

Sep 08, 11 5:31

diff_residue.txt

Page 1/6

```

146,148c146,156
< function [r, p, k, e] = residue (b, a, varargin)
<
<   if (nargin < 2 || nargin > 4)
---
> %%%%%%%%%%% old_code %%%%%%%%%%%
> %%%function [r, p, k, e] = residue (b, a, varargin)
> %%%
> %%%   if (nargin < 2 || nargin > 4)
> %%%%%%%%%%% new_code %%%%%%%%%%%
> function [r, p, k, e] = bg_residue (b, a, varargin)
>   global      G_FID;
>   global      G_DEBUG;
>   bg_enable_notes=1;   %% set to 1 to print notes in files
>   if (nargin < 2 || nargin > 5)
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
151a160,167
> %%%%%%%%%%% new_code %%%%%%%%%%%
>   if (nargin>2 && ischar(varargin{nargin-2}))
>       MODE=varargin{nargin-2};
>       nargin=nargin-1;
>   else
>       MODE="MODE_S";
>   end
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
162c178,204
<   [r, p] = rresidue (b, a, varargin{1}, toler, e);
---
> %%%%%%%%%%% old_code %%%%%%%%%%%
> %%%   [r, p] = rresidue (b, a, varargin{1}, toler, e);
> %%%%%%%%%%% new_code %%%%%%%%%%%
> %% Here the role of r and p are interchanged with a and b.
>   if G_DEBUG
>       fprintf(G_FID,'Inputs to bg_rresidue are:\n');
>       bg_print('r',b);
>       bg_print('p',a);
>       fprintf(G_FID,'MODE =%s\n',MODE);
>       fprintf(G_FID,'toler =%+17.10e\n',toler);
>       fprintf(G_FID,'----- ');
>       fprintf(G_FID,'Into [a,b]=bg_rresidue');
>       fprintf(G_FID,'(r,p,varargin{1},toler,e,MODE);\n');
>   end
> %% Here the role of r and p are interchanged with b and a.
>   [r, p] = bg_rresidue (b, a, varargin{1}, toler, e, MODE);
> %% Here the role of r and p are interchanged with b and a.
>   if G_DEBUG
>       fprintf(G_FID,'----- ');
>       fprintf(G_FID,'Out of [a,b]=bg_rresidue');
>       fprintf(G_FID,'(r,p,varargin{1},toler,e,MODE);\n');
>       fprintf(G_FID,'Outputs from bg_rresidue are:\n');
>       bg_print('b',r);
>       bg_print('a',p);
>   end
> %% Here the role of r and p are interchanged with b and a.
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
220,226c262,342
<   small = max (abs (p));
<   if (isa (a, "single") || isa (b, "single"))
<       small = max ([small, 1]) * eps ("single") * 1e4 * (1 + numel (p))^2;
<   else
<       small = max ([small, 1]) * eps * 1e4 * (1 + numel (p))^2;
<   endif
<   p(abs (p) < small) = 0;
---
> %%%%%%%%%%% old_code %%%%%%%%%%%
> %%%   small = max (abs (p));
> %%%   if (isa (a, "single") || isa (b, "single"))
> %%%       small = max ([small, 1]) * eps ("single") * 1e4 * (1 + numel (p))^2;
> %%%   else
> %%%       small = max ([small, 1]) * eps * 1e4 * (1 + numel (p))^2;
> %%%   endif
> %%%   p(abs (p) < small) = 0;
> %%%%%%%%%%% new_code %%%%%%%%%%%
>   if G_DEBUG
>       fprintf(G_FID,'MODE =%s\n',MODE);
>       if (isa (a, "single") || isa (b, "single"))
>           fprintf(G_FID,'eps("single")=%+17.10e\n',eps("single"));
>       else
>           fprintf(G_FID,'eps =%+17.10e\n',eps);
>       end
>   endif
>   if (isa (a, "single") || isa (b, "single"))
>       eps_mod=eps("single");
>   else
>       eps_mod=eps;
>   endif

