SIMULATION OF ROOM TEMPERATURE CONTROL SYSTEM (RTCS) WITH MSC/NASTRAN *

Ernest B. Paxson, Jr. **
Paxson Engineering Services
Albuquerque, NM 87111

I. Introduction -

The problem to which the title of this paper refers was spawned in the process of learning how to implement MSC/NASTRAN in a structure-optics-controls application. In choosing a vehicle to accomplish the first step of this learning process, one might reason that a Room Temperature Control System (RTCS) would furnish a non-trivial mathematical model but one not so complex as to prevent obtaining a closed form solution with which to compare the MSC/NASTRAN analysis results.

Both the Heater/Cooler Distribution System and the Room (house walls and space) Thermal Behavior were modelled as 1st-order components. This presented an interesting problem for MSC/NASTRAN since there were no 2nd-order (mass or inertia) terms in the RTCS equations representing control loop behavior. The problem was made doubly interesting due to the non-linear ON-OFF switching component in that the switch function manifested an oscillatory behavior instead of being constant in the ON (or OFF) mode during the initial MSC/NASTRAN executions of this simulation. Adding the right amount of mass and damping resulted in a very well behaved switching function and excellent correlation between the MSC/NASTRAN analysis and an independent FORTRAN simulation of the RTCS.

---

* This work was partially funded on subcontract No. 0118-10.85-01 with Applied Technology Associates; Albuquerque, NM

** Consultant to Air Force Weapons Labs System Communications Division, Kirtland AFB, NM

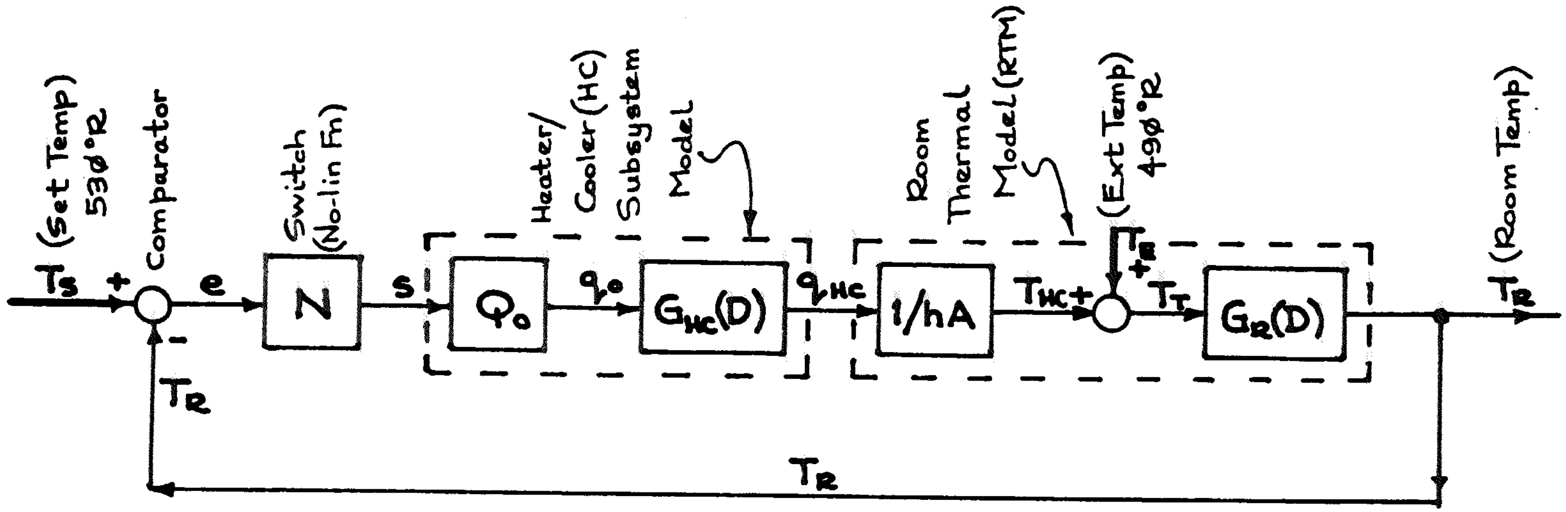
II. Definition of the RTCS

Illustrated in Figure 1 is a block diagram of the RTCS. The values chosen for the several parameters (e.g., τ_R, τ_{HC}, Q_0,) could represent the characteristics of an average-sized house and heating (or air-conditioning) system. Both the Heater/Cooler (HC) and house (internal room space + walls + insulation) thermal behaviors were modelled as 1st-order systems. The thermal time constant (τ_R) for the house was assumed to be 5.0 hrs; and, the thermal time constant (τ_{HC}) for the HC was selected to be 0.2 hrs. The value for TAU_R is based on approximate observations made in a home (ours) in Albuquerque, NM. The assumed τ_{HC} is probably twice as long as it takes a forced air system (ours again) to put out warm air. In checking various gas-fired furnace nameplate data, one might conclude that a Q_0 of 150,000 Btu/hr would be a reasonable value. Since the average winter night temperature in Albuquerque is a little below freezing, the exterior temperature, T_E, was chosen to be 30 ^oF (490 ^oR). The thermostat set temperature, T_S, was fixed at 70 ^oF (530 ^oR); and, the initial condition on room (or house interior) temperature was assumed to be 50 ^oF (510 ^oR). Under these conditions the HC would be acting, of course, as a heater. The thermostat switch function, s(t), was designed to turn off the HC when T_R came within 2.0 ^oF of the set temperature. An ON-OFF temperature dependent switch function constitutes a non-linear (with respect to time) component in the RTCS.

