



A High-Performance Kinetic Gas Molecule Optimization (KGMO) Based ADPLL for Induction Motors Control Applications

Rajat Kumar Dwibedi, E. Logashanmugam

Abstract: Nowadays control applications play a major role in different applications such as industrial automation, robotics, automobiles, power systems etc. The all-digital phase-locked loop (ADPLL) is a digital controller can be used commonly for all kind of control applications based on the application requirement. In the work, KGMO based ADPLL is proposed to perform the rotation control of an induction motor (KGMO-ADPLL-IMSC) under various load conditions by the varying current. The proposed ADPLL is tested in an induction motor (IM) to control the rotation of the motor under certain load conditions. Various parameters such as speed, torque, phase, amplitude and offset values are considered to generate the control signal. Root mean square error (RMSE) is considered as the objective function for the KGMO algorithm. Least error is considered for the best fitness function for the better performance. In this work, the current control technique is used to perform the rotation control of the induction motor. Various performance such as torque, speed and stability are measured under various load conditions. KGMO-ADPLL-IMSC design is implemented in an integrated MATLAB/SIMULINK environment under various inbuilt and custom libraries. Experiments show that the KGMO-ADPLL-IMSC strategy is altogether improved the performance in terms of all performance evaluation graphs.

Keywords: Induction motor, speed control, Kinetic Gas Molecule Optimization, All-digital phase-locked loop((ADPLL)), current control, Root mean square error, control systems, speed control.

I. INTRODUCTION

A Phase-locked loop (PLL) is an important block of all synchronization systems. Which is basically a closed-loop feedback system that is designed as frequency multipliers, tracking generators, demodulators and clock recuperation circuits. In PLL it changes with the frequency of an input signal to match with its value and improves the system performance. PLL has four types 1. Linear PLL 2. Digital phase-locked (DPLL) loop 3. All digital phase-locked loop (ADPLL) 4. Software PLL (SPLL). [1] The analog PLL has high power consumption and large area. That makes the PLL's are noise sensitive and some are expensive.

Revised Manuscript Received on March 30, 2020.

* Correspondence Author

Mr. Rajat Kumar Dwibedi, Department of Electronics and Communication Engineering, Jeppiar SRR Engineering College, Chennai, India. Email: rajatdwibedikumar@gmail.com

Dr.E.Logashanmugam*, Department of Electronics and Communication Engineering, Sathyabama Institute of Science and Technology, Chennai, India. Email: ogu999@yahoo.com

© The Authors. Published by Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP). This is an [open access](http://creativecommons.org/licenses/by-nc-nd/4.0/) article under the CC BY-NC-ND license (<http://creativecommons.org/licenses/by-nc-nd/4.0/>)

That makes ADPLL is among the most requesting research area in such manner as it has numerous advantages over analog PLL. It offers a smaller size by eliminating the requirement of big size capacitors in the filter, process, temperature and voltage (PVT) variations independence, and flexibility in programming and control. [2], [3] Beginning of ADPLL started in 1980. [4] The studies reveal that ADPLL outperforms classical PLLs and have more significance towards communication systems as well as consumer applications. With the fast improvement and wide utilization of large-scale programmable logic devices, ADPLL is very much improved as well. In addition to the points of interest, an ADPLL is both portable and reusable across different designs and semiconductor technologies. Phase-frequency detector (PFD) determines the differences in frequency and phase between the output signal and the input reference signal. Also, ADPLL got positive properties such as anti-interference performance, high reliability, low cost, convenience of integration, Reduced circuit area, and a simple external circuit, all-digital phase-locked loop moreover effectively takes care of issues, which is unsolvable for analog components, including temperature drift and the dispersion of the parameters of discrete components. An ADPLL is also a phase-locked loop that is executed only by digital blocks. The signals are binary (digital) and maybe a combination of parallel digital signals or single digital signal. It uses DLL (delay-locked loop) for its frequency generation instead of VCO and the control voltage is a bit biased. ADPLL achieves a fast frequency locking by using feed-forward compensation and also by reusing frequency divider [5].

In the purpose of device power switching, synchronization of the real-time phase is very important. The method of synchronization stage is basically implemented and it is depending upon the power system of a phase-locked loop (PLL) and zero-crossing detector. In this paper for this purpose, the entire control system with phase synchronization and PLL is used. The combined utilization of hardware and software realize the phase of synchronization. The implementation of software synchronization is carried out via software methods such as FPGA, CPU and DSP. So, the techniques assure strong control of tracking time, high precision and simple circuitry. The process of achieving better stability modules required an effective use of a control algorithm. Usually, there are several frequency search algorithms used in ADPLLs. Ultimately, in this paper, the experimental requirement is proposed and tested and simulated using MATLAB/Simulink.



