) $v_{i,d} = w*v_{i,d} +$
$C_1*Random(0,1)*(pbest_{i,d} – x_{i,d})$
$+C_2*Random(0,1)*(gbest_d – x_{i,d})$
22) $x_{i,d} = x_{i,d} + v_{i,d}$
23) **end for**
24) **end for**
25) **else** // for odd iterations
26) **for** each particle i **do**
27) **for** each dimension d **do**
28) $v_{i,d} = w*v_{i,d} +$

C_1*Random(0,1)*($x_{i,d}$ - $pworst_{i,d}$)
+ C_2*Random(0,1)*($x_{i,d}$ - $gworst_d$)
29) $x_{i,d} = x_{i,d} + v_{i,d}$
30) **end for**
31) **end for**
32) **end if**
33) iterations = iterations + 1
34) **while** (termination condition is false)

7 Human Thinking Particle Swarm Optimization

In [12], the particles move towards best locations and away from worst locations in the same iteration/generation. The Concept used in [12] and [27] is same. The only difference is that a new name titled "Human Thinking Particle Swarm Optimization (HTPSO)" is given in [12] for the concept in [27].

Almost all Particle Swarm Optimization (PSO) algorithms are proposed such that the particles move towards best particles. But Human Thinking is such that they not only move towards best but also moves away from the worst. This concept was used to design algorithm titled "Multiple Strategy Human Optimization (MSHO)" in [4]. In MSHO, artificial Humans move towards the best in even generations and move away from the worst in odd generations. But in Human Thinking Particle Swarm Optimization, both strategies happen in the same generation and all generations follow the same strategy. That is moving towards the best and moving away from the worst strategies happen simultaneously in the same generation unlike MSHO designed in [4]. The HTPSO algorithm proposed in [12] and [27] is shown below:

Procedure: Human Thinking Particle Swarm Optimization (HTPSO)

1) Initialize all particles
2) iterations = 0
3) **do**
4) **for** each particle i **do**
5) **If** (f(x_i) < f($pbest_i$)) **then**
6) $pbest_i = x_i$
7) **end if**
8) **if** (f($pbest_i$) < f(gbest)) **then**
9) gbest = $pbest_i$
10) **end if**
11) **If** (f(x_i) > f($pworst_i$)) **then**
12) $pworst_i = x_i$
13) **end if**
14) **if** (f($pworst_i$) > f(gworst)) **then**
15) gworst = $pworst_i$
16) **end if**
17) **end for**
18) **for** each particle i **do**
19) **for** each dimension d **do**
20) $v_{i,d} = w*v_{i,d}$ + Random(0,1)*($pbest_{i,d} - x_{i,d}$) + Random(0,1)*($gbest_d - x_{i,d}$)

```
21)                        vi,d = vi,d + Random(0,1)*( xi,d - pworsti,d ) + Random(0,1)*( xi,d - gworstd)
22)                        xi,d = xi,d + vi,d
23)              end for
24)      end for
25)      iterations = iterations + 1
26) while (termination condition is false)
```

8 Human Disease Particle Swarm Optimization

In this section, an innovative Hybrid PSO algorithm titled “Human Disease Particle Swarm Optimization (HDPSO)” is proposed which is based on Bipolar Disorder Human Disease. People with Bipolar Disorder Human Disease experience changes in moods between depression and mania. This disease is also known as manic depression. The mood swings between highs of mania (very happy) and lows of depression (very sad) are significant and usually extreme. The mood of either mania or depression can exist for few days, few weeks or even few months.

In Human Disease Particle Swarm Optimization, the strategy for updating velocity is different in odd and even generations. Person affected with Bipolar Disorder Human Disease goes through Very happy (UP) and very sad (Down) phases. Very happy and Very sad phases of Bipolar Disorder Human Disease are imitated in proposed HDPSO algorithm by incorporating different updating strategies in Particle Swarm Optimization algorithm. If person is very happy then he moves towards global best and local best of particles. If the person is very sad then he moves away from global best and local best of particles. In line 15 in below procedure, the person moves towards global best and local best of particles. In line 22, the person moves away from global best and local best of particles. The proposed HDPSO algorithm is shown below:

Procedure: Human Disease Particle Swarm Optimization (HDPSO)

```
1) Initialize all particles
2) iterations = 0
3) do
4)       for each particle i do
5)             If ( f( xi ) < f( pbesti ) ) then
6)                   pbesti = xi
7)             end if
8)             if ( f( pbesti ) < f( gbest ) ) then
9)                   gbest = pbesti
10)            end if
11)      end for

12)      If ((iterations == 0) || (iterations%2==0)) then
          // for starting and even iterations
13)            for each particle i do
14)                  for each dimension d do
15)                     vi,d = w*vi,d +
                      C1*Random(0,1)*(pbesti,d – xi,d)
                     +C2*Random(0,1)*(gbestd – xi,d)
16)                     xi,d = xi,d + vi,d
17)                  end for
```

```
            end for
      else // for odd iterations
            for each particle i do
                  for each dimension d do
                     v_{i,d} = w*v_{i,d} +
                        C_1*Random(0,1)*( x_{i,d} -
                        pbest_{i,d} )
                       + C_2*Random(0,1)*( x_{i,d} -
                        gbest_d )
                     x_{i,d} = x_{i,d} + v_{i,d}
                  end for
            end for
      end if
      iterations = iterations + 1
while ( termination condition is false)
```

9 Interesting Findings in Artificial Human Optimization Field

Human Thinking Particle Swarm Optimization (HTPSO) was proposed in [12] by Satish Gajawada et al. in 2018. Velocity is updated in HTPSO such that particle moves towards its local best, global best of particles, its local worst and global worst of particles. But this idea of velocity update is already proposed in [27]. Hence from here on it should be noted that HTPSO algorithm in [12] was originally proposed in [27]. This mistake happened because researchers added concept of Artificial Humans into PSO algorithms but they have not included the word “Human” in naming the new algorithm or in the entire paper. Another reason is that there are common things between Artificial Birds and Artificial Humans. The idea in [12] and [27] belongs to this intersection. PSO researchers added this common behavior to PSO and considered it as an algorithm based on Artificial Birds. Artificial Human Optimization Field (AHO Field) is new and hence there are not many algorithms under this new field. It will be very difficult for an AHO researcher to find that his concept/algorithm already exists in the form of PSO variant.

The algorithm in [28] comes under Artificial Human Optimization Field. There are millions of articles on internet. It will be very difficult for Artificial Human Optimization (AHO) researcher to find the fact that there exists paper [28] which added human safety into PSO and hence this work in [28] comes under Artificial Human Optimization Field. Also the title of paper [28] is “New Particle Swarm Optimization Technique”. Hence AHO researchers might think this is another algorithm inspired by birds. Hence from here on researchers should include word “Human” or some other word to know that it is a AHO concept algorithm.

10 Results

The results obtained after applying HuSaPSO, HKPSO, HRPSO, MSHPSO, HTPSO, HDPSO and PSO algorithms on various benchmark functions are shown in this section. The figures of benchmark functions are taken from [29].

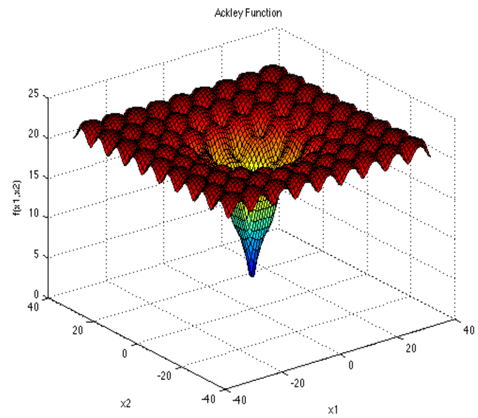
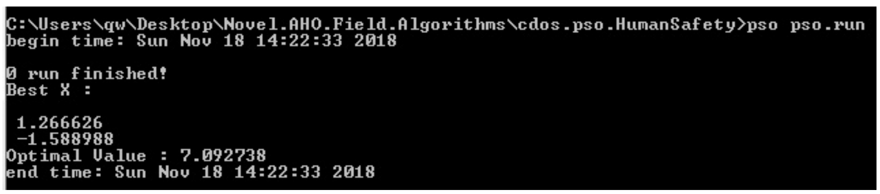


Figure 1. Ackley Function

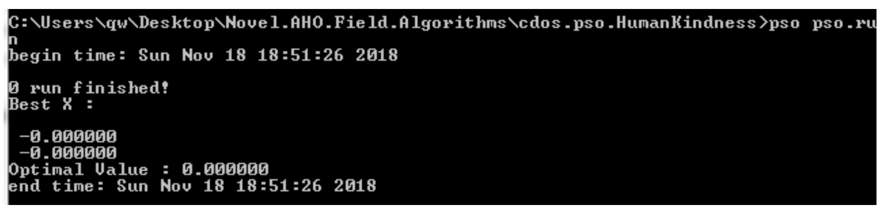


```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanSafety>pso pso.run
begin time: Sun Nov 18 14:22:33 2018

0 run finished!
Best X :

 1.266626
 -1.588988
Optimal Value : 7.092738
end time: Sun Nov 18 14:22:33 2018
```

Figure 2. Result given by Human Safety Particle Swarm Optimization (HuSaPSO) on Ackley Function



```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanKindness>pso pso.ru
n
begin time: Sun Nov 18 18:51:26 2018

0 run finished!
Best X :

 -0.000000
 -0.000000
Optimal Value : 0.000000
end time: Sun Nov 18 18:51:26 2018
```

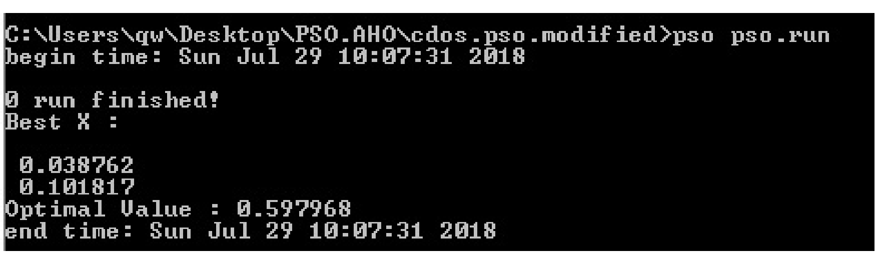
Figure 3. Result given by Human Kindness Particle Swarm Optimization (HKPSO) on Ackley Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanRelaxation>pso pso.
run
begin time: Mon Nov 19 13:02:17 2018

0 run finished!
Best X :

 0.000000
 0.000000
Optimal Value : 0.000000
end time: Mon Nov 19 13:02:17 2018
```

Figure 4. Result given by Human Relaxation Particle Swarm Optimization (HRPSO) on Ackley Function



```
C:\Users\qw\Desktop\PSO.AHO\cdos.pso.modified>pso pso.run
begin time: Sun Jul 29 10:07:31 2018

0 run finished!
Best X :

 0.038762
 0.101817
Optimal Value : 0.597968
end time: Sun Jul 29 10:07:31 2018
```

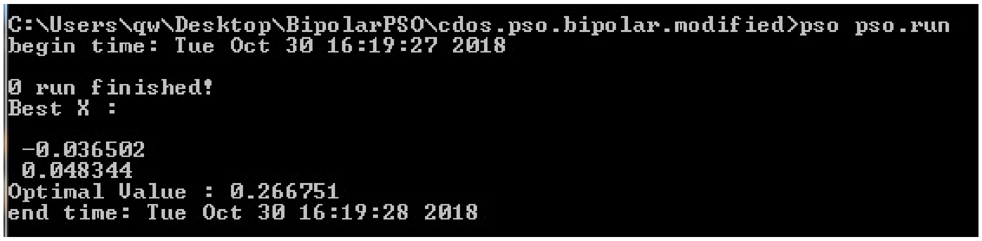
Figure 5. Result given by Multiple Strategy Human Particle Swarm Optimization (MSHPSO) on Ackley Function

```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\HTPSO.cdos.pso.modified>pso pso.run
begin time: Wed Jul 25 16:19:22 2018

0 run finished!
Best X :

 0.429100
 -0.591114
Optimal Value : 4.262748
end time: Wed Jul 25 16:19:22 2018
```

Figure 6. Result given by Human Thinking Particle Swarm Optimization (HTPSO) on Ackley Function

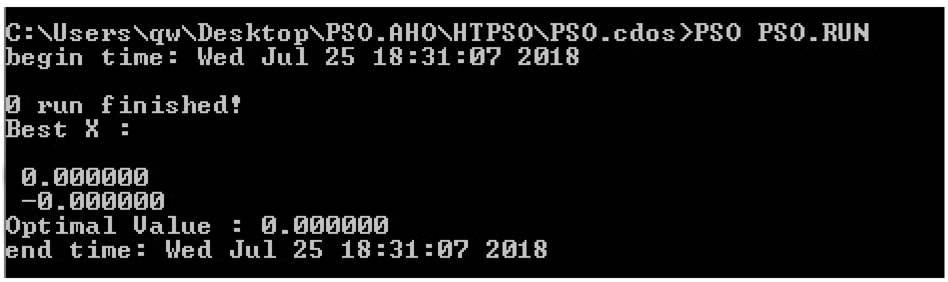


