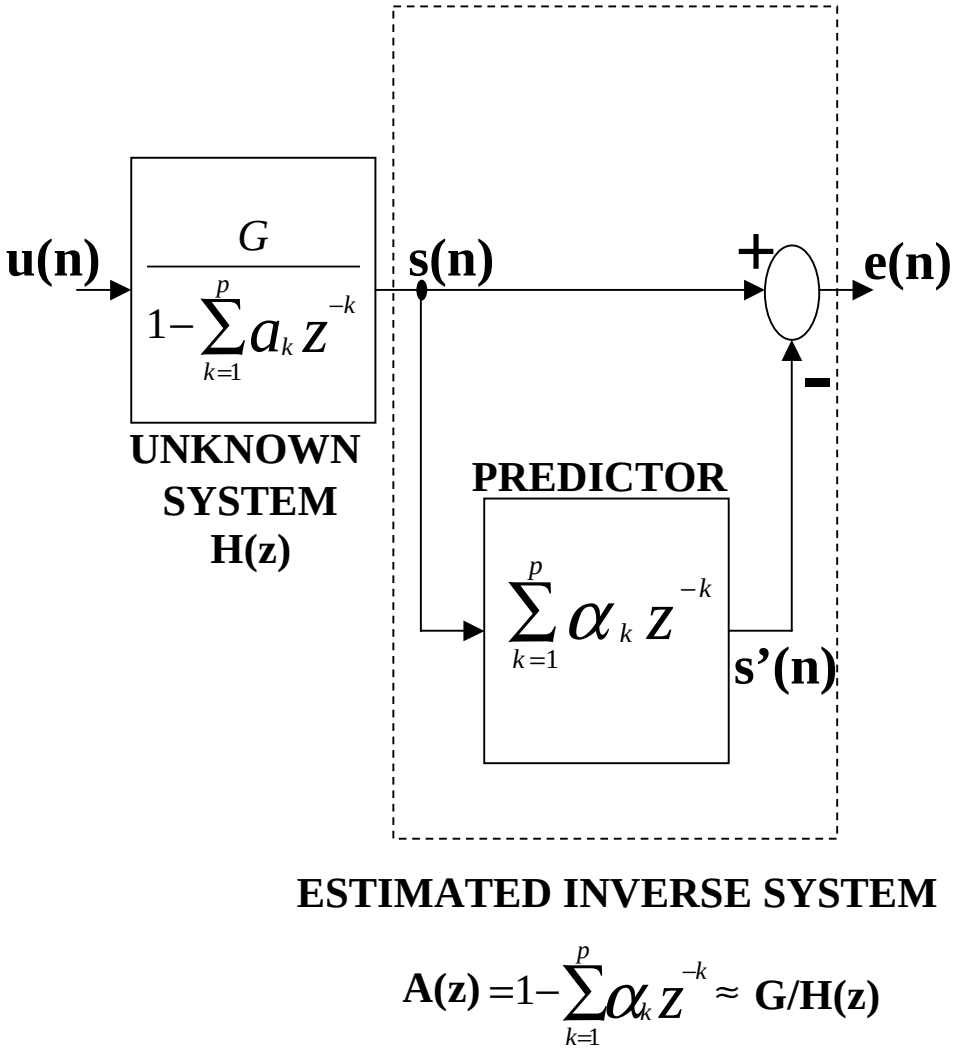

Figure 2.2. Block diagram of the process of LP. An unknown system H(z), assumed to be modeled by an all-pole resonator chain, is driven by a signal u(n). Only the output s(n) is observable. A predictor FIR filter is derived via least squares minimization between s(n) and the predictor output s'(n). The predictor is used to form A(z), which approximates an inverse system to H(z), and generates the error signal e(n). If u(n) is impulsive, A(z) models H(z) well. If, however, s(n) incorporates dynamics from both vocal tract and source signal, A(z) is unable to distinguish source from signal via LP alone.