II. LITERATURE SURVEY

Based on the different application of ADPLL with numerous works are presented and some of them are reviewed as follows, Bohan Wu, Weixin Gai, Te Han introduced a Novel Frequency Search Algorithm to achieve fast locking without phase tracking so as to accomplish fast locking in ADPLL with no phase tracking being required. In this proposal instead of phase tracking, according to phase and frequency error, the normalized tuning word (NTW) is calculated so that the output frequency reaches the desired frequency immediately. In order to avoid the risk of non-idealities such as DCO gain estimation error and TDC finite resolution, which greatly affects the accuracy of the calculation this study suggests that the output frequency is continuously measured and frequency error is averaged to minimize those impacts. [6] Deepika Ghai 1, Neelu Jain did a detailed Review of All-Digital Phase-Locked Loop (ADPLL). in which the parts of ADPLL, for example, phase detector, detector, VCO have been talked about in detail. The examination recommends that Various issues in Digital PLL like leakage, noise, parasitic element etc. can be removed with the help of All-Digital PLL. Also, they did a systematic comparison of Various parameters of ADPLL such as power consumption, jitter, input and output frequency etc. [7]

spur of PLL and Settling time are one of the most significant issues in wireless communication systems. Wei Xu and Wei Li introduced Algorithms Based on ADPLL for Fast-locking and Spur Free in order to reduce such issues. The proposed fast-locking algorithm in this study embraces an enhanced AFC (programmed frequency control) strategy with a dynamic bandwidth technique that doesn't have to adjust circuits contrasted with ordinary OTW (oscillator turning word) assessing technique. Likewise, they presented a Multi-bits LSB Dithering module that is proposed to further suppress partial spur brought about by SDM (sigma-delta modulator) periodic output. This claim that to solve nonlinearity resulted from varactor mismatch the DEM module is applied which will enhance the spur performance. The theory was confirmed with Simulation results, which shows that when AMS comparing to the ADPLL without fast-locking algorithms there is an upgrade in settling time is about 27%~72%. [8]

Wenlong Jiang, Armin Tavakol, Popong Effendrik, Marcel van de Gevel, Frank Verwaal, and R. Bogdan Staszewski developed a Design of ADPLL System for WiMAX Applications in 40-nm CMOS. The study is basically an introduction on ADPLL system which is proposed for WiMAX applications, they accomplished this through Via frequency arranging and design of multiple capacitor-banks in a digitally-controlled oscillator (DCO), the ADPLL covers dual bands of 2.3–2.7 GHz and 3.3–3.8 GHz, while achieving a fine frequency resolution of 25 Hz. Key circuit blocks like TDC and DCO are implemented in 40-nm CMOS technology with sufficient performance. The time-to-digital converter (TDC) achieves a resolution of better than 13 ps. the phase-rotation technique with a FREF dithering algorithm is adopted in this study in order to suppress fractional spurs. they did the whole system simulation on Verilog-AMS with digital circuits at the gate level I order to get the very detailed information on their proposition. [9]

Chao-Ching Hung and Shen-Iuan Liu proposed A 40-GHz Fast-Locked ADPLL Using a Modified Bang-Bang

Algorithm. In order to extend the range of the frequency tuning 40-GHz DCO and inductor are used. They did This ADPLL is a fabrication on a 90-nm complementary metal–oxide–semiconductor process. they used a bang-bang phase detector (BBPD) is adopted in an ADPLL because of its robustness and low power consumption. In order to shorten the locked time of an ADPLL, a well-refined bang-bang algorithm is introduced. This brief is organized in this proposal. [10]

Xin Chen, Jun Yang, and Long-Xing Shi proposed A Fast Locking ADPLL via Feed-Forward Compensation Technique. The proposed execution of All-digital phase-locked loop via two operation modes namely phase acquisition mode and frequency acquisition mode. The paper proposed a feed-forward compensation algorithm In frequency acquisition mode, for the ADPLL achieves a fast frequency locking. Also, it makes sure ADPLL accomplishes a finer phase via phase acquisition mode. [11]

III. KINETIC GAS MOLECULE OPTIMIZATION - ADPLL BASED INDUCTION MOTOR SPEED CONTROL

In this section, the description of ADPLL by means of control structures and motor Induction is implemented by using Kinetic Gas Molecule Optimization (KGMO). Which is one of the powerful optimization algorithms. The whole circuit design is basically implemented as two stages namely (i) Induction Motor and Measurement Unit (ii) And a proposed optimized algorithm is designed and utilized with suitable outcomes. Since there are two stages that are used, the initial one is the discrete and the second one in the form of the transfer function. Therefore, the planned architecture of IM is depicted in fig 1. The brief introduction procedure of the proposed system is described in the following section. Therefore, the analysis of the PLL and the ADPLL is described and specified with the transfer of their functions.

