Frequency Domain Versus Time Domain Methods in System Identification — Revisited

Lennart Ljung

Division of Automatic Control

E-mail: ljung@isy.liu.se

14th June 2007

Report no.: [LiTH-isy-R-2788](#)

Accepted for publication in Keith Glover's 60'th Birthday Symposium.
Springer Verlag

Address:

Department of Electrical Engineering

Linköpings universitet

SE-581 83 Linköping, Sweden

WWW: <http://www.control.isy.liu.se>

AUTOMATIC CONTROL
REGLERTEKNIK
LINKÖPINGS UNIVERSITET

Abstract

For some reason I now forget, Keith Glover and I were asked to prepare a discussion paper on frequency and time domain methods in system identification by the organizers of the IFAC symposium on System Identification in Darmstadt 1979. I visited Keith in Cambridge in April 1979 to prepare our paper. It later evolved into an Automatica article. Since then, there has been a useful development of several of the problems and questions dealt with in that paper. Questions of Maximum Likelihood formulations, asymptotic analysis, choice and design of noise filters and numerical algorithms, for example have been discussed in many references. In this contribution for the Glover Birthday Conference I will focus on three aspects that relate to using frequency domain data for identification: The role of periodic input data, the role of the input intersample behavior and packaging a transparent time-frequency domain tool for identification.

Keywords: identification

Frequency Domain Versus Time Domain Methods in System Identification — Revisited

Lennart Ljung¹

Dept of Electrical Engineering, Linköping University, SE-582 45 Linköping,
Sweden, ljung@siy.liu.se

1 A Personal Introduction

For some reason I now forget, Keith Glover and I were asked to prepare a “discussion paper” on frequency and time domain methods in system identification by the organizers of the IFAC symposium on System Identification in Darmstadt 1979, [8]. I visited Keith in Cambridge in April 1979 to prepare our paper. This was my first “scientific” visit to Cambridge, and I had some very productive and pleasant days with Keith there. After the symposium we were asked to produce an Automatica version of the discussion paper, and this was eventually published as [9]. Although I have met Keith and had interesting and enjoyable discussions with him many, many times since then, these two papers remain our only joint publications.

While Keith has moved on to many more exciting problems, I have continued to stay close to system identification. Quite recently, as I extended the System Identification toolbox [6] to deal with frequency domain data, I had occasion to revisit the issues we dealt with in our joint paper [9]. This paper has probably been all but forgotten by both of us. It is not a central publication of either of us and has been cited only 25 times over the years according to the ISI Web of Knowledge. Anyway, rereading it today shows that it has not aged that much, and several of the issues are still relevant. I think it is a witness to Keith’s ability and ambition to do a professional job in all situations.

This is the background why I found it appropriate to celebrate Keith’s birthday and illustrious career by going back to our joint work, a quarter of a century ago.

There has been a useful development of several of the problems and questions dealt with in [9]. Questions of Maximum Likelihood formulations, asymptotic analysis, choice and design of noise filters and numerical algorithms, for example have been discussed in, among many references, [4], [3], [19], [15], [12], [13], [20], [10], and [16].

In this contribution I will focus on three aspects that relate to using frequency domain data for identification: The role of periodic input data, the role of the input intersample behavior and packaging a transparent time/frequency domain tool for identification.

Section 2 describes the direct use of frequency domain data for identification, while the apparent importance of periodic inputs is illustrated in Section 3. The fact that also non-periodic data can be handled in a correct way by proper use of initial conditions is explained in Section 4. The issues of the intersample behavior of the input and the role of bandlimited data are discussed in Sections 5 and 6. Finally, the software aspects are dealt with in Section 7.

2 Data in the Frequency Domain

Our paper [9] primarily deals with frequency domain methods such as spectral analysis, estimating frequency functions and spectra and using Parseval's relationship to gain insight into time domain expressions.

At the same time, in the instrumentation and measurement community, frequency domain data were being used as primary measurements for model building. This involved periodic, band-limited inputs in well controlled experiments, to construct transfer functions and refined frequency function estimates. Frequency analyzers were often used to generate, collect and compress data in such experiments, which could be seen as direct extensions of frequency analysis (sine-wave testing). Vibration and modal analysis were and are common applications of this type.

