SUST Journal of Engineering and Computer Sciences (JECS), Vol. 24, No. 1, 2026

The Stability Analysis of a Modified Cascade Control Strategy for Khartoum Refinery Delayed Coking Unit's Drums

Lina Amer Alsharif Ahmed * and Tomadir Awad Ibraheem Hamed

Department of Chemical Engineering, Sudan University of Science and Technology, Khartoum, Sudan
eng_lina_amer@hotmail.com

Abstract-- **The delayed coking unit is a process that converts heavy oil residues into more valuable products. The pressure control of the coking drums is important for process safety, yield, and stability. However, it has received limited study compared to temperature control. Cascade control is suitable for many industrial processes, especially when enhanced by combining proportional (P), integral (I), and derivative (D) actions in the control loops. This study focused on modifying a previous cascade control strategy developed to control the pressure inside the coking drums at Khartoum Refinery by controlling the heating fuel flow rate. The modification included replacing the Proportional (P) controller in the secondary loop with Proportional-Integral (PI) and Proportional-Integral-Derivative (PID) controllers to evaluate their impact on the overall control system stability. MATLAB simulations were used to calculate system transfer functions and test stability through direct substitution, root locus, Bode plots, and step response analysis. The results showed instability with both controllers due to reduced phase margins and increased high-frequency noise. Several control strategies were recommended to enhance the system performance, such as Lead/Lag compensators, Model Predictive Control (MPC), and State Feedback control.**

Index Terms—**Cascade Control, Delayed Coking Unit, MATLAB Simulation, PI and PID Controllers, Stability Analysis, Step Response**

الملخص— **تُعد وحدة التفحيم المؤخر عملية يتم فيها تحويل البقايا الثقيلة إلى منتجات أكثر قيمة. ويُعد التحكم في ضغط براميل التفحيم مهمًا لسلامة العملية، والإنتاجية، والاستقرار. ومع ذلك، فقد حظي بدراسة محدودة مقارنةً بالتحكم في درجة الحرارة. يُعد التحكم التتابعي مناسبًا للعديد من العمليات الصناعية، خاصةً عند تحسينه من خلال دمج الأفعال التناسبية (P)، والتكاملية (I)، والتفاضلية (D) في حلقات التحكم. ركزت هذه الدراسة على تعديل استراتيجية تحكم تتابعي سابقة طُورت للتحكم في الضغط داخل براميل التفحيم في مصفاة الخرطوم من خلال التحكم في معدل تدفق وقود التسخين. وقد شمل التعديل استبدال المتحكم التناسبي (P) في الحلقة الثانوية بمتحكمي التناسبي-التكاملي (PI) والتناسبي-التكاملي-التفاضلي (PID) لتقييم تأثيرهما على الاستقرار الكلي لنظام التحكم. استُخدمت محاكاة MATLAB لحساب دوال انتقال النظام واختبار الاستقرار باستخدام التعويض المباشر، ومخطط موضع الجذور، ومخططات بود، وتحليل استجابة الخطوة. وأظهرت النتائج عدم الاستقرار مع كلا المتحكمين بسبب انخفاض هوامش الطور وزيادة الضوضاء عالية التردد. وأوصت الدراسة بعدة استراتيجيات تحكم لتحسين أداء النظام، مثل معوضات التقديم/التأخير (Lead/Lag)، والتحكم التنبؤي بالنموذج (MPC)، والتحكم بتغذية الحالة الراجعة (State Feedback).**

I. INTRODUCTION

DELAYED coking is a type of thermal cracking in which the heat required to complete the coking reactions is supplied by a furnace, while coking itself takes place in drums operating continuously on a 24- h filling and 24- h emptying cycles. Petroleum coke's importance lies in its high energy content and its ability to replace more expensive fuels [1, 2].

Additionally, petroleum coke, including calcined coke, has become a valuable resource for maximizing efficiency and utilization of crude oil residues. To maintain product quality and ensure safe operation, effective control systems must be applied. Cascade control is one of the advanced control systems that are widely used in industrial processes to reduce the effect of a load disturbance. It employs two controllers, a primary controller and a secondary controller. The primary controller provides the set point for the secondary controller [3, 4]. Recent research has explored advanced control techniques, including Adaptive State Feedback Predictive Control (SFPC), Expert Systems, and PID control integrations to enhance temperature control and process stability in delayed coking units [5–7]. However, few studies have addressed the control of drum pressure specifically, a critical factor in product yield and unit stability. Hamed et al. [8] proposed a cascade control scheme using a P controller in the secondary loop, showing moderate success in pressure regulation.

This research aims to explore the recommended upgrades to the cascade control approach designed by Hamed et al. [8] to regulate the coking drum pressure within the delayed coking unit of the Khartoum refinery by modulating the flow of heating fuel, as well as evaluating the stability and efficacy of this revised system. Integral (I) and Integral-Derivative (ID) controllers were added to the (P) controller in the secondary control loop to study their effect on controlling the heating fluid flow rate and thus their effect on pressure control within the coking drums in the primary loop. The study employed MATLAB software for calculations and simulations of stability analysis using various methods, including direct substitution, root-locus, Bode plots, and step response.

II. METHODOLOGY

This study used a model from previous research because real plant data was not available. The work focused on stability analysis only and did not test the effect of noise or nonlinearity. Also, only PI and PID controllers were studied. The secondary loop was first changed from a P-only controller to PI and then

PID controllers individually, followed by calculating the primary loop transfer functions, stability, and step response simulations through several methods using MATLAB software; a modern programming language environment that has sophisticated data structures, contains built-in editing and debugging tools, and supports object-oriented programming. These factors make MATLAB an excellent tool for teaching and research [9]. The physical and block diagrams depicting the cascade control, the primary and the secondary loops were illustrated in Fig. 1 and Fig. 2, respectively.

