Notes on Steering Algorithms

Wyatt Newman

February, 2015

1) Robot motion model:

Consider a simple vehicle that travels in the x-y plane, and which can be commanded with a speed, v, and an angular-rotation rate, ω. The robot has a 3-D pose that can be specified as its x and y coordinates plus its heading, ψ. The heading will be defined as measured CCW from the x-axis.

The motion of the robot is described by the following differential equations:

$$\frac{d}{dt}\begin{bmatrix} x \\ y \\ \psi \end{bmatrix} = \begin{bmatrix} v\cos(\psi) \\ v\sin(\psi) \\ \omega \end{bmatrix} .$$

It will be useful to define tracking-error coordinates d_{err}, a lateral offset error, and ψ_{err}, a heading error, relative to a desired path. An example desired path segment is a directed line segment starting from (x_0, y_0), and with tangent $\boldsymbol{t} = \begin{bmatrix} \cos(\psi_{des}) \\ \sin(\psi_{des}) \end{bmatrix}$ and with edge normal $\boldsymbol{n} = \begin{bmatrix} -\sin(\psi_{des}) \\ \cos(\psi_{des}) \end{bmatrix}$. The edge normal is a positive 90-deg rotation of $\boldsymbol{t}$, where positive rotation is defined as "up" (rotation about z, where the z-axis is defined normal to the x-y plane, consistent with forming a right-hand coordinate system x-y-z).

With respect to this directed line segment, we can compute a lateral offset error of the robot at position (x,y) as $d_{err} = \left(\begin{bmatrix} x_0 \\ y_0 \end{bmatrix} - \begin{bmatrix} x \\ y \end{bmatrix} \right) \cdot \boldsymbol{n}$. The heading error, ψ_{err}, is computed as the desired heading vs the actual heading: $\psi_{err} = \psi_{des} - \psi$. With rotational variables, it is important to consider periodicity. For example, consider a desired heading of $\psi_{des} = 0.0$ (i.e., pointing parallel to the x axis) and an actual heading of $\psi = 0.1$. This corresponds to a heading error of -0.1 rad, which is close to the desired heading. However, for a heading of $\psi = 6.0$, the formula $\psi_{err} = \psi_{des} - \psi$ would seem to imply a heading error of -6.0 rad—a large heading error. Instead, the periodic solutions of this result should be checked to find the smallest error interpretation. The value of $\psi_{err} + 2\pi = \psi_{des} - \psi + 2\pi$ corresponds to a positive heading error of approximately 0.28 rad. That is, the heading is more easily corrected by rotating by 0.28 rad (CW =positive) than by rotating -6 rad (negative, or CCW). This periodic condition must be checked within iterations of a control algorithm, else it can lead to gross instability. We will choose to define the heading error as the smallest-magnitude option among periodic alternatives.

We will define a 2-D path-following error vector in terms of these error components as:

$$\boldsymbol{e} = -\begin{bmatrix} d_{err} \\ \psi_{err} \end{bmatrix} .$$

In the following, we will assume that the robot is moving at a constant speed, v, but that we can command a rotation rate, ω, with the objective of driving both components of the error vector to zero. If the error vector is zero, then the robot will have zero offset (i.e. will be positioned on top of the path segment) and zero heading error (i.e. will be pointing consistent with the desired heading).

An objective of a steering algorithm is to compute an appropriate spin command, ω, to drive the error components to zero, and to do so with desirable dynamics (e.g., rapidly and stably).

2) Linear steering of linear robot:

We first consider a simplified system: a linear approximation of the robot dynamics controlled by a linear controller. For simplicity, assume that the desired path is the positive x axis, for which the error vector is simply $\boldsymbol{e}=-\begin{bmatrix} y \\ \psi \end{bmatrix}$ (assuming ψ is expressed as the smallest-magnitude option among periodic alternatives).

Assuming a linear controller, we may command $\omega=\boldsymbol{K}\cdot\boldsymbol{e}$, where $\boldsymbol{K}=[K_d, K_\psi]$. The two components of **K** are control gains, which should be chosen to result in desirable dynamics of the robot approaching the desired path.

Consider simplified (linearized) vehicle dynamics of:

$\frac{d}{dt}\begin{bmatrix} x \\ y \\ \psi \end{bmatrix}=\begin{bmatrix} v \\ v\psi \\ \omega \end{bmatrix}$, corresponding to a small-angle approximation for small values of ψ .

