Controls basics

Laplace(time to freq, d.e. to algebra, conv to mult; $L[f'(t)] = sF(s) - f(0)$; $L[\int_0^t f(t)dt] = F(s)/s$; $L[f''(t)] = s^2F(s) - sf(0) - f'(0)$ **Fourier**(laplace but $s = jw$) **First order**(i.e. $y(t) = \frac{2}{5} + \frac{3}{5}e^{-5t}$: transient/natural response= $\frac{3}{5}e^{-5t}$, SS response= $\frac{2}{5}$) **Poles**(roots of denom (x); pole at origin = step function in output; pole at $-a$ means transient response e^{-5t}; pole further to left = faster transient response decays to zero) **Zeros**(roots of num (o); zeros and poles generate amplitude for both forced and natural resp)**Char eqn**(setting denom of CLTF to 0)

System response

First order output of gen: $Y(s) = X(s)G(s) = \frac{a}{s(s+a)}$; time domain out $y(t) = 1 - e^{-at}$; a = only parameter affecting output; $\tau(1/a$ = time constant, time for step response to reach 63% of final value); T_r(rise time, time to go from 10% to 90% of FV, $T_r = 2.2/a$) T_s(settling time, time to reach and stay within 2% of FV, $T_s = 4/a$) **2nd Order** gen form: $G(s) = \frac{b}{s^2+as+b} = \frac{\omega_n^2}{s^2+2\zeta\omega_n s+\omega_n^2}$, poles: $s_1, s_2 = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2-1}$, Damping coef ζ(complex poles have real part $\sigma = \frac{-a}{2}$, $|\sigma|$= exponential decay freq, $|\omega_n|$= nat freq, $\zeta = \frac{|\sigma|}{\omega_n} = \frac{a}{2\omega_n}$, $a = 2\zeta\omega_n$) Undamped case($a = 0/\zeta = 0$, $G(s) = \frac{b}{s^2+b}$ poles will only have imaginary part $\pm jw$ and the natural freq $w_n = \sqrt{b}$ is the oscilltion frequency, nat resp: $c(t) = A\cos(w_n t - \phi)$, 2 imaginary poles) Underdamped($0 < \zeta < 1$, 2 complex poles at $\sigma_d \pm j\omega_d$, $c(t) = Ae^{-\sigma_d t}\cos(\omega_d t - \phi), \omega_d = \omega_n\sqrt{1-\zeta^2}$) Critically damped($\zeta = 1$, two real poles at σ_1, $c(t) = Kte^{\sigma_1 t}$) Over-damped($\zeta > 1$, $c(t) = K(e^{\sigma_1 t} + e^{\sigma 2t})$) Peak time($T_p = \pi/(\omega_n\sqrt{1-\zeta^2})$) %OS($\%OS = e^{(-\pi\zeta)/(\sqrt{1-\zeta^2})}$, $\zeta = -\ln(\%OS)/\sqrt{\pi^2+\ln^2(\%OS)}$)

Root Locus

FVT($\lim_{t\to\infty} x(t) = \lim s \to 0sX(s)$, not all functions have FV, depends on poles. I.e. if poles in rhs of s plane or pairs of complex conj poles on the im axis FV not exist) RL(tells us how the roots of a closed loop system change when a system parameter varies) CL sys($CLTF = K*G(s)/(1+K*G(s))$ for $K \to G(s)$ feedback loop. Pole locations change with K.) Sketching RL(#branches=#closed loop poles=#finite open loop poles=#finite open loop zeros, RL is symmetrical on real axis, RL begins at poles and ends at zeros) RL rules(RL approaches straight lines as $K \to \infty$, asymptotes defined by $\sigma_a = \frac{\Sigma\text{finite poles}=\Sigma\text{finite zeros}}{\#\text{finite poles}-\#\text{finite zeros}}$ and $\theta_a = \frac{(2k+1)\pi}{\#\text{finite poles}-\#\text{finite zeros}}$) Break pts(the roots of $\frac{d}{ds}[G(s)H(s)] = 0$

