

Chương 6

VÒNG KHÓA PHA VÀ PHƯƠNG PHÁP GIẢI PHƯƠNG TRÌNH VI PHÂN

6.1. Mở đầu

Dù vẫn còn sớm trong nghiên cứu mô phỏng nhưng ta có công cụ cần thiết để: sử dụng hiệu quả mô phỏng, giám sát việc thiết kế và các đặc tính hoạt động của một số hệ thống quan trọng. Mặc dù tồn tại nhiều hệ thống lựa chọn cho vấn đề này, nhưng trọng tâm của chương là vòng khóa pha PLL, có một vài lý do cho sự lựa chọn này. *Trước hết*, PLL là một trong những *khối cơ bản* nền tảng trong nhiều phân hệ của hệ thống truyền thông hiện đại. PLL được dùng rộng rãi trong: bộ tổng hợp tần số, bộ chia tần số, bộ nhân tần số, để đồng bộ sóng mang và ký hiệu ở các máy thu nhất quán. Do tính *phi tuyến* của PLL nên rất khó phân tích và thiết kế theo các phương pháp truyền thống (không dựa vào mô phỏng) vì vậy mô phỏng là giải pháp được sử dụng rộng rãi để thiết kế và phân tích. Cần lưu ý rằng, kiến thức cơ bản về phương pháp tích phân số cung cấp những khái niệm cần thiết để triển khai mô phỏng hiệu quả cho các PLL và cho nhiều hệ thống quan tâm khác. *Một lý do khác* để sử dụng PLL trong mô phỏng là PLL chứa đường hồi tiếp. Ta sẽ thấy rằng các hệ thống chứa đường hồi tiếp phải được mô phỏng cẩn thận, đặc biệt là tần số lấy mẫu nếu muốn giảm thiểu lỗi mô phỏng.

Như đã đề cập ở chương 4, việc phân tích dựa trên các mô hình hệ thống thông thấp là tương đương với các mô hình hệ thống băng thông tương ứng, tất nhiên các mô hình thông thấp phải được triển khai chính xác. Ngoài ra, thời gian thực hiện mô phỏng ở các mô hình mô phỏng được rút ra từ những mô hình phân tích thông thấp nhanh hơn nhiều. Cuối cùng, tất cả các mô phỏng được trình bày ở chương này là những mô phỏng *tất định* như được đề cập ở chương 1.

Mặc dù trọng tâm của chương 6 là minh họa một số kỹ thuật mô phỏng cơ bản và chỉ sử dụng PLL phục vụ nghiên cứu hoạt động truyền thông nhưng lý thuyết được trình bày cũng đủ để hiểu và "kiểm tra tính đúng đắn" các kết quả mô phỏng. Chương này được bắt đầu bằng việc mô tả cơ bản về PLL. Sử dụng vòng bậc nhất làm ví dụ minh họa, ta sẽ chỉ ra rằng làm thế nào vòng đạt được khóa pha, sau đó mở rộng nghiên cứu cho vòng bậc hai. Dùng mô phỏng để minh họa hiện tượng trượt tuần hoàn và xác định thời gian cần thiết để đạt được khóa pha. Mô phỏng được thực hiện bằng cách triển khai mô hình mô phỏng rời rạc theo thời gian cho PLL.

Phần cuối của chương chỉ ra, không những có được các công cụ để nghiên cứu PLL mà còn có được công cụ mô phỏng cần thiết để nghiên cứu hệ thống được định nghĩa ở dạng phương trình vi phân. Phương trình vi phân, hệ thống tương ứng có thể phi tuyến thay đổi theo

thời gian (vừa phi tuyến vừa thay đổi theo thời gian). Vì vậy kết thúc chương này, ta sẽ có các công cụ cần thiết để mô phỏng nhiều hệ thống phức tạp khác nhau.

Lưu ý rằng: Nghiên cứu chi tiết về PLL và ứng dụng trong việc đồng bộ sóng mang và định thời ký hiệu được cho ở Phụ lục 6B

6.2. Khái niệm vòng khóa pha cơ bản

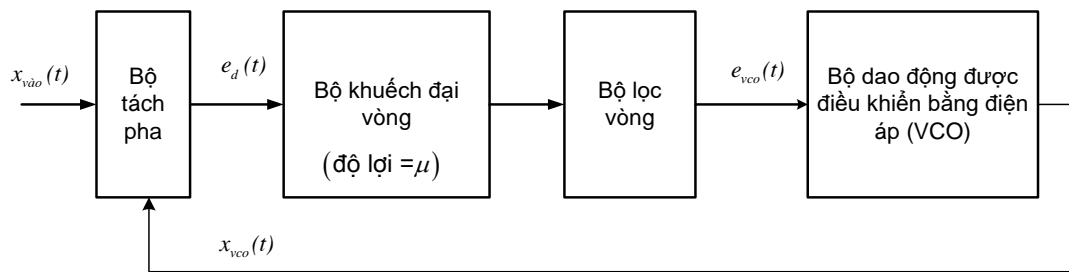
Mô hình cơ bản của PLL được minh họa ở hình 6.1, tín hiệu vào được giả định là:

$$x_{vào}(t) = A_c \cos[2\pi f_c t + \phi(t)] \quad (6.1)$$

Và tín hiệu tại đầu ra của bộ dao động được điều khiển bằng điện áp (VCO) được giả định có dạng:

$$x_{vco}(t) = -A_v \sin[2\pi f_c t + \theta(t)] \quad (6.2)$$

Vai trò cơ bản của PLL là đồng bộ pha của VCO với pha của tín hiệu vào sao cho lỗi pha $\phi(t) - \theta(t)$ là nhỏ nhất có thể. Ta sẽ làm rõ, cách thực hiện vấn đề này trong phần sau.



Hình 6.1: Sơ đồ khối vòng khóa pha PLL

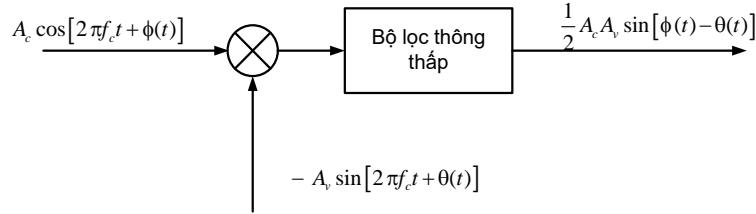
PLL thường hoạt động ở một trong hai chế độ, *bắt* và *bám*. Trong chế độ *bắt*, PLL thử để đồng bộ tần số và pha của tín hiệu ra của VCO với tín hiệu vào. Khi này, các lỗi pha có thể rất lớn, khi này rõ ràng hoạt động của PLL là phi tuyến và cần phải phân tích mô hình phi tuyến. Phân tích các mô hình *phi tuyến* là rất khó và thường cần đến mô phỏng. Tuy nhiên trong chế độ *bám*, lỗi pha thường nhỏ trong những khoảng thời gian dài, và phân tích sử dụng mô hình tuyến tính đơn giản thường cho các kết quả thỏa đáng mà không cần đến mô phỏng. Ta sẽ làm rõ, các tham số vòng tiêu chuẩn được định nghĩa theo mô hình *tuyến tính*, vì vậy cần phải quan tâm cả mô hình tuyến tính và phi tuyến. *Trọng tâm của các mô phỏng được triển khai trong chương này sẽ là tính cách phi tuyến của hệ thống.*

6.2.1. Các mô hình vòng khóa pha

Bước 1: Mô hình hóa bộ tách pha

Các đặc điểm của bộ tách pha phần lớn xác định những đặc điểm hoạt động của PLL. Tồn tại nhiều loại bộ tách pha, việc chọn mô hình bộ tách pha là phụ thuộc vào ứng dụng. Mô hình bộ tách pha thông dụng nhất là bộ tách pha hình **sin** (là bộ tách pha mà đầu ra của nó tỷ lệ với **sin** của lỗi pha). Bộ tách pha hình sin gồm một bộ nhân và một bộ lọc thông thấp như được

minh họa ở hình 6.2. Chức năng duy nhất của bộ lọc thông thấp là loại bỏ hài thứ cấp của tần số sóng mang ở đầu ra bộ nhân. Ta sẽ thấy rằng, trong các ứng dụng thực tế thì bộ lọc thông thấp là không cần thiết.



Hình 6.2: Mô hình bộ tách pha hình sin

Với mô hình bộ tách pha hình 6.2, đầu ra bộ tách pha là:

$$e_d(t) = \frac{1}{2} A_c A_v \sin[\phi(t) - \theta(t)] \quad (6.3)$$

Trong đó $\phi(t) - \theta(t)$ là lỗi pha. Phần sau sẽ ký hiệu lỗi pha là $\psi(t)$, nhưng ở đây tốt hơn là biểu diễn theo pha đầu vào $\phi(t)$ và pha đầu ra bộ VCO $\theta(t)$. Vì ta muốn pha VCO là một **bộ ước tính pha đầu vào** vì vậy hoạt động đúng của PLL là phải điều khiển lỗi pha về 0. Lỗi pha ở trạng thái ổn định có thể bằng 0 hoặc khác 0 phụ thuộc vào các đặc điểm tín hiệu vào và bộ lọc vòng.

Lưu ý rằng đầu vào PLL và đầu ra VCO là vuông pha nhau khi $\phi(t) = \theta(t)$. Điều này cần thiết nếu đầu ra bộ tách pha là *hàm lẻ* của lỗi pha. Dễ dàng thấy rằng nếu các hàm \cos được sử dụng trong cả (6.1) và (6.2) với đối số không thay đổi thì đầu ra bộ tách pha sẽ tỷ lệ nghịch với $\cos[\phi(t) - \theta(t)]$ (là *hàm chẵn* của lỗi pha). Tất nhiên điều này tạo ra tình huống không mong muốn là không thể phân biệt được các lỗi pha âm với các lỗi pha dương.