```
C:\Users\qw\Desktop\BipolarPSO\cdos.pso.bipolar.modified>pso pso.run
begin time: Tue Oct 30 16:19:27 2018

0 run finished!
Best X :

 -0.036502
 0.048344
Optimal Value : 0.266751
end time: Tue Oct 30 16:19:28 2018
```

Figure 7. Result given by Human Disease Particle Swarm Optimization (HDPSO) on Ackley Function



```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\PSO.cdos>PSO PSO.RUN
begin time: Wed Jul 25 18:31:07 2018

0 run finished!
Best X :

 0.000000
 -0.000000
Optimal Value : 0.000000
end time: Wed Jul 25 18:31:07 2018
```

Figure 8. Result given by Particle Swarm Optimization (PSO) on Ackley Function

From Figure 2 to Figure 8 it can be observed that HKPSO, HRPSO, PSO gave optimum solution and performed well on Ackley Function. But HuSaPSO, HTPSO, HDPSO, MSHPSO algorithms didn't perform well on Ackley Function.

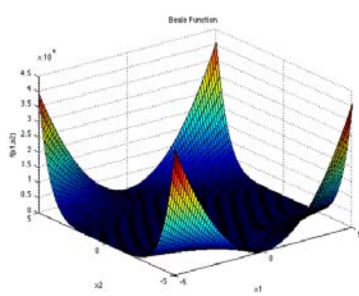


Figure 9. Beale Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanSafety>pso pso.run
begin time: Sun Nov 18 14:31:14 2018

0 run finished!
Best X :

 2.891090
 0.319284
Optimal Value : 0.368583
end time: Sun Nov 18 14:31:15 2018
```

Figure 10. Result given by Human Safety Particle Swarm Optimization (HuSaPSO) on Beale Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanKindness>pso pso.ru
n
begin time: Sun Nov 18 18:56:12 2018

0 run finished!
Best X :

 3.000000
 0.500000
Optimal Value : 0.000000
end time: Sun Nov 18 18:56:12 2018
```

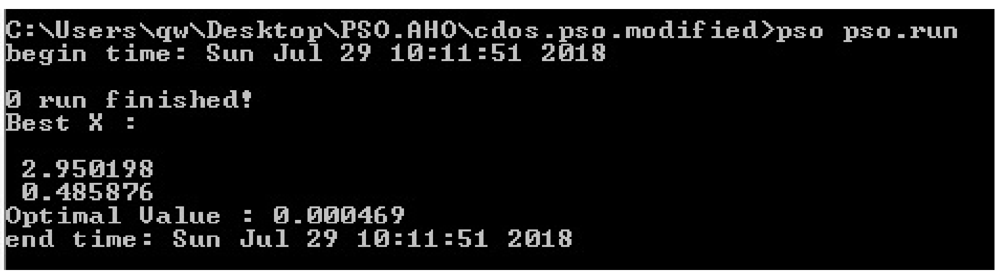
Figure 11. Result given by Huma Kindness Particle Swarm Optimization (HKPSO) on Beale Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanRelaxation>pso pso.
run
begin time: Mon Nov 19 13:05:07 2018

0 run finished!
Best X :

 3.000000
 0.500000
Optimal Value : 0.000000
end time: Mon Nov 19 13:05:07 2018
```

Figure 12. Result given by Human Relaxation Particle Swarm Optimization (HRPSO) on Beale Function



```
C:\Users\qw\Desktop\PSO.AHO\cdos.pso.modified>pso pso.run
begin time: Sun Jul 29 10:11:51 2018

0 run finished!
Best X :

 2.950198
 0.485876
Optimal Value : 0.000469
end time: Sun Jul 29 10:11:51 2018
```

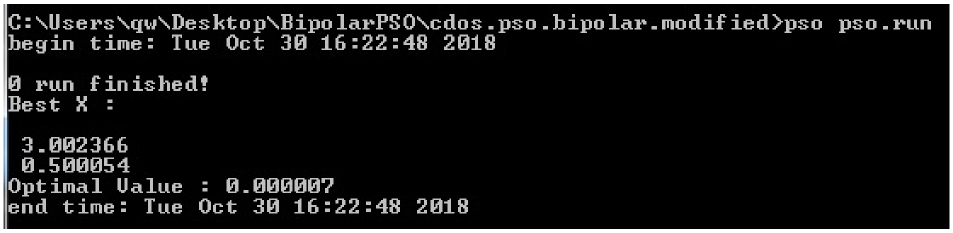
Figure 13. Result given by Multiple Strategy Human Particle Swarm Optimization (MSHPSO) on Beale Function

```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\HTPSO.cdos.pso.modified>PSO PSO.RUN
begin time: Wed Jul 25 17:52:43 2018

0 run finished!
Best X :

 2.729012
 0.332734
Optimal Value : 0.134325
end time: Wed Jul 25 17:52:43 2018
```

Figure 14. Result given by Human Thinking Particle Swarm Optimization (HTPSO) on Beale Function

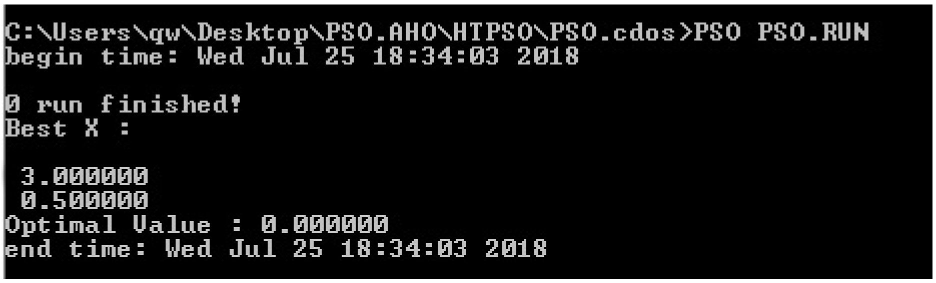


```
C:\Users\qw\Desktop\BipolarPSO\cdos.pso.bipolar.modified>pso pso.run
begin time: Tue Oct 30 16:22:48 2018

0 run finished!
Best X :

 3.002366
 0.500054
Optimal Value : 0.000007
end time: Tue Oct 30 16:22:48 2018
```

Figure 15. Result given by Human Disease Particle Swarm Optimization (HDPSO) on Beale Function



```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\PSO.cdos>PSO PSO.RUN
begin time: Wed Jul 25 18:34:03 2018

0 run finished!
Best X :

 3.000000
 0.500000
Optimal Value : 0.000000
end time: Wed Jul 25 18:34:03 2018
```

Figure 16. Result given by Particle Swarm Optimization (PSO) on Beale Function

From Figure 10 to Figure 16 it can be observed that HKPSO, HRPSO, MSHPSO, HDPSO and PSO gave optimal result and performed well on Beale Function. But HuSaPSO, HTPSO didn't perform well on Beale Function.

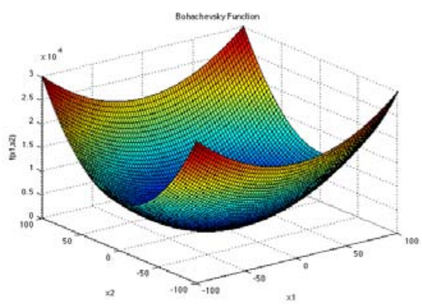


Figure 17. Bohachevsky Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanSafety>pso pso.run
begin time: Sun Nov 18 14:36:50 2018

0 run finished!
Best X :

 4.222084
 -5.296627
Optimal Value : 75.115540
end time: Sun Nov 18 14:36:50 2018
```

Figure 18. Result given by Human Safety Particle Swarm Optimization (HuSaPSO) on Bohachevsky Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanKindness>pso pso.ru
n
begin time: Sun Nov 18 18:58:57 2018

0 run finished!
Best X :

 0.000011
 -0.000004
Optimal Value : -0.000000
end time: Sun Nov 18 18:58:57 2018
```

Figure 19. Result given by Human Kindness Particle Swarm Optimization (HKPSO) on Bohachevsky Function

Figure 20. Result given by Human Relaxation Particle Swarm Optimization (HRPSO) on Bohachevsky Function

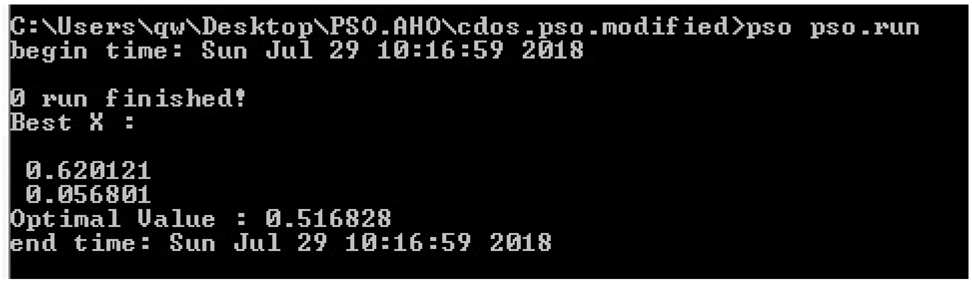


Figure 21. Result given by Multiple Strategy Human Particle Swarm Optimization (MSHPSO) on Bohachevsky Function

Figure 22. Result given by Human Thinking Particle Swarm Optimization (HTPSO) on Bohachevsky Function

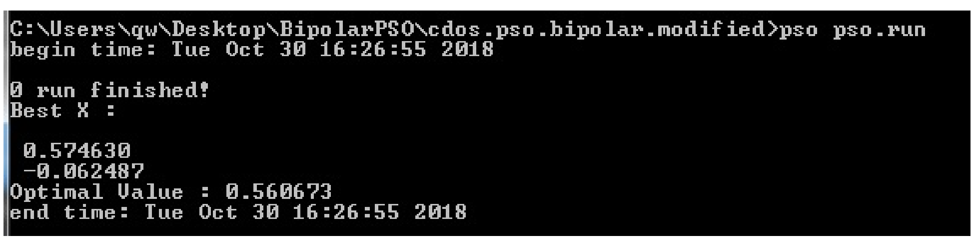


Figure 23. Result given by Human Disease Particle Swarm Optimization (HDPSO) on Bohachevsky Function

Figure 24. Result given by Particle Swarm Optimization (PSO) on Bohachevsky Function

From Figure 18 to Figure 24 it can be observed that HKPSO, HRPSO and PSO gave optimal result and performed well on Bohachevsky Function. But HuSaPSO, MSHPSO, HTPSO and HDPSO didn't perform well on Bohachevsky Function.

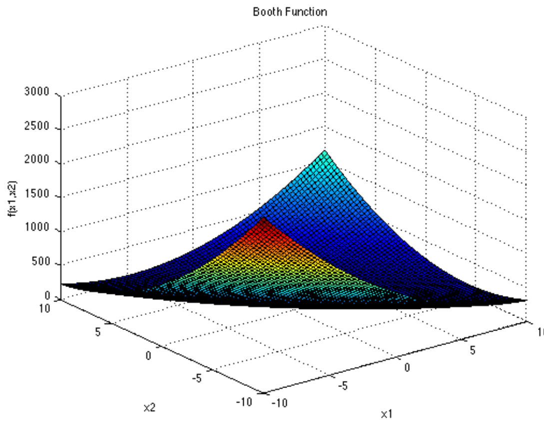


Figure 25. Booth Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanSafety>pso pso.run
begin time: Sun Nov 18 14:41:00 2018

0 run finished!
Best X :

 0.845791
 3.062923
Optimal Value : 0.061072
end time: Sun Nov 18 14:41:00 2018
```

Figure 26. Result given by Human Safety Particle Swarm Optimization (HuSaPSO) on Booth Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanKindness>pso pso.ru
n
begin time: Sun Nov 18 19:01:17 2018

0 run finished!
Best X :

 1.000000
 3.000000
Optimal Value : 0.000000
end time: Sun Nov 18 19:01:17 2018
```

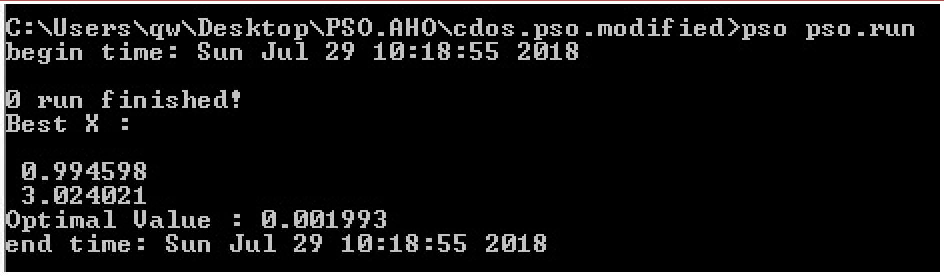
Figure 27. Result given by Human Kindness Particle Swarm Optimization (HKPSO) on Booth Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanRelaxation>pso pso.
run
begin time: Mon Nov 19 13:11:41 2018