III. Procedure for Implementing Simulation of RTCS in MSC/NASTRAN

Since there was no physical load-bearing structure involved in the RTCS, the first approach was to assign all the system variables with unique EXTRA POINT identification numbers and input the system equations via the transfer function (TF) cards. There may be a way to use such an approach without getting fatal error messages, but a complete execution of the RTCS could not be accomplished until certain variables were assigned scalar (SPOINT) identification numbers; in particular, the set temperature, T_S; exterior temperature, T_E; and the switch function, s(t). Since the temperature variables were being treated as analogous to displacement components, it became necessary to attach scalar springs (CELAS2) with spring constants K = 1.0 to the scalar points representing T_S and T_E. In this way displacements (temperatures) would be numerically equal to the applied forces (temperature load) so that one could specify on the DAREA card image the actual temperature loadings pertaining to those points. Similarly, the NOLIN1 card image implies a force being applied to the scalar point representing the switch function, s(t). With a scalar spring of K = 1 attached thereto, it became permissible to specify the non-linear switch function with its actual values on the referenced TABLED1 card.

BLOCK DIAGRAM FOR ROOM TEMPERATURE CONTROL SYSTEM



where: $G_{HC} = \frac{1}{\tau_{HC}D+1}$ and $G_R = \frac{1}{\tau_R D+1}$

System Parameters:
$\tau_{HC} = 0.2$ hr Time Constant for HC
$Q_0 = 1.5 \times 10^5$ Btu/hr Max Output of HC
$\tau_R = 5.0$ hr Time Constant for RTM
$hA = 800$ Btu/hr °F (See NOTE below)

Equations for Block Diagram Components:

Comparator: (1) $e = T_S - T_R$

Switch: (2) $s = \begin{cases} 1.0 & \text{if } e > T_{INC} \\ 0.0 & \text{if } |e| \le T_{INC} \\ -1.0 & \text{if } e < -T_{INC} \end{cases}$

where: $T_{INC} = 2.0$°F

HC: (3) $q_0 = Q_0 s$

(4) $q_{HC} = \frac{1}{\tau_{HC}D+1} \cdot q_0$

RTM: (5) $T_{HC} = \frac{1}{hA} q_{HC}$

RTM: (6) $T_T = T_E + T_{HC}$

(7) $T_R = \frac{1}{\tau_R D+1} \cdot T_T$

NOTE: Heat Loss (or Gain) per °F temp difference between inside and outside of Room represented by hA.

Figure 1 - Block Diagram for Room Temperature Control System (RTCS)

A. Transfer Function Representation in MSC/NASTRAN

Second-order (or lower) transfer functions are representable by the form:

$$u_e = -\sum_i (a_0^{(i)} + a_1^{(i)} p + a_2^{(i)} p^2) u_i / (b_0 + b_1 p + b_2 p^2) \qquad (1)$$

in which u_e represents the variables in the control system of interest and u_i can represent other control system variables or pertinent structure displacements. (Higher order transfer functions may be handled by this method if they are decomposed into a consistent number of second- or lower-order transfer functions.) In order for MSC/NASTRAN to treat Eqn (1) as a differential equation (DE), it is arranged in the following format:

$$(b_0 + b_1 p + b_2 p) u_e + \sum_i (a_0^{(i)} + a_1^{(i)} p + a_2^{(i)} p^2) u_i = 0 \qquad (2)$$

In this way Eqn (2) is easily incorporated into the general dynamic matrix equation

$$[K_{dd} + B_{dd} p + M_{dd} p^2]\{u_d\} = \{P_d\} \qquad (3)$$

which MSC/NASTRAN uses in the transient response analysis solution sequences. The coefficients of the zeroeth, first, and second order terms in Eqn (2) will correspond to the stiffness, $K_{dd}^{(2)}$; damping, $B_{dd}^{(2)}$; and mass, $M_{dd}^{(2)}$, matrices; which are merged with the Analysis SET (ASET) in the solution sequence before the DE set is integrated with respect to time. $K_{dd}^{(2)}$, $B_{dd}^{(2)}$, and $M_{dd}^{(2)}$ entries may be input via DMIG or TF Bulk Data cards. TF cards were used in this problem because they were more compatible with the format of Eqn (2).

B. Transfer Function Table

In order to facilitate inputting the system equations for the RTCS, a Transfer Function Table was set up as presented in TABLE I. In TABLE I the numbers given in the "TFEQN #" column correspond to those given to the equations shown in Figure 1. Also shown are the EPOINT (or SPOINT) identification numbers for ease in associating the TABLE I entries with the TF Bulk Data card images appearing in the MSC/NASTRAN input deck reproduced in Appendix "A".