Sau khi nhân với độ lợi bộ khuếch đại vòng μ , đầu ra bộ tách pha $e_d(t)$ được lọc bởi bộ lọc vòng có hàm truyền đạt $F(s)$ và đáp ứng xung kim đơn vị $f(t)$. Vì vậy đầu vào VCO được cho bởi:

$$e_{vco}(t) = \int_{-\infty}^{\infty} \mu e_d(\lambda) f(t-\lambda) d\lambda \quad (6.4)$$

Chỉ là *tích chập* của đáp ứng xung kim bộ lọc vòng với đầu vào bộ lọc vòng. Với bộ tách pha hình **sin** thì:

$$e_{vco}(t) = \int_{-\infty}^{\infty} \frac{1}{2} \mu A_c A_v \sin[\phi(\lambda) - \theta(\lambda)] f(t-\lambda) d\lambda \quad (6.5)$$

Bước 2: Thiết lập quan hệ độ lệch pha VCO $\theta(t)$ với đầu vào VCO

Theo định nghĩa, độ lệch tần của đầu ra VCO là tỷ lệ nghịch với tín hiệu đầu vào VCO sao cho:

$$\frac{d\theta}{dt} = 2\pi K_d \cdot e_{vco}(t) \quad (6.6)$$

Trong đó K_d là hằng số VCO và có đơn vị là Hz/v. Giải phương trình để tìm $\theta(t)$ ta được:

$$\theta(t) = 2\pi K_d \int_{-\infty}^t e_{vco}(\lambda) d\lambda \quad (6.7)$$

Thay $e_{vco}(t)$ từ (6.5) vào (6.7) ta được:

$$\theta(t) = G \cdot \int_{-\infty}^t \int_{-\infty}^{\infty} \sin[\phi(\lambda) - \theta(\lambda)] \cdot f(\tau - \lambda) d\lambda d\tau \quad (6.8)$$

Trong đó độ lợi vòng G được định nghĩa là:

$$G = \pi \cdot K_d \cdot \mu \cdot A_c \cdot A_v \quad (6.9)$$

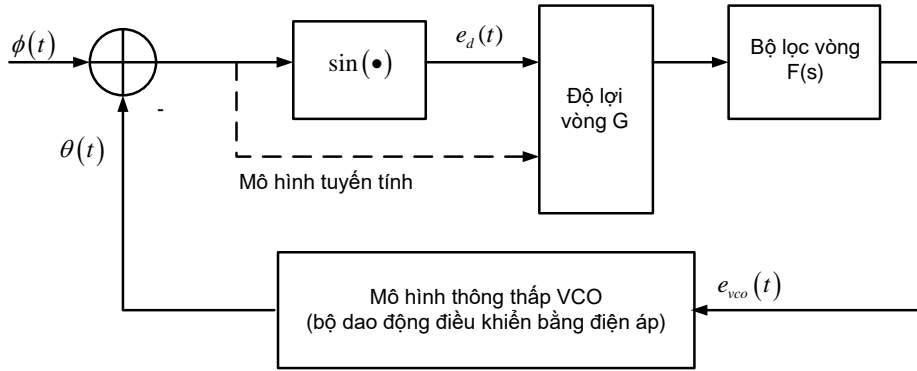
Phương trình tích phân phi tuyến (6.8) thiết lập quan hệ giữa độ lệch pha của đầu vào $\phi(t)$ với độ lệch pha $\theta(t)$ của VCO. Cần lưu ý rằng, đáp ứng xung kim của bộ lọc vòng $f(t)$ vẫn là bất kỳ.

Khi triển khai mô phỏng hệ thống, nên cẩn thận kết hợp các tham số như ta đã làm trong (6.9). Kết hợp các số hạng là một bước hợp lý nếu mục đích của mô phỏng là xác định các đặc tính vào/ra của hệ thống hoặc các đặc tính của toàn bộ hệ thống, như thời gian cần thiết để PLL đạt được khóa pha. Mặt khác, nếu mục đích của mô phỏng là để kiểm tra sự xuất hiện dạng sóng tại đầu vào/đầu ra các khối chức năng trong hệ thống, thì các tham số định nghĩa cho G không được nhóm lại với nhau. Nếu chúng được nhóm lại với nhau, thì dạng sóng có mặt tại các điểm khác nhau trong hệ thống sẽ không thể lấy tỉ lệ chính xác được hoặc thậm chí không thể nhận dạng.

6.2.2. Mô hình pha phi tuyến

Từ biểu thức (6.8) thấy rõ quan hệ giữa $\theta(t)$ và $\phi(t)$ là không phụ thuộc vào tần số sóng mang f_c vì vậy tần số sóng mang không được xét trong mô hình mô phỏng. Theo đó, ta tìm kiếm một mô hình thiết lập quan hệ giữa $\theta(t)$ và $\phi(t)$ mà không cần phải xét tần số sóng mang. Mô hình này được cho ở hình 6.3 là mô hình pha phi tuyến của PLL. Đây là **mô hình pha phi tuyến** vì hai lý do: (i) là mô hình *phi tuyến* (vì tính phi tuyến của hàm **sin**); (ii) là mô hình *pha* (vì nó thiết lập quan hệ giữa độ lệch pha đầu vào và độ lệch pha VCO chứ không phải là thiết lập quan hệ giữa tín hiệu đầu ra VCO và đầu vào vòng thực tế như được biểu diễn bởi (6.1) và (6.2)). Cần lưu ý rằng, đưa vào mô hình được minh họa trong hình 6.1 là tín hiệu thông băng thực tế, trong khi đó khi đưa vào mô hình pha phi tuyến được cho ở hình 6.3 là độ

lệch pha của tín hiệu băng thông đầu vào. Nếu độ lệch pha của tín hiệu đầu vào $\phi(t)$, tần số sóng mang và biên độ tín hiệu được biết, thì (6.1) được xác định hoàn toàn. Tương tự, nếu độ lệch pha $\theta(t)$ của VCO, tần số sóng mang và biên độ được biết trước, thì đầu ra VCO được biểu diễn bởi (6.2) cũng được xác định hoàn toàn. Vì vậy, mô hình pha phi tuyến biểu diễn quan hệ giữa pha đầu vào và pha đầu ra VCO là đại lượng quan trọng. Trong mô phỏng, mô hình pha phi tuyến phải chịu số bị chia bỏ sung. Vì độ lệch pha đầu ra VCO và đầu vào vòng là các tín hiệu thông thấp, nên chúng được lấy mẫu tại tốc độ lấy mẫu thấp hơn nhiều so các tín hiệu thông băng được biểu diễn bởi (6.1) và (6.2).



Hình 6.3: Mô hình pha PLL phi tuyến
(mô hình tuyến tính được thể hiện bằng đường nét đứt)

Tại đây, dùng bộ lọc thông thấp để khử hài thứ cấp (do bộ nhân trong mô hình tách pha gây ra). Bộ lọc này chỉ là một phần của mô hình khái niệm, như đã được đề cập, nó không cần xuất hiện trong thiết bị vật lý. Dễ dàng thấy rằng, bộ lọc này có thể được loại bỏ. Phương trình (6.7) cho thấy mô hình VCO chỉ là bộ tích phân. Vì bộ tích phân là bộ lọc thông thấp có độ lợi vô hạn tại $f = 0$ và độ lợi đơn vị tại $f = 1/(2\pi)$ Hz, nên VCO sẽ ngăn chặn hài thứ cấp của tần số sóng mang truyền lan quanh vòng và xuất hiện tại đầu ra VCO.

Ví dụ 6.1: Xét kỹ thuật tổng quát để mô hình hóa bộ tách pha. Mặc dù bộ tách pha hình **sin** được giả định trong suốt chương này, nhưng nó dễ dàng để mô hình hóa bộ tách pha có đặc tính bất kỳ. Kỹ thuật tổng quát để biểu diễn hàm liên hệ đầu ra bộ tách pha với đầu vào bộ tách pha là chuỗi Fourier:

$$e_d(t) = \sum_{k=1}^N c_k \sin[k\psi(t)] \quad (6.10)$$

Trong đó c_k biểu diễn các hệ số Fourier của đặc tính bộ tách pha, $\psi(t) = \phi(t) - \theta(t)$ là lỗi pha. Có thể mô hình hóa đặc tính bộ tách pha cho trước với độ chính xác cần thiết bất kỳ bằng cách thay đổi giá trị của N trong khai triển chuỗi. Cần lưu ý đến các thành phần làm cho $e_d(t)$ là hàm lẻ của lỗi pha. Ví dụ: bộ tách pha hình **sin** được định nghĩa trong hình 6.2 được biểu diễn bởi:

$$c_k = \begin{cases} \frac{1}{2} A_c A_v, & k = 1 \\ & , N \geq 1 \\ 0 & , \text{nếu khác} \end{cases} \quad (6.11)$$

Mã chương trình Matlab cho bộ tách pha được mô tả bởi (6.10) được viết dễ dàng ở dạng véc tơ. Véc tơ B được định nghĩa là:

$$\mathbf{B} = [1 \quad 2 \quad 3 \quad \dots \quad N]$$

Và véc tơ hệ số Fourier là:

$$\mathbf{C} = [c_1 \quad c_2 \quad c_3 \quad \dots \quad c_N]$$

Giả sử p_{din} và p_{dout} là đầu vào/ra của mô hình bộ tách pha tương ứng, thì đầu ra bộ tách pha có thể được viết là:

$$p_{\text{dout}} = (\sin(\mathbf{B} * p_{\text{din}})) * \mathbf{C}'$$

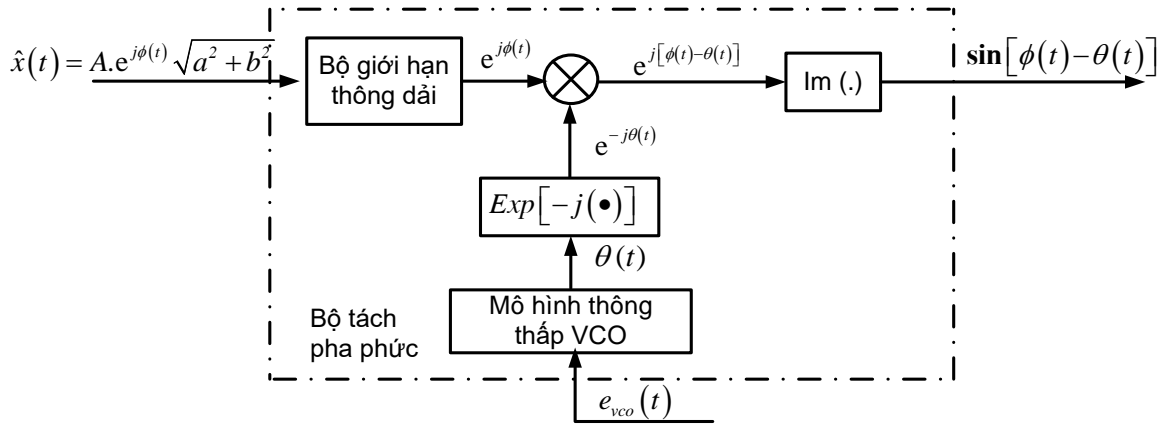
Trong đó \mathbf{C}' là chuyển vị của \mathbf{C} . Vì \mathbf{B} và \mathbf{C} hoàn toàn được xác định bởi mô hình bộ tách pha, nên chúng là cố định và ta nên đặt chúng ở ngoài vòng mô phỏng.

