SC Corr-Code Package:
Theory and Reference Manual

Matthew S. Church
Ananth Group
Cornell University
msc336@cornell.edu

Version 1.1

October 29, 2018

This is the reference manual for the SC Corr-Code Package (FORTRAN) developed by M. S. Church, S. V. Antipov and N. Ananth. It is available on the Ananth group GitHub page and the Ananth Group website. The package contains a variety of methods that compute the real-time correlation function within the framework of the semiclassical initial value representation. Here we provide some theoretical background and useful references for additional reading, outline the structure of the program, and describe how the program can be used and altered.

Contents

1 Introduction **3**
1.1 Useful References 3

2 Semiclassical Propagators **3**
2.1 Van Vleck-Gutzwiller Propagator 3
2.2 Position-space (Van Vleck) SC-IVR 4
2.3 Coherent State (Herman-Kluk) SC-IVR 4

3 SC-IVR Correlation Functions **5**
3.1 Linearized SC-IVR 6
3.2 Mixed Quantum-Classical IVR 6
3.2.1 Forward-Backward MQC-IVR 7
3.2.2 Double-Forward MQC-IVR 8

4 Analytical Quantum-Classical IVR **9**

5 Semiclassical Nonadiabatic Dynamics **10**

5.1 The MInt Algorithm 11

6 SC-IVR Code Package **11**

6.1 Program Overview 11

6.2 Running the Program 12

6.3 The Makefile, External Libraries, and the MPI Submission Script 12

6.4 Tutorial 1: 1D Anharmonic Oscillator 13

6.5 Tutorial 2: 2D Anharmonic Oscillator 15

6.6 Altering the Code 16

6.6.1 Tutorial 3: Other Potentials 17

6.6.2 Tutorial 4: Other Observables 18

6.7 Tutorial 5: Nonadiabatic Dynamics with MQC-IVR 19

6.8 The `MonteCarlo` files 19

6.9 The `supply` files 20

6.10 The `traj` files 20

6.11 Other comments 20

7 Acknowledgements **20**

1 Introduction

Methods based on the semiclassical initial value representation (SC-IVR) are a promising approach to simulating complex chemical systems in real-time. They are rigorously derived, describe quantum effects accurately, and employ near-classical trajectories. The aim of this program is to provide several means of computing the real-time correlation function using the SC-IVR formalism. We provide simple 1D and 2D model systems with which to test and become familiar with the program, although the code is general, and extensions to arbitrary multidimensional systems can be easily implemented, as described below. Section II of this manual provides some brief theoretical background of the SC-IVR formalism, and Section III provides a description of the program structure as well as some useful tutorials.

1.1 Useful References

Textbooks:
1. D. Tannor "Introduction to Quantum Mechanics: A Time Dependent Perspective" Univ. Sci. Books, Chapter 10 (2007)
2. M.S. Child "Semiclassical Mechanics with Molecular Applications" Oxford University Press 2^{nd} Ed.(2014)
3. M.C. Gutzwiller "Chaos in Classical and Quantum Mechanics" Springer-Verlag (1990)
Journal Articles
4. W.H. Miller, *J. Phys. Chem. A.* **105**, 2942 (2001)
6. S.V. Antipov, Z. Ye, and N. Ananth, *J. Chem. Phys.* **142**, 184102 (2015)
7. M.S. Church, S.V. Antipov, and N. Ananth, *J. Chem. Phys.* **146**, 234104 (2017)
8. M.S. Church, T.J.H. Hele, G.S. Ezra, and N. Ananth. *J. Chem. Phys.* **148**, 102326 (2018)

2 Semiclassical Propagators

2.1 Van Vleck-Gutzwiller Propagator

We begin with a brief discussion of semiclassical propagators. Using the Feynman path integral formulation of quantum mechanics, the amplitude associated with a transition from position state $|\mathbf{q}_0\rangle$ to $|\mathbf{q}_t\rangle$ in time t is given by

$$\langle \mathbf{q}_t | e^{-\frac{i}{\hbar}\hat{H}t} | \mathbf{q}_0 \rangle \;=\; \int \mathcal{D}[\mathbf{q}(t)]\; e^{\frac{i}{\hbar}S[\mathbf{q}(t)]}. \tag{1}$$

The integral in Eq. (1) is over the continuum of paths, $\mathbf{q}(t)$, that begin at $\mathbf{q}_0$ and end at $\mathbf{q}_t$ in time t. $S[\mathbf{q}(t)]$ is the action integral of the path, $\mathcal{D}[\mathbf{q}(t)]$ is the path differential, and $\hat{H}$ is the Hamiltonian. The oscillatory phase on the right hand side of Eq. (1) suggests that stationary paths (i.e. the classical paths, according to Hamilton's principle) contribute the most to the transition amplitude, whereas non-classical paths exhibit significant phase cancellation leading to lower contributions. Expanding $S[\mathbf{q}(t)]$ around the classical paths to second order and evaluating the integral analytically leads to the Van Vleck-Gutzwiller propagator (VVG),

$$\langle \mathbf{q}_t | e^{-\frac{i}{\hbar}\hat{H}t} | \mathbf{q}_0 \rangle \;=\; \sum_{\text{classical paths}} \sqrt{\frac{1}{(2\pi i\hbar)^N} \det \left| \frac{\partial \mathbf{q}_t}{\partial \mathbf{p}_0} \right|^{-1}}\; e^{\frac{i}{\hbar}S[\mathbf{q}(t)]}, \tag{2}$$

where $\mathbf{p}_0$ is the initial momentum of the trajectory starting at $\mathbf{q}_0$, and N is the full system dimensionality. Since Eq. (2) is obtained as the stationary phase approximation to the exact propagator, it is an exact replacement for Eq. (1) in the limit that $\hbar \to 0$. Although this propagator is capable of describing virtually all quantum effects, there are practical challenges to consider when computing Eq. (2). Finding all trajectories that satisfy the double-ended boundary condition requires a computationally inefficient search, and the prefactor may diverge when the determinant of the $N \times N$ matrix $\frac{\partial \mathbf{q}_t}{\partial \mathbf{p}_0}$ approaches zero, among others.

2.2 Position-space (Van Vleck) SC-IVR

The trajectory search problem that arises from computing Eq. (2) can be removed by formulating the propagator in terms of an integral over initial phase space conditions. Using completeness to express the time evolution operator,

$$e^{-\frac{i}{\hbar}\hat{H}t} = \int \mathrm{d}\mathbf{q}_t \int \mathrm{d}\mathbf{q}_0 \, |\mathbf{q}_t\rangle \langle \mathbf{q}_t | e^{-\frac{i}{\hbar}\hat{H}t} | \mathbf{q}_0 \rangle \langle \mathbf{q}_0 | \,,$$

and then substituting the expression in Eq. (2) for the transition amplitude we obtain,

$$e^{-\frac{i}{\hbar}\hat{H}t} \approx \frac{1}{(2\pi i\hbar)^{\frac{N}{2}}} \sum_{\text{classical paths}} \int \mathrm{d}\mathbf{q}_t \int \mathrm{d}\mathbf{q}_0 \, \det \left| \frac{\partial \mathbf{q}_t}{\partial \mathbf{p}_0} \right|^{-\frac{1}{2}} e^{\frac{i}{\hbar} S_t(\mathbf{p}_0, \mathbf{q}_0)} \, |\mathbf{q}_t\rangle \langle \mathbf{q}_0 | \,. \tag{3}$$

Now make a simple change of variables to replace the integral over $\mathrm{d}\mathbf{q}_t$ in Eq. 3 with an integral over $\mathrm{d}\mathbf{p}_0$,

$$\sum_{\text{classical paths}} \int \mathrm{d}\mathbf{q}_t = \int \mathrm{d}\mathbf{p}_0 \left| \frac{\partial \mathbf{q}_t}{\partial \mathbf{p}_0} \right|,$$

to recover the Van Vleck-IVR (VV-IVR) expression for the propagator,

$$e^{-\frac{i}{\hbar}\hat{H}t} \approx \frac{1}{(2\pi i\hbar)^{\frac{N}{2}}} \int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \, \det \left| \frac{\partial \mathbf{q}_t}{\partial \mathbf{p}_0} \right|^{\frac{1}{2}} e^{\frac{i}{\hbar} S_t(\mathbf{p}_0, \mathbf{q}_0)} \, |\mathbf{q}_t\rangle \langle \mathbf{q}_0 | \,. \tag{4}$$

Though the conversion removes the double-ended boundary condition and moves the prefactor to the numerator, evaluating Eq. (4) remains challenging for multidimensional systems due to the oscillatory phase and the lack of initial momentum information with which to sample.

