D. P. Paris

Digital Simulation of Image-Forming Systems

In speaking about the simulation of continuous systems, one usually has in mind the simulation of dynamic systems—that is, systems in which the variables are functions of time. Such systems have frequently been simulated by using an analog computer; but lately, digital simulation techniques have become more popular.[1]

In this communication we will discuss systems whose variables depend on *space* coordinates x and y, rather than on the *time* coordinate t. We will call those systems having space-dependent variables *spatial systems*. Photo-optical image-forming systems, of course, are realizations of spatial systems. Here the system variables, e.g., light intensities, transmittances, or photographic densities, are continuous functions of the two space coordinates x and y. For simplicity, we will restrict the formulations in this communication to one spatial dimension only. The extension to two dimensions is straightforward and does not add anything significantly new.

When studying the theory of photo-optical image-forming systems, it becomes apparent that one can look upon them as consisting of a sequence of connected blocks, very similar to the case of dynamic systems. We find linear blocks which represent lenses, light spread in the photographic emulsion, blurring caused by various kinds of image motion, and so on. Nonlinear blocks may represent the characteristic or H & D curve of the photographic emulsion. The most striking analogy between dynamic systems and spatial systems is that both types can be described either by their impulse response or by their transfer function.*

This analogy suggests that the well-developed methods of dynamic systems simulation might also be applied to spatial systems. This is not possible, however, because of a fundamental difference between the two types of systems. In the time domain there is directionality, while in the space domain there is none. Another way of saying this is that the cause/effect relationship in time has no analogy in space. The consequences of this difference for the problem of simulation are discussed in the second section.

Simonds[6] used the digital computer to simulate the photographic printing process with emphasis on the influence of the adjacency effect. His computer program is not available. Rabedeau[7] also uses the digital computer for simulation of photo-optical systems. His program comes much closer than Simonds' to being a general image-forming systems simulator. Neither of these, however, provides a "simulation language" to permit application to other types of simulation problems.

The Image Forming Systems Simulator IMSIM/1 was designed with such a simulation language, which resembles common English, to permit the engineer to apply the program to photo-optical design problems.* The third section describes briefly one application of this language; a more detailed discussion has been published elsewhere.[8]

Recently, in an independent effort, Lerman, Minnick, and Shannon reported the design of another photo-optical simulation language, FRAP.[9]

Gray and Kippenhan[10] extended IMSIM/1 for use with the IBM 7226 Special Graphic Data Processing System in conjunction with the IBM 7044 Data Processing System. A cathode ray tube display console and an associated light pen of the IBM 7226 provide on-line communication between the user and the program.

Dynamic systems vs spatial systems

The significant difference between dynamic and spatial systems—the directionality in the time domain—has already been pointed out by Elias.[11] If at time t' a unit impulse (Dirac δ-function) is applied to a linear block of a dynamic system (the cause), the impulse response of this block (the effect) is necessarily equal to zero for times less than t' (Fig. 1). The block cannot respond to an input not yet received.

Now consider the corresponding situation for a linear block of a spatial system, for example, a lens imaging the object plane into the image plane. The input and output

* The first book about optical transfer theory was published by P. M. Duffieux[2] but is difficult to obtain. Two recent books (by E. L. O'Neill and by E. H. Linfoot)[3] provide a comprehensive introduction. Tutorial papers by F. H. Perrin[4] and H. H. Hopkins[5] are also recommended.

* The program package is available from the IBM Program Information Department, Hawthorne, New York 10532.

variables are the light-intensity distributions in the object plane and image plane, respectively.* A unit impulse entering a dynamic block at time t' corresponds to a line source of light, located in the object plane at x', entering a spatial block.

Because no lens is ideal, the image of the line in the image plane will be a more or less extended light distribution around x' (Fig. 1). This light distribution in a spatial system (the line-spread function) corresponds to the impulse response in a dynamic system. From the physical behavior of linear spatial blocks, we know that no condition exists in the space domain which corresponds to what we called directionality in the time domain. Although the line-spread function may very well be asymmetrical, it generally will be unequal to zero on both sides of x'.

In order to obtain the input-output relation of any linear block in the form of a convolution, the response of the block must be shift-invariant. This means that the response must be independent of the particular position chosen for the input. Because of the cause/effect relation in the time domain, the convolution integrals for dynamic and spatial systems differ:

Dynamic systems

$$v(t) = \int_{-\infty}^{t} u(t')\,A(t - t')\,dt'$$

Spatial systems

$$v(x) = \int_{-\infty}^{\infty} u(x')\,A(x - x')\,dx'.$$

From this formulation, it becomes apparent that in simulating a dynamic system, only the knowledge of the past history [$u(t')$ for t' from $-\infty$ to t] is necessary in order to find the present state of the system. Thus, the simulation is performed by starting at an appropriate initial time and then advancing time simultaneously everywhere in the system. This is the well-known way in which the analog computer performs the simulation and in which all the computer programs for simulation of dynamic systems work, in principle.