Equivalently, the error dynamics can be expressed as:

$$\frac{d}{dt}\boldsymbol{e}=\begin{bmatrix} v\psi \\ \omega \end{bmatrix}=\begin{bmatrix} 0 & v \\ 0 & 0 \end{bmatrix}\boldsymbol{e}+\begin{bmatrix} 0 \\ -1 \end{bmatrix}\omega$$

Substituting in our control algorithm yields:

$$\frac{d}{dt}\boldsymbol{e}=\begin{bmatrix} 0 & v \\ 0 & 0 \end{bmatrix}\boldsymbol{e}+\begin{bmatrix} 0 \\ -1 \end{bmatrix}\boldsymbol{K}\,\boldsymbol{e}=-\begin{bmatrix} 0 & -v \\ K_d & K_\psi \end{bmatrix}\boldsymbol{e}$$

In the LaPlace domain, this corresponds to:

$\begin{bmatrix} s & -v \\ K_d & s+K_\psi \end{bmatrix}\boldsymbol{e}=\begin{bmatrix} 0 \\ 0 \end{bmatrix}$ which has a characteristic equation of:

$$0=s^2+K_\psi s+v\,K_{psi}$$

Any choice of positive values for the control gains theoretically results in a stably-controlled system (for this linear approximation). We can choose gains intelligently by choosing values interpreted in terms of the generic second-order system response: $0=s^2+2\zeta\omega_n s+\omega_n^2$.

E.g., if we choose $\omega_n=6$ (roughly 2 pi, or about 1Hz) and $\zeta=1$ (i.e., critical damping), we would expect convergence to the desired path with a time constant of approximately 1 second and zero overshoot.

Figures 1, 2 and 3 below show a simulation of the linearized system with the chosen controller. The initial offset error and the heading error both converge to zero within about 1 second, as expected, and the offset error does not overshoot, consistent with a critically-damped system.

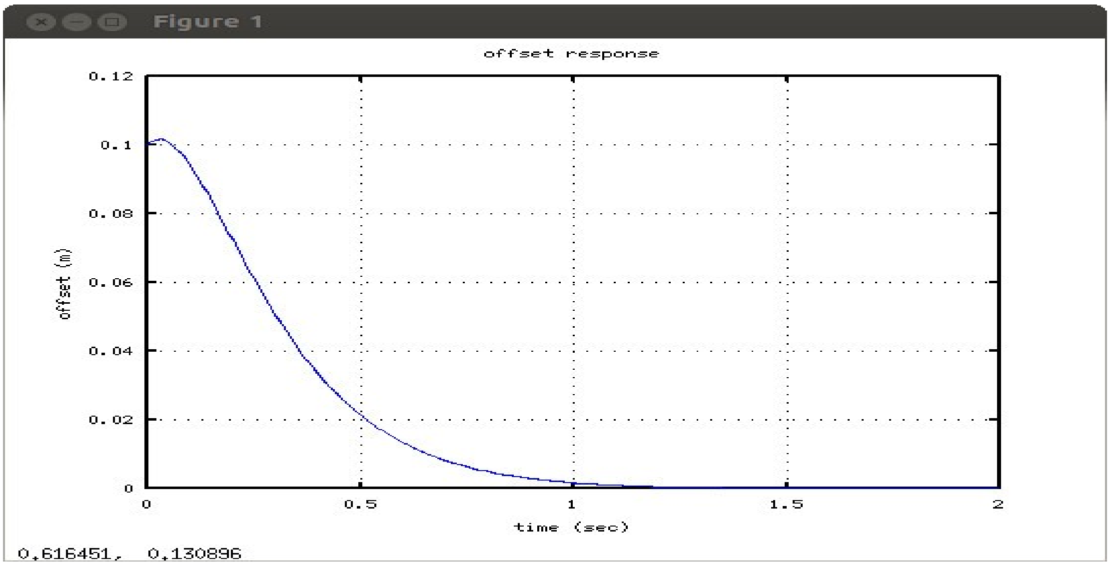

Illustration 1: offset response of linear system, 1Hz controller, critically damped