With band-limited, periodic data, the Fourier transforms (or Fourier coefficients) of the input and output signals $U(i\omega)$ and $Y(i\omega)$ can readily be determined from sampled data. In case non-band-limited noise affects the output, this should first be anti-alias filtered, in order not to confuse the transform below the Nyquist frequency. If the linear system that relates the input and output has the frequency function $G(i\omega)$, the relationship is simply

$$Y(i\omega) = G(i\omega)U(i\omega) + V(i\omega) \quad (1)$$

where $V(i\omega)$ is the Fourier transform of the additive noise (contributions below the Nyquist frequency). For good signal-to-noise ratios the quotient

$$\hat{G}(i\omega) = \frac{Y(i\omega)}{U(i\omega)} \quad (2)$$

(“the empirical transfer function estimate”, ETFE) will be a good estimate of the frequency function and is typically delivered as the output of frequency analyzers. If the frequency function is parameterized as $G(i\omega, \theta)$, a natural, least squares estimate of θ will be

$$\hat{\theta}_N = \arg \min_{\theta} \sum_{k=1}^N |Y(i\omega_k) - G(i\omega_k, \theta)U(i\omega_k)|^2 \quad (3)$$

where the summation is over the DFT-frequency grid.

A weighted version is obtained by

$$\hat{\theta}_N = \arg \min_{\theta} \sum_{k=1}^N c_k |Y(i\omega_k) - G(i\omega_k, \theta)U(i\omega_k)|^2 \quad (4)$$

where varying reliability and relevance of the measurements at different frequencies can be taken care of by proper choices of the weights c_k . All this is very basic, and can be elaborated into a maximum likelihood framework, e.g. [18], [15], [5], Section 7.7.

3 Periodic Inputs

For (1) to hold exactly, the input and output must be periodic, otherwise transient effects from circular convolution will corrupt the relationship. This could be a non-trivial problem, as the following simple example shows:

Consider the discrete time system

$$G(q) = \frac{1 + 2q^{-1} + 3q^{-2}}{1 - q^{-1} + 0.9q^{-2}}$$

(q denotes the shift operator). If this system is simulated noise-free with a white input over 50 samples, a time domain fit to the data will give us the true system, exactly. By Fourier transforming the data and applying (3) we get instead the estimate

$$\hat{G}(q) = \frac{1.617 + 2.056q^{-1} + 3.217q^{-2}}{1 - 0.96q^{-1} + 0.83q^{-2}}$$

Since the data is noise free, the discrepancy is entirely due to non-periodic convolution effects in (1).

Now, there are several good reasons to use periodic inputs in system identification, whenever possible. Perhaps the most important one is that after the transient has died out, the differences in the output over different periods must be due to the noise influence. This means that simple estimates of the noise level will be possible. Such knowledge is invaluable for evaluating model quality.

Still, it may not be possible to apply periodic inputs in experiments that are not fully controlled. Then one would think that frequency domain methods like (3) can not be applied. In fact, it seems like this was a major reason why methods based on frequency-domain data were rejected – or at least not used – in the control oriented system identification community until recently.

However, it eventually was found, as shown in the next section, that the question of periodic or not periodic is just a reflection of unknown initial conditions, which is equally relevant for time-domain methods.

4 Transients and Initial States

Let us go back to the basic relationship (1) and focus on discrete time. The arguments below are applicable to multi-input-multi-output system, even though the notation suggests a SISO system.

Let us consider the noise-free part of the response

$$y_u(t) = G(q)u(t) \quad (5)$$

We assume only a finite number of samples of inputs and outputs be known:

$$\begin{aligned} y(1), y(2), \dots, y(N) \\ u(1), u(2), \dots, u(N) \end{aligned}$$

The inputs prior to $t = 1$ are thus not known. Let $\tilde{y}_u(t)$ denote what would be the output corresponding to a particular assumption about $u(t)$, $t = -\infty, \dots, -2, -1, 0$. Two typical cases would be

$$\begin{aligned} \tilde{y}_u(t) &= y_u^0(t) \text{ is the output of (5) if } u(t) = 0, t \leq 0 \\ \tilde{y}_u(t) &= y_u^p(t) \text{ is the output of (5) if } u(t) \text{ is periodic} \\ &\quad \text{with period } N \text{ from } t = -\infty \text{ to } t = N \end{aligned}$$

