Chapter 4

Dynamics on Nonlinear Manifolds

Proper Orthogonal Decomposition generates linear spaces onto which we may project system dynamics. However, the underlying space in which dynamics actually occur may not be linear. Instead, we may wish to find a more general manifold on which the dynamics occurs. To find such a manifold from a quantum trajectory, we turn to results from the field of unsupervised manifold learning. "Unsupervised" in this context means that the process of finding and defining a manifold is algorithmic, although as we will see there is still some room for researcher intervention and intuition (e.g., determining the dimension of the manifold for which the algorithm is searching, and how many neighbors to include in the definition of a point's "neighborhood"). There have been several manifold learning algorithms proposed in recent years in the machine learning community and elsewhere (including [32, 35, 36, 37, 38, 39, 40] and references therein). These manifold learning algorithms share a focus on locally linear manifolds, which are then stitched together into a global manifold definition. As a result, the manifold ends up being defined locally, with a different function at each point mapping from the full space onto the manifold.

Of the algorithms we considered, we find the Local Tangent Space Alignment algorithm from Zhang and Zha [32] to be the most satisfying. Most prominently, its definition of the tangent space at each point has more rigorous differential geometry treatment. This is intimately related to the fact that the algorithm starts by doing what is effectively Proper Orthogonal Decomposition at each point to define the optimal linear subspace. In fact, if we were to define the full sample as "nearest neighbors" of every point, the LTSA algorithm would reduce to POD.

4.1 Local Tangent Space Alignment

The Local Tangent Space Alignment algorithm for manifold learning was introduced by Zhang and Zha [32], and in this section we will briefly review that algorithm, clearly paralleling their derivation (but choosing only to describe the form of the algorithm I use in analyzing our cavity QED dynamics). The input for the algorithm is a set of points in an m-dimensional space, which we believe may fall

on or near a d-dimensional (likely nonlinear) manifold. Noise may make these points not fall exactly on the lower-dimensional manifold, but we assume that structureless noise will average out, allowing us to extract the underlying manifold.

We will construct the manifold by aligning the tangent spaces ($\mathcal{T}_\tau$) of each point (τ). If we had a function $f : \Omega \subset \mathbb{R}^d \to \mathbb{R}^m$ that mapped from the d-dimensional manifold Ω to the full space, we could use its Jacobian $J_f(\tau)$ at each point to find the tangent space. If Q_τ is an orthonormal basis of $\mathcal{T}_\tau$, $\bar{\tau}$ is a neighbor of τ, and $\theta^*_\tau(\bar{\tau})$ is the local coordinate of $\bar{\tau}$ in the basis Q_τ, then

$$J_f(\tau)(\bar{\tau} - \tau) = Q_\tau \theta^*_\tau(\bar{\tau}). \tag{4.1}$$

We would like to reconstruct the coordinates τ from the data sampled in the m-dimensional space, without knowing (or being forced to calculate) f. The alignment process will allow us to go from the local coordinates θ^*_τ to the global coordinates τ. Denote $P_\tau = Q_\tau^T J_f(\tau)$, and then we have

$$\theta^*_\tau(\bar{\tau}) = Q_\tau^T J_f(\tau)(\bar{\tau} - \tau) \equiv P_\tau(\bar{\tau} - \tau). \tag{4.2}$$

(Recall that Q_τ is orthonormal so that $Q_\tau^T = Q_\tau^{-1}$.) Now P_τ is a local map from the global coordinates to the local. The Jacobian is a creation of the first derivatives of f, so the local coordinates θ^*_τ tell us about the linear part of $f(\bar{\tau}) - f(\tau)$. We can approximate them instead by the orthogonal projection of $f(\bar{\tau}) - f(\tau)$ onto $\mathcal{T}_\tau$; we denote these projected coordinates $\theta_\tau(\bar{\tau})$. P_τ should be invertible because the Jacobian should be full rank, so we can denote its inverse L_τ, and we see that (to first order)

$$\bar{\tau} - \tau \approx P_\tau^{-1} \theta_\tau(\bar{\tau}) = L_\tau \theta_\tau(\bar{\tau}). \tag{4.3}$$

So, instead of finding the function f, instead we are looking for a set of coordinates τ, each with a local affine transformation L_τ, to minimize

$$\int_\Omega \left(\int_{\Omega(\tau)} \|\bar{\tau} - \tau - L_\tau \theta_\tau(\bar{\tau})\| d\bar{\tau} \Big/ \int_{\Omega(\tau)} d\bar{\tau} \right) d\tau \tag{4.4}$$

over all possible L_τ.

Moving one step closer to the practical implementation of the algorithm, let us now consider implementing this process with a given finite set of data points $X = [x_1, \ldots, x_N]$. In practice, the LTSA algorithm begins by defining a local neighborhood for each point, using some metric on the m-dimensional space. In the quantum trajectory case, I used the 2-norm distance between the vectorized versions of density matrices, themselves produced from trajectory wavefunctions. This is the same distance measure as in Proper Orthogonal Decomposition; however in this case we simply

want a collection of points in the neighborhood of each point, and the actual distances are not used. Another alternative would have been the 1-norm matrix distance, $|A - B|$, but I chose the 2-norm for consistency with POD and for calculation time (the 2-norm is fast to calculate, while $|A - B|$ requires calculating eigenvalues for every pair). Despite its poor scaling properties, I chose to use a brute-force method to generate these neighborhoods: I calculated the distances between every pair of points, and then sorted them (an order N^2 algorithm for N points). While some algorithms exist which scale better with respect to the number of points, they generally scale poorly in the dimension of the space. In our case, the density matrix space may have more than 10,000 dimensions, and we calculate all the distances between up to 2,000 points.

We would like to use the set of neighbors of each point τ to approximate the tangent space $\mathcal{T}_\tau$ at that point. In effect, we would like to do at each point what we did globally in Chapter 3: compute the best d-dimensional affine subspace approximation for the data points in the neighborhood of each point. Let us denote a matrix of the k nearest neighbors of x_i (including x_i) as $X_i = [x_{i_1}, \ldots, x_{i_k}]$. As in our discussion of Proper Orthogonal Decomposition, we could calculate the basis of this optimal subspace by calculating the eigenvectors corresponding to the largest d eigenvalues of the correlation matrix

$$\left(X_i - \bar{x}_i e^T\right)\left(X_i - \bar{x}_i e^T\right)^T \tag{4.5}$$

where $\bar{x}_i$ is the mean of the neighbors of x_i (recall that the optimal affine subspace will go through the mean), and e is a vector of length k and all ones (to make the dimensions work out). What we would like, instead of the definition of the basis vectors of the subspace, however, are the local coordinates of the points in that space, with that basis. These are calculated similarly, from the leading d eigenvectors of

$$\left(X_i - \bar{x}_i e^T\right)^T\left(X_i - \bar{x}_i e^T\right) \tag{4.6}$$

(The leading eigenvector, of length k, gives the coordinate of each of the k points in the direction of the leading basis vector from Eqn. (4.5), and so-on for each eigenvector.)

These local coordinates in the basis for each point, our $\theta_j^{(i)}$s ($j = 1, \ldots, k$, $i = 1, \ldots, N$), tell us about the local geometry. We would like global coordinates, τ_i, which respect that geometry:

$$\tau_{i_j} = \bar{\tau}_i + L_i \theta_j^{(i)} + \epsilon_j^{(i)} \tag{4.7}$$

where $\bar{\tau}_i$ is the mean of the k τ_{i_j}s, L_i is a local affine transformation matrix (to be determined), and $\epsilon_j^{(i)}$ is the local error. If we combine these k equations into one matrix equation, and denote the local coordinates Θ_i, $T_i = [\tau_{i_1}, \ldots, \tau_{i_k}]$, and $E_i = [\epsilon_1^{(i)}, \ldots, \epsilon_k^{(i)}]$, then we have

$$T_i = \frac{1}{k} T_i e e^T + L_i \Theta_i + E_i. \tag{4.8}$$

We would like to minimize the error term, so we should minimize

$$\sum_{i=1}^{N} \|E_i\|^2 = \sum_{i=1}^{N} \left\| T_i \left(I - \frac{1}{k} e e^T \right) - L_i \Theta_i \right\|^2 . \tag{4.9}$$