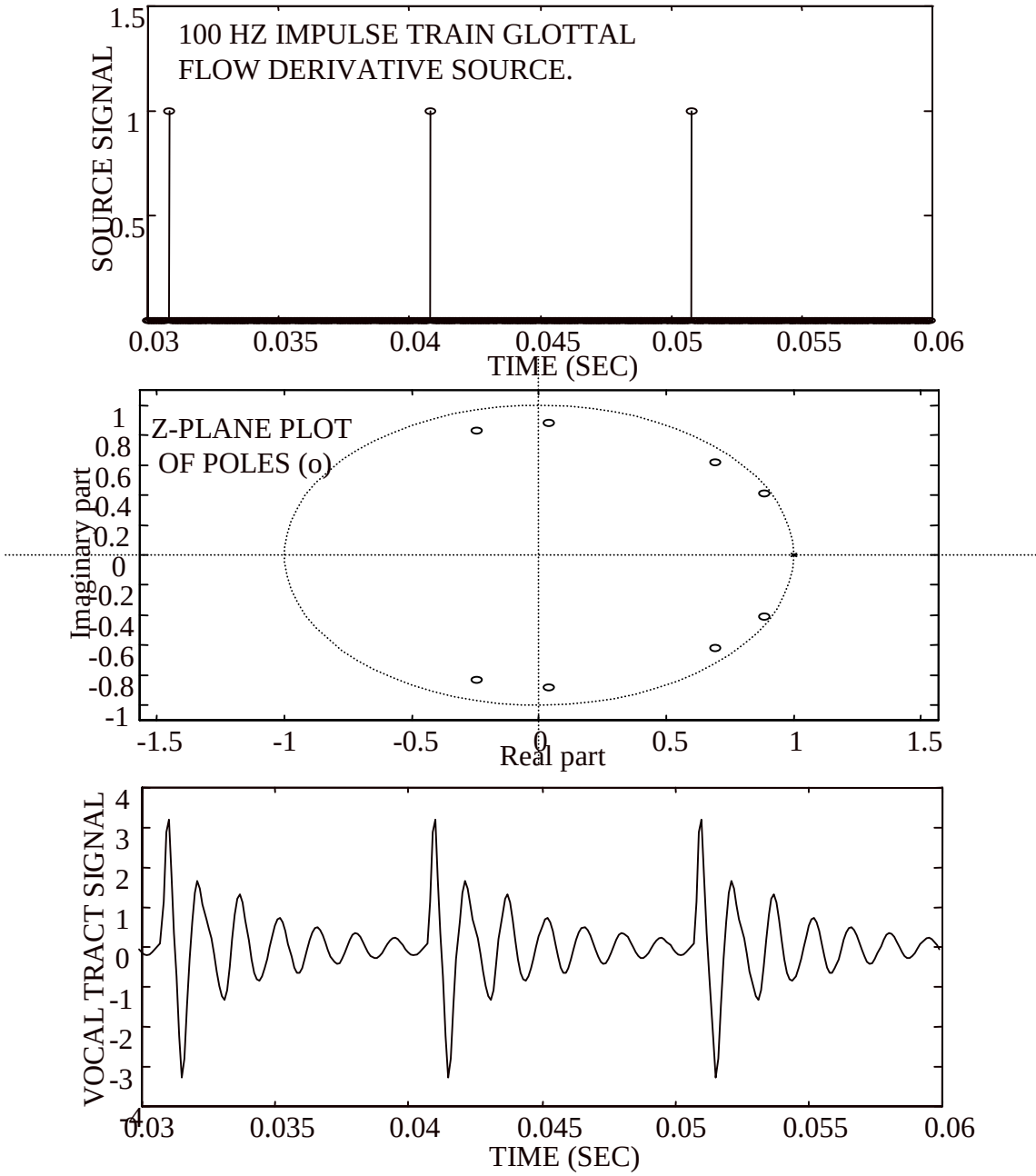

Figure 2.3. Synthetic impulse train response of an 8-pole vocal tract model for /a/. The top plot displays the input impulse train, the middle plot shows the pole locations for the all-pole resonator vocal tract model, and the bottom plot shows the resulting voice time series.