Let now (5) be realized in state space form:

$$x(t+1) = Ax(t) + Bu(t) \quad (6)$$

$$y(t) = Cx(t) \quad (7)$$

Whatever assumption about prior values of $u(t)$ we have made, it would have left us in a certain state $x(0) = \tilde{x}$ at time 0. (For example, all prior u 's being zero would give $x(0) = 0$.) Let the actual, typically unknown, initial state be $x(0) = x^*$. Then

$$\begin{aligned} y_u(t) &= \tilde{y}_u(t) + \tilde{y}_u(t) \\ \tilde{y}_u(t) &= C(qI - A)^{-1}A(x^* - \tilde{x})\delta(t) \\ \delta(t) &= \begin{cases} 0 & \text{if } t \neq 0 \\ 1 & \text{if } t = 0 \end{cases} \end{aligned}$$

The term $\tilde{y}_u(t)$ is thus the response from the initial conditions. Alternatively it can be seen as the impulse response *from an additional input, which is an impulse*:

$$\begin{aligned} x(t+1) &= Ax(t) + Bu(t) + R\delta(t) \\ x(0) &= \tilde{x} \text{ (the assumed input behavior prior to } t = 0) \\ y(t) &= Cx(t) \\ R &= A(x^* - \tilde{x}) \end{aligned}$$

The consequence is that *any (possibly erroneous) guess of input behavior prior to time $t = 0$ can always be made up for by adding an extra input which is an impulse at time 0. The dynamics from this input has the same poles as the system but unknown zeros.* Note that *one* extra input is sufficient, even if there are several regular inputs.

Of course, the whole process can be described without reference to a state space realization (6). The model to be considered in input-output form would be

$$y(t) = G(q)u(t) + G_i(q)\delta(t) = \frac{B(q)}{F(q)}u(t) + \frac{R(q)}{F(q)}\delta(t) \quad (8)$$

where the second step corresponds to a rational transfer function parameterization of the model. Note that the denominators are the same, and that the order of the R -polynomial is one less than the order of $F(q)$. Also, if the input u is of dimension r , there would be r different numerator polynomials $B_k(q)$, but still just one $R(q)$ -polynomial.

The typical two cases for assumed prior behavior of the inputs are

1. In the time domain: Assume that all prior values of $u(t)$ are zero. This will give the simple predictor $\hat{y}(t) = G(q)u(t)$ with all values of u prior to $t = 1$ being zero.
2. In the frequency domain: Assume that all prior values of u are obtained by periodic continuation of u backwards in time. This will make the Fourier transformed relation

$$Y_N(e^{i\omega T_s}) = G(e^{i\omega T_s})U_N(e^{i\omega T_s}) \quad (9)$$

exact for the u -influence at the DFT-grid-points. Here Y_N and U_N are the Discrete Fourier Transforms of the output and input (defined by (12) below).

Now, for general data sets, these two assumptions are not correct, but the point is that an extra input signal which is an impulse will make them correct, if this input is passed through a system with the same poles as the model, and the zeros are adjusted to data (to match the assumption.) This extra input can be neglected, only if we know that the input is periodic in the frequency domain case, or past values are zero in the time domain. For long data records, it may be of less importance, since the effects of this impulse response may decay quickly compared to the data length.

This way to compensate for non-periodic frequency domain data was described in [17]. See also [19] for an instructive discussion.

Note that the DFT of the impulse $\delta(t)$ is just a sequence of 1's. To actually achieve this treatment of the non-periodicity is thus quite simple in practice: Assume in the model structure that there is one more input and append to the input DFT U_N one column of 1's.

The analysis has also brought out the kinship between time- and frequency methods: The non-periodicity is for frequency domain data *exactly the same*

problem as non-zero initial state for time-domain data. One should note that the estimated R -vector (or R -polynomial) is specific for the estimation data. In particular, one must be careful when using an estimated initial condition for another data sequence. This is the same problem for time and frequency domain data.

Example, [7]

The system

$$y(t) = \frac{q^{-1} + 0.5q^{-2}}{1 - 1.5q^{-1} + 0.99q^{-2}}$$

was simulated noise-free over 150 data points with a white noise input. Samples 101 to 150 were selected for identification. Models were fit to these data both in the time and frequency domains and both with and without adding an extra input being an impulse. This gave the following estimates:

- Time domain and frequency domain with extra input:

$$y(t) = \frac{q^{-1} + 0.5q^{-2}}{1 - 1.5q^{-1} + 0.99q^{-2}}$$

The “initial state” or numerator polynomial for the extra impulse input were $R_i(q) = -14.06q^{-1} - 1.079q^{-2}$ and $R_i(q) = -18.99 - 1.63q^{-1}$ respectively.