In order to keep the performance of the Induction Motor (IM) consistent have to maintain appropriate power supply. In this case, for example, we have to maintain a constant speed, say 1200 rpm. whether IM on load or not. It is obvious that the IM with load causes the decrease in its rotation, to prevent that have to increase the power supply with the drop-in rotation. To select this power level, I designed a control block with ADPLL on the KGMO algorithm. To adjust the power supply for the favorable rotation speed we considered certain parameters such as phase, speed of motor, amplitude, offset of DC wave. The rotational data from the shaft is collected using the speed sensor linked with it, and the data is then compared with the reference signal provided. The comparator will give the exact difference value comparing the observed parameters with the reference parameters. The loop filter will eliminate excess peaks in outputs from the comparator converting unit pass the signal to the KGMO algorithm unit in which the selection of the appropriate parameter. The data passed to the DCO which generates a signal as per the KGMO algorithm unit section referred. The signal from the output of the DCO counts on the next iteration going to occur in the comparator section. Using continuous iteration, the KGMO unit can choose the appropriate parameter to adjust which will give favorable power supply output cause the constant rotation on IM.



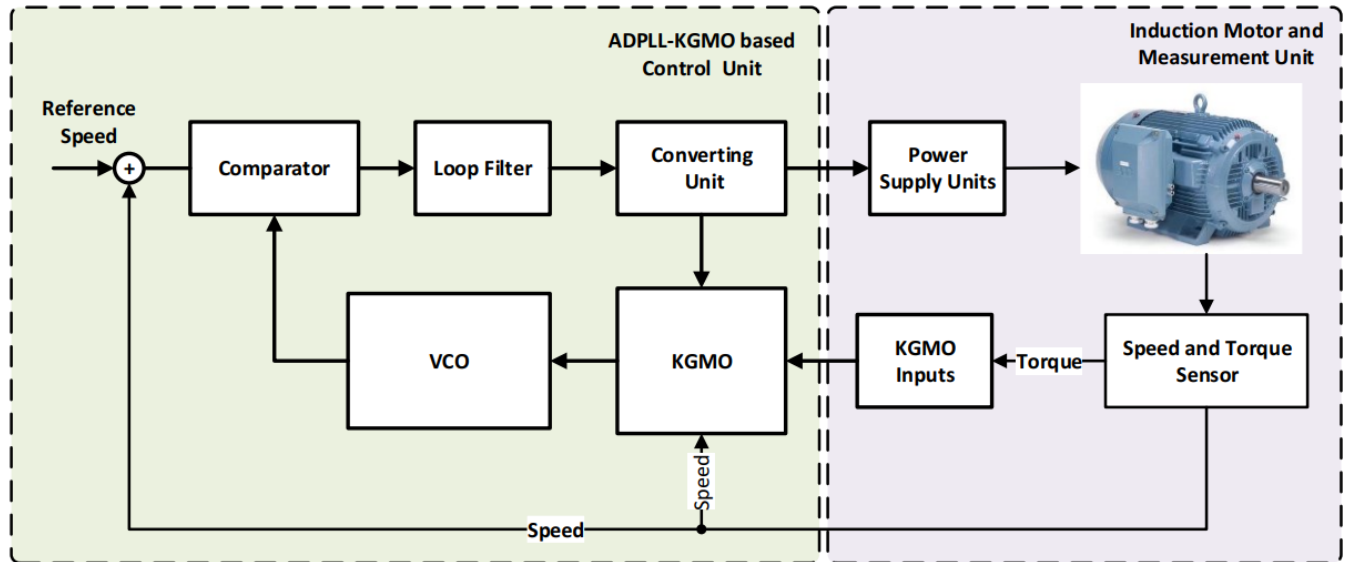


Fig.1. Design of KGMO based ADPLL

A. INDUCTION MOTOR AND MEASUREMENT UNIT

As the procedure of speed and torque data estimation of the induction motor requires the actual motor speed from direct measurement by using a shaft-mounted digital speed sensor. The speed and torque sensor mainly work based on the fact sequence of the digital pulse generated from the sensor because of the rotation of IM which is directly proportional to the angular motor speed $\omega(t)$. also, the induction motor system working under the excitation voltage U_c power. The rotational speed fixed on a favorable value of 1200 rpm. The output from the speed and torque sensor feed as the KGMO input. The power supply unit associated with the IM will adjust the power supply according to the control unit output.

