

2004-01-01

Comparison between theoretical and practical phase computation

Pauline Sourdille

Technological University Dublin

Aidan O'Dwyer

Technological University Dublin, aidan.odwyer@tudublin.ie

Follow this and additional works at: <https://arrow.tudublin.ie/engscheleart>



Part of the [Controls and Control Theory Commons](#)

Recommended Citation

Sourdille, Pauline and O'Dwyer, Aidan : Comparison between theoretical and practical phase computation. Proceedings of the 5th IFAC Workshop on Time Delay Systems, K.U. Leuven, Belgium, September, 2004. doi:10.21427/attr-5z17

This Conference Paper is brought to you for free and open access by the School of Electrical and Electronic Engineering at ARROW@TU Dublin. It has been accepted for inclusion in Conference papers by an authorized administrator of ARROW@TU Dublin. For more information, please contact arrow.admin@tudublin.ie, aisling.coyne@tudublin.ie.



This work is licensed under a [Creative Commons Attribution-NonCommercial-Share Alike 4.0 License](#)

Comparison between Theoretical and Practical phase Computation

Pauline Sourdille[♠] and Aidan O'Dwyer

[♠]*Department of Control Systems
Dublin Institute of Technology, Kevin Street
Dublin 8
IRELAND
E-mail: pauline.sourdille@dit.ie*

^{*}*Department of Control Systems
Dublin Institute of Technology, Kevin Street
Dublin 8
IRELAND
E-mail: aidan.odwyer@dit.ie*

Abstract: Transfer functions which are a composite of delay terms cause problems for computer packages, such as MATLAB, in calculating the frequency response open loop transfer function, as the form of the open loop transfer function is irrational. So, the delay terms have to be approximated. There are several ways to evaluate a delay term. A calculator such as MATHEMATICA uses the exact representation of a delay called the Euler form. A computer package such as MATLAB uses a rational approximation for the delay. Using the MATLAB built-in function *pade*(n, τ), the open loop transfer function of a system with delay can be calculated and the bode plot may be obtained using another MATLAB built-in function *bode*(*transfer function*). Unfortunately, it was noticed that the phase was shifted by an angle from the correct phase, due to the trigonometric properties of the \tan^{-1} function. Consequently, methods to construct the correct bode plot were studied and developed. Copyright © 2004 IFAC

Keywords: Smith predictor, Dead-time compensator, delay compensation, arctangent function

1. INTRODUCTION

The starting point of this analysis was the need to evaluate the stability and the robustness of the first and second modified Smith predictor structures developed by Sourdille and O'Dwyer (2003a, b, c), compared to a classical Smith predictor structure (Smith, 1957). Most methods for the evaluation of the stability and robustness of a closed loop system use the open loop transfer function of this system in the frequency domain. By plotting the magnitude of the transfer function in decibels, and the phase in degrees, both versus frequency, ω , a bode plot may be obtained (D'Azzo and Houpis, 1995 and Kuo, 1995). A general procedure to find the frequency response of any systems represented by its transform domain transfer function $g(s)$ (Ogunnaike and Ray, 1994) is to first, substitute $j\omega$ for s in the transfer function expression to obtain the corresponding frequency response transfer function, $g(j\omega)$, then rationalise $g(j\omega)$ to obtain the Cartesian form and finally compute the magnitude and the phase using

the imaginary and real parts of the frequency response transfer function. The phase is of concern as it defines the stability of a system: if the phase is more negative than -180° when the magnitude equals 0 dB, the system is unstable. Unfortunately, the open loop transfer function of a Smith predictor structure (equation (1)) cannot be calculated by direct programming using computer packages, such as MATLAB or MATHEMATICA.

$$G = \frac{G_c G_p e^{-s\tau_p}}{1 + G_c G_m - G_c G_m e^{-s\tau_m}} \quad (1)$$

where G_p and G_m are the process and model non-delay transfer functions, τ_p and τ_m are the process and model delays and G_c is the primary controller of the Smith predictor structure.