- Time domain without extra input:

$$y(t) = \frac{1.32q^{-1} - 0.26q^{-2}}{1 - 1.55q^{-1} + 0.94q^{-2}}$$

- Frequency domain without extra input

$$y(t) = \frac{1.04q^{-1} + 0.68q^{-2}}{1 - 1.47q^{-1} + 0.87q^{-2}}$$

This illustrates clearly that erroneous assumptions about past inputs (they are neither zero nor periodic in this case) can give bad results both in time- and frequency domains even for noise-free data. However, they can be handled with proper use of initial conditions (an extra input impulse) in the same way in the time and the frequency domains.

(The four models could be reproduced in [6] by the code

```
m0 = idpoly(1,[0 1 0.5],1,1,[1 -1.5 0.99]);
u = randn(150,1);
y = sim(m0,u);
```



```

z = iddata(y,u);
ze = z(101:150);
udel = [1;zeros(49,1)];
zed = ze;
zed.u = [ze.u udel];
m1 = oe(zed,[2 2 2 2 1 1], 'ini', 'z');
m2 = oe(fft(zed),[2 2 2 2 1 0], 'ini', 'z');
m3 = oe(ze,[2 2 1], 'ini', 'z');
m4 = oe(fft(ze),[2 2 1], 'ini', 'z');

```

(The pair 'ini', 'z' is there to inhibit the default estimation of initial states.) Of course, the addition of the extra impulse input and the estimation of it associated zeros can easily be automated. This is done in [6] by

```

m1 = oe(ze,[2 2 1], 'ini', 'e'); %'e' for 'estimate'
m2 = oe(fft(ze),[2 2 1], 'ini', 'e');

```

5 Continuous-time Data with Band-limited Input

In the time-domain there is no real possibility to deal with continuous-time data directly. A great feature of frequency domain data is that it can handle continuous time measurements. Let $U(i\omega)$ and $Y(i\omega)$ be the (continuous time) Fourier transforms of the input and the output:

$$\begin{aligned}
 U(i\omega) &= \int_0^\infty u(t)e^{-i\omega t} dt \\
 Y(i\omega) &= \int_0^\infty y(t)e^{-i\omega t} dt
 \end{aligned} \tag{10}$$

Then a finite dimensional representation of the data is obtained by collecting these transforms at a finite number of frequencies:

$$Z^N = \{U(i\omega_1), Y(i\omega_1), \dots, U(i\omega_N), Y(i\omega_N)\} \tag{11}$$

Such data could be used for directly fitting continuous time models as in (3). Clearly, in practice we neither have infinite nor continuous time data records. Still, the Fourier transform (10) could be evaluated under some circumstances.

To get correct values for (10)-(11) in practice, it is required that the whole continuous signal $u(t)$, $-\infty < t < \infty$ can be reconstructed from the sampled, observed data $\{u(T_s), \dots, u(NT_s)\}$ (T_s being the sampling interval). This is the case if u is *periodic* and *band-limited*. (In the periodic case, the Fourier transform values $U(i\omega_k)$ will have to be interpreted as the Fourier coefficients corresponding to the fundamental frequency and its harmonics, resulting from applying (10) to one period of data only.) That is to say, that the results of the simple DFT transformation

$$U_N(e^{i\omega T_s}) = T_s \sum_{k=0}^{N-1} u(kT_s) e^{-i\omega kT_s} \quad (12)$$

evaluated on the “DFT-grid”

$$\omega_k = 2\pi k / (NT_s), k = 0, 1, \dots, N-1 \quad (13)$$

can be interpreted as the Fourier coefficients of the infinite signal obtained by making $u(t)$ periodic with period NT_s , and making it time continuous by trigonometric interpolation (no power above the Nyquist frequency).

If the relation between input and output is

$$y(t) = G(p)u(t) + v(t)$$

the output y need not be periodic and band-limited even if u is that, due to the influence of the noise v . However, *the component of y that originates from u (i.e. Gu) is periodic and band-limited*. Since the estimate of G only should depend on this component, frequencies in y higher than the highest frequency in u should be removed. Then the output could be averaged over the periods to get accurate values of the relevant components of $Y_N(i\omega_k)$.