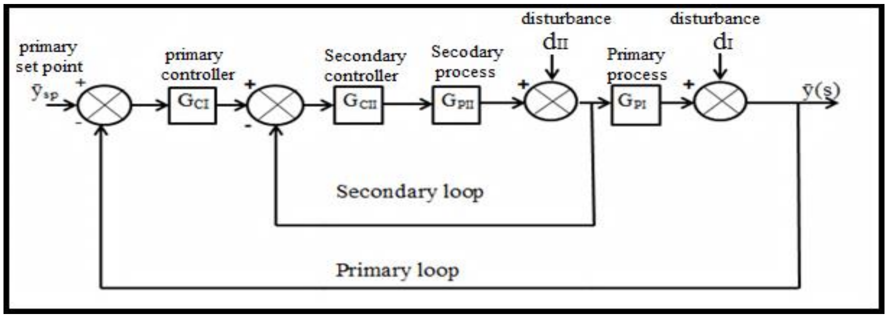

Fig. 1. Cascade control loop block diagram.

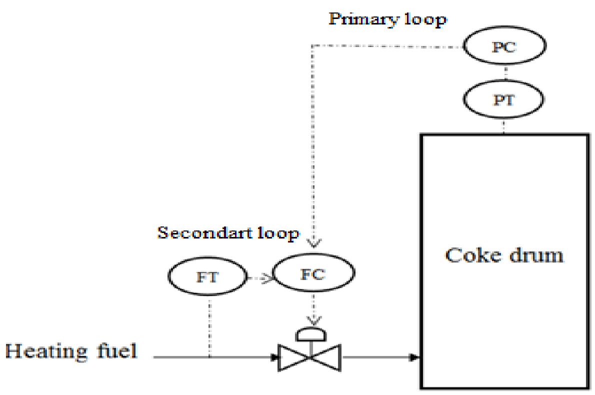

Fig. 2. Cascade control physical diagram.

A. *Transfer Function Development*

From [8], the primary and secondary process transfer functions (G_{P1} and G_{P2}) of this cascade control system were derived from actual drum data and processed using MATLAB. These transfer functions were summarized in Tables 1 and 2, and they were adopted as the initial data of this study:

TABLE I

Primary and Secondary Process Transfer Functions G_{P1}, G_{P2}

Control loop	Process Transfer Function (G_P)
Primary	$G_{P1}=\frac{-0.08803S+0.049}{S^2+1.271S+0.1447}$
Secondary	$G_{P2}=\frac{-0.0005854S^2-0.01121S+0.0003176}{S^3+1.117S^2+10.18S+0.0005226}$

TABLE II

Secondary controller parameters

Parameters	Secondary PI	Secondary PID
K_{C2}	395.5	522
τ_{I2}(s)	8	4.81
τ_{D2}(s)	-	1.2

For each controller, both open-loop and closed-loop transfer functions were developed. The numerator and denominator of OLTF$_s$ were calculated by multiplying the numerator matrix of G_P by the numerator matrix of G_C and the denominator matrix of G_P by the denominator matrix of G_C using the order conv ([ ],[ ]) respectively. Following this, the full equation of OLTF was calculated using the order tf (num, den). Finally, the closed-loop transfer functions TF$_s$ were calculated using MATLAB order feedback (OLTF).

B. *Stability Analysis Techniques*

The primary controller was set to Proportional p- only so as to obtain the values of the ultimate gain K_u and the frequency ω from the characteristic equation using the direct substitution method. Then, the primary controller gain K_C was set equal to unity in order to obtain the root locus and Bode plots, and therefore the values K_u and ω.

1) Direct Substitution method:

In this method, (K_u) and (iω) were substituted in the characteristic equation instead of (K_C) and (s). Two new equations were then obtained for each control system category: The real part equation and the imaginary part equation. MATLAB was used to solve these equations using the 'solve' function. The two positive values of ω and K_u were chosen to be the right values.

2) Root-locus and Bode plots:

The root-locus plots and Bode plots for the primary loop OLTF with PI secondary controller and then PID, were obtained using the order rlocus (OLTF) and bode (OLTF). In root-locus plots, the ultimate gain (K_u) and the crossover frequency (ω) were determined by clicking on the intersection point in the plots shown in Fig. 3 and Fig. 5, where the curve intersects the Y-axis.

In Bode plots, the frequency (ω) was obtained from the phase angle curve illustrated in Fig. 4 (A) and Fig. 6 (A) at a phase angle equal to -180°. To acquire the value of the Amplification Ratio 'AR' Which is equal to the inverse of the controller's gain, the magnitude unit in the upper curve of the figures was altered by right-clicking on the plot, selecting the property option in the newly appeared window, then choosing the 'units' option to switch from (dB) to absolute (abs) and finally, the values of the Amplification Ratio 'AR' were extracted from the new curves depicted in Fig. 4. (B) and Fig. 6. (B) and ultimate gain K_u was calculated.

C. Step response

A step response analysis was conducted for all combinations of controllers to evaluate their dynamic performance parameters, including rise time, overshoot, and settling time. The responses from the different configurations showed similar stability characteristics; therefore, the primary loops with a P controller combined with a secondary PI controller, and with a secondary PID controller, were selected as the representative case studies.

III. Results and Discussion

1) Calculated Open and Closed Loop Transfer Functions for the Secondary loop:

a) PI- Secondary controller:

$OLTF_{2(PI)}=$

$$395.5\left(s+\frac{1}{8}\right)\frac{-0.0006s^2-0.01120s+0.0003}{s^4+1.1170s^3+10.1800s^2+0.0005s} \quad (1)$$

$$OLTF_{2(PI)}=\frac{-0.2310s^3-4.4624s^2-0.4286*s+0.0157}{s^3+1.1170s^2+10.1800s+0.0005} \quad (2)$$

$$TF_{2(PI)}=\frac{-0.2310s^3-4.4624s^2-0.4286s+0.0157}{s^4+0.8860s^3+5.7176s^2-0.4281s+0.0157} \quad (3)$$

b) PID- Secondary controller:

$OLTF_{2(PID)}$ =

$$522\left(1.2s^2+s+\frac{1}{4.8}\right)\frac{-0.0006s^2-0.0112s+0.0003}{s^4+1.1170s^3+10.1800^2+0.0005}$$

(4)