Note that VV-IVR can also be written in terms of momentum states as

$$e^{-\frac{i}{\hbar}\hat{H}t} \approx \frac{1}{(2\pi i\hbar)^{\frac{N}{2}}} \int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \, \det \left| \frac{\partial \mathbf{p}_t}{\partial \mathbf{q}_0} \right|^{\frac{1}{2}} e^{\frac{i}{\hbar} S_t(\mathbf{p}_0, \mathbf{q}_0)} \, |\mathbf{p}_t\rangle \langle \mathbf{p}_0 | \,. \tag{5}$$

Due to the existence of more efficient methods, the propagators of Eq. (4) and Eq. (5) are not included in this package. We discuss them here for pedagogical reasons, and because they can be used to derive the linearized SC-IVR described below.

2.3 Coherent State (Herman-Kluk) SC-IVR

An appealing alternative to the previously discussed propagators is the Herman-Kluk IVR (HK-IVR), which is represented in the basis of coherent states as,

$$e^{-\frac{i}{\hbar}\hat{H}t} \approx \frac{1}{(2\pi\hbar)^N} \int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \, C_t(\mathbf{p}_0, \mathbf{q}_0) \, e^{\frac{i}{\hbar} S_t(\mathbf{p}_0, \mathbf{q}_0)} \, |\mathbf{p}_t \mathbf{q}_t\rangle \langle \mathbf{p}_0 \mathbf{q}_0 | \,. \tag{6}$$

The prefactor here is given by,

$$C_t(\mathbf{p}_0, \mathbf{q}_0) \;=\; \det\left|\gamma_t^{\frac{1}{2}}\mathbf{M}_{qq}\gamma_0^{-\frac{1}{2}} + \gamma_t^{-\frac{1}{2}}\mathbf{M}_{pp}\gamma_0^{\frac{1}{2}} - i\hbar\gamma_t^{\frac{1}{2}}\mathbf{M}_{qp}\gamma_0^{\frac{1}{2}} + \frac{i}{\hbar}\gamma_t^{-\frac{1}{2}}\mathbf{M}_{pq}\gamma_0^{-\frac{1}{2}}\right|^{\frac{1}{2}},$$

with monodromy matrix elements $\mathbf{M}_{\alpha\beta} = \frac{\partial\alpha_t}{\partial\beta_0}$, and γ_t is an $N \times N$ diagonal matrix describing the width of the coherent states. The coherent state wavefunctions in position and momentum space are,

$$\psi(\bar{\mathbf{x}}) \;=\; \langle\bar{\mathbf{x}}|\mathbf{q}\,\mathbf{p}\rangle = \left(\frac{\det|\gamma|}{\pi^N}\right)^{\frac{1}{4}} e^{-\frac{1}{2}(\bar{\mathbf{x}}-\mathbf{q})\cdot\gamma\cdot(\bar{\mathbf{x}}-\mathbf{q})+\frac{i}{\hbar}\mathbf{p}\cdot(\bar{\mathbf{x}}-\mathbf{q})}, \tag{7}$$

$$\tilde{\psi}(\bar{\mathbf{p}}) \;=\; \langle\bar{\mathbf{p}}|\mathbf{q}\,\mathbf{p}\rangle = \left(\frac{1}{\det|\gamma|\pi^N}\right)^{\frac{1}{4}} e^{-\frac{1}{2}(\bar{\mathbf{p}}-\mathbf{p})\cdot\gamma^{-1}\cdot(\bar{\mathbf{p}}-\mathbf{p})-\frac{i}{\hbar}\bar{\mathbf{p}}\cdot\mathbf{q}}, \tag{8}$$

respectively. HK-IVR is favorable over Eq. (4) and Eq. (5) since the coherent states give rise to integrals that usually involve Gaussian forms and because coherent state matrix elements generally provide a convenient choice for importance sampling both initial positions and momenta.

3 SC-IVR Correlation Functions

This package computes real-time correlation functions of the form

$$C_{AB}(t) \;=\; \mathrm{Tr}\left[\hat{A}\, e^{\frac{i}{\hbar}\hat{H}t}\hat{B}\, e^{-\frac{i}{\hbar}\hat{H}t}\right], \tag{9}$$

with operator $\hat{A}$ typically representing the initial state of the system, and operator $\hat{B}$ representing an observable. The SC-IVR correlation function is obtained by replacing the quantum propagators in Eq. (9) with SC approximations previously described. The double Van Vleck representation (DVV-IVR) of the correlation function is then,

$$\begin{aligned} C_{AB}(t) \;&=\; \frac{1}{(2\pi\hbar)^N}\int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \int \mathrm{d}\mathbf{p}_t' \int \mathrm{d}\mathbf{q}_t'\, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0,\mathbf{q}_0)+S_{-t}(\mathbf{p}_t,\mathbf{q}_t)]} \\ &\quad\times \det\left|\mathbf{M}_{qp}\,\mathbf{M}_{qp}^{'}\right|^{\frac{1}{2}} \langle\mathbf{q}_0|\hat{A}|\mathbf{q}_0'\rangle\,\langle\mathbf{q}_t'|\hat{B}|\mathbf{q}_t\rangle \end{aligned} \tag{10}$$

with $(\mathbf{p}_0, \mathbf{q}_0)$ and $(\mathbf{p}_t', \mathbf{q}_t')$ the initial conditions of a forward and backward trajectory, respectively. Similarly, the momentum representation is given by

$$\begin{aligned} C_{AB}(t) \;&=\; \frac{1}{(2\pi i\hbar)^N}\int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \int \mathrm{d}\mathbf{p}_t' \int \mathrm{d}\mathbf{q}_t'\, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0,\mathbf{q}_0)+S_{-t}(\mathbf{p}_t,\mathbf{q}_t)]} \\ &\quad\times \det\left|\mathbf{M}_{pq}\,\mathbf{M}_{pq}^{'}\right|^{\frac{1}{2}} \langle\mathbf{p}_0|\hat{A}|\mathbf{p}_0'\rangle\,\langle\mathbf{p}_t'|\hat{B}|\mathbf{p}_t\rangle\,. \end{aligned} \tag{11}$$

Finally there is the double Herman-Kluk correlation function (DHK-IVR),

$$\begin{aligned} C_{AB}(t) \;&=\; \frac{1}{(2\pi i\hbar)^{2N}}\int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \int \mathrm{d}\mathbf{p}_t' \int \mathrm{d}\mathbf{q}_t'\, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0,\mathbf{q}_0)+S_{-t}(\mathbf{p}_t,\mathbf{q}_t)]} \\ &\quad\times C_t(\mathbf{p}_0,\mathbf{q}_0)C_{-t}(\mathbf{p}_t',\mathbf{q}_t')\;\langle\mathbf{p}_0\mathbf{q}_0|\hat{A}|\mathbf{p}_0'\mathbf{q}_0'\rangle\,\langle\mathbf{p}_t'\mathbf{q}_t'|\hat{B}|\mathbf{p}_t\mathbf{q}_t\rangle\,. \end{aligned} \tag{12}$$

Though each of these time correlation functions can formally capture all important quantum effects, the oscillatory phase factors often make convergence impossible to achieve with large multidimensional systems. It is clear that additional approximations are required to extend the applicability of these methods to complex systems. Due to this challenge, and the other challenges associated with DVV-IVR that were previously outlined, DHK-IVR is included in this program but DVV-IVR and its momentum counterpart are not.

3.1 Linearized SC-IVR

The linearized approximation (LSC-IVR) assumes that the forward and backward trajectories employed in calculating correlation functions, as described in the section above, are very close to each other. It is derived by taking DVV-IVR and converting to mean and difference variables: $\Delta_\alpha = \alpha_t^{'} - \alpha_t$, $\bar{\alpha} = \frac{1}{2}(\alpha_t^{'} + \alpha_t)$, with $\alpha = \mathbf{p}, \mathbf{q}$. After expanding all time-dependent quantities to first order in the difference variables, the expression reduces to a single phase space average over the Wigner transforms of operators $\hat{A}$ and $\hat{B}$,

$$C_{AB}(t) \; = \; \frac{1}{(2\pi\hbar)^N} \int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \, A_w(\mathbf{p}_0, \mathbf{q}_0) \, B_w(\mathbf{p}_t, \mathbf{q}_t). \tag{13}$$

The Wigner transform of a quantum mechanical operator $\hat{\Omega}$ is given by,

$$O_w(\mathbf{p}, \mathbf{q}) \; = \; \int \mathrm{d}\bar{\mathbf{q}} \, e^{-\frac{i}{\hbar}\mathbf{p}\cdot\bar{\mathbf{q}}} \, \langle \mathbf{q} + \frac{1}{2}\bar{\mathbf{q}} | \hat{O} | \mathbf{q} - \frac{1}{2}\bar{\mathbf{q}} \rangle \, . \tag{14}$$

LSC-IVR has been widely used in simulations of condensed phase systems, however this method generally fails to capture quantum coherence effects. It is included in this package.