Another, more pragmatic way is when the sampling interval T is small compared to the interesting dynamics of the system and compared to the rate of change in u . Then the input could be considered to be band-limited in practice, and the values from (12) (after suitable normalization) will be good estimates of $U_N(i\omega)$.

6 Other Intersample Behaviors

The problem to compute the continuous-time Fourier transform from equidistantly sampled data is, as we saw, trivial if the signal is band-limited. In general, the continuous-time transform can always be computed from sampled data if the *intersample* behavior of the signal is known.

Typical intersample behaviors of the input are

- Zero-order hold (zoh): The input is piecewise constant between samples
- First-order hold (foh): The input is piecewise linear between samples

In the zoh case with sampling interval T_s we have

$$u(t) = u(kT_s) \text{ for } kT_s \leq t < (k+1)T_s$$

so assuming a periodic input with period $T = NT_s$ we have

$$\begin{aligned}
U_c(i\omega) &= \int_0^T u(t)e^{-i\omega t} dt = \sum_{k=0}^{N-1} \int_{kT_s}^{(k+1)T_s} u(kT_s)e^{-i\omega t} dt \\
&= \sum_{k=0}^{N-1} u(kT_s)e^{-i\omega kT_s} \frac{1 - e^{-i\omega T_s}}{i\omega} = U_N(e^{i\omega T_s})H_{T_s}^z(i\omega) \\
H_{T_s}^z(i\omega) &= \frac{1 - e^{-i\omega T_s}}{T_s i\omega}
\end{aligned} \tag{14}$$

Similarly, for a first order hold input we have

$$\begin{aligned}
U_c(i\omega) &= U_N(e^{i\omega T_s})H_{T_s}^f(i\omega) \\
H_{T_s}^f(i\omega) &= e^{i\omega T_s} \left(\frac{1 - e^{-i\omega T_s}}{T_s i\omega} \right)^2
\end{aligned} \tag{15}$$

A very important aspect of band-limited input is that also the model-dependent part of the output is band-limited, so that also the continuous-time Fourier transform of this part can be readily computed. This is not the case for zoh and foh inputs: If we have a continuous time system

$$y(t) = G_c(p)u(t)$$

which is sampled with a zero-order hold input, the sampled signals obey

$$y(kT_s) = G_d(q)u(kT_s)$$

where G_d is computed from G_c in a well-defined and well-known manner. This gives the relationships between the Fourier transforms (U_N is the DFT (12) and U_c is defined by the first equality in (14), and analogously for Y):

$$\begin{aligned}
Y_N(e^{i\omega T_s}) &= G_d(e^{i\omega T_s})U_N(e^{i\omega T_s}) \\
Y_c(i\omega) &= G_c(i\omega)U_c(i\omega) = G_c(i\omega)H_{T_s}^z(i\omega)U_N(e^{i\omega T_s})
\end{aligned}$$

from which we find the continuous time transform for the output:

$$\begin{aligned}
Y_c(i\omega) &= F_{T_s}(i\omega)Y_N(e^{i\omega T_s}) \\
F_{T_s}(i\omega) &= \frac{G_c(i\omega)}{G_d(e^{i\omega T_s})}H_{T_s}^z(i\omega)
\end{aligned} \tag{16}$$

where H^z was defined in (14). Note that the function $F_{T_s}(i\omega)$ depends on the system G_c . For example, if $G_c(s) = 1/s$ we obtain, after simple calculations, that $F_{T_s}(i\omega) = H_{T_s}^f(i\omega)$, which naturally enough shows that a zoh input becomes a foh output after having been integrated. Anyway, the result (16) cannot be applied in an identification situation, where G_c is unknown. This shows the unique advantage of band-limited input for estimation continuous time models in the frequency domain.

Even though (16) cannot be applied directly for identification, we can reason as follows, see [2]: If the signals are reasonably fast sampled, it is only the high frequency behavior of F that matters, that is only the roll off of the frequency functions G_c and G_d is important. That means that if we know that the system G_c has a pole excess of ℓ then we can approximate G_c by ℓ integrations:

$$G_c(s) = \frac{b}{s^\ell} \quad (17)$$

when computing F_{T_s} in (16). The system approximation (17) has been used in several related contexts, like approximating zeros of sampled systems, [22], [21]. Sampling (17) gives expressions involving the Euler-Frobenius polynomials, so these will play a role in (16).