B. KGMO BASED ADPLL

a) Digital Loop Filter

Basically, the counter K has two self-governing counters, which are frequently noted as “DOWN counter” and “UP-counter”. Nowadays, mutually (both) counters and every counting have been in an upward direction. The mutual counters are denoted as K and the mutual contents are in a range from 0 . . . K-1. K respectively. The mutual counter contents are managed by means of control input of K modulus also it is the power of 2 integers. While the signal of DN/UP is higher with the counter down becomes active also the frozen-up counter contents are collected. Another way, while low DN/UP with contents of up counter and frozen down counter contents are collected. To reset the mutual counters with zero content and that are grown into K-1. The up and down counter is obtained from the Carry and borrow of the most significant bit.

b) Digital Controlled Oscillator

The DCO's diversity is structured and is based on the loop filter output with frequency has been changed. Therefore, the design of the ADPLL required an Increment Decrement counter. This contains three inputs such as decrement, increment and input clock. While never pulses of carry or borrow are achieved. Thus, the counter ID delivers the pulse output going on all clock secondary ID. The INC input emerges at carry pulse after that the pulse ID becomes superior time with the solitary period ID clock also the pulse

of borrow emerges to the input DEC after that pulse ID becomes time delayed with a clock period of single ID.

c) Kinetic Gas Molecule Optimization (KGMO)

KGMO algorithm, the gas molecules are considered as agents within the search space, and K.E. is employed in calculating the performance. [19] The gas molecules move within the instrumentation till they converge within the part of the instrumentation that has the lowest temperature and K.E. it's best-known that gas molecules attract one another supported on weak electrical intermolecular Van Der Waal forces, wherever the electrical force is the result of positive and negative charges within the molecules. In the KGMO, each gas molecule (agent) has four specifications: position, K.E., velocity and mass. The K.E. of every gas molecule determines its velocity and position. within the algorithm, the gas molecules explore the entire search space to reach the point that has the lowest temperature.

Velocity

Next, consider a system that has N agents (gas molecules). The position of the i^{th} agent is defined by

$$X_i = (X_i^1 \dots X_i^d \dots X_i^n), \text{ for } (i = 1, 2, 3 \dots n) \quad (10)$$

where X_i^d represents the position of the i^{th} agent in the d^{th} dimension. The velocity of the i^{th} agent is presented by

$$V_i = (V_i^1 \dots V_i^d \dots V_i^n), \text{ for } (i = 1, 2, 3 \dots n) \quad (11)$$

Where V_i^d represents the velocity of the i^{th} agent in the d^{th} dimension. The movement of the gas molecules in the cylinder is based on the Boltzmann distribution, which means that its velocity is proportional to the exponential of the molecules' kinetic energy. This kinetic energy, in turn, is defined as

$$K_i^d(t) = 3/2 (N_b T_i^d(t), k_i = (K_i^1 \dots K_i^d \dots K_i^n), \text{for } (i = 1, 2, 3 \dots n)) \quad (12)$$

where N is the number of gas molecules, b is the Boltzmann constant, and $T_i^d(t)$ is the temperature of the i^{th} agent in the d^{th} dimension at time t. The velocity of the molecule is updated by

$$V_i^d(t+1) = T_i^d(t) w v_i^d(t) + c1randi(t) (g_{best}^d - X_i^d(t)) + c2randi(t) (P_{best}^d(t) - X_i^d(t)) \quad (13)$$

where T_i^d for the converging molecules reduces exponentially over time and is calculated as

$$T_i^d(t) = 0.95X(T_i^d(t-1)) \quad (13a)$$

The vector $pbest_i = (pbest_i^1, pbest_i^2, \dots, pbest_i^n)$ represents the best previous position of the i^{th} gas molecule, and $gbest_i = (gbest_i^1, gbest_i^2, \dots, gbest_i^n)$ is the best previous position among all of the molecules within the instrumentation. the velocity and therefore the of each particle is initialized by random vectors within the corresponding ranges. Here, [-vmin, vmax] is used as the limits of the gas molecules' velocity. If $|v_i| > v_{max}$, then $|v_i| = v_{max}$. w is the inertia weight that reflects the gas molecule's resistance to slow its movement. Additionally, randi(t) is a uniform random variable in the interval [0,1] at time t, which is used to provide a randomized characteristic to the search algorithm. C1 and C2 are two acceleration constants.