Equation (1) is a composite of delay terms, which causes problems for MATLAB, for example, in calculating the frequency response open loop transfer function, as the form of the open loop transfer function is irrational. So, the delay terms have to be

approximated. There are several ways to evaluate a delay term. A calculator such as MATHEMATICA uses the exact representation of a delay called the Euler form:

$$e^{-j\omega\tau} = \cos(\omega\tau) - j\sin(\omega\tau) \quad (2)$$

A computer package, such as MATLAB, uses rational approximations for the delay. Rational functions have a numerator and denominator, which can be a composite of real and imaginary parts. The most commonly employed approximation is the Padé approximation. But there are also other approximations such as the Laguerre approximation, the Product approximation and the Taylor approximation (Sourdille, 2003d). These rational functions have been developed because the Euler form is irrational in the time domain.

Using the MATLAB built-in function $pade(n, \tau)$ (where n is the order of the approximation and τ is the delay time), the open loop transfer function of the Smith predictor structure (equation (1)) can be calculated and the bode plot may be obtained using another MATLAB built-in function [$bode(transfer\ function)$]. Unfortunately, it was noticed when this function was used, that the phase was shifted by an angle from the correct phase. Figure 1 shows the bode plot of the Smith predictor open loop transfer function (equation (1)) using the built-in function $bode(transfer\ function)$ and a second order Padé approximation for the delay. For this plot, the process and model transfer functions are given by:

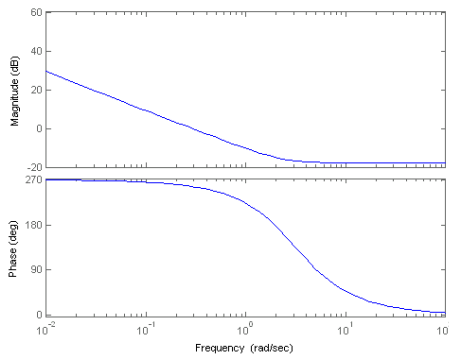
$$G_p e^{-s\tau_p} = \frac{K_p}{T_p s + 1} e^{-s\tau_p} = \frac{1.6}{0.5s + 1} e^{-1.2s} \quad \text{and}$$

$$G_m e^{-s\tau_m} = \frac{K_m}{T_m s + 1} e^{-s\tau_m} = \frac{2}{0.7s + 1} e^{-1.4s}$$

The primary controller is chosen to be:

$$G_c = \frac{T_m s + 1}{K_m (s + 1)(1 - e^{-s\tau_p})} = \frac{0.7s + 1}{2(s + 1)(1 - e^{-1.4s})}$$

Figure 1: Smith predictor bode plot using the built-in function $bode(transfer\ function)$



The analytical computation of the phase using the Euler form gives a phase tending to zero when the frequency tends to zero. As stated earlier, it is obvious that the built-in function $bode(transfer\ function)$ does not give the correct phase, as at zero frequency the phase has been shifted by an angle of 270° . It has also been noticed that this phase shift

increases as the order of the Padé approximation increases. So, it is not possible to assess the stability of the Smith predictor structures using the MATLAB built-in function $bode(transfer\ function)$. Consequently, methods to construct the correct bode plot were studied and developed. The cause of the problem was discovered to be the trigonometric properties of the arctangent function. In this paper, this issue will be explored with reference to a proportionally-controlled third order lag process, and a proportionally-controlled third order lag plus delay process, rather than the Smith predictor, due to the complexity of the open loop transfer function in the latter structure.

2. THIRD ORDER LAG PROCESS

The open loop transfer function of a proportionally-controlled process is given by equation (3).

$$G_{OL} = G_c G_p \quad (3)$$

To investigate the phenomenon associated with the arctangent function in calculating the open loop phase, a third order lag process (equation (4)) is used with a controller given by equation (5).

$$G_p = \frac{1}{(s+1)^3} = \frac{1}{s^3 + 3s^2 + 3s + 1} \quad (4)$$

$$G_c = 1 \quad (5)$$

By letting $s = j\omega$, and using the expressions for the process and controller transfer functions, the open loop transfer function in the frequency domain becomes:

$$G_{OL} = \frac{1}{(j\omega+1)^3} = \frac{1}{-j\omega^3 - 3\omega^2 + 3j\omega + 1} \quad (6)$$

