ECEN 667
Power System Stability

Lecture 2: Numeric Solution of Differential Equations

Prof. Tom Overbye

Dept. of Electrical and Computer Engineering

Texas A&M University

overbye@tamu.edu

Announcements

- Start reading Chapters 1 and 2 from the book (Chapter 1 is Introduction, Chapter 2 is Electromagnetic Transients)
- EPG Dinner is on September 9 at 5pm; please RSVP using the link that was emailed to all.

Sources of Technical Information

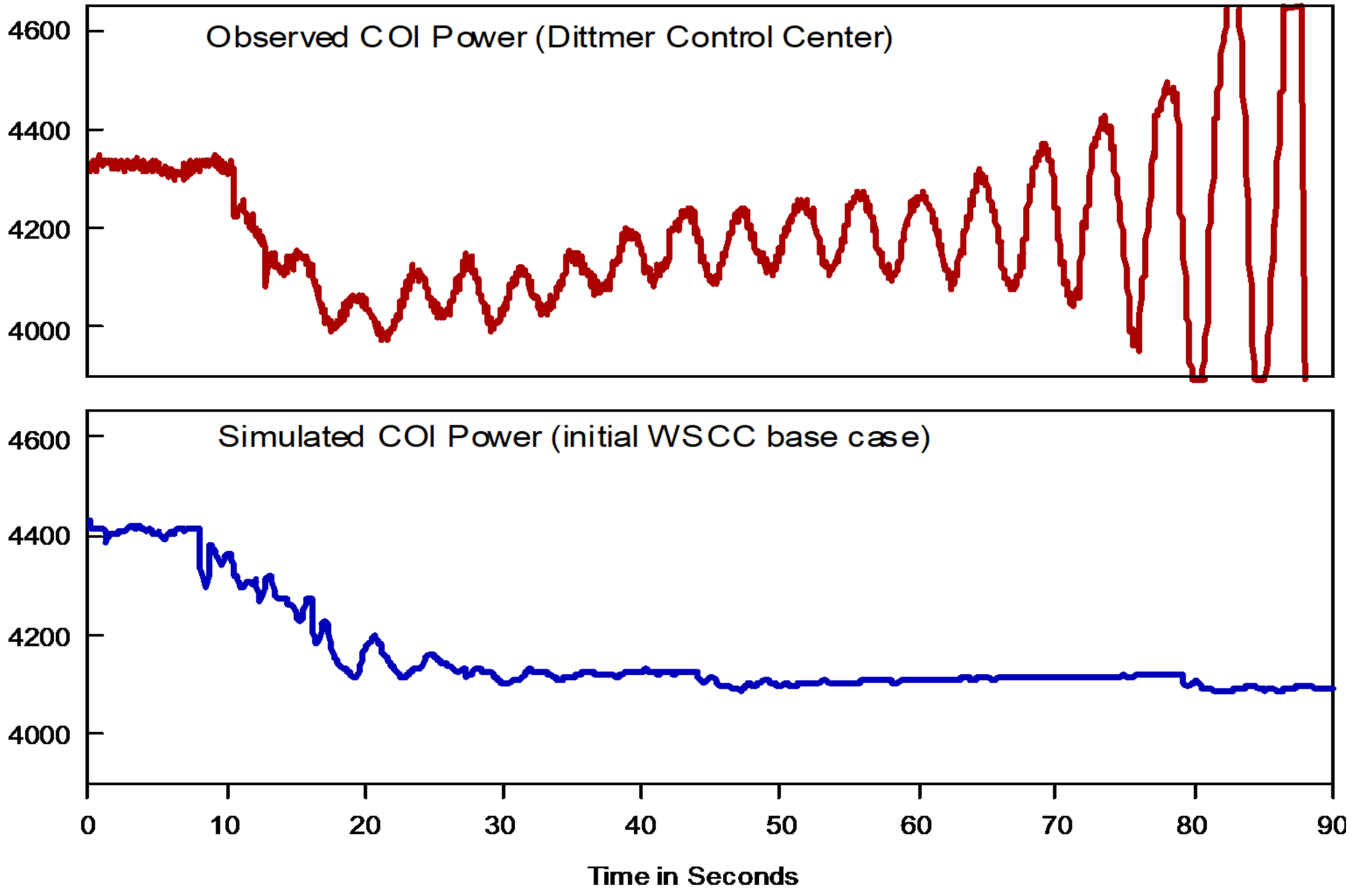
- There are many sources of technical information, with many students are most familiar with textbooks and courses
- Journals, conferences, reports, and presentations (e.g., webinars) are extremely useful additional sources
- In rough order of the newest of the information (from oldest to newest)
  - Books including textbooks; often lectures (depending on the instructor); reports
  - Journals, Conferences; both can have peer review
  - Webinars, and industry periodicals
- Many are available open access. However, the challenge is ultimately someone needs to cover the publication expenses.
- TAMU students can access IEEE Xplore at ieeexplore.ieee.org/Xplore/home.jsp

Slow versus Fast Dynamics

- Key analysis question in setting up and solving models is to determine the time frame of interest
- Values that change slowing (relative to the time frame of interest) can be assumed as constant
  - Power flow example is the load real and reactive values are assumed constant (sometimes voltage dependence is included)
- Values that change quickly (relative to the time frame of interest) can be assumed to be algebraic
  - A generator's terminal voltage in power flow is an algebraic constraint, but not in transient stability
  - In power flow and transient stability the network power balance equations are assumed algebraic