Cũng sẽ thấy rõ rằng, phương pháp tổng quát nhất để mô phỏng hệ thống là sử dụng kỹ thuật Monte Carlo, nó cần có thời gian mô phỏng lâu. Vì thời gian mô phỏng theo kỹ thuật Monte Carlo lâu nên cần có các thuật toán hiệu quả nhất có thể. Vì vậy công thức véc tơ được dùng tính toán p_{dout} nhằm tránh hoạt động vòng thường được sử dụng để ước lượng các tổng.

6.2.3. Mô hình phi tuyến với đầu vào phức

Các vòng khóa pha thường được mô hình hóa sao cho đầu vào vòng là độ lệch pha $\phi(t)$. Khi PLL được dùng trong hệ thống, ta thường mong muốn triển khai mô hình trong đó đầu vào vòng là tín hiệu thông thấp đường bao phức $Ae^{j\phi(t)}$ biểu diễn cho (6.1). Một mô hình như vậy được minh họa ở hình 6.4 cho bộ tách pha hình **sin**. Các bộ tách pha có các đặc điểm khác dễ dàng được rút ra.

Như được minh họa trong hình 6.4, tín hiệu đầu vào ở dạng đường bao phức, trước hết nó được đi qua bộ giới hạn thông dải đã được đề cập trong ví dụ 4.10 chương 4 (ở đây ta giả sử rằng tham số B trong ví dụ 4.10 được đặt bằng 1). Hoạt động của mô hình bộ tách pha phức được sáng tỏ từ các biểu thức trong hình 6.4. Lưu ý rằng, các đầu vào bộ nhân được biểu diễn ở dạng số mũ phức và đầu ra bộ lấy phần ảnh của đầu ra bộ nhân cho ta $\sin[\phi(t) - \theta(t)]$.



Hình 6.4: Mô hình bộ tách pha thông thấp phức

6.2.4. Mô hình tuyến tính và hàm truyền đạt vòng

Nếu lỗi pha là nhỏ sao cho có thể lấy xấp xỉ tuyến tính sau:

$$\sin[\phi(t) - \theta(t)] \approx \phi(t) - \theta(t) \quad (6.12)$$

Thì phương trình vòng (6.8) trở thành:

$$\theta(t) = G \int_{-\infty}^t \int_{-\infty}^{\lambda} [\phi(\lambda) - \theta(\lambda)] \cdot f(\tau - \lambda) d\lambda d\tau \quad (6.13)$$

Kết quả này dẫn đến **mô hình pha tuyến tính** của PLL, nó giống với mô hình pha phi tuyến ngoại trừ tính phi tuyến hình **sin** được loại bỏ (đường nét đứt trong hình 6.3).

Lấy biến đổi Laplace (6.13), thừa nhận rằng phép tích phân trong miền thời gian tương đương với phép chia cho s trong miền tần số và phép chập vòng trong miền thời gian tương đương với phép nhân trong miền tần số, theo đó ta được:

$$\Theta(s) = G \cdot [\Phi(s) - \Theta(s)] \cdot \frac{F(s)}{s} \quad (6.14)$$

Vì vậy, hàm truyền đạt $H(s)$ liên hệ giữa pha VCO với pha đầu vào được cho bởi:

$$H(s) = \frac{\Theta(s)}{\Phi(s)} = \frac{G \cdot F(s)}{s + G \cdot F(s)} \quad (6.15)$$

Cần phải lưu ý rằng, hàm truyền đạt (6.15) là dựa trên giả định tuyến tính, ở dạng tổng quát thì không còn đúng nữa. Nói một cách chính xác, không tồn tại các hàm truyền đạt cho các hệ thống phi tuyến.

Mặc dù mô phỏng thường được dùng để nghiên cứu tính cách phi tuyến của PLL, nhưng các mô hình tuyến tính vẫn hữu hiệu trong một vài mục đích bởi lẽ: Thứ nhất, các tham số vòng hầu như luôn được định nghĩa theo mô hình tuyến tính. Thứ hai, dễ dàng phân tích giải tích dựa trên mô hình tuyến tính và thường giúp kiểm tra tính đúng đắn các kết quả mô phỏng.

Thứ ba, thường dùng các mô hình tuyến tính để nghiên cứu tính cách *bám* khi đó tín hiệu đang được bám đang thay đổi tại tốc độ thấp hơn độ rộng băng thông của vòng.

6.3. Vòng bậc một và bậc hai

Vì tính *bắt* và *bám* của PLL phụ thuộc nhiều vào *bậc* của vòng, nên ta xét hai lựa chọn cho hàm truyền đạt bộ lọc vòng. Bậc thực hiện PLL là bằng số điểm cực trong hàm truyền đạt $H(s)$ như được cho trong biểu thức (6.15). Theo đó, bậc thực hiện PLL lớn hơn số điểm cực của $F(s)$ là 1. Tại đây ta xét ngắn gọn PLL bậc nhất (không có bộ lọc vòng) và PLL bậc hai (có bộ lọc vòng bậc nhất). Dù PLL bậc nhất không có các đặc tính hoạt động mong muốn, nhưng ta sẽ sử dụng PLL bậc nhất để minh họa cơ chế vòng nào đạt được khóa pha.

6.3.1. PLL bậc một

Đối với PLL bậc một $F(s) = 1$ sao cho:

$$f(t) = \delta(t) \quad (6.16)$$

Thay vào (6.8) ta được:

$$\theta(t) = G \int_{-\infty}^t \int_{-\infty}^{\infty} \sin[\phi(\lambda) - \theta(\lambda)] \cdot \delta(\tau - \lambda) d\lambda d\tau \quad (6.17)$$

Lấy tích phân theo biến λ dùng tính chất dịch của hàm xung kim (delta) ta được:

$$\theta(t) = G \int_{-\infty}^t \sin[\phi(\tau) - \theta(\tau)] d\tau \quad (6.18)$$

Lấy vi phân theo t nhận được phương trình vi phân:

$$\frac{d\theta}{dt} = G \sin[\phi(t) - \theta(t)] \quad (6.19)$$

Viết (6.19) theo lỗi pha $\psi(t) = \phi(t) - \theta(t)$ cho ta phương trình vi phân liên hệ giữa lỗi pha và pha đầu vào. Phương trình là:

$$\frac{d\phi}{dt} - \frac{d\psi}{dt} = G \cdot \sin \psi(t) \quad (6.20)$$

Tại đây ta xác định lỗi pha đối với bước nhảy tần số đầu vào.

Để nghiên cứu đáp ứng của PLL bậc nhất đối với bước nhảy tần số f_{Δ} Hz tại thời điểm t_0 ta đặt:

$$\frac{d\phi}{dt} = 2\pi f_{\Delta} u(t - t_0) \quad (6.21)$$

Sao cho (6.20) với $t > t_0$ trở thành:

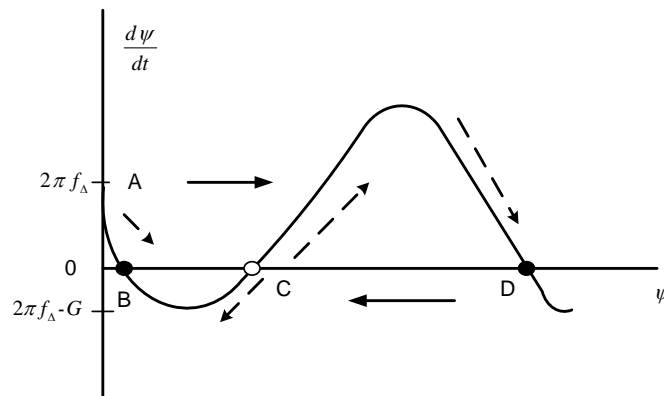
$$\frac{d\psi}{dt} = 2\pi f_{\Delta} - G \cdot \sin \psi(t) \quad (6.22)$$

Cho ta quan hệ giữa lỗi tần số và lỗi pha khi $t > t_0$.

Phương trình (6.22) được minh họa trong hình 6.5 và được gọi là *phương trình mặt phẳng pha* (gọi đơn giản là mặt phẳng pha) mô tả tính cách *động* của hệ thống. Mặt phẳng pha có một số tính chất quan trọng, hiểu biết chúng sẽ hiểu sâu sắc cách cũng như điều kiện để vòng đạt được khóa. Lưu ý rằng, tại mọi thời điểm quan hệ giữa lỗi pha và lỗi tần số phải thỏa mãn (6.22). Những điểm phụ thuộc thời gian này được coi là những điểm hoạt động. Trong mặt phẳng pha nửa trên, điểm đang hoạt động di chuyển từ trái sang phải và trong mặt phẳng pha nửa dưới, điểm đang hoạt động di chuyển từ phải sang trái. Điều này là dễ dàng thấy được. Trước hết ta đặt:

$$\frac{d\psi}{dt} \approx \frac{\Delta\psi}{\Delta t} \quad (6.23)$$

Trong đó $\Delta\psi$ và Δt biểu thị cho gia tăng nhỏ về lỗi pha và thời gian. Rõ ràng $\Delta t > 0$ với mọi t vì thời gian luôn tăng. Theo đó, $d\psi > 0$ trong mặt phẳng pha nửa trên và $d\psi < 0$ trong mặt phẳng pha nửa dưới. Vì vậy, lỗi pha tăng (di chuyển từ trái sang phải) trong mặt phẳng pha nửa trên và lỗi pha giảm (di chuyển từ phải sang trái) trong mặt phẳng pha nửa dưới. Điều này được minh họa bởi mũi tên liền trong hình 6.5. Điểm hoạt động ổn định (dừng) có thể chỉ nằm trên biên giới giữa hai mặt phẳng pha này. Tất nhiên, điều này chứng tỏ rằng lỗi pha là hằng số (tương đương lỗi tần số bằng 0). Điểm hoạt động là ổn định, nếu sau dao động nhỏ điểm hoạt động trở về vị trí ban đầu của nó. Nếu dao động nhỏ dẫn đến điểm hoạt động di chuyển tới vị trí mới thì điểm hoạt động ban đầu được gọi là không ổn định. Vì vậy, các điểm B và D trong hình 6.5 là các điểm hoạt động ổn định và điểm C là điểm hoạt động không ổn định. Sự di chuyển của các điểm hoạt động được chỉ ra bởi các đường nét đứt trong hình 6.5.