All this means that from sampled input-output data we can easily find the continuous-time Fourier transforms if the input is band-limited. If the intersample behavior of the input is known (e.g. zoh or foh), its continuous-time transform can also be constructed, while the reconstruction of transform of the output would require knowledge of the system. However, various approximations can be formed, whose accuracies improve as the sampling gets faster. In such cases it could be an advantage from a numerical point of view to use a δ -parameterization of the transfer function, see [14]. Also, some of the expressions (14)-(16) become more transparent when expressed in their δ -parameterized version.

7 Software Aspects

We have described how frequency domain data can be used for identification, also for non periodic data, and the role of continuous time data in the frequency domain. This gives a ground for a package for identifying linear systems that truly integrates time- and frequency data. In this section we shall describe how this is done in the System Identification Toolbox (SITB) [6]. The goal of the syntax is to handle time domain data, frequency domain input-output data, and frequency response data in entirely analogous fashions both for estimating and validating models.

7.1 Input-Output Fourier Transform Data

The `iddata` object contains input – output data. In the time domain case the definition of the object from data vectors or matrices is straightforward:

```
dat = iddata(y,u,Ts)
```

where `Ts` is the sampling interval. If the data instead is defined as continuous time Fourier transform data over arbitrary frequencies as in (11) it can be collected into the data object as

```
dat = iddata(Y,U,'Frequencies',W,'Ts',0);
```

Note that the sampling interval, T_s , ('Ts') is still relevant, since it has information of how the signal Fourier transforms Y and U have been computed from time domain data. Discrete time Fourier Transforms conceptually have the frequency argument $e^{i\omega T_s}$. Note however, that frequency domain data, unlike time domain data allow continuous time signals ($T_s = 0$). Compare the discussion in Section 5.

With frequency domain data objects, several MATLAB commands are naturally extended to be directly applied to the objects:

```
DF = fft(dat)
dat = ifft(DF)
da = abs(dat)
df = phase(dat)
```

etc.

7.2 Frequency Response Data

Frequency response data for a system given as $G(i\omega_k)$ for a frequency vector ω_k can be obtained, e.g. as the ETFE (2). Such data can be stored in the **idfrd** object in the SITB. It corresponds to the frequency response data object **frd** in the CONTROL SYSTEMS TOOLBOX. The object is formed by

```
dat = idfrd(G,W,Ts,'cov',C)
```

G contains the response data $G(i\omega_k)$, and W the frequencies ω_k . Ts is the sampling interval T_s ($Ts = 0$ denotes continuous time). It may be that a measure of reliability can be associated with the different values of the frequency response. If the number C_k corresponds to an estimate of the covariance of the value $G(i\omega_k)$, this information can be stored in the **idfrd** object by defining the vector $C = [C_1, C_2, \dots, C_N]$ as indicated above. If such an uncertainty measure is not known, it can be omitted.

It is often the case that experimental frequency domain response data are stored in this way as frequency functions rather than as inputs and outputs, separately. This also could be more economical, e.g. by using logarithmically spaced frequencies. There is also data acquisition equipment that collect and deliver the result of the experiment as frequency response data.

7.3 Estimation and Validation

The point now is that whatever the format of **dat**, estimation and validation of models follow the same syntax:

```
m1 = oe(dat,[2 2 1])
m2 = n4sid(dat,3)
m3 = pem(dat)
```

```
compare(dat,m1,m2,m3)
resid(dat,m1)
```

etc. The prediction error approaches (`oe`, `pem` etc) implement variants of the minimization problem (3), while the subspace estimation command `n4sid` is described in [12] for frequency domain data. (See also [11].)

Arbitrary weighings c_k in the frequency domain fits, as in (4), can be obtained by

```
m = oe(dat,[2 2 1], 'focus',c)
```

By default, initial states are always estimated, as described in Section 4. This estimation can be inhibited by

```
m = oe(dat,[2 2 1], 'InitialState', 'zero')
```

If the frequency domain data is denoted as continuous time, a continuous time model is estimated directly by fitting its frequency function to data as in (3). Compare also [1].