Dynamics Example 1

1996: Transient Stability Model Errors Lead to Blackouts



Dynamics Example: August 14 Blackout

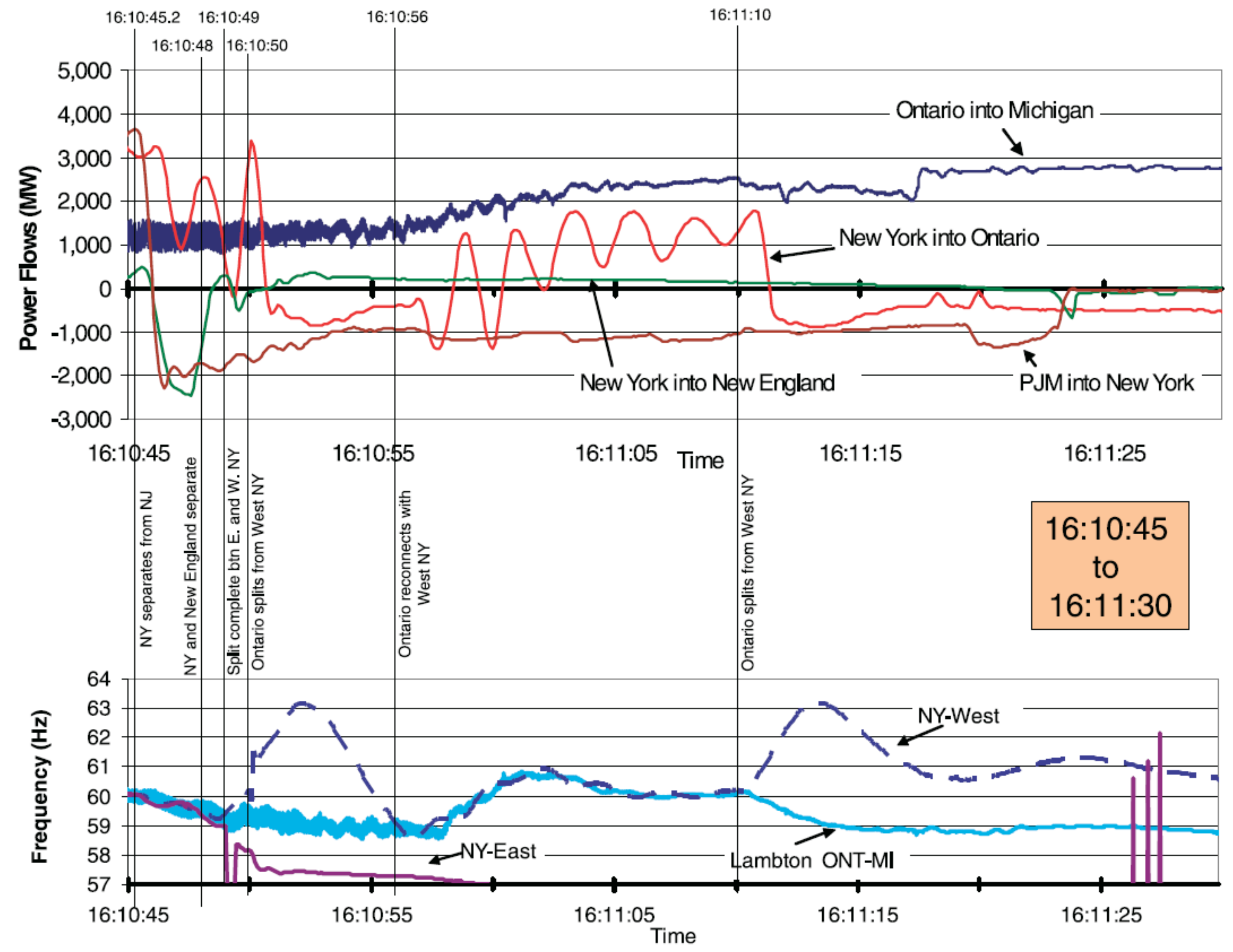
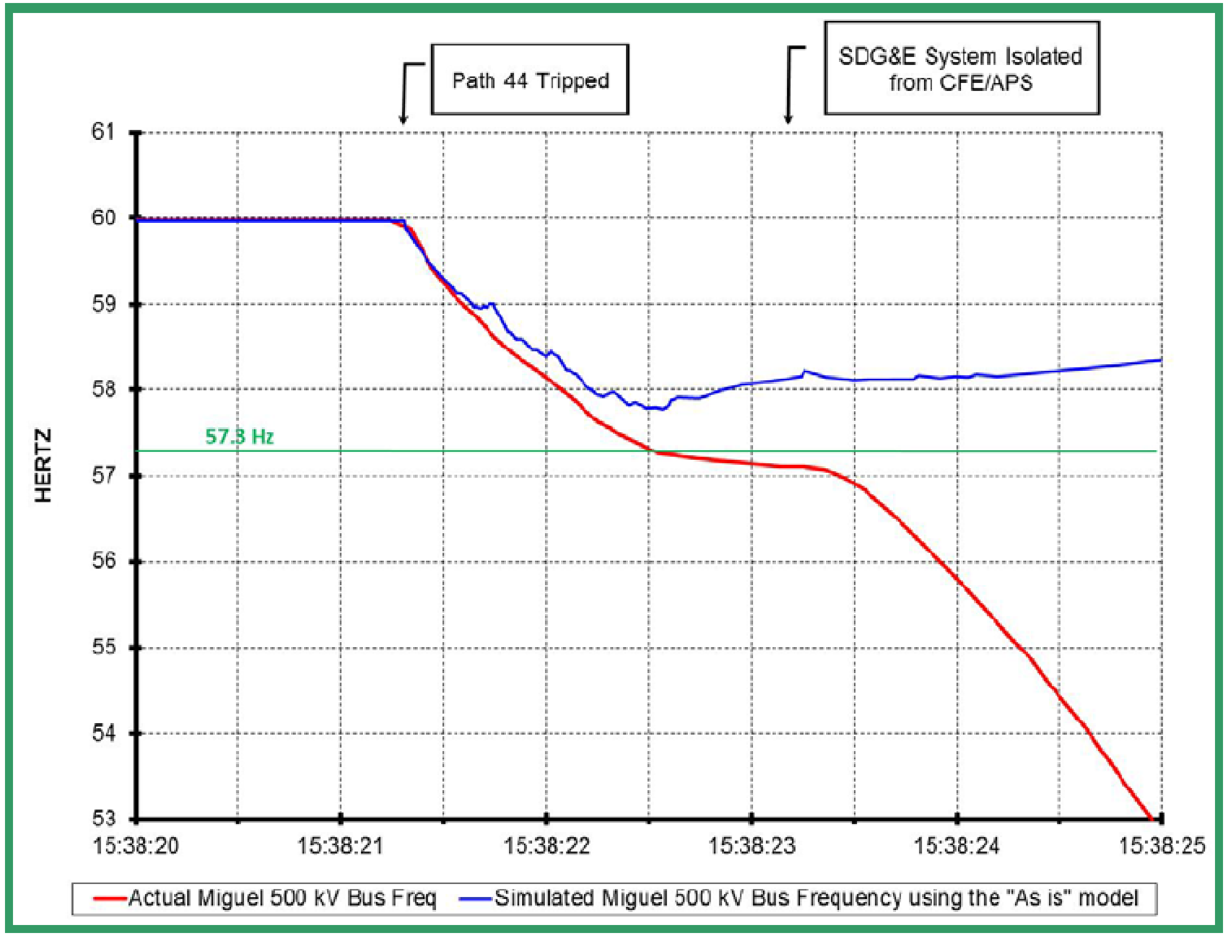