Hình 6.5: Mặt phẳng pha bậc một

Từ (6.22) cho thấy, nếu $2\pi f_\Delta < G$ thì điểm hoạt động ở trạng thái ổn định là điểm ổn định B với giả thiết điểm hoạt động khởi đầu do bước nhảy tần số là A. Tại điểm này lỗi tần số bằng 0 và lỗi pha ở trạng thái ổn định là nghiệm của phương trình (6.22) với $d\psi/dt = 0$. Điều này cho ta lỗi pha ở trạng thái ổn định là:

$$\psi_{ss} = \sin^{-1} \left(\frac{2\pi f_\Delta}{G} \right) \quad (6.24)$$

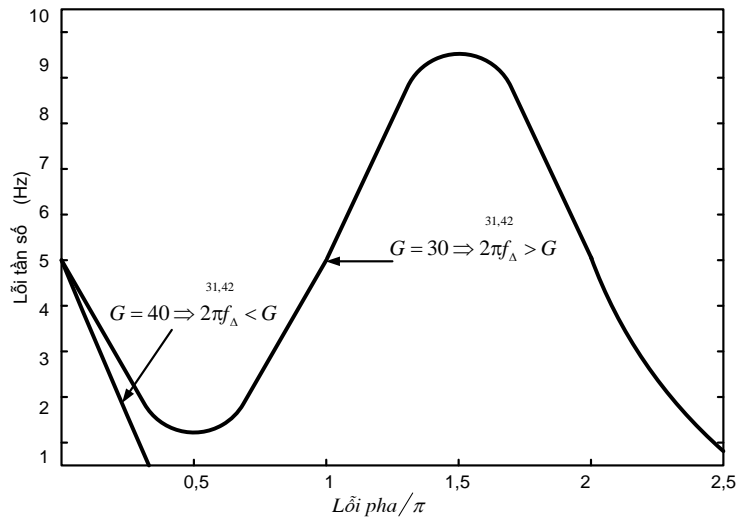
Như sự quan sát cuối cùng về mặt phẳng pha, lưu ý rằng nếu $2\pi f_{\Delta} > G$ thì không có lời giải cho (6.22) để lỗi tần số bằng 0 $d\psi/dt$ và điểm hoạt động sẽ di chuyển sang bên phải $\forall t$ khi $f_{\Delta} > 0$ và sẽ di chuyển sang trái $\forall t$ khi $f_{\Delta} < 0$. Vì vậy độ lợi vòng G trở thành *khoảng khóa* cho vòng bậc nhất. Lưu ý từ (6.15) rằng G cũng là độ rộng băng thông vòng (rad/s) đối với PLL bậc nhất.

Ví dụ 6.2: Như một ví dụ đơn giản, giả sử: (i) $f_{\Delta} = 5$ sao cho $2\pi f_{\Delta} = 31,42$; (ii) độ lợi vòng lấy hai giá trị là $G = 30$ và $G = 40$.

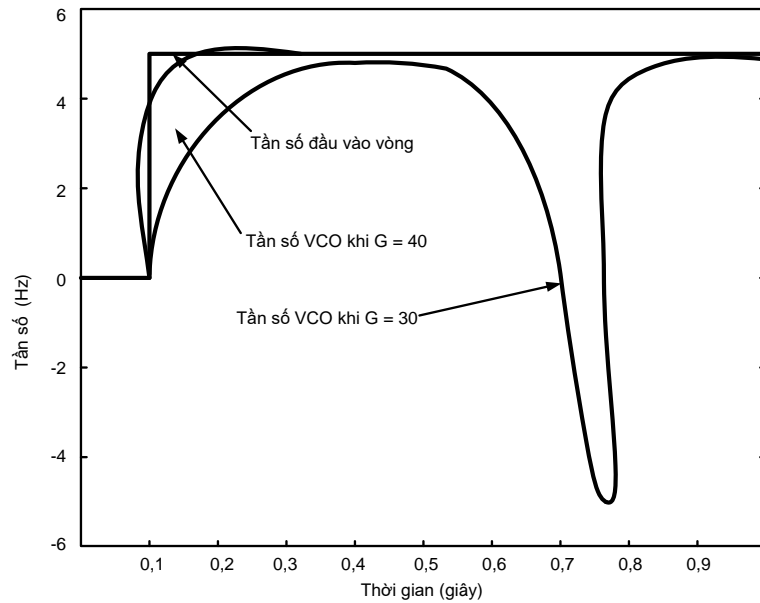
Kết quả, các mặt phẳng pha được cho ở hình 6.6, nó cho thấy: (i) khi $f_{\Delta} = 5$; $G = 40 \Rightarrow 2\pi f_{\Delta} < G$ thì lỗi tần số giảm đều đều tới 0; (ii) Khi $f_{\Delta} = 5$; $G = 30 \Rightarrow 2\pi f_{\Delta} > G$ thì (6.22) không có lời giải để lỗi tần số bằng 0. Đối với trường hợp này, không đạt được khóa pha và hệ thống sẽ mãi mãi dao động.

Kết quả, mặt phẳng pha được tạo ra rõ hơn ở hình 6.7, nó cho thấy độ lệch tần số đầu vào $d\phi/dt$ và độ lệch tần số VCO $d\theta/dt$ đối với $G = 30$ và $G = 40$. Kết quả, độ lệch tần số đầu vào, độ lệch tần số VCO được cho ở hình 6.7 khi $G = 30$ và $G = 40$. Lưu ý rằng, với $G = 40$ thì vòng đạt được khóa pha, trong khi đó $G = 30$ thì không đạt được khóa pha và độ lệch pha VCO dao động mãi mãi.

Ta đã thấy rằng, cả *khoảng khóa* của PLL bậc nhất và *độ rộng băng thông vòng* đều được xác định bởi tham số G . Với hầu hết các ứng dụng, đều mong muốn khoảng khóa lớn và băng thông vòng nhỏ. Điều này không thể thực hiện được bằng vòng bậc nhất, vì vậy PLL bậc nhất thường không thực tế. Vòng bậc hai là hệ thống thực tế cho nhiều ứng dụng và sẽ nhận được thu hút nghiên cứu mô phỏng.



Hình 6.6: Mặt phẳng pha đối với PLL bậc một ($G = 30$ và $G = 40$)



Hình 6.7: Tần số vào và tần số VCO khi $G = 30$ và $G = 40$

6.3.2. PLL bậc hai

PLL bậc nhất có khoảng khóa bị giới hạn. Ngoài ra, vì vòng chỉ có một tham số duy nhất, nên khả năng điều chỉnh các tham số vòng để thỏa mãn một tập các đặc tính kỹ thuật cho trước bị hạn chế nhiều. Các đặc tính hoạt động được cải thiện và những khả năng thiết kế đạt được bằng cách thay đổi bộ lọc vòng sao cho nhận được PLL bậc hai.

Dạng tổng quát của bộ lọc vòng đối với PLL bậc hai là:

$$F(s) = \frac{s+a}{s+\lambda a} \quad (6.25)$$

PLL bậc hai hoàn hảo được định nghĩa bởi $\lambda = 0$, nó thể hiện cho bộ lọc vòng chứa tích phân lý tưởng (điểm cực tại $s = 0$). Trong các ứng dụng điển hình thì $\lambda \ll 1$. Thay hàm truyền đạt bộ lọc vòng vào (6.15), ta có:

$$H(s) = \frac{G.(s+a)}{s^2 + (G+\lambda a).s + G.a} \quad (6.26)$$

đối với hàm truyền đạt mô hình tuyến tính vòng.

Các hệ thống bậc hai tuyến tính thường được tham số hóa theo hệ số tắt dần ζ và tần số tự nhiên ω_n của hệ thống. Mẫu số của hàm truyền đạt được gọi là đa thức đặc trưng, với các hệ thống bậc hai được biểu diễn ở dạng chuẩn là:

$$s^2 + 2\zeta\omega_n s + \omega_n^2$$

Cân bằng đa thức đặc trưng trong (6.26) theo dạng chuẩn ta được:

$$s^2 + (G+\lambda a)s + Ga = s^2 + 2\zeta\omega_n s + \omega_n^2 \quad (6.27)$$

Trong các ứng dụng điển hình, PLL được thiết kế đối với hệ số tắt dần và tần số tự nhiên cho trước bằng cách định rõ ζ và ω_n . Các tham số vật lý cần thiết (trong trường hợp này là G và a) được xác định sao cho đạt được các giá trị thiết kế ζ và ω_n . Cân bằng các thành phần có cùng số mũ của s trong (6.27) ta được:

$$2\zeta\omega_n = G + \lambda a \quad (6.28)$$

$$\omega_n^2 = Ga \quad (6.29)$$

Giả sử λ là hằng số đã biết, ta có thể giải phương trình (6.29) để tìm a và thay kết quả vào (6.28), ta có:

$$G^2 - 2\zeta\omega_n G + \lambda\omega_n^2 = 0 \quad (6.30)$$

Tìm được G :

$$G = \zeta\omega_n + \omega_n\sqrt{\zeta^2 - \lambda} \quad (6.31)$$

Kết quả này cùng với (6.29) cho ta:

$$a = \frac{\omega_n}{\zeta + \sqrt{\zeta^2 - \lambda}} \quad (6.32)$$

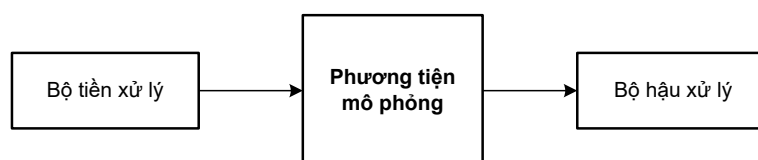
Lưu ý rằng vì a là tham số giá trị thực nên λ phải nhỏ hơn ζ^2 . Các giá trị điển hình của ζ^2 nằm trong lân cận $\frac{1}{2}$ ($\zeta = 1/\sqrt{2}$ là lựa chọn chung) và $\lambda \ll 1$. Lưu ý rằng đối với PLL bậc hai hoàn hảo thì $G = 2\zeta\omega_n$, $a = \frac{\omega_n}{2\zeta}$.