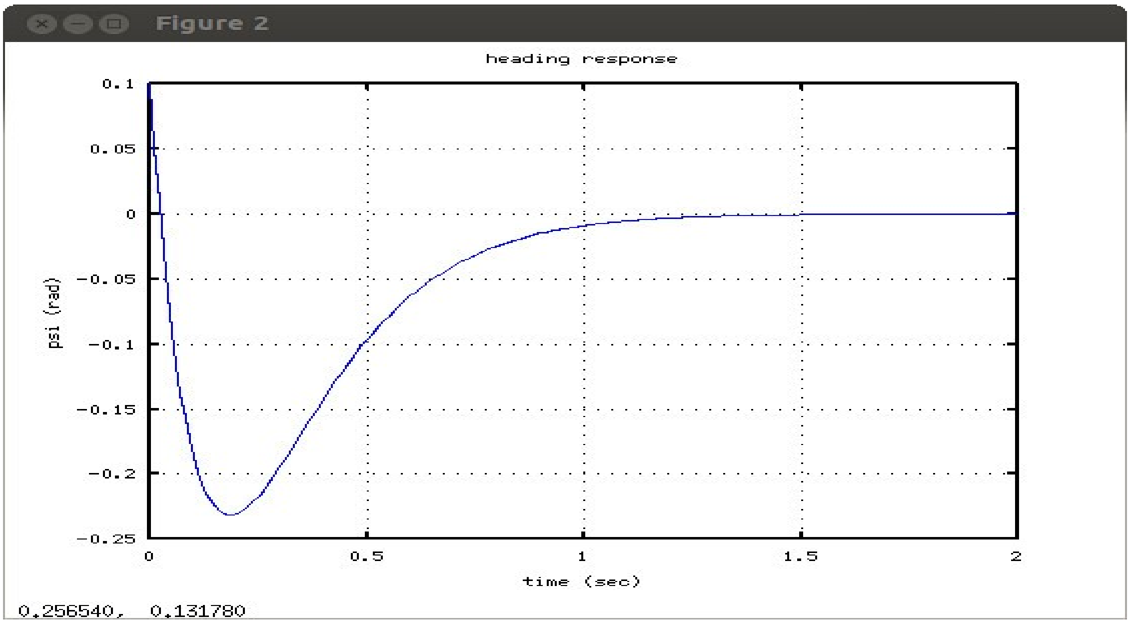

Illustration 2: Heading response of linear system

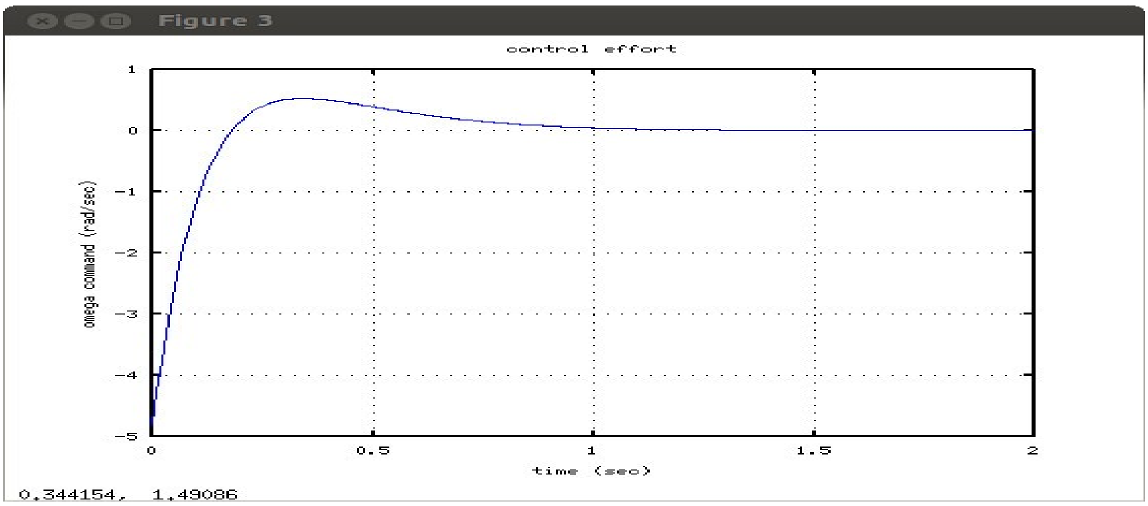

Illustration 3: Control effort history. Note that the effort command may exceed the capabilities of the physical system.

3) Linear steering of a nonlinear robot:
In the preceding section, we considered a linear controller acting on a hypothetical linear system (a small-angle approximation of our robot dynamics). We should evaluate the consequences of attempting use of a linear controller on a more realistic model of the robot.

With the same gains as in Section 2, but with the dynamic model of Section 1 (not linearized), we consider a directed line-segment desired path, specified as follows. The start of the line segment is at (x,y) = (1,0), and the slope of the desired path is 45-deg. Path heading error and offset error are computed as described in Section 1. The angular-rotation command is computed based on the linear control algorithm and control gains of Section 2.

The figures below show the response to relatively small initial errors, displayed as lateral-offset error (Fig 4), and path vs desired path in the x-y plane (Fig 5). The response is good, similar to that of the linear analysis.