Stability

Proportional(impacts speed of response, $T(s) = \frac{K_pG_p}{1+K_pG_p}$) Integral(squashes SSE to 0, $G_c = K_I/s$, CLTF = $T(s) = \frac{G_cG_p}{1+G_cG_p}$) PI(speed and force SSE to 0, $G_c = K_I/s + K_p$, $T(s) = \frac{K_I+sK_p}{s^2+(1-K_p)s+K_I}$) Derivative(Introduces open loop zero, improves transient response, may introduce instability as K_D increases, $G_c = K_Ds$, $T(s) = \frac{G_cG_p}{1+G_cG_p}$; Changes damping effect $\zeta' = \zeta + K_d * P * \omega_n/2$) PID(SSE is reduced by increasing K_p, reduces to zero with K_I (trade-off of stability), K_D settling/peak time and damping effect. Approximates to $u(k) = K_pe(k) + K_IT\sum_{i=1}^k e(i) + \frac{K_D}{T}[e(k)-e(k-1)]$)

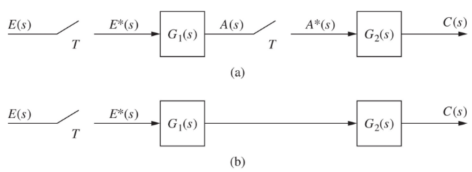
Z transforms

Resolution = max analog voltage$/2^n$, Sampling(ref input r is a seq of sample values $r(kT)$. Sampler: closes every T seconds for an instant $r(kT)\delta(t-kT)$) Z-transform($Z\{f(t)\} = F(z) = \sum_{k=0}^\infty f(kT)z^{-k}$, where $z = e^{sT}$) ZoH(holds the sampled signal to a constant value for the duration of the samping period, $ZoH(s) = \frac{1}{s} - \frac{e^{sT}}{s}$) Finding $G(z)$(Start with $G(s)$, inv L to find $g(t)$, z trans on $g(t)$ for $G(z)$) Power series($G(z) = a_o, a_1z^{-1} + ...$: i.e. $G(z) = \frac{z}{z^2-3z+2} \to$ long division $\to G(z) = z^{-1} + 3z^{-2} + 7z^{-3} + \ldots$ gives $g(0) = 0, g(T) = 1, g(2T) = 3, g(3T) = 7\ldots$)

Z trans ++

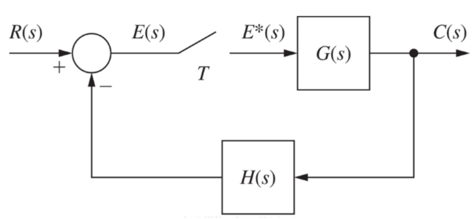
Properties(Linearity: If $x(n) = af_1(n) + bf_2(n), X(z) = aF_1(z) + bF_2(z)$. Time shifting: $Z[X(t)] = X(z), x(k-n) \to z^{-n}X(z)$ i.e. $x(k-1) \to z^{-1}X(z)$)

Discrete CL systems

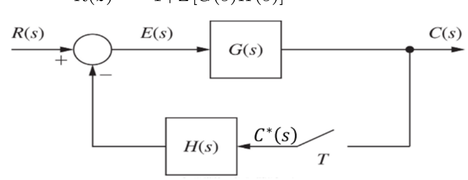


a) $TF = \frac{C(z)}{E(z)} = G_1(z)G_2(z)$

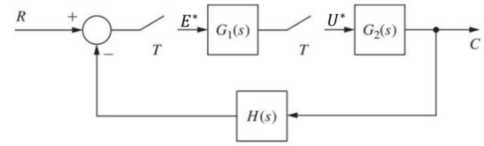
b) $TF = \frac{C(z)}{E(z)} = Z[G_1(s)G_2(s)]$



$TF = \frac{C(z)}{R(z)} = \frac{G(z)}{1+Z[G(s)H(s)]}$



$C(z) = \frac{Z[G(s)R(s)]}{1+Z[G(s)H(s)]}$



$TF = \frac{C(z)}{R(z)} = \frac{G_1(z)G_1(z)}{1+G_1(z)Z[G_2(s)H(s)]}$

Z stability

S-plane to z-plane(points on $j\omega \to$ on unit circle. If $a > 0$ and/or $e^{aT} > 1$, points on +ve real axis $\to$ outside the unit circle. Vise versa for -ve real ax and inside. Vertical lines at σ on s = circle on z w/r $= \varepsilon^{\sigma T}$. Horiz lines on s plane at $\omega \to$ lines from origin with angle ωT) Stability(Stable if all CLTF poles are inside unit circle. Unstable if any poles outside. Marginally stable if one or more poles are on unit circle and all others are inside)