All the toolbox commands for simulation and validation handle time/frequency domain data in a transparent manner. The only restriction is that, for the moment, estimation of noise models and time-series models is not supported for frequency domain data. The reason is the more complex criteria of fit that must be used for continuous-time and non-equidistantly spaced frequency data. See, e.g. page 230 in [5].

7.4 Some Further Features

Frequency domain data offer useful potentials also for other problems:

- A *focus filter* can be implemented as specific frequencies for which the fit should be made. For example,

```
m = oe(dat,[2 2 1], 'focus',[0.2 1])
```

will concentrate the fit to the pass band from 0.2 to 1 rad/s. This idea of *prefiltering* is of course also available in the time domain, but it is particularly easy to implement in the frequency domain: Just cut out the relevant part of the frequency domain data vectors.

The desired frequency bands may not necessarily be known *a priori*, but could be selected from a preliminary model, like using frequencies that correspond to the Nyquist curve being in the third quadrant, or being close to the critical point -1 . Example: Build a fifth order black-box model of the data, and compute its frequency response. Then adjust a simple first order process model with a delay

$$G(s) = \frac{K}{1 + T_1 s} e^{-T_d s} \quad (\text{P1D})$$

to this frequency response, restricted to the third quadrant:

```

m = n4sid(data,5);
f = idfrd(m);
ph = phase(squeeze(f.resp));
fs = fselect(f,find(ph>-pi & ph<-pi/2));
mp = pem(fs,'pid');

```

- If the inter-sample input behavior is band-limited, moving to the frequency domain will be the easiest way to handle the sampling. The FFT (discrete Fourier transform) of the input will then be equal to the Fourier transform of the underlying continuous time input signal. (Cf Section 5.) The FFT of the output will similarly correctly describe the continuous time Fourier transform of that part of the output that originates from the input, and we can directly fit a continuous time model – in this case of the kind

$$G(s) = \frac{b}{s^2 + f_1 s + f_2}$$

```

dat = iddata(y,u,0.5);
datf = fft(dat);
datf.Ts = 0;
mp = oe(datf,[1 2])

```

This will work also if the input is not band-limited, but sampled fast compared to dominating time constants. Then it will be a good idea to focus the fit to lower frequencies:

```
mp = oe(datf,[1 2], 'foc',[0 1])
```

7.5 GUI support

The graphical user interface (GUI) has been extended to be transparent wrt the data domain. Frequency domain `iddata` and frequency response data as `frd` or `idfrd` objects can be imported into the GUI in the same way as time domain data. See Figure 1. The icons for the different types of data sets are marked by different background colors. The **data preprocessing** menus allow the transformation between the various representations. Also the use of data objects of different types for estimation and validation is entirely transparent. For example, if an `idfrd` object is chosen as validation data, the **Model output view** shows the frequency responses of the models, together with the data.

8 Conclusions

A fair amount of development of the understanding of frequency domain methods in system identification has taken place in the 25 years since [9] was written. Among the most important ones are the practical insights into how to handle non-periodic data, and the potential to use frequency domain data for estimating continuous-time models.

Fig. 1. The GUI

References

1. H. Garnier and P. C. Young. Time-domain approaches to continuous-time model identification of dynamical systems from sampled data. In *Proc. American Control Conference*, Boston, MA, July 2004.
2. J. Gillberg and L. Ljung. Frequency domain identification of continuous-time output error models from sampled data. In *Proc. 16th IFAC World Congress*, Prague, Czech Republic, July 2005.
3. I. Kollar. *Frequency Domain System Identification Toolbox for use with MATLAB*. The MathWorks Inc., Natick, MA, USA, 1994.
4. L. Ljung. Building models from frequency domain data. In K.J. Åström, G.C. Goodwin, and P.R. Kumar, editors, *Adaptive Control, Filtering and Signal Processing*, volume 74 of *The IMA volumes in Mathematics and its Applications*, pages 229–240. Springer-Verlag, New York, 1995.
5. L. Ljung. *System Identification - Theory for the User*. Prentice-Hall, Upper Saddle River, N.J., 2nd edition, 1999.
6. L. Ljung. *System Identification Toolbox for use with MATLAB. Version 6*. The MathWorks, Inc, Natick, MA, 6th edition, 2003.