Image from August 14, 2003 Blackout Final Report, energy.gov, Figure 6.26

Dynamics Example 3

Figure 14: Actual and Simulated Frequency at Miguel 500 kV Bus



We've come a long ways since 1996 towards improved simulations. Still, a finding from the 2011 Blackout is the simulations didn't match the actual system response and need to be improved.

Source: *Arizona-Southern California Outages n September 8, 2011 Report*, FERC and NERC, April 2012

Models and Their Parameters

- Models and their parameters are often tightly coupled
  - The parameters for a particular model might have been derived from actual results on the object of interest
- Changing the model (even correcting an "incorrect" simulation implementation) can result in unexpected results!
- Using a more detailed simulation approach without changing the model can also result in incorrect results
  - More detailed models are not necessarily more accurate

Positive Sequence Versus Full Three-Phase

- Large-scale electrical systems are almost exclusively three-phase. Common analysis tools such as power flow and transient stability often assume balanced operation
  - This allows modeling of just the positive sequence though full three-phase models are sometimes used particularly for distribution systems
  - Course assumes knowledge of sequence analysis
- Other applications, such as electromagnetic transients (commonly known as electromagnetic transients programs [EMTP]) consider the full three-phase models

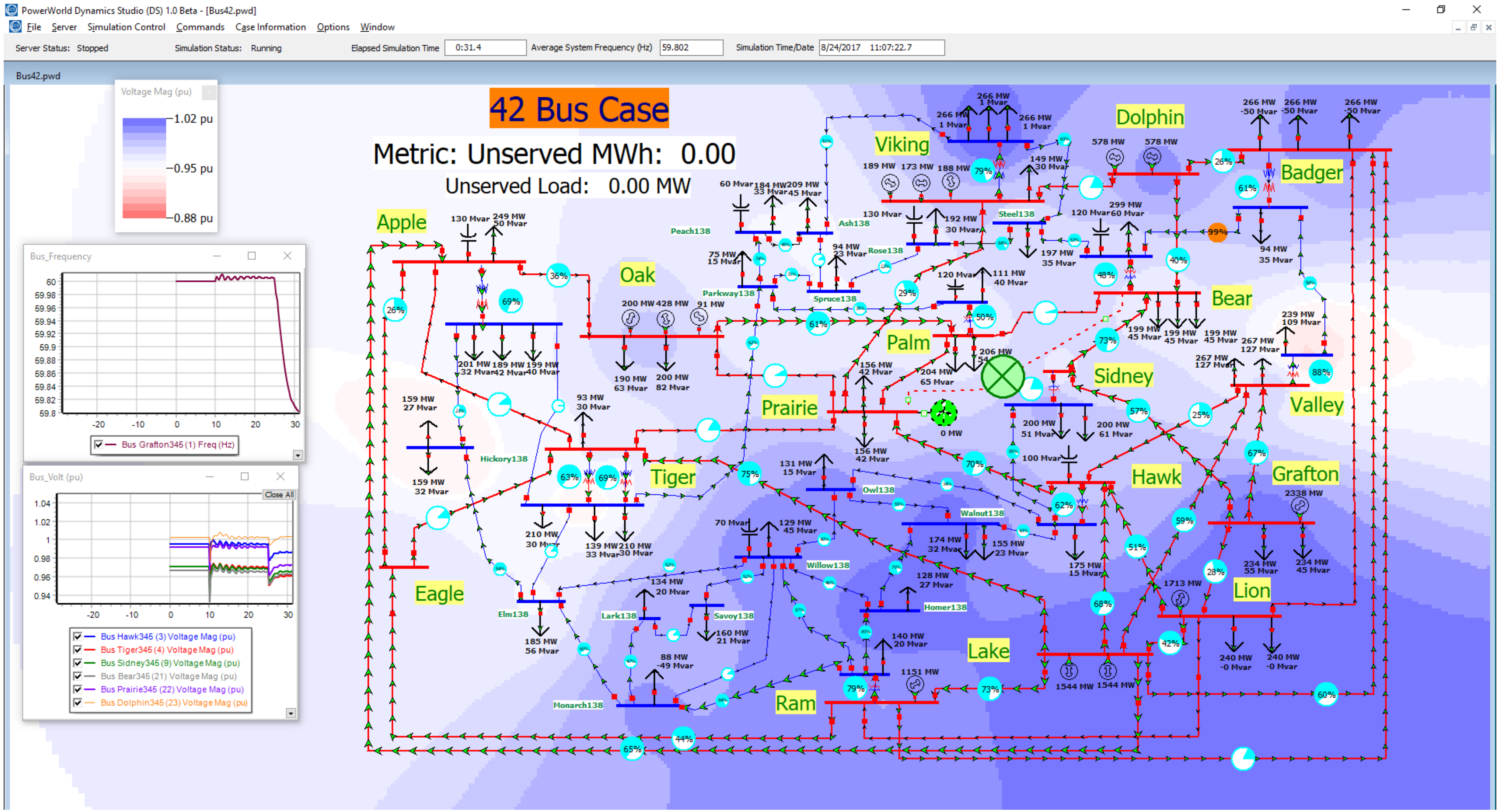
Power Flow Versus Dynamics

- The power flow is used to determine a quasi steady-state operating condition for a power system
  - Goal is to solve a set of algebraic equations $\mathbf{g(x) = 0}$
  - Models employed reflect the steady-state assumption, such as generator PV buses, constant power loads, LTC transformers
- Dynamic analysis is used to determine how the system changes with time, usually after some disturbance perturbs it away from a quasi-steady state equilibrium point

Example: Transient Stability

- Transient stability is used to determine whether following a disturbance (contingency) the power system returns to a steady-state operating point
  - Goal is to solve a set of differential and algebraic equations, $d\mathbf{x}/dt = \mathbf{f}(\mathbf{x},\mathbf{y})$, $\mathbf{g}(\mathbf{x},\mathbf{y}) = \mathbf{0}$
  - Starts in steady-state, and hopefully returns to steady-state.
  - Models reflect the transient stability time frame (up to dozens of seconds), with some values assumed to be slow enough to hold constant (LTC tap changing), while others are still fast enough to treat as algebraic (synchronous machine stator dynamics, voltage source converter dynamics).

Interactive Simulation: PowerWorld Dynamics Studio (DS)



Power System Stability Terms

- Terms continue to evolve, but a good reference is [1]; image shows Figure 4 from this reference