This looks familiar — again we calculate eigenvectors. However, this time we would like the eigenvectors which correspond to the smallest nonzero eigenvalues. First, though, we need to expand from the single point to consider all of the points making the full manifold. Let $T = [\tau_1, \ldots, \tau_N]$ and S_i be the selection matrix so that $TS_i = T_i$. S_i can, of course, be calculated from our global distance matrix, and does not require T (which we need S_i to solve for). Construct $S = [S_1, \ldots, S_N]$ and $W = \mathrm{diag}(\mathrm{W}_1, \ldots, \mathrm{W}_\mathrm{N})$ with

$$W_i = \left(I - \frac{1}{k} e e^T \right) \left(I - \Theta_i^+ \Theta_i \right) \tag{4.10}$$

where Θ_i^+ is the Moore-Penrose generalized inverse of Θ_i. Then the overall reconstruction error is given by

$$\sum_{i=1}^{N} \|E_i\|^2 = \|TSW\|^2. \tag{4.11}$$

This could of course be minimized by $T = 0$, so we require that $TT^T = I$. This allows us to construct the correlation matrix

$$B \equiv SWW^T S^T. \tag{4.12}$$

The vector of all ones (e) is an eigenvector of B with eigenvalue zero; the optimal T is given by the d eigenvectors with the 2nd to $d+1$st smallest eigenvalues.

From a numerical standpoint, what we need, then, is the matrix B. Rather than explicitly calculating with the selection matrix S, we simply calculate WW^T at each point and sum them up. Define $g_1, \ldots, g_d$ to be the unit eigenvectors corresponding to the d largest eigenvalues of Eqn. (4.6). Define $G_i = [e/\sqrt{k}, g_1, \ldots, g_d]$, and one can then verify that

$$W_i = I - G_i G_i^T. \tag{4.13}$$

Now see that $W_i W_i^T = W_i$, so that B may be computed by locally summing $I - G_i G_i^T$. This is in fact what we do when solving the algorithm numerically: at each point, we find G_i, and we then sum $I - G_i G_i^T$ over the full set of points to build the full B. Computing the eigenvectors of B with the d smallest nonzero eigenvalues gives us the global coordinates of each point in our new basis, resulting in a point-by-point definition of the manifold $\Omega \subset \mathbb{R}^d$. Zhang and Zha [32] provide an error analysis of the algorithm, and provide bounds for the reconstruction errors. For our purposes, it suffices that the errors are proven to be bounded, and can be quite small.

It should now be clear that LTSA effectively reduces to Proper Orthogonal Decomposition if we define the "neighborhood" of each point to include all the points. The additional degree of freedom provided by k, the number of neighbors to use, makes the LTSA algorithm more flexible, but also less exact. The same data set, processed with different values of k, will result in different manifold definitions. This makes it difficult to define a "definitive" manifold corresponding to a set of data. However, if we run the LTSA algorithm with a range of values of k (and assume that the dimensionality d is appropriate to the data), there should be a range of k over which the general structure of the manifold does not change. We assume that this structure is the "correct" result from the algorithm, and analyze one such result as representative of that data set. If the data set is well sampled, and the dimension d chosen is reasonable, we expect that a similar manifold structure and topology would be determined from any such data set sampled from the same dynamical system.

Unlike Proper Orthogonal Decomposition, the LTSA algorithm requires fixing the value of d prior to running the algorithm. In POD, one may simply choose the dimensionality of the calculated subspace at the end, by choosing the number of eigenstates/basis vectors to use. In LTSA, however, we calculate the d-dimensional space at each point, and then integrate them together (the form of G_i depends on d). While the costly distance calculation may be reused for different ds, the bulk of the algorithm must be run separately for each d. In practice, we do not know d in advance, and must run the algorithm multiple times, to see at what point d is sufficient. For example, take a case for which the inherent dimension of the manifold is 2. When run with $d = 3$, the resulting manifold should show no interesting structure in the third dimension: it should look like a flat 2-dimensional surface suspended in 3 dimensions. This is straightforward to see on screen for 2 and 3 dimensions; for higher dimensions one has to look at the variation across the sample in the d^{th} dimension, and make a decision as to whether it is simply noise from the input data set. In the end, we are simply lucky that the underlying structures for the dynamics in the cavity QED cases we consider seem to fit well in few dimensions, which allows us to proceed to build approximations to those dynamical systems which reside on or close to these manifolds.

4.2 Deriving dynamical systems on LTSA manifolds

Once we have a manifold determined from a set of trajectories, processed by the LTSA algorithm, we would like to derive a set of equations for the dynamics of reduced-order systems which live on that manifold. Unfortunately, because of the local nature of the LTSA algorithm, we do not have a global function which we can use to map the dynamics of the system of equations which generated the trajectories into dynamics on the lower-dimensional manifold. In particular, we cannot use the filter projection technique discussed in Chapter 2 without a functional form for the maps between the coordinates on the d-dimensional manifold and the coordinates in the m-dimensional space of

trajectories.

One option would be to find and utilize the local maps between the tangent spaces of each point and the larger space. This would be highly accurate (not much of an approximation). However, it would be impractical: as we calculate a new trajectory, we would need to find the nearest point from the originating trajectory at each time step, project the dynamical equations into its particular space, advance the point in local coordinates, revert to global coordinates, and repeat.

Another option, this time approximate, would be to attempt to utilize a least-squares fitting algorithm to find a suitable map from the d-dimensional manifold to the m-dimensional originating space. In the case of cavity QED, however, m is very large. For example, $m = 14,400$ for a 120×120 density matrix, while (as we will see below), d may be less than 5. More than m fit parameters would need to be calculated, requiring more than m samples from the trajectory. In addition, there is no clear way to decide *a priori* what functional form to assume for the fitting function, meaning either a very high-dimensioned guess with a linear form, or building out polynomials (with the requisite larger number of fit parameters).

The process I have chosen instead is to relax the requirement to reproduce the full density matrix from the global manifold coordinates (and *vice versa*), and instead focus only on the expectation values of operators. Here we are effectively assuming that the simple operators (those we have the possibility to measure) carry the information we care about regarding the system, and hoping that the manifold can be easily fit by a simple polynomials of these operators.

Carrying this plan forward, let us assume we have equations mapping expectation values to manifold coordinates, similar to this:

$$\tau_1 = v_1 \langle x \rangle + v_2 \langle y \rangle + v_3 \langle \sigma_x \rangle + ... + v_k \langle xy\sigma_x \rangle \langle x \rangle + ..., \tag{4.14}$$

where v_j are fit parameters, and we want an equation for motion for τ_1. That is, we would like an expression for $d\tau_1$ (and similar for $d\tau_2$ to $d\tau_d$).

In order to generate these equations, we need a general prescription for $d\langle \mathcal{O} \rangle$.

$$\begin{aligned} d\langle \mathcal{O} \rangle &= d\left(Tr\left(\mathcal{O}\rho\right)\right) && (4.15)\\ &= Tr\left(d\left(\mathcal{O}\rho\right)\right) && (4.16)\\ &= Tr\left(\left(d\mathcal{O}\right)\rho + \mathcal{O}\left(d\rho\right)\right). && (4.17) \end{aligned}$$

In the Schrödinger picture, $d\mathcal{O} = 0$ while $d\rho \neq 0$, so we really just need to know

$$d\langle \mathcal{O} \rangle = Tr\left(\mathcal{O}\left(d\rho\right)\right). \tag{4.18}$$