The phase is given in equation (7) using the result which states that the phase of a transfer function of the form $\frac{1}{(Ts+a)^n}$ ($\frac{1}{(Tj\omega+a)^n}$ in the frequency

domain) is $-n \tan^{-1}\left[\frac{\omega T}{a}\right]$.

$$\phi_c = -3 \tan^{-1}[\omega] \quad (7)$$

But many computer packages do not use the factorised form of G_{OL} (left hand side of equation (4)). Effectively, they use the extended form (right hand side of equation (4)) to calculate the Cartesian form, as they do not have subroutines to evaluate the correct phase using the factorised form.

Using the extended form of (6), the practical computation of the open loop transfer function phase results in:

$$\phi_{G_{OL}} = -\tan^{-1}\left[\frac{-\omega^3 + 3\omega}{-3\omega^2 + 1}\right] \quad (8)$$

It can be noticed that there is a discontinuity at $-3\omega^2 + 1 = 0$ i.e. when $\omega = \sqrt{\frac{1}{3}}$ rad.s⁻¹. The

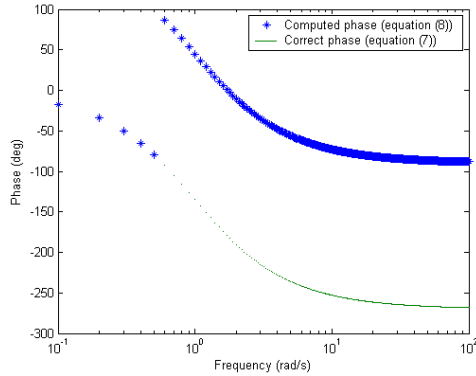
discontinuity is due to the arctangent function, which is not defined when the denominator equals zero. Figure 2 represents a graph of the open loop phase versus frequency, applying equations (7) and (8). The

limits of equation (8) are zero when the frequency tends to zero and -90° when the frequency tends to infinity. At the discontinuity, the limits tend to -90° and $+90^\circ$ as ω increases towards $\frac{1}{\sqrt{3}}$ and ω decreases

towards $\frac{1}{\sqrt{3}}$, respectively. These limits are different

from the correct limits, which are zero when the frequency tends to zero and -270° when the frequency tends to infinity. It can also be noticed that there is a shift of 180° at the discontinuity. Finally, it can be noticed that at $\omega < \frac{1}{\sqrt{3}}$, the phases are identical.

Figure 2: Phase versus frequency characteristic of equations (7) and (8)



To analyse this problem, first phase values are calculated at small frequencies.

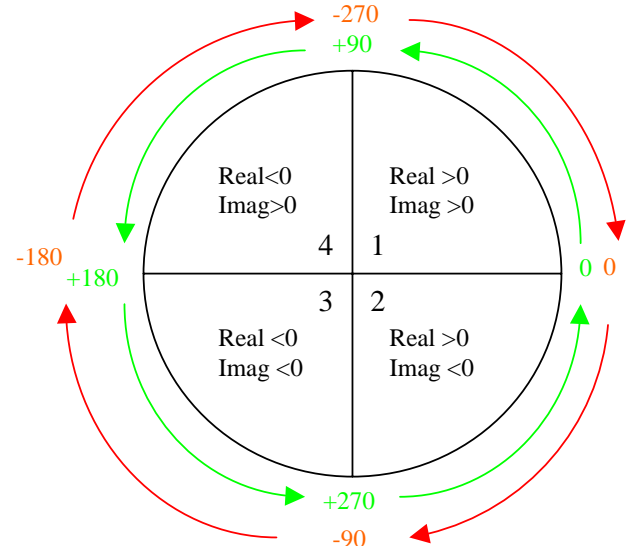
At $\omega=0$, $\phi = -\tan^{-1}[0] = 0^\circ$ and at $\omega = 1$,

$$\phi = -\tan^{-1}\left[\frac{-1+3}{-3+1}\right] = -\tan^{-1}\left[\frac{2}{-2}\right] = 45^\circ.$$

The latter phase is wrong, as it is known that the phase should be in the bottom right quadrant of the trigonometric circle, as the numerator is positive and the denominator is negative. Figure 3 shows the variation of signs for the numerator ($\text{Im}(\omega)$) and the denominator ($\text{Re}(\omega)$) around the trigonometric circle. The computed phase (equation (8)) covers only quadrants 1 and 2 of the trigonometric circle, which represents angles between $+90^\circ$ and -90° .