$OLTF_{2(PID)}=$

$$\frac{-0.3667\,s^4-7.328\,s^3-5.716s^2-1.0530s+0.0345}{s^4+1.1170s^3+10.1800s^2+0.0005s} \quad (5)$$

$$TF_{2(PI)}=\frac{-0.3667\,s^4-7.3280s^3-5.7160s^2-1.0530s+0.0345}{0.6333s^4-6.2110s^3+4.4640s^2-1.0530s+0.0345} \quad (6)$$

2) Calculated Open and Closed Loop Transfer Functions for the Primary loop:

a) Primary loop with Secondary PI controller:

$$\mathbf{OLTF_{1(P/PI)}}=TF_{2(PI)}\ K_{C1(P/PI)}\ \frac{-0.0880s+0.0490}{s^2+1.2710s+0.1447}$$

$$=\frac{K_{C1(P/PI)}(0.0204s^4+0.3815s^3-0.1810s^2-0.2240s+0.0008)}{s^6+2.1560s^5+6.9870s^4+6.9670s^3+0.3000s^2-0.0420s+0.0023} \quad (7)$$

$$\mathbf{TF_{1(P/PI)}}=\frac{K_{C1(P/PI)}(0.0204s^4+0.3810s^3-0.1810s^2-0.2240s+0.0008)}{s^6+2.1560s^5+6.9870s^4+6.9670s^3+0.3000s^2-0.0420s+0.0023+K_{C1(P/PI)}(0.0204s^4+0.3810s^3-0.1810s^2-0.2240s+0.0008)} \quad (8)$$

b) Primary loop with Secondary PID controller:

$$\mathbf{OLTF_{1(P/PID)}}=TF_{2(PI)}\ K_{C1(P/PID)}\ \frac{-0.0880s+0.0490}{s^2+1.2710s+0.1447}$$

$$=\frac{K_{C1(P/PID)}(0.0323s^5+0.6271s^4+0.1442\,s^3-0.1874s^2-0.0546s+0.0017)}{0.6333s^6-5.4060s^5-3.3390s^4+3.7220s^3-0.6579\,s^2-0.1085s+0.0050} \quad (9)$$

$\mathbf{TF_{1(P/PID)}}$

$$=\frac{K_{C1(P/PID)}(0.0323s^5+0.6271s^4+0.1442\,s^3-0.1874s^2-0.0546s+0.0017)}{0.6333s^6-5.4060s^5-3.3390s^4+3.7220s^3-0.6579s^2-0.1085s+0.0050+K_{C1(P/PID)}(0.0323s^5+0.6271s^4+0.1442\,s^3-0.1874s^2-0.0546s+0.0017)} \quad (10)$$

B. Primary controller stability results with PI secondary controller

1) Direct substitution

The denominator of (8) represents the characteristic equation, which is equal to $1 + OLTF_{1(P/PI)} = 0$

$\mathbf{1+OLTF_{1(P/PI)}}= s^6 + 2.1560\ s^5 + (6.9870 + 0.0204\ K_C)\ s^4 + (6.9670 + 0.3810\ K_C)\ s^3 + (0.3000 - 0.1810\ K_C)\ s^2 - (0.0420 + 0.2240\ K_C)\ s + (0.0023 + 0.0008\ K_C) = 0$

Substitute $s = i\omega$, $K_C = K_u$

Real part = $(\omega)^6 + (6.9870+0.0204\ K_u)\ (\omega)^4-(0.3000 - 0.1810\ K_u)\ (\omega)^2+(0.0023+0.0008\ K_u) = 0$

Imaginary part = $2.1560\ i\omega^5- (6.9670 + 0.3810\ K_u)\ i\omega^3-(0.042+0.224\ K_u)\ i\omega = 0$

Solve real and imaginary parts of the equations using MATLAB:

$K_{u1} = 27.40$, $\omega_1 = 2.85$ rad/sec.

2) Root-locus plots and Bode plots

Equation (7) was inserted in MATLAB after setting K_C =1

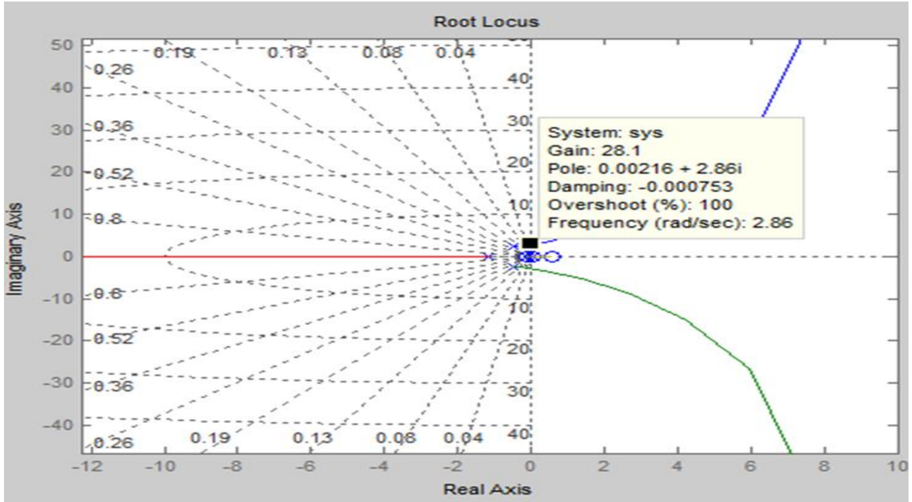

Fig. 3. Root locus plot for primary loop with PI in the secondary loop.

From Fig. 3: $K_{u1} = 28.1$, $\omega_1 = 2.86$ rad/sec.