For our Cavity QED system, we substitute the full Stochastic Master Equation, and see that we

will need to calculate

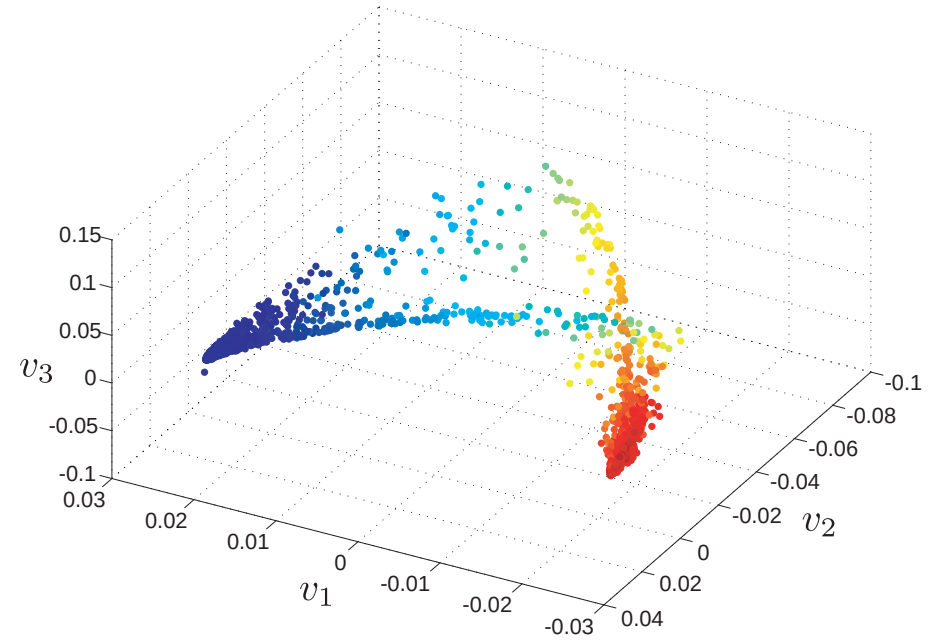
$$\begin{aligned} d\langle\mathcal{O}\rangle &= Tr(-i\mathcal{O}[H,\rho]dt + \kappa\mathcal{O}\left(2a\rho a^\dagger - a^\dagger a\rho - \rho a^\dagger a\right)dt \\ &\quad +\gamma\mathcal{O}\left(2\sigma\rho\sigma^\dagger - \sigma^\dagger\sigma\rho - \rho\sigma^\dagger\sigma\right)dt \\ &\quad +i\sqrt{2\kappa}\mathcal{O}\left(\rho a^\dagger - a\rho - \mathrm{Tr}[\rho\left(a^\dagger - a\right)]\rho\right)dW) \qquad (4.19) \\ &= -i\langle\mathcal{O}H\rangle dt + i\langle H\mathcal{O}\rangle dt + 2\kappa\left\langle a^\dagger\mathcal{O}a\right\rangle dt - \kappa\left\langle\mathcal{O}a^\dagger a\right\rangle dt \\ &\quad -\kappa\left\langle a^\dagger a\mathcal{O}\right\rangle dt + 2\gamma\left\langle\sigma^\dagger\mathcal{O}\sigma\right\rangle dt - \gamma\left\langle\mathcal{O}\sigma^\dagger\sigma\right\rangle dt - \gamma\left\langle\sigma^\dagger\sigma\mathcal{O}\right\rangle dt \\ &\quad +i\sqrt{2\kappa}\langle a\mathcal{O}\rangle dW + \sqrt{2\kappa}\left\langle\mathcal{O}a^\dagger\right\rangle dW - \sqrt{2\kappa}\left\langle a^\dagger - a\right\rangle\langle\mathcal{O}\rangle dW. \qquad (4.20) \end{aligned}$$

This plan depends on finding understandable, finite sets of expectation values from which to determine the manifold coordinates. I found such a set of expectation values for the simpler case of phase bistability, but did not for absorptive bistability.

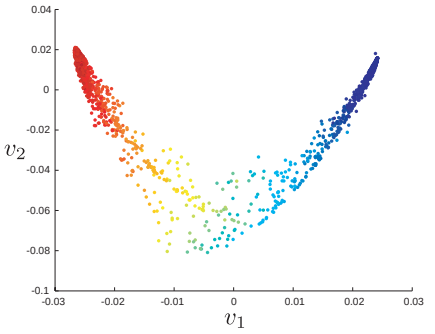
4.3 Phase bistability: Dynamics on three dimensional manifolds from LTSA

The full quantum dynamics of cavity QED, our model open quantum system, take place in an infinite-dimensional space, which I have approximated in this research by a space of density matrices with tens of thousands of elements. Quantum trajectory simulations preserve pure states, so the space in which they take place is much smaller. However, if we keep the first 150 Fock states, as I have in my simulations, the wavefunction still resides in a space with hundreds of dimensions. Thankfully, the dynamics of our canonical phase bistable system use only a small fraction of those degrees of freedom, which we can identify using the LTSA algorithm. Unlike Proper Orthogonal Decomposition, the full algorithm must be run for each potential subspace dimensionality (that is, run separately for $d = 1$, $d = 2$, etc). I ran the LTSA algorithm on a selection of 2000 samples from the same long trajectory analyzed by POD in Section 3.3. $k = 60$ nearest neighbors proved to be in a broad range of k values with similar manifold topology.

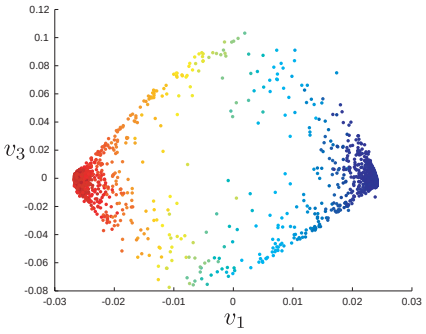
For $d = 1$, the manifold consists of a line segment with samples from one stable point at one end, and the other stable point at the other. It is almost monotonic in $\langle y\rangle$. Expanding to two dimensions ($d = 2$), the manifold takes a roughly parabolic shape. However, at the base of the parabola (in the transition between stable points), the samples align in two bands, which correspond to different $\langle\sigma_y\rangle$. Recall that physically the jumps between phase states correspond to atomic spontaneous emissions events: the atom switches before the field, with the field switching slower because of the cavity. The two bands at the base of the parabola, therefore, correspond to upward-going and downward-going transitions.



(a)



(b)



(c)

Figure 4.1: Local Tangent Space Alignment 3-dimensional manifold from a long quantum trajectory in the phase bistable regime. LTSA algorithm run on 2,000 samples from the trajectory, with $k = 60$ nearest neighbors. Scatter points are colored according to the value of $\langle y \rangle$, the expectation value of the phase quadrature of the cavity field. Part (a) shows a 3D plot; (b) shows just v_2 plotted against v_1; (c) shows v_3 versus v_1. In (c), the upper and lower bands correspond to upward-going and downward-going transitions between the two stable states. Trajectory parameters: $\Theta = 0$, $\Delta = 0$, $\kappa = 4$, $\gamma = 2$, $g_0 = 12$, $\mathcal{E} = 23.57$, $t_{final} = 5000$.

Id	x	y	σ_x	σ_y
σ_z	x^2	$xy + yx$	$x\sigma_x$	$x\sigma_y$
$x\sigma_z$	y^2	$y\sigma_x$	$y\sigma_y$	$y\sigma_z$

Table 4.1: The 15 unique first- and second-order Hermitian operators constructed from the Identity and the five operators from the Maxwell Bloch equations.

The continued structure in the second dimension of the LTSA manifold prompts us to expand to $d = 3$, which produces the manifold show in Figure 4.1. The first two dimensions remain qualitatively similar to their appearance at lower dimensions, while the third dimension shows a split between the two transition bands described above. Extending to $d = 4$, however, does not provide additional insight: this fourth coordinate is apparently randomly scattered with respect to the first three, and shows no structure related to system expectation values. I therefore conclude that the underlying manifold for this regime is very likely three dimensional, and continue with further analysis of only the $d = 3$ manifold.

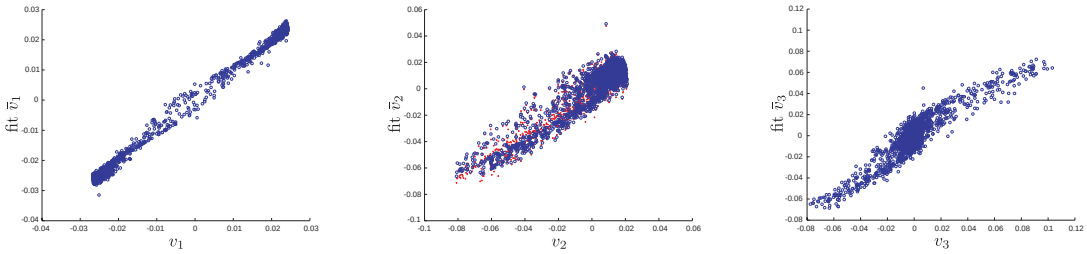
As mentioned in Section 4.2, I attempted to fit the three dimensions which define this manifold with polynomial combinations of operator expectation values. In particular, I built the set of all independent first and second order Hermitian operators which can be created from combinations of the five operators familiar from the Maxwell-Bloch equations: x and y the field quadratures, and σ_x, σ_y, and σ_z the atomic operators. This set of 15 independent Hermitian operators is shown in Table 4.1. If we call these operators in general $\mathcal{O}_i$, then I attempted to fit each of the three global manifold coordinates $v_i, i = 1, 2, 3$ with quadratic combinations

$$v_i = \sum_{j=1}^{15} \sum_{k=j}^{15} x_{jk} \langle \mathcal{O}_j \rangle \langle \mathcal{O}_k \rangle . \tag{4.21}$$

Note that one of the operators is the Identity, so this includes linear terms.