```

Sep 08, 11 5:31

diff_residue.txt

Page 2/6

```

> if strcmp(MODE,"MODE_S")
>
>         if G_DEBUG
>             fprintf(G_FID,"Eliminate small poles here using\n");
>             fprintf(G_FID,' eps_mod = %17.10e\n',eps_mod);
>             fprintf(G_FID,"small = max (abs (p));\n");
>             fprintf(G_FID,"small = max ([small,1])*eps*1e4*(1 + numel (p))^2;\n");
>         end
>
>         small = max (abs (p));
>         factor=(1 + numel (p))^2;
>
>         if G_DEBUG
>             fprintf(G_FID,' small_1   =%17.10e\n',small);
>         end
>         small = max ([small, 1]) * eps_mod * 1e4 * factor;
>         if G_DEBUG
>             fprintf(G_FID,' small_2   =%17.10e\n',small);
>             bg_print('p_1',    p);
>             fprintf(G_FID,'small =%17.10e\n',small);
>         end
>
>         p(abs (p) < small) = 0;
>
>         if G_DEBUG
>             bg_print('p_2',    p);
>         end
>     elseif strcmp(MODE,"MODE_Z")||strcmp(MODE,"MODE_Z2")
>         if G_DEBUG
>             fprintf(G_FID,"Eliminate small poles here using\n");
>             fprintf(G_FID,' eps_mod = %17.10e\n',eps_mod);
>         end
>
>         small = max (abs (1./p));
>         factor=(1 + numel (p))^2;
>
>         if G_DEBUG
>             fprintf(G_FID," small_1 = %17.10e\n",small );
>         end
>         small = max ([small, 1]) * eps_mod * 1e4 * factor;
>         if G_DEBUG
>             fprintf(G_FID," small_2 = %17.10e\n",small );
>         end
>
>         small=small/1000;
>
>         if G_DEBUG
>             fprintf(G_FID,' small   = %17.10e\n',small );
>             bg_print('p_1',1./p);
>         end
>
>         p(abs (1./p) < small) = 0;
>
>         if G_DEBUG
>             bg_print('p_2',1./p);
>         end
>
>         p = polyreduce (p);
>     else
>         fprintf("Error: MODE not equal to \"MODE_S\", \"MODE_Z\", or \"MODE_Z2\"\n");
>         exit;
>     end
>
>         if G_DEBUG
>             if strcmp(MODE,"MODE_S")
>                 bg_print('p',p);
>             elseif strcmp(MODE,"MODE_Z")
>                 bg_print('p',1./p);
>             else
>                 bg_print('P',p);
>             end
>         end
>     end
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
251a368,374
> %%%%%%%%%%% new_code %%%%%%%%%%%
>         Count=0;
>         if G_DEBUG
>             fprintf(G_FID,'----- ');
>             fprintf(G_FID,'Count=%i;\n',Count);
>         end
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
258c381,420
<     A(:,ip) = prepad (rresidue (ri, p, [], toler), border+1, 0).';
---
> %%%%%%%%%%% old_code %%%%%%%%%%%
> %%%     A(:,ip) = prepad (rresidue (ri, p, [], toler), border+1, 0).';
> %%%%%%%%%%% new_code %%%%%%%%%%%
>         if G_DEBUG
>             fprintf(G_FID,'Inputs to bg_rresidue are:\n');
>             fprintf(G_FID,'MODE =%s\n',MODE);
>             fprintf(G_FID,'toler =%17.10e\n',toler);
>             if strcmp(MODE,"MODE_S")
>                 bg_print('ri',ri);
>                 bg_print('p',p);
>                 fprintf(G_FID,'----- ');
>                 fprintf(G_FID,'Into A(:,ip)=prepad(bg_rresidue');
>                 fprintf(G_FID,'(ri,p,[],toler,[],MODE),border+1,0)\n');
>             elseif strcmp(MODE,"MODE_Z")||strcmp(MODE,"MODE_Z2")
>                 bg_print('RI',ri);
>                 bg_print('P',p);

```

Sep 08, 11 5:31

diff_residue.txt

Page 3/6

```

>                                     fprintf(G_FID,'----- ');
>                                     fprintf(G_FID,'Into A(:,IP)=prepad(bg_rresidue');
>                                     fprintf(G_FID,'(RI,P,[],toler,[],MODE),border+1,0)\n');
>                                     end
>                                     end
>     A(:,ip) = prepad (bg_rresidue (ri, p, [], toler,[], MODE), border+1, 0).';
>     Count++;
>     if G_DEBUG
>         if strcmp(MODE,"MODE_S")
>             fprintf(G_FID,'----- ');
>             fprintf(G_FID,'Out of A(:,ip)=prepad(bg_rresidue');
>             fprintf(G_FID,'(ri,p,[],toler,[],MODE),border+1,0)\n');
>         elseif strcmp(MODE,"MODE_Z") || strcmp(MODE,"MODE_Z2")
>             fprintf(G_FID,'----- ');
>             fprintf(G_FID,'Out of A(:,IP)=prepad(bg_rresidue');
>             fprintf(G_FID,'(RI,P,[],toler,[],MODE),border+1,0)\n');
>         end
>         fprintf(G_FID,'----- ');
>         fprintf(G_FID,'Count=%i;\n',Count);
>         fprintf(G_FID,'Type 1 Calculations for A gives:\n');
>         bg_print('A',A);
>         fprintf(G_FID,'This value of A is not used in the following.\n');
>     end
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
261a424,430
> %%%%%%%%%%% new_code %%%%%%%%%%%
>     A = cal_M(p,e);
>                                     if G_DEBUG
>                                     fprintf(G_FID,'Type 2 Calculation for A gives:\n');
>                                     bg_print('A',A);
>                                     end
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
267c436,440
< function [pnum, pden, e] = rresidue (r, p, k, toler, e)
---
> %%%%%%%%%%% old_code %%%%%%%%%%%
> %function [pnum, pden, e] = rresidue (r, p, k, toler, e)
> %%%%%%%%%%% new_code %%%%%%%%%%%
> function [pnum, pden, e] = bg_rresidue (r, p, k, toler, e, MODE)
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
272c445,452
<     if (nargin < 2 || nargin > 5)
---
> %%%%%%%%%%% old_code %%%%%%%%%%%
> %%% if (nargin < 2 || nargin > 5)
> %%%%%%%%%%% new_code %%%%%%%%%%%
>                                     global G_FID;
>                                     global G_DEBUG;
>                                     bg_enable_notes=1;
>
>     if (nargin < 2 || nargin > 6)
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
318a499,505
> %%%%%%%%%%% new_code %%%%%%%%%%%
>                                     if G_DEBUG
>                                     M = zeros(N,N); %% initialize matrix M for storing results
>                                     %M = zeros(D,D); %% initialize matrix M for storing results
>                                     bg_print('pden',pden);
>                                     end
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
328a516,520
> %%%%%%%%%%% new_code %%%%%%%%%%%
>                                     if G_DEBUG
>                                     for l = 1:length(pn) M(l,n)=pn(l);endfor %% save values for reference
>                                     end
> %%%%%%%