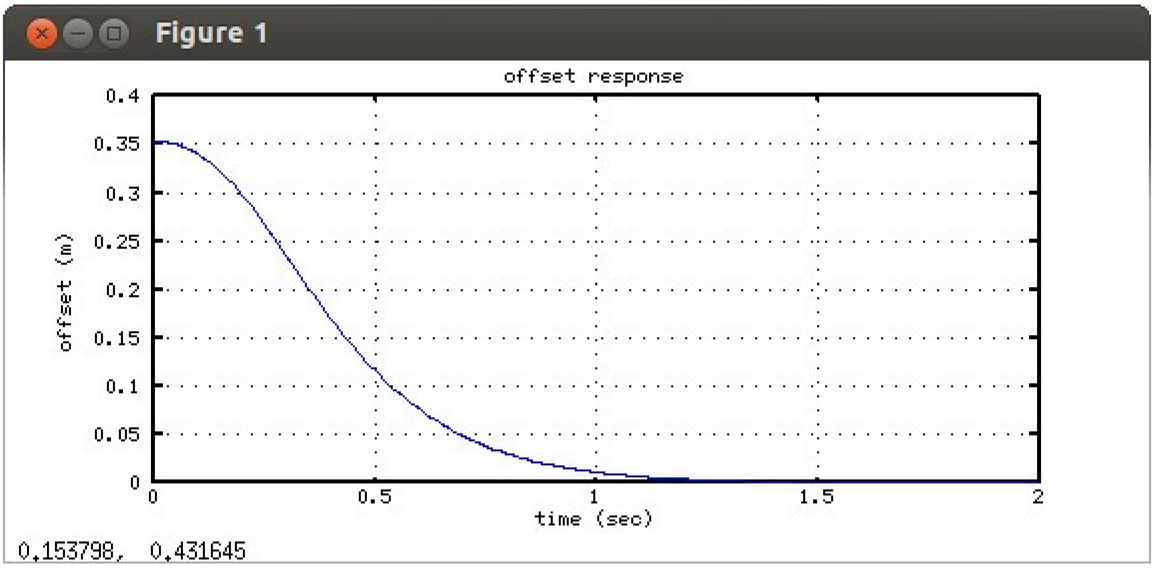

Illustration 4: Offset convergence of linear controller with nonlinear robot with relatively small initial error

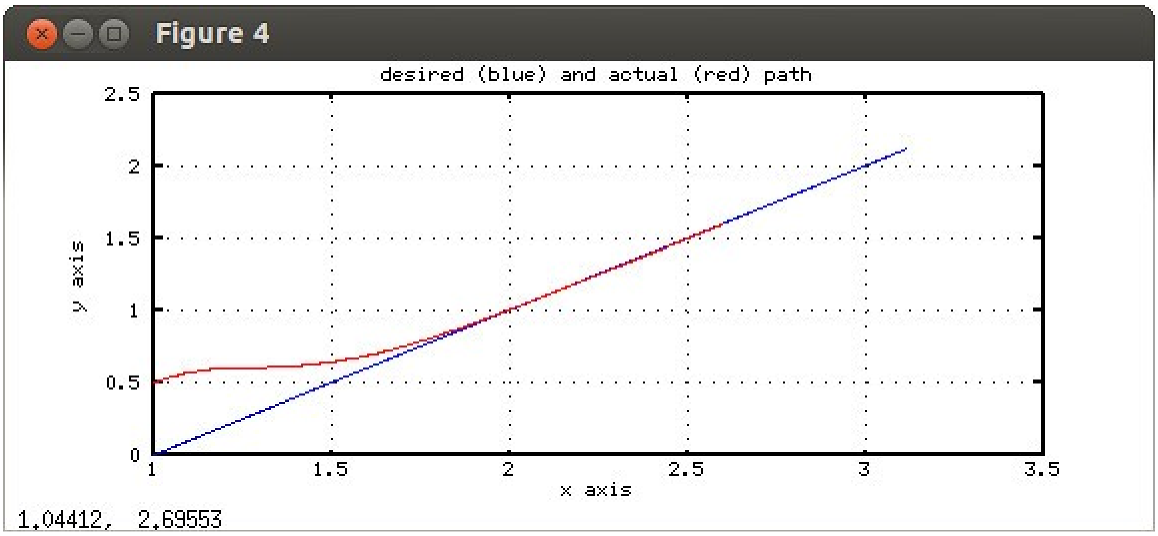

Illustration 5: Path convergence shown in x-y plane for linear controller, nonlinear robot, small initial displacement

When the initial conditions have larger error, however, the linear controller can behave very badly with the nonlinear robot dynamics. The figures below show an example with a larger initial offset error.

