

**Introduction**

Signal synthesis and recovery is all about the situation illustrated in Figure 1, when the system and the output are given, and the goal is to find a corresponding input. In the signal synthesis problem, the output is a design specification, and the goal is to find an input that causes the system to generate the desired output. In the signal recovery problem, the output is measurement data, and the goal is to find the input that generated it. In practice, there may be many inputs that can generate the same output; hence, additional constraints must be imposed on the input to select a particular solution.

We can pose the situation in Figure 1 somewhat more formally as shown in Figure 2, which suggests the equation

$$\begin{align*} x & \quad \boxed{f} \quad y \\
\end{align*}$$

Figure 2. A mathematically defined system.

Equation (1) immediately raises several mathematical questions. First, what kind of object is \( x \)? We answer this by requiring that \( x \in X \), where \( X \) is some set of admissible system inputs; i.e., admissible arguments for \( f \). Second, what kind of object is \( y \)? Certainly, \( y \) must be the same kind of object as \( f(x) \) for any \( x \in X \). In general, we require that for all \( x \in X \), \( f(x) \in Y \) for some fixed set \( Y \). Note that it is not required that for all \( y \in Y \), there exist an \( x \in X \) with \( f(x) = y \).

In many problems, we have a mathematical model in which a measurement \( y_0 \in Y \) is equal to \( f(x_0) \) for some \( x_0 \in X \). However, due to noise or modeling errors, when \( x_0 \) is applied to the system, the output that is actually measured is

$$y_1 \neq y_0.$$  \( \quad (2) \)

Somehow, based on the observation \( y_1 \), we want to find \( x_0 \). There are two situations to consider. First suppose there is \( x_1 \) such that \( y_1 = f(x_1) \). Then we would like to say something like, “if \( y_1 \) is close to \( y_0 \) then \( x_1 \) will be close to \( x_0 \).” Second, suppose that there is no \( x_1 \in X \) with \( y_1 = f(x_1) \). In this case, we could consider the problem

$$\min_{x \in X} \text{distance}(y_1, f(x)).$$  \( \quad (3) \)

What do “close” and “distance” mean? In general, since \( x \) and \( y \) may be very different kinds of objects, we may need different notions of closeness or distance. In order to examine these questions precisely, we must learn a bit about metric spaces.

In many signal processing applications, the sets \( X \) and \( Y \) are vector spaces. Among other things, this means that there is a notion of addition for objects in \( X \) and a notion of addition for objects in \( Y \). Hence, we can employ additive noise models. For example, instead of (2), we can write \( y_1 = y_0 + \Delta y \) for some nonzero \( \Delta y \). Distance in vector spaces is often measured by a norm. In this case, every vector has a notion of size associated with it. This is usually denoted by \( \|x\| \). The distance between two vectors \( x_0 \) and \( x_1 \) is then taken to be \( \|x_0 - x_1\| \). Since different spaces have different norms, for emphasis we sometimes write \( \|x\|_X \) for \( x \in X \) and \( \|y\|_Y \) for \( y \in Y \).

Another advantage of having \( X \) and \( Y \) be vector spaces is that it makes sense to talk about linear functions (usually called linear transformations or linear operators). In this case, we often denote the function (transformation/operator) by \( A \); we write \( y = Ax \) instead of \( y = f(x) \). When \( Y \) is a normed vector space, (3) becomes

$$\min_{x \in X} \|y_1 - Ax\|_Y.$$  \( \quad (4) \)

As \( x \) runs over \( X \), \( Ax \) runs over

$$\text{range} A := \{ Ax: x \in X \}.$$  

Hence, we are trying to find a point in \( \text{range} A \) that is closest to \( y_1 \). This is the projection problem. Since \( A \) is linear, its range is a subspace. When \( y_1 \) is a point in the plane and the subspace is a line through the origin, the projection problem is straightforward, as shown in Figure 3. The point we need has the property that the error vector is perpendicular (orthogonal) to every vector in the subspace. How can we generalize this idea when \( y_1 \) is a waveform, e.g., a sine wave, instead of a point in two-dimensional space? This brings us to the topic of inner-product spaces. If \( Y \) is an inner-product space with inner product denoted by \( \langle \cdot, \cdot \rangle \), we show later that \( x_1 \) achieves the minimum in (4) if and only if

$$\langle y_1 - Ax_1, Ax \rangle = 0, \quad \text{for all } x \in X.$$  

If in addition \( X \) is an inner-product space with inner product denoted by \( \langle \cdot, \cdot \rangle \), we show later that the above formula holds if and only if

$$\langle A^*y_1 - (A^*A)x_1, x \rangle = 0, \quad \text{for all } x \in X,$$

where \( A^* \) is the adjoint of \( A \) (defined later). Since \( x \) is arbitrary, it follows that \( x_1 \) satisfies the linear equation

$$A^*A)x_1 = A^*y_1.$$  

In many cases the solution of this equation can be found. In particular, if \( X \) is finite dimensional, then \( A^*A \) can be identified
with a matrix, and $A^*y_1$ is a column vector, and $x_1$ can be found using MATLAB.

If you have studied linear algebra, you may be familiar with the **diagonalization** of matrices and the **singular-value decomposition** (SVD) of matrices. These are fundamental tools for studying linear operators on finite-dimensional spaces. However, operators encountered in applications are often defined on infinite-dimensional spaces. Fortunately, the notions of diagonalization and SVD can be generalized to infinite-dimensional settings.

Formula (4) is an example of an optimization problem. Not all such problems can be solved so easily. To minimize a real-valued function of $x$, which we now call $f$, what should we do? If $x$ is a real number or an element of $\mathbb{R}^d$, we can differentiate. What should we do if $x$ is a waveform? How does one differentiate with respect to a waveform? Later we generalize the notion of derivative to functions defined on infinite-dimensional spaces by introducing the Fréchet and Gâteaux derivatives. Since setting these derivatives equal to zero means solving $f'(x) = 0$, we have a special case of (1). It’s all about solving equations...