7. L. Ljung. State of the art in linear system identification: Time and frequency domain methods. In *Proc. American Control Conference*, Boston, MA, July 2004.
8. L. Ljung and K. Glover. Frequency domain versus time domain methods in system identification - a brief discussion,. In R. Isermann, editor, *Proc. Fifth IFAC Symposium on Identification and System Parameter Estimation*,, pages 1309–1322, Darmstadt, Germany, September 1979. Discussion Paper.
9. L. Ljung and K. Glover. Frequency domain versus time domain methods in system identification. *Automatica*, 17(1):71 – 86, 1981.
10. T. McKelvey. Frequency domain identification. In R. Smith and D. Seborg, editors, *Proc 12th IFAC Symposium on System Identification*, Santa Barbara, CA, USA, June 2000. Plenary Paper.
11. T. McKelvey. Subspace methods for frequency domain data. In *Proc. American Control Conference*, Boston, MA, July 2004.
12. T. McKelvey, H. Akçay, and L. Ljung. Subspace-based multivariable system identification from frequency response data. *IEEE Trans. on Automatic Control*, 41(7):960–979, Jul 1996.
13. T. McKelvey and L. Ljung. Frequency domain maximum likelihood identification. In *Proc. 10th IFAC Symposium on System Identification*, Fokuoka, Japan, 1997.
14. R.H. Middleton and G.C. Goodwin. *Digital Control and Estimation: A Unified Approach*. Prentice-Hall, 1990.
15. R. Pintelon and J. Schoukens. *System Identification – A Frequency Domain Approach*. IEEE Press, New York, 2001.
16. R. Pintelon and J. Schoukens. Box-Jenkins identification revisited - Parts I and II. *Automatica*, 2005. To appear.
17. R. Pintelon, J. Schoukens, and G. Vandersteen. Frequency domain system identification using arbitrary signals. *IEEE Trans. Automatic Control*, 42:17171–1720, 1997.
18. J. Schoukens and R. Pintelon. *Identification of Linear Systems: A Practical Guideline to Accurate Modeling*. Pergamon Press, London (U.K.), 1991.
19. J. Schoukens, R. Pintelon, and Y. Rolain. Time domain identification, frequency domain identification. equivalences! differences? In *Proc. American Control Conference*, Boston, MA, July 2004.
20. J. Schoukens, R. Pintelon, G. Vandersteen, and P. Guillaume. Frequency-domain system identification using non-parametric noise models estimated from a small number of data sets. *Automatica*, 33(6):1073–1087, 1997.
21. S. R. Weller, W. Moran, B. Ninness, and A. D. Pollinton. Sampling zeros and the Euler-Frobenius polynomials. *IEEE Transactions on Automatic Control*, 46(2):340–343, 2001.
22. K. J. Åström, P. Hagander, and J. Sternby. Zeros of sampled systems. *Automatica*, 20(1):31–38, Jan 1984.

Avdelning, Institution Division, Department Division of Automatic Control Department of Electrical Engineering		Datum Date 2007-06-14
Språk Language ☐ Svenska/Swedish ☒ Engelska/English ☐ _____	Rapporttyp Report category ☐ Licentiatavhandling ☐ Examensarbete ☐ C-uppsats ☐ D-uppsats ☒ Övrig rapport ☐ _____	ISBN _____ ISRN _____ Serietitel och serienummer ISSN Title of series, numbering 1400-3902
URL för elektronisk version http://www.control.isy.liu.se		LiTH-ISY-R-2788
Titel Frequency Domain Versus Time Domain Methods in System Identification — Revisited Title		
Författare Lennart Ljung Author		
Sammanfattning Abstract <p>For some reason I now forget, Keith Glover and I were asked to prepare a discussion paper on frequency and time domain methods in system identification by the organizers of the IFAC symposium on System Identification in Darmstadt 1979. I visited Keith in Cambridge in April 1979 to prepare our paper. It later evolved into an Automatica article. Since then, there has been a useful development of several of the problems and questions dealt with in that paper. Questions of Maximum Likelihood formulations, asymptotic analysis, choice and design of noise filters and numerical algorithms, for example have been discussed in many references. In this contribution for the Glover Birthday Conference I will focus on three aspects that relate to using frequency domain data for identification: The role of periodic input data, the role of the input intersample behavior and packaging a transparent time-frequency domain tool for identification.</p>		
Nyckelord Keywords identification		