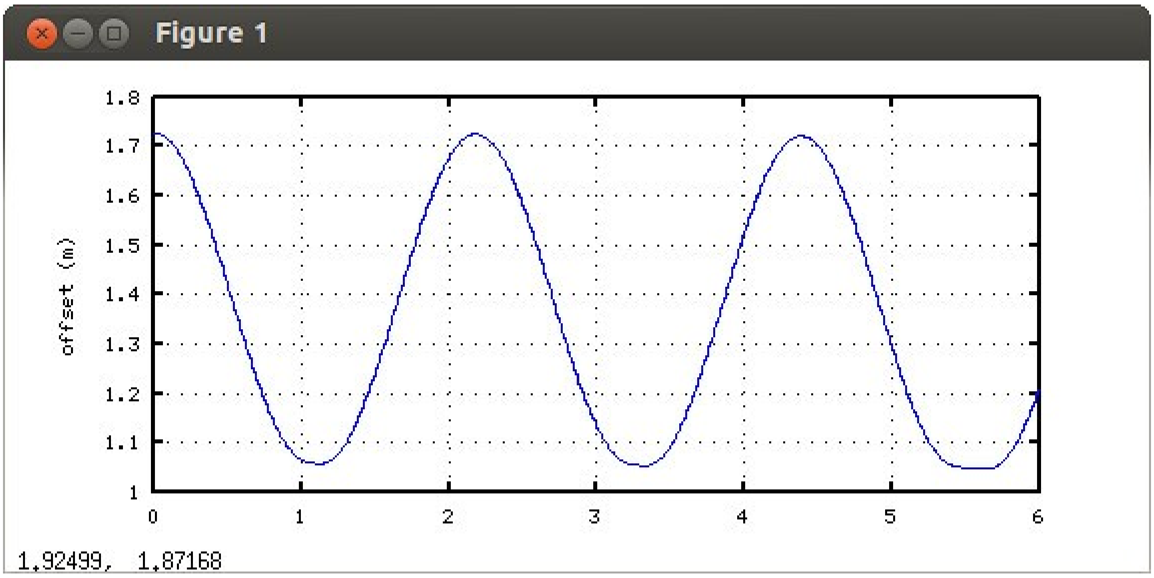

Illustration 6: Linear controller on nonlinear robot with larger initial displacement. Displacement error vs time oscillates.

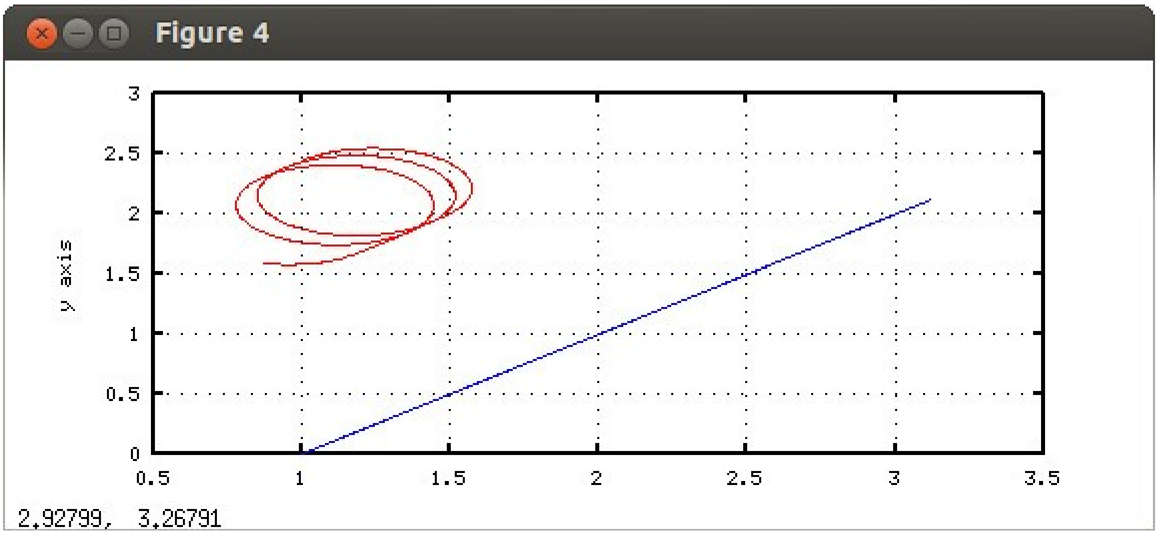

Illustration 7: Path from linear controller on nonlinear robot with large initial error. Robot spins in circles, failing to converge on desired path.

Unlike linear systems, the response of the nonlinear system depends on initial conditions. Tuning the controller for a desired response may give the impression that it is well behaved, but different initial conditions can result in wildly unstable response. In short, a linear controller cannot be trusted on the actual robot.

4) Nonlinear steering of a nonlinear robot:
In Sections 2 and 3, it was shown that a linear controller works well as a steering algorithm, provided the path-following errors are sufficiently small. The control algorithm computes a spin-rate command as a weighted sum of heading and displacement errors. However, when the robot is far from the goal (or has a large heading error), this linear mapping can behave badly.

When the robot is far from the desired path (or has a large heading error), the appropriate response is quite different from a linear mapping. If the lateral displacement is large, the robot should first attempt to minimize the displacement errors, and to do so, the best heading is to orient the robot towards the path. In this case, a heading error of $\pm\pi/2$ is desirable and should not be penalized. Instead, the ω command should drive the heading to point towards the desired path segment. However, when the displacement is sufficiently small, the heading should tend towards the path tangent (i.e. the heading error should tend towards zero).