RL on Z

Steps(1. Get OLTF, 2. Factor to get OL zeros and poles, 3. Plot roots of denom (CE) in z plane as k varies.) Rules(1. Loci originate on poles of OLTF and terminate on zeros, 2. Loci are symmetrical w.r.t. real axis, 3. # asymptotes = # fin poles (n_p) - # fin zeros (n_z), 3b. Angles of asymptotes: $\theta_a = \frac{(2k+1)\pi}{n_p-n_z}, k = 0,1,2,...(n_p-n_z-1)$. 4. Origin of asymptotes on real axis $\sigma = \frac{\sum\text{poles of OLTF}-\sum\text{zeros of OLTF}}{n_p-n_z}$, 5. Break-away point for locus between two poles (or breakin point for locus between two zeros) roots of $\frac{d}{dz}$OLTF = 0) Marginal(get CE, force magnitude of z to be 1, solve for K)

CCS to DCS and Diff eqn

$y(k)$(a function of current and previous inputs $u(k), u(k-1), u(k-2)\ldots$ and previous output $y(k-1), y(k-2)$. Use $u(k-n) \to z^{-n}U(z)$. i.e. $\frac{Y(z)}{U(z)} = \frac{z^2-0.0765}{z^2-0.153z+0.00243} \to y(k) = 0.153y(k-1) - 0.00243y(k-2) + u(k) - 0.0765u(k-2)$)

Euler's($u' = (u(k+1)-u(k))/T$)

Fwd Rect($\frac{U(z)}{E(z)} = \frac{a}{(z-1)/T+a}$)

Bwd Rect($\frac{U(z)}{E(z)} = \frac{a}{(z-1)/(Tz)+a}$)

Trap Rect($\frac{U(z)}{E(z)} = \frac{a}{2[(z-1)/(z+1)]/T+a}$)