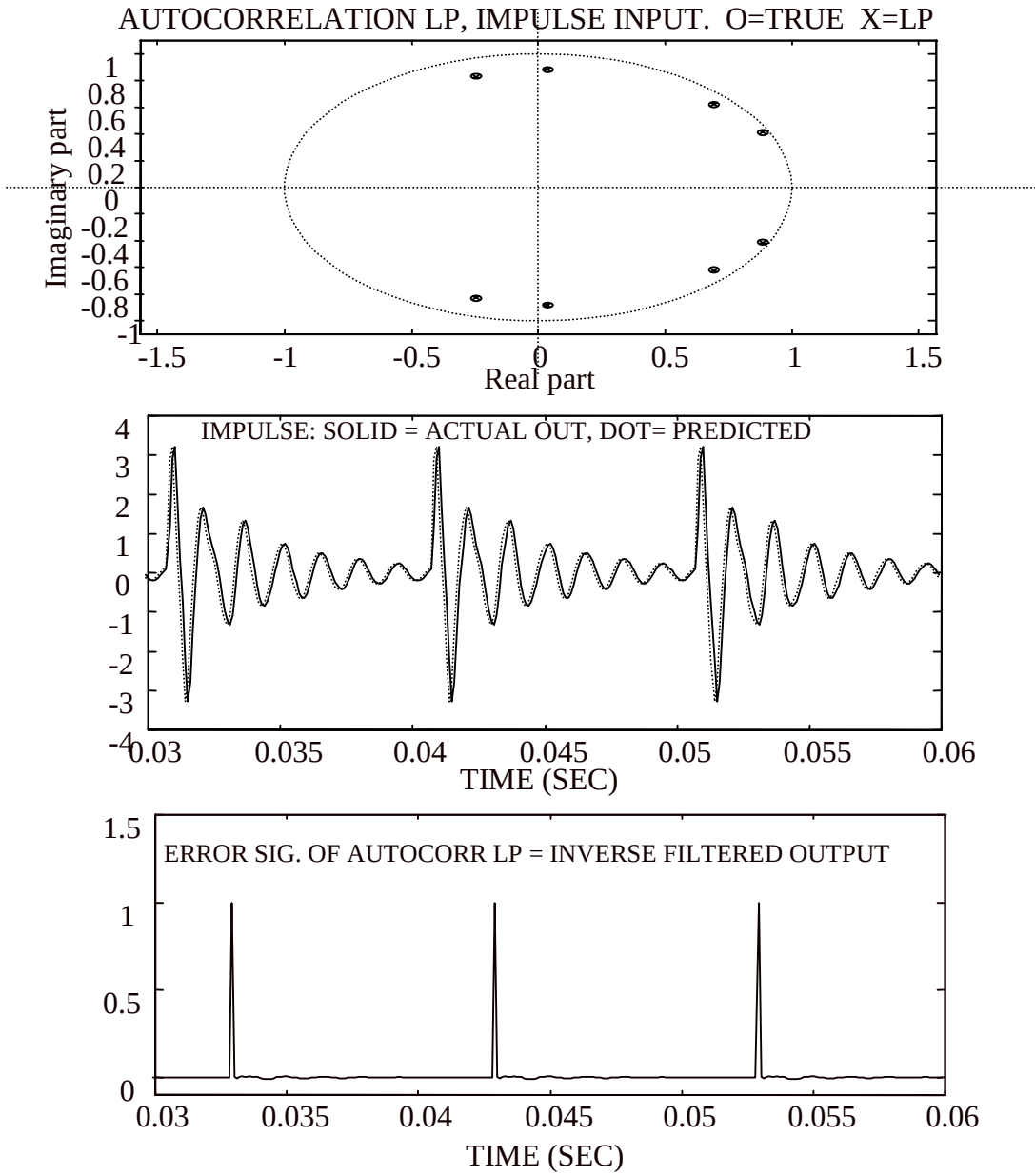

Figure 2.4. Results of autocorrelation LP on the system of Fig 2.3. The top plot repeats the actual pole locations (o) and shows the poles (x) estimated via autocorrelation LP on 10 cycles of the output voice windowed with a hamming window; the estimated poles are within 1% of the actual poles. The middle plot shows the actual voice output (solid) and the predictor output; again, there is close agreement. The bottom plot displays the error signal, which closely agrees with the actual impulse train driving function.