The mass m of each gas molecule is a random number within the range 0 m 6 1; once identified, it remains constant throughout the execution of the algorithm because the container is assumed to contain only one type of gas at any one time. The random number is used to simulate different types of gases in different executions of the algorithm. From the equations of motion in physics, the position of the molecule is defined by

$$X_{t+1}^i = \frac{1}{2} a_i^d(t+1)t^2 + V_i^d(t+1)t + X_i^d(t) \quad (14)$$

where a_i^d represents the acceleration of the i^{th} agent in the d^{th} dimension. From the acceleration equation, we obtain

$$a_i^d = \frac{dv_i^d}{dt} \quad \text{On the other hand, from Eq. (18) of the gas molecule laws, we have}$$

$$dk_i^d = \frac{1}{2} m (dv_i^d)^2 \Rightarrow dv_i^d = \sqrt{\frac{2(dK_i^d)}{m}} \quad (15)$$

Therefore, from Eqs. (21) and (22), the acceleration is defined

$$\text{as } a_i^d = \sqrt{\frac{2(dK_i^d)}{m}} \quad (16)$$

In the time interval Dt, Eq. (23) can be re-written as

$$a_i^d = \sqrt{\frac{2(\Delta K_i^d)}{m}} \quad (17)$$

Thus, in a unit time interval, the acceleration would be

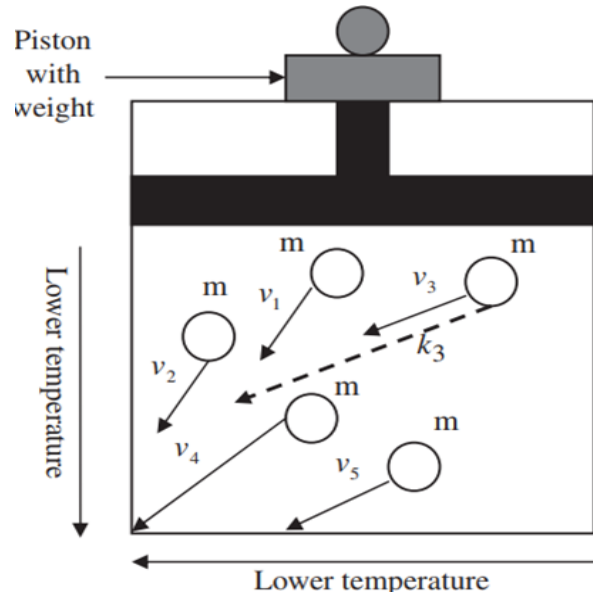


Fig. 2. Under constant pressure, the kinetic energy of gas molecules decreases by decreasing the velocity

$$a_i^d = \sqrt{\frac{2(dK_i^d)}{m}} \quad (18)$$

Then, from Eqs. (12) and (17), the position of the molecule is calculated by

$$X_{t+1}^i = \frac{1}{2} a_i^d(t+1)\Delta t^2 + V_i^d(t+1)\Delta t + X_i^d(t) \quad (19)$$

$$X_{t+1}^i = \frac{1}{2} \sqrt{\frac{2(\Delta K_i^d)}{m}} (t+1)\Delta t^2 + V_i^d(t+1)\Delta t + X_i^d(t) \quad (20)$$

Finally, considering that the molecule mass (m) is a random number in each execution of the algorithm but the same for all of the molecules in execution, for simplicity, the position is updated for the unit time interval by

$$X_{t+1}^i = \frac{1}{2} \sqrt{\frac{2(\Delta K_i^d)}{m}} (t+1)\Delta t^2 + V_i^d(t+1) + X_i^d(t) \quad (21)$$

The minimum fitness function is found by using

$$p_{best} = f(x_i), \text{if } f(x_i) < f(p_{best}_i) \quad (22)$$

$$g_{best} = f(x_i), \text{if } f(x_i) < f(g_{best}_i) \quad (23)$$

Each gas molecule attempts to modify its position X_i^d by using the distance between the current position $\llbracket p_{best}_i$ the distance between the current position and g_{best}_i . The general steps in the KGMO algorithm are summarized by the pseudocode in Algorithm.1