Discrete approximation

Rules FWD: $D(s) = \frac{a}{s+a} \to D(z) = \frac{a}{(z-1)/T+a}$

BWD: $D(s) = \frac{a}{s+a} \to D(z) = \frac{a}{(z-1)/(Tz)+a}$

Trp: $D(s) = \frac{a}{s+a} \to D(z) = \frac{a}{2[(z-1)/(z+1)]/T+a}$

Approximating s

FWD: $s = (z-1)/T$, BWD: $s = (z-1)/(Tz)$,

TRP: $s = 2(z-1)/(T(z+1))$

Approximating z

FWD: $z = sT+1$, BWD: $z = 1/(1-Ts)$,

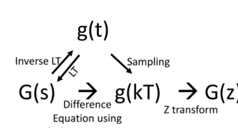
TRP: $s = (1+Ts/2)/(1-T2/2)$

Stability regions

FWD: possible for s sys stable but z unstable

BWD: possible for z sys stable but s unstable

TRP: mapped exactly, $z \leftrightarrow s$



RTOS
monotonicity clock is always increasing in value over time, no repeated events (times) Sys call(user space and kernel space are separate. When sys call, args passed from user to kernel. User proc becomes kernel proc when it executes sys call. Why APIs? easy to use, security, portability, efficiency) kernel module(can be loaded/unloaded on demand, extends func without reboot. Modules save memory, and make it so users do not have to rebuild and reboot every time something new is needed)Commands(insmod, rmmod, lsmod, modinfo)Threading(Threads share global data, code, and a heap but not a stack or registers)Priority(0-99 reserved for rt tasks, 100-139 for user proc. Nice val = [-20, 19], prio val = [100, 139]. Lower val = high prio, default nice val is 0 (120 prio val). Unpriv user can lower prio)RT Scheduling(Parameters $r_{i,j}$ release or arrival time, d_i abs deadline, D_i rel deadline, e_i exec time, R_i response time. Period p_i min length between release times of consecutive tasks. Phase $\phi_i = r_{i,1}$ (release time of first task)) Representation(4: (ϕ_i, P_i, e_i, D_i), 3: $(P_i, e_i, D_i) = (0, P_i < e_i, D_i)$, 2: $(P_i, e_i) = (0, P_i, e_i, P_i)$)Utilization($U = \sum e_i/p_i$)
Scheduling algos
CE (Adv: simple, predictable, no race/deadlock, efficient task dispatch. Dis: brittle, # of frames could be huge, release times are fixed, slicing tasks is difficult. **hyperperiod**: LCM of all periodic tasks. **Frames** divide a H into frames, timing is enforced only at frame boundaries, each task must fit within a single frame, multiple tasks can execute in a single frame, f= frame size, frames per H = $F = H/f$) Frame constraints(C1: $f \geq \max_{1\leq i\leq n} e_i$ (tasks should finish executing within a single frame). C2: H must be evenly divided by f. C3: f should be small enough that there is a complete frame between the release and deadline of each task: $2f -$ GCD$(P_i, f) \leq D_i$ for each task.) RM(prio = 1/period, Schedulability **test 1**: SUFF & NEC $D_i \geq P_i$ fo all proc, periods are integer multiples of eachother, then $U \leq 1$ means is schedulable.**test 2**: SUFF BUT !NEC $U \leq 0.693$. **Test 3**: SUFF & NEC (the complicated one) $\omega(t) = \sum_{k=1}^{i} \lceil \frac{t}{P_k} \rceil * e_k \leq t$ where $0 \leq t \leq P_i, t = k_j * P_j, j = 1...i, k_j = 1, ..., \lfloor \frac{P_i}{P_j} \rceil$. If $P_i \neq D_i$ use min.) EDF(**Test1** SUFF & NEC $\sum_{i=1}^{n} \frac{e_i}{P_i} \leq 1$ **Test 2** SUFF BUT !NEC $\sum_{i=1}^{n} \frac{e_i}{\min(D_i, P_i} \leq 1$, RM vs EDF: edf more flex less U, RM more predictable and stable) NPCS(when a task holds a resource, it executes at a higher prio than all other tasks. High prio task only blocked when low prio in CS. Once blocker completes, no low prio gets resource until high prio completes. Task recovers its actual prio when all resources unlocked. ADV: simple, static & dynamic sched, effective. DIS: can be blocked for long time.)PIP(increase priorities only when there's resource contention. When a task T1 is blocked due to the non-availability of a resource, the task T2 that holds the resource inherits prio of T1. T2 executes at inherited prio until it releases R. If it still blocks a task after release, it inherits from whatever it is still blocking. STILL HAS DEADLOCKS)PCP(ceil(R) is highest prio among all tasks that need R. Prio ceil of a sys at any given time a set of resources are being used = highest ceil(R) in the set. Resource allocation: when a task requests R, if R is held by another task the req fails and task is blocked. Else, if R is free, if req task's prio $>$ current ceiling, R is allocated to the task. If not, request denied and blocked. **Prio inheritance** when T2 blocks T1, T2 inherits prio T1 until it release every resource whose ceil is $\geq$ inherited prio. A task can acquire a resource iff the resource is free and it has higher prio than ceiling of existing resources)