In conclusion, when the phases are calculated using many computer packages, the obtained phases are wrong due to the trigonometric properties of the arctangent function. In fact, the computed phase covers half of the trigonometric circle instead of the complete circle. In addition, there are discontinuities when the denominator of the arctangent function equals zero. At the discontinuity, there is a shift in the phase, which may vary depending on the system being analysed, and may be defined as $-n*180^\circ$, where n is an integer.

Figure 3: Trigonometric circle



5. THIRD ORDER LAG WITH TIME DELAY PROCESS

To investigate the consequences of the introduction of a time delay in the computation of the phase, the transfer function considered in Section 2 (equation (4)) is multiplied by a delay term (delay value=1 second). The corresponding frequency response open loop transfer function is given by:

$$G_{OL} = \frac{e^{-j\omega}}{(j\omega+1)^3} = \frac{e^{-j\omega}}{-j\omega^3 - 3\omega^2 + 3j\omega + 1} \quad (9)$$

The correct value of the phase for a third order lag with a delay is as follows:

$$\phi_c = -\omega - 3 \tan^{-1}[\omega] \quad (10)$$

Using the Euler form for the delay (equation (2)), the open loop transfer function in the frequency domain becomes:

$$G_{OL} = \frac{(\cos \omega - j \sin \omega)}{(j\omega+1)^3} = \frac{\cos \omega - j \sin \omega}{-j\omega^3 - 3\omega^2 + 3j\omega + 1} \quad (11)$$

The phase computed using a computer package such as MATLAB is as follows:

$$\phi = \tan^{-1}\left[\frac{-\sin \omega}{\cos \omega}\right] - \tan^{-1}\left[\frac{-\omega^3 + 3\omega}{-3\omega^2 + 1}\right] \quad (12)$$

There are two types of discontinuities for equation (12). The first type corresponds to the value of ω when $\cos \omega = 0$. The frequencies at which these discontinuities occur are $\omega = \frac{n\pi}{2\tau}$, where n is an odd

integer. The second type of discontinuity corresponds to the (unique) ω value when $-3\omega^2 + 1 = 0$. Figure 4 represents the phase versus frequency characteristic of equation (12). Table 1 represents relevant data corresponding to the first discontinuity points of equation (12), indexed on Figure 4. It can be added that the phase shift at the discontinuities is 180° .

Figure 4: Phase versus frequency characteristic of equation (12)

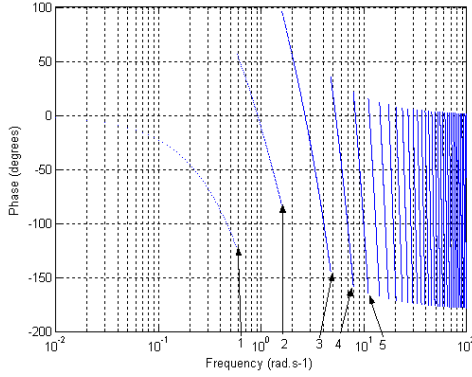


Table 1: Values at the discontinuities

Points	Discontinuities at: (rad.s ⁻¹)
1	$0.577 \approx \sqrt{\frac{1}{3}}$
2	$1.66 \approx \frac{\pi}{2}$
3	$4.75 \approx \frac{3\pi}{2}$
4	$7.8 \approx \frac{5\pi}{2}$
5	$11 \approx \frac{7\pi}{2}$