Algorithm.1

```

For each gas molecule {
Repeat initialize gas molecules until it satisfies all constraints
}
Do {
For each particle {
Calculate fitness value
If the calculated fitness value is better than the existing best
fitness value (pbest) OR no existing pbest {
Set current value as the new pbest
}
}
For each particle {
If particle's pbest is better than the global best fitness value
(gbest) so far OR no existing gbest {
Set current particle's pbest as new gbest
}
}
Calculate the kinetic energy of each gas molecule (Eq. (12))
Update particle velocity (Eq. (13)) and position (Eq. (19))
    
```

```

}
} while maximum iterations or minimum error criterion not
attained
    
```

IV. RESULT AND DISCUSSION

The proposed control system is implemented using the MATLAB2018a, fig.4 shows the overall Simulink model of the system. In Fig.3 the block labeled PLL is the proposed PLL model the whole model is converted into the subsystem and used in this model. In the above fig VSI is the voltage source inverter it will provide the supply for the motor here the firing angle of the inverter is controlled as per the speed change of the motor.

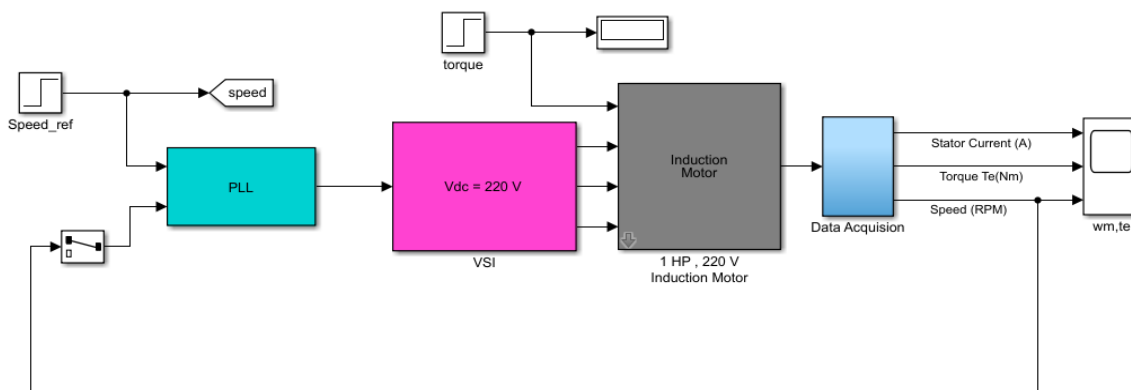


Fig.3. Simulink Model of the proposed System

Here we used a 3phase induction motor for simulation the parameter of the motor is shown in Table.I. Here the load is simulated using a step input that is given as the input to the machine the load is varied over time.abstract. Abbreviations such as IEEE, SI, MKS, CGS, sc, dc,

TABLE I.

Parameters of the Induction Motor

Parameters	Value
Phase	3
Voltage	220v
Power	1HP
Speed	1500 RPM
Frequency	50Hz

The response of the model is shown in Fig.4 and Fig.5. Fig.4 shows the speed curve of the proposed model. Here initially the speed of the machine is increased over the reference speed (1440) when the speed exceeds the reference speed immediately the controller will produce the control over the speed and brings it into the normal speed. The load is increased at 2sec at that time the speed is decreased below the reference speed at that time the controller will produce the control signal and brings back the speed into the normal level. Fig.5. Shows the progression of the stator current. When there is no load that is the input torque is zero the current exceeds the limit at that