6.4. Mô phỏng vòng khóa pha PLL

Tại đây, ta đang ở giai đoạn mô phỏng PLL bậc hai và sử dụng kiến thức rất cơ bản về lý thuyết PLL để kiểm tra sự đúng đắn của các kết quả. Tuy nhiên, trước hết ta xét kiến trúc đơn giản cho chương trình mô phỏng toàn diện. Sau đó triển khai mô hình mô phỏng và thực hiện mô phỏng. Cuối cùng, đề cập ngắn gọn các nguồn gây lỗi trong mô phỏng.

6.4.1. Kiến trúc mô phỏng

Thường là hữu ích khi chia phần mềm mô phỏng thành một vài chương trình riêng biệt như ở hình 6.8 gồm 3 phần tử chương trình mô phỏng, được định rõ là bộ tiền xử lý, phương tiện mô phỏng và bộ hậu xử lý. Ba chương trình này thực hiện 3 chức năng riêng biệt và có thể được triển khai trong 3 ngôn ngữ khác nhau. Phân chia mô phỏng theo cách này cho phép dễ dàng triển khai mã chương trình mô phỏng, dễ hiểu và dễ quản lý. Nó cũng hiệu quả vì các đoạn mã thường tái sử dụng trong các ứng dụng khác nhau. Điều này đặc biệt đúng đối với bộ hậu xử lý.



Hình 6.8: Kiến trúc mô phỏng

Mục đích của bộ tiền xử lý là định rõ tất cả các tham số cần thiết để định nghĩa hệ thống (các tham số hệ thống) và thiết lập các tham số nhằm quản lý việc thực hiện mô phỏng (các tham số bên trong). Ví dụ các tham số hệ thống như: bậc bộ lọc, loại bộ lọc, băng thông, khuếch đại, tỷ lệ mã, tỷ số trải phổ, tần số sóng mang, tốc độ bit và SNR. Các tham số bên trong gồm tần số lấy mẫu, thời gian ổn định và số mẫu được xử lý. Ngoài ra dữ liệu được qua bộ hậu xử lý để phân tích, hoặc để vẽ, nó phải được lưu giữ bởi mô phỏng, vì vậy cần có các file dữ liệu được xác định bởi bộ hậu xử lý phải. Ví dụ, véc tơ các mẫu dạng sóng, véc tơ thời điểm mẫu (địa chỉ mẫu) được sử dụng trong bộ hậu xử lý để hiệu chỉnh trục thời gian của đồ thị tín hiệu. Một khi mọi thông tin cần thiết được định rõ trong bộ tiền xử lý thì dữ liệu được tạo ra bên trong bộ tiền xử lý thường được ghi vào một file sao cho khả dụng đối với phương tiện mô phỏng và bộ hậu xử lý. Trong các ví dụ Matlab, dữ liệu bộ tiền xử lý thường được viết lên không gian làm việc Matlab.

Phương tiện mô phỏng sẽ đọc dữ liệu trong các file (được tạo bởi bộ tiền xử lý) và thực hiện mô phỏng. Tất nhiên, mục đích của mô phỏng là tạo dữ liệu để giám sát bằng cách sử dụng bộ hậu xử lý. Điều này có thể là lấy dạng các số (SNR, BER, độ lợi mã,...), hoặc véc tơ các giá trị mẫu để xử lý bổ sung bởi bộ hậu xử lý. Thông tin được tạo bởi phương tiện mô phỏng được lưu trong các file, và đi qua bộ hậu xử lý. Trong trường hợp Matlab thường chuyển dữ liệu được tạo bởi phương tiện mô phỏng vào không gian làm việc Matlab.

Bộ hậu xử lý lấy dữ liệu được tạo bởi phương tiện mô phỏng, tạo các sản phẩm mô phỏng cuối cùng cần thiết. Chúng gồm đồ thị dạng sóng, chùm tín hiệu, BER là hàm của E_b/N_0 , mật độ phổ công suất, biểu đồ mất, biểu đồ thống kê... Chi tiết hóa hậu xử lý được xét trong chương 8.

Bộ hậu xử lý cần có mức tín hiệu thoả đáng để hỗ trợ đồ hoạ. Đây là một lý do dùng Matlab cho các ứng dụng trình bày ở đây. Các bộ hậu xử lý hữu hiệu nhất khi chúng có điều khiển menu, tạo ra xử lý tín hiệu khác nhau và các lựa chọn hiển thị. Mô phỏng trong chương này là khá đơn giản như bộ tiền xử lý và bộ hậu xử lý, nhằm giải thích vai trò của mỗi phần tử mô phỏng.

Bộ tiền xử lý, phương tiện mô phỏng và bộ hậu xử lý có thể được viết ở nhiều ngôn ngữ khác nhau. Dự án gần đây của các tác giả sử dụng Visual Basic cho bộ tiền xử lý, C++ cho phương tiện mô phỏng và Matlab cho bộ hậu xử lý. Matlab thường được lựa chọn để triển khai bộ hậu xử lý vì thư viện đồ hoạ phong phú của Matlab.

6.4.2. Mô phỏng

Mô hình mô phỏng là dễ hiểu ngoại trừ có lẽ đối với bộ lọc vòng. Vì hàm truyền đạt bộ lọc vòng như định nghĩa trong (6.25) không phải là hàm quy cách (hàm quy cách là hàm trong đó bậc của đa thức mẫu số lớn hơn bậc của đa thức tử số ít nhất là một), thực hiện chia ta được:

$$F(s) = 1 + \frac{(1-\lambda)a}{s+\lambda a} = 1 + F_I(s) \quad (6.33)$$

Vì vậy, có thể thực thi hóa $F(s)$ là sự kết hợp hai hàm truyền đạt, hàm thứ nhất là một hằng số và hàm thứ hai là $F_I(s)$. Rõ ràng:

$$F_I(s) = \frac{(1-\lambda)a}{s+\lambda a} = \frac{Y_I(s)}{X_I(s)} \quad (6.34)$$

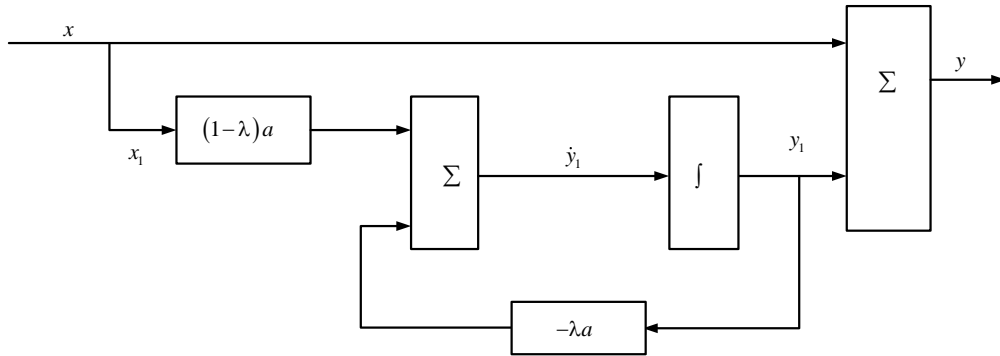
Trong đó $X_I(s)$ và $Y_I(s)$ là biến đổi Laplace đầu vào/ra của bộ lọc con $F_I(s)$. Nhân chéo ta có:

$$sY_I(s) = (1-\lambda)a X_I(s) - \lambda a Y_I(s) \quad (6.35)$$

Trong miền thời gian nó là:

$$\frac{dy_I}{dt} = (1-\lambda)a x_I(t) - \lambda a y_I(t) \quad (6.36)$$

Theo đó, ta có bộ lọc vòng được minh họa trong hình 6.9.



Hình 6.9: Mô hình bộ lọc vòng

Tiếp theo, ta triển khai mã chương trình mô phỏng cho PLL bậc hai, ta triển khai *biểu đồ tiến trình tín hiệu* cho hệ thống và chỉ rõ những điểm trên biểu đồ tiến trình, định nghĩa các tín hiệu. Biểu đồ tiến trình tín hiệu được minh họa trong hình 6.10. Lưu ý rằng mô hình bộ lọc vòng tuân theo hình 6.9 với:

$$a_1 = (1-\lambda)a \quad (6.37)$$

$$a_2 = \lambda a \quad (6.38)$$


```
s6 = s3 + s5;      % Đầu vào bộ tích phân VCO
w1c = s6 + w2c;    % Bộ tích phân VCO (bước 1)
w2c = s6 + w1c;    % Bộ tích phân VCO (bước 2)
phivco = w1c/twofs; % Tạo đầu ra VCO
end
```

Hằng số **twofs** là gấp hai lần tần số lấy mẫu, được cần đến đối với tích phân hình thang và được tính toán bên ngoài vòng mô phỏng.

Mã chương trình mô phỏng đầy đủ cho PLL bậc hai được cho bởi **NVD6_PLLsim.m** ở Phụ lục 6A. Lưu ý rằng, mã được cho trong Phụ lục 6A là khác chút ít với mã được cho ở đây vì các véc tơ được triển khai cho pha đầu vào, lỗi pha và tần số VCO. Những véc tơ này được yêu cầu bởi bộ hậu xử lý để vẽ các dạng sóng. Mã Matlab cho bộ tiền xử lý ở file **NVD6_PLLpre.m** và bộ hậu xử lý được cho ở file **NVD6_PLLpost.m** cùng với các chương trình con kèm theo có trong Phụ lục 6A. Bộ hậu xử lý được thiết kế theo kiểu điều khiển menu. Các bộ hậu xử lý điều khiển menu thường nằm trong các gói mô phỏng.