3.2 Mixed Quantum-Classical IVR

DHK-IVR is then the most appealing approach to incorporate all quantum effects in MD simulations, but convergence is often too difficult to achieve due to the oscillatory phase. One approach to this problem is to use modified Filinov filtration (MFF), which amounts to multiplying the integrand of DHK-IVR by a damping factor,

$$F(\mathbf{z}\,; \mathbf{c}) \; = \; \det \left| \mathbb{I} + i\mathbf{c}\frac{\partial^2 \phi}{\partial \mathbf{z}^2} \right|^{\frac{1}{2}} e^{-\frac{1}{2}\frac{\partial \phi}{\partial \mathbf{z}}^T \mathbf{c} \frac{\partial \phi}{\partial \mathbf{z}}} . \tag{15}$$

The damping factor down-weights the regions of $\mathbf{z}$ (i.e. the variables of integration) where the phase, $\phi(\mathbf{z})$, varies quickly. And the extent of the damping is determined by the diagonal matrix $\mathbf{c}$. In the limit that $\mathbf{c} \to 0$ then $F(\mathbf{z}, \mathbf{c}) \to 1$ and the original integral, DHK-IVR, is recovered. In the limit that $\mathbf{c} \to \infty$ then DHK-IVR is evaluated in the limit of stationary phase. MQC-IVR is derived by specifically defining $F(\mathbf{z}, \mathbf{c})$ to damp phase contributions from diverging trajectory pairs. The Forward-Backward (FB) form of MQC-IVR is given by,

$$\begin{aligned} C_{AB}(t) \; &= \frac{1}{(2\pi\hbar)^{2N}} \int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \int \mathrm{d}\boldsymbol{\Delta}_p \int \mathrm{d}\boldsymbol{\Delta}_q \, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0, \mathbf{q}_0) + S_{-t}(\mathbf{p}_t, \mathbf{q}_t)]} \\ &\times D_t(\mathbf{p}_0, \mathbf{q}_0, \boldsymbol{\Delta}_p, \boldsymbol{\Delta}_q; \mathbf{c}_p, \mathbf{c}_q) \, \langle \mathbf{p}_0 \mathbf{q}_0 | \hat{A} | \mathbf{p}_0' \mathbf{q}_0' \rangle \, \langle \mathbf{p}_t' \mathbf{q}_t' | \hat{B} | \mathbf{p}_t \mathbf{q}_t \rangle \\ &\times e^{-\frac{1}{2}\boldsymbol{\Delta}_q^T \mathbf{c}_q \boldsymbol{\Delta}_q} e^{-\frac{1}{2}\boldsymbol{\Delta}_p^T \mathbf{c}_p \boldsymbol{\Delta}_p}, \end{aligned} \tag{16}$$

with $\boldsymbol{\Delta}_q = \mathbf{q}_t' - \mathbf{q}_t$ and $\boldsymbol{\Delta}_p = \mathbf{p}_t' - \mathbf{p}_t$. The prefactor, D_t, is given by

$$\begin{aligned} D_t \; &= 2^{-\frac{N}{2}} \det(\gamma_0 \mathbf{G})^{\frac{1}{2}} \det \Big| \frac{1}{2}(\mathbf{M}_{pp}^b - i\gamma_0 \mathbf{M}_{qp}^b)(\mathbf{G}^{-1} + \mathbb{I})(\mathbf{M}_{pp}^f \gamma_0 + i\mathbf{M}_{pq}^f) \\ &+ (\gamma_0 \mathbf{M}_{qq}^b + i\mathbf{M}_{pq}^b)(\frac{1}{2}\gamma_t + \mathbf{c}_p)\mathbf{G}^{-1}(\mathbf{M}_{pp}^b \gamma_0 + i\mathbf{M}_{pq}^f) \\ &+ \frac{1}{2}(\mathbf{M}_{qq}^b + i\mathbf{M}_{pq}^b)(\mathbf{G}^{-1} + \mathbb{I})(\mathbf{M}_{qq}^f - i\mathbf{M}_{qp}^f \gamma_0) \\ &+ (\mathbf{M}_{pp}^b - i\gamma_0 \mathbf{M}_{qp}^b)(\frac{1}{2}\gamma_t + \mathbf{c}_q)\mathbf{G}^{-1}(\mathbf{M}_{qq}^b - i\mathbf{M}_{qp}^f \gamma_0) \Big|^{\frac{1}{2}} . \end{aligned} \tag{17}$$

Note that as the tuning parameters $\mathbf{c}_p, \mathbf{c}_q \to 0$ we recover the original DHK-IVR correlation function, and as $\mathbf{c}_p, \mathbf{c}_q \to \infty$ we obtain a classical limit result similar to LSC-IVR,

$$C_{AB}(t) \;=\; \frac{1}{(2\pi\hbar)^N} \int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \, \langle \mathbf{p}_0\mathbf{q}_0|\hat{A}|\mathbf{p}_0\mathbf{q}_0\rangle \, \langle \mathbf{p}_t\mathbf{q}_t|\hat{B}|\mathbf{p}_t\mathbf{q}_t\rangle \,, \tag{18}$$

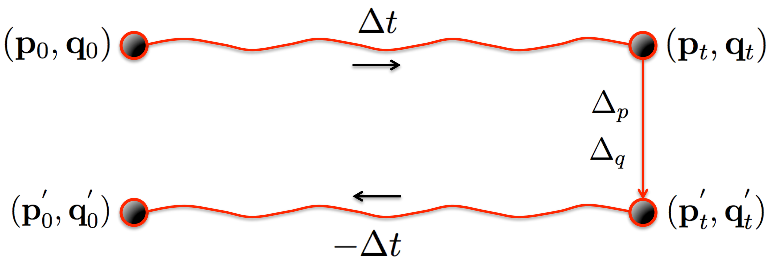
which we refer to as the Husimi-IVR. With intermediate values of $\mathbf{c}_p, \mathbf{c}_q$, one can select a set of tuning parameters that both reduces computational expense and recovers accurate quantum information. Moreover, the tuning parameters give us a handle over the ”quantumness” of any given degree of freedom in a dynamically consistent framework. The following two subsections describe two implementations of MQC-IVR.

3.2.1 Forward-Backward MQC-IVR

The first step in evaluating Eq. (16) is to sample the initial conditions of the forward trajectory $(\mathbf{p}_0, \mathbf{q}_0)$ as well as the phase space displacements at time t, $(\boldsymbol{\Delta}_p, \boldsymbol{\Delta}_q)$. For example, the models provided in this program define the initial state as a pure coherent state, $\hat{A} = |\mathbf{p}_i\mathbf{q}_i\rangle\,\langle\mathbf{p}_i\mathbf{q}_i|$, so one sampling distribution in this case may look like,

$$\omega(\mathbf{p}_0, \mathbf{q}_0, \boldsymbol{\Delta}_p, \boldsymbol{\Delta}_q) \;=\; |\,\langle \mathbf{p}_0\mathbf{q}_0|\mathbf{p}_i\mathbf{q}_i\rangle\,| e^{-\frac{1}{2}\boldsymbol{\Delta}_q^T \mathbf{c}_q \boldsymbol{\Delta}_q} e^{-\frac{1}{2}\boldsymbol{\Delta}_p^T \mathbf{c}_p \boldsymbol{\Delta}_p}. \tag{19}$$

The following diagram outlines how the program computes the dynamics of a single trajectory beginning at $(\mathbf{p}_0, \mathbf{q}_0)$. The initial point is propagated for a single time step Δt to point $(\mathbf{p}_t, \mathbf{q}_t)$, followed by an instantaneous jump in phase space to point $(\mathbf{p}_t', \mathbf{q}_t')$. From there, a new trajectory is propagated backward for time for Δt to point $(\mathbf{p}_0', \mathbf{q}_0')$. The process is repeated for each additional timestep.


$(\mathbf{p}_0, \mathbf{q}_0)$
Δt
$\longrightarrow$
$(\mathbf{p}_t, \mathbf{q}_t)$
Δ_p
Δ_q
$(\mathbf{p}_0^{'}, \mathbf{q}_0^{'})$
$\longleftarrow$
$-\Delta t$
$(\mathbf{p}_t^{'}, \mathbf{q}_t^{'})$

A symplectic integrator (see below for details) is used to time-evolve the monodromy matrix and classical action along with the corresponding forward and backward trajectories.