As evident from Figure. 4.1, the leading dimension, v_1, of this LTSA manifold has a very strongly monotonic relationship with the phase quadrature of the cavity field, $\langle y \rangle$. The two stable points reside at either end of the arc, with points from transitions in between. The roughly parabolic shape of the arc in Figure 4.1b leads us to guess that $\langle y^2 \rangle$ or $\langle y \rangle^2$ will play a significant role in approximations of v_2. However, rather than simply guess, I used a "basis pursuit" technique (also known as "l_1-form regularization") to algorithmically identify the leading contributors to each dimension [41]. The CVX convex optimization library for Matlab [42] makes this technique straightforward.

A normal least-squares fit involves minimizing the l_2 norm of the difference between the fit and the data, $\|Ax - v\|_2$, where $A \in \mathbb{R}^{m \times n}$ (the columns of A are the values of the basis functions we're using to fit the data, called the regressors), $v \in \mathbb{R}^m$ are the data, and $x \in \mathbb{R}^n$ are the fit coefficients (the variable we're solving for). *Basis pursuit* minimizes the sum of this l_2 norm and the l_1 norm of the coefficients of the fit. In general, minimizing an l_1 norm puts pressure on the



(a) (b) (c)

Figure 4.2: Scatter plots of 2,000 data points showing the fit between Local Tangent Space Alignment coordinates v_i and fits to those coordinates using only a few expectation values. (a) v_1 fit by $\langle y \rangle$; (b) v_2 fit by $\langle y^2 \rangle$ (blue circles) and by $\langle y^2 \rangle$ and $\langle y\sigma_y \rangle$ (red crosses); (c) v_3 fit by $\langle x\sigma_y \rangle$, $\langle y \rangle \langle y\sigma_y \rangle$, and $\langle x\sigma_y \rangle \langle y^2 \rangle$.

fit to have coefficients which are exactly zero. This allows us to eliminate some coefficients for variables which duplicate the contributions of other variables by selecting the single or few variables which best fit the data. The balance between the l_2 and l_1 norm terms is set by the parameter γ in minimizing $(\|Ax - v\|_2 + \gamma\|x\|_1)$. I weighted the l_1 term by the root mean square of each corresponding expectation value; this controls for the fact that the coefficient of a much larger-valued operator will necessarily be smaller, meaning a naked basis pursuit process will eliminate it even if it is a better fit. I used the basis pursuit process to narrow the number of variables by increasing γ until the number of surviving variables fell to a reasonable number (between one and five) while still allowing a good fit to the data. The presence of the l_1 term can skew the results for the optimal values of the least-squares parameters, so after settling on a small set of variables to include, I did a simple l_2 norm (least-squares) fit with only that parameter set.

Applying the basis pursuit process to the leading dimension of the 3-dimensional phase bistability manifold, v_1, the number of included operators shrinks rapidly as γ is increased. 11 operators survive at $\gamma = 5$, but $\langle y \rangle$ is already clearly dominant. At $\gamma = 14$, only $\langle y \rangle$ remains. Figure 4.2a shows the almost-identity relationship between the values of v_1 and the fit

$$\hat{v}_1 = -0.0011 - 0.0178 \langle y \rangle . \tag{4.22}$$

The second dimension, as predicted, is related to $\langle y^2 \rangle$. At $\gamma = 5$, 14 operators remain, dominated by $\langle y\sigma_y \rangle$. At $\gamma = 20$, there are 6 terms with non-zero coefficients; the leading three are $\langle y^2 \rangle$, $\langle y\sigma_y \rangle$, and $\langle y^2 \rangle \langle y\sigma_y \rangle$. These three all carry roughly the same information ($\langle y \rangle$ and $\langle \sigma_y \rangle$ are largely anti-correlated). At $\gamma = 30$, only $\langle y^2 \rangle$ and $\langle y\sigma_y \rangle$ survive. For $\gamma \geq 38$, only $\langle y^2 \rangle$ survives. The best fit with just $\langle y^2 \rangle$ is shown in the blue circles in Figure 4.2b; the functional form is

$$\hat{v}_2 = -0.0767 + 0.0402 \langle y^2 \rangle . \tag{4.23}$$

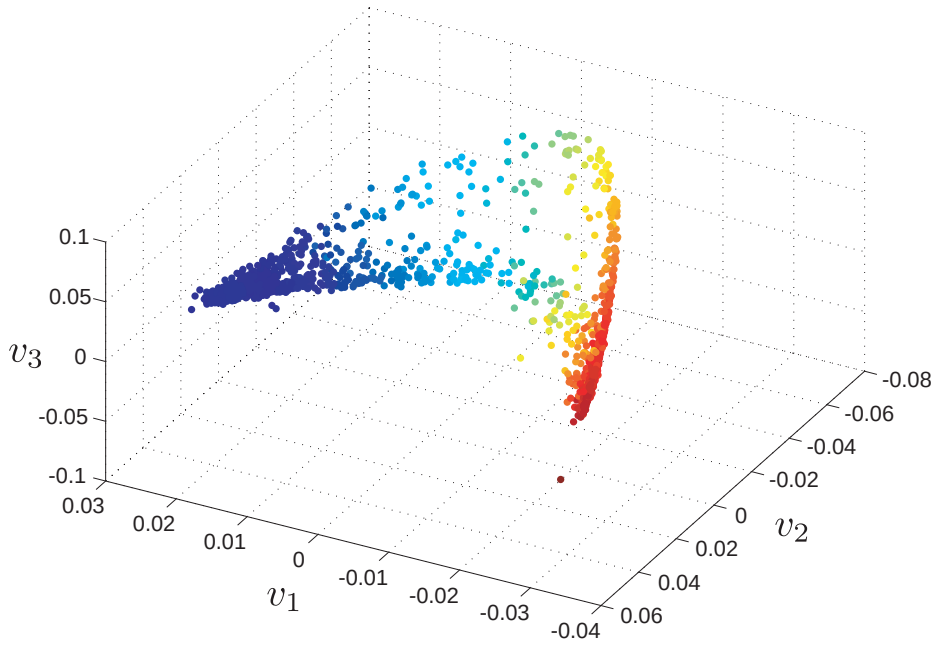


Figure 4.3: Three dimensional fit to the LTSA phase bistability manifold, from Eqns. (4.22), (4.23), and (4.25). Compare with Figure 4.1a to evaluate the fit.

If we allow $\langle y\sigma_y \rangle$ to survive as well (red crosses in Figure 4.2b), the best fit is

$$\tilde{v}_2 = -0.0811 + 0.0284 \langle y^2 \rangle - 0.0244 \langle y\sigma_y \rangle . \tag{4.24}$$

In contrast to the leading two dimensions, the third is not a clear function of any one or two parameters. Basis pursuit narrows the search down to $\langle x\sigma_y \rangle$, $\langle y \rangle \langle y\sigma_y \rangle$ and $\langle x\sigma_y \rangle \langle y^2 \rangle$. The best fit is shown in comparison with the data in Figure 4.2c, and its functional form is

$$\hat{v}_3 = 8.43 \times 10^{-4} - 9.52 \times 10^{-3} \langle x\sigma_y \rangle + 8.26 \times 10^{-2} \langle y \rangle \langle y\sigma_y \rangle - 9.314 \times 10^{-3} \langle x\sigma_y \rangle \langle y^2 \rangle . \tag{4.25}$$

Figure 4.3 shows the 3-dimensional manifold from the fit; compare with Figure 4.1a to see that just a few expectation values carry almost all of the information necessary to reproduce the manifold originally created by analyzing the full density matrices.