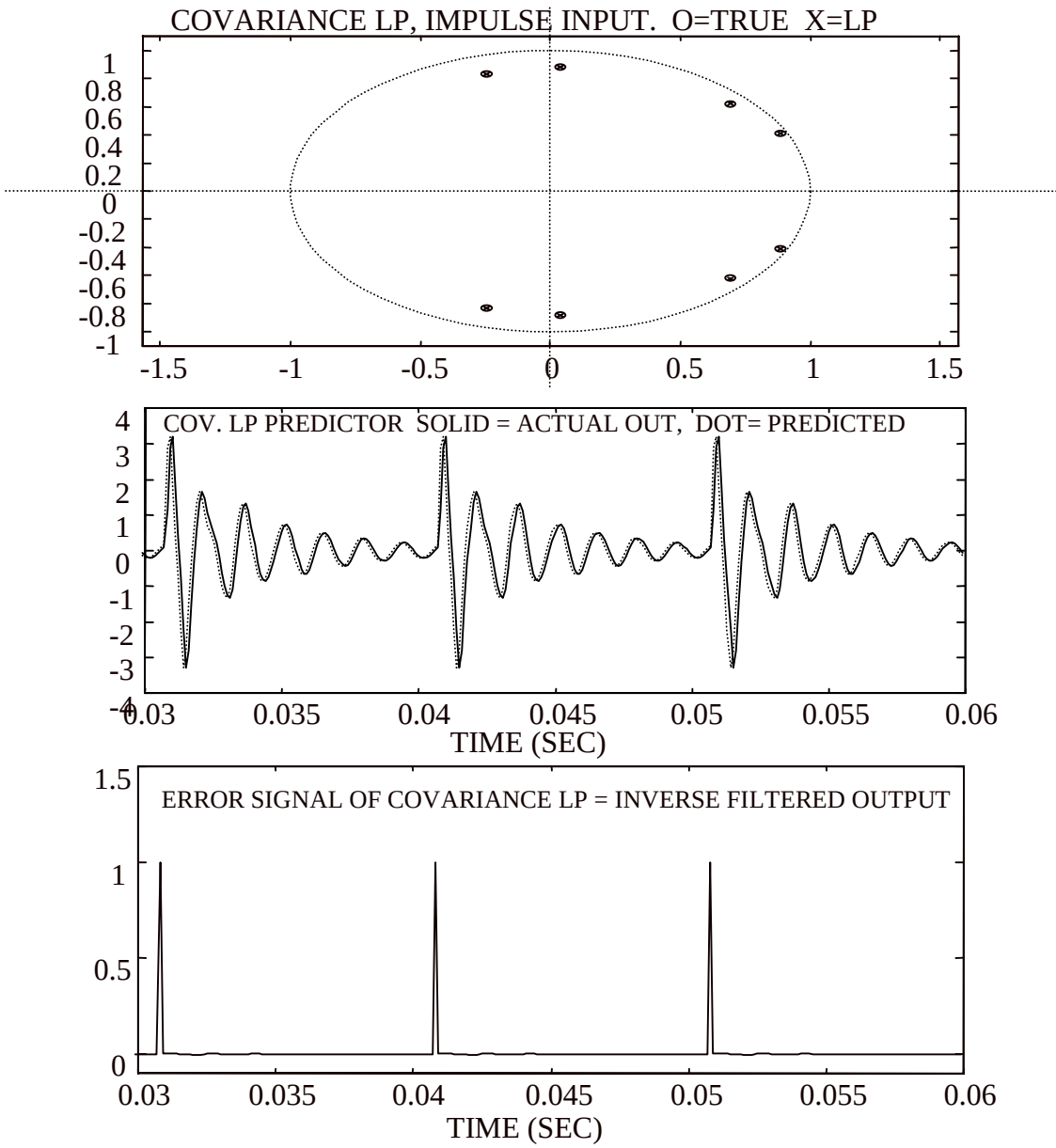

Figure 2.5. Results of covariance LP on the system of Fig 2.3. The top plot repeats the actual pole locations (o) and shows the poles (x) estimated via covariance LP on a 70 sample window of the output voice, which includes an impulse and its response. The estimated poles are within 1% of the actual poles. The middle plot shows the actual voice output (solid) and the predictor output; again, there is close agreement. The bottom plot displays the error signal, which closely agrees with the actual impulse train driving function.