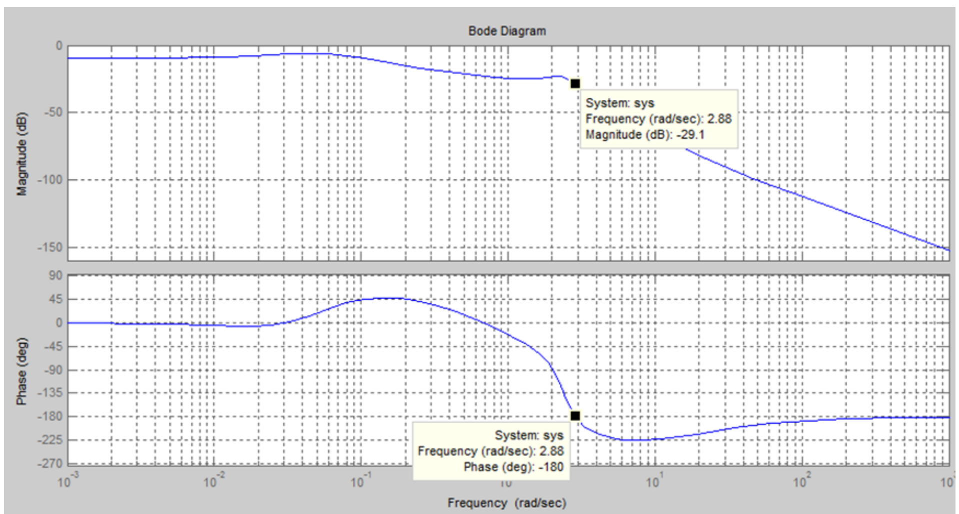

Fig. 4. A. Bode plot for primary loop using (dB) magnitude in Y- axis and secondary PI controller.

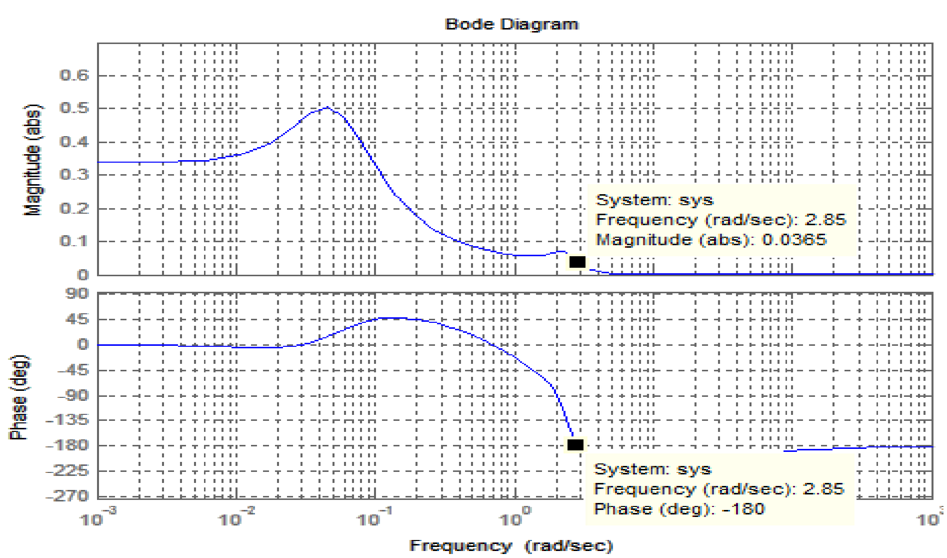

Fig. 4. B. Bode plot for primary loop using (abs) magnitude in Y- axis and a secondary PI controller.

From Fig. 4. A and B: $K_{u1} = 27.4$, $\omega_1 = 2.85$ rad/sec.

C. Primary controller stability results with PID secondary controller

1) Direct substitution:

The denominator of (10) represents the characteristic equation, which is equal to $1 + OLTF_{1(P/PID)} = 0$

$\mathbf{1+ OLTF_{1(P/PID)}} = 0.6333\ s^6 - (5.4060 - 0.0323\ K_u)\ s^5 - (3.339 - 0.6271\ K_u)\ s^4 + (3.722 + 0.1442\ K_u)\ s^3 - (0.6579 + 0.1874\ K_u)\ s^2 - (0.1085 + 0.05465\ K_u)\ s + (0.005 + 0.0017\ K_u) = 0$

Substitute $s = i\omega$, $K_C = K_u$

Real part= $-0.6333\ \omega^6 - (3.339 - 0.6271\ K_u)\ \omega^4 + (0.6579 + 0.1874\ K_u)\ \omega^2 + (0.005 + 0.001692\ K_u) = 0$

Imaginary part= $-(5.406 - 0.03228\ K_u)\ i\omega^5 - (3.722 + 0.1442\ K_u)\ i\omega^3 - (0.1085 + 0.05465\ K_u)\ i\omega = 0$MATLAB format:

```
>> real = -0.6333*w^6-(3.339 - 0.6271*k) *w^4 + (0.6579 + 0.1874*k) *w^2 +(0.005+0.001692*k);
>> Imaginary = -(5.406 - 0.03228*k) *w^4 - (3.722 + 0.1442*k) *w^2 - (0.1085 + 0.05465*k);
>> [k w] = solve (real, Imaginary);
>> vpa ([k w],5)
```

$K_{u2} = 172.82$, $\omega_2 = 12.891$ rad/sec.

2) Root-locus plots and Bode plots

Equation (10) was inserted in MATLAB after setting $K_C = 1$, and the format below was followed:

```
>> num1P_PID= conv (conv (522*[1.2 1 1/4.8], [-0.0005854 -0.01121 0.0003176]), [-0.08803 0.049]);
>> den1P_PID= conv ([0.6333 -6.211 4.464 -1.053 0.03454], [1 1.271 0.1447]);
>> OLTF1_PID=tf (num1P_PID, den1P_PID);
>> rlocus (OLTF1_P)
>> bode (OLTF1_P)
```

From Fig. 5: $K_{u2} = 170$, $\omega_2 = 12.8$ rad/sec.