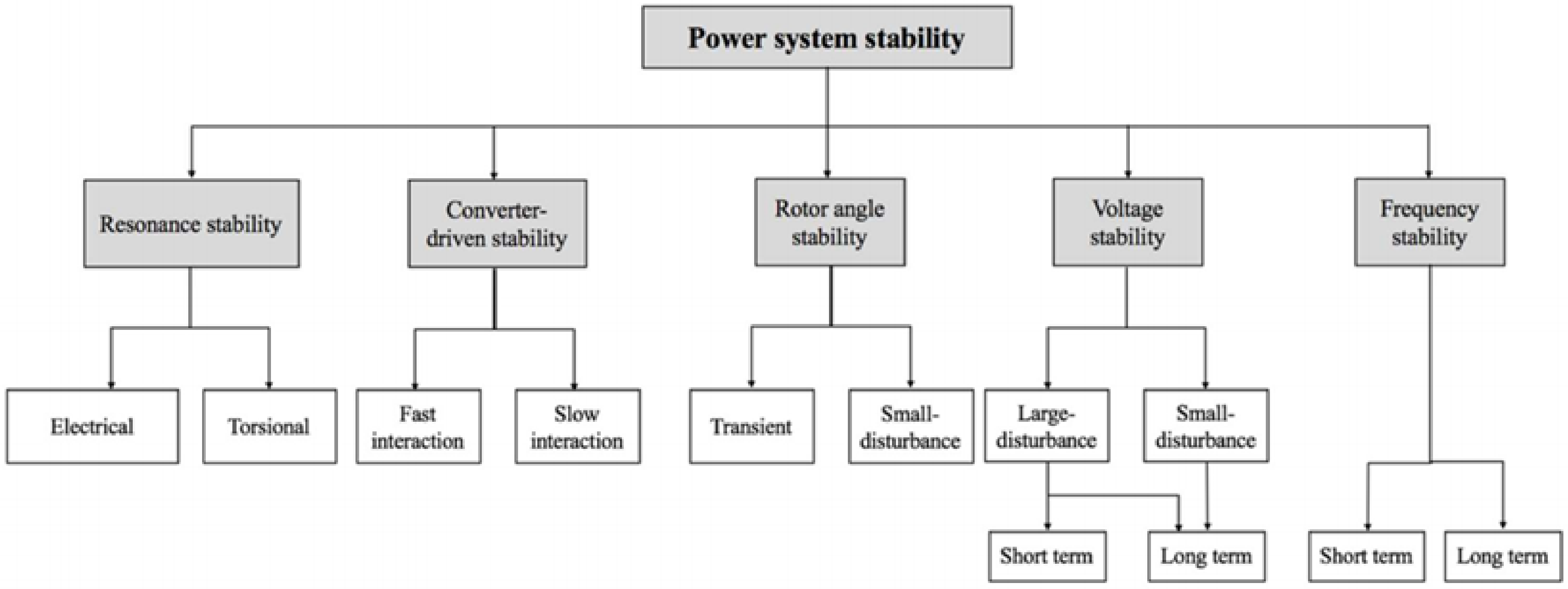
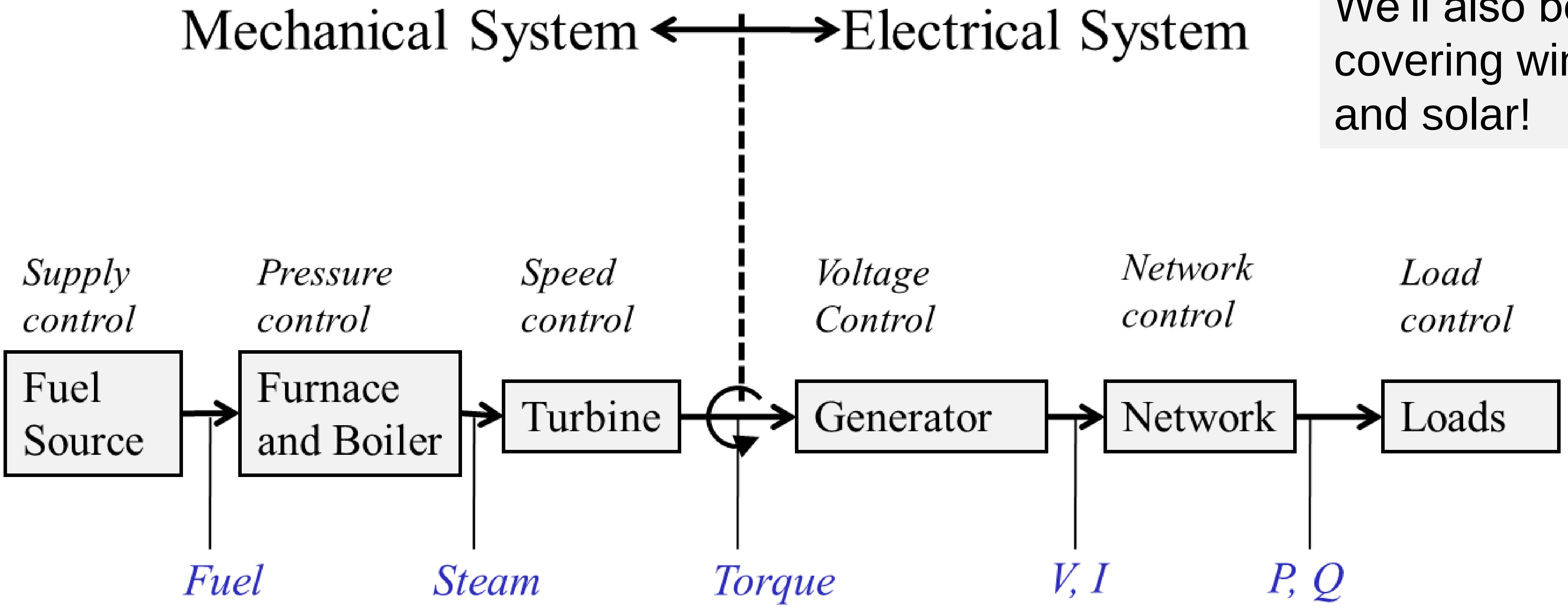


Fig. 4. Classification of power system stability

[1] IEEE/PES Power System Dynamic Performance Committee, "Stability definitions and characterization of dynamic behavior in systems with high penetration of power electronic interfaced technologies", PES-TR77, April 2020