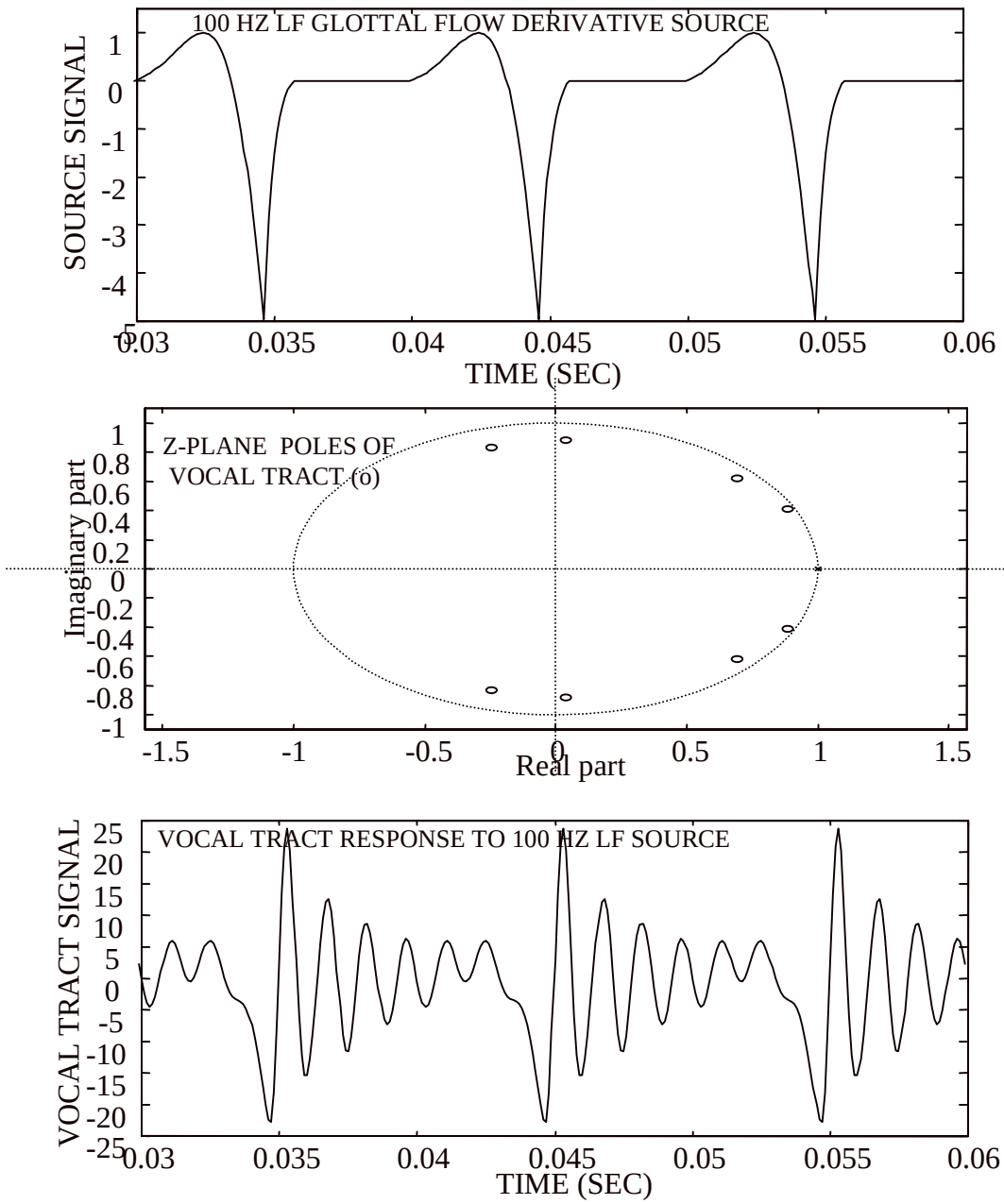

Figure 2.6. Synthetic LF source response of the same system of Fig. 2.3. The top plot displays the input LF source time series, the middle plot shows the pole locations, and the bottom plot shows the resulting voice time series.

