(d.1) CMA1-2 from Prob. 5.25, where γ is now chosen as $\gamma = \mathsf{E}\,|s|^2/\mathsf{E}\,|s|$ in terms of the second moment of the symbol constellation divided by the mean of magnitude of the symbols. For 16-QAM data we find $\gamma = 3.3385$ (verify this value). Use $\mu = 0.0001$ and increase the SNR level at the input of the equalizer to 60 dB. Simulate for 30000 iterations and plot the scatter diagram of the output of the equalizer after ignoring the first 15000 samples.

(d.2) The reduced constellation algorithm (RCA) of Prob. 5.23, where γ is now chosen as $\gamma = \mathsf{E}\,|s|^2/\mathsf{E}\,|s|_1$, in terms of the second moment of the symbol constellation divided by the mean of the l_1-norm of the symbols (remember that the l_1 norm of a complex number amounts to adding the absolute values of its real and imaginary parts, as in Prob. 4.13). For 16-QAM data we find $\gamma = 2.5$ (verify this value). Use the same step-size and same simulation duration as part (d.1).

(d.3) The "stop-and-go" algorithm is a blind adaptation scheme that employs the decision-directed error,

$$e_d(i) = \check{s}(i-\Delta) - z(i)$$

It also employs a flag to indicate how reliable $e_d(i)$ is. The flag is set by comparing $e_d(i)$ to another error signal, say the one used by the RCA recursion,

$$e_s(i) = \gamma \mathsf{csgn}(z(i)) - z(i), \qquad \gamma = \mathsf{E}\,|s|^2/\mathsf{E}\,|s|_1$$

If the complex signs of $\{e_d(i), e_s(i)\}$ differ, then $e_d(i)$ is assumed unreliable and the flag is set to zero (see Picchi and Prati (1987)). More explicitly, the stop-and-go recursion takes the form:

$$\left\{\begin{array}{ll} z(i) = u_i w_{i-1} & \\ e_d(i) = \check{s}(i-\Delta) - z(i) & \text{(decision-directed error)} \\ e_s(i) = \gamma \mathsf{csgn}(z(i)) - z(i) & \text{(RCA error)} \\ f(i) = \left\{\begin{array}{ll} 1 & \text{if } \mathsf{csgn}(e_d(i)) = \mathsf{csgn}(e_s(i)) \\ 0 & \text{if } \mathsf{csgn}(e_d(i)) \neq \mathsf{csgn}(e_s(i)) \end{array}\right. & \text{(a flag)} \\ e(i) = f(i) e_d(i) & \\ w_i = w_{i-1} + \mu u_i^* e(i) & \end{array}\right.$$

Use the same step-size and simulation duration as in part (d.1).

The programs that solve this problem are the following.

1. partA.m This program solves part (a) and generates two plots. Figure 5.28 shows the impulse responses of the channel, the equalizer, and the combination channel-equalizer. Observe that the latter has a peak at sample 19 so that the delay introduced by the channel and equalizer is 19. Figure 5.29 shows a typical plot of the scatter diagrams at transmission, reception, and output of the equalizer.
2. partB.m This program solves part (b) and generates three plots. Figure 5.30 shows the impulse responses of the channel, the equalizer, and the combination channel-equalizer. Figure 5.31 shows a typical plot of the scatter diagrams at transmission, reception, and output of the equalizer. Observe that although symbols from 16−QAM do not have constant modulus, CMA2-2 still functions properly; albeit more slowly. Figure 5.32 shows the real and imaginary parts of 50 transmitted symbols and the corresponding 50 decisions by the slicer (with the delay of 19 samples accounted for in order to synchronize the signals). These 50 symbols are chosen from the later part of the data after the equalizer has converged.
3. partC.m This program solves part (c) and generates three plots as in part (b). Here we only show in Fig. 5.33 the resulting scatter diagrams for MMA. Observe how the scatter diagram at the output of the equalizer is more focused when compared with CMA2-2.
4. partD.m This program solves part (d) and generates scatter diagrams for three additional blind adaptive algorithms. Typical outputs are shown in Fig. 5.34.

algorithm is examined in Computer Project 5.3. In Yang, Werner, and Dumont (1997), a multi-modulus algorithm (MMA) was introduced that is suitable for two-dimensional modulation schemes (see Prob. 5.24); this algorithm was studied in the context of broadband access systems in Werner et al. (1999).