0 run finished!
Best X :

 1.000000
 3.000000
Optimal Value : 0.000000
end time: Mon Nov 19 13:11:41 2018
```

Figure 28. Result given by Human Relaxation Particle Swarm Optimization (HRPSO) on Booth Function



```
C:\Users\qw\Desktop\PSO.AHO\cdos.pso.modified>pso pso.run
begin time: Sun Jul 29 10:18:55 2018

0 run finished!
Best X :

 0.994598
 3.024021
Optimal Value : 0.001993
end time: Sun Jul 29 10:18:55 2018
```

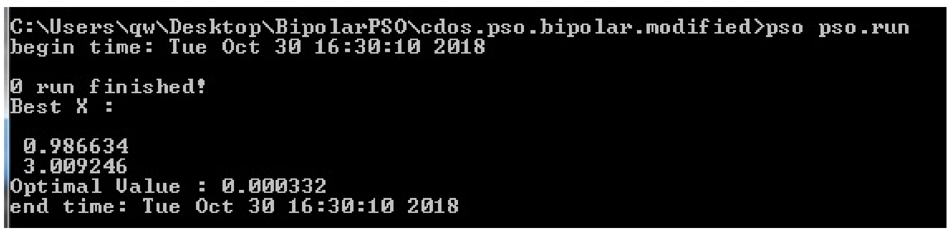
Figure 29. Result given by Multiple Strategy Human Particle Swarm Optimization (MSHPSO) on Booth Function

```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\HTPSO.cdos.pso.modified>PSO PSO.RUN
begin time: Wed Jul 25 17:22:55 2018

0 run finished!
Best X :

 1.274603
 2.578953
Optimal Value : 0.338471
end time: Wed Jul 25 17:22:56 2018
```

Figure 30. Result given by Human Thinking Particle Swarm Optimization (HTPSO) on Booth Function

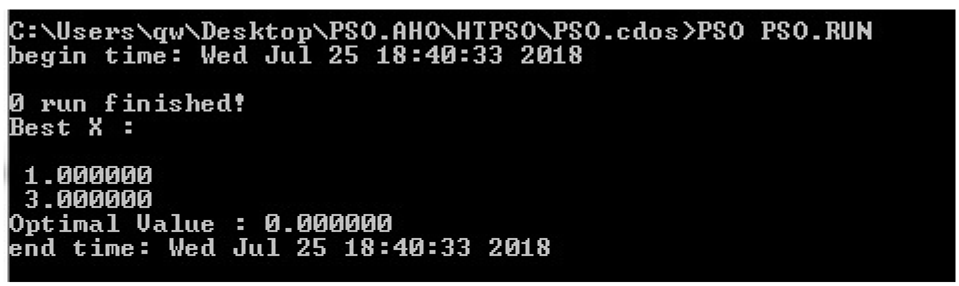


```
C:\Users\qw\Desktop\BipolarPSO\cdos.pso.bipolar.modified>pso pso.run
begin time: Tue Oct 30 16:30:10 2018

0 run finished!
Best X :

 0.986634
 3.009246
Optimal Value : 0.000332
end time: Tue Oct 30 16:30:10 2018
```

Figure 31. Result given by Human Disease Particle Swarm Optimization (HDPSO) on Booth Function



```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\PSO.cdos>PSO PSO.RUN
begin time: Wed Jul 25 18:40:33 2018

0 run finished!
Best X :

 1.000000
 3.000000
Optimal Value : 0.000000
end time: Wed Jul 25 18:40:33 2018
```

Figure 32. Result given by Particle Swarm Optimization (PSO) on Booth Function

From Figure 26 to Figure 32 it can be observed that HuSaPSO gave result close to optimal solution and performed O.K. HKPSO, HRPSO, HDPSO, MSHPSO and PSO gave optimal result and performed well on Booth Function. But HTPSO didn't perform well on Booth Function.

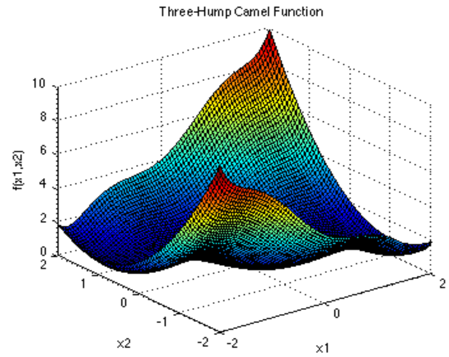


Figure 33. Three-Hump Camel Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanSafety>pso pso.run
begin time: Sun Nov 18 14:45:18 2018

0 run finished!
Best X :

 0.211103
 -0.264831
Optimal Value : 0.101288
end time: Sun Nov 18 14:45:18 2018
```

Figure 34. Result given by Human Safety Particle Swarm Optimization (HuSaPSO) on Three-Hump Camel Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanKindness>pso pso.ru
n
begin time: Sun Nov 18 19:04:07 2018

0 run finished!
Best X :

 0.000000
 0.000000
Optimal Value : 0.000000
end time: Sun Nov 18 19:04:07 2018
```

Figure 35. Result given by Human Kindness Particle Swarm Optimization (HKPSO) on Three-Hump Camel Function

```
C:\Users\qw\Desktop\Novel.AHO.Field.Algorithms\cdos.pso.HumanRelaxation>pso pso.
run
begin time: Mon Nov 19 13:14:30 2018

0 run finished!
Best X :

 0.000000
 0.000000
Optimal Value : 0.000000
end time: Mon Nov 19 13:14:30 2018
```

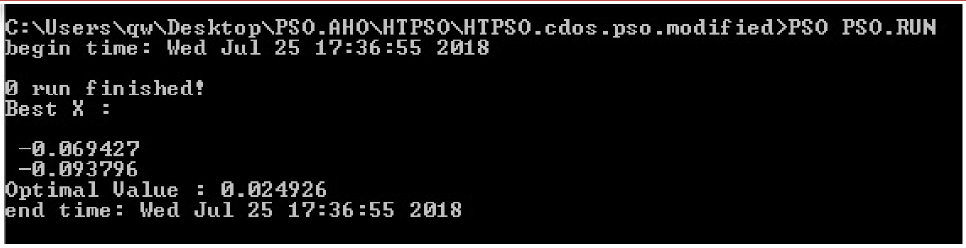
Figure 36. Result given by Human Relaxation Particle Swarm Optimization (HRPSO) on Three-Hump Camel Function

```
C:\Users\qw\Desktop\PSO.AHO\cdos.pso.modified>pso pso.run
begin time: Sun Jul 29 10:20:44 2018

0 run finished!
Best X :

 -0.008841
 -0.034073
Optimal Value : 0.001618
end time: Sun Jul 29 10:20:44 2018
```

Figure 37. Result given by Multiple Strategy Human Particle Swarm Optimization (MSHPSO) on Three-Hump Camel Function

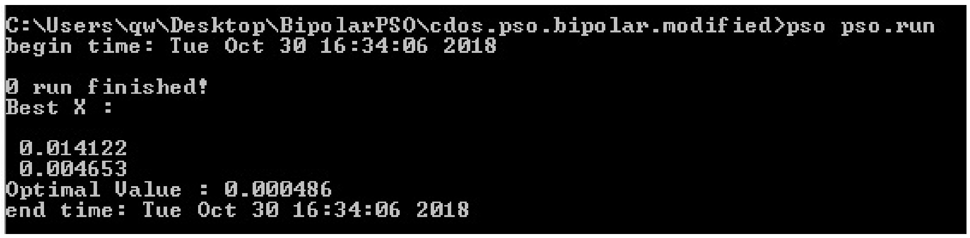


```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\HTPSO.cdos.pso.modified>PSO PSO.RUN
begin time: Wed Jul 25 17:36:55 2018

0 run finished!
Best X :

 -0.069427
 -0.093796
Optimal Value : 0.024926
end time: Wed Jul 25 17:36:55 2018
```

Figure 38. Result given by Human Thinking Particle Swarm Optimization (HTPSO) on Three-Hump Camel Function



```
C:\Users\qw\Desktop\BipolarPSO\cdos.pso.bipolar.modified>pso pso.run
begin time: Tue Oct 30 16:34:06 2018

0 run finished!
Best X :

 0.014122
 0.004653
Optimal Value : 0.000486
end time: Tue Oct 30 16:34:06 2018
```

Figure 39. Result given by Human Disease Particle Swarm Optimization (HDPSO) on Three-Hump Camel Function

```
C:\Users\qw\Desktop\PSO.AHO\HTPSO\PSO.cdos>PSO PSO.RUN
begin time: Wed Jul 25 18:44:40 2018

0 run finished!
Best X :