The dimensionality of the integral can be reduced if operator $\hat{B}$ is purely a function of the position or momentum operator. For example, when $\hat{B} = B(\hat{\mathbf{q}})$ we can collapse the coherent states at time t to position states by evaluating the following limit,

$$\lim_{\gamma_t \to \infty} \det(\gamma_t)^{\frac{1}{2}} \, \langle \mathbf{p}_t'\mathbf{q}_t'|B(\hat{\mathbf{q}})|\mathbf{p}_t\mathbf{q}_t\rangle \;=\; (4\pi)^{\frac{N}{2}} \delta(\boldsymbol{\Delta}_q) B\left(\frac{\mathbf{q}_t' + \mathbf{q}_t}{2}\right). \tag{20}$$

After inserting (20) into (16) and evaluating the integral over $\boldsymbol{\Delta}_q$ we have

$$\begin{aligned} C_{AB}(t) \;&= 2^{-2N}\pi^{-\frac{3N}{2}} \int \mathrm{d}\mathbf{p}_0 \int \mathrm{d}\mathbf{q}_0 \int \mathrm{d}\boldsymbol{\Delta}_p \, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0,\mathbf{q}_0)+S_{-t}(\mathbf{p}_t,\mathbf{q}_t)]} \\ &\quad \times D_q(\mathbf{p}_0, \mathbf{q}_0, \boldsymbol{\Delta}_p; \mathbf{c}_p) \, \langle \mathbf{p}_0\mathbf{q}_0|\hat{A}|\mathbf{p}_0'\mathbf{q}_0'\rangle \, B(\mathbf{q}_t) e^{-\frac{1}{2}\boldsymbol{\Delta}_p^T \mathbf{c}_p \boldsymbol{\Delta}_p}, \end{aligned} \tag{21}$$

with the prefactor now given by

$$\begin{aligned} D_q(\mathbf{p}_0,\mathbf{q}_0,\boldsymbol{\Delta}_p;\mathbf{c}_p) \ &= \ \det\bigg|\gamma_0^{-1}\mathbf{c}_p\bigg[(\mathbf{M}^b_{pp}-i\gamma_0\mathbf{M}^b_{qp})(\mathbf{M}^f_{pp}\gamma_0+i\mathbf{M}^f_{pq}) \\ &\quad +(\gamma_0\mathbf{M}^b_{qq}+i\mathbf{M}^b_{pq})(\mathbf{M}^f_{qq}-i\mathbf{M}^f_{qp}\gamma_0) \\ &\quad +(\mathbf{M}^b_{pp}-i\gamma_0\mathbf{M}^b_{qp})\mathbf{c}_p^{-1}(\mathbf{M}^f_{qq}-i\mathbf{M}^f_{qp}\gamma_0)\bigg]\bigg|^{\frac{1}{2}}. \end{aligned} \tag{22}$$

Only momentum jumps are imposed in this limit. Similarly, we can evaluate the limit that $\gamma_t \to 0$ to recover an expression when operator $\hat{B}$ is of the type $\hat{B} = B(\hat{\mathbf{p}})$,

$$\begin{aligned} C_{AB}(t) \ &= \frac{1}{(2\pi\hbar)^{2N}}\int \mathrm{d}\mathbf{p}_0\int \mathrm{d}\mathbf{q}_0\int \mathrm{d}\boldsymbol{\Delta}_q\, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0,\mathbf{q}_0)+S_{-t}(\mathbf{p}_t,\mathbf{q}_t)]} \\ &\quad \times D_p(\mathbf{p}_0,\mathbf{q}_0,\boldsymbol{\Delta}_q;\mathbf{c}_q)\,\langle\mathbf{p}_0\mathbf{q}_0|\hat{A}|\mathbf{p}_0'\mathbf{q}_0'\rangle\, B(\mathbf{p}_t)e^{-\frac{1}{2}\boldsymbol{\Delta}_q^T\mathbf{c}_p\boldsymbol{\Delta}_q}, \end{aligned} \tag{23}$$

with the prefactor

$$\begin{aligned} D_p(\mathbf{p}_0,\mathbf{q}_0,\boldsymbol{\Delta}_q;\mathbf{c}_q) \ &= \ \det\bigg|\gamma_0^{-1}\mathbf{c}_q\bigg[(\mathbf{M}^b_{pp}-i\gamma_0\mathbf{M}^b_{qp})(\mathbf{M}^f_{pp}\gamma_0+i\mathbf{M}^f_{qp}) \\ &\quad +(\gamma_0\mathbf{M}^b_{qq}+i\mathbf{M}^b_{pq})(\mathbf{M}^f_{qq}-i\mathbf{M}^f_{qp}\gamma_0) \\ &\quad +(\gamma_0\mathbf{M}^b_{qq}+i\mathbf{M}^b_{pq})\mathbf{c}_q^{-1}(\mathbf{M}^f_{pp}\gamma_0+i\mathbf{M}^f_{pq})\bigg]\bigg|^{\frac{1}{2}}. \end{aligned} \tag{24}$$

When evaluating Eq. (21) the sampling distribution is the same as before, Eq. (19), but with $\boldsymbol{\Delta}_q \to 0$. Likewise when evaluating Eq. (23) take $\boldsymbol{\Delta}_p \to 0$.

Note that only the general FB implementation of MQC-IVR, Eq. (16), is included as a default option in this package. But the subroutines required to sample initial conditions and compute prefactors for the specific cases of Eq. (21) and Eq. (23) are also provided.

3.2.2 Double-Forward MQC-IVR

It is far more efficient to propagate two independent trajectories forward in time than it is to use the forward-backward implementation of the previous section. MQC-IVR can be re-derived to exploit this efficiency. Begin by using Liouville's theorem, $\mathrm{d}\mathbf{p}_t'\mathrm{d}\mathbf{q}_t' = \mathrm{d}\mathbf{p}_0'\mathrm{d}\mathbf{q}_0'$, to rewrite DHK-IVR,

$$\begin{aligned} C_{AB}(t) \ &= \frac{1}{(2\pi\hbar)^{2N}}\int \mathrm{d}\mathbf{p}_0\int \mathrm{d}\mathbf{q}_0\int \mathrm{d}\mathbf{p}_0'\int \mathrm{d}\mathbf{q}_0'\, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0,\mathbf{q}_0)-S_t(\mathbf{p}_0',\mathbf{q}_0')]} \\ &\quad \times C_t(\mathbf{p}_0,\mathbf{q}_0)\,C_t^*(\mathbf{p}_0',\mathbf{q}_0')\ \langle\mathbf{p}_0\mathbf{q}_0|\hat{A}|\mathbf{p}_0'\mathbf{q}_0'\rangle\,\langle\mathbf{p}_t'\mathbf{q}_t'|\hat{B}|\mathbf{p}_t\mathbf{q}_t\rangle\,. \end{aligned} \tag{25}$$

We now introduce a very similar MFF damping factor to filter trajectory pairs that are widely displaced only at time $t = 0$, rather than at all t. The final result is given by

$$\begin{aligned} C_{AB}(t) \ &= \frac{1}{(2\pi\hbar)^{2N}}\int \mathrm{d}\mathbf{p}_0\int \mathrm{d}\mathbf{q}_0\int \mathrm{d}\mathbf{p}_0'\int \mathrm{d}\mathbf{q}_0'\, e^{\frac{i}{\hbar}[S_t(\mathbf{p}_0,\mathbf{q}_0)-S_t(\mathbf{p}_0',\mathbf{q}_0')]} \\ &\quad \times D_t(\mathbf{p}_0,\mathbf{q}_0,\mathbf{p}_0',\mathbf{q}_0';\mathbf{c}_p,\mathbf{c}_q)\,\langle\mathbf{p}_0\mathbf{q}_0|\hat{A}|\mathbf{p}_0'\mathbf{q}_0'\rangle\,\langle\mathbf{p}_t'\mathbf{q}_t'|\hat{B}|\mathbf{p}_t\mathbf{q}_t\rangle \\ &\quad \times e^{-\frac{1}{2}\boldsymbol{\Delta}_{q_0}^T\mathbf{c}_q\boldsymbol{\Delta}_{q_0}}\,e^{-\frac{1}{2}\boldsymbol{\Delta}_{p_0}^T\mathbf{c}_p\boldsymbol{\Delta}_{p_0}}\,. \end{aligned} \tag{26}$$

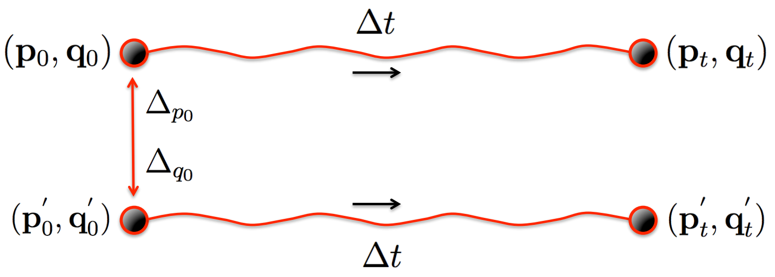
with $\mathbf{\Delta}_{p_0} = \mathbf{p}_0' - \mathbf{p}_0$ and $\mathbf{\Delta}_{q_0} = \mathbf{q}_0' - \mathbf{q}_0$, and the prefactor is given by

$$\begin{aligned} D_t \;=\; & \det(\frac{1}{2}\gamma_t^{-1}\mathbf{G})^{\frac{1}{2}} \det \Bigg| \frac{1}{2}(\mathbf{M}_{pp}^f - i\gamma_t \mathbf{M}_{qp}^f)(\mathbf{G}^{-1} + \mathbb{I})(\mathbf{M}_{pp}^b \gamma_t + i\mathbf{M}_{pq}^b) \\ & +(\gamma_t \mathbf{M}_{qq}^f + i\mathbf{M}_{pq}^f)(\frac{1}{2}\gamma_0^{-1} + \mathbf{c}_p)\mathbf{G}^{-1}(\mathbf{M}_{pp}^b \gamma_t + i\mathbf{M}_{pq}^b) \\ & +\frac{1}{2}(\gamma_t \mathbf{M}_{qq}^f + i\mathbf{M}_{pq}^f)(\mathbf{G}^{-1} + \mathbb{I})(\mathbf{M}_{qq}^b - i\mathbf{M}_{qp}^b \gamma_t) \\ & +(\mathbf{M}_{pp}^f - i\gamma_t \mathbf{M}_{qp}^f)(\frac{1}{2}\gamma_0 + \mathbf{c}_q)\mathbf{G}^{-1}(\mathbf{M}_{qq}^b - i\mathbf{M}_{qp}^b \gamma_t) \Bigg]^{\frac{1}{2}}. \end{aligned} \tag{27}$$