Due to the irrational property of the Euler form in the time domain, another approximation, which rationalises the transfer function of the time delay has been developed to analyse systems with dead time. A rational function is the ratio of two polynomial functions:

$$y = \frac{a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0}{b_m x^m + b_{m-1} x^{m-1} + \dots + b_2 x^2 + b_1 x + b_0} \quad (13)$$

where n and m are non-negative integers that define the degree of the numerator and denominator, respectively. Fitting rational function models to the delay may be referred as the Padé approximation process (Ogunnaike and Ray, 1994). There are several advantages in using a rational function (Rational Function, 2004). First, rational function models have moderately simple form and are typically smoother and less oscillatory than polynomial models. They also have excellent asymptotic properties. Finally, they can often be used to model complicated structures, with a fairly low degree in both the numerator and denominator; this means that fewer coefficients will be required compared to the polynomial model. However, rational functions have disadvantages: the properties of the rational function family are not as well known to engineers and scientists as are those of the polynomial family; the literature on the rational function family is more limited; unconstrained rational function fitting can at times, result in undesired nuisance asymptotes (vertically) due to poles in the denominator

polynomial. The range of values affected by the function “blowing up” may be quite narrow, but such asymptotes, when they occur, are a nuisance for local interpolation in the neighborhood of the asymptote point.

The determination of the open loop phase is now carried out when the time delay is approximated by a Padé approximation. The n^{th} order Padé approximation used by MATLAB built-in function `pade(n,τ)` and by numerous authors (Silva *et al.* (2001), Seborg *et al.* (1989), Stephanopoulos (1984), Bequette (2003) and Dutton *et al.* (1997)), is given by the following expression:

$$e^{-s\tau} = \frac{\sum_{k=0}^n \frac{(2n-k)!}{k!(n-k)!} (-s\tau)^k}{\sum_{k=0}^n \frac{(2n-k)!}{k!(n-k)!} (s\tau)^k} \quad (14)$$

If a second order approximation is used, the open loop transfer function in the frequency domain is given by:

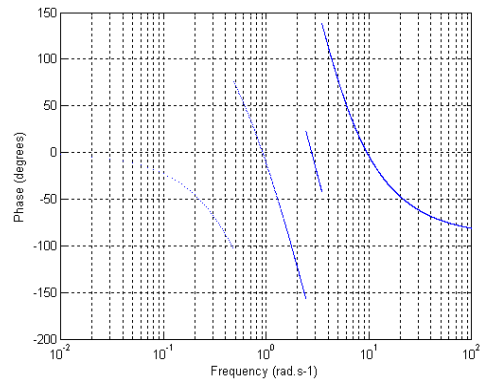
$$G_{OL} = \frac{12 - 6j\omega - \omega^2}{9\omega^4 - 55\omega^2 + 12 + j(\omega^5 - 33\omega^2 + 12)} \quad (15)$$

The phase may be computed:

$$\phi = \tan^{-1} \left[\frac{-6\omega}{12 - \omega^2} \right] - \tan^{-1} \left[\frac{\omega^5 - 33\omega^3 + 42\omega}{9\omega^4 - 55\omega^2 + 12} \right] \quad (16)$$

There are discontinuities at $2\sqrt{3}$ rad.s⁻¹, 2.426 rad.s⁻¹ and 0.476 rad.s⁻¹ (see Figure 5). It is clear that the phenomenon associated with the arctangent function as detailed in the earlier part of the paper, arises again. It has also been noticed that the number of discontinuities increases as the order of the approximation increase.

Figure 5: Phase versus frequency characteristic for equation (16)



4. CORRECTED PHASE CALCULATION

This section will focus on correcting the phase by developing a procedure that can be programmed. As explained, the computation of the phase depends on the sign of the numerator and the denominator of the open loop transfer function. The following rule, labelled Rule 1, computes the correct phase, depending on the sign of the real and imaginary parts.