 0.000000
 0.000000
Optimal Value : 0.000000
end time: Wed Jul 25 18:44:40 2018
```

Figure 40. Result given by Particle Swarm Optimization (PSO) on Three-Hump Camel Function

From Figure 34 to Figure 40 it can be observed that HKPSO, HRPSO, MSHPSO, HTPSO, HDPSO and PSO gave optimal result and performed well on Three-Hump Camel Function. But HuSaPSO didn't perform well on Three-Hump Camel Function.

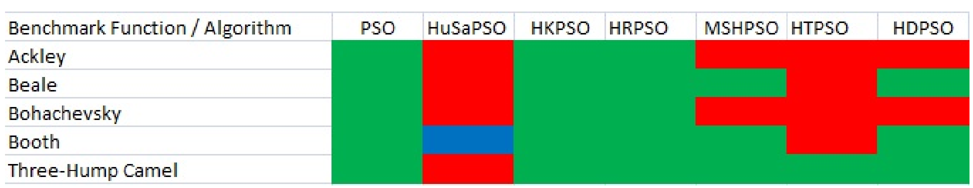


Figure 41. Overall Result

In Figure 41 first row shows PSO algorithms and first column shows benchmark functions. Green represents "Performed Well". Red represents "Didn't Performed Well". Blue represents "Performed O.K." or "Performed Between Well and Not Well".

From above figure it is clear that HKPSO, HRPSO and PSO performed Well for all benchmark functions whereas HuSaPSO didn't perform well even on single benchmark function. MSHPSO and HSPSO performed well on three benchmark functions. HTPSO performed well on only single benchmark function.

11 Conclusions

Hybrid PSO algorithms inspired by Human Kindness (HKPSO), Bipolar Disorder Human Disease (HDPSO) and Human Relaxation (HRPSO) are proposed in this novel work. Two previous Hybrid PSO algorithms are given a new name titled "Human Safety Particle Swarm Optimization (HuSaPSO)" and "Multiple Strategy Human Particle Swarm Optimization (MSHPSO)" in this research paper. A total of 7 algorithms are applied on set of 5 benchmark functions and results obtained are shown in this work. It can be concluded that just because some optimization algorithm is inspired by Humans doesn't mean it will perform better than other optimization algorithms like optimization algorithms inspired by other beings like Birds (PSO). It can be observed from this work that some AHO algorithms performed as good as PSO where as some other AHO algorithms didn't perform as good as PSO. This is just the beginning of research in Artificial Human Optimization Field (AHO Field).

REFERENCES

[1] Satish Gajawada. "POSTDOC : The Human Optimization", Computer Science & Information Technology (CS & IT), CSCP, pp. 183-187, 2013.

[2] Satish Gajawada. "CEO: Different Reviews on PhD in Artificial Intelligence", Global Journal of Advanced Research, vol. 1, no.2, pp. 155-158, 2014.

[3] Satish Gajawada. "Entrepreneur: Artificial Human Optimization". Transactions on Machine Learning and Artificial Intelligence, Volume 4 No 6 December (2016); pp: 64-70

[4] Satish Gajawada. "Artificial Human Optimization – An Introduction", Transactions on Machine Learning and Artificial Intelligence, Volume 6, No 2, pp: 1-9, April 2018

[5] Satish Gajawada. "An Ocean of Opportunities in Artificial Human Optimization Field", Transactions on Machine Learning and Artificial Intelligence, Volume 6, No 3, June 2018

[6] Satish Gajawada. "25 Reviews on Artificial Human Optimization Field for the First Time in Research Industry", International Journal of Research Publications, Vol. 5, no. 2, United Kingdom.

[7] Satish Gajawada and Hassan M. H. Mustafa, "Collection of Abstracts in Artificial Human Optimization Field", International Journal of Research Publications, Volume 7, No 1, United Kingdom, 2018.

[8] Satish Gajawada, Hassan M. H. Mustafa, HIDE : Human Inspired Differential Evolution - An Algorithm under Artificial Human Optimization Field , International Journal of Research Publications (Volume: 7, Issue: 1), http://ijrp.org/paper-detail/264

[9] Satish Gajawada, Hassan M. H. Mustafa , Artificial Human Optimization – An Overview. Transactions on Machine Learning and Artificial Intelligence, Volume 6, No 4, August 2018.

[10] Satish Gajawada, Hassan M. H. Mustafa, Testing Multiple Strategy Human Optimization based Artificial Human Optimization Algorithms, Computer Reviews Journal, vol. 1, no.2, 2018.

[11] Satish Gajawada, Hassan M. H. Mustafa. Hybridization Concepts of Artificial Human Optimization Field Algorithms Incorporated into Particle Swarm Optimization. *International Journal of Computer Applications* 181(19):10-14, September 2018.

[12] Satish Gajawada , Hassan M. H. Mustafa (2018). An Artificial Human Optimization Algorithm Titled Human Thinking Particle Swarm Optimization. *International Journal of Mathematical Research*, 7(1): 18-25. DOI: 10.18488/journal.24.2018.71.18.25

[13] Liu H, Xu G, Ding GY, Sun YB, “Human behavior-based particle swarm optimization”, The Scientific World Journal, 2014.

[14] Ruo-Li Tang, Yan-Jun Fang, "Modification of particle swarm optimization with human simulated property", Neurocomputing, Volume 153, Pages 319–331, 2015.

[15] Muhammad Rizwan Tanweer, Suresh Sundaram, "Human cognition inspired particle swarm optimization algorithm", 2014 IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP), 2014.

[16] M.R. Tanweer, S. Suresh, N. Sundararajan, "Self regulating particle swarm optimization algorithm", Information Sciences: an International Journal, Volume 294, Issue C, Pages 182-202, 2015.

[17] M. R. Tanweer, S. Suresh, N. Sundararajan, "Improved SRPSO algorithm for solving CEC 2015 computationally expensive numerical optimization problems", 2015 IEEE Congress on Evolutionary Computation (CEC), pp. 1943-1949, 2015.

[18] Saptarshi Sengupta, Sanchita Basak, Richard Alan Peters II. Particle Swarm Optimization: A survey of historical and recent developments with hybridization perspectives. https://arxiv.org/abs/1804.05319, 2018.

[19] Yudong Zhang, Shuihua Wang, and Genlin Ji, “A Comprehensive Survey on Particle Swarm Optimization Algorithm and Its Applications,” Mathematical Problems in Engineering, vol. 2015, Article ID 931256, 38 pages, 2015. https://doi.org/10.1155/2015/931256.

[20] M. R. AlRashidi, M. E. El-Hawary. A Survey of Particle Swarm Optimization Applications in Electric Power Systems. IEEE Transactions on Evolutionary Computation. Volume 13, Issue 4, August 2009.

[21] Sharandeep Singh. A Review on Particle Swarm Optimization Algorithm. International Journal of Scientific & Engineering Research, Volume 5, Issue 4, April-2014.

[22] T. Saravanan and V. Srinivasan. Overview of Particle Swarm Optimization. Indian Journal of Science and Technology, Vol 8(32), November 2015.

[23] Muhammad Imran, Rathiah Hashim, Noor Elaiza Abd Khalid.An Overview of Particle Swarm Optimization Variants. Procedia Engineering. Elsevier.Volume 53, Pages 491-496, 2013.

[24] Riccardo Poli, James Kennedy, Tim Blackwell. Particle swarm optimization - An overview. Swarm Intelligence. Volume 1, Issue 1, pp 33–57, Springer, 2007.

[25] Shahri Asta., and Sima Uyar., "A Novel Particle Swarm Optimization Algorithm", In Proceedings of 10th International Conference on Artificial Evolution, Angers, France, 2011.

[26] A.I. Selvakumar, K. Thanushkodi, "A new particle swarm optimization solution to nonconvex economic dispatch problems", IEEE Trans. Power Syst., vol. 22, no. 1, pp. 42-51, 2007.

[27] Jayabarathi, T.; Kolipakula, R.T.; Krishna, M.V.; Yazdani, A. Application and comparison of pso, its variants and hde techniques to emission/economic dispatch. Arabian J. Sci. Eng. 2013, 39, 967–976.

[28] Yang Chunming, Simon Dan, "A New Particle Swarm Optimization Technique", Proceedings of the 18th International Conference on Systems Engineering, 2005.

[29] https://www.sfu.ca/~ssurjano/optimization.html

SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
Volume 7 No. 1
ISSN 2054-7390

Dimension Reduction in Multivariate Linear Regression

William W.S. Chen
Department of Statistics, the George Washington University
Washington D.C. 20013
williamwschen@gmail.com

ABSTRACT

We defined some elementary terminology. It includes the vector space, linear combination, set of independent vectors, dependent vectors, basis of vector space, and direct sum of subspaces. This theory can help us lower the dimension of a given vector spaces. We apply to multivariate linear multiple regression analysis. It not only simplifies the computation and eases the interpretation, but also reduce the rate of errors. Cook (2010) developed an envelope model for the same reason. The main objective in that model is decomposing the covariance matrix into the sum of two matrices, each of whose column spaces either contains, or is orthogonal to, the subspace containing the mean. In other words, break the covariance matrix into the direct sum of the subspaces.

Mathematical Subject Classification: 15-00

Keywords and Phrases: basis of the vector space, direct sum of subspaces, decompose the covariance matrix, envelope model, finite dimensional vector space, linear combinations of vector, linearly dependent vectors, linearly independent vectors, multivariate linear regression, span of the vector space.

1 Introduction

Author worked for Internal Revenue Service for a long period of time. That organization has over one-hundred thousand variables available for study. It is not uncommon to study more than a hundred variables simultaneously at only one time. For example, in a case study there are 130 variables needed to find a relationship between the variables and predict its future trend. We need to run a multivariate regression analysis. Our experiences are calling a conference meeting with the subject matter group members. Using their suggestions, we can break down the variables into smaller groups. Let say 130 variables are cut into six groups of 20 variables each, and 10 variables in the remainder one. In this way, it is not only easier to run the program, but it will also help to interpret the output, and reduce the chance to make errors. Cook (2010) developed the envelope model that decomposed the covariance matrix into the direct sum of two matrices, each of whose column spaces either contains or is orthogonal to the subspace containing the mean. The objective of his model is consistent with ours.

DOI: 10.14738/tmlai.71.6070
Publication Date: 23rd February, 2019
URL: http://dx.doi.org/10.14738/tmlai.71.6070

2 Linear Transformation

Let V be n-dimensional vector space over a field F and let $v_1,\ldots v_N$ be a basis of V over F. If $T \in A(\mathrm{v})$, then T is determined on any vector as soon as we know its action on a basis of V. Since T maps V into V, $\mathrm{v}_1T, v_2T, \ldots . v_nT$ must all be in V. As elements of V, each of these is a realization in a unique way as a linear combination of $v_1,\ldots v_N$ over F. Thus,

$$\mathrm{v}_1T = \alpha_{11}v_1 + \alpha_{12}v_2 + \ldots\ldots\ldots + \alpha_{1n}v_n$$

$$\mathrm{v}_2T = \alpha_{21}v_1 + \alpha_{22}v_2 + \ldots\ldots\ldots + \alpha_{2n}v_n$$

$$\mathrm{v}_\mathrm{n}T = \alpha_{n1}v_1 + \alpha_{n2}v_2 + \ldots\ldots\ldots + \alpha_{nn}v_n$$

Where each $\alpha_{\mathrm{ij}} \in F$ This system of equation can be written more compactly as

$$\mathrm{v}_\mathrm{i}T = \sum_{j=1}^{n} \alpha_{ij}v_j \qquad \mathrm{i}=1,2\ldots.\mathrm{n}\ \mathrm{j}$$

The ordered set of n^2 numbers α_{ij} in F completely describes T. They will serve as the measure of representing T.

Definition 2.1

Let V be an n dimensional vector space over F and let $v_1, \ldots . v_n$

be a basis for v over F. If $\mathrm{T} \in \mathrm{A(V)}$ then the matrix of T in the basis $v_1, \ldots . v_n$ written as m(T), is

$$m(T) = \begin{pmatrix} \alpha_{11} & \alpha_{12} & . & . & \alpha_{1n} \\ \alpha_{21} & \alpha_{22} & . & . & \alpha_{2n} \\ . & . & . & . & . \\ . & . & . & . & . \\ \alpha_{n1} & \alpha_{n2} & . & . & \alpha_{nn} \end{pmatrix}$$

$$where \quad \mathrm{v}_\mathrm{i}T = \sum_{j=1}^{n} \alpha_{ij}v_j$$

Example 2.1 Let the vector space V of 2x2 matrix over R and the following usual basis E of V.

$$\mathrm{E}=\{\mathrm{E}_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}, E_2 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix}, \mathrm{E}_3 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix}, \mathrm{E}_4 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}\}$$

Let $M=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}$ and T_1 be the linear transformation on V defined by $T_1(E)=ME$. Find the matrix representation of T_1 relative to the above usual basis of V.

$$T_1(E_1)=ME_1=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}=\begin{pmatrix}1 & 0\\ 3 & 0\end{pmatrix}=1E_1+0E_2+3E_3+0E_4$$