6.4.3 Kết quả mô phỏng

Giả sử rằng độ lệch tần số đầu vào PLL là bước nhảy đơn vị tại thời điểm t_0 . Vì vậy:

$$\frac{d\phi}{dt} = 2\pi f_{\Delta} \underbrace{u(t - t_0)}_{\text{bước nhảy đơn vị}} \quad (6.39)$$

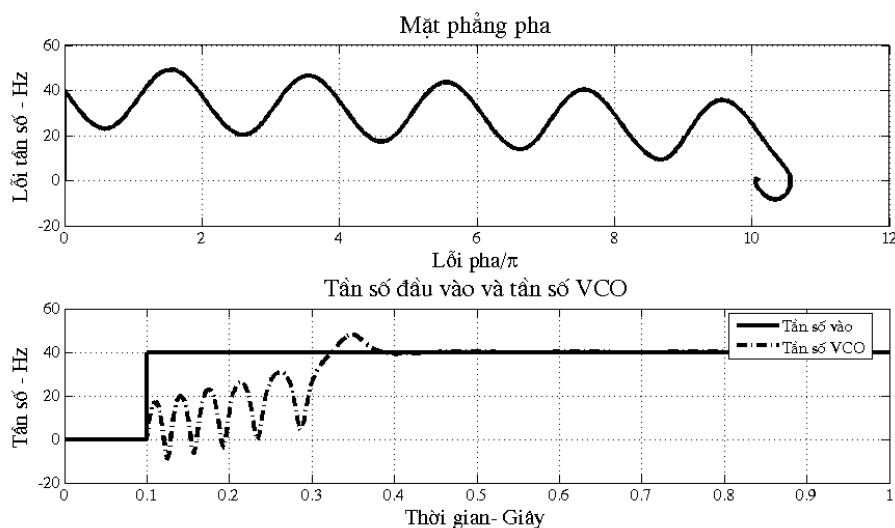
Với các tham số PLL dưới đây:

$$\begin{aligned} \text{Bước nhảy tần số,} \quad f_{\Delta} &= 40 \text{ Hz} \\ \text{Tần số vòng tự nhiên,} \quad \omega_n / 2\pi &= 10 \text{ Hz} \\ \text{Độ dịch điểm cực tương đối,} \quad \lambda &= 0,1 \\ \text{Hệ số suy giảm vòng,} \quad \varsigma &= 1/\sqrt{2} \\ \text{Tần số lấy mẫu,} \quad f_s &= 5000 \text{ Hz} \\ T_{\text{stop}} &= 0,8 \text{ s} \end{aligned} \quad (6.40)$$

Sử dụng bộ tiền xử lý để nhập các tham số này.

Các lựa chọn **menu** trong bộ hậu xử lý **NVD6_post.m** (Phụ lục 6A) cho phép dễ dàng khảo sát một số các mục quan tâm. Chúng gồm tần số đầu vào và tần số VCO, lỗi tần số và mặt phẳng pha. Ở đây ta chỉ minh họa hai đồ thị được tạo bởi bộ hậu xử lý. Mặt phẳng pha được minh họa trong nửa trên của hình 6.11. Tần số đầu vào (bước nhảy đơn vị là 40 Hz tại $t_0 = 0,08$) và tần số VCO (dạng sóng dao động) được minh họa trong nửa dưới của hình 6.11. Cần lưu ý rằng, ta đang xét các mô hình thông thấp. Vì vậy tần số đầu vào và tần số VCO thực ra là độ lệch tần đầu vào và độ lệch tần VCO so với tần số sóng mang chuẩn hóa f_c .

Từ mặt phẳng pha, trượt 5 chu kỳ và lỗi trạng thái ổn định là lớn hơn 10π . Từ hình 6.11 cho thấy bước nhảy tần số đầu vào là 40 Hz. Tuy nhiên, tần số VCO dao động qua 5 chu kỳ và sau đó đạt được khóa pha ở chu kỳ thứ 6. Đặc điểm này gọi là "*trượt tuần hoàn*" và là các đặc trưng của các bộ đồng bộ phi tuyến khi thay đổi tần số đầu vào vượt quá tần số tự nhiên.



Hình 6.11: Các kết quả mô phỏng

6.4.4. Nguồn lỗi trong mô phỏng

Có một số nguồn lỗi xuất hiện trong mô phỏng. Những nguồn lỗi này đã được đề cập trong chương 1 là do: (i) việc ánh xạ thiết bị vật lý sang mô hình phân tích; (ii) ánh xạ mô hình phân tích sang mô hình mô phỏng. Dưới đây ta đề cập ngắn gọn.

Mô hình giải tích

Khi triển khai mô hình phân tích cho thiết bị ta thực hiện một số giả định. Một cách điển hình, những xấp xỉ này bao gồm việc ý tưởng hóa các thành phần vòng và có thể là không hợp lệ nếu muốn có được các mô phỏng thiết bị vật lý với độ chính xác cao. Để các kết quả mô phỏng với độ chính xác cao cần phải đo kiểm để xác định các mô hình thành phần đủ chính xác. Những hạn chế khác có thể cũng đưa các tín hiệu vào trong vòng mà không được tính trong mô hình phân tích cơ bản.

Mô hình mô phỏng

Tồn tại một số nguồn lỗi quan trọng thuộc về quá trình ánh xạ từ mô hình giải tích sang mô hình mô phỏng. Ví dụ, trong khi hệ thống vật lý xử lý các tín hiệu liên tục theo thời gian thì mô phỏng xử lý các mẫu tín hiệu đó. Tất nhiên, lấy mẫu dẫn đến xuất hiện các lỗi, cần tính đến các lỗi lượng tử. Ngoài ra các bộ lọc tương tự trong phần cứng (hoặc mô hình giải tích) phải được thể bởi bộ lọc số trong mô hình mô phỏng. Biết rằng, bộ lọc số không bao giờ có cùng đáp ứng biên độ và pha như bộ lọc tương tự. Các lỗi do chồng phổ phải được xét đến.

Một nguồn lỗi quan trọng xuất hiện trong tất cả các hệ thống có hồi tiếp. Ví dụ, đầu ra của bộ tách pha trong PLL tương tự (phần cứng) được định nghĩa là:

$$e_d(t) = \sin[\phi(t) - \theta(t)] \quad (6.41)$$

Mô hình rời rạc theo thời gian cho đầu ra này sẽ là:

$$e_d(nT) = \sin [\phi(nT) - \theta(nT)] \quad (6.42)$$

Trong đó $\phi(nT)$ là đầu vào "hiện thời" của PLL. Tuy nhiên, không có sẵn độ lệch pha đầu ra $\theta(nT)$ của VCO vì cần phải có $e_d(nT)$ để tính toán nó. "Sự bế tắc" tính toán do tính phụ thuộc lẫn nhau giữa $e_d(nT)$ và $\theta(nT)$ được xử lý bằng cách sử dụng giá trị tính toán độ lệch pha trước đó $\theta((n-1)T)$ của VCO để tính $e_d(nT)$. Nói cách khác, mô hình được định nghĩa bởi (6.42) được thay bằng:

$$e_d(nT) = \sin \left[\phi(nT) - \underbrace{\theta((n-1)T)} \right] \quad (6.43)$$

Theo đó, đã gây ra trễ một mẫu trong vòng hồi tiếp (không có trong mô hình vật lý hoặc trong mô hình phân tích). Nếu kích thước bước mô phỏng nhỏ (tần số lấy mẫu cao) thì trễ một mẫu có thể không ảnh hưởng đến tính chính xác của mô phỏng, nhưng không bỏ qua được đối với các kích thước bước mô phỏng thực tế (tần số lấy mẫu thực tế không đủ lớn). Ảnh hưởng trễ một mẫu quy vào dịch pha tuyến tính (trễ thời gian không đổi) trong hàm truyền đạt vòng hở. Điều này làm giảm độ dự trữ pha của hệ thống và nếu kích thước bước mô phỏng đủ lớn thì có thể đẩy hệ thống vào trạng thái bất ổn định.

Nếu mô phỏng là để có mức chính xác cho trước, thì các mô hình mô phỏng phải dựa trên các mô hình giải tích ít nhất là có cùng mức độ chính xác. Nếu thiết bị vật lý đang được mô hình tỏ ra thay đổi đáng kể thì các mô hình mô phỏng cần phải dựa vào dữ liệu đo (được kiểm định từ dụng cụ hiệu chuẩn chính xác). Các bộ khuếch đại phi tuyến thường rơi vào loại này, là thiết bị chứa các thành phần lão hóa. Ví dụ khác là các mô hình cho các kênh đa đường.

Ở đây ta chỉ đề cập một vài nguồn lỗi, nhưng chúng rất quan trọng, là mô hình giải tích bắt giữ một phần đặc tính (hy vọng đó là các đặc tính quan trọng nhất) của các thiết bị vật lý và chấp nhận mắc lỗi phụ khi chuyển mô hình giải tích sang mô hình mô phỏng, mức độ gây lỗi phụ thuộc vào mô hình (giải tích) hệ thống và phương pháp luận mô phỏng. Người dùng mô phỏng phải nhận ra những nguồn lỗi này và đảm bảo lỗi tích lũy đủ nhỏ.

6.5. Dùng mô phỏng để giải phương trình vi phân

Khi máy tính số chưa được dùng phổ biến, máy tính tương tự cho ta công cụ để giải các phương trình vi phân. Máy tính tương tự đã chứng tỏ hữu hiệu nhất đối với các phương trình thay đổi theo thời gian phi tuyến trong đó không tồn tại các kỹ thuật tìm nghiệm giải tích tổng quát. Vì kỹ thuật máy tính tương tự có thể được áp dụng cho các hệ thống phi tuyến hoặc thay đổi theo thời gian hoặc cả hai, nên các máy tính tương tự đã được sử dụng rộng rãi. Trong khi "nghiệm máy tính tương tự" không có dạng của một phương trình (là dạng tiêu chuẩn để biểu diễn nghiệm cho phương trình vi phân) thì nó cho phép dễ dàng vẽ các dạng sóng xuất hiện tại các thời điểm khác nhau trong hệ thống. Dẫn đến dễ dàng nghiên cứu tham số vì vậy cho phép hiểu biết sâu sắc về các đặc tính hoạt động của các hệ thống phức tạp bằng việc sử dụng máy

tính tương tự. Không lấy làm ngạc nhiên về máy tính tương tự đã triển khai thành một công cụ mô phỏng mạnh mẽ.