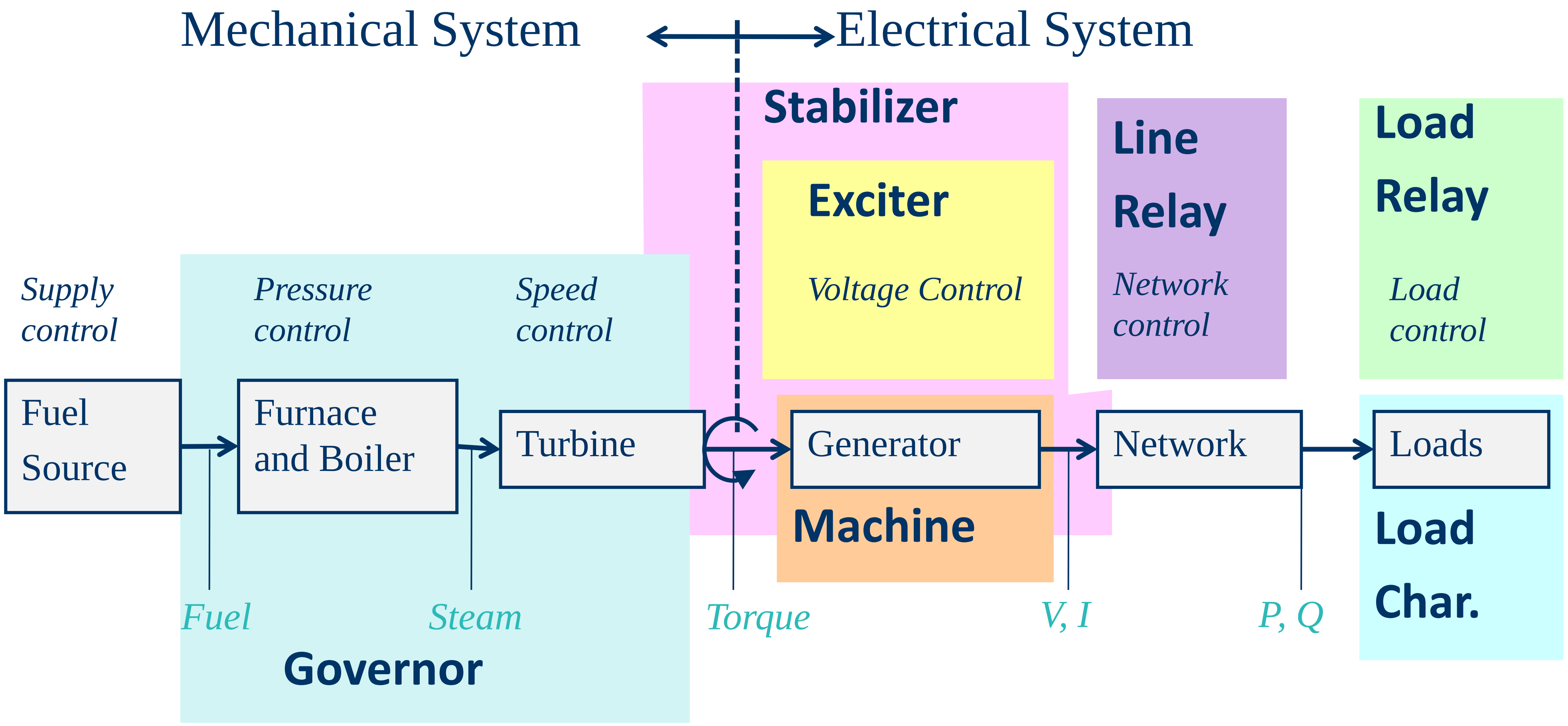
Physical Structure Power System Components

We'll also be covering wind and solar!



P. Sauer and M. Pai, *Power System Dynamics and Stability*

Physical Structure Power System Components



Differential Algebraic Equations

- Many problems, including many in the power area, can be formulated as a set of differential, algebraic equations (DAE) of the form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{y})$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{y})$$

- A power example is transient stability, in which **f** represents (primarily) the generator dynamics, and **g** (primarily) the bus power balance equations
- We'll initially consider the simpler problem of just

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$$

Ordinary Differential Equations (ODEs)

- Assume we have a problem of the form

  $$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) \quad \text{with } \mathbf{x}(t_0) = \mathbf{x}_0$$

- This is known as an initial value problem, since the initial value of **x** is given at some time t_0
  - We need to determine **x**(t) for future time
  - Initial value, $\mathbf{x}_0$, must be either be given or determined by solving for an equilibrium point, **f**(**x**) = **0**
  - Higher-order systems can be put into this first order form
- Except for special cases, such as linear systems, an analytic solution is usually not possible – numerical methods must be used

Equilibrium Points

- An equilibrium point $\mathbf{x}^*$ satisfies

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}^*) = \mathbf{0}$$

- An equilibrium point is stable if the response to a small disturbance remains small
  - This is known as Lyapunov stability
  - Formally, if for every $\varepsilon > 0$, there exists a $\delta = \delta(\varepsilon) > 0$ such that if $\|\mathbf{x}(0) - \mathbf{x}^*\| < \delta$, then $\|\mathbf{x}(t) - \mathbf{x}^*\| < \varepsilon$ for $t \geq 0$
- An equilibrium point has asymptotic stability if there exists a $\delta > 0$ such that if $\|\mathbf{x}(0) - \mathbf{x}^*\| < \delta$, then

$$\lim_{t \to \infty} \left\| \mathbf{x}(t) - \mathbf{x}^* \right\| = \mathbf{0}$$

Power System Application

- A typical power system application is to assume the power flow solution represents an equilibrium point
- Back solve to determine the initial state variables, **x**(0)
- At some point a contingency occurs, perturbing the state away from the equilibrium point
- Time domain simulation is used to determine whether the system returns to the equilibrium point