TABLE I

Transfer Function Table for RTCS

TFEQN #	EPOINT ID	u_e	b_0	b_1	b_2	EPOINT ID	u_i	a_0	a_1	a_2
1	11	e	-1.0			18	T_S	1.0		
						16	T_R	-1.0		
3	12	q_0	-1.0			17	s(t)	Q_0		
4	13	q_{HC}	-1.0	$-\tau_{HC}$		12	q_0	1.0		
5	14	T_F	-1.0			13	q_{HC}	1/hA		
6	15	T_T	-1.0			19	T_E	1.0		
						14	T_{HC}	1.0		
7	16	T_R	-1.0	$-\tau_R$		15	T_T	1.0		

NOTE: Eqn (2) for the non-linear Switch Function in Figure 1 is handled by the NOLIN1 card.

C. Initial Conditions and Time Window

TIC Bulk Data cards were used to specify initial conditions on the RTCS for the following variables:

$e(0) = 20.0\ ^oF$ $\quad$ $q_0(0) = 1.5$ E+5 Btu/hr

$T_T(0) = 490\ ^oR$ $\quad$ $T_R(0) = 510\ ^oR$

$s(0) = 1.0$

The time window selected was for 2.0 hrs with an integration time step of 0.0001 hr for the MSC/NASTRAN execution.

IV. Results of the MSC/NASTRAN Simulation of the RTCS

A. Behavior of Switch Function in RTCS with No Mass or Damping

Since page 3.6-8 of the MSC/NASTRAN "Handbook for Dynamic Analysis" (HDA) was not read before the first successful (no error messages) run was made, no mass or damping was added to the RTCS. Even though the Room Temperature, T_R, and the Heat Added, q_{HC}, to the Room seemed fairly well behaved; the Switch Function, s(t), had rather severe oscillations. Illustrated in Figure 2 is a plot of the Switch Function, s(t), as represented by $q_0 = s(t)Q_0$, which is the heat (or cold) provided by the Heater/Cooler unit. Heat buildup in the distribution system that inputs the heat to the room , q_{HC}, is also plotted in the same Figure 2; and, it behaves reasonably well.

One cannot add mass to just any point in the RTCS; otherwise, the desired improvement in the Switch Function will not result. The mass must be attached to SPOINT ID = 17 in the RTCS, which represents the value of the Switch Function.

B. Behavior of Switch Function in RTCS with Added Mass

It was surmised that the parasitic mass discussed on page 3.6-8 of the HDA should be a small value; however, in order to give added assurance to this conjecture, a small ramp temperature of 0.1 ^{o}F was given to the Switch Function so that the "ON-OFF" switch would not have infinite slope. Figure 3 shows the results of adding M_{17} = 1.0E-4 to SPOINT ID = 17. One might note that the Switch Function did fine until a change occurred which started the oscillations again. At this point it seemed some added damping might help the situation.

C. Behavior of Switch Function in RTCS with Added Mass and Damping

Adding B_{17} = 1.0E-3 via a CDAMP2 card to the Switch Function improved its response quite a bit as indicated by the plot in Figure 4, but some oscillations were still present. Thinking back to an experience with critically damped galvanometers brought up the idea to try using critical damping for the RTCS Switch Function.

D. Behavior of Switch Function in RTCS with Critical Damping

Since the Switch Function could be viewed as a one degree-of-freedom spring-mass-damper system, one may calculate the critical damping by the relationship:

$$B_{crit} = 2\sqrt{(KM)}$$

With K_{17}= 1.0 and M_{17}= 1.0E-6, a B_{crit}= 2.0E-3 was obtained. The response of the Switch Function with these values of stiffness, mass, and damping is depicted in Figure 5; which is a significant improvement over earlier responses.