For spatial systems simulation, however, we must proceed from block to block and always determine the entire output function $v(x)$ from the entire input function $u(x')$. Consequently, we cannot treat the space coordinate x as equivalent to the time coordinate t in a simulation procedure. Thus, we cannot use dynamic systems simulation programs and apply them to spatial systems simulation. The need for image-forming systems simulation programs is apparent; and the next section describes the one we have developed.

* We consider here an incoherent system, i.e., a system which is linear in the light intensities. In contrast, a coherent system is linear in the complex amplitudes.

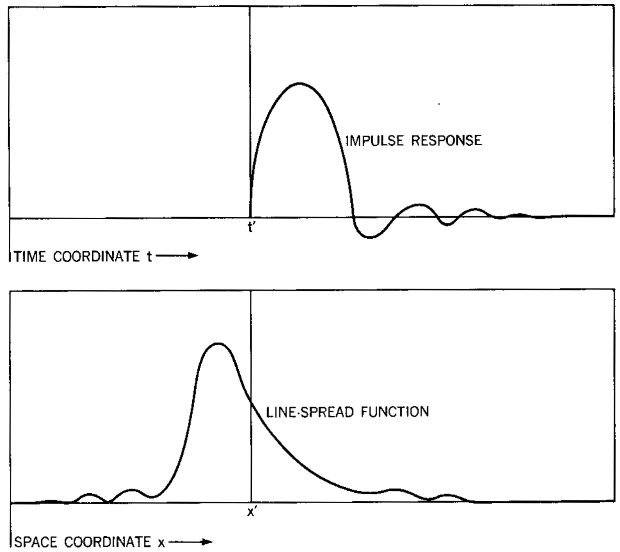

Figure 1 Impulse response and line-spread function: responses of a linear block to the unit impulse (δ-function) for a continuous dynamic and a spatial system.

Image-forming systems simulation program

In optical systems analysis, it is generally accepted practice to test the performance of the system for two separate one-dimensional objects, both varying in perpendicular directions. For example, in the image plane of a lens, structures which vary radially are often better reproduced than those varying tangentially. Therefore, it suffices if the simulation program treats image formation for objects that vary in one dimension only.

A digital computer program, the Image Forming Systems Simulator IMSIM/1, has been developed and coded in FORTRAN IV. This program accepts as input a description of an image-forming system, written in a simple problem-oriented language. Programs written in this language consist of a sequence of statements. The first statement describes the desired object, that is, the one-dimensional distribution of the object intensity, photographic density, or transmittance. Standard objects provided are sine, cosine, crenelate (square wave), line source (Dirac δ-function), slit, Gaussian function, and Dirac comb. It is also possible to describe the object by a tabulated function.*)

After the object statement, there follow statements containing descriptions of the spatial blocks together with associated parameter values. Examples of available blocks are: perfect lens (that is, diffraction-limited lens with defocusing and longitudinal vibration); photo-

* The use of the program and the preparation of the data are described in detail in the documentation accompanying the program package.

graphic emulsion; linear and random image motion; and transverse vibration. If the user wants to provide the block characteristic in tabulated form (for example, a line-spread function, a transfer function, or an H & D curve), means are available to do so. The sequence of the IMSIM/1 statements must correspond to the sequence of the spatial systems blocks in the order in which the light passes through them on its way from the object to the image. The execution of an IMSIM/1 program will result in calculating the image produced by the specified one-dimensional object.

In any sequence of linear spatial blocks, we can make use of the convolution theorem. According to this, the input-output relation can be written as $\tilde{v}(R) = \tilde{u}(R)\tilde{A}(R)$, where "~" denotes Fourier transform and R is the spatial frequency measured in cycles per unit length. The Fourier transform of the line-spread function, $\tilde{A}(R)$, is the transfer function of the block. Repeated application of this multiplicative relation means that for a sequence of linear blocks the transfer functions can be multiplied together. Thus we replace the multiple convolutions by much simpler multiplications. This concept has been used in the simulation program. If, then, a nonlinear block is encountered, the program automatically performs the necessary Fourier transformation from the frequency domain to the space domain. Also, if there is a linear block following a nonlinear one, the necessary Fourier transformation is automatically executed.