Figure 8 shows the response of the linear controller on the nonlinear robot for a variety of initial conditions for which convergence is well behaved. In this view, the delta-heading is plotted as a function of displacement, which is a "phase space" plot. It can be observed that as the robot approaches convergence to the desired path segment, the delta-heading is approximately linearly proportional to the lateral-offset error. This observation can be used to help design a nonlinear controller that is tolerant of initial conditions with large errors.

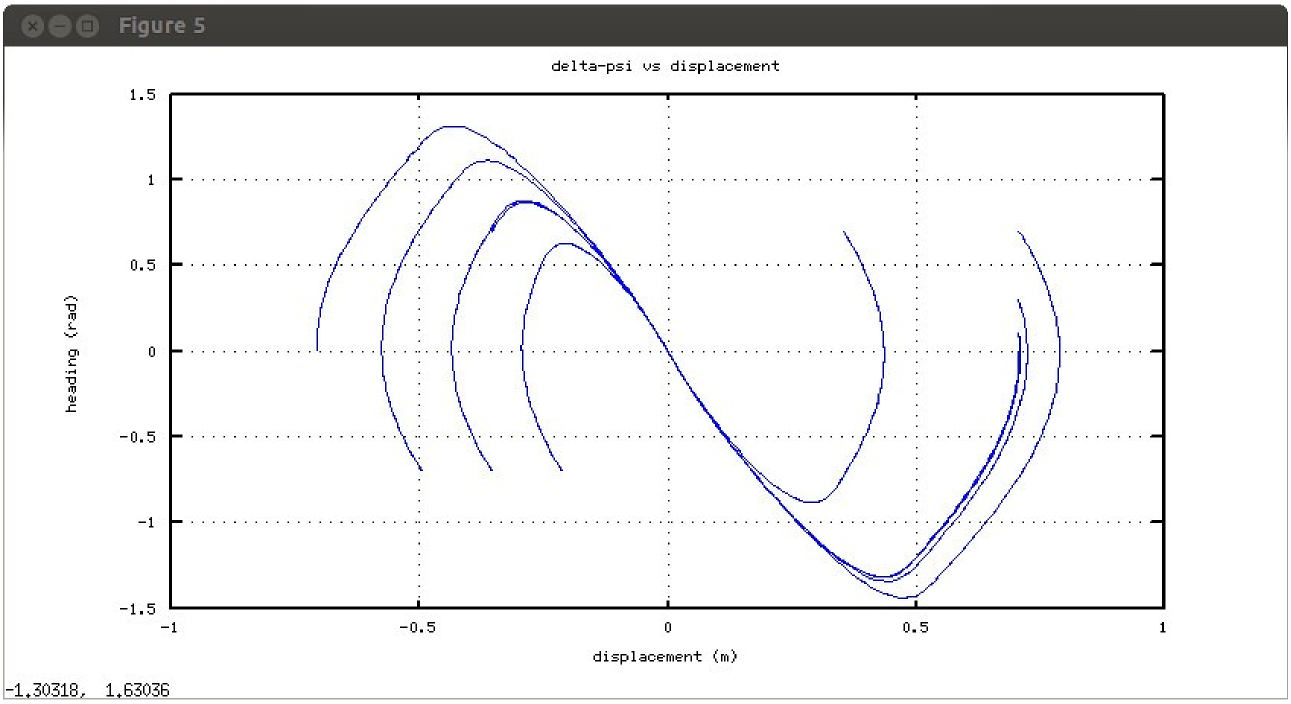

Illustration 8: phase space of linear control of nonlinear robot. Note linear relation between displacement and delta-heading near the convergence region.

The above observations can be combined in a rule that specifies the desired heading as a function of lateral offset, plus a 1st-order controller that commands angular velocity to reach the target heading. For a phase-space plan, we can try saturation function. For large values of offset, the desired delta heading is $\pm\pi/2$ (depending on the sign of the offset), and for small values of offset, we desire the delta heading to be proportional to the offset.

We may define $d_{offset} = \left(\begin{bmatrix} x \\ y \end{bmatrix} - \begin{bmatrix} x_0 \\ y_0 \end{bmatrix} \right) \cdot \boldsymbol{n}$, which is just the negative of d_{err} . Similarly, define

$\psi_{target} = \psi_{plan}(d_{offset}) + \psi_{path}$, where ψ_{target} is the desired robot heading according to a convergence plan (to be designed), and ψ_{path} is the heading of the current path segment. For our convergence plan, we desire $\psi_{plan}(d_{offset}) = -sgn(d_{offset})\, pi/2$ for large d_{offset} , and $\psi_{plan}(d_{offset}) = -K_\psi d_{offset}$ when d_{offset} is sufficiently small. To blend these regimes together, we can say: $\psi_{plan}(d_{offset}) = -\pi/2\, sat(d_{offset}/d_{thresh})$, where "sat(x)" is the saturation function, defined as: sat(x) = 1 for x>1, sat(x) = -1 for x<-1, and sat(x) = x for -1<x<1.