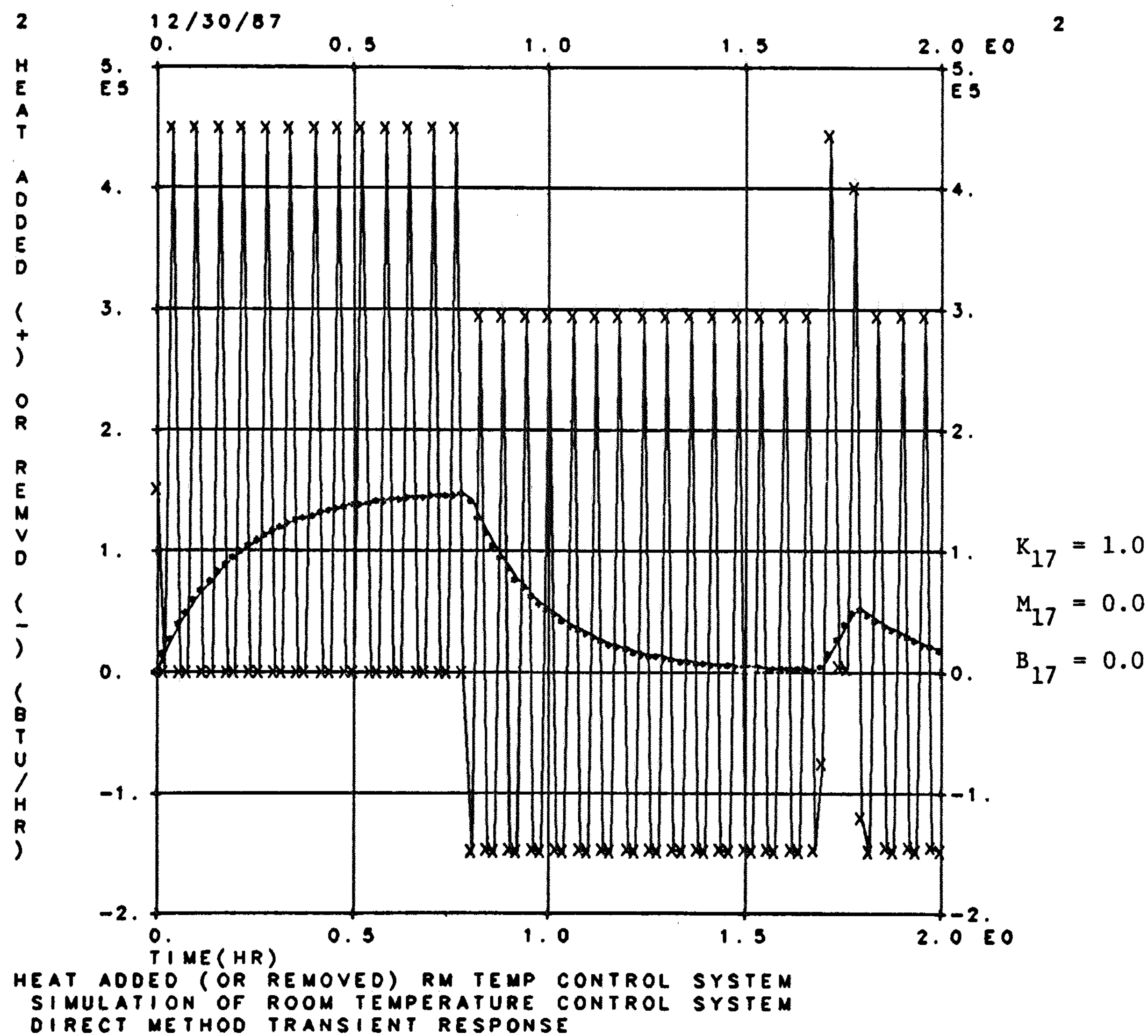


Figure 2 - Switch Function as Represented by $q_o = s(t)Q_o$ and Heat Added, q_{HC}, to Room Temperature Control System (RTCS) - No Mass or Damping