Initial value Problem Examples

Example 1: Exponential Decay

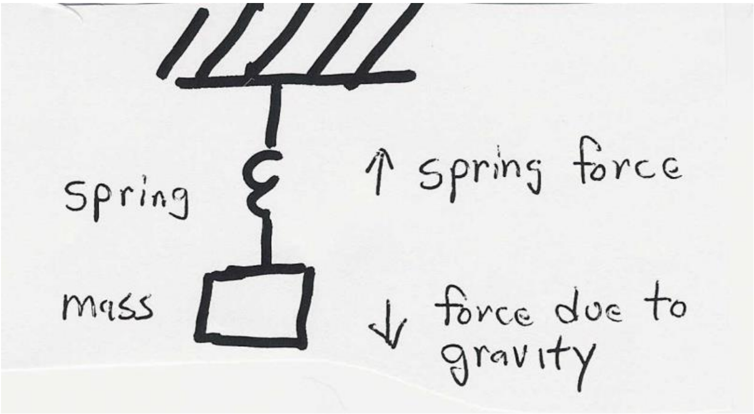
A simple example with an analytic solution is

$$\dot{x} = -x \qquad \text{with } x(0) = x_0$$

This has a solution $x(t) = x_0 e^{-t}$

Example 2: Mass-Spring System

$$kx - gM = M\ddot{x} + D\dot{x}$$



or

$$\dot{x}_1 = x_2$$

$$\dot{x}_2 = \frac{1}{M}\left(k x_1 - g M - D x_2\right)$$

Numerical Solution Methods

- Numerical solution methods do not generate exact solutions; they practically always introduce some error
  - Methods assume time advances in discrete increments, called a stepsize (or time step), Δt
  - Speed accuracy tradeoff: a smaller Δt usually gives a better solution, but it takes longer to compute
  - Numeric roundoff error due to finite computer word size
- Key issue is the derivative of **x, f(x)** depends on **x**, the value we are trying to determine
- A solution exists as long as **f**(**x**) is continuously differentiable

Numerical Solution Methods

- There are a wide variety of different solution approaches, we will only touch on several
- One-step methods: require information about solution just at one point, **x**(t)
  - Forward Euler
  - Runge-Kutta
- Multi-step methods: make use of information at more than one point, **x**(t), **x**(t-Δt), **x**(t-Δ2t)…
  - Adams-Bashforth
- Predictor-Corrector Methods: implicit
  - Backward Euler

Error Propagation

- At each time step the total round-off error is the sum of the local round-off at time and the propagated error from steps $1, 2, \ldots, k-1$
- An algorithm with the desirable property that local round-off error decays with increasing number of steps is said to be numerically stable
- Otherwise, the algorithm is numerically unstable
- Numerically unstable algorithms can nevertheless give quite good performance if appropriate time steps are used
  - This is particularly true when coupled with algebraic equations

Forward Euler's Method

- The simplest technique for numerically integrating such equations is known as the Euler's Method (sometimes the Forward Euler's Method)
- Key idea is to approximate

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}(t)) = \frac{d\mathbf{x}}{dt} \text{ as } \frac{\Delta\mathbf{x}}{\Delta t}$$

Then

$$\mathbf{x}(t+\Delta t) \approx \mathbf{x}(t) + \Delta t\, \mathbf{f}(\mathbf{x}(t))$$

- In general, the smaller the Δt, the more accurate the solution, but it also takes more time steps

Euler's Method Algorithm

Set t = t_0 (usually 0)

$\mathbf{x}(t_0) = \mathbf{x}_0$

Pick the time step Δt, which is problem specific

While $t \leq t^{end}$ Do

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \Delta t\, \mathbf{f}(\mathbf{x}(t))$$

$$t = t + \Delta t$$

End While

Euler's Method Example 1

Consider the Exponential Decay Example

$$\dot{x} = -x \qquad \text{with } x(0) = x_0$$

This has a solution $x(t) = x_0 e^{-t}$

Since we know the solution we can compare the accuracy of Euler's method for different time steps

t	$x^{actual}(t)$	x(t) Δt=0.1	x(t) Δt=0.05
0	10	10	10
0.1	9.048	9	9.02
0.2	8.187	8.10	8.15
0.3	7.408	7.29	7.35
…	…	…	…
1.0	3.678	3.49	3.58
…	…	…	…
2.0	1.353	1.22	1.29

Euler's Method Example 2

Consider the equations describing the horizontal position of a cart attached to a lossless spring:

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -x_1$$

Assuming initial conditions of $x_1(0) = 1$ and $x_2(0) = 0$, the analytic solution is $x_1(t) = \cos t$.

We can again compare the results of the analytic and numerical solutions

Euler's Method Example 2, cont'd

Starting from the initial conditions at t =0 we next calculate the value of x(t) at time t = 0.25.

$$
\begin{aligned}
x_1(0.25) &= x_1(0) + 0.25\,x_2(0) &&= 1.0 \\
x_2(0.25) &= x_2(0) - 0.25\,x_1(0) &&= -0.25
\end{aligned}
$$

Then we continue on to the next time step, t = 0.50

$$
\begin{aligned}
x_1(0.50) &= x_1(0.25) + 0.25\,x_2(0.25) &&= \\
&= 1.0 + 0.25 \times (-0.25) &&= 0.9375 \\
x_2(0.50) &= x_2(0.25) - 0.25\,x_1(0.25) &&= \\
&= -0.25 - 0.25 \times (1.0) &&= -0.50
\end{aligned}
$$