$$T_1(E_2)=ME_2=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}=\begin{pmatrix}0 & 1\\ 0 & 3\end{pmatrix}=0E_1+1E_2+0E_3+3E_4$$

$$T_1(E_3)=ME_3=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}\begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}=\begin{pmatrix}2 & 0\\ 4 & 0\end{pmatrix}=2E_1+0E_2+4E_3+0E_4$$

$$T_1(E_4)=ME_4=\begin{pmatrix}1 & 2\\ 3 & 4\end{pmatrix}\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}=\begin{pmatrix}0 & 2\\ 0 & 4\end{pmatrix}=0E_1+2E_2+0E_3+4E_4$$

Since dim V=4, any matrix representation of a linear transformation on V must be a 4-square matrix.

Hence $|T_1|_E=\begin{pmatrix}1 & 0 & 2 & 0\\ 0 & 1 & 0 & 2\\ 3 & 0 & 4 & 0\\ 0 & 3 & 0 & 4\end{pmatrix}$

Example 2.2 Let $M=\begin{pmatrix}a & b\\ c & d\end{pmatrix}$ and T_2 V->V be the linear transformation on V defined by $T_2(E)=EM$. Find the matrix representation of T_2 in the above usual basis of V.

$$T_2(E_1)=E_1M=\begin{pmatrix}1 & 0\\ 0 & 0\end{pmatrix}\begin{pmatrix}a & b\\ c & d\end{pmatrix}=\begin{pmatrix}a & b\\ 0 & 0\end{pmatrix}=aE_1+bE_2+0E_3+0E_4$$

$$T_2(E_2)=E_2M=\begin{pmatrix}0 & 1\\ 0 & 0\end{pmatrix}\begin{pmatrix}a & b\\ c & d\end{pmatrix}=\begin{pmatrix}c & d\\ 0 & 0\end{pmatrix}=cE_1+dE_2+0E_3+0E_4$$

$$T_2(E_3)=E_3M=\begin{pmatrix}0 & 0\\ 1 & 0\end{pmatrix}\begin{pmatrix}a & b\\ c & d\end{pmatrix}=\begin{pmatrix}0 & 0\\ a & b\end{pmatrix}=0E_1+0E_2+aE_3+bE_4$$

$$T_2(E_4)=E_4M=\begin{pmatrix}0 & 0\\ 0 & 1\end{pmatrix}\begin{pmatrix}a & b\\ c & d\end{pmatrix}=\begin{pmatrix}0 & 0\\ c & d\end{pmatrix}=0E_1+0E_2+cE_3+dE_4$$

Hence $$|T_2|_E = \begin{pmatrix} a & b & 0 & 0 \\ c & d & 0 & 0 \\ 0 & 0 & a & b \\ 0 & 0 & c & d \end{pmatrix} = \begin{pmatrix} M & 0 \\ 0 & M \end{pmatrix}$$

3 Define Terms

In this section we will define a sequence of useful terminology.

It includes vector space, subspace of a vector space, linear combination of the vector, span of the vectors, set of linear independent vectors, basis of the vector space, dimension of vector space. All these terms are required for us to understand dimension reduction. Define the vector space as follow.

Definition 3.1: A nonempty set V is said to be a vector space over a field F if V is an abelian group under an operation which we denote by +, and if for every $\alpha \in F, v \in V$ there is defined an element written as $\alpha v \in V$ subject to:

$$(1)\alpha(v+w) = \alpha v + \alpha w,\ (2)(\alpha+\beta)v = \alpha v + \beta v,\ (3)\alpha(\beta \mathrm{v}) = (\alpha\beta)\mathrm{v},$$
$$(4)\ 1\mathrm{v} = \mathrm{v} \text{ for all } \alpha, \beta \in \mathrm{F},\ \mathrm{v}, \mathrm{w} \in \mathrm{V}.$$

Where 1 represents the unit element of F under multiplication.

definition 3.2: If V is a vector space over F if $W \subset V$ then W is a subspace of V if under the operation of V, W, itself, forms

a vector space over F. Equivalently, W is a subspace of V whenever $w_1, w_2 \in W,\ \alpha\beta \in \mathrm{F}$ implies that $\alpha w_1 + \beta w_2 \in W$.

Definition 3.3: A linear combination of the vector space

is a sum of scalar multiples of these vectors, that is ,

$c_1 v_1 + c_2 v_2 +, \ldots. c_n v_n$, for some scalar coefficients, $c_1, c_2, \ldots. c_n \in F$. If S is a set of vectors in V, a linear combination of vectors in S is a vector of the form $c_1 v_1 + c_2 v_2 +, \ldots. c_n v_n \text{ with } \mathrm{i} \in \mathrm{N},\ \mathrm{v_i} \in S,\ \mathrm{c_i} \in F$

Definition 3.4: The span of the vectors $c_1, c_2, \ldots. c_n \in V$ is the set of all linear combinations of these vectors, denoted by $span(c_1, c_2, \ldots. c_n)$. If S is a (finite or infinite) set of vectors in V, then the span of S, denoted by span(S), is the set of all linear combinations of vectors in S. If V=span(S), then S spans the vector space V.

Definition 3.5: A (finite or infinite) set of vectors S in V is linearly independent if the only linear combination of distinct vectors in S that produces the zero vector is a trivial linear combination. That is, if $\mathrm{v_i}$ are distinct vectors in S and $c_1 v_1 + c_2 v_2 +, \ldots. c_n v_n = 0, \text{ then } \mathrm{c_1} = \mathrm{c_2} = \ldots. = \mathrm{c_n} = 0$. Vectors that are not linearly independent are linear dependent. That is, there exist distinct vectors $v_1, v_2, \ldots. v_n \in S \text{ and } c_1, c_2, \ldots. c_n$ not all 0 such that

$c_1 v_1 + c_2 v_2 +, \ldots . c_n v_n = 0.$

Definition 3.6: Let V be a vector space over a field F. A set of vectors B in a vector space V is a basis for V if (1) B is a linearly independent set and (2) Span(B)=V.

Definition 3.7: The number of vectors in a basis for a vector space V is the dimension of V denoted by dim(V).

Definition 3.8: The sum of two subspaces is direct if and only if the intersection of these subspaces is zero.

Since our objective is reduction the dimension of multivariate regression vector space using base theory. The following theorems can be found in Mal^Cev(1963) linear algebra. We only claim the theorems and not repeat its proof.

Theorem 3.1 The dimension of the sum of two linear subspaces is the sum of their dimensions minus the dimension of their intersection.

Proof: see Mal^Cev(1963), page 50. Alternatively, see Cook,R.D.,

Li,B. and Chiaromonte,F.(2010), page 936, Corollary 3.1.

Theorem 3.2 The dimension of a direct sum of subspaces is the sum of their dimension.

Proof: By theorem 3.1 the dimension of the sum is the sum of the dimensions minus the dimension of the intersection. From the definition 3.8, the intersection of the subspaces is zero

and has dimension zero. Therefore, the dimension of the direct sum of two subspaces is the sum of their dimensions.

If the number of summands is greater than 2, the proof carried out by induction. From theorem 3.2 we have the following.

Theorem 3.3 If a subspace A is the direct sum of subspaces

$A_1, \ldots\ldots A_n$, then taking a basis $a'_{i1} \ldots\ldots a'_{im_i}$ of each subspace A_i

for i=1,....n and combining these bases into one system,

$a_{11} \ldots\ldots a_{1m_1} \ldots\ldots a_{s1} \ldots\ldots a_{sm_1}$ we obtain a Basis of the subspace A.

The above definitions and three theorems are the foundation of our applications. More of the related facts will be discussed in the section 4. We now give some examples to demonstrate its the real meaning and its possible applications.

Example 3.1 If we are given that

$$A_1 = \begin{pmatrix} 2 & 3 \\ 1 & 5 \end{pmatrix},\ \mathrm{A}_2 = \begin{pmatrix} 1 & 2 \end{pmatrix},\ \mathrm{A}_3 = \begin{pmatrix} 3 \end{pmatrix}\ then$$

$$A_1 \oplus A_2 \oplus A_3 = \begin{pmatrix} 2 & 3 & 0 & 0 & 0 \\ 1 & 5 & 0 & 0 & 0 \\ 0 & 0 & 1 & 2 & 0 \\ 0 & 0 & 0 & 0 & 3 \end{pmatrix}$$

Example 3.2 Let $\mathbf{B} = \{V_1 ......V_n\}$ be a basis for V then

$$V = span(V_1) \oplus .... \oplus span(V_n)$$

Example 3.3 In the direct sum $R(x;2)xR^{2x2}$

$$(2x^2 + 7, \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}) + 3(x^2 + 4x - 2, \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}) = (5x^2 + 12x + 1, \begin{pmatrix} 1 & 5 \\ 0 & 4 \end{pmatrix})$$

4 Some Facts

Although in section 3 we have defined the terms of basis and dimension of a vector space, it may not be good enough in actual application. We realize that these two terms are critical and useful to us. We list more facts to help us to more deeply understand these two terms. Fact 1: Every vector space has a basis. Fact 2: The standard basis for F^n is a basis for F^n, and so dim $F^n = n$. Fact 3: A basis B in a vector space V is the largest set of linearly independent vectors in V that contain B, and it is the smallest set of vectors in V that contains B and spans V. Fact 4: The empty set is a basis for the trivial vector space <0>, and dim<0>=0. Fact 5: If S is a linearly independent set in a vector space V, then S can be expanded, if necessary, to a basis for V. Fact 6: If the set $S = < v_1, ...v_p >$ spans a vector space V, then some subset of S forms a basis for V. For example, if one of the vectors, say v_i, is a linear combination of the remaining vectors, then the set formed from S by removing v_i, will be "closer" to a basis for V. This process can be continued until the remaining vectors form a basis for V.

Fact 7: If a vector space V has a basis containing n vectors, then every basis of V must contain n vectors. Similarly if V has infinite basis, then every basis of V must be infinite. So the dimension of V is unique. Fact 8: Let dim(V)=n and let S be a set containing n vectors. The following are equivalent: (8.1) S is a basis for V. (8.2) S span V. (8.3) S is linear independent. Fact 9: If dim(v)=n, then any subset of V containing more than n vectors is linearly dependent. Fact 10: If $B = < b_1, ...b_p >$ is a basis for a vector space V, then each $x \in V$ can be expressed as a unique linear combination of the vectors in B. That is, for each $x \in V$ there is a unique set of scalars $c_1, c_2, ...c_p$ such that $x = c_1 b_1 + c_2 b_2 + ... + c_p b_p$. The concept of direct sum is the critical important to us. In section 3 we have clearly defined the direct sum and three important theorems. We found that the following fact can help us in real applications. Fact 1: $W = W_1 \oplus W_2$ if and only if $\mathrm{W}=\mathrm{W}_1+\mathrm{W}_2$ and $\mathrm{W}_1 \cap \mathrm{W}_2 = <0>$ Fact 2: If W is a subspace of V, then there exists a subspace U of V such that $V=W\oplus U$. Note that U is not usually unique. Fact 3: $\text{Let } W=W_1+....+W_n$. The following statements are equivalent: (3.1) $\text{Let } W=W_1\oplus....\oplus W_n$. That is, for all i=1…n,

we have $W_i\cap\sum_{j\neq i}W_j=<0>$. (3.2) $W_i\cap\sum_{j=1}^{i-1}W_j=<0>$ for all i=2,......n

(3.3) For each $w\in W$, w can be expressed in exactly one way as a sum of vectors in $W_1....W_n$. That is, there exists a unique $w_i\in W_i$, such that $w=w_1+....+w_n$. (3.4) The subspace w_i, *for* $i=1....n$ are independent. (3.5) If B_i is an ordered basis for W_i, then $B=\bigcup_{i=1}^{n}B_i$ is an ordered basis for W. Fact 4: If B is a basis for V and B is partitioned into disjoint subset B_i, for i=1……n. then $V=\text{span}(B_1)\oplus........\oplus\text{span}(B_n)$. Fact 5: If S is a linearly independent subset of V and S is partitioned into disjoint subsets S_i, for i=1…n, then the subspaces $\text{span}(S_1),........\text{span}(S_n)$ are independent. Fact 6: If V is finite dimensional and $V=W_1+........+W_n$, then $\dim(V)=\dim(W_1)+........+\dim(W_n)$ if and only if $V=W_1\oplus........\oplus W_n$.

5 Concluding Remarks

The Multivariate general linear multiple regression model is given by Y=XB+E where Y is nxp, X is nxq, B is qxp, and E is nxp error matrix. If we move E matrix to the other side of equation and combine with Y matrix, then we can view X matrix as a linear transformation on the parameter matrix B to the response matrix Y. This paper is seeking some square submatrix W_i such that $W_i\cap\sum_{j\neq i}W_j=<0>$ W_i *for* $i=1,2,...p$ and $W=W_1\oplus.....\oplus W_p$. We are sure that such square matrix exist as can be seen from example 2.2. However, the method to identify such submatrix from the given data set may not be trivial. Compare to the Cook (2010) envelope model, we found that our basic objective to reduce the dimension of vector space is identical. Envelope model attempt to decompose the covariance matrix into the direct sum of two submatrix, each of whose column space either contain or is orthogonal to the subspace containing the mean. The only way to do this is to create a split based on the eigenvector of the covariance. Of course, this will lead to a large computer related computation.

REFERENCES

[1] Anderson, T.W. (1958). An Introduction to Multivariate Statistical Analysis. Second Edition, Wiley, New York.

[2] Cook, R.D. (2007). Fisher lecture: Dimension Reduction in Regression (with discussion). Statist. Sci. 22 p1-26.

[3] Cook, R.D., Li, B. and Chiaromonte, F. (2010). Envelope Models for Parsimonious and Efficient Multivariate Linear Regression. Statistica Sinica 20, p927-1010.

[4] Cook, R.D., Li, B. and Chiaromonte, F. (2007). Dimension Reduction without matrix inversion. Biometrika, 94, p569-584.

[5] Halmos, P.R. (1974,1987). Finite-Dimensional Vector Space. Published by Springer-Verlag, New York Inc.