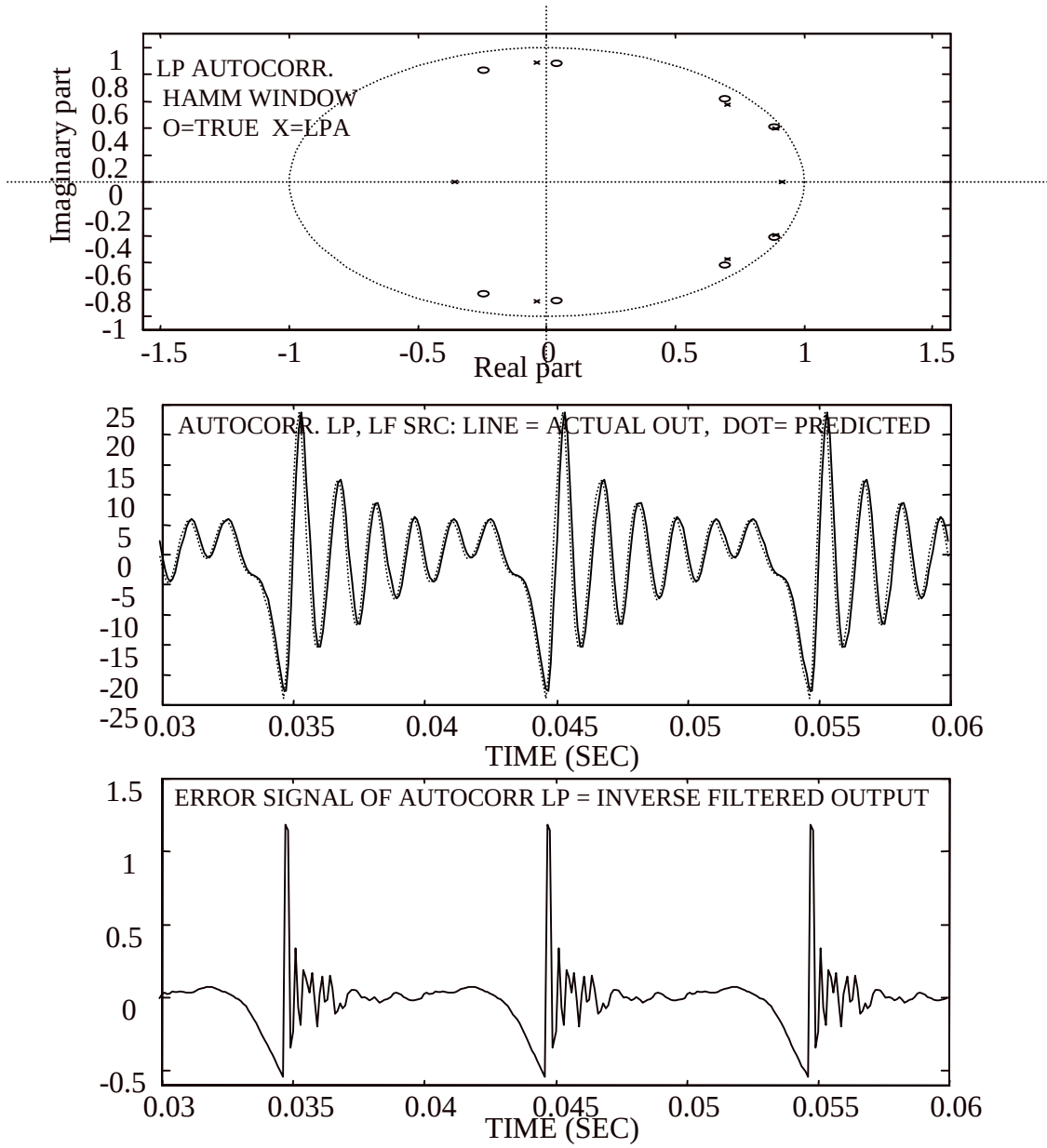

Figure 2.7. Result of autocorrelation LP on the LF source model system of Fig. 2.6; compare directly with Fig 2.4. The top plot of pole locations of estimate (x) and actual (o) reveals considerable error, especially at the higher frequencies; in fact, one of the complex pairs has become 2 real roots. The middle plot, however, reveals LP is still performing its intended primary function of prediction quite well. The bottom plot shows that, because of the error in pole locations, the inverse filter works poorly (compare to top plot of Fig 2.6).