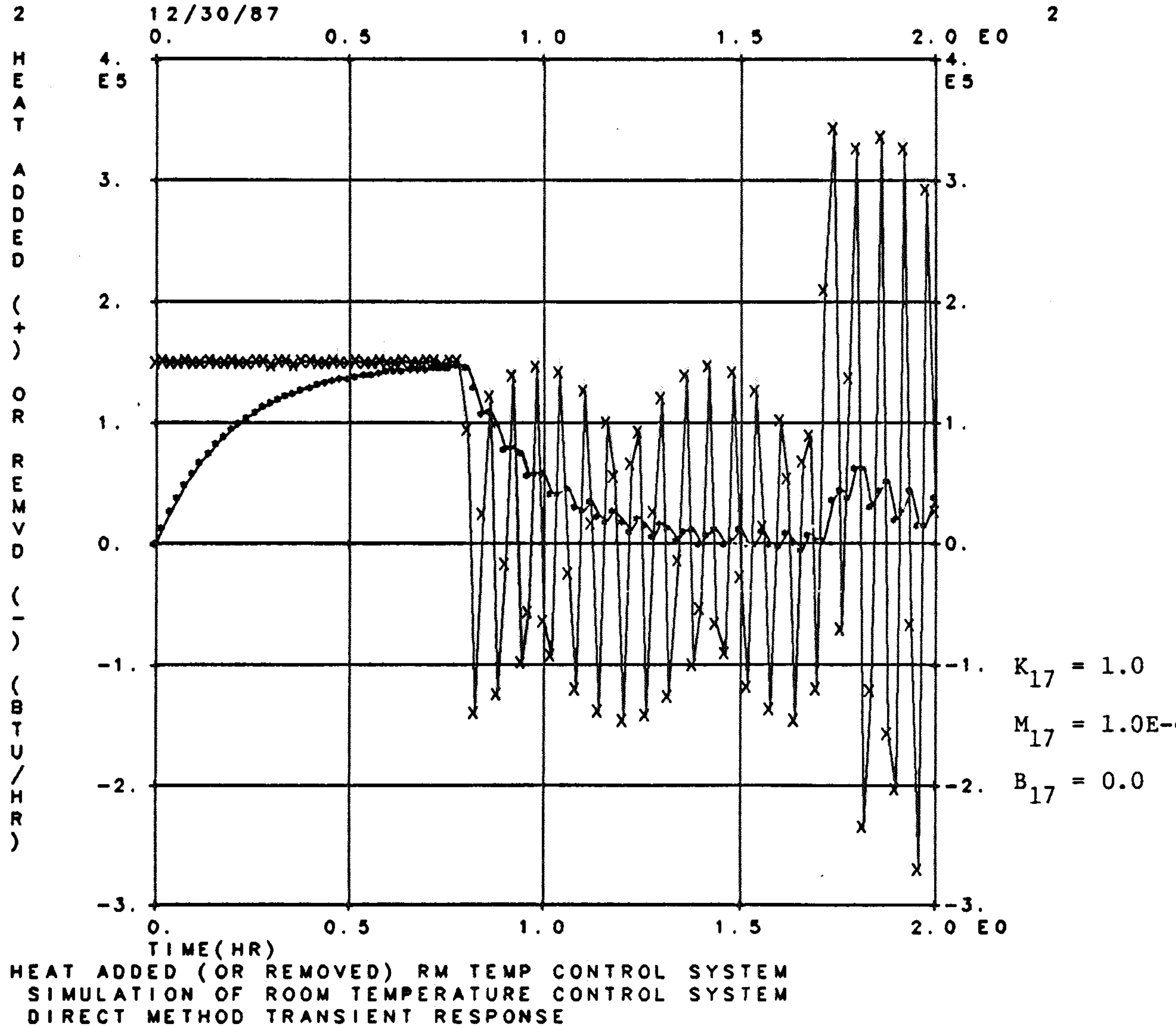
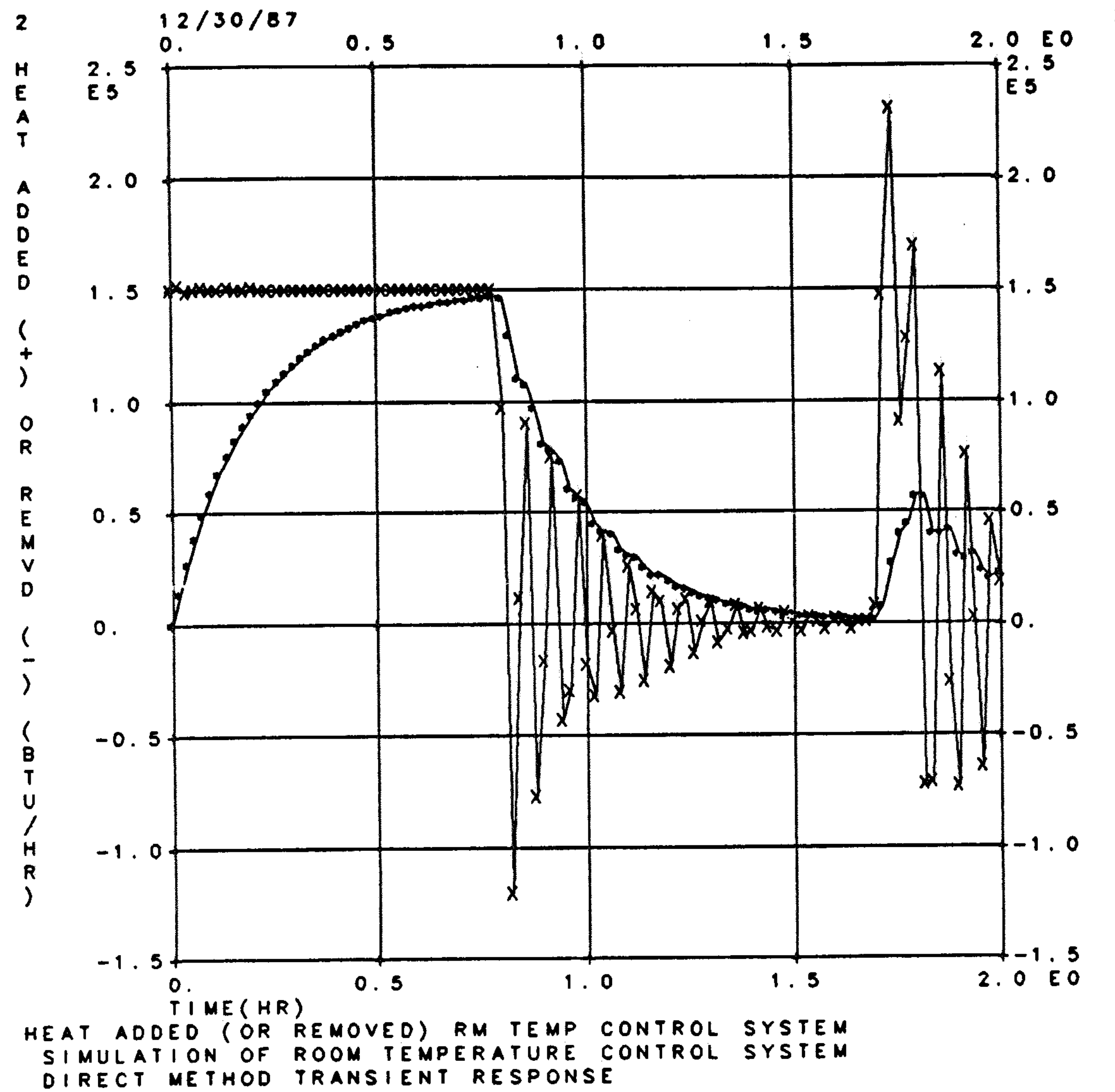


Figure 3 - Switch Function as Represented by $q_o = s(t)Q_o$ and Heat Added, q_{HC}, to Room Temperature Control System (RTCS) - No Damping