- If $\text{Im}(\omega) > 0$ and $\text{Re}(\omega) > 0$ or if $\text{Im}(\omega) < 0$ and $\text{Re}(\omega) > 0$

$$\phi = \tan^{-1} \left[\frac{\text{Im}(\omega)}{\text{Re}(\omega)} \right]$$

- If $\text{Im}(\omega) < 0$ and $\text{Re}(\omega) < 0$ or if $\text{Im}(\omega) > 0$ and $\text{Re}(\omega) < 0$

$$\phi = - \left[\pi + \tan^{-1} \left[\frac{\text{Im}(\omega)}{\text{Re}(\omega)} \right] \right]$$

A comparison between phase values using equation (7) and expression (8), modified by the implementation of Rule 1, shows that the results obtained are identical. The determination of the limiting phase also shows that the calculated phase tends to the same limits as the correct phase.

$$\lim_{\omega \rightarrow +\infty} \phi_c = \lim_{\omega \rightarrow +\infty} -3 \tan^{-1}[\omega] = -3 * 90 = -270^\circ$$

$$\lim_{\omega \rightarrow +\infty} \phi = \lim_{\omega \rightarrow +\infty} - \tan^{-1} \left[\frac{-\omega^3 + 3\omega}{-3\omega^2 + 1} \right] = \lim_{\omega \rightarrow +\infty} - \tan^{-1} \left[\frac{-\omega^3}{-3\omega^2} \right]$$

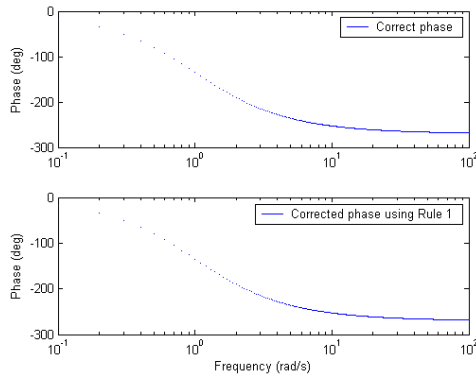
Using Rule 1 for the computation of the phase:

$$\lim_{\omega \rightarrow +\infty} \phi = \lim_{\omega \rightarrow +\infty} - \left[180 + \tan^{-1}[\omega] \right] = -270^\circ$$

Using MATLAB, a program has been developed to integrate Rule 1 into the computation of the phase. Figure 6 shows the correct phase (equation (7)) and the corrected phase (equation (8) with Rule 1).

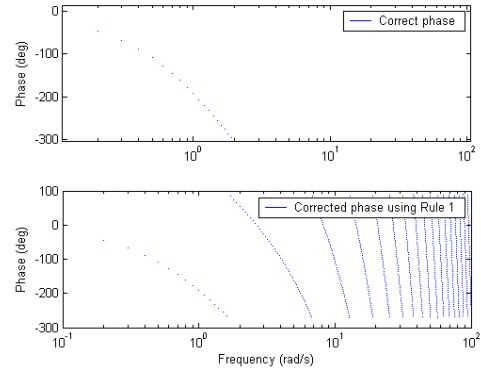
As it can be noticed, equation (8) using Rule 1 gives the same phase as the correct phase (equation (7)). So, it is possible to obtain the correct phase by developing a specific program with computer packages such as MATLAB and MATHEMATICA.

Figure 6: Correct phase and corrected phase using Rule 1 for third order lag system



If Rule 1 is applied to the third order lag system with delay, Figure 7 is obtained.

Figure 7: Correct phase and corrected phase using Rule 1 for third order lag system with delay



As can be noticed, the phase shift is 360° . It can also be noticed that Rule 1 corrects only the phase for the first discontinuity. So, Rule 1 has to be modified. By using a modified version of Rule 1, which shifts the phase by -360° for the second rotation around the trigonometric circle, the correct phase may be recovered. However, these modifications are only valid for the second rotation; another phase shift has to be done for the third rotation and so on. This represents a programming difficulty, as the discontinuities have to be determined. A possibility is to consider the following assumption:

Assumption 1:

As processes are generally low pass in nature, it may be assumed that phases calculated at higher frequencies will be more negative than phases calculated at lower frequencies. Phase shifts observed represent discontinuities in the phase expression. These discontinuities are caused by trigonometric properties of the function \tan^{-1} .

Possible alternative approaches would be to use a polynomial approximation for the arctangent function, e.g. Lyons (2004) or the use of a “look up table” where the values of the imaginary and real parts of the complex number specify an address in the read only memory containing an approximation of the corresponding angle. These approaches remain to be investigated fully.