4.3.1 A new set of "Maxwell-Bloch" equations

Based on simulations and the LTSA algorithm, I have determined that the first two dimensions of the manifold on which phase bistability dynamics take place can be well approximated with only $\langle y \rangle$ and $\langle y^2 \rangle$. We would like to build analytic expressions for system dynamics on this manifold. In

order to build a complete set of equations for the dynamics of expectation values, we will need to add the three directions on the atomic Bloch sphere as our other observables. We can assume that, because y and y^2 do such a good job, alone, with the leading two dimensions of the phase bistability dynamics, we will be able to make reasonable assumptions to remove other terms and derive a closed set of equations, and in fact we can.

Let us now derive the equations of motion for five expectation values, $\langle y\rangle$, $\langle y^2\rangle$, $\langle\sigma_x\rangle$, $\langle\sigma_y\rangle$, and $\langle\sigma_z\rangle$. This process looks very much like the derivation of the Maxwell-Bloch equations, except that we're going to make a few assumptions (based on the simulation results) along the way to simplify things.

First, let us calculate $\langle y\rangle$, noting that y commutes with the $\mathcal{E}$ term in the Hamiltonian, as well as all σ operators. Also, we have set the detuning parameters to zero, because that is what they are for phase bistable behavior. The commutator of y and the Hamiltonian is

$$\begin{aligned}
[y,H] &= ig_0[y,a^\dagger\sigma_- - a\sigma_+] \\
&= ig_0\left(ya^\dagger\sigma_- - ya\sigma_+ - a^\dagger\sigma_- y + a\sigma_+ y\right) \\
&= ig_0\left(\left(ya^\dagger - a^\dagger y\right)\sigma_- + \left(ay - ya\right)\sigma_+\right) \\
&= -ig_0\left(\frac{i}{2}\sigma_- + \frac{i}{2}\sigma_+\right) \\
&= \frac{-ig_0}{2}\sigma_y. \qquad (4.26)
\end{aligned}$$

Therefore, the equation for the motion of $\langle y\rangle$ is

$$\begin{aligned}
d\langle y\rangle &= -i\langle[y,H]\rangle\,dt + \kappa\left\langle 2a^\dagger ya - a^\dagger ay - ya^\dagger a\right\rangle dt \\
&\quad +\sqrt{2\kappa}\left\langle a^\dagger y - ya - \left\langle a^\dagger - a\right\rangle y\right\rangle dW \\
&= \frac{-g_0}{2}\langle\sigma_y\rangle\,dt - \kappa\langle y\rangle\,dt + \sqrt{8\kappa}\left(\left\langle y^2\right\rangle - \langle y\rangle^2\right)dW. \qquad (4.27)
\end{aligned}$$

Note that we could have derived this (aside from the dW term) from the $\langle\dot{a}\rangle$ equation in the standard Maxwell Bloch equations. The dW always takes a standard form related to the difference between $\langle\mathcal{O}y\rangle$ and $\langle\mathcal{O}\rangle\langle y\rangle$, so we can avoid calculating the equations of motion for $\langle\sigma_{x,y,z}\rangle$, and just state them here:

$$d\langle\sigma_x\rangle = 2g_0\langle x\sigma_z\rangle\,dt - \gamma\langle\sigma_x\rangle\,dt + \sqrt{8\kappa}\left(\langle y\sigma_x\rangle - \langle y\rangle\langle\sigma_x\rangle\right)dW \qquad (4.28)$$

$$d\langle\sigma_y\rangle = -2g_0\langle y\sigma_z\rangle\,dt - \gamma\langle\sigma_y\rangle\,dt + \sqrt{8\kappa}\left(\langle y\sigma_y\rangle - \langle y\rangle\langle\sigma_y\rangle\right)dW \qquad (4.29)$$

$$\begin{aligned}
d\langle\sigma_z\rangle &= -g_0\left(\langle x\sigma_x\rangle - \langle y\sigma_y\rangle\right)dt - 2\gamma\left(\langle\sigma_z\rangle + 1\right)dt \\
&\quad +\sqrt{8\kappa}\left(\langle y\sigma_z\rangle - \langle y\rangle\langle\sigma_z\rangle\right)dW. \qquad (4.30)
\end{aligned}$$

The $\langle y^2 \rangle$ equation is new, and cannot be simply written down based on the Maxwell Bloch equations. y^2 commutes with the same terms in the Hamiltonian that y does, simplifying that commutation calculation.

$$\begin{aligned}
[y, H] &= ig_0[y^2, a^\dagger\sigma_- - a\sigma_+] \\
&= ig_0\left(y^2a^\dagger\sigma_- - y^2a\sigma_+ - a^\dagger\sigma_-y^2 + a\sigma_+y^2\right) \\
&= ig_0\left(\left(y^2a^\dagger - a^\dagger y^2\right)\sigma_- + \left(ay^2 - y^2a\right)\sigma_+\right) \\
&= -g_0\left(y\sigma_+ - y\sigma_-\right) \\
&= -ig_0y\sigma_y
\end{aligned} \tag{4.31}$$

where we have used the commutation relations between y^2 and a and $a^\dagger$:

$$\left[y^2, a\right] = yya - ayy = y(ay - \frac{i}{2}) - (ya + \frac{i}{2})y = -iy \tag{4.32}$$

$$\left[y^2, a^\dagger\right] = yya^\dagger - a^\dagger yy = y(a^\dagger y - \frac{i}{2}) - (ya^\dagger + \frac{i}{2})y = -iy. \tag{4.33}$$

Looking now to the full differential equation, and plugging in the result for the commutator, we have

$$\begin{aligned}
d\left\langle y^2\right\rangle &= -i\left\langle [y, H]\right\rangle dt + \kappa\left\langle 2a^\dagger y^2a - a^\dagger ay^2 - y^2a^\dagger a\right\rangle dt \\
&\quad +\sqrt{2\kappa}\left\langle a^\dagger y^2 - y^2a - \left\langle a^\dagger - a\right\rangle y^2\right\rangle dW \\
&= -g_0\left\langle y\sigma_y\right\rangle dt + \kappa\left\langle a^\dagger\left(ay^2 - iy\right) - a^\dagger ay^2 + \left(y^2a^\dagger + iy\right)a - y^2a^\dagger a\right\rangle dt \\
&\quad +\sqrt{8\kappa}\left\langle y^3 - \left\langle y\right\rangle y^2\right\rangle dW \\
&= -g_0\left\langle y\sigma_y\right\rangle dt - i\kappa\left\langle a^\dagger y - ya\right\rangle dt + \sqrt{8\kappa}\left(\left\langle y^3\right\rangle - \left\langle y\right\rangle\left\langle y^2\right\rangle\right)dW \\
&= -g_0\left\langle y\sigma_y\right\rangle dt - i\kappa\left\langle a^\dagger y - (ay - i/2)\right\rangle dt + \sqrt{8\kappa}\left(\left\langle y^3\right\rangle - \left\langle y\right\rangle\left\langle y^2\right\rangle\right)dW \\
&= -g_0\left\langle y\sigma_y\right\rangle dt - 2\kappa\left\langle y^2\right\rangle dt + \kappa/2dt + \sqrt{8\kappa}\left(\left\langle y^3\right\rangle - \left\langle y\right\rangle\left\langle y^2\right\rangle\right)dW.
\end{aligned} \tag{4.34}$$