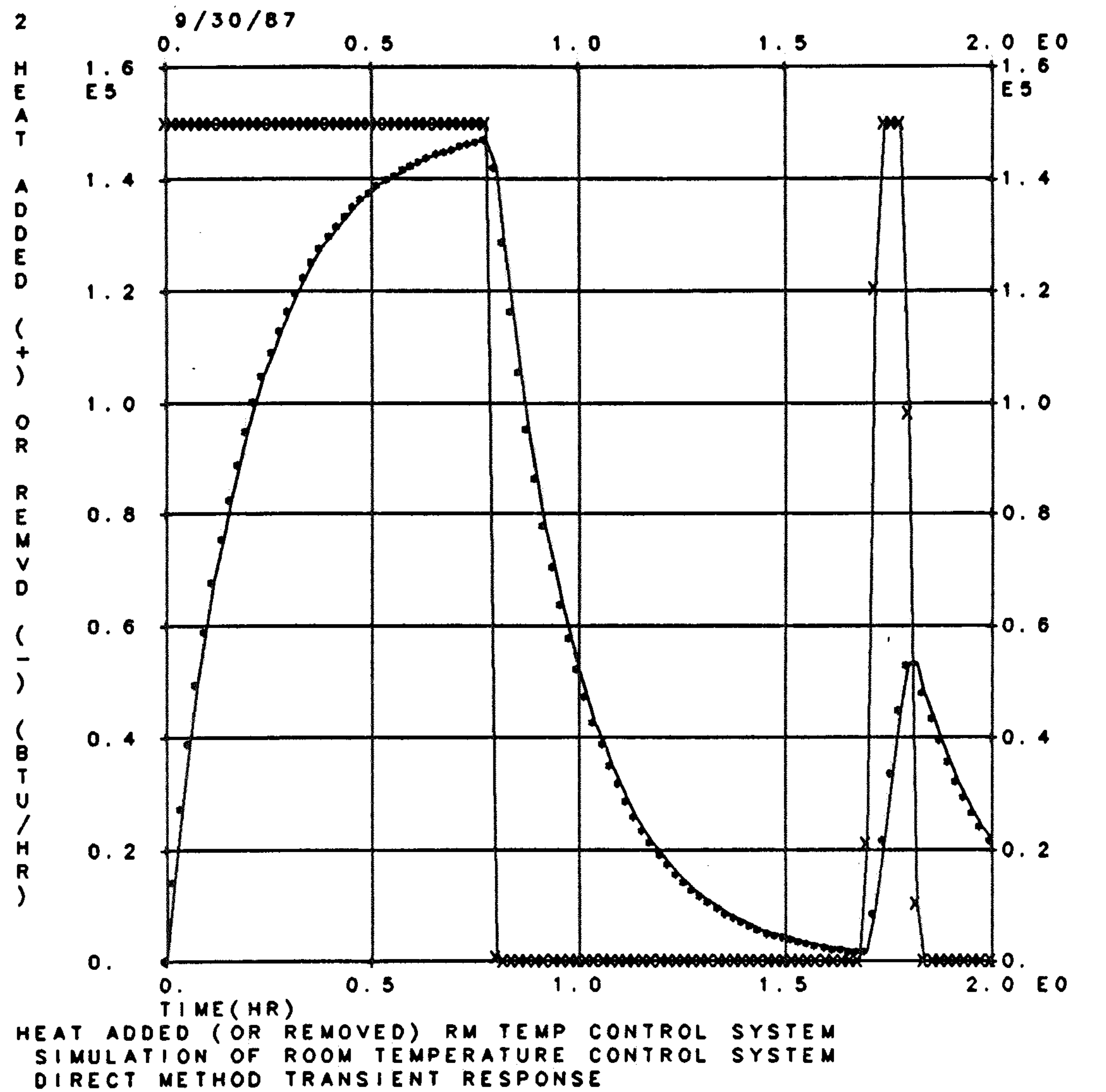


$K_{17} = 1.0$

$M_{17} = 1.0E-4$

$B_{17} = 1.0E-3$

Figure 4 - Switch Function as Represented by $q_o = s(t)Q_o$ and Heat Added, q_{HC}, to Room Temperature Control System (RTCS) - Mass and Damping Added



$K_{17} = 1.0$

$M_{17} = 1.0E-6$

$B_{17} = 2.0E-3$

Figure 5 - Switch Function as Represented by $q_o = s(t)Q_o$ and Heat Added, q_{HC}, to Room Temperature Control System (RTCS) - Critical Damping

E. Comparison of MSC/NASTRAN and FORTRAN Simulations of the RTCS

In TABLE II a comparison of the RTCS simulation for selected values of time by two independent computer codes is presented. The FORTRAN program used to simulate the RTCS is included as Appendix "B". Since the FORTRAN program was primarily an evaluation of the closed form solution (and not an integration) of the RTCS equations, the 0.002 hr time step appeared to sufficiently small for the control components to sense changes in the system variables. Figure 6 shows a plot of the Room Temperature, T_R, variation for the RTCS for the conditions specified in this paper.

V. Conclusions

The RTCS seemed to be an ideal test article for this author to gain experience in using the MSC/NASTRAN capability to integrate control equations with structures or other interdisciplinary systems.