We will issue angular-velocity commands, ω , with a linear controller to attempt to achieve the target heading, as: $\omega = K_\psi(\psi_{target} - \psi)$. To incorporate a nonlinear (saturation) constraint on ω , recognizing that angular-velocity commands are limited to $\pm\omega_{max}$, the control law can be modified to: $\omega_{command} = \omega_{max}\, sat(K_\psi(\psi_{target} - \psi))$.

Combining the above, we derive an angular-velocity command, $\omega_{command}$ as a function of robot heading, ψ , path-segment heading, ψ_{path} , and lateral offset, d_{offset} , as follows:

$$\omega_{command} = \omega_{max}\, sat(K_\psi((\psi_{plan}(d_{offset}) + \psi_{path}) - \psi))$$

and thus:

$$\omega_{command} = \omega_{max}\, sat(K_\psi(-\pi/2\, sat(d_{offset}/d_{thresh}) + \psi_{path} - \psi))$$

The above control law can be tuned by first tuning a first-order linear controller of the form:

$$\omega = -K_\psi \psi \quad .$$

The value of K_ψ should be set experimentally to the largest practical value for good response to small heading errors. Given K_ψ , the value of d_{thresh} can be tuned experimentally, with smaller values resulting in faster approach, but sharper turns upon approach to the desired path segment. For small enough values of d_{offset}/d_{thresh} and $\psi_{path} - \psi$, neither saturation value will be limiting, and the control law will reduce to:

$$\omega_{command} = -(\omega_{max} K_\psi \pi/2/d_{thresh})\, d_{offset} + (\omega_{max} K_\psi)(\psi_{path} - \psi)$$

which is the same as the linear control law of Section 2 (which also provided guidance for choosing values for K_ψ and d_{thresh} .

At the other extreme, for large values of d_{offset} , the nonlinear control law simplifies to:

$$\omega_{command} = \omega_{max}\, sat(K_\psi(-\pi/2\, sgn(d_{offset}) + \psi_{path} - \psi))$$

which exerts control to point the robot's heading perpendicular to (and pointing towards) the path segment.

Figure 9 shows phase-space trajectories resulting from the above non-linear controller, initiated with a range of initial errors in offset and heading. The control parameters for these simulations were: $\omega_{max} = 2.0$, $d_{thresh} = 1.0$, $K_\psi = 30$.

All trajectories converge to zero offset error and zero heading error. For large offset errors, the approach initiates with heading offsets of $\pm\pi/2$. In regions near to convergence, attraction to the desired path segment are equivalent to those of linear controllers. Discontinuous jumps in delta-heading are periodic remappings between $\pm\pi$.