The limits of the filtering strength are identical to the FB implementation, yielding Eq. (25) as $\mathbf{c}_p, \mathbf{c}_q \to 0$, and yielding the Husimi-IVR as $\mathbf{c}_p, \mathbf{c}_q \to \infty$. In this program, the default sampling distribution generates mean ($\bar{x} = \frac{1}{2}(x_0^{'} - x_0)$) and difference variables at time zero,

$$\omega(\mathbf{p}_0, \mathbf{q}_0, \mathbf{p}_0', \mathbf{q}_0') \;=\; |\langle \bar{\mathbf{p}}\, \bar{\mathbf{q}} | \mathbf{p}_i \mathbf{q}_i \rangle| e^{-\frac{1}{2}\mathbf{\Delta}_{p_0} \mathbf{c}_p \mathbf{\Delta}_{p_0}} e^{-\frac{1}{2}\mathbf{\Delta}_{q_0} \mathbf{c}_q \mathbf{\Delta}_{q_0}}. \tag{28}$$

From the mean and difference variables we can transform to $(\mathbf{p}_0, \mathbf{q}_0)$ and $(\mathbf{p}_0', \mathbf{q}_0')$. Each of these points are then independently propagated forward in time.


$(\mathbf{p}_0, \mathbf{q}_0)$
Δt
$(\mathbf{p}_t, \mathbf{q}_t)$
Δ_{p_0}
Δ_{q_0}
$(\mathbf{p}_0^{'}, \mathbf{q}_0^{'})$
$(\mathbf{p}_t^{'}, \mathbf{q}_t^{'})$
Δt

Under the FB implementation, the total number of phase space points to evolve per trajectory pair goes as $\frac{1}{2}N(N_t^2 + 3N_t)$, where N_t is the number of time steps. With the DF implementation, however, the scaling goes as $2NN_t$: a significant improvement in computational expense.

4 Analytical Quantum-Classical IVR

One existing challenge with DF MQC-IVR is the task of choosing an optimal set of tuning parameters to minimize phase information as well as loss of accuracy. One approach to this challenge is to simply choose a subset of full-system dofs to quantize (say we choose F of them), and evaluate DF MQC-IVR in the limit that the tuning parameters of the F quantum dofs go to zero, and the limit that the tuning parameters of the remaining $N - F$ classical dofs go to infinity. The result is an analytical quantum-classical limit of DHK-IVR,

$$\begin{aligned} C_{AB}(t) =& \frac{1}{(2\pi)^{N+F}} \int d\mathbf{z} \prod_{j=1}^{F} \left[\int dz_j' \right] e^{i\left[S_t(\mathbf{z}) - S_t(z_1', \ldots, z_F', z_{F+1}, \ldots, z_N)\right]} \\ & \times C_t(\mathbf{z}; \gamma_0, \gamma_t) C_t^*(z_1', \ldots, z_F', z_{F+1}, \ldots, z_N) \Lambda_t(\mathbf{z}, z_1', \ldots, z_F'). \end{aligned} \tag{29}$$

For convenience we have used a different notation to represent the phase space variables, $\mathbf{z} = (\mathbf{p}_0, \mathbf{q}_0)$. Note that the initial conditions of the classical dofs are the same in a given trajectory pair, and the initial conditions of the quantum dofs are different. This similarity in initial conditions gives rise to significant phase cancellation, but the displacements among the quantum dofs gives rise to enough phase information so as to retain quantum mechanical accuracy. The additional prefactor, $\Lambda_t(\mathbf{z}, z'_1, \ldots, z'_F)$, accounts for treating the classical dofs in the stationary phase limit of MFF and is given by

$$\Lambda_t(\mathbf{z}, z'_1, \ldots, z'_F) = (-1)^{\frac{N}{2}} \det\left[\mathbb{I} + \mathbf{\Sigma}(\mathbf{K}^{\mathrm{T}})^{-1}\right]^{\frac{1}{2}}, \tag{30}$$

where $\mathbb{I}$ is the $4N \times 4N$ identity matrix, $\mathbf{K}^{\mathrm{T}}$ is a complex $4N \times 4N$ matrix given by,

$$\mathbf{K}^{\mathrm{T}} = \begin{pmatrix} \mathbf{X} & \mathbf{X}^* \\ \mathbf{Y}\mathbf{M}' & \mathbf{Y}^*\mathbf{M} \end{pmatrix}, \tag{31}$$

with $2N \times 2N$ blocks,

$$\mathbf{X} = \begin{pmatrix} \frac{i}{2}\gamma_0 & -\frac{1}{2}\mathbb{I} \\ \frac{1}{2}\mathbb{I} & \frac{i}{2}\gamma_0^{-1} \end{pmatrix}, \tag{32}$$

$$\mathbf{Y} = \begin{pmatrix} \frac{i}{2}\gamma_t & \frac{1}{2}\mathbb{I} \\ -\frac{1}{2}\mathbb{I} & \frac{i}{2}\gamma_t^{-1} \end{pmatrix}, \tag{33}$$

and $\mathbf{M}$ and $\mathbf{M}'$ are the full $2N \times 2N$ monodromy matrices for the trajectories beginning at $\mathbf{z}$ and $\mathbf{z}'$, respectively. We also have,

$$\mathbf{\Sigma} = \begin{pmatrix} \mathbf{\Omega} & \mathbf{\Omega}^* \\ \mathbb{O} & \mathbb{O} \end{pmatrix}, \tag{34}$$

$$(\mathbf{\Omega})_{jk} = \begin{cases} i(1 - \frac{1}{2}\gamma_{jj})\delta_{j,k} & F < j, k \leq N, \\ i(1 - \frac{1}{2}\gamma_{jj}^{-1})\delta_{j,k} & N + F < j, k \leq 2N, \\ -\frac{1}{2}\delta_{j-N,k} & N + F < j \leq 2N, F < k \leq N, \\ \frac{1}{2}\delta_{j,k-N} & F < j \leq N, N + F < k \leq 2N \\ 0 & \text{else.} \end{cases} \tag{35}$$

5 Semiclassical Nonadiabatic Dynamics

One way of representing a general F-level vibronic system with continuous variables is with the Meyer-Miller-Stock-Thoss (MMST) Hamiltonian,

$$H = \frac{1}{2}\mathbf{P}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{P} + \frac{1}{2}\mathbf{p}^{\mathrm{T}}\mathbf{V}(\mathbf{R})\mathbf{p} + \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{V}(\mathbf{R})\mathbf{x} - \frac{1}{2}\operatorname{Tr}\left[\mathbf{V}(\mathbf{R})\right], \tag{36}$$

where $(\mathbf{P}, \mathbf{R})$ and $(\mathbf{x}, \mathbf{p})$ are nuclear and electronic phase space variables, respectively, $\mathbf{V}(\mathbf{R})$ is the $F \times F$ diabatic electronic potential energy matrix, $\mathbf{M}$ is a $G \times G$ diagonal matrix of nuclear masses, and G is the number of nuclear degrees of freedom. In this representation, the electronic mapping variables $(\mathbf{x}, \mathbf{p})$ are restricted to the singly-excited harmonic oscillator subspace. For example, the total wavefunction of a nuclear wavepacket, $|\phi\rangle$, occupying electronic state s in an F-level system may be given by

$$|\Psi\rangle = |\phi\, 0_1 0_2 \ldots 1_s \ldots 0_F\rangle, \tag{37}$$

where $|0_\lambda\rangle$ and $|1_\lambda\rangle$ are the ground and first excited harmonic oscillator states associated with the mapping variables of electronic state λ, (x_λ, p_λ).

The SC correlation functions outlined in the previous sections can easily be applied to systems evolved under the MMST Hamiltonian because each degree of freedom, both nuclear and electronic, is represented by a continuous phase space variable. The MInt algorithm for computing trajectories and the monodromy matrix with the MMST Hamiltonian is outlined in the following section.

