(∗ **Explanation** ∗)

(∗ To appreciate the physical meaning of a numerically computed Schrodinger propagator K, it is desirable to evaluate integrals of the form

$$\psi(t,x) = \int_{-\infty}^{\infty} dy\, K(t,x,y)\, f(y)$$

with a well localized initial wave packet f.

Typically, K does not decay as $|x-y|$ becomes large, but instead becomes rapidly oscillatory. Therefore, a naive numerical integration is unsatisfactory for two reasons:

1. The oscillations require a very small step size, making the numerical quadrature unacceptably slow.
2. More fundamentally, such a numerical approximation is inherently nonuniform in x. A standard numerical approximation of this integral (such as by Simpson's rule) is a linear combination of *finitely* many terms $K(x-y_j,t)$. From the known form of the free propagator we expect that in the neighborhood of a point x, each such term resembles an elementary trigonometric function of angular frequency $m|x-y_j|/\hbar t$. The sum therefore behaves like a periodic or almost periodic function; destructive interference is only temporary and gives way again to constructive interference as x increases. The computed function displays spurious echoes of the central peak periodically along the x axis. Only when *all* frequencies are included, in a true integral, do we attain the correct decay of $\psi(t,x)$ at all large x. Obviously, these numerical artifacts can be seriously misleading when we don't know the answer beforehand. ∗)

(∗ **Demonstration code in** *Mathematica* ∗)

(∗ *The propagator.* This function contains the essence of the one-dimensional free propagator, without extraneous complications. ∗)

```
kr[x_, y_] := Cos[(x - y)^2]
```

(∗ *Numerical integration.* This integrates `kr` against a step function (the characteristic function of the unit interval) by the trapezoidal rule and plots the result. ∗)

```
trap[n_, range_] := Plot[
     (kr[x,0] + kr[x,1])/(2n) + Sum[kr[x, j/n], {j, 1, n-1}]/n,
     {x, -range, range}, PlotRange -> {-1, 1}]
```

(∗ *Execution.* After loading **naivgaus.m** into *Mathematica* at the command line, the user should execute the command `trap[`*n*`,` *range*`]` for $n = 1, 2, 4, 8, \ldots$ and *range* $= 10$, 20, 40, . . . to observe the effect. ∗)

(* **Gaussian initial data** *)

(* To separate the Fourier echoes from the effects of the discontinuities in the initial data, let's consider a Gaussian initial packet that falls nearly to zero at the endpoints of the integration interval. *)

(* *Numerical integration.* We are now approximating

$$\int_{-\infty}^{\infty} dy\, \cos[(x-y)^2]\, e^{-c(y-\frac{1}{2})^2}$$

where c must be chosen sufficiently large that $e^{-c/4}$ is negligible. *)

```
gausstrap[c_, n_, range_] := Plot[
     Exp[-c/4]*(kr[x,0] + kr[x,1])/(2n) +
     Sum[Exp[-c*((j/n)-0.5)^2]*kr[x, j/n], {j, 1, n-1}]/n,
     {x, -range, range}, PlotRange -> {-1, 1}]
```

(* *Execution.* Try `gausstrap[`*c*`, `*n*`, `*range*`]` for c between 1 and 40 and the other parameters as before. *)

(* *The exact solution.* Our Gaussian integral can be evaluated as

$$\Re\left\{\left(\frac{\pi}{c^2+1}\right)^{\frac{1}{2}}(c+i)^{1/2}\exp\frac{(ic^2-c)(x-\frac{1}{2})^2}{c^2+1}\right\}.$$

Since we are dealing with a fairly large c, it is a good approximation to neglect 1 relative to c^2 and neglect i relative to c; in that approximation we have the more transparent expression

$$\left(\frac{\pi}{c}\right)^{\frac{1}{2}} e^{-(x-\frac{1}{2})^2/c}\cos[(x-\tfrac{1}{2})^2].$$

That is, for a sufficiently narrow initial packet the output is essentially the free propagator with a *broad* Gaussian envelope. The centroid is the same as that of the initial data (since the mean momentum was 0), but the spread is reminiscent of the initial *momentum* distribution, which overwhelms the initial position spread.

The plot of `gausstrap[16, 8, 10]` is nearly identical to that of `exact[16, 10]`. But increasing the range from 10 to 40 in `gausstrap` reveals that the echo (or aliasing) is still present at larger x. *)

```
exact[c_, range_] := Plot[Re[Sqrt[Pi*(c+I)/(c^2+1)] *
          Exp[(I*c^2 - c)*(x - 0.5)^2 / (c^2 + 1)] ] ,
     {x, -range, range}, PlotRange -> {-1,1}]
```

(* *Comparison.* Here we plot the difference between the exact solution and the trapezoidal approximation. *)

```
exactfn[c_, x_] := Re[Sqrt[Pi*(c+I)/(c^2+1)] *
     Exp[(I*c^2 - c)*(x - 0.5)^2 / (c^2 + 1)] ]
trapfn[c_, n_, x_] := Exp[-c/4]*(kr[x,0] + kr[x,1])/(2n) +
Sum[Exp[-c*((j/n)-0.5)^2]*kr[x, j/n], {j, 1, n-1}]/n

compare[c_, n_, range_, height_] := Plot[ exactfn[c,x] - trapfn[c,n,x],
     {x, -range, range}, PlotRange -> {-height, height}]
```

(* *end* *)

naivgaus

Demonstration of the propagator integration problem

Page

Apr 17, 2000 @10:38 by MathLily