The available nonlinear blocks provide application of a point-to-point nonlinearity of the form $v = F(u)$. The nonlinear function F may be supplied by the user in tabular form.

In order to allow the user to study the influence of particular blocks on the performance of the system, IMSIM/1 provides means for supplying to the program several sets of parameters for each block instead of one set only. An important application of this feature is the study of the performance of a lens system across its image field. Tolerances for depth of focus are determined easily by the use of different amounts of defocusing. The resolution of an image-forming system can be obtained by using periodic object functions with various spatial frequencies.

IMSIM/1 allows the user to request printing or plotting of intermediate images produced by any one of the blocks and thus enables him to find the weakest link in the chain of system components. He may request as output the space domain or the frequency domain, and the program automatically performs any necessary Fourier transformations.

Figure 2 shows an example of an IMSIM/1 program. Each statement is punched on one card, whereby blanks may be inserted freely. Parameter values may be written as signed or unsigned decimal numbers, with or without decimal point. The statements of Fig. 2 describe the

```
  JOB 12345    CAMERA EXAMPLE  (RESCLUTICN OF 3-BAR-TARGET)

APERIODIC TABLE  (1, .5, 1.0)  (1, .25, 1.0)  (1, .125, 1.0)   EVEN
   PLOT SPACE (1) (2) (3)  SP
DENSITY TRANSMITTANCE CONVERSION
MAGNIFICATION  (.1)
PERFECT LENS  (250)
EMULSION SPREAD  (100)
CHARACTERISTIC CURVE (2, 1.25)
   PLOT SPACE (12) (22) (32)  S P
TABLE (1,1)  (0,0) (0,-0.9) (.2,-0.9) (.2,0) (.4,0) (.4,-0.9)
             (.6,-0.9) (.6,0) (.8,0) (.8,-0.9) (1,-0.9) (1,0)
TABLE (2,3)  (0,.08) (.25,.1) (.5,.27) (.75,.69) (1,1.47)
             (1.25,2.5) (1.5,3.14)
END
```

Figure 2 IMSIM/1 program for the simulation of a camera.

simulation of a simple camera with a three-bar target as object. The target is imaged by a diffraction limited lens (PERFECT LENS) with a reduction in size of 10:1. The target is described to the program in terms of its photographic density distribution by the APERIODIC TABLE statement in connection with TABLE 1. The three groups of parameter values, each enclosed in parentheses, signify that there are three runs to be executed. The target sizes differ by a factor of one-half between consecutive runs. Two output statements (PLOT SPACE) provide graphs of the object and image for each of the three runs.

The graphs produced by the example program are shown in Fig. 3. The first graph of each run shows the target: three lines with a photographic density of 0.1 on a dark background of density 1. The second graphs show the images, which are negatives of the object. The sequence of three runs demonstrates the deterioration of the image with decreasing size of the target (note the different abscissa scales between second and third run).

Execution time on an IBM 7094 Data Processing System was 27 seconds. This includes time required for creation of the plotting tape, but not for execution of the off-line plotting.

Summary

In this communication we discussed briefly the difference between spatial and dynamic systems. Because the impulse response of a dynamic block and the impulse response (spread function) of a spatial block differ in a fundamental way, spatial systems cannot be simulated by applying the techniques developed for dynamic systems simulation. Therefore, there is a need for spatial systems simulation programs.

We then described the Image Forming Systems Simulator IMSIM/1, a computer program developed for the simulation of incoherent photo-optical systems. Such a program provides a convenient tool for the photo-optical design engineer to study the performance of photo-optical systems before they actually are built.