[6] Handbook of Linear Algebra. (2007). Edited by Leslie Hogben,Associate editors, Richard Brualdi, Anne Greenbaum, Roy Mathias. Chapman & Hall/CRC, Taylor & Francis Group, Boca Raton, London, New York.

[7] Mal^Cev, A.I.(1963). Foundations of Linear Algebra. Translated from the Russian by Thomas Craig Brown, Edited by Roberts, J.B. Published by W.H. Freeman and Company, San Francisco and London.

SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
Volume 7 No. 1
ISSN 2054-7390

Named Entity Recognition for Characteristic of Medical Herbs Using Modified HMM Approach

Lailil Muflikhah, Agung Setiyono, Nurul Hidayat
Faculty of Computer Science; Brawijaya University; Malang, Indonesia
lailil@ub.ac.id; setiyoajiagung@gmail.com; ntayadih@ub.ac.id

ABSTRACT

The amount of articles in medicinal herbs is very huge. It is performed with unstructured format so that it takes time to get information as reader's need. Therefore, this research purposes to recognize the name entity of article from internet in order to increase information retrieval or other analysis data purposes. Named entity recognition is one of the goals of information extraction which is to identify the name and characteristics of the herbs. This paper is propose the modified method of Hidden Marcov Model (HMM) with Viterbi algorithm. In this method, it is enclosed gazetteer list for labeling name and location of data training to construct HMM. The data sets are taken from three web sites including: miliaton, aliweb, and plants. As a result, the performance is achieved at average precision value of 0.93, recall of 0.83 and f-measure of 0.85.

Keywords: Hidden Marcov Model, Gazetteer, Viterbi, Named entity, Medicinal herbs

1 Introduction

The number of websites about health is more and more increase various information including about health. Recently, medicinal herbs is main issue on medical analysis. Many people assume that there is no side effect to consume as medicine. Website is a site to share information through internet. The growth of information in internet is very huge, then it need to classify by information extraction based on herbs' characteristic.

Named Entity Recognition (NER) is a subtask of information extraction to identify and classify named entities in document text into pre-defined user's categories such as name of plant, location, characteristic, and advantages. In general, there are two methods of NER, i.e. rule based approach and machine learning based approach. The first method is need information from domain expert to construct many patterns. However, the second method, the pattern is generated from training data. Therefore, this research is proposed to combine the both methods.

The related research of NER is conducted by Todorovic, et al. that Hidden Marcov Model (HMM) in English text disregard to the grammar is achieved at accuracy of 91,71% [1]. HMM is a machine learning method with sequence probability model to solve the problem. It is need a set of training data for labeling words as POS-Tagging. The other research is conducted by Alfred et al., [2]. It is about named entity recognition for Malay article using rule based approach, but the accuracy is achieved of 89.47%. Furthermore, the NER

DOI: 10.14738/tmlai.71.6086
Publication Date: 23rd February, 2019
URL: http://dx.doi.org/10.14738/tmlai.71.6086

is applied to health article using Support Vector Machine, a machine learning method, and the accuracy rate is 90% [3].

2 Research Methods

This research is applied to the modified HMM with Viterbi algorithm for identifying named entity in news article of health. The method is involved rule based approach. Generally, the steps of named entity recognition system is shown in Figure 1, the block diagram. The first step is preprocessing data, including tokenizing only. In this steps, it is also put labelling for named entity of training set. The rule method is address to support the HMM construction. It is applied to find the pattern of word sequence, especially for ambiguity of term. Then, the next step is to construct the HMM model of term sequence. And as a result is the named entity of herb.

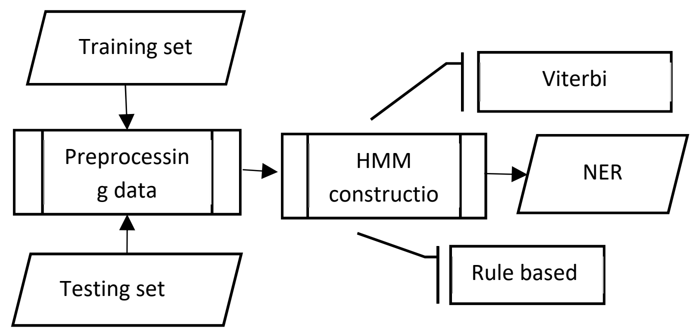


Figure 1. Block diagram of Named Entity Recognition

2.1 Text Mining

Text document has unstructured format which need to be transformed in structured format in order to get useful meaning of information and it is known as text mining. Information extraction is process to get information from document text [4]. This method is applied after preprocessing step, including: tokenizing, filtering, and stemming. It is address to index the term and make easy to create the HMM model and constructing the gazetteer in rule based.

2.2 Preprocessing Data

Preprocessing is a preliminary step which involves tokenization, stop words removal, stemming and misspellings words normalization. It involves tokenization, stop words removal, stemming and normalization including query expansion. Tokenization is a process removing punctuation, numbers, and characters other than the alphabet [5]. It is also conducted case folding, which is changing all capital letters into lowercase. Then, stop words removal or filtering is removing uninformative words referring to the existing stop word dictionary. Meanwhile, stemming is a process to convert every words to its root. This process is done by removing affixes such as prefix, infix and suffix.

2.3 Named Entity Recognition (NER)

Named entity recognition is information extraction derivation which is identify the type of word (entity) in document. In this research, it is used to identify name, substance and characteristics of herbal plants. The identification is proposed to organize the document in order to make easy for information retrieval. NER is an important tool in almost all of the Natural Language processing applications such as Information Retrieval (IR), Information Extraction (IE), Question Answering (QA), Machine Translation (MT) and

Automatic Summarization (AS) etc. NER can be defined as a two stage problem:- Identification of Proper Noun and classification of the Proper Noun into a set of classes such as Plant names, Location names, Substances (chemical). These words are collectively defined as “ENAMEX” by MUC-6[6]. Thus NER can be said as the process of identifying and classifying the tokens into the above predefined classes.

Basically, there are two approaches of NER, i.e. rule based approach and machine learning based approach. The first approach is concerned to manual rules or regular expression. Many rules based NER contains Lexicalized Grammar, Gazetteer list, and List of triggered words. Then, the machine learning based approach is concerned to pre-defined method, such as: Hidden Marcov Models (HMM), Decision Trees, Maximum Entropy (ME), SVM and CRF. Therefore, this research is proposed combined the both approach. The gazetteer list is used to support the HMM method.

2.4 Hidden Marcov Model (HMM)

Hidden Marcov Model (HMM) is one of NER approaches which based on statistical method and is depend on the sequence of word. Therefore, this research is developed and applied to statistical approach and combined to rule based using gazetteer list. HMM is a statistical method that involves the sequence probability of term-document. The method is a machine learning method using sequence probability approach to solve the problem. Each word is put pre-labelling of POS-Tagging to construct markov model [1] as in Equation 1.

$$\lambda = (A, B, \pi) \quad (1)$$

where: λ is marcov model, B is emission probability, and π is initial probability. There are five tuples in modelling HMM, including observed state (O), hidden state (Q), transition probability matrix (A), emission probability matrix (B) and initial state probability (π) [7].

1. Observed state (O)

   In the observed state, it is made with the symbols $O = O_1, O_2, O_3, \ldots O_n$ observed state, the observed modeling is observed.

2. Hidden state (Q)

   Hidden State is a state that is hidden and cannot be observed, symbolized by $Q = Q_1, Q_2, Q_3, \ldots Q_n$.

3. Transition Probability Matrix (*A*)

   Transition probability is an probability to move from state i to state j. It is symbolized by $A = a_{01}, a_{02}, a_{n1}, \ldots a_{nm}; a_{ij}$, the number of transition probability matrix is Q x Q.

4. Emission Probability Matrix (B)

   Emission probabilities are an probability to move the state i with the O_t (Observed State) time requirement. Symbolized by $B = b_i (O_t)$ the number of emission probability matrix is Q x O.

5. Initial State Probabilities (π)

   The initial probability is symbolized by π. In named entity recognition that is the number of names of word entities, if the chance for each name of the entity word to be added will be worth one.

2.5 Viterbi Algorithm

Viterbi algorithm is an algorithm to optimally find the sequence of hidden state from real problem. It is used Viterbi trellis, reverse counting or recursive. The Viterbi algorithm is implemented to find the most likely tag sequence in the state space of the possible tag distribution based on the state transition probabilities [8] .The Viterbi algorithm allows us to find the optimal tags in linear time. The idea of the algorithm is only the most probable of all the state sequences to be considered.

2.6 Gazetteer List Method

Gazetteer list method is a rule based approach to construct the list of for different Named Entities and then applies search operations to classify the names [9]. In this research, the method needs two types of input to collection of gazetteer, one for Name of herb and second for Location of herb. The list is automatically created from scanning the document with label name or location.

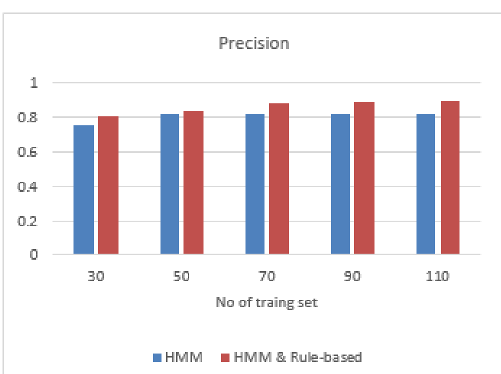
3 Result and Discussion

Furthermore, the comparison of performance between HMM and proposed method to recognize the named entity of characteristic for herbal plant is shown on Table 1. As general, the validity of modified HMM is better than the HMM method only. However, recognition of name and location is unreachable 100% (1). This indicates that there is ambiguity of two or more word to be recognized in the same named entity.

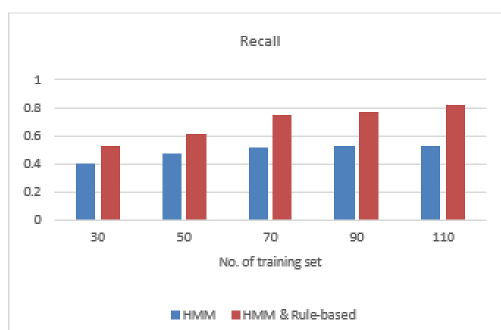
Table 1. Performance of accuracy result

Named Entity	Proposed method			HMM		
	Precision	recall	f-measure	Precision	recall	f-measure
NAME	0.847	0.639	0.655	0.748	0.422	0.448
LOC	0.929	0.895	0.881	0.927	0.774	0.776
SUBST	0.992	0.895	0.929	0.992	0.812	0.860
FUNC	0.953	0.908	0.922	0.926	0.829	0.842

Then, the 2nd scenario is to know the effect of the number of training set in HMM model construction and gazetteer list of rule for terms sequence. The experimental result shows that the more number of the training set, the higher of performance is. Also, the proposed method has the higher performance than the HMM method as shown at Figure 2. (a), (b), and (c).



(a) Precision



(a) Recall

(a) F-measure

Figure 2. Performance of the number of training set

As general, the performance result of the proposed method is higher than the HMM using viterbi algorithm only. It is shown the bar chart of comparison for the both method as in Figure 3.

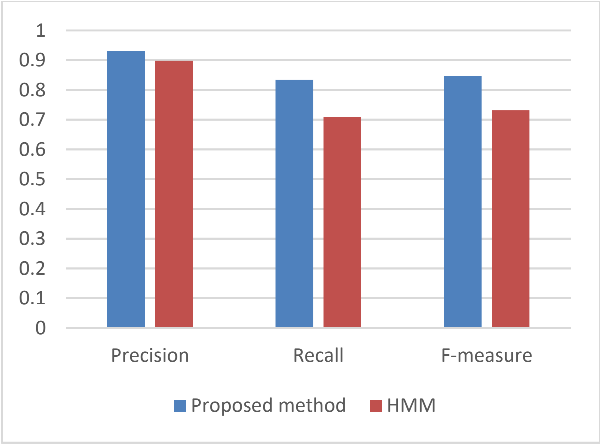


Fig.3. Comparison of average accuracy result

4. Conclusion

The proposed method which involving rule based to HMM method has been applied to the named entity recognition for herbs article. The entity is including name of herbs, substance, location and their function. The rule is applied into constructing HMM process for labelling name and location in term sequences of training set. The rule is address to reduce the ambiguity of words. It is impact for modelling in HMM. Generally, the performance result of the proposed method is better than the conventional method of HMM. The accuracy rate is also depend on the number of training set. The more number of data set, the higher of the performance is.

REFERENCES

[1] Alfred, R., Leong, L.C., Kim On, C., Antony, P., et.al. (2013). Named Entity Recognition for Malay Articles. Lecture Notes in Computer Science (LNCS, Volume 8346. pp:288-299).DOI: 10.1007/978-3-642-53914-5_25