5.1 The MInt Algorithm

It is known that when a Hamiltonian is decomposed to a sum of sub-Hamiltonians, a symmetric composition of exact evolutions under the sub-Hamiltonians results in an approximate evolution under the total Hamiltonian that is exactly symplectic. By symplectic we mean that the following condition,

$$\mathbf{M}^{\mathrm{T}}\mathbf{J}^{\mathbf{-1}}\mathbf{M} = \mathbf{J}^{\mathbf{-1}}, \tag{38}$$

where $\mathbf{M}$ is the monodromy matrix and $\mathbf{J}$ is the structure matrix,

$$\mathbf{J} = \begin{pmatrix} \mathbb{O} & \mathbb{I} \\ -\mathbb{I} & \mathbb{O} \end{pmatrix}, \tag{39}$$

is conserved exactly. We therefore split Eq. (36) into two parts,

$$H = H_1 + H_2 \tag{40}$$

$$H_1 = \frac{1}{2}\mathbf{P}^{\mathrm{T}}\mathbf{M}^{\mathbf{-1}}\mathbf{P} \tag{41}$$

$$H_2 = \frac{1}{2}\mathbf{p}^{\mathrm{T}}\mathbf{V}(\mathbf{R})\mathbf{p} + \frac{\mathbf{1}}{\mathbf{2}}\mathbf{x}^{\mathrm{T}}\mathbf{V}(\mathbf{R})\mathbf{x} - \frac{\mathbf{1}}{\mathbf{2}}\,\mathrm{Tr}\left[\mathbf{V}(\mathbf{R})\right], \tag{42}$$

and define the following flow map,

$$\Psi_{H,\Delta t} = \Phi_{H_1,\Delta t/2} \circ \Phi_{H_2,\Delta t} \circ \Phi_{H_1,\Delta t/2}, \tag{43}$$

which defines the propagation scheme. The symbol $\Phi_{H_i,t}$ represents exact evolution under Hamiltonian H_i for time t, i.e. $\Phi_{H_i,t}(\mathbf{z_0}) = \mathbf{z_t}$, and the circles represent the composition operation: $f \circ g(z) = f(g(z))$. In words, Eq. (43) says to first propagate the system under H_1 for half a timestep, then propagate the system under H_2 for a full timestep, and then under H_1 again for another half timestep. So long as the sub-evolutions are exact the total evolution, $\Psi_{H,\Delta t}$, will be exactly symplectic. Hamilton's equations are therefore solved according to the scheme in Eq. (43). See reference 8 of Section 1.1 for a complete derivation of the equations of motion for the phase space variables and monodromy matrix. Also see *J. Chem. Phys.*, **136** 084101 (2012) by Kelly et. al. for additional reference.

6 SC-IVR Code Package

6.1 Program Overview

The package contains a collection of FORTRAN programs that compute the SC-IVR correlation functions of Eqs. (12), (13), (16), (18), (21), (23) and (26) for 1D, multidimensional, and nonadiabatic systems. The input file `theory.in` allows you to choose which SC-IVR to use, and defines all simulation parameters one would need. We provide three adiabatic model systems that may serve as a template for other systems of your choosing: the harmonic oscillator,

$$V(x) \;=\; \frac{1}{2}m\omega^2x^2, \tag{44}$$

an anharmonic oscillator

$$V(x) \;=\; \frac{1}{2}m\omega^2x^2 - 0.1x^3 + 0.1x^4, \tag{45}$$

and a 2D system-bath model where Eq. (45) is coupled to a harmonic mode

$$V(x,y) \;=\; \frac{1}{2}m_x\omega_x^2x^2 - 0.1x^3 + 0.1x^4 + \frac{1}{2}m_y\omega_y^2y^2 + kxy. \tag{46}$$

Also included is a nonadiabatic model system with two electronic states and one nuclear degree of freedom. Elements of the diabatic electronic potential energy matrix for this system are given by

$$V_{11}(R) = V_0\left[1+\tanh(\alpha R)\right] \tag{47}$$

$$V_{22}(R) = V_0\left[1-\tanh(\alpha R)\right] \tag{48}$$

$$V_{12}(R) = ae^{-bR^2}, \tag{49}$$

with parameters specified in the program.

For the adiabatic model systems, the program computes a quantity of the form

$$\langle B\rangle_t \;=\; \langle \mathbf{p}_i\mathbf{q}_i|e^{\frac{i}{\hbar}\hat{H}t}\hat{B}e^{-\frac{i}{\hbar}\hat{H}t}|\mathbf{p}_i\mathbf{q}_i\rangle\,, \tag{50}$$

which is Eq. (9) with an initial coherent state projection for operator $\hat{A} = |\mathbf{p}_i\mathbf{q}_i\rangle\,\langle\mathbf{p}_i\mathbf{q}_i|$. With the provided models, you have the choice of taking $\hat{B} = \hat{\mathbf{q}}$ or $\hat{B} = \hat{\mathbf{p}}$. In the nonadiabatic model system, the program computes the particle's momentum distribution after it traverses the crossing region,

$$C(P_f) = \lim_{t\to\infty}\langle P_{Ri}R_i 1_1 0_2|\delta(P_f-\hat{P})|P_{Ri}R_i 1_1 0_2\rangle\,, \tag{51}$$

given that the initial state of the system is a nuclear coherent state occupying electronic state 1.

6.2 Running the Program

Here, we assume a UNIX or Windows-X or MAC-OSX terminal environment. In the parent directory, specify the SC-IVR of your choice as well as your input parameters in the file `theory.in`, and then execute the bash script `execute.sh`. This will generate a directory called `EXPERIMENT` which contains the program you specified. Typing `make` while in the `EXPERIMENT` directory will compile the program and generate an executable file `dyn.x`. To run the program, type `./dyn.x` to start the simulation. All system parameters in the `input_1D` (`input_mD`) file can be edited without having to recompile the program, but changing the first five system options will do nothing while you are in the `EXPERIMENT` directory; these can only be specified in the parent directory.

Note that executing the bash script `execute.sh` will overwrite an existing `EXPERIMENT` directory, so the `EXPERIMENT` directories you generate should be renamed to be safe.

If you intend to parallelize the calculation with MPI, then use `jobrun.sh` to submit the script (see below, this script will need to be edited according to your computational resources).

6.3 The Makefile, External Libraries, and the MPI Submission Script

In the directory `makefiles` is a universal makefile. When the script `execute.sh` is run, the makefile is copied into the `EXPERIMENT` directory, and a simple search-and-replace routine tailors the makefile to your specifications there. Depending on your operating system, the makefile in the `makefiles` directory may need to be edited. Note that we specify paths to the LAPACK and BLAS libraries in the makefile, and mpif90 is used as well.

Also note that we do NOT encourage you to edit the names of any FORTRAN files, modules or scripts that do not sit in the `EXPERIMENT` directory, as this may prevent the program from assembling the `EXPERIMENT` directory appropriately.

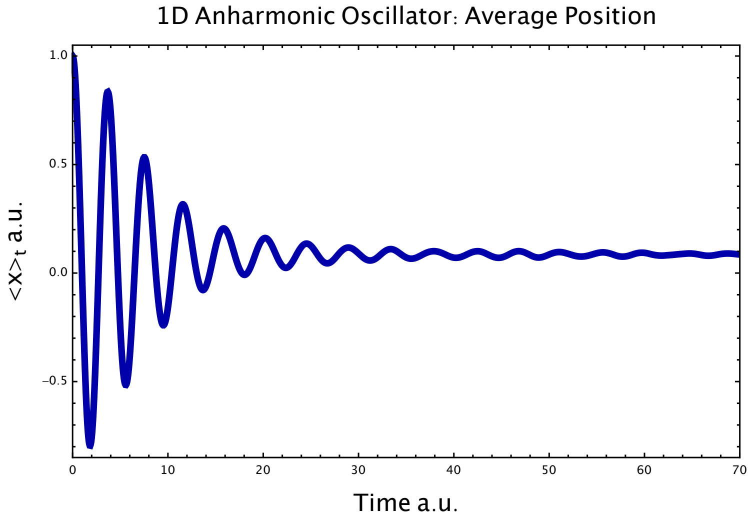
In the directory `Scripts` there is a submission script called `jobrun.sh`. This uses the Portable Batch System (PBS) scheduler, and will need to be edited if another scheduler is used.