Tuy máy tính tương tự là công cụ hữu ích nhưng việc sử dụng nó không hề đơn giản. Thành phần cơ bản của máy tính tương tự là bộ khuếch đại thuật toán. Đặc biệt, những bộ khuếch đại thuật toán đó sử dụng trong suốt những ngày, phải chịu trôi dạt làm việc. Vì vậy thường xuyên phải hiệu chuẩn. Ngoài ra việc thực hiện nhân và chia là khó khăn. Các máy tính tương tự chỉ có thể xử lý các tín hiệu có băng thông hạn chế dẫn đến thường xuyên phải lấy tỉ lệ về tần số và thời gian. Khi máy tính số trở nên phổ cập cùng với tính thân thiện trong sử dụng thì máy tính tương tự không được dùng nữa. Tuy nhiên, nhiều phương pháp luận đã được triển khai cho các máy tính tương tự lại có khả năng ứng dụng để mô phỏng sử dụng các máy tính số. Vì vậy kiến thức cơ bản đã triển khai cho các máy tính tương tự mang lại một tập hợp các kỹ thuật mô phỏng quan trọng có khả năng ứng dụng cho các máy tính số. Ví dụ, một trong các chương trình mô phỏng đầu tiên, CSMP (chương trình mô hình hóa hệ thống liên tục) được triển khai bởi IBM đối với dòng máy tính hệ 360, là mô phỏng máy tính số cơ bản của máy tính tương tự. Khi máy tính cá nhân đã phát triển thành công cụ hữu ích cho các nhà khoa học và các kỹ sư thì các phiên bản dựa trên PC khác nhau của CSMP đã được phát triển và một vài số chúng vẫn còn sử dụng cho tới nay. Một trong những chương trình mô phỏng đầu tiên hướng vào các hệ thống truyền thông, TOPSIM dựa trên CSMP.

Sơ đồ mô phỏng

Bước đầu tiên để giải phương trình vi phân sử dụng kỹ thuật máy tính tương tự để triển khai sơ đồ mô phỏng máy tính tương tự. Quá trình này là dễ hiểu. Ví dụ đơn giản, giả sử hệ thống được định nghĩa bởi phương trình sau:

$$\frac{d^2y}{dt^2} + a\frac{dy}{dt} + by = x(t) \quad (6.44)$$

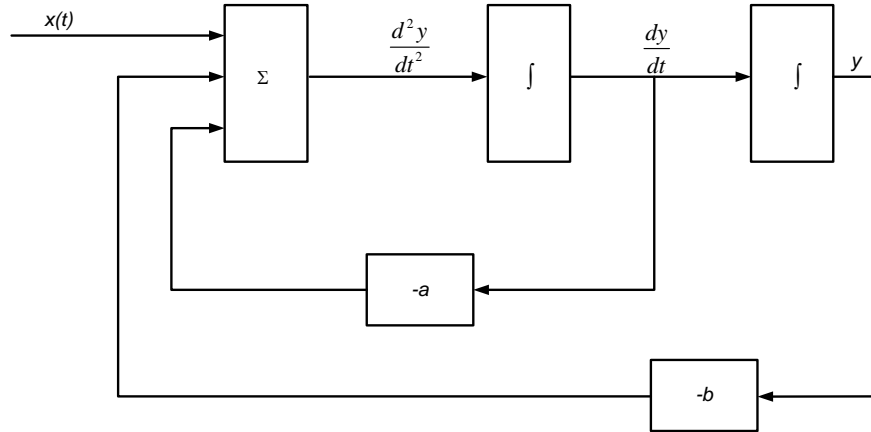
trong đó a và b nhận giá trị hằng số. Giải phương trình vi phân cho $\frac{d^2y}{dt^2}$ được:

$$\frac{d^2y}{dt^2} = x(t) - a\frac{dy}{dt} - by \quad (6.45)$$

Lấy tích phân d^2y/dt^2 để được dy/dt và lấy tích phân dy/dt để được $y(t)$. Sau đó những đại lượng này được nhân, hoặc được thao tác bằng các hằng số phù hợp hoặc các hàm để tạo các số hạng cần thiết nhằm biểu diễn các số hạng khác nhau trong (6.45). Kết hợp chúng lại tạo phương trình vi phân phù hợp và sơ đồ mô phỏng máy tính tương tự cho phương trình vi phân này được cho ở hình 6.12. Trong khi hình 6.12 đã triển khai cho (6.45), là phương trình tuyến tính đơn giản với các hệ số không đổi và có khả năng giải được bằng cách sử dụng nhiều phương pháp khác nhau thì cái hay của kỹ thuật máy tính tương tự là khả năng ứng dụng vào các hệ thống phi tuyến và thay đổi theo thời gian với bậc bất kỳ.

Ví dụ hệ thống vừa phi tuyến vừa thay đổi theo thời gian, giả sử rằng hệ thống được định nghĩa bởi phương trình vi phân sau:

$$\frac{d^3 y}{dt^3} + a(t)y(t)\frac{d^2 y}{dt^2} + b(t)y^2(t)\frac{dy}{dt} + c(t)y(t) = x(t) \quad (6.46)$$

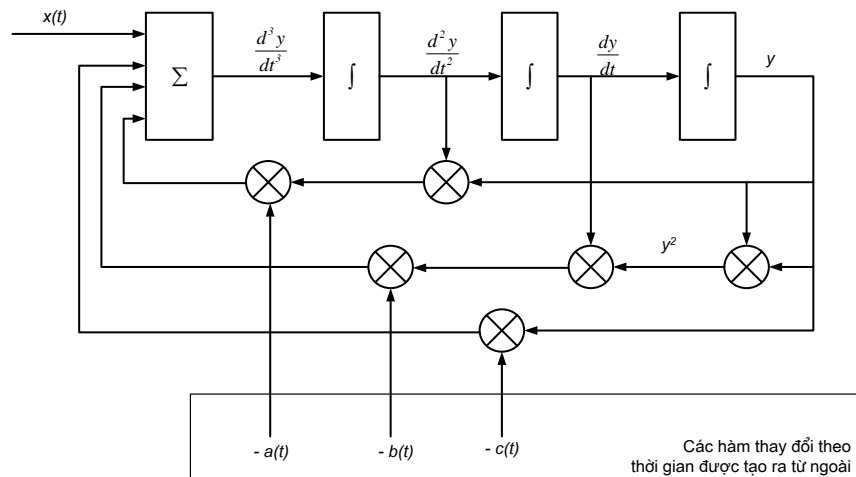


Hình 6.12: Sơ đồ mô phỏng cho hệ thống tuyến tính bất biến theo thời gian bậc hai

Giải tìm d^3y/dt^3 được:

$$\frac{d^3 y}{dt^3} = x(t) - a(t)y(t)\frac{d^2 y}{dt^2} - b(t)y^2(t)\frac{dy}{dt} - c(t)y(t) \quad (6.47)$$

Từ đó có sơ đồ mô phỏng được cho ở hình 6.13. Các hệ số thay đổi theo thời gian $a(t)$, $b(t)$ và $c(t)$ có thể được định rõ bởi các phương trình phù hợp hoặc có thể là các file của dữ liệu đo được tập hợp từ thực nghiệm.



Hình 6.13: Sơ đồ mô phỏng cho hệ thống bậc ba phi tuyến thay đổi theo thời gian

Nhắc lại PLL

Tại đây ta xét mô phỏng PLL sử dụng giải pháp phương trình vi phân.

Trước hết là rút ra phương trình vi phân. Từ hình 6.3 ta có:

$$E_{vco}(s) = G.F(s).E_d(s) \quad (6.48)$$

Vì VCO trong mô hình PLL được minh họa trong hình 6.3 được biểu diễn bởi một bộ tích phân, nên ta có:

$$\Theta(s) = \frac{1}{s} E_{vco}(s) \quad (6.49)$$

Thay (6.48) vào (6.49) và sử dụng (6.25) cho bộ lọc vòng tạo ra:

$$\Theta(s) = \frac{G}{s} \left(\frac{s+a}{s+\lambda a} \right) E_d(s) \quad (6.50)$$

Đơn giản hóa bài toán bằng cách cho rằng vòng bậc hai "lý tưởng" khi $\lambda = 0$, dẫn đến:

$$s^2 \Theta(s) = s.G.E_d(s) + G.a.E_d(s) \quad (6.51)$$

Vì nhân với s là tương đương với vi phân trong miền thời gian nên ta có phương trình vi phân:

$$\frac{d^2 \theta}{dt^2} = G \cdot \frac{d}{dt} e_d(t) + G.a.e_d(t) \quad (6.52)$$

Theo định nghĩa:

$$e_d(t) = \sin \psi(t) \quad (6.53)$$

$$\frac{d^2 \psi}{dt^2} = \frac{d^2 \phi}{dt^2} - \frac{d^2 \theta}{dt^2} \quad (6.54)$$

Phương trình (6.52) được viết là:

$$\frac{d^2 \psi}{dt^2} + G \frac{d}{dt} \{\sin \psi(t)\} + G.a.\sin \psi(t) = \frac{d^2 \phi}{dt^2} \quad (6.55)$$

Trong đó $\psi(t)$ là lỗi pha PLL và $\phi(t)$ là độ lệch pha của tín hiệu đầu vào. Trước khi triển khai sơ đồ mô phỏng máy tính tương tự, ta viết phương trình trên ở dạng:

$$\frac{d^2 \psi}{dt^2} + G \cos \psi(t) \frac{d\psi}{dt} + G.a.\sin \psi(t) = \frac{d^2 \phi}{dt^2} \quad (6.56)$$