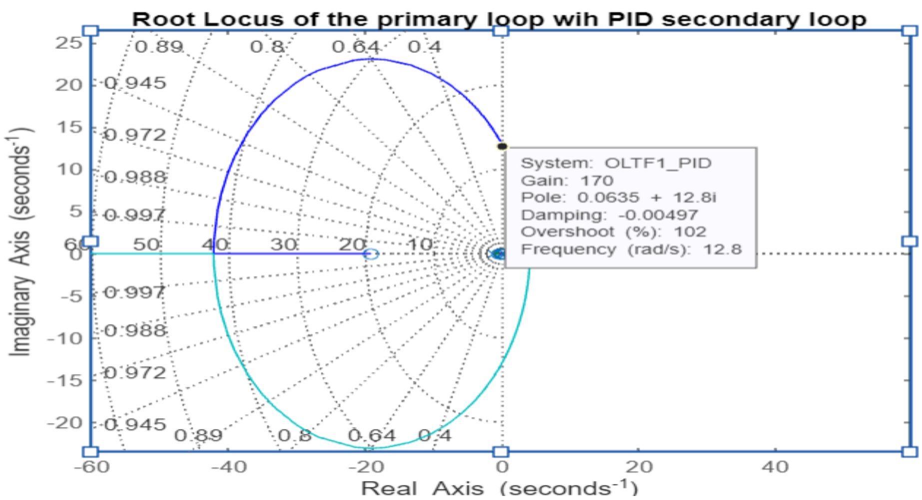

Fig. 5. Root locus plot for primary loop with PID in the secondary loop.

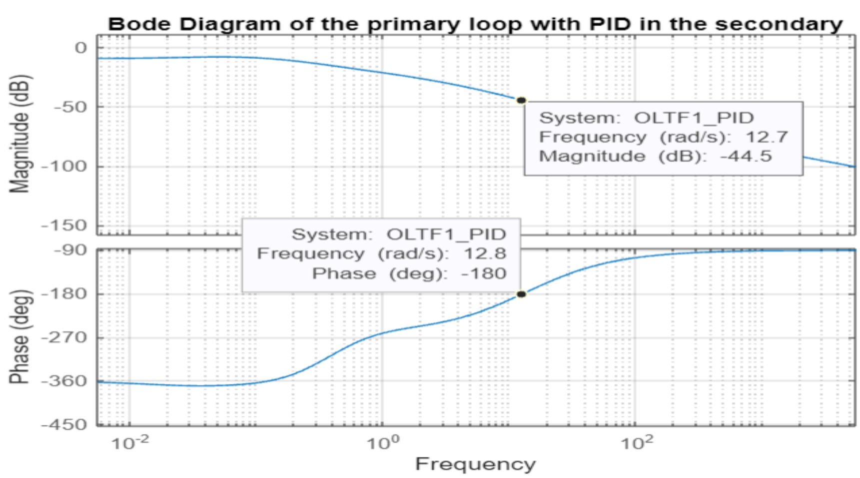

Fig. 6. A. Bode plot for primary loop using (dB) magnitude in Y- axis and secondary PID controller

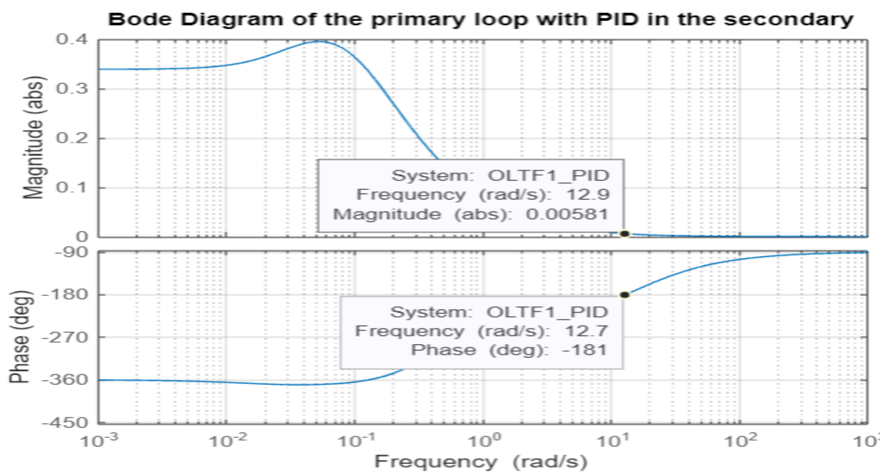

Fig. 6. B. Bode plot for primary loop using (abs) magnitude in Y- axis and secondary PID controller

From Fig 6. A and B: K_{u2} = 1/0.000581= 172.11.
ω_2 = 12.7 rad/sec.

D. Comparison of Ultimate Gain and Frequency

The results of calculated ultimate gains (K_u) and frequencies (ω) using the three methods: direct substitution, root locus, and Bode plots are summarized in Table III.

TABLE III

Calculated ultimate gains (K_u) and frequency (ω) for primary controllers under PI and PID secondary control

Method	**PI secondary**		**PID secondary**	
	K_u	**ω (rad/sec)**	**K_u**	**ω (rad/sec)**
Direct substitution	27.4	2.85	172.82	12.89
Root-locus plot	28.1	2.86	170	12.8
Bode plot	27.4	2.85	172.11	12.7

E. Step Response Results

The step response of the primary P controller combined with the secondary PI controller was generated using the MATLAB code provided below:

```
>> k=13.815;
>> num1P_PI=conv ([-0.2315 -4.4625 -0.4286 0.0157], (k*[-0.08803 0.049]));
>> den1P_PI= conv ([1 0.8855 5.718 -0.4281 0.0157], [1 1.271 0.1447]);
>> OLTF1_PI= tf (num1P_PI, den1P_PI);
>> TF1P_PI=feedback (OLTF1_PI,1);
>> step (TF1P_PI)
```

The step response of the primary P controller combined with the secondary PID controller was generated using the MATLAB code provided below:

```
>>k=85.82;
>> num1P_PID=conv([-0.3667 -7.328 -5.716 -1.053 0.03454], k*[-0.08803 0.049]);
>> den1P_PID= conv ([0.6333 -6.211 4.464 -1.053 0.03454],[1 1.271 0.1447]);
>> OLTF1P_PID= tf (num1P_PID, den1P_PID);
>> TF1P_PID=feedback(OLTF1P_PID,1);
>> step(TF1P_PID)
```

(A) (B)