TABLE II

Room Temperature Control System
Comparison of FORTRAN and NASTRAN Transient Response Analyses
(Te = 490 deg R, Ts = 590 deg R, Tr(0) = 510 deg R)

Time (hr)	FORTRAN Tr (deg R)	NASTRAN Tr (deg R)		FORTRAN Qhc (Btu/hr)	NASTRAN Qhc (Btu/hr)
0.00	510.000	510.000	** ON	0.00	0.00
0.02	509.956	509.957		14,274	14,297
0.04	509.981	509.981		27,190	27,210
0.06	510.066	510.067		38,877	38,894
0.08	510.207	510.208		49,452	49,467
0.10	510.397	510.398		59,020	59,033
0.20	511.936	511.937		94,818	94,823
0.40	516.723	516.725		129,700	129,699
0.60	522.401	522.402		142,530	142,529
0.78	527.668	527.669		146,960	146,961
0.80	528.246	528.248	* OFF	141,370	142,327
0.82	528.764	528.772		127,920	128,787
0.84	529.217	529.228		115,740	116,531
0.88	529.949	529.968		94,764	95,407
0.90	530.240	530.261		85,746	86,328
1.00	531.112	531.114		52,008	52,360
1.20	531.106	531.148		19,132	19,261
1.40	530.085	530.129		7,038	7,085
1.70	528.014	528.057		1,570	1,806
1.72	527.899	527.928	** ON	14,331	8,617
1.74	527.852	527.851		27,242	21,557
1.76	527.866	527.838		38,924	33,780
1.78	527.936	527.884		49,494	44,840
1.80	528.050	527.980	* OFF	53,177	53,131
2.00	528.200	528.298		19,563	21,656

NASTRAN Time Step - 0.0001 hr
FORTRAN Time Step - 0.002 hr

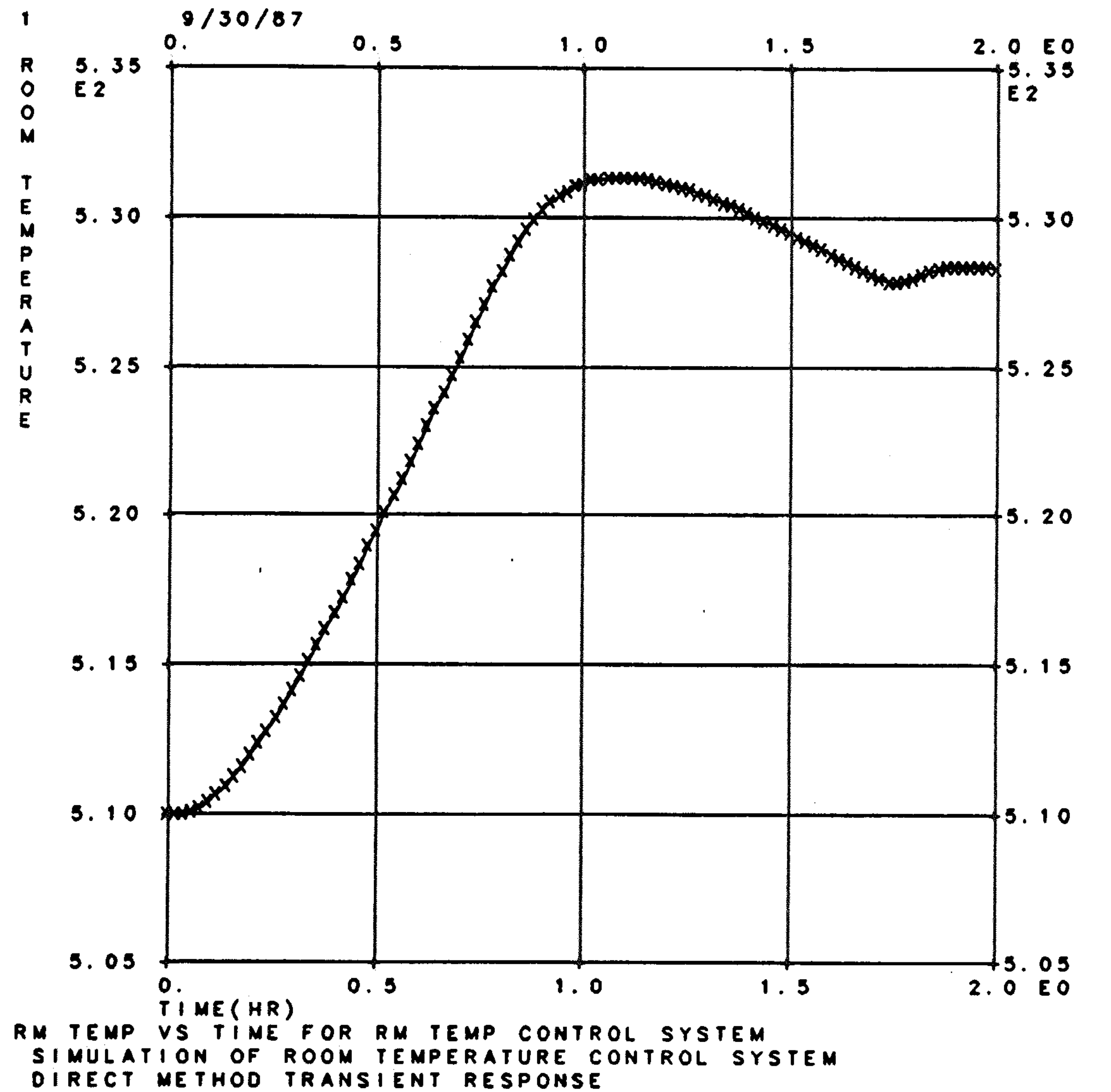


Figure 6 - Room Temperature, T_R, vs Time for Room Temperature Control System (RTCS)