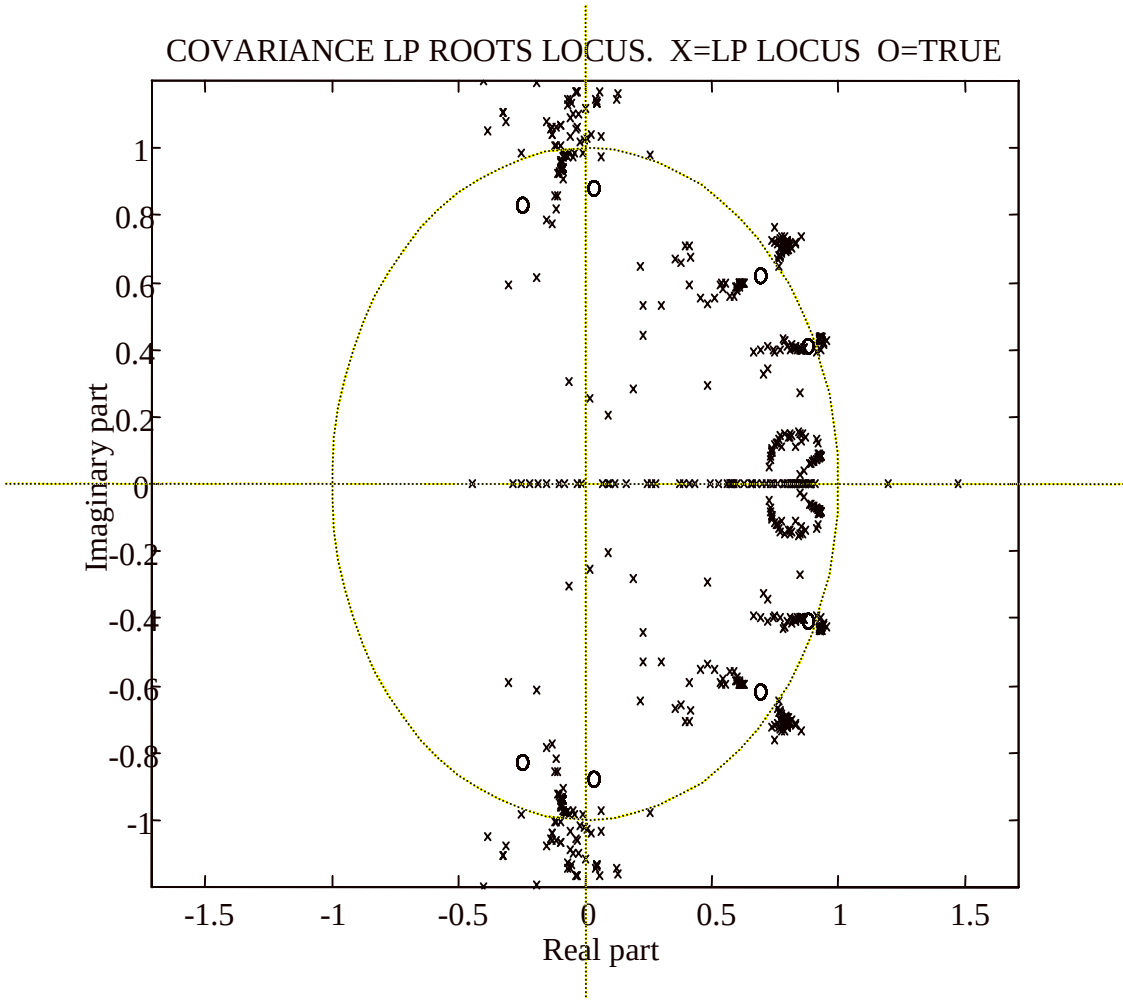

Figure 2.8. Covariance LP poles for a range of analysis window positions. The analysis window length is set to 40 samples, and the position of the window is swept through one complete fundamental period, generating and plotting the resulting LP pole positions for each position. Note that calculated positions (x) never reach true positions (o). The same is true of other selections of window length.

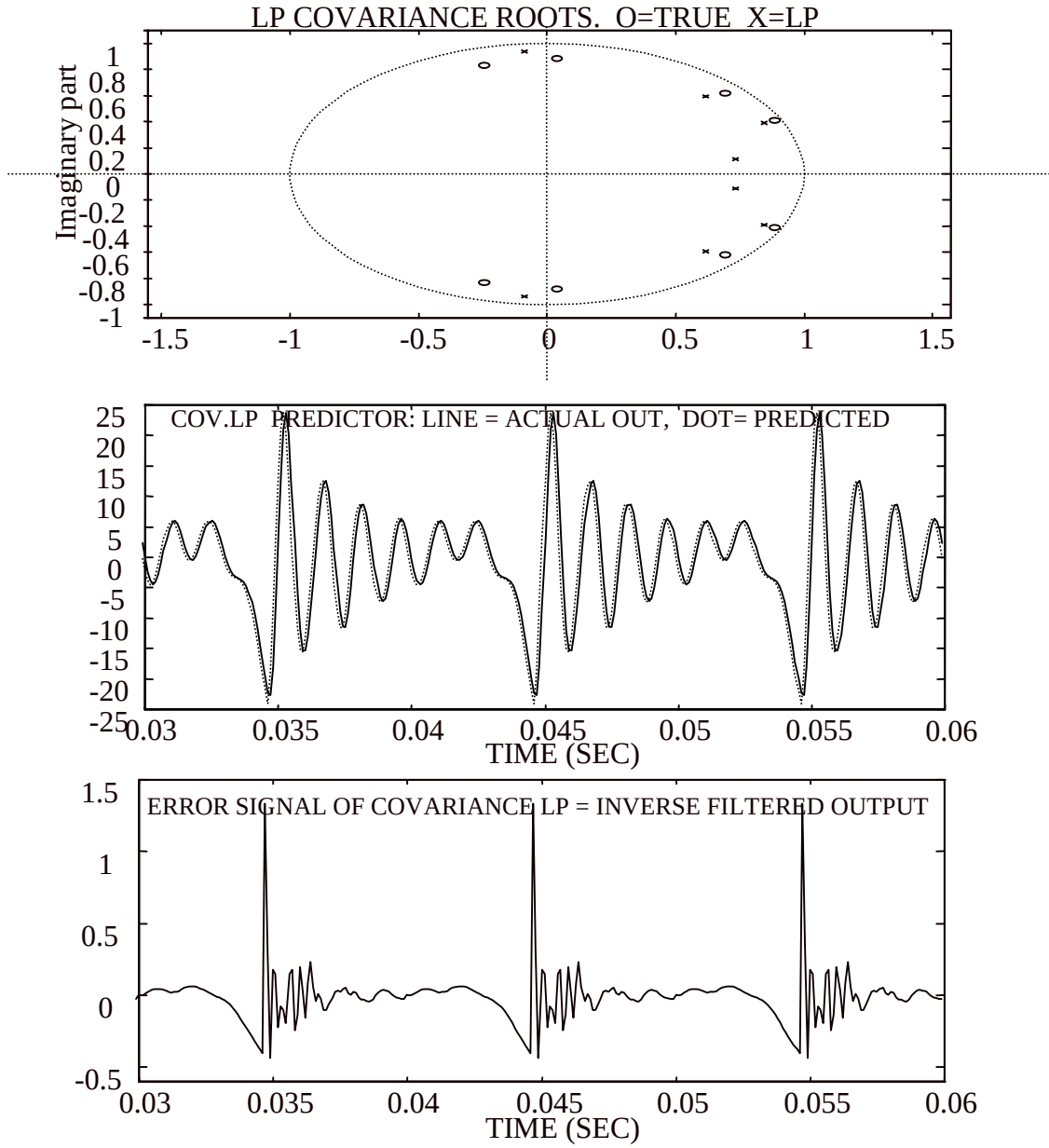

Figure 2.9. Result of covariance LP on non-impulsive system of Fig. 2.6 for an optimal analysis window position; compare with Fig 2.7. The top plot of pole locations of estimate (x) and actual (o) still reveals considerable error. The middle plot again reveals LP is still performing its intended primary function of prediction. The bottom plot again shows that, because of the error in pole locations, the inverse filter works poorly (should be like the top plot of Fig 2.6).