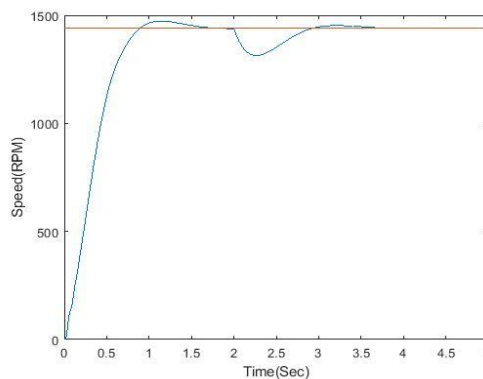


Fig.4. Speed curve of the proposed model

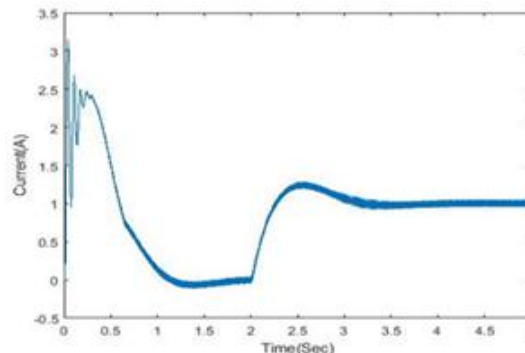


Fig.5. Stator Current curve of the proposed model

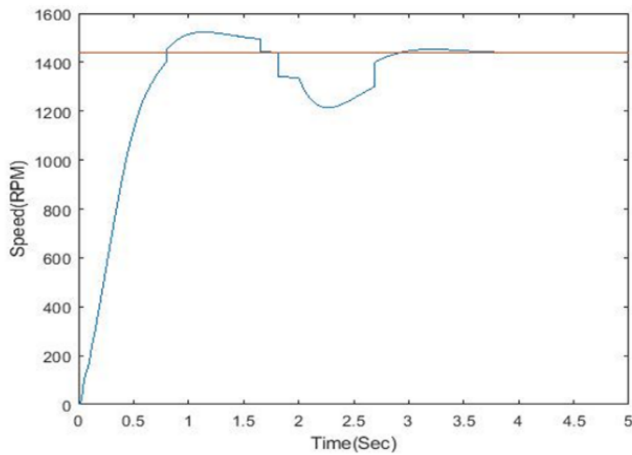


Fig.6. Speed curve of the Existing Model

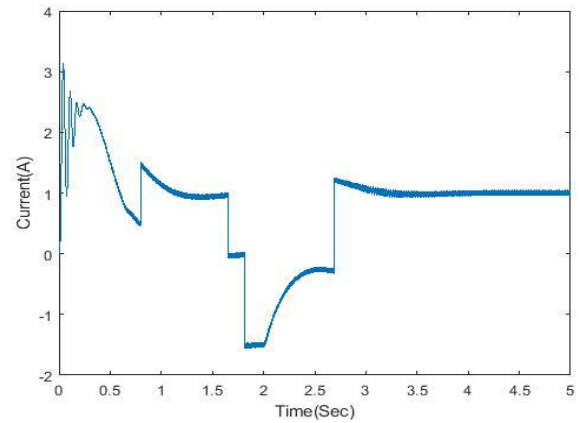


Fig.7. Stator current curve of the Existing Model

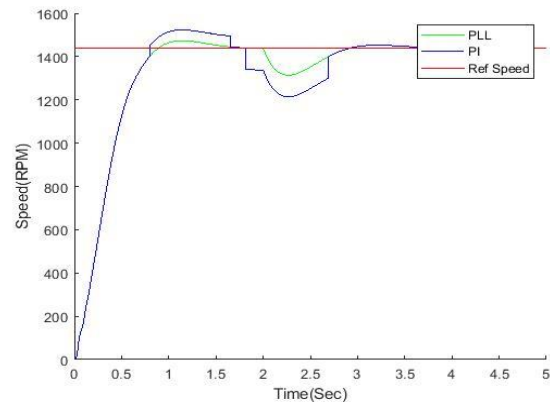
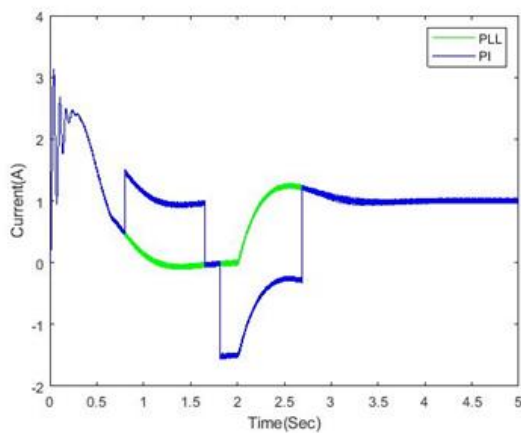


Fig.8. Combined Performance

V. CONCLUSION

The paper proposed, A High-Performance Kinetic Gas Molecule Optimization (KGMO) Based ADPLL for Induction Motors Control Applications. The ADPLL used for realizing the proposed design contain simply digitalized fundamentals. Hence, ADPLL includes flip-flops and combinational logic as fundamental building blocks. Because of this feature of an ADPLL, which is moveable and reusable transversely various designs logic transversely to SoC also dissimilar nodes technology. At the same time as the KGMO application in ADPLL with the self-motivated feature of the ADPLL is enhanced also it assists to make available to the optimal consequences. since the KGMO algorithm is one of the finest algorithms, the combined use of it on the ADPLL gives many improved results than the other existing method. The usage of the algorithm helps for effective control over IM performance by controlling the power supply in a favorable way. the suggested method of putting into practice ADPLL in the platform of MATLAB/SIMULINK also the Induction Motor is examined properly. The speed control of motor synchronization with the proposed ADPLL is employed on it in addition to noises are reduced. The simulation of the performance of the proposed method gives a better result.