[2] Todorovic, B.T., Rancic, S.R., Markovic, L.M., Mulalic, E.H., Dan Ilic, V.M. (2008). *Named Entity Recognition And Classification Using Context Hidden Markov Model* . Symposium On Neural Network Applications In Electrical Engineering. Neurel-2008.

[3] Suwarningsih, W., Supriana, I., and Purwarianti, A.(2014). *Imner Indonesian Medical Named Entity Recognition*. 2nd International Conference On Technology, Informatics, Management, Engineering & Environment.

[4] Sumathy, K.L., Chidambaram, M. (2013). *Text Mining: Concept, Applications, Tools and Issues- An Overview.* International Journal of Computer Applications (00975-8887). Volume 80(4).

[5] Fauzi, M.A., Arifin, A.Z. and Yuniarti, (2017). An Arabic Book Retrieval using Class and Book Index Based Term Weighting. *International Journal of Electrical and Computer Engineering (IJECE). Volume 7*(6)

[6] David Nadeau, Satoshi Sekine , "A survey of named entity recognition and classification" National Research Council Canada / New York University

[7] Lin, J. Dan Dyer, C. (2010). Data-Intensive Text Processing with Mapreduce. Available https://Lintool.Github.Io/Mapreducealgorithms/Mapreduce-Book-Final.Pdf

[8] Viterbi, A. (1967). Optimum decoding algorithm for convolutional codes. In IEEE Trans. Info. Theory (supported by AFOSR)

[9] Padmaja Sharma, Utpal Sharma, Jugal Kalita. (2011), "Named Entity Recognition: A Survey for the Indian Languages".

SOCIETY FOR SCIENCE AND EDUCATION UNITED KINGDOM
TMLAI TRANSACTIONS ON MACHINE LEARNING AND ARTIFICIAL INTELLIGENCE
Volume 7 No. 1
ISSN 2054-7390

Investigation of the Proof Complexity Measures of Strongly Equal K-Tautologies in Some Proof Systems

Anahit Chubaryan, Artur Khamisyan, Garik Petrosyan
Department of Informatics and Applied Mathematics
Yerevan State University
achubaryan@ysu.am, Artur.Khamisyan@gmail.com, garik.petrosyan.1@gmail.com

ABSTRACT

Here we generalize the notions of determinative conjunct and strongly equal tautologies formany-valued logic (MVL) and compare the proof complexity measures of strongly equal many-valued tautologies in some proof systems of MVL. It is proved that in some "weak" proof system the strongly equal many-valued tautologies have the same proof complexities, while in the "strong" proof systems the measures of proof complexities for strongly equal tautologies can essentially differ from each other.

Keywords: many-valued logic, determinative conjunct, strongly equal tautologies, proof complexity characteristics.

1 Introduction

In the mean time many interesting applications of many-valued logic (MVL) were found in such fields as Logic, Mathematics, Formal Verification, Artificial Intelligence, Operations Research, Computational Biology, Cryptography, Data Mining, Machine Learning, Hardware Design etc., therefore the investigations of proof complexity for different systems of MVL are very important.

The traditional assumption that all tautologies as Boolean functions are equal to each other is not fine-grained enough to support a sharp distinction among tautologies. The authors of [1] have provided a different picture of equality for classical tautologies. They have introduced the notion of strong equality of 2-valued tautologies on the basis of determinative conjunct notion. The idea to revise the notion of equivalence between tautologies in such way that is takes into account an appropriate measure of their "complexity".

It was proved in [2,3] that in "weak" proof systems the strongly equal 2-valued tautologies have the same proof complexities, while in the "strong" proof systems the measures of proof complexities for strongly equal tautologies can essentially differ from each other.

Here we generalize the notions of determinative conjunct and strongly equal tautologies for MVL and compare the proof complexity measures of **strongly equal many-valued tautologies** in some proof systems of MVL.

DOI: 10.14738/tmlai.71.6187
Publication Date: 23rd February, 2019
URL: http://dx.doi.org/10.14738/tmlai.71.6187

2 Preliminaries.

2.1 Main notions and notations of k-valued logic.

Let E_k be the set $\left\{0, \frac{1}{k-1}, \ldots, \frac{k-2}{k-1}, 1\right\}$. We use the well-known notions of propositional formula, which defined as usual from propositional variables with values from E_k, (may be also propositional constants), parentheses (,), and logical connectives $\&$, $\vee$, $\supset$, $\neg$, every of which can be defined by different mode. Additionaly we use two modes of exponential function p^{σ} and introduce the additional notion of formula: for every formulas A and B the expression $\boldsymbol{A^B}$ (for both modes) is formula also.

In the considered logics either only **1** or every of values $\frac{1}{2} \leq \frac{\boldsymbol{i}}{\mathbf{k-1}} \leq 1$ can be fixed as ***designated values***.

Definitions of main logical functions are:

$\boldsymbol{p} \vee \boldsymbol{q} = max(p, q)$ (1) disjunction or

$\boldsymbol{p} \vee \boldsymbol{q} = ((k-1)(p+q))(mod\ k)/(k-1)$ (2) disjunction,

$\boldsymbol{p}\&q = min(p, q)$ (1) conjunction or

$\boldsymbol{p}\&q = \max(\boldsymbol{p} + q - 1, 0)$ (2) conjunction

Sometimes (1) conjunction is denoted by **^**.

For implication we have two following versions:

$$\boldsymbol{p} \supset \boldsymbol{q} = \begin{cases} 1, & for\ \ p \leq q \\ 1 - p + q, & for\ \ p > q \end{cases}$$ (1) Łukasiewicz's implication or

$$\mathbf{p} \supset \boldsymbol{q} = \begin{cases} 1, & for \quad p \leq q \\ q, & for \quad p > q \end{cases}$$ (2) Gödel's implication

And for negation two versions also:

$\neg\boldsymbol{p} = 1 - p$ (1) Łukasiewicz's negation or

$\neg\boldsymbol{p} = ((k-1)p + 1)(mod\ k)/(k-1)$ (2) cyclically permuting negation.

Sometimes we can use the notation $\overline{\boldsymbol{p}}$ instead of $\neg\boldsymbol{p}$.

For propositional variable ***p*** and $\boldsymbol{\delta} = \frac{i}{k-1}$ $(0 \leq i \leq k\text{-}1)$ additionally "exponent" functions are defined in (4):

$\boldsymbol{p^{\delta}}$ as $(p \supset \delta)\&(\delta \supset p)$ with (1) implication (1) exponent,

$\boldsymbol{p^{\delta}}$ as *p* with $(k\text{-}1)(1 - \boldsymbol{\delta})$ (2) negations. (2) exponent.

Note, that both (1) exponent and (2) exponent are no new logical functions.

If we fix "**1**" (every of values $\frac{1}{2} \leq \frac{i}{k-1} \leq 1$) as designated value, so a formula ϕ with variables $p_1, p_2, \ldots p_n$ is called ***1-k*-tautology** (***≥1/2-k-tautology***) if for every $\tilde{\delta} = (\delta_1, \delta_2, \ldots, \delta_n) \in E_k^n$ assigning δ_j ($1 \leq j \leq n$) to each p_j gives the value 1 (or some value $\frac{1}{2} \leq \frac{i}{k-1} \leq 1$) of ϕ.

Sometimes we call *1-k*-tautology or *≥1/2-k-tautology* simply *k*-tautology.

2.2 Determinative Disjunctive Normal Form for MVL

The notions of determinative conjunct and determinative disjunctive normal forms are introduced at first in [1]. Based on these notions some new proof system for classical propositional logic, dual to resolution system, was defined. Then the analogous systems were given for intuitionistic, minimal, monotone, positive and some others two-valued propositional logics.

The notions of determinative conjunct and determinative disjunctive normal form are generalized for all variants of MVL in [4]. For every propositional variable p in k-valued logic $p^0, p^{1/_{k-1}}, \ldots, p^{k-2/_{k-1}}$ and p^1 in sense of both exponent modes are the literals. The conjunct K (term) can be represented simply as a set of literals (no conjunct contains a variable with different measures of exponents simultaneously), and DNF can be represented as a set of conjuncts.

Each of the following trivial identities for a propositional formula $\boldsymbol{\psi}$ are called *replacement-rule*:

for both conjunction and (1) disjunction

$$\varphi \& 0 = 0 \& \varphi = 0, \quad \varphi \vee 0 = 0 \vee \varphi = \varphi, \quad \varphi \& 1 = 1 \& \varphi = \varphi, \quad \varphi \vee 1 = 1 \vee \varphi = 1,$$

for (2) disjunction

$$\left(\varphi \vee \frac{i}{k-1}\right) = \left(\frac{i}{k-1} \vee \varphi\right) = \overbrace{\neg\neg \ldots \neg}^{i} \varphi \qquad (0 \leq i \leq k-1),$$

for (1) implication

$\varphi \supset 0 = \bar{\varphi}$ with (1) negation, $0 \supset \varphi = 1$, $\varphi \supset 1 = 1$, $1 \supset \varphi = \varphi$,

for (2) implication

$\varphi \supset 1 = 1$, $0 \supset \varphi = 1$, $\varphi \supset 0 = \overline{sg}\varphi$, where $\overline{sg}\varphi$ *is* 0 for φ>0 and 1 for φ=0,

for (1) negation

¬(i/k-1)=1-i/k-1 (0≤i≤k-1), $\neg\boldsymbol{\psi} = \boldsymbol{\psi}$,

for (2) negation

$$\neg(i/k-1) = i+1/k-1 \ (0 \leq i \leq k-2), \quad \neg\mathbf{1} = \mathbf{0}, \ \overbrace{\neg\neg \ldots \neg}^{\boldsymbol{k}} \boldsymbol{\psi} = \boldsymbol{\psi}.$$

Application of a replacement-rule to some word consists in replacing of its subwords, having the form of the left-hand side of one of the above identities, by the corresponding right-hand side.

The following *auxiliary relations for replacement* are introduced in [5] as well:

for both variants of conjunction

$\left(\varphi\&\frac{i}{k-1}\right)=\left(\frac{i}{k-1}\&\varphi\right)\leq\frac{i}{k-1}\qquad(1\leq i\leq k-2),$

for (1) implication

$\left(\varphi\supset\frac{i}{k-1}\right)\geq\frac{i}{k-1}$ and $\left(\frac{i}{k-1}\supset\varphi\right)\geq\frac{k-(i+1)}{k-1}\qquad(1\leq i\leq k-2),$

for (2) implication

$\left(\varphi\supset\frac{i}{k-1}\right)\geq\frac{i}{k-1}\qquad(1\leq i\leq k-2),\quad\left(\frac{i}{k-1}\supset\varphi\right)\geq\varphi\qquad(1\leq i\leq k-1).$

Let $\boldsymbol{\varphi}$ be a propositional formula of *k*-valued logic, $\boldsymbol{P}=\{\boldsymbol{p_1},\boldsymbol{p_2},\dots,\boldsymbol{p_n}\}$ be the set of all variables of ϕ and $\boldsymbol{P'}=\{\boldsymbol{p_{i_1}},\boldsymbol{p_{i_2}},\dots,\boldsymbol{p_{i_m}}\}$ $(\mathbf{1}\leq\boldsymbol{m}\leq\boldsymbol{n})$ be some subset of $\boldsymbol{P}$.

Definition 1: Given $\widetilde{\boldsymbol{\sigma}}=(\boldsymbol{\sigma_1},\boldsymbol{\sigma_2},\dots,\boldsymbol{\sigma_m})\in\boldsymbol{E}_{\mathbf{k}}^{\boldsymbol{m}}$, the conjunct $\boldsymbol{K^{\sigma}}=\{\boldsymbol{p_{i_1}}^{\boldsymbol{\sigma_1}},\boldsymbol{p_{i_2}}^{\boldsymbol{\sigma_2}},\dots,\boldsymbol{p_{i_m}}^{\boldsymbol{\sigma_m}}\}$ is called $\boldsymbol{\varphi}-\frac{\boldsymbol{i}}{\boldsymbol{k-1}}$-determinative $(\mathbf{0}\leq\boldsymbol{i}\leq\boldsymbol{k}-\mathbf{1})$, if assigning $\boldsymbol{\sigma_j}$ $(\mathbf{1}\leq\boldsymbol{j}\leq\boldsymbol{m})$ to each $\boldsymbol{p_{i_j}}$ and successively using replacement-rules and, if it is necessary, the auxiliary relations for replacement also, we obtain the value $\frac{\boldsymbol{i}}{\boldsymbol{k-1}}$ of $\boldsymbol{\varphi}$ independently of the values of the remaining variables.

Every $\boldsymbol{\varphi}-\frac{\boldsymbol{i}}{\boldsymbol{k-1}}-$determinative conjunct is called also $\boldsymbol{\varphi}$-determinative or determinative for $\boldsymbol{\varphi}$.

Example. It is not difficult to see that the conjuncts $\{\mathrm{p}_1\}$, $\{\neg\neg\mathrm{p}_3\}$, $\{\mathrm{p}_2\}$, $\{\neg\mathrm{p}_1,\neg\mathrm{p}_2\}$ are determinative for formula$(\mathrm{p}_1\supset\mathrm{p}_2)\supset(\mathrm{p}_3\supset(\neg\mathrm{p}_2\supset\mathrm{p}_1))$ in 3-valued Łukasiewicz's system based on (1) conjunction, (1) disjunction, (1) implication, (1) negation and (1) exponent. Note that correctness of this statement for conjunct $\{\neg\mathrm{p}_1,\neg\mathrm{p}_2\}$ must be proved by using the auxiliary relations for replacement as well.

Definition 2. A DNF $\boldsymbol{D}=\{\boldsymbol{K_1},\boldsymbol{K_2},\dots,\boldsymbol{K_j}\}$ is called determinative DNF (dDNF) for $\boldsymbol{\varphi}$ if $\boldsymbol{\varphi}=\boldsymbol{D}$ and if "1" (every of values $\frac{\mathbf{1}}{\mathbf{2}}\leq\frac{\boldsymbol{i}}{\mathbf{k-1}}\leq\mathbf{1}$) is (are) fixed as designated value, then every conjunct $\boldsymbol{K_i}(\mathbf{1}\leq\boldsymbol{i}\leq\boldsymbol{j})$ is 1-determinative ($\frac{\boldsymbol{i}}{\mathbf{k-1}}-\textbf{determinative from indicated intervale}$) for $\boldsymbol{\varphi}$.

Remark As in [3] it is also easily proved, that