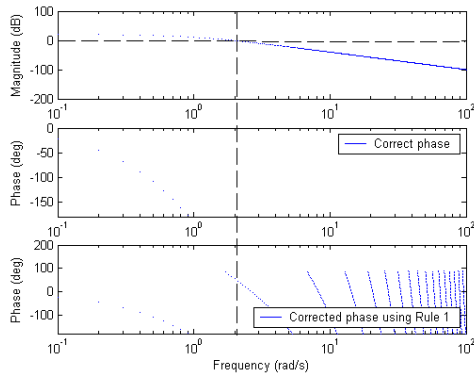
6. AN EXAMPLE

As stated in the introduction, the concern of this analysis is the evaluation of the stability of a process. To validate the use of Assumption 1, let's consider an unstable closed loop transfer function. The open loop transfer function of this system is given by equation (3); the process transfer function is given by equation (9) and the controller transfer function is as follows:

$$G_c = 10 \quad (17)$$

Figure 8 shows the correct phase and the corrected phase using Rule 1. The y-axis limit for the phase responses is -180° , i.e. the horizontal line of the plot represents -180° .

Figure 8: Correct phase and corrected phase using Rule 1 for unstable third order system with delay



From the correct phase response, it clear that the system is unstable as the phase is more negative than -180° when the magnitude equals 0 dB. If Assumption 1 is used for the corrected phase, the same conclusion may be drawn. So, it can be concluded that Rule 1 and Assumption 1 may be used to evaluate the stability of a system.

6. CONCLUSION

It has been shown that the phase formulae, as used in a computer package such as MATLAB, vary from the correct phase expressions. This is due to the trigonometric properties of the function \tan^{-1} . The correct phase expression can be obtained by shifting the phase at the discontinuities by an angle of -180° . Rule 1 and Assumption 1 have to be used to obtain the correct phase values.

Approximations are necessary to calculate transfer functions which have a time delay term, to produce an overall rational transfer functions.

REFERENCES

- Bequette, B.W. (2003). Process control, modelling, design and simulation, *Prentice Hall International Series in the physical and chemical Engineering Sciences*
- D'Azzo, J. and Houpis, C., H. (1995). Nyquist, Bode and Nichols plots, *The control Handbook, IEEE Press*
- Dutton, S., Thompson, S. and Barraclough, B. (1997). The art of control engineering, *Addison-Wesley*
- Kuo, B.,C., Automatic (1995). Control Systems, Seventh Edition, *Prentice Hall International Editions*
- Lyons R. (2004). Another Contender in the Arctangent Race, *IEEE Signal Processing Magazine, January 2004*, pp 109-110
- Ogunnaike, B.A. and Ray, W.H. (1994) Process dynamics modelling and Control, *Oxford University Press*
- Rational function. (2004). www.itl.nist.gov/div898/handbook/pmd/section8/pmd812.htm

- Seborg, D.E., Edgar, T.F. and Mellichamp, D.A. (1989). Process dynamics and control, *John Wiley and Sons*
- Smith, O.J.M. (1957). Closer control of loops with dead time. *Chemical Engineering Progress*, **53**, 217-219,
- Silva, G., Datta, A. and Bhattacharyya, S.P. (2001). Control Design via Pade approximation can lead to instability, *Proceedings of the 40th IEEE Conference on Decision and control*, Orlando, 2001, pp 4733-4737
- Sourdille, P. and O'Dwyer, A. (2003a). An outline and further development of Smith predictor based methods for compensation of processes with time delay. *Irish Signals and Systems Conference*, Limerick, Ireland ,pp. 338-343
- Sourdille, P. and O'Dwyer, A. (2003b). A new modified Smith predictor Design. *International Symposium on Information and Communication Technologies Conference*, Trinity College, Dublin, Ireland ,pp.404-409
- Sourdille, P. and O'Dwyer, A. (2003c). New modified smith predictor designs. *International Federation of Automatic Control workshop on Time-delay systems, INRIA, Rocquencourt, France*
- Sourdille, P. (2003). Delay approximations. *Technical report*, Dublin Institute of Technology
- Stephanopoulos, G. (1984). Chemical process control: an introduction to theory and practise, *Prentice Hall International Editions*