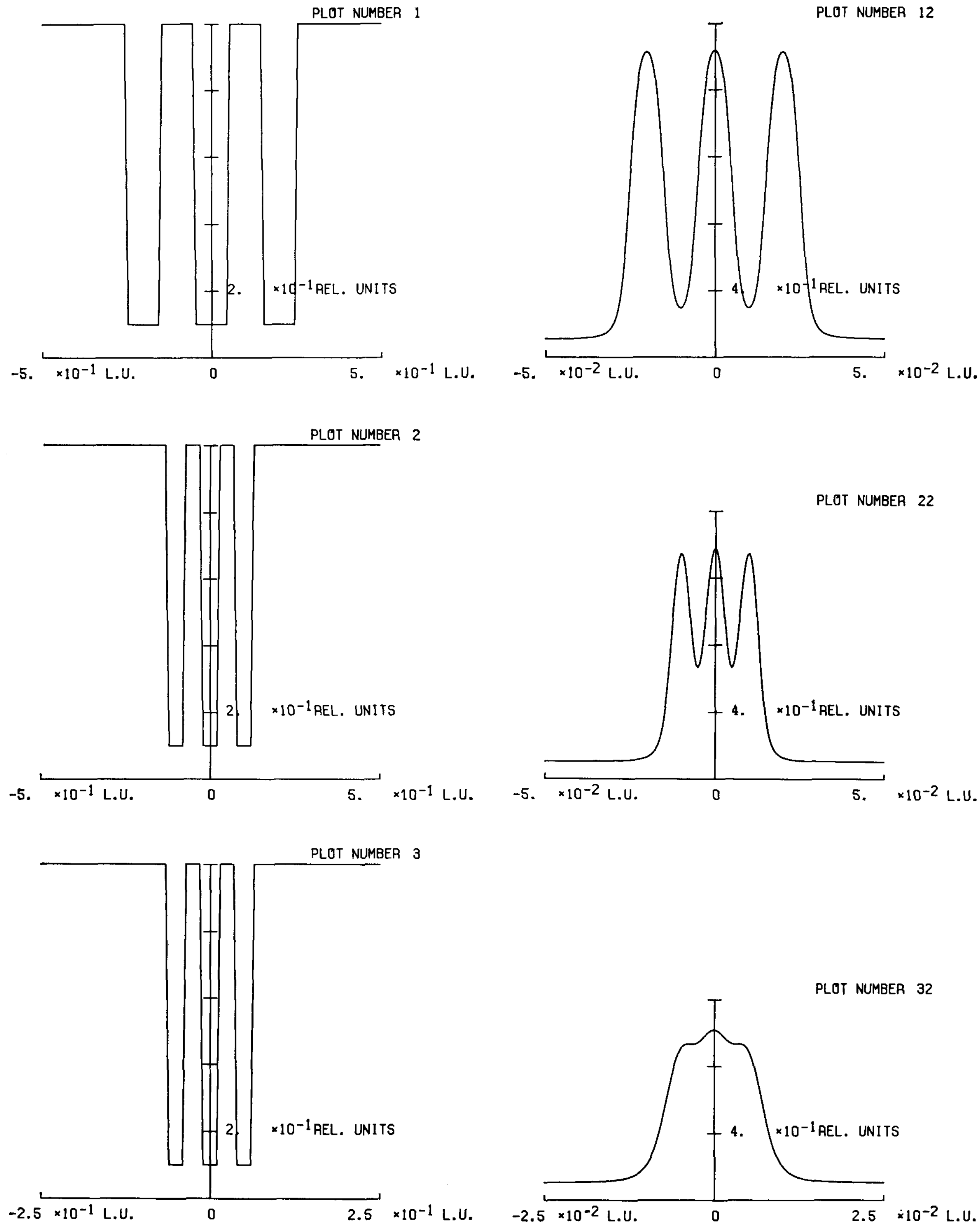

Figure 3 Graphs produced by the camera example of Figure 2. The plot Nos. 1, 2, and 3 represent the objects, while Nos. 12, 22, and 32 represent the images on the photographic material. (Ordinates are photographic densities.)

References

1. R. D. Brennan and R. N. Linebarger, *Simulation* 3, 22 (1964).
2. P. M. Duffieux, *L'integrale de Fourier et ses applications à l'optique*, Besancon, 1946 (private publication).
3. E. L. O'Neill, *Introduction to Statistical Optics*, Addison-Wesley Publ., Reading, Massachusetts, 1963.
   E. H. Linfoot, *Fourier Methods in Optical Image Evaluation*, The Focal Press, London/New York, 1964.
4. F. H. Perrin, *J. of the SMPTE* **69**, 151 and 239 (1960).
5. H. H. Hopkins, *Proc. Phys. Soc.* 79, 889 (1962).
6. J. L. Simonds, *Phot. Sci. and Eng.* **8**, 172 and 174 (1964).
7. M. E. Rabedeau, IBM Research Laboratory, San Jose, California. Private communication.
8. D. P. Paris, *Phot. Sci. and Eng.* **10**, 69 (1966).
9. S. Lerman and R. Shannon, *Annual Conference of Photographic Science and Engineering*, published by the Society of Photographic Scientists and Engineers, Washington, D. C. (1966), p. 137.
   S. Lerman and W. Minnick, *ibid.*, p. 140.
10. R. G. Gray and B. W. Kippenhan, *Annual Conference of Photographic Science and Engineering*, (1966) p. 58.
11. P. Elias, *J. Opt. Soc. of America* **43**, 229 (1953).

Received April 25, 1966