Collecting our five equations together, we now have:

$$d\left\langle y\right\rangle = \frac{-g_0}{2}\left\langle\sigma_y\right\rangle dt - \kappa\left\langle y\right\rangle dt + \sqrt{8\kappa}\left(\left\langle y^2\right\rangle - \left\langle y\right\rangle^2\right)dW \tag{4.35}$$

$$d\left\langle y^2\right\rangle = -g_0\left\langle y\sigma_y\right\rangle dt - 2\kappa\left\langle y^2\right\rangle dt + \kappa/2dt + \sqrt{8\kappa}\left(\left\langle y^3\right\rangle - \left\langle y\right\rangle\left\langle y^2\right\rangle\right)dW \tag{4.36}$$

$$d\left\langle\sigma_x\right\rangle = 2g_0\left\langle x\sigma_z\right\rangle dt - \gamma\left\langle\sigma_x\right\rangle dt + \sqrt{8\kappa}\left(\left\langle y\sigma_x\right\rangle - \left\langle y\right\rangle\left\langle\sigma_x\right\rangle\right)dW \tag{4.37}$$

$$d\left\langle\sigma_y\right\rangle = -2g_0\left\langle y\sigma_z\right\rangle dt - \gamma\left\langle\sigma_y\right\rangle dt + \sqrt{8\kappa}\left(\left\langle y\sigma_y\right\rangle - \left\langle y\right\rangle\left\langle\sigma_y\right\rangle\right)dW \tag{4.38}$$

$$\begin{aligned}
d\left\langle\sigma_z\right\rangle &= -g_0\left(\left\langle x\sigma_x\right\rangle - \left\langle y\sigma_y\right\rangle\right)dt - 2\gamma\left(\left\langle\sigma_z\right\rangle + 1\right)dt \\
&\quad +\sqrt{8\kappa}\left(\left\langle y\sigma_z\right\rangle - \left\langle y\right\rangle\left\langle\sigma_z\right\rangle\right)dW.
\end{aligned} \tag{4.39}$$

Note that this set of equations is not closed: there are many terms on the right hand side which have no corresponding dynamical equation. This is where approximation and calculations from the simulations come in. I fit the expectation values of these operators using basis pursuit followed by a least-squares fit:

$$\begin{aligned}
\langle y\sigma_x\rangle &\approx A_0 + A_1\langle y\rangle\langle\sigma_x\rangle \\
\langle y\sigma_y\rangle &\approx B_0 + B_1\langle y\rangle\langle\sigma_y\rangle \\
\langle y\sigma_z\rangle &\approx C_0 + C_1\langle y\rangle\langle\sigma_z\rangle \\
\langle x\sigma_x\rangle &\approx D_0 + D_1\langle x\rangle\langle\sigma_x\rangle = D_0 + D_1(E_0 + E_1\langle y^2\rangle)\langle\sigma_x\rangle \\
&\quad H_0 + H_1\langle\sigma_x\rangle + H_2\langle y^2\rangle\langle\sigma_x\rangle \\
\langle x\sigma_z\rangle &\approx F_0 + F_1\langle x\rangle\langle\sigma_z\rangle = F_0 + F_1(E_0 + E_1\langle y^2\rangle)\langle\sigma_z\rangle \\
&= J_0 + J_1\langle\sigma_z\rangle + J_2\langle y^2\rangle\langle\sigma_z\rangle \\
\langle y^3\rangle &\approx G_0 + G_1\langle y\rangle + G_2\langle y\rangle\langle y^2\rangle . \qquad (4.40)
\end{aligned}$$

All the fits which support these approximations are quite good (although there is significant scatter on some, there is no clear break from a linear response). The best fit values for these constants are:

$$\begin{array}{lll}
A_0 = 0.0059 & B_0 = -0.4392 & C_0 = 0.0003 \\
A_1 = 1.0060 & B_1 = 0.6706 & C_1 = 0.8751 \\
H_0 = 0.0217 & J_0 = -0.0468 & G_0 = 0.0071 \\
H_1 = 5.8911 & J_1 = 5.8120 & G_1 = 0.8526 \\
H_2 = -0.1602 & J_2 = -0.1213 & G_2 = 0.8487
\end{array}$$

Renaming variables to simplify the display and emphasize that we're now dealing with a simple set of classical stochastic differential equations, we now have

$$\begin{aligned}
dv_1 &= \frac{-g_0}{2}v_4dt - \kappa v_1dt + \sqrt{8\kappa}\left(v_2 - v_1^2\right)dW \\
dv_2 &= -g_0\left(B_0 + B_1v_1v_4\right)dt - 2\kappa v_2dt + \kappa/2dt \\
&\quad +\sqrt{8\kappa}\left(G_0 + G_1v_1 + (G_2-1)v_1v_2\right)dW \\
dv_3 &= 2g_0\left(J_0 + J_1v_5 + J_2v_2v_5\right)dt - \gamma v_3dt \\
&\quad +\sqrt{8\kappa}\left(A_0 + (A_1-1)v_1v_3\right)dW \\
dv_4 &= -2g_0\left(C_0 + C_1v_1v_5\right)dt - \gamma v_4dt + \sqrt{8\kappa}\left(B_0 + (B_1-1)v_1v_4\right)dW \\
dv_5 &= g_0\left((B_0 + B_1v_1v_4) - (H_0 + H_1v_3 + H_2v_2v_3)\right)dt - 2\gamma\left(v_5+1\right)dt \\
&\quad +\sqrt{8\kappa}\left(C_0 + (C_1-1)\,v_1v_5\right)dW. \qquad (4.41)
\end{aligned}$$

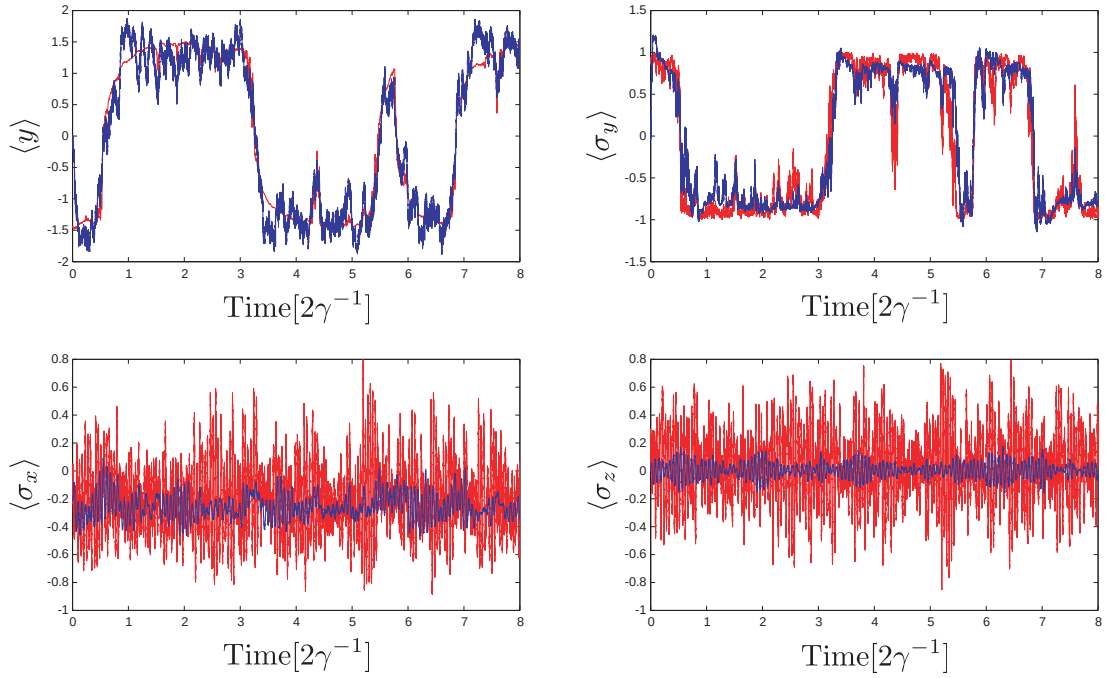


Figure 4.4: Excerpt traces for the expectation values of $\langle y \rangle$, $\langle \sigma_y \rangle$, $\langle \sigma_x \rangle$, and $\langle \sigma_z \rangle$ for a quantum trajectory simulation (red) and the filter in Eqn. (4.41) (blue). The output of the filter in Eqn. (4.42) would be virtually indistinguishable for $\langle y \rangle$ and $\langle \sigma_y \rangle$; that filter has no role for $\langle \sigma_x \rangle$ and $\langle \sigma_z \rangle$.

Simulations of these equations with white noise input for dW show that v_1 fluctuates in the range of -2 to 2, with possible stable points at near ± 1.5 (close to where we would expect the stable points of the quantum system to be). The distribution of v_1 is bimodal, with peaks at ± 1.5.