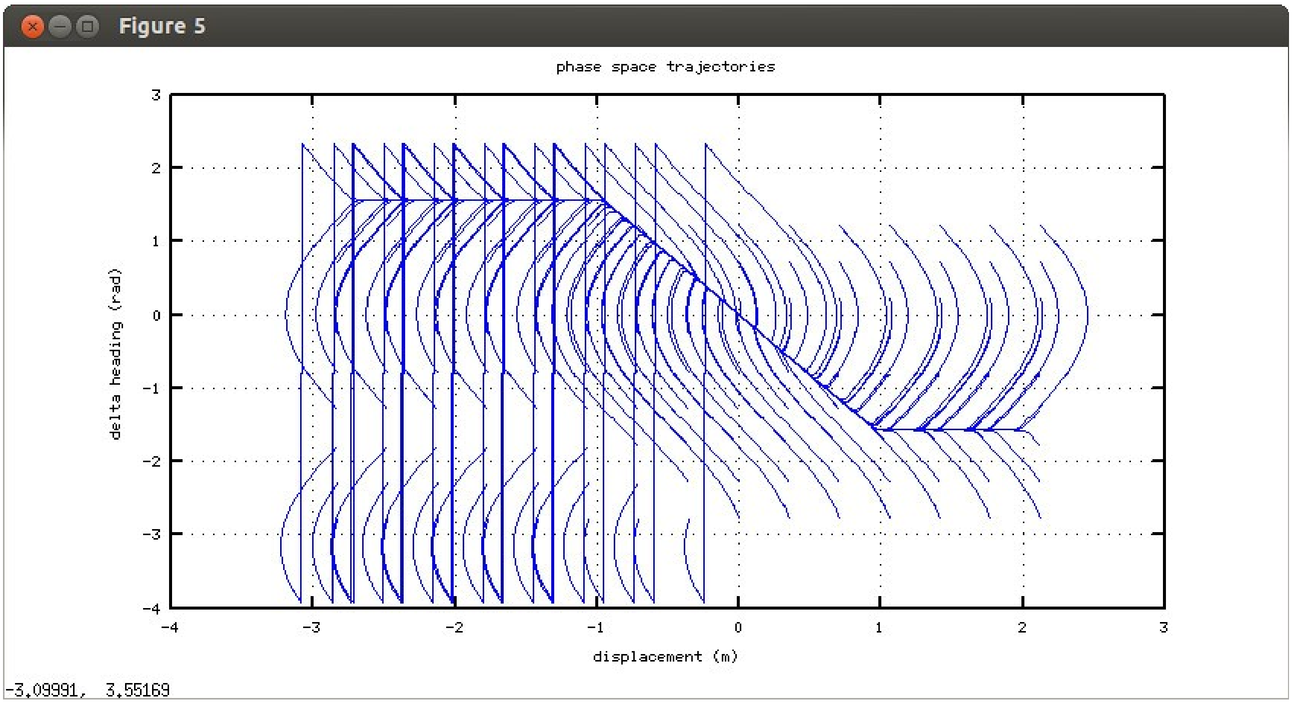

Illustration 9: Phase-space trajectories with nonlinear controller

An example path in the x-y plane is shown below.

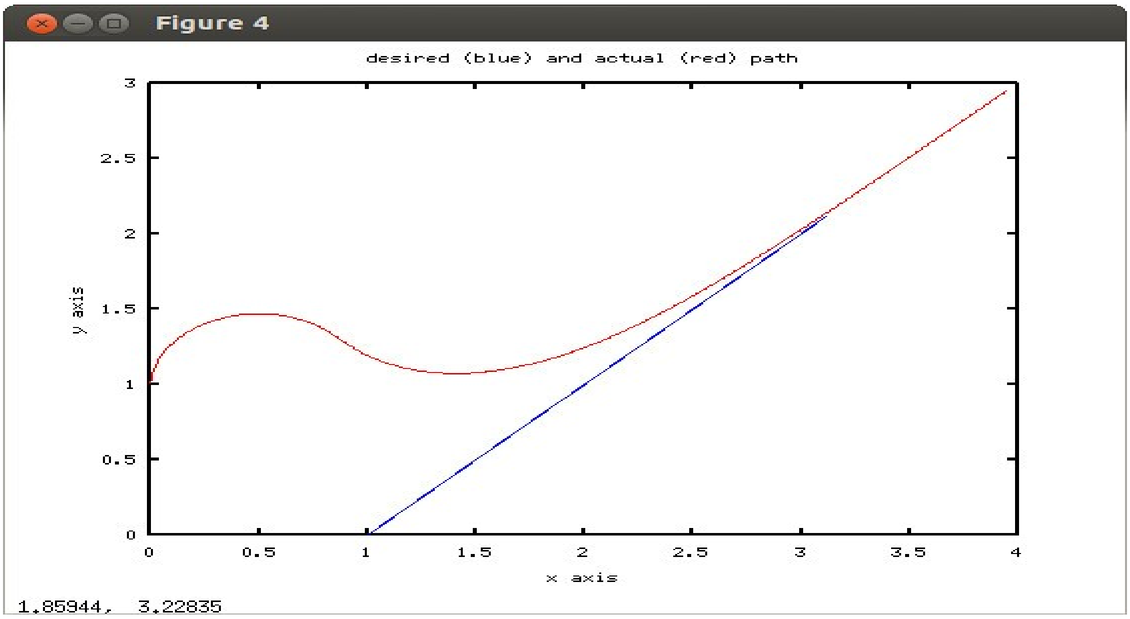

Illustration 10: Convergence with nonlinear controller and large initial condition errors

5) Conclusion and Extensions:

The preceding analysis demonstrated that a 2nd-order linear controller provides good steering performance when offset and heading errors are small. However, linear control breaks down when small-angle approximations are not valid. In the linear regime, it was recognized that a simple, proportional relationship in phase space holds between offset and delta-heading. Combining this with logic recommending relative heading of $\pm\pi/2$ for large offsets, a single, nonlinear but continuous control law is derived as a function of offset and relative heading. This nonlinear control law may be tuned similar to the 2nd-order linear controller, but it is well behaved regardless of initial conditions.

The present analysis has been limited to consideration of directed line-segment paths. More generally, a path may be expressed as a continuous sequence of path segments. In this context, the definitions of lateral offset and relative heading are ambiguous, since these depend on the path segment to be referenced. Further, with respect to a path segment that is curved, the offset and relative heading are not defined by the above. Additional definitions must be introduced to apply this controller to multi-segment paths.