MC. Given the task set: T1(4, 1); T2(5, 1); T3(10, 2), and you are asked to find a CE schedule using flow graph as shown in Figure 1. Assume that the frame size is 2. Which of the following is true? (There are 11 job nodes and 10 frame nodes in the flow graph.) MC. The __ scheduling algorithm schedules periodic tasks using a static priority policy with preemption (RM) MC. What does the term priority inversion refer to? (A situation where a high priority process must wait for a low priority process.)MC. How can a priority inversion be corrected? (Temporarily raising the priority of a low priority process.) MC. If a set of processes cannot be successfully scheduled by RM scheduling algorithm, then: (None of the above) MC. Consider the task set T = (8, 4), (10, 2), (12, 3), which of the following is true: (T is not RM schedulable but EDF schedulable) MC. Which of the following protocol can handle the deadlock and priority inversion? (PCP) MC. A Task set consists of n pre-emptive and periodic tasks. If the task is NOT RM schedulable, which of the following is correct? (The CPU utilization is over 1) MC. What is the value of output value of g if the following code is given: (There is a race condition, so the value of g cannot be determined) MC. Which of the following is a drawback of thread programming? (Without synchronization, race condition on shared variables can be disastrous) MC. Which of the following output is not possible? Odd and Even code (1 0 2 4 5 3 6 7 9 8) MC. In CE scheduling, which of the following are correct? (Frame size of CE algorithm cannot be too small since we want an instance of a task completed within a single frame) MC. What is the output of the program? Note that SIGCHILD is sent to the parent and child process when it exits, is interrupted, or resumes after being interrupted. Handler and starts with val=9 (21). MC. Real-time systems must have __ (preemtive kernels) MC. Given 3 periodic tasks T1, T2, T3. They have the same execution time, but different periods. The periods of the tasks are 4, 8, and 16. What is the maximum execution time so that the tasks are RM schedulable (16/7) MC. Which of the following is NOT a benefit of using kernel module instead of installing all anticipated functionalities into a base kernel? (Allow preemption) MC. What is the output of the program? Swap a and b. (a=13, b=23) MC. What is the last line of the output program? foo, int a, static int sa. (a=15, sa=35)

MC. When executing a C program, CPU runs in __ mode unless it is making a system call. (user) MC. Can child process access static variable created by a parent process before fork()? (Yes, but the modification can be seen only in child process, and the value in parent process will not be changed) MC. Multi-processor systems have advantage of __ over multi-thread systems. (reliability) MC. Which of the following is NOT shared by threads in the same process? (stack) MC. Which of the following is NOT true? (signal() function is used to send a signal to a process) (Actual truth is a field is updated in the signal table when the signal is sent) MC. For PCP, which of the following is true? (The system priority ceiling may change only when a resource is allocated or released) MC. Which of the following about the protocols to prevent priority inversion is correct? (Even if NPCS protocol is used, high priority process may still have to wait for a low priority process to release the resources) MC. A Task set consists of 5 pre-emptive and periodic tasks. If the task set is NOT RM schedulable which of the following can be inferred? (The CPU utilization is over 0.743) MC. Consider the task set T = {(P = 2, C = 1), (P = 4, C = 1), (P = 5, C = 1)}, which of the following is true? (T is both RM and EDF schedulable) MC. Which of the following statements about RM is correct? (Sufficient RM schedulability condition is based on processor utilization. It can be used at run-time to predict the schedulability but may lead to poor processor utilization) hlMC. Which of the following tasks are DM schedulable? (T1:(5, 1) T2:(8, 2) T3:(12, 4) and T4:(20, 2))

MC. The impulse response of a system is given by $Y(z) = (z^3 + 2z^2 + 2)/(z^3 - 25z^2 + 0.6z)$. Determine the values of y(nT) at the first four sampling instants. $(y(0) = 1, y(T) = 27, y(2T) = 674.4, y(3T) = 16845.8)$ MC. The clock of the Raspberry Pi slows down at a rate of 40×10-6 seconds per second. Suppose that you can connect the device to a clock server that allows you to correct the time to within 5 seconds of the true time. You would like the displayed time to be accurate to within one minute of the true time. How often should the device be synchronized to the clock server? (382 hrs) MC. The root locus is the trace of the roots of the characteristic equation in the s-plane? (as a system parameter is changed) MC. Which is the controller that will improve the transient response? (Derivative) MC. In order to simulate the given signal as below, you have to change the value of ( ) module: (PID Controller) MC. Consider s-domain function $Y(s) = 10/(s(s+2)(s+6))$. Let T be the sampling time. Then, in the z-domain the function Y(z) is: $(Y(z) = 5/6(z/(z-1)) - 5/4(z/(z-e^(-2T)))+5/12(z/(z-e^(-6T))))$ MC. If input $x[n] = [1, 4, 7]$ and output is $[-1, -4]$, the system difference equation is? $(y[n]+4y[n-1]+7y[n-2] = -x[n]-4x[n-1])$ MC. A discrete time signal $x[n] = \delta[n-3] + 2\delta[n-5]$ has a z-transform X(z). If Y(z)=X(-z) is the z-transform of another signal y[n], then (y[n]=-x[n])