1) if for some *k*-tautology $\boldsymbol{\varphi}$, the minimal number of literals, containing in $\boldsymbol{\varphi}$-determinative conjunct, is $\boldsymbol{m}$,then $\boldsymbol{\varphi}$-determinative DNF has at least $\mathbf{k}^{\boldsymbol{m}}$conjuncts;

2) if for some *k*-tautology $\boldsymbol{\varphi}$ there is such $\boldsymbol{m}$ that every conjunct with $\boldsymbol{m}$ literals is $\boldsymbol{\varphi}$-determinative, then there is $\boldsymbol{\varphi}$-determinative DNF with no more than $\mathbf{k}^{\boldsymbol{m}}$conjuncts.

Main Definition. The k- tautologies φ and ψ are strongly equal in given version of many-valued logic if every φ -determinative conjunct is also ψ -determinative and vice versa.

2.3 Definition of considered systems.

First of considered system is the following universal elimination system **UE** for all versions of MVL, which is defined in mentioned paper [5].

The axioms of Elimination systems **UE** aren't fixed, but for every formula $\boldsymbol{k-valued}\ \boldsymbol{\varphi}$ each conjunct from some DDNF of $\boldsymbol{\varphi}$ can be considered as an axiom.

For k-valued logic the inference rule is ***elimination rule*** (ε-rule)

$$\frac{K_0\cup\{p^0\},\ K_1\cup\left\{p^{\frac{1}{k-1}}\right\},\ \ldots,\ K_{k-2}\cup\left\{p^{\frac{k-2}{k-1}}\right\},\ K_{k-1}\cup\{p^1\}}{K_0\cup K_1\cup\ldots\cup K_{k-2}\cup K_{k-1}},$$

where mutual supplementary literals (variables with corresponding (1) or (2) exponents) are eliminated.

A finite sequence of conjuncts such that every conjunct in the sequence is one of the axioms of **UE** or is inferred from earlier conjuncts in the sequence by ε-rule is called a proof in **UE**.

A DNF $D=\{K_1,K_2,\ldots,K_l\}$ is k-tautological if by using ε-rule can be proved the empty conjunct ($\emptyset$) from the axioms $\{K_1,K_2,\ldots,K_l\}$.

We consider also the well-known Frege style systems of MVL. We define Gödel's (**G**) and Łukasiewicz's (**L**) systems following [6,7].

Łukasiewicz's proof system (**L**) uses (1) definitions for all logical functions.

For every formula *A, B, C* of k-valued logic the following formulas are axioms schemes of ***L*** [6]*:*

1. $A\supset(B\supset A)$ **2.** $(A\supset(B\supset C))\supset(B\supset(A\supset C))$

3. $(A\supset B)\supset((B\supset C)\supset(A\supset C))$

4. $(A\supset(A\supset B))\supset((\neg B\supset(\neg B\supset\neg A))\supset(A\supset B))$

5. $(A\supset B)\supset(\neg B\supset\neg A)$ **6**. $A\supset\neg\neg A$ **7**. $\neg\neg A\supset A$

8. $A\&B\supset B$ **9**. $A\&B\supset A$ **10**. $(C\supset A)\supset((C\supset B)\supset(C\supset A\&B))$

11. $A\supset A\vee B$ **12**. $B\supset A\vee B$ **13**. $(A\supset C)\supset((B\supset C)\supset(A\vee B\supset C))$

Inference rule is modus ponens /m.p./ $A, A\supset B\vdash B$.

Gödel's proof system (**G)** uses (1) definitions for conjunction and disjunction, (2) for implication and negation.

For every formula *A, B, C* of k-valued logic the following formulas are axioms schemes of **G** [7]:

1.$(A\supset B)\supset((B\supset C)\supset(A\supset C))$

2. $A\supset A\vee B$ 3. $B\supset A\vee B$ 4. $(A\supset C)\supset((B\supset C)\supset(A\vee B\supset C))$

5. $A\&B\supset B$ 6. $A\&B\supset A$ 7. $(C\supset A)\supset((C\supset B)\supset(C\supset A\&B))$

8. $(A\supset(B\supset C))\supset(\&B\supset C)$ 9. $(A\&B\supset C)\supset(A\supset(B\supset C))$

10. $A\&\neg A\supset B$ 11. $(A\supset\&\neg A)\supset\neg A$ 12.$(A\supset B)\vee(B\supset A)$.

Inference rule is modus ponens /m.p./ $A, A\supset B\vdash B$.

In both systems "**1**" is fixed as designated value.

2.4 Proof complexity measures

In the theory of proof complexity two main characteristics of the proof are: $t-\mathbf{complexity}$, defined as the number of proof steps (length) and $l-\mathbf{complexity}$, defined as total number of proof symbols (size). We consider two measures (space and width) as well : $s-\mathbf{complexity}$ (space), informal defined as maximum of minimal number of symbols on blackboard, needed to verify all steps in the proof and $w-\mathbf{complexity}$ (width), defined as the maximum of widths of proof formulas (the strong definitions of all proof complexity characteristics see in [9]).

Let $\mathbf{\Phi}$ be a proof system and $\boldsymbol{\varphi}$ be a *k*-tautology. We denote by $\boldsymbol{t_{\varphi}^{\Phi}}(\boldsymbol{l_{\varphi}^{\Phi}}, \boldsymbol{s_{\varphi}^{\Phi}}, \boldsymbol{w_{\varphi}^{\Phi}})$ the minimal possible value of $\boldsymbol{t-complexity}$ $(\boldsymbol{l-complexity}, \boldsymbol{s-complexity}, \boldsymbol{w-complexity})$ for all proofs of tautology $\boldsymbol{\varphi}$ in $\mathbf{\Phi}$.

By $|\boldsymbol{\varphi}|$ we denote the size of a formula $\boldsymbol{\varphi}$, defined as the number of all logical signs entries. It is obvious that the full size of a formula, which is understood to be the number of all symbols is bounded by some linear function in$|\boldsymbol{\varphi}|$.

2.5 Essential subformulas of tautologies

For proving the main results we use also the notion of *essential subformulas*, introduced in [8].

Let *F* be some formula and $Sf(F)$ be the set of all non-elementary subformulas of formula F .

For every formula F , for every $\varphi \in Sf(F)$ and for every variable p by F_{φ}^{p} is denoted the result of the replacement of the subformulas φ everywhere in F by the variable p . If $\varphi \notin Sf(F)$, then F_{φ}^{p} is F .

We denote by $Var(F)$ the set of variables in F .

Definition 3. Let p be some variable that $p \notin Var(F)$ and $\varphi \in Sf(F)$ for some tautology F . We say that φ is an ***essential subformula*** in F iff F_{φ}^{p} is non-tautology.

We denote by $Essf(F)$ the set of essential subformulas in tautology F .

If F is minimal tautology, i.e. F is not a substitution of a shorter tautology, then $Essf(F) = Sf(F)$.

It is not difficult to prove the following statement.

Proposition. Let F be a minimal tautology and $\varphi \in Essf(F)$, then in every ***L***-proof (***G***-proof) of F subformula φ must be essential either at least in some axiom of this system. .

Really, if some subformula is essential in formula B, which is derived from formlas A and $A \supset B$, then this subformula must be essential in A or in $A \supset B$.

Note that for both systems ***L*** and ***G*** the number of essential subformulas in every axioms is bounded with some constant.

3 Main results.

Here we compare the proof complexities measures of strongly equal *k- tautologies* in above defined systems of some versions of MVL.

Theorem 1. The strongly equal *k*-tautologies have the same *t,l,s,w* complexities in the systems ***U E*** for all versions of MVL.

The proof is based on the fact that refutations in the systems UE deal exclusively with the conjuncts of dDNF, which are the same for strongly equal tautologies.

The situation for the systems ***L*** and ***G*** is the essentially other.

For simplification of our result presentation, we demonstrate them only for 3-tautoogies.

Let us consider

a) for Łukasiewicz's 3-valued logic the following two 3-tautologies:

$A_n = (p^1 \& p^{1/2} \& p^0)^{1/2} \supset ((p^1 \& p^{1/2} \& p^0)^1 \supset (\overbrace{\neg\neg\ldots\neg}^{2n} (p^1 \vee p^{1/2} \vee p^0)))$ *with* (1) exponent, $(n \geq 0)$,

$B_n = (p^1 \vee p^{1/2} \vee p^0) \& (\overbrace{\neg\neg\ldots\neg}^{2n} (p^1 \vee p^{1/2} \vee p^0)))))$ *with* (1) exponent, $(n \geq 0)$,

b) for Gödel's 3-valued logic the following two 3-tautologies:

$C_n = \neg(\neg\neg p \& \neg p \& p) \supset ((\neg\neg p \& \neg p \& p) \supset (\overbrace{\neg\neg\ldots\neg}^{3n} (\neg\neg p \vee \neg p \vee p)))$ $(n \geq 0)$,

$D_n = (\neg\neg p \vee \neg p \vee p) \& (\overbrace{\neg\neg\ldots\neg}^{3n} (\neg\neg p \vee \neg p \vee p)))))$ $(n \geq 0)$.

It isn't difficult to see that dDNF for both A_n and B_n is $\{p^1, p^{1/2}, p^0\}$ and for both C_n **and** D_n is $\{\neg\neg \boldsymbol{p}, \neg \boldsymbol{p}, \boldsymbol{p}\}$, therefore A_n and B_n are strongly equal and C_n **and** D_n are strongly equal as well.

Note also that the sizes of all above formulas are Θ(n).

Theorem 2. a) $\boldsymbol{t}^{\boldsymbol{L}}_{\mathrm{An}} = O(1)$, $\boldsymbol{l}^{\boldsymbol{L}}_{\mathrm{An}} = O(n)$

$\boldsymbol{t}^{\boldsymbol{L}}_{\mathrm{Bn}} = \boldsymbol{\Omega}(\boldsymbol{n})$, $\boldsymbol{l}^{\boldsymbol{L}}_{\mathrm{Bn}} = \Omega(n^2)$.

b) $\boldsymbol{t}^{\boldsymbol{G}}_{\mathrm{Cn}} = O(1)$, $\boldsymbol{l}^{\boldsymbol{G}}_{\mathrm{Cn}} = O(n)$,

$\boldsymbol{t}^{\boldsymbol{G}}_{\mathrm{Dn}} = \boldsymbol{\Omega}(\boldsymbol{n})$, $\boldsymbol{l}^{\boldsymbol{G}}_{\mathrm{Dn}} = \Omega(n^2)$.

Proof. a) We can derive A_n as follow.

At first we derive the 3-tautology $(p^1 \& p^{1/2} \& p^0)^0$, then the 3-tautology

$(p^1 \& p^{1/2} \& p^0)^0 \supset ((p^1 \& p^{1/2} \& p^0)^{1/2} \supset ((p^1 \& p^{1/2} \& p^0)^1 \supset (\overbrace{\neg\neg\ldots\neg}^{2n} (p^1 \vee p^{1/2} \vee p^0))))$, after them we derive by *modus ponens* the formula A_n. The lower bounds can be received by the same techniques as for 2-valued logic.

b) We can derive C_n as follow.

At first we derive the 3-tautology $\neg\neg(\neg\neg p \& \neg p \& p)$, then the 3-tautology

$\neg\neg(\neg\neg p \& \neg p \& p) \supset (\neg(\neg\neg p \& \neg p \& p) \supset ((\neg\neg p \& \neg p \& p) \supset (\overbrace{\neg\neg\ldots\neg}^{3n} (\neg\neg p \vee \neg p \vee p)))$

after them we derive by *modus ponens* the formula C_n . The lower bounds can be received by the same techniques as for 2-valued logic.

Remark. If as formula A_n (C_n) we take the new one, in which the number of repeated "negations" before the last subformula is 2^n , and in B_n (D_n) is 3^m for m=[n$\log_3 2$], then the sizes for such formulas will be the same by order as well, but the bounds for steps will be more contrast: O(1) and $\Omega(2^n)$ for strongly equal new 3-tautologies A_n (C_n) and B_n (D_n) accordingly.

4 Conclusion

We introduce the notion of strong equality of many-valued tautologies on the basis of determinative conjunct notion. The idea to revise the notion of equivalence between tautologies in such way that is takes into account an appropriate measure of their "complexity".

It is proved that in "weak" proof systems the strongly equal many-valued tautologies have the same proof complexities, while in the "strong" proof systems the measures of proof complexities for strongly equal tautologies can essentially differ from each other.

ACKNOWLEDGMENTS

This work was supported by the RA MES State Committee of Science, in the frames of the research project № 18T-1B034.

REFERENCES

(1) An. Chubaryan, Arm. Chubaryan, "A new conception of Equality of Tautologies", L&PS, Vol.V, No,1, 3-8,Triest, Italy, 2007.

(2) A.Chubaryan, G.Petrosyan, The relations between the proof complexities of strongly equal classical tautologies in Frege systems, *Российско-китайский научный журнал «Содружество» № I (1), 2016 / ФИЗ-МАТ НАУКИ,78-80.*

(3) An. Chubaryan, A.Mnatsakanyan, On the bounds of the main proof measures in some propositional roof systems, Scholar Journal of Phis. Math. And Stat.,2014, Vol.1,Issue-2, pp.111-117.

(4) Chubaryan Anahit, Khamisyan Artur, Arman Tshitoyan, On some systems for Łukasiewicz's many-valued logic and its properties, Fundamentalis Scientiam, Vol.8(8), Spain, 2017, 74-79.

(5) Anahit Chubaryan, Artur Khamisyan, Two types of universal proof systems for all variants of many-valued logics and some properties of them, Iran Journal of Computer Science, https://doi.org/10.1007/s42044-018-0015-4.

(6) J. Lukasiewicz, O Logice Trojwartosciowej, Ruch filoseficzny (Lwow), Vol. 5, 1920, 169-171.

(7) K.Godel, Zum intuitionistishen Aussagenkalkul, Akademie der Wissenshaften in Wien, Mathematische-naturwissenschaftliche Klasse, Auzeiger, Vol. 69, 1932, 65-66.

(8) A.A.Chubaryan: Comparison of proof sizes in systems and substitution systems of Frege, *Izvestiya NAN Armenii, Matematika, vol. 35, No. 5*, 2002, 71-84.

(9) Y. Filmus, M. Lauria, J. Nordstrom, N. Thapen, N. Ron-Zewi: Space Complexity in Polynomial Calculus, 2012 IEEE Conference on Computational Complexity (CCC), 2012, 334-344.