These equations may be used as a filter as well. Unfortunately, when run as a filter, with dW equal to the innovation from a quantum trajectory's homodyne photocurrent, these equations do not tend to be very stable. Sections of the photocurrent appear able to drive the filter into unphysical states more quickly than they can recover. The traces shown in Figure 4.4 were selected to lie between these points of instability. A practical implementation of this filter would need to be on the lookout for these points, and reset the filter quickly. The lack of a full density matrix means that we cannot evaluate this filter on the basis of the fidelity, like we could in the case of Proper Orthogonal Decomposition. However, we may see how it compares with POD in predicting expectation values, comparing Figure 4.4 with Figure 3.3 and Figure 3.4.

Van Handel and Mabuchi [11] use a projection filter to derive a simple set of three SDEs which also act as a very good filter for the phase bistable regime. They, however, make a fundamentally different choice (driven by their desire for an analytic solution): the manifold onto which they project requires that $\langle \sigma_x \rangle$ and $\langle \sigma_z \rangle$ are both exactly 0 (they fix the atom in a state polarized along σ_y). They are able to do a good job of reproducing the behavior of the optimal filter for $\langle y \rangle$ and $\langle \sigma_y \rangle$, but are fundamentally unable to handle any situation in which the atom has components in another

direction. It is true that in simulations of the filter equations I have derived above, the values of $\langle\sigma_x\rangle$ and $\langle\sigma_z\rangle$ appear to be uncorrelated with the switching behavior. The average of $\langle\sigma_z\rangle$ is close to zero for any given realization, while the mean of $\langle\sigma_x\rangle$ is small but nonzero.

Following the example of [11], we decouple our equations from $\langle\sigma_x\rangle$ and $\langle\sigma_z\rangle$, by setting $C_0 = 0$ and $C_1 = 0$, and are left with:

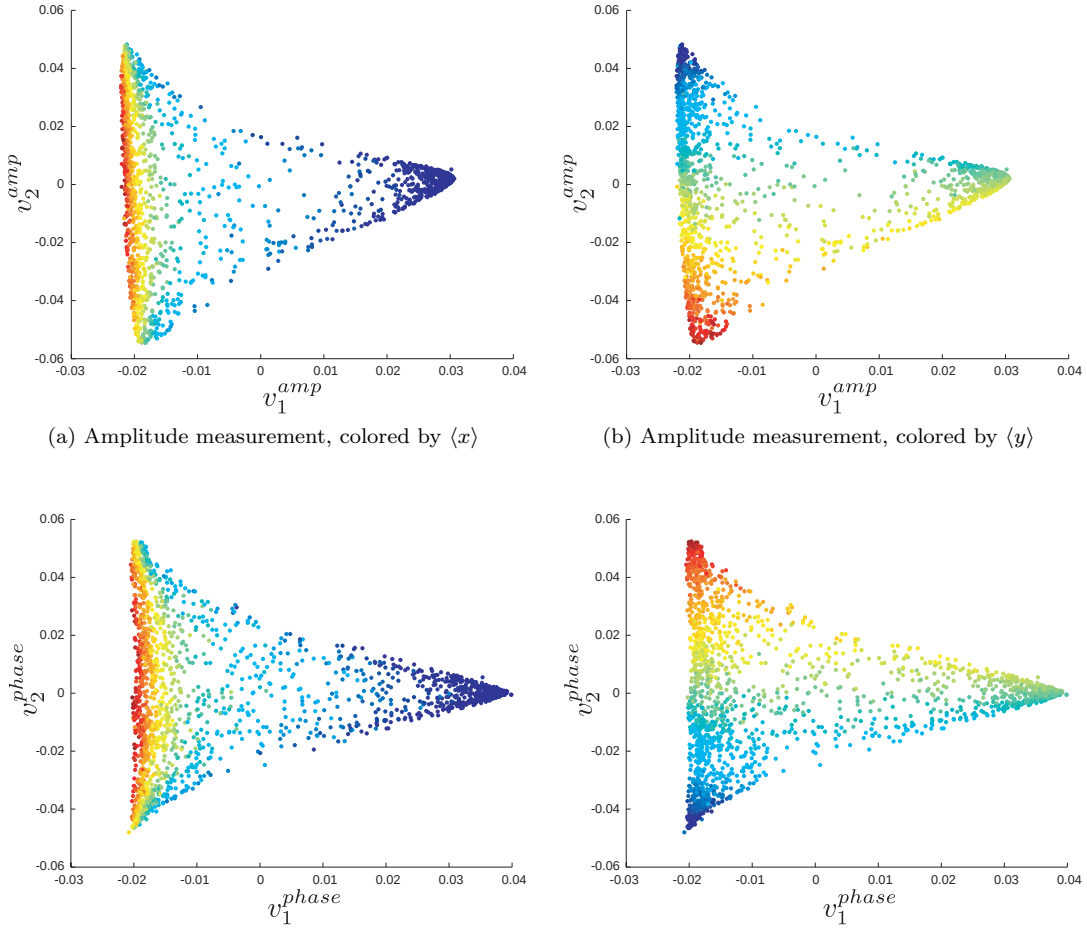
$$\begin{aligned} dv_1 &= \frac{-g_0}{2} v_4 dt - \kappa v_1 dt + \sqrt{8\kappa}\left(v_2 - v_1^2\right) dW \\ dv_2 &= -g_0\left(B_0 + B_1 v_1 v_4\right) dt - 2\kappa v_2 dt + \kappa/2 dt \\ &\quad + \sqrt{8\kappa}\left(G_0 + G_1 v_1 + (G_2 - 1) v_1 v_2\right) dW \\ dv_4 &= -\gamma v_4 dt + \sqrt{8\kappa}\left(B_0 + (B_1 - 1) v_1 v_4\right) dW. \end{aligned} \tag{4.42}$$

(Strictly speaking we are approximating that $\langle y\sigma_z\rangle = 0$ rather than $\langle\sigma_z\rangle = 0$, but this is a consistent approximation.) When it comes to filtering, the output of this filter is almost indistinguishable from the output of the 5-dimensional filter above (the three expectation values differ on average by less than 3%). Decoupling from the x and z components of the atom is clearly a very good approximation. It also seems to make numerical simulation mildly more stable, but does not solve the filter instability problem. This three-dimensional filter is also consistent with the experience of the LTSA manifold generation process: three dimensions appears to be sufficient. It is unfortunate that we were forced to develop a filter only in terms of system observables, which prevents us from evaluating the full quantum fidelity of the approximation. However, the filter appears to perform very well with the observables a experiment is likely to measure and care about, and we cannot deny the value of an easy to implement, exact functional form. These "Maxwell-Bloch" equations are a product of this particular trajectory, and these particular parameters. This process would need to be repeated to generate any other filter for experimental parameters.

4.4 Two and three dimensional manifolds for absorptive bistability

Absorptive bistability in cavity QED is arguably a more complicated process than phase bistability. The two stable states are not symmetric, and in fact the upper state is much broader than the vacuum, with significant variation within the state (or zone). The transitions do not have a obvious model or explanation (like the phase bistable case does with atomic spontaneous emission). It is therefore somewhat of a surprise that the Local Tangent Space Alignment-generated manifold for our canonical absorptive bistability conditions is quite simple.