Sơ đồ mô phỏng cho hệ thống này được minh họa trong hình 6.14. Ta mong muốn khảo sát hệ thống này theo sự thay đổi bước nhảy về độ lệch tần số đầu vào tại $t = t_0$. Vì vậy, ta đặt:

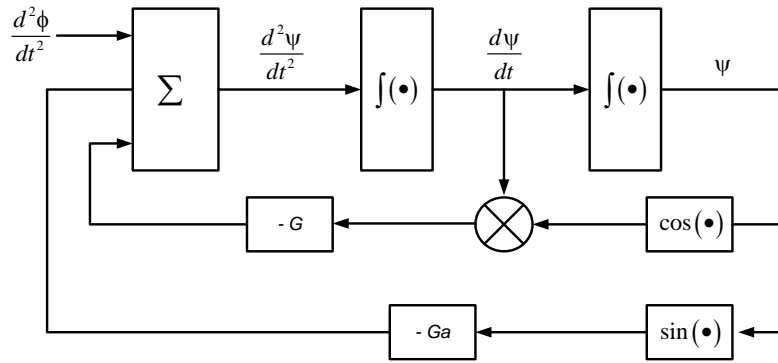
$$\frac{d\phi}{dt} = 2\pi f_\Delta u(t-t_0) \quad (6.57)$$

Dẫn đến:

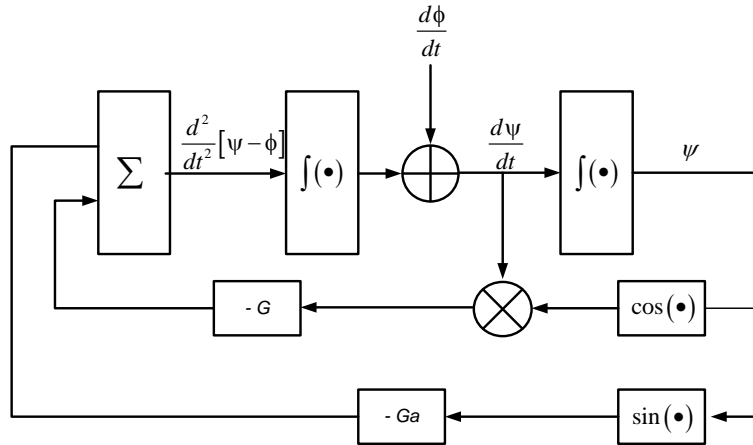
$$\frac{d^2\phi}{dt^2} = 2\pi f_{\Delta} \delta(t-t_0) \quad (6.58)$$

Vì vậy đầu vào được chỉ ra trong hình 6.14 là một xung kim.

Ta thường mong muốn đặt đầu vào tại một điểm trong sơ đồ mô phỏng sao cho dễ dàng mô phỏng. Ví dụ, đầu vào trong hình 6.14 là một xung kim. Lấy tích phân xung kim và chuyển nó sang phải bộ tích phân đầu tiên như được minh họa ở hình 6.15 cho phép ta sử dụng hàm bước nhảy được định nghĩa bởi (6.57) như một "đầu vào tương đương". Những biến đổi đơn giản như kiểu này thường làm mô phỏng đơn giản đáng kể.



Hình 6.14: Sơ đồ mô phỏng cho PLL bậc hai hoàn hảo dựa trên phương trình vi phân



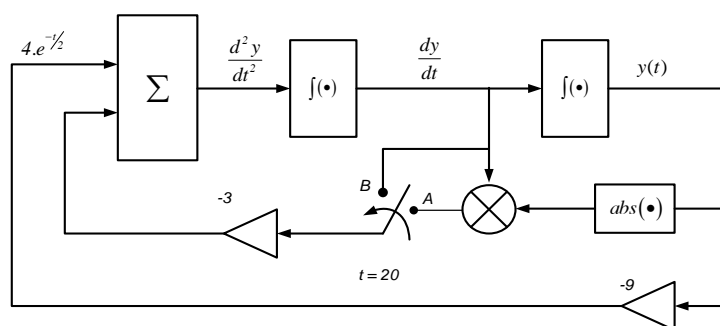
Hình 6.15: Sơ đồ mô phỏng có bước tần số làm đầu vào

Ví dụ 6.3: Mô phỏng hệ thống vừa có tính phi tuyến vừa có tính thay đổi theo thời gian. Giả sử hệ thống được định nghĩa bởi các phương trình vi phân

$$\frac{d^2 y}{dt^2} + \underbrace{3|y(t)|}_{\text{thành phần phi tuyến}} \frac{dy}{dt} + 9y(t) = 4e^{-t/2} \quad t < 20 \quad (6.59)$$

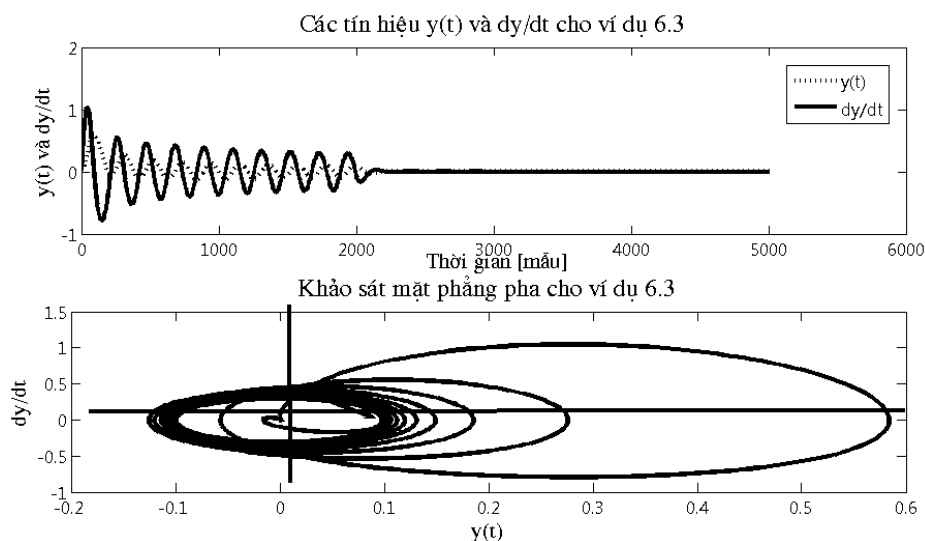
$$\frac{d^2 y}{dt^2} + 3 \frac{dy}{dt} + 9y(t) = 4e^{-t/2} \quad t \geq 20 \quad (6.60)$$

Hệ thống là phi tuyến vì số hạng $|y(t)| \frac{dy}{dt}$ trong phương trình *định nghĩa* hệ thống khi $t < 20$, và hệ thống thay đổi theo thời gian vì dạng của phương trình vi phân là phụ thuộc thời gian. Hệ thống được đặc trưng hóa bởi hai phương trình vi phân riêng biệt được thực hiện bằng cách chuyển mạch vị trí tại $t = 20$ giây như được minh họa ở hình 6.16. Ở vị trí A hệ thống là *phi tuyến* (6.59), ở vị trí B hệ thống là *tuyến tính* (6.60). Mặt phẳng pha sẽ minh họa như mong đợi là: đáp ứng của hệ thống *phi tuyến* là hoàn toàn khác với đáp ứng của hệ thống *tuyến tính*. Như ta đã biết, giải các phương trình vi phân phi tuyến là một nhiệm vụ vô cùng khó khăn vì vậy ta chuyển sang mô phỏng. Chương trình Matlab thực thi hệ thống ví dụ được cho ở file **NVD6_nltvde.m** trong Phụ lục 6A. Lưu ý những nét giống nhau giữa hệ thống này và PLL bậc hai. Vì cả hai hệ thống đều là bậc hai nên cả hai hệ thống này đều phải có 2 bộ tích phân nhưng điểm khác nhau cơ bản giữa chúng là phương trình tại đầu ra của chỗ nối tổng (vị trí làm kín vòng mô phỏng).



Hình 6.16: Minh họa hệ thống phi tuyến thay đổi theo thời gian

Thực hiện chạy chương trình mô phỏng **NVD6_nltvde.m** ở Phụ lục 6A, kết quả là một mặt phẳng pha được minh họa trong hình 6.17. Lưu ý rằng quỹ đạo pha cả hai trường hợp đều bắt đầu và kết thúc tại gốc tọa độ (0,0). Điểm mà tại đó chuyển mạch lật trạng thái $t = 20$ s được chỉ thị một cách xấp xỉ bởi mũi tên (xem hình 6.17). Ta thấy rằng mặt phẳng pha dần tới vòng tròn giới hạn như được chỉ ra bởi sự hội tụ của quỹ đạo pha tới vòng khép kín. Hệ thống A hoạt động trong chế độ chu kỳ-hữu hạn là một bộ dao động. Tần số của bộ dao động và dạng sóng của $y(t)$, hoặc đạo hàm của nó có thể được xác định bằng cách vẽ các dạng sóng trong miền thời gian. Dao động sẽ liên tục miễn là chuyển mạch ở vị trí A.



Hình 6.17: Mặt phẳng pha đối với mô phỏng ví dụ 6.3

Chuyển mạch sang vị trí B, dẫn đến hệ thống tuyến tính. Tại thời điểm chuyển sang vị trí B ($t = 20$ giây) thì đầu vào là không đáng kể (chính xác là $4.e^{-t/20} \Big|_{t=20} = 4.e^{-10}$) và đáp ứng hệ thống xấp xỉ kín thuộc loại hệ thống bậc hai không kích thích, với các điều kiện đầu được cho bởi các giá trị của $y(t)$ và $\frac{dy}{dt}$ tại thời điểm chuyển mạch chuyển từ vị trí A sang vị trí B.

Trong toàn bộ chương này ta đã sử dụng quy tắc tích phân số hình thang để thực hiện xấp xỉ tích phân. Thực hiện vấn đề này là đơn giản, đã được đề cập ở chương trước, thấy rõ là quy tắc hình thang khá chính xác nếu tần số lấy mẫu đủ lớn. Các quy tắc tích phân khác sẽ là chủ đề nghiên cứu sau này khi trình bày nghiên cứu về hệ thống phi tuyến tổng quát hơn (xem chương 12).