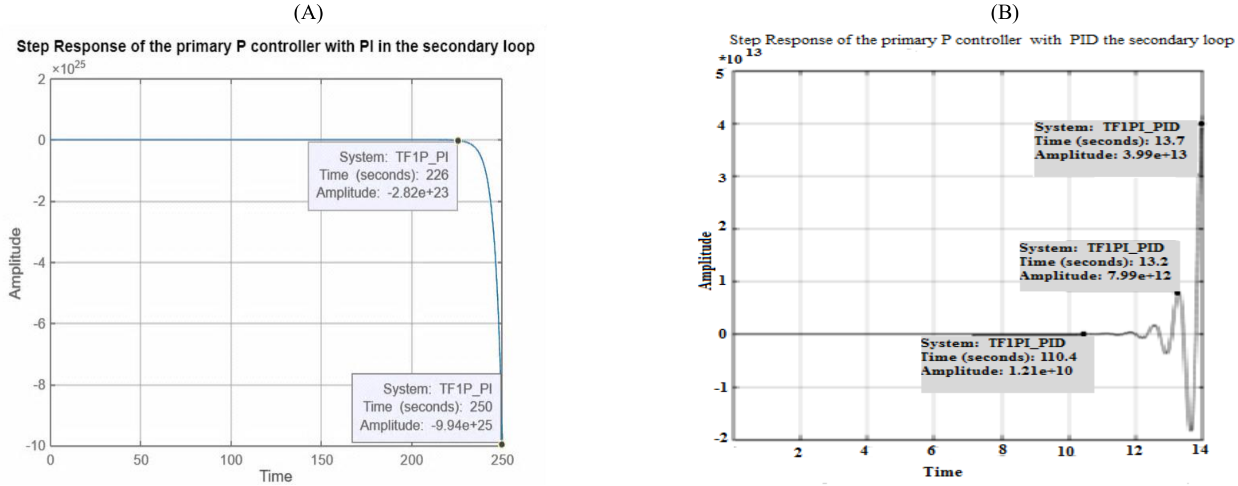

Fig. 7. Comparison of step responses for the primary P controller with secondary (A) PI controller and (B) PID controller.

From Fig. 7(A), when the secondary loop used a PI controller, the output stayed near zero until about 226 s, after which it began to diverge. No rise time, oscillations, or steady-state value could be measured.

From Fig. 7(B), with a PID controller in the secondary loop, oscillations started at around 11 s. Their amplitude increased over time, leading to unlimited overshoot and moving further from the desired value. The response showed sustained oscillations with a frequency of 1.43 Hz and amplitude of about $4.1 * 10^{12}$, with no measurable rise time or settling time. TABLE IV summarizes the above results.

TABLE IV

Measured dynamic response parameters for the primary P-controller with PI and PID controllers applied in the secondary loop

Controller metric	P- PI	P- PID	Unit
Rise time (T_r)	N/A	N/A	s
Settling time (T_s)	N/A	N/A	S
Overshoot	N/A	Unlimited	%
Growth rate (σ)	N/A	Positive	1/s
Amplitude of oscillation	N/A	$\sim 4.1*10^{12}$	-
Oscillation frequency	N/A	1.43	Hz

F. Discussion

Based on the root-locus plots for the primary loops, the resulting output indicates the instability of the cascade control systems. In Fig. 3 for the primary loop when using a PI secondary controller, one pole was moving right, suggesting that as gain varies, the system remains unstable, with the closed-loop pole in the right half-plane, confirming oscillatory instability. An attempt was made to modify the results by adding derivative (D) action to the secondary controller and studying its effect on the primary controller. However, the root-locus plot for the PID case (Fig. 5) also showed poles crossing into the unstable region and suggesting instability.

The Bode plot for the primary control loop with PI and PID secondary controllers in Figs. 4 and 6 both show a positive Gain Margin and infinite Phase Margin, indicating stability, but will have trouble tracking various reference signals because the gain is below 0 dB for all frequencies. It is important to note that Bode analysis evaluates loop gain and phase but does not directly account for pole placement, making it less suitable for detecting oscillatory modes in high-order systems [10].

The step response of the primary P controller combined with a secondary PI controller (Fig. 7. A) confirmed instability, exhibiting a continuously increasing output without settling. Because of this divergence, rise time, settling time, and oscillation frequency cannot be defined. This instability arises from integral action, which tends to reduce phase margin, increasing the risk of oscillations in high-order systems [11]. With the secondary PID controller (Fig 7. B), the response also did not reach steady state, showing sustained oscillations at approximately 1.4 Hz. The oscillation amplitude grew dramatically from about $1.21*10^{10}$ at 10.4 sec to $3.99* 10^{13}$ at 13.9 sec, indicating growth and instability. This behavior suggests that the Derivative action may introduce aggressive corrections and amplify noise.

Attempts to improve performance by tuning τ_{I2} and τ_{D2} were unsuccessful, and both control loops remained unstable. This may be because, in the delayed coking unit's semi-batch operation, pressure signals often contain high-frequency noise from changing feed conditions, vapor–liquid interactions, and sensor limits. In the cascade control system, this noise can be amplified by the derivative action in the secondary controller and passed to the primary loop, making the system less stable. Nonlinear effects- such as changes in process gain during the coking cycle, valve sticking, and coke buildup-can cause the real process to behave differently from the identified linear transfer functions. When this happens, PI/PID controllers tuned for the ideal model may lose their ability to keep the system stable, as the process dynamics shift and the designed stability margins are reduced [12, 13].