REFERENCES

1. Kusum Lata and Manoj Kumar, ALL Digital Phase-Locked Loop (ADPLL): A Survey ,International Journal of Future Computer and Communication, Vol. 2, No. 6, December 2013
2. Imran Ali1, Seong-Jin Oh1, Hamed Abbasizadeh1, Behnam Samadpoor Rikan1,Muhammad Riaz Ur Rehman1, Dong-Soo Lee1, Kang-Yoon Lee, An Ultra Low Power Fully Synthesizable Digital Phase and Frequency Detector for ADPLL Applications in 55 nm CMOS Technology
3. D. Lee et al., "Low power FSK transceiver using ADPLL with direct modulation and integrated SPDT for BLE application", IEEE Asian Solid-State Circuits Conference (A-SSCC), pp. 113-116, Nov. 201
4. Tin-Yam Yau, Tri Caohuu and Jeonghee Kim ,An Efficient All-Digital Phase-Locked Loop with Input Fault Detection, R. E. Best, Phase Locked Loops Design Simulation and Applications, McGraw-Hill Professional, ch. 6, pp. 205-246, 5th Edition, 2003.,An Efficient All-Digital Phase-Locked Loop with Input Fault Detection
5. J.Jeslin Jijo, R.Dinesh, A Survey about ADPLL and IoT Applications IJRECE VOL. 6 ISSUE 4 (OCTOBER- DECEMBER 2018)
6. Bohan Wu, Weixin Gai, Te Han, A Novel Frequency Search Algorithm to Achieve Fast Locking Without Phase Tracking in ADPLL
7. Deepika Ghai 1, Neelu Jain ,All-Digital Phase Locked Loop (ADPLL) -A Review International Journal of Electronics and Computer Science Engineering
8. Wei Xu , Wei Li ,Algorithms Based on All-Digital Phase-Locked Loop for Fast-locking and Spur Free (2015 IEEE)
9. Wenlong Jiang, Armin Tavakol, Popong Effendrik, Marcel van de Gevel, Frank Verwaal, and R. Bogdan Staszewski ,Design of ADPLL System for WiMAX Applications in 40-nm CMOS (2012 IEEE)

10. Chao-Ching Hung and Shen-Iuan Liu , A 40-GHz Fast-Locked All-Digital Phase-Locked Loop Using a Modified Bang-Bang Algorithm IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, VOL. 58, NO. 6, JUNE 2
11. Xin Chen, Jun Yang, and Long-Xing Shi, A Fast Locking All-Digital Phase-Locked Loop via Feed-Forward Compensation Technique
12. Moein, Sara, and Rajasvaran Logeswaran. "KGMO: A swarm optimization algorithm based on the kinetic energy of gas molecules." *Information Sciences* 275 (2014): 127-144.
13. J. Jones. (1991, May 10). *Networks* (2nd ed.)
14. (Journal Online Sources style) K. Author. (year, month). Title. *Journal* [Type of medium]. Volume(issue), paging if given. Available: [http://www.\(URL\)](http://www.(URL))

AUTHORS PROFILE



Mr. Rajat Kumar, Dwibedi obtained his Bachelor's Degree in Electronics and Communication Engineering from Institution of Engineers (India), Kolkata in the year 2001 and obtained Master's Degree in VLSI Design in the year 2009 from Sathyabama University, Chennai, India. He is currently pursuing Ph.D., in Sathyabama Institute of Science and Technology, Chennai, India and working as an Assistant Professor Gr.-II in the Faculty of Electronics and Communication Engineering at Aarupadai Veedu Institute of Technology, Vinayaka Mission's Research Foundation, Chennai, India. His area of specialization includes Signal Processing, VLSI Design and Digital Electronics. His current research interest is VLSI Design and Signal processing. He is a member of ISTE and IE.



Dr. E. Logashanmugam, was born in Madurai, Tamilnadu, India in 1969. He received his Bachelor Degree in Electronics and Communication Engineering from Madurai Kamaraj University in the year 1991, Master Degree in Electronics Engineering from Anna University, Chennai in the year 2001 and Ph.D. from Sathyabama University, Chennai, in the year 2009. His field of interest includes VLSI Design, Digital Image Processing, Cryptography Nano Technology and Wireless Communications. He has more than 111 Research Publications in International Journal and Conferences to his Credit. He is serving as an editorial member and reviewer of several international reputed journals. He has 26 years of experience in teaching and presently working as Professor, School of Electrical and Electronics Engineering, Sathyabama University, Chennai. He is a member of ISTE and IEEE.