The article by Johnson (1998) provides a survey of the developments in the area of blind algorithms until the late 1990s, including applications to fractionally-spaced equalization (see also Mai and Sayed (2000)). Constant modulus algorithms have also been used for adaptive beamforming (e.g., Keerthi, Mathur, and Shynk (1998)) and for interference cancellation (e.g., Kwon, Un, and Lee (1992)).

Variable step-size LMS. As we are going to see in Chapters 6 and 9, the step-size in LMS adaptation controls the rate of convergence and the steady-state performance of the filter. In order to meet the conflicting requirements of fast convergence and good steady-state performance, the step-size needs to be controlled. Various schemes for controlling the step-size of LMS have been proposed in the literature, for instance, by Kwong and Johnston (1992), Mathews and Xie (1993), Aboulnasr and Mayyas (1997), Pazaitis and Constantinides (1999), and Shin, Sayed, and Song (2003a). Kwong and Johnston (1992) use squared instantaneous errors, while Aboulnasr and Mayyas (1997) use the squared autocorrelation of errors at adjacent times, Pazaitis and Constantinides (1999) adopt the fourth-order cumulant of the instantaneous error, and Shin, Sayed, and Song (2003a) attempt to maximize the decrease in the weight-error-vector energy. The step-sizes in these variants are evaluated in the manner shown in Table 5.14 (for the case of real data):

Table 5.14. Three variable-step-size LMS implementations.

VSS-LMS Kwong and Johnston (1992)	$\mu(i) = \alpha\mu(i-1) + \gamma e^2(i)$
RVS-LMS Aboulnasr and Mayyas (1997)	$\mu(i) = \alpha\mu(i-1) + \gamma p^2(i)$ $p(i) = \beta p(i-1) + (1-\beta)e(i)e(i-1)$
KVS-LMS Pazaitis and Constantinides (1999)	$\mu(i) = \mu_{\max}\left(1 - e^{-\alpha C_4^e(i)}\right)$ $C_4^e(i) = f(i) - 3p^2(i)$ $f(i) = \beta f(i-1) + (1-\beta)e^4(i)$ $p(i) = \beta p(i-1) + (1-\beta)e^2(i)$
VSS-NLMS Shin, Sayed, and Song (2003a)	$\mu(i) = \dfrac{\mu_{\max}}{\epsilon + \|u_i\|^2} \cdot \dfrac{\|p_i\|^2}{c + \|p_i\|^2}$ $c \approx 1/\mathsf{SNR}$ $p_i = \beta p_{i-1} + (1-\beta)\dfrac{u_i}{\epsilon + \|u_i\|^2}e(i)$

Adaptive equalization. In the concluding remarks of Chapter 3, we commented on the early history of channel equalization and how the mean-square error criterion was proposed for this purpose by Widrow (1966), Gersho (1969), and Proakis and Miller (1969). In this chapter, we described how channel equalizers could be designed adaptively, as opposed to the closed-form solution methods of Secs. 2.7.3 and 3.3. Historically, the LMS algorithm was developed at the right time, in the early 1960s, right when interest in equalization was starting to build up (Lucky (1965)). Soon afterwards, the use of adaptive filters for equalization purposes was studied in greater detail by Gersho (1969), Proakis and Miller (1969), Ungerboeck (1972), and Salz (1973). In these adaptive implementations, an equalizer would have two modes of operations: a training mode (Gersho (1969)) whereby data that are known to the receiver and the transmitter are used to train the equalizer, and a decision-directed mode (Salz (1973)), whereby decisions are used to train the equalizer.