6.4 Tutorial 1: 1D Anharmonic Oscillator

In the parent directory, set up the input file `theory.in` to compute the average position of the anharmonic oscillator with the Husimi-IVR.

```
Degrees of freedom
1
Level of theory ( ...
2
Implementation ( ...
1
Type of observable ( ...
1
Model Potentials ( ...
2
List the diagonal ...
1.0
List the coherent state and ...
1.0     0.0     1.414     1.414     10.0     10.0
Timestep, number of timesteps, ...
0.05     1400     1e-5
Number of trajectories
120000
```

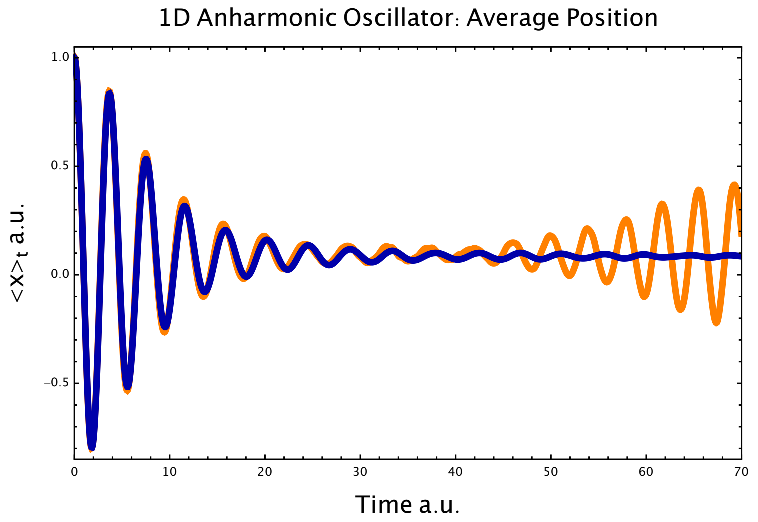
Leave an arbitrary number for the tuning parameters when the MQC-IVR is not being used, in this case we use 10.0. Also, even though the `Implementation` line is irrelevant in this case, leave it as 1 for the Husimi and LSC-IVRs. Execute the bash script with `sh execute.sh`. This generates a directory called `EXPERIMENT` which contains the program specified by the input file. Rename the `EXPERIMENT` directory and `cd` into it. There are several FORTRAN files in the experiment directory. The input file in this directory is called `input_1D`: a copy of the `theory.in` file from the parent directory. The file `MonteCarlo_1D.f90` contains subroutines that generate initial configurations for each of the 1D implementations, `params_1D.f90` contains the global parameters and a subroutine that reads the input file, `traj_1D.f90` contains subroutines that propagate the classical trajectories (depending upon the type of SC-IVR), `potential_1D.f90` contains the model potentials, `jobrun.sh` is a submission script for jobs using MPI, and `supply_1D.f90` contains a symplectic integrator and various subroutines that compute matrix elements and prefactors. The file `HUS_timecorr_1D.f90` contains the program which computes the correlation function, drawing upon the subroutines that you originally specified in `theory.in`. Note that the default potential parameters (in this case the frequency ω) are hard-coded in the potential subroutine of the file `potential_1D.f90`. Start the simulation with `./dyn.x` ($\sim$ 20sec) and plot the real part of the result contained in the file `TCF.out`. This should resemble the following figure.



Now go back to the parent directory and start a new experiment, identical to the previous one but with the Double-Forward MQC-IVR and $c_p = c_q = 10$. The input file `theory.in` should read

```
Degrees of freedom
1
Level of theory ( ...
3
Implementation ( ...
1
Type of observable ( ...
1
Model Potentials ( ...
2
List the diagonal ...
1.0
List the coherent state and ...
1.0     0.0     1.414     1.414      10.0     10.0
Timestep, number of timesteps, ...
0.05    1400     1e-5
Number of trajectories
120000
```

Compile and run the program as before ($\sim$ 1min) and compare the two results. The large amplitudes at later times should be apparent in the MQC-IVR result, as shown in the figure below. This is characteristic of the exact quantum result, which the MQC-IVR approaches as the $\mathbf{c}_p, \mathbf{c}_q \to 0$. This limit however requires more trajectories.

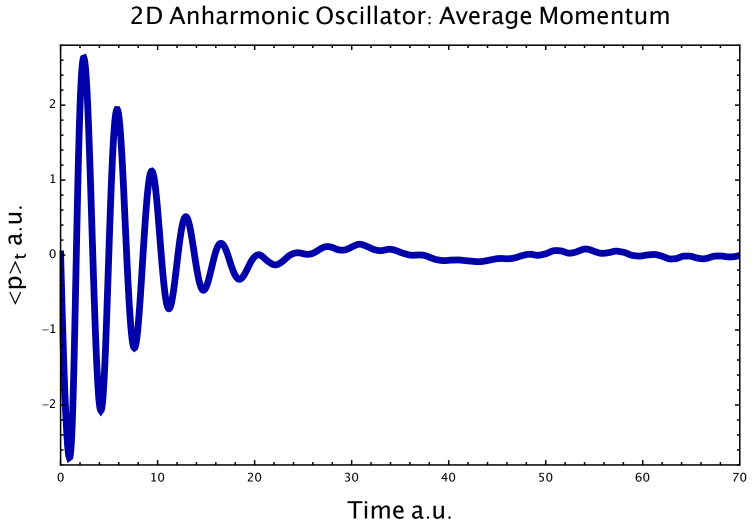


6.5 Tutorial 2: 2D Anharmonic Oscillator

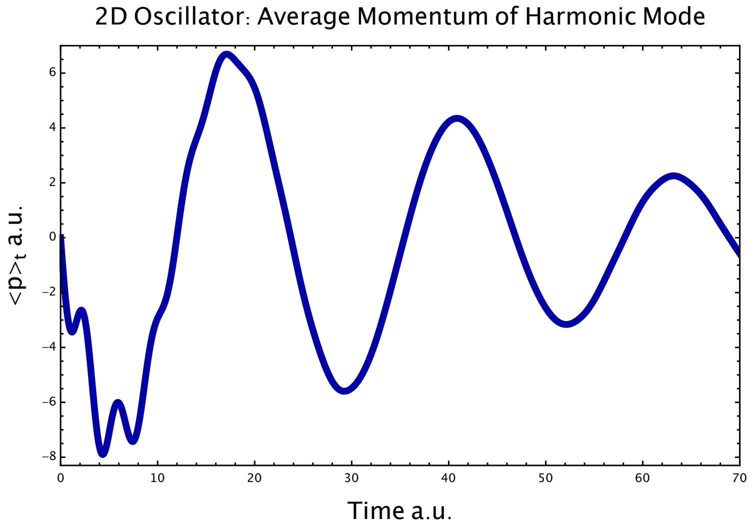
Now we will compute the average momentum of an anharmonic oscillator that is coupled to a heavy harmonic mode with the Husimi-IVR. Recall that the potential energy parameters (in this case the two frequencies and coupling constant) are hard-coded into the potential energy subroutine. The `theory.in` file should read

```
Degrees of freedom
2
Level of theory ( ...
2
Implementation ( ...
1
Type of observable ( ...
2
Model Potentials ( ...
2
List the diagonal ...
1.0     25.0
List the coherent state and ...
1.0     0.0     1.414     1.414     10.0     10.0
1.0     0.0     8.333     8.333     10.0   10.0
Timestep, number of timesteps, ...
0.05    1400     1e-5
Number of trajectories
36000
```

With this input file, generate an `EXPERIMENT` directory, compile, run the simulation ($\sim$ 1min), and plot the result in `TCF.out`. The real part is plotted in the following figure.



In order to observe the average momentum of the harmonic mode of this system rather than the anharmonic mode, `vi` into the file `HUS_timecorr_mD.f90` and change the `doflabel` in line 37 from 1 to 2. Re-compile with `make` and then run the simulation. The real part of the output is shown in the following figure.



6.6 Altering the Code

We do not advise that you alter any files, directories, or scripts unless they created in an `EXPERIMENT` directory. In order to design your own experiment, it is suggested that you use `theory.in` to generate an `EXPERIMENT` directory with the SC-IVR of your choosing with an arbitrary model potential. Then while working in the `EXPERIMENT` directory you have the freedom to alter the potential and any subroutine you may wish to edit. The next two tutorials suggest how an original experiment can be created.

6.6.1 Tutorial 3: Other Potentials

We wish to calculate the average position of the anharmonic oscillator which is now coupled to two identical harmonic bath modes. We'll use the Double-Forward MQC-IVR and a large tuning parameter to set up `theory.in` as follows.

```
Degrees of freedom
3
Level of theory ( ...
3
Implementation ( ...
1
Type of observable ( ...
1
Model Potentials ( ...
2
List the diagonal ...
1.0     25.0     25.0
List the coherent state and ...
1.0     0.0     1.414     1.414      1000      1000
1.0     0.0     8.333     8.333      1000      1000
1.0     0.0     8.333     8.333      1000      1000
Timestep, number of timesteps, ...
0.05    1400     1e-5
Number of trajectories
36000
```