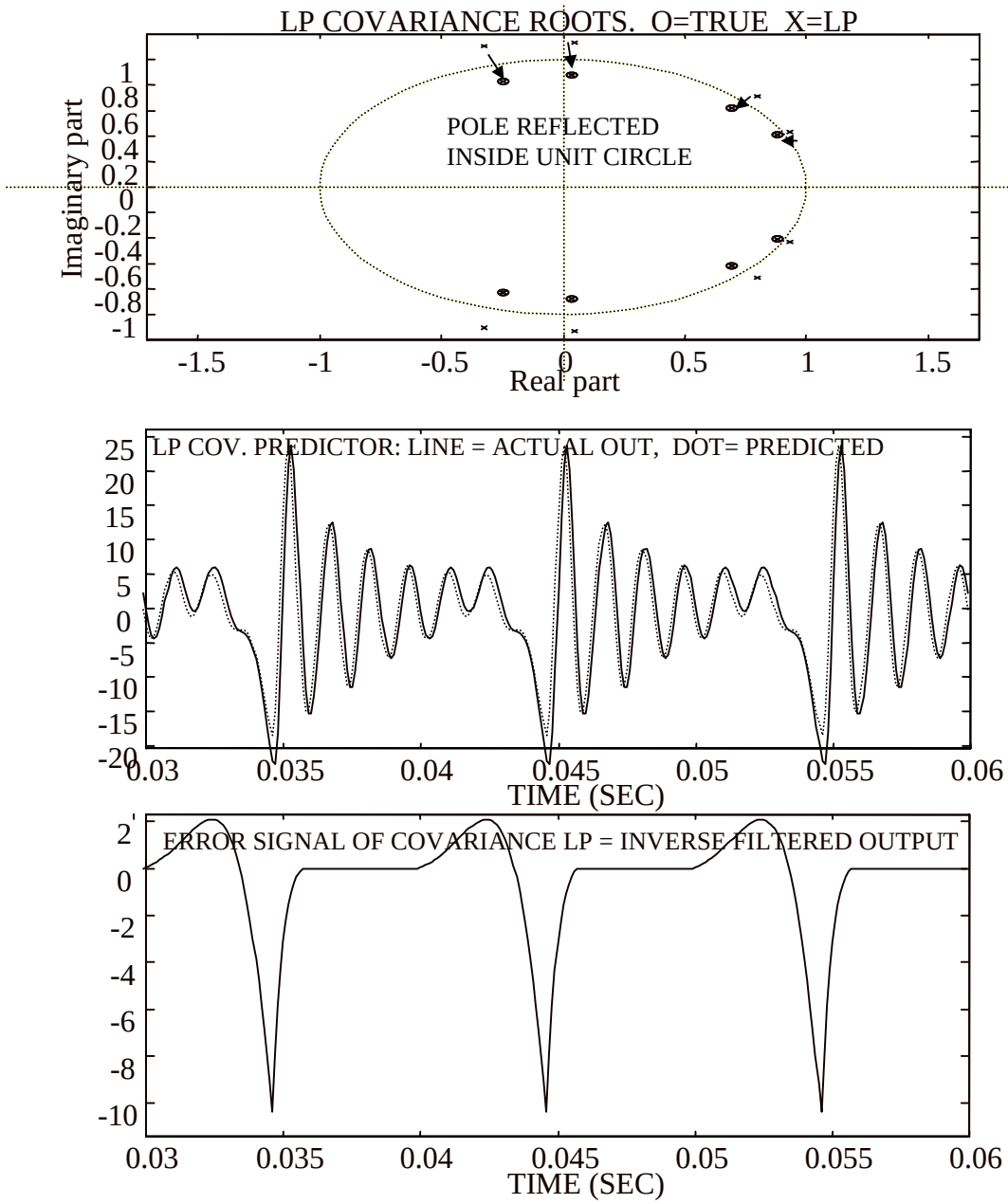

Figure 2.10. Reflection of covariance LP poles inside unit circle. For the same analysis of Figs 2.8, the top plot shows the poles outside the unit circle are inverted to reflect them inside; the resulting new positions coincide almost exactly with the true positions. Prediction (middle plot) is somewhat less accurate, but good inverse filtering is achieved (bottom plot.)