A re-analysis of the root locus in the previous study [8] revealed that some zeros seem to be close to the imaginary axis in the secondary control loop with a high ultimate gain value, as shown in the root-locus plot for the secondary controller in Fig. 8 below:

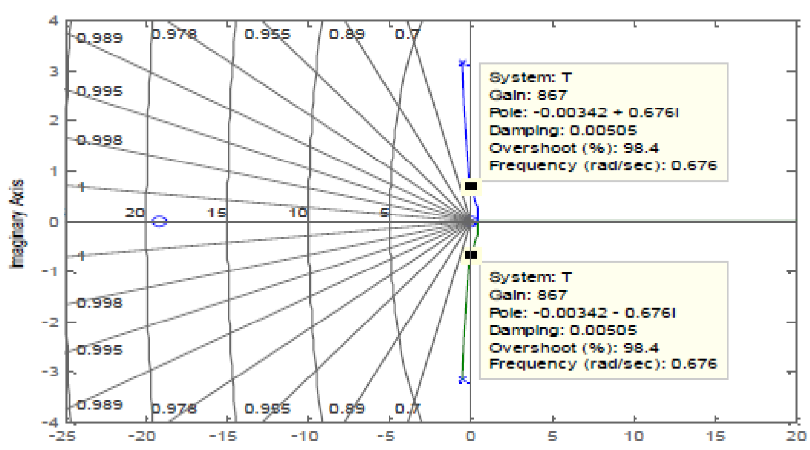

Fig. 8. Root locus plots for the Secondary loop with P controller, Hamed et al., [8].

The step response from the previous study for the same secondary control loop with a P controller exhibited acceptable stability and overall performance; however, it showed a noticeably high overshoot and a relatively slow settling, as illustrated in Fig. 9.

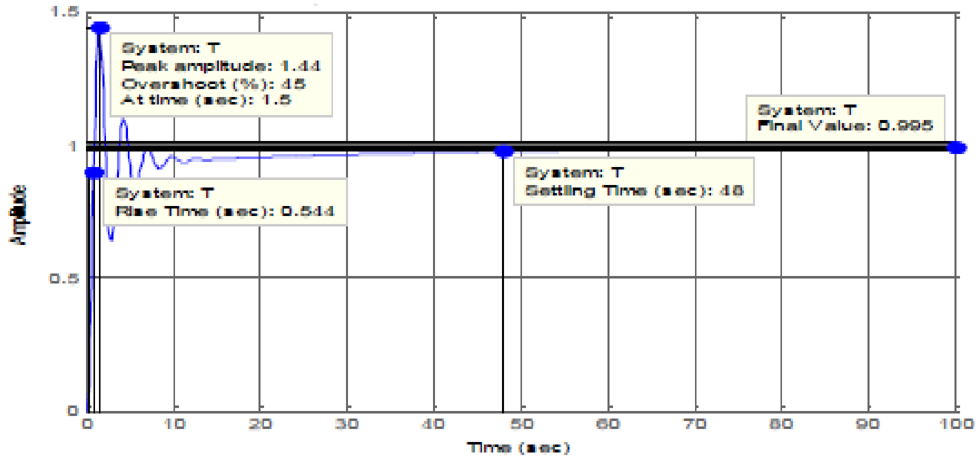

Fig. 9. System response for secondary closed loop with P controller, Hamed et al., [8].

The step response of the primary loop with a primary P-controller and a secondary P-controller, shown in Fig. 10, had no overshoot but was very slow, with a rise time of 61 s and a settling time of 171 s. The steady-state tracking was poor, reaching only 0.34 at 350 s, which suggests low outer loop gain or limits in the loop design and tuning to remove steady-state error.

The primary loop with a PI controller showed a much slower response, with a rise time of 419 s and a settling time of 767 s, as seen in Fig. 11. The steady-state error was eliminated, but the slow response and big undershoot meant the tuning was too conservative, causing poor transient performance. The PID primary loop step response in Figure 12 shows the controller reached steady-state accurately with no overshoot and fast settling, but the speed of the response was still less than what is needed for the application. Since the P-only controller gave high overshoot and slow settling, and the PI and PID controllers caused instability, it is better to try methods that match the delayed coking process. These include adjusting the secondary-loop tuning during different coking phases, using two-degree-of-freedom control to reduce overshoot, adding feedforward from known disturbances, and filtering noisy measurements. Classical methods like lead/lag compensation, MPC, and state feedback can also be used to improve stability and performance.

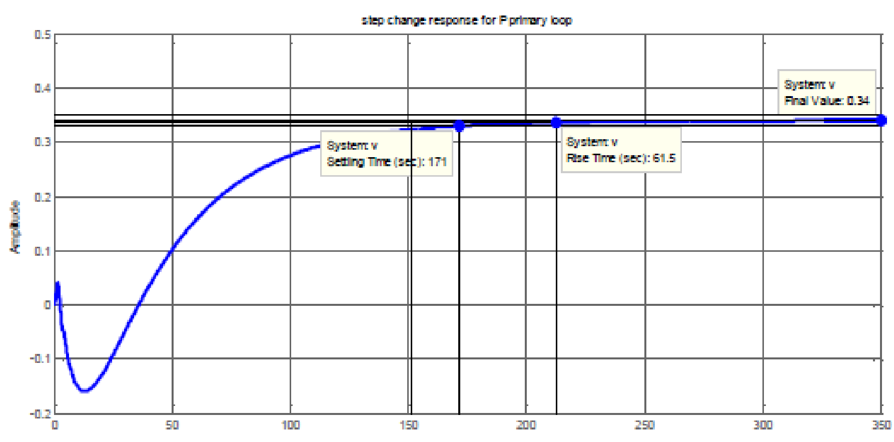

Fig. 10. System response for primary closed loop with P secondary controller, Hamed et al., [8].

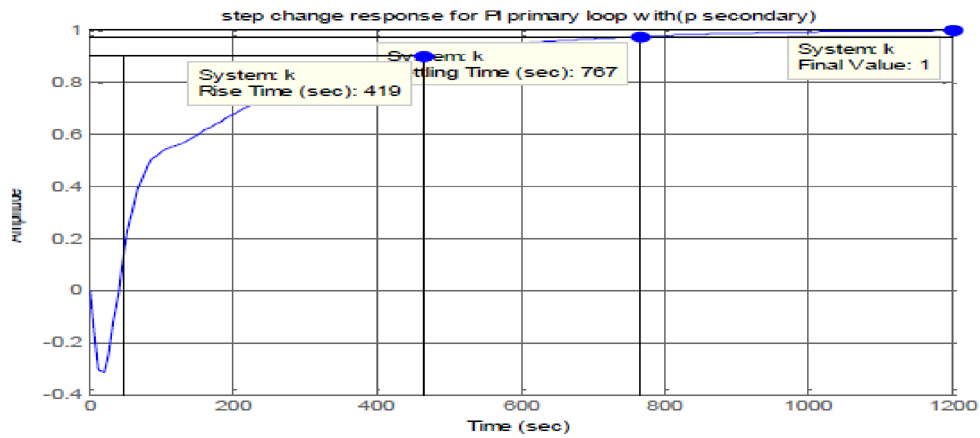

Fig. 11. System response for primary PI closed loop with P secondary controller, Hamed et al.,[8].