Calculated by MSC/NASTRAN

APPENDIX "A"

NASTRAN Input Deck for RTCS Simulation

```
*deck irmtmp
id rm,tmp
sol 27
time 10
diag 8
alter 382
matprn   k2pp,b2pp,m2pp//  $
alter 391
matprn   kdd,bdd,mdd//  $
endalter
cend
title = simulation of room temperature control system
subtitle = direct method transient response
nonlinear=17
ic=10
tfl=1
tstep=72
dload=75
set 8=18,19
oload=8
set 9=17
nlload=9
set 7=11,12,13,16,17
sdispl=7
output(xyplot)
plotter nastplot(t,0)
xpaper=11.0
ypaper=11.0
xgrid=yes
ygrid=yes
xaxis=yes
yaxis=yes
curvelinesymbol=1
xtitle=time(hr)
ytitle=room temperature
tcurve=rm temp vs time for rm temp control system
xyplot sdispl / 16(t1)
ytitle=heat added (+) or remvd (-) (btu/hr)
tcurve=heat added (or removed) rm temp control system
xyplot sdispl / 12(t1), 13(t1)
$
```

```
begin bulk
$
celas2,17,1.0,17,0
cmass2,17,1.0-6,17,0
cdamp2,17,2.0-3,17,0
celas2,18,1.0,18,0
celas2,19,1.0,19,0
spoint,17,18,19
tf,1,11,0,-1.0,,,,,+tf1a
+tf1a,18,0,1.0,,,,,,+tf1b
+tf1b,16,0,-1.0
$$tf,1,17,0,1.0
tf,1,12,0,-1.0,,,,,+tf3a
+tf3a,17,0,1.5+5
tf,1,13,0,-1.0,-0.2,,,,+tf4a   $heat pump transf fn
+tf4a,12,0,1.0
tf,1,14,0,-1.0,,,,,+tf5a
+tf5a,13,0,1.25-3
tf,1,15,0,-1.0,,,,,+tf6a
+tf6a,19,0,1.0,,,,,,+tf6b
+tf6b,14,0,1.0
tf,1,16,0,-1.0,-5.0,,,,+tf7a   $rm temp transf fn
+tf7a,15,0,1.0
tic,10,11,0,20.0
tic,10,12,0,1.5+5
tic,10,15,0,490.0
tic,10,16,0,510.0              $initial rm temp
tic,10,17,0,1.0
epoint,11,thru,16
nolin1,17,17,0,1.0,11,0,117
tabled1,117,,,,,,,,+tb117a
+tb117a,-50.0,-1.0,-2.1,-1.0,-1.9,0.0,1.9,0.0,+tb117b
+tb117b,2.1,1.0,50.0,1.0,endt
tload1,75,76,,0,118
darea,76,19,0,490.0,18,0,530.0
tabled1,118,,,,,,,,+tb118
+tb118,0.0,1.0,5.0,1.0,endt
tstep,72,20000,0.0001,200
param,newseq,-1
enddata
```

APPENDIX "B"

FORTRAN Program for RTCS Simulation

```
      program rmtmp(tape4,tty,tape5=tty,output,tape6=output,tape7)
c
c   input data
c
      read(4,*)tset,tout,taur,qfo,tinc,tauhc,cpm,tr0
      read(4,*)itf,del
c
      write(6,9)
 9    format(1x,'time(hr)',6x,'tr(deg R)',6x,'qhc(btu/hr)',//)
      tt=0.0
      sw=0.0
      ton=tt
      i=0
      np=0
      tr=tr0
      do 10 it=1,itf+1
           et=tset-tr
           if(et.ge.tinc)then
           st=1.0
           go to 12
           endif
c
           if(et.le.-tinc)then
           st=-1.0
           go to 12
           endif
c
           if(abs(et).lt.tinc)then
           st=0.0
           if(sw.eq.0.0) go to 15
                sw=0.0
                toff=tt
                troff=tr
                qhcoff=qhc
                cf=qhcoff*exp(toff/tauhc)
                cr=(troff-tout+tf*trat*cf*exp(-toff/tauhc))*
     1             exp(toff/taur)
                go to 15
           endif
c
           go to 30
c
 12        if(sw.eq.1.0) go to 15
                sw=1.0
                ton=tt
                tron=tr
                qhcon=qhc
                trat=tauhc/(taur-tauhc)
                tf=taur/cpm
                cf=(qhcon-st*qfo)*exp(ton/tauhc)
                cr=(tron-tout-tf*(st*qfo-trat*cf*exp(-ton/tauhc)))*
     2             exp(ton/taur)
c
c   calculate tr and qhc as a function of time tt
c
 15        tr=tout+tf*(st*qfo-trat*cf*exp(-tt/tauhc))+cr*
     3        exp(-tt/taur)
           qhc=st*qfo+cf*exp(-tt/tauhc)
c
c   check to see if 1st or 2nd tr calculation after del time incr
c
           if(i.eq.1) go to 10
                if((it-1).ne.(10*np)) go to 14
                np=np+1
                write(6,13)tt,tr,qhc
 13             format(1x,f8.2,6x,f9.4,5x,e12.5)
 14             tt=it*del
                i=1
                go to 15
 10   i=2
 30   stop
      end
```