Euler's Method Example 2, cont'd

t	$x_1^{actual}(t)$	$x_1(t)$ $\Delta t=0.25$
0	1	1
0.25	0.9689	1
0.50	0.8776	0.9375
0.75	0.7317	0.8125
1.00	0.5403	0.6289
…	…	…
10.0	-0.8391	-3.129
100.0	0.8623	-151,983

Since we know from the exact solution that x_1 is bounded between -1 and 1, clearly the method is numerically unstable

Euler's Method Example 2, cont'd

Below is a comparison of the solution values for $x_1(t)$ at time t = 10 seconds

Δt	$x_1(10)$
actual	-0.8391
0.25	-3.129
0.10	-1.4088
0.01	-0.8823
0.001	-0.8423

Second Order Runge-Kutta Method

- Runge-Kutta methods improve on Euler's method by evaluating **f**(**x**) at selected points over the time step
- Simplest method is the second order method in which

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \frac{1}{2}(\mathbf{k}_1 + \mathbf{k}_2)$$

where

$$\mathbf{k}_1 = \Delta t\ \mathbf{f}(\mathbf{x}(t))$$

$$\mathbf{k}_2 = \Delta t\ \mathbf{f}(\mathbf{x}(t) + \mathbf{k}_1)$$

- That is, $\mathbf{k}_1$ is what we get from Euler's; $\mathbf{k}_2$ improves on this by reevaluating at the estimated end of the time step

Second Order Runge-Kutta Algorithm

$t = 0$, $x(0) = x_0$, Δt = step size

While $t \leq t^{final}$ Do

$k1 = \Delta t\ f(x(t))$

$k2 = \Delta t\ f(x(t) + k1)$

$x(t+\Delta t) = x(t) + (k1 + k2)/2$

$t = t + \Delta t$

End While

RK2 Oscillating Cart

- Consider the same example from before the position of a cart attached to a lossless spring. Again, with initial conditions of $x_1(0) = 1$ and $x_2(0) = 0$, the analytic solution is $x_1(t) = \cos(t)$

$$\dot{x}_1 = x_2$$
$$\dot{x}_2 = -x_1$$

- With $\Delta t = 0.25$ at $t = 0$

$$\mathbf{k}_1 = (0.25)\times\begin{bmatrix}0\\-1\end{bmatrix} = \begin{bmatrix}0\\-0.25\end{bmatrix}, \quad \mathbf{x}(0)+\mathbf{k}1 = \begin{bmatrix}1\\0\end{bmatrix} + \begin{bmatrix}0\\-0.25\end{bmatrix} = \begin{bmatrix}1\\-0.25\end{bmatrix}$$

$$\mathbf{k}_2 = (0.25)\times\mathbf{f}\left(\mathbf{x}(0)+\mathbf{k}_1\right) = \begin{bmatrix}-0.0625\\-0.25\end{bmatrix}, \quad \mathbf{x}(0.25) = \begin{bmatrix}1\\0\end{bmatrix} + \frac{1}{2}\left(\mathbf{k}_1+\mathbf{k}_2\right) = \begin{bmatrix}0.96875\\-0.25\end{bmatrix}$$

Comparison

- The below table compares the numeric and exact solutions for $x_1(t)$ using the RK2 algorithm

time	actual $x_1(t)$	$x_1(t)$ with RK2 $\Delta t=0.25$
0	1	1
0.25	0.9689	0.969
0.50	0.8776	0.876
0.75	0.7317	0.728
1.00	0.5403	0.533
10.0	-0.8391	-0.795
100.0	0.8623	1.072

Comparison of $x_1(10)$ for varying Δt

- The below table compares the $x_1(10)$ values for different values of Δt; recall with Euler's with Δt=0.1 was -1.41 and with 0.01 was -0.8823

Δt	$x_1(10)$
actual	-0.8391
0.25	-0.7946
0.10	-0.8310
0.01	-0.8390
0.001	-0.8391

RK2 Versus Euler's

- RK2 requires twice the function evaluations per iteration, but gives much better results
- With RK2 the error tends to vary with the cube of the step size, compared with the square of the step size for Euler's
- The smaller error allows for larger step sizes compared to Euler's

Fourth Order Runge-Kutta

- Other Runge-Kutta algorithms are possible, including the fourth order

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \frac{1}{6}\left(\mathbf{k}_1 + 2\mathbf{k}_2 + 2\mathbf{k}_3 + \mathbf{k}_4\right)$$

where

$$\mathbf{k}_1 = \Delta t\ \mathbf{f}\left(\mathbf{x}(t)\right)$$

$$\mathbf{k}_2 = \Delta t\ \mathbf{f}\left(\mathbf{x}(t) + \frac{1}{2}\mathbf{k}_1\right)$$

$$\mathbf{k}_3 = \Delta t\ \mathbf{f}\left(\mathbf{x}(t) + \frac{1}{2}\mathbf{k}_2\right)$$

$$\mathbf{k}_4 = \Delta t\ \mathbf{f}\left(\mathbf{x}(t) + \mathbf{k}_2\right)$$

RK4 Oscillating Cart Example

- RK4 gives much better results, with error varying with the time step to the fifth power

time	actual $x_1(t)$	$x_1(t)$ with RK4 $\Delta t=0.25$
0	1	1
0.25	0.9689	0.9689
0.50	0.8776	0.8776
0.75	0.7317	0.7317
1.00	0.5403	0.5403
10.0	-0.8391	-0.8392
100.0	0.8623	0.8601