I generated $d = 2$, $d = 3$, $d = 5$, and $d = 8$ manifolds with $k = 40$ nearest neighbors from



(a) Amplitude measurement, colored by $\langle x \rangle$

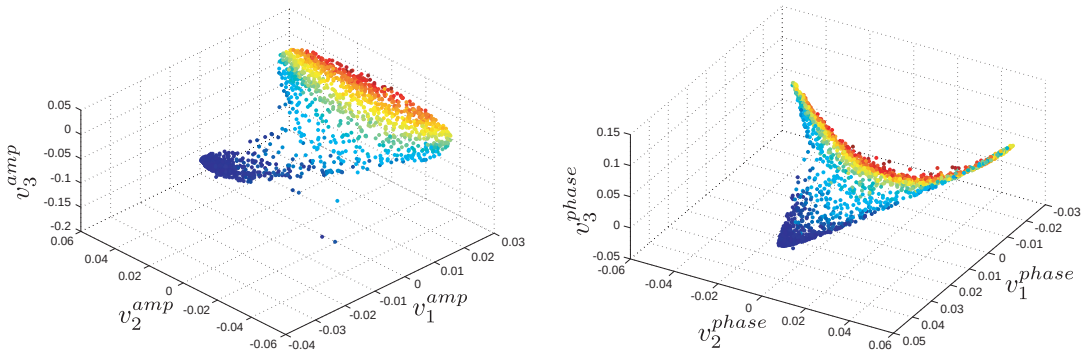
(b) Amplitude measurement, colored by $\langle y \rangle$

(c) Phase measurement, colored by $\langle x \rangle$

(d) Phase measurement, colored by $\langle y \rangle$

Figure 4.5: Two-dimensional Local Tangent Space Alignment-produced manifolds for absorptive bistability. (a) and (b) are from a trajectory in which the amplitude quadrature is measured by homodyne detection; (c) and (d) are from a trajectory in which the phase quadrature was measured. System parameters are our canonical set; LTSA parameters are $d = 2$ and $k = 40$, on a set of 2000 samples.

the same trajectories I used for Proper Orthogonal Decomposition, for both amplitude and phase quadrature homodyne measurement of the cavity field. The $d = 2$ manifold (Figure 4.5) shows a clear triangular-like structure, with the lower state at one point of the triangle, and the extended upper state stretching along the opposing edge. This edge corresponds roughly to variation in $\langle y \rangle$ within the upper state, and the points of highest amplitude quadrature reside in the middle of that edge. These two dimensions can be mapped one-to-one onto the phase and amplitude quadratures of the quantum trajectory points used to generate the manifold. The first dimension corresponds roughly to the amplitude quadrature variation (or perhaps more accurately the distance from the vacuum state), and the second to the phase. The eigenvalues of B for this case show that the first dimension is dominant relative to the second: the corresponding eigenvalue is roughly a factor of 7.5 smaller for amplitude measurement, and half the size for phase measurement (recall that in LTSA



(a) amplitude measurement, colored by $\langle x \rangle$

(b) phase measurement, colored by $\langle x \rangle$

Figure 4.6: Three-dimensional Local Tangent Space Alignment-produced manifolds for absorptive bistability.

what matters is the small eigenvalue states, not the large eigenvalues as in POD). The simple atomic expectation values (σ_x, σ_y, and σ_z) do not have an obvious relation to these two dimensions.

Extending to $d = 3$, shown in Figure 4.6, we see that structure continues in this dimension, although the manifold now looks very much like a two-dimensional surface embedded in three dimensions. This may correspond to the continued growing eigenvalues from the second to the third; the eigenvalue for the third dimension is roughly 50% larger than that for the second in the fuzzier amplitude-measurement case, and over twice as large for the more crisply shaped phase quadrature manifold. The shape of the $d = 2$ manifold is very similar regardless of whether we measure the phase or amplitude quadrature (see Figure 4.5). The shape in the third dimension, however, is quite different: in the phase measurement case, the points with largest $\langle y \rangle$ curve away in the third dimension; in the amplitude measurement case, the whole upper state curves together. Perhaps this dimension corresponds to the absolute value of the measured quadrature (or possibly the square of the observable given the somewhat parabolic shape of the curve).

If we run the same process with $d = 5$, we encounter some mild structure within these added dimensions. However, the third (as before) and fourth dimensions do not show any clear relation with system observables. In the case of amplitude quadrature measurement, the fifth dimension shows a similar slight amount of structure and is monotonic with σ_y; a similar thing happens for the sixth dimension for phase quadrature measurement. This leads me to think that the intermediate dimensions cannot be entirely overlooked, although their relation to system observables and dynamics remains somewhat of a mystery. For amplitude quadrature measurement, the eigenvalue for the fifth dimension is about 25% larger than those for the third and fourth dimensions (which are quite close together and each about 12 times larger than that for the leading dimension).

With $d = 8$, we learn that the fifth dimension was the last to show any useful structure for amplitude measurement, and the sixth for phase measurement. In both cases, the eigenvalues for the 6th, 7th, and 8th dimensions rise significantly relative to the lower dimensions, and the scatter

of points for each of the higher dimensions show that they pull out a handful of points to large displacement from the rest of the data: we have reached the threshold where structureless scatter in the data dominates.

While the values of the eigenvalues of B change as we raise d, the approximate ratios of eigenvalues appear to be quite robust to changes in d. They are slightly more dependent on k. Changing k from 40 to 60 in the amplitude measurement case, for example, changes the ratio of the leading pair of eigenvalues only from 7.5 to 8.6, and the third and fourth eigenvalues are still very close.

The manifold for absorptive bistability does not show the same clear dimension that the phase bistable case demonstrates. The underlying dynamics may require five or six dimensions (which is still a dramatic reduction compared with the size of the density matrix space), or it is even possible that different parts of the dynamics (such as the upper and lower "states") require different numbers of dimensions. Regardless, in this regime the leading two dimensions seem to dominate, and remain reassuringly independent of the measurement used, implying that they reflect behavior inherent to the physical system.

As I mentioned in the Introduction, the driving field we have used throughout this thesis when examining absorptive bistability is actually above the bifurcation point for the related semi-classical system (the Maxwell-Bloch equations). The structure of this transition is still unknown in the quantum case, so I ran the LTSA algorithm on trajectories with driving fields ranging from $\mathcal{E} = 0.4$ to $\mathcal{E} = 0.65$. Over this range, the Q functions for the steady state of the quantum system transition from entirely single-peaked around the lower state, to entirely peaked around the upper state. The $d = 3$ LTSA manifold at $\mathcal{E} = 0.4$ consists of a cluster at the lower state, with its own shape and relation to system observables; as the driving field is increased, a second cluster appears and becomes more populated, eventually outgrowing the lower state near $\mathcal{E} = 11.2$, our canonical case. This process continues as the driving field $\mathcal{E}$ increases, and at $\mathcal{E} = 0.65$, the upper state is left alone, with the LTSA manifold displaying the leading dimensions of its internal structure. Unfortunately, this process shows no fundamental or sharp change in the manifold topology corresponding to the bifurcation, and leaves us without significant additional insight into the bifurcation process.

4.4.1 Fitting to observables

As discussed above, the leading two dimensions from the LTSA algorithm applied to absorptive bistability are roughly monotonic in $\langle x \rangle$ and $\langle y \rangle$, which gives us hope that we will be able apply the same basis pursuit technique developed for phase bistability to these dimensions. The mystery of the third and fourth dimensions might be resolved by the search through variable space which basis pursuit automates, and the fifth (or sixth) dimension is likely related to σ_y.

Unfortunately, the possible basis set of first-, second- order polynomials of the first-, second-, and third-order combinations of our basic set of Hermitian system operators (x, y, σ_x, σ_y, and σ_z)

only do a good job of fitting the leading dimension when a large, unwieldy, number of the basis functions are included. The fit is generally better in the two stable regions, but becomes poor in the transition region as we try to restrict the number of number of basis states to fewer than 10 or so. Most likely, future work would require a larger basis extending beyond simple polynomials (and might require more computing power and/or more powerful fitting software), or external insight into the best functional form. While polynomials do span the space of possible functions, we may have lost considerably by considering only Hermitian polynomial terms independently.

The second dimension is well fit by

$$\hat{v}_2 = -0.0016 - 0.0178\,\langle y\rangle - 0.004\,\langle \sigma_z\rangle + 0.0014\,\langle y\rangle\,\left\langle y^2\right\rangle \tag{4.43}$$

for our example amplitude quadrature measurement data set with 2000 samples, $d = 5$, and $k = 40$. The leading contributors to the fit for the fifth dimension, as expected, relate to σ_y: $\langle\sigma_y\rangle$, $\langle\sigma_y\rangle\,\langle x\rangle$, $\langle\sigma_y\rangle\,\left\langle y^2\right\rangle$, and $\langle\sigma_y\rangle\,\langle\sigma_z\rangle$. However, this set does only a marginal job at fitting this dimension alone, and several additional operators are necessary to get a high quality fit. Even with over a dozen terms contributing to each, the fit to the third and fourth dimensions are quite poor. Without a reasonable functional form for the leading dimension, we are unable to construct the equivalent of the new "Maxwell-Bloch" equations we constructed for phase bistability in Section 4.3.1. Regardless of this (perhaps temporary) obstacle, it is clear that Local Tangent Space Alignment provides valuable insight into the fundamental behavior of our example quantum systems.