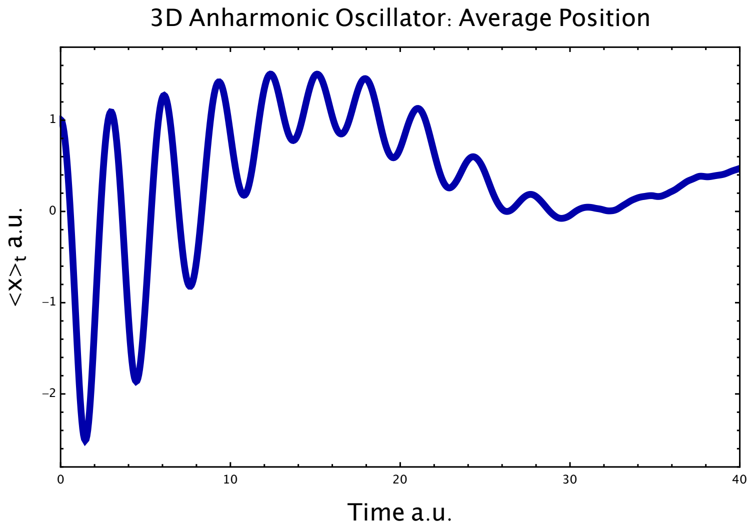
Generate an `EXPERIMENT` directory with this input file and `cd` into it. The program will compile but the potential energy subroutine needs to account for the third body. This is found in `potential_mD.f90`; `vi` into this file and edit the potential to include the second harmonic mode. Use the same potential parameters as the other bath mode. The subroutine should now read

```
v = x(1)**2 - 0.1d0*x(1)**3 + 0.1d0*x(1)**4   &
     + 0.5d0*ky*x(2)**2 + coupling*x(1)*x(2)  &
     + 0.5d0*ky*x(3)**2 + coupling*x(1)*x(3)

dv(1) = 2.d0*x(1) - 0.3d0*x(1)**2 + 0.4d0*x(1)**3 &
          + coupling*x(2) + coupling*x(3)
dv(2) = ky*x(2) + coupling*x(1)
dv(3) = ky*x(3) + coupling*x(1)

d2v(1,1) = 2.d0 - 0.6d0*x(1) + 1.2d0*x(1)**2
d2v(1,2) = coupling
d2v(1,3) = coupling
d2v(2,1) = coupling
d2v(2,2) = ky
d2v(2,3) = 0.d0
d2v(3,1) = coupling
d2v(3,2) = 0.d0
d2v(3,3) = ky
```

Compile and run the program (∼ 4min). The real part of the output should look as follows.



6.6.2 Tutorial 4: Other Observables

If the observable you wish to use is not one of the two provided ($\hat{\mathbf{p}}$ or $\hat{\mathbf{q}}$), then expressions for the following must be derived and coded into the program.

$$\begin{array}{cc}
\langle \mathbf{p}_t'\mathbf{q}_t'|\hat{B}|\mathbf{p}_t\mathbf{q}_t\rangle & \text{(FF-MQC-IVR, DHK-IVR)} \\
\langle \mathbf{p}_t\mathbf{q}_t|\hat{B}|\mathbf{p}_t\mathbf{q}_t\rangle & \text{(HUSIMI-IVR)} \\
B_w(\mathbf{p}_t, \mathbf{q}_t) & \text{(LSC-IVR)} \\
B(\mathbf{p}_t, \mathbf{q}_t) & \text{(FB-MQC-IVR)}
\end{array}$$

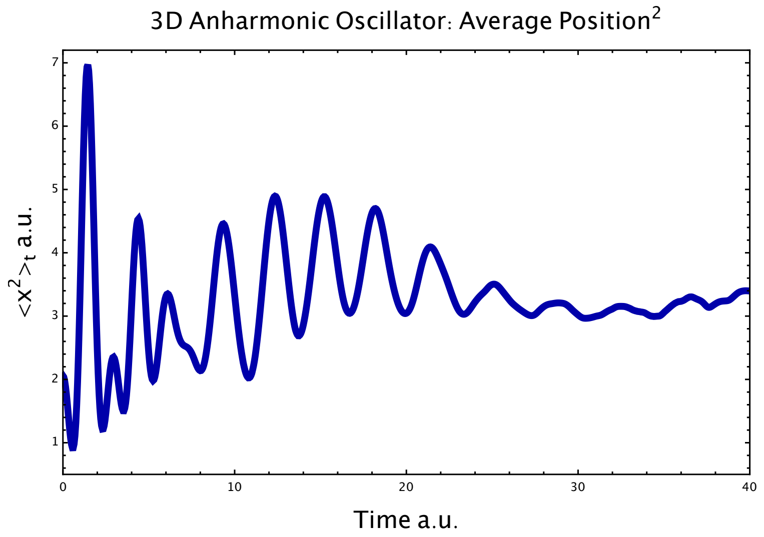
The subroutines in the supply_*.f90 files can be edited for this purpose if you don't wish to write your own. For example, repeat the calculation from the previous tutorial but use $\hat{B} = \hat{x}^2$ rather than $\hat{x}$. To evaluate the matrix element, insert a complete set of position states and evaluate the integral analytically,

$$\begin{aligned}
\langle \mathbf{p}_t'\mathbf{q}_t'|\hat{x}^2|\mathbf{p}_t\mathbf{q}_t\rangle &= \int \mathrm{d}\mathbf{x}\, \langle \mathbf{p}_t'\mathbf{q}_t'|\hat{x}^2|\mathbf{x}\rangle \langle \mathbf{x}|\mathbf{p_t q_t}\rangle \\
&= \langle p_{zt}^{'} z_t^{'}|p_{zt}z_t\rangle \langle p_{yt}^{'} y_t^{'}|p_{yt}y_t\rangle \int \mathrm{d}x\, x^2 \langle p_{xt}^{'} x_t^{'}|x\rangle \langle x|p_{zt}z_t\rangle \\
&= \frac{1}{4}\left(\frac{2}{\gamma_x} - \frac{1}{\gamma_x^2}\left(p_{tx}^{'} - p_{tx} + i(q_{tx} + q_{tx}^{'})\gamma_x\right)^2\right) \langle \mathbf{p}_t'\mathbf{q}_t'|\mathbf{p}_t\mathbf{q}_t\rangle\,,
\end{aligned}$$

where we have used both Eq. (7) and Eq. (**??**). The code near line 132 of supply_mD.f90 should now be edited to look something like the following.

```
Bq = 0.25d0*(2.d0/WidthT(1,1)-(pp(1)-p(1)+Iu*(q(1)+qp(1))&
     *WidthT(1,1))**2/WidthT(1,1)**2)&
     *dexp(-0.25d0*dot_product(q-qp,q1)-0.25d0*dot_product(p-pp,p1))&
     *cdexp(-0.5d0*Iu*dot_product(p+pp,q-qp))
```

Compile and run the program, the output is plotted in the following figure.



6.7 Tutorial 5: Nonadiabatic Dynamics with MQC-IVR

An `EXPERIMENT` directory containing a program that uses DF MQC-IVR to simulate the model nonadiabatic system can be created simply by entering `-1` under `Degrees of freedom` in the `theory.in` file.

```
Degrees of freedom
-1
```

All other entries in the input file are neglected here. Running the script `execute.sh` will then generate the `EXPERIMENT` directory containing the appropriate program. Note that it is typical of a semiclassical calculation of nonadiabatic systems to require a large number of trajectories to converge. So a quantum-limit calculation with MQC-IVR will require parallelization. See reference 8 of section 1.1 for an idea of the convergence requirements.

6.8 The `MonteCarlo` files

In each of the `MonteCarlo` subroutines we use a Gaussian random number generator to produce initial configurations. The function `gauss` takes in the standard deviation σ of the sampling distribution and outputs a phase-space point. For example, the following sampling distribution is used with the 1D Husimi-IVR,

$$\omega(p_0, q_0) \;=\; \frac{1}{2\pi} e^{-\frac{\gamma_0}{2}(q_0-q_i)^2} e^{-\frac{1}{2\gamma_0}(p_0-p_i)^2}. \tag{52}$$

So the use of

```
gauss(1/dsqrt(Width0)) + qIn
```

and

```
gauss(dsqrt(Width0)) + pIn
```

generates the sampled points q_0 and p_0, respectively, from Eq. (52). Note that the normalization constant is either hard-coded into the `timecorr` files, or absent all together if it cancels with constants that appear in the correlation function. In the present case of the 1D Husimi-IVR it is absent altogether.

6.9 The supply files

The supply files contain formulas for the coherent state overlap, the coherent state matrix elements of the provided operators, all SC-IVR prefactors, a fourth-order symplectic integrator, and an MPI initialization subroutine.

6.10 The traj files

The traj files contain three subroutines that take in an initial phase space point and output arrays containing trajectories, classical action, and monodromy matrix elements at each time step. Most of the provided SC-IVR methods use the subroutine PropagateFwd_FF to accomplish this, the otherwo are specifically used for the FB-MQC-IVR. Each of the three subroutines are configured to signal a flag when energy conservation or symplecticity breaks to within a small tolerance that is specified in the input file (default 10^{-5}). A flagged trajectory is halted and removed from the statistics.

6.11 Other comments

When running execute.sh, certain combinations of system options in theory.in will signal a flag and suggest to you how to proceed.

7 Acknowledgements

We acknowledge Dr. Sergey Antipov for providing version 1 of the MQC-IVR code, Srinath Ranya for comprehensive testing of the current implementation, and Dr. Timothy J. H. Hele for provided version 1 of the MInt algorithm code. We further acknowledge ARO (Grant No. W911NFD-13-1-0102) for funding.