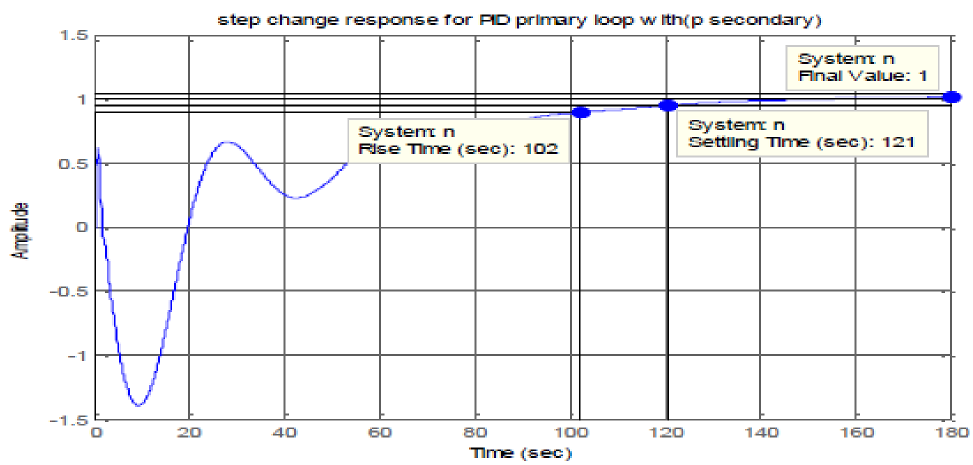

Fig. 12. System response for primary PID closed loop with P secondary controller, Hamed et al., [8].

IV. CONCLUSIONS

This study analyzed the stability of a modified cascade control strategy for the coking drum of the delayed coking unit at Khartoum Refinery using a model from previous research, as real plant data were not available. The modified design evaluated the impact of replacing the secondary P controller with PI and PID controllers on the stability of the primary controller and therefore the drum pressure control. Using MATLAB simulations, the open- and closed-loop transfer functions were developed, and stability was analyzed via direct substitution, root locus, Bode plot, and step response.

The results have demonstrated that the integration of PI and PID controllers in the secondary loop is not effective with the cascade control for the system examined in this study and leads to instability due to reduced phase margin and noise amplification. This indicates that while these controllers are generally advantageous, they are not suitable for all high-order systems, particularly those with inherent lags or coupled dynamics like delayed coking units, while the P-only controller gives high overshoot and slow settling. Adaptive secondary-loop tuning, two-degree-of-freedom control, feed-forward compensation, and filtering of noisy measurements are recommended. Future work can also include exploring advanced control strategies such as lead/lag compensators, Model Predictive Control (MPC), and State Feedback Control (SFC) for more robust and responsive control.

ACKNOWLEDGEMENTS

The authors would like to express their sincere gratitude to the Faculty of Graduate Studies and the Department of Chemical Engineering of Sudan University of Science and Technology for their continuous support and guidance. This paper is derived from a master's thesis submitted in partial fulfillment of the requirements for the degree of M.Sc. in Chemical Engineering.

REFERENCES

[1] M. A. Fahim, T. A. Al-Sahhaf, and A. S. Elkilani, *Fundamentals of Petroleum Refining*. Amsterdam, The Netherlands: Elsevier, 2010.
[2] A. Zavarce, "Petroleum coke: What it is, what it's used for, and its applications in industry," *Inspenet*, 2024.
[3] S. E. LeBlanc and D. R. Coughanowr, *Process Systems Analysis and Control*, 3rd ed. New York, NY, USA: McGraw-Hill, 2009.
[4] D. E. Seborg, T. F. Edgar, D. A. Mellichamp, and F. J. Doyle III, *Process Dynamics and Control*, 4th ed. Hoboken, NJ, USA: Wiley, 2016.

[5] W. Zhang, D. Huang, Y. Wang, and J. Wang, “Adaptive state feedback predictive control and expert control for a delayed coking furnace,” *Chin. J. Chem. Eng.*, vol. 16, no. 4, pp. 590–598, 2008.
[6] J. Smith and A. Jones, “Application of advanced process control in delayed coking unit,” *J. Process Control*, vol. 15, no. 2, pp. 123–130, 2023.
[7] N. Lima et al., “Simulation, optimization and control of heavy oil upgrading processes: Application to a delayed coking unit,” *J. Petrol. Eng.*, vol. 45, no. 2, pp. 123–145, 2023.
[8] T. A. I. Hamed, G. A. Gasmelseed, and I. H. Elamin, “Advanced control of the delayed coking unit in Khartoum refinery,” *J. Appl. Ind. Sci.*, vol. 2, no. 4, pp. 188–200, 2014.
[9] D. Houcque, *Introduction to MATLAB for Engineering Students*. Evanston, IL, USA: Northwestern Univ., 2005.
[10] G. F. Franklin, J. D. Powell, and A. Emami-Naeini, *Feedback Control of Dynamic Systems*, 7th ed. Upper Saddle River, NJ, USA: Pearson, 2014.
[11] K. Ogata, *Modern Control Engineering*, 5th ed. Upper Saddle River, NJ, USA: Prentice Hall, 2010.
[12] K. J. Åström and R. M. Murray, *Feedback Systems: An Introduction for Scientists and Engineers*. Princeton, NJ, USA: Princeton Univ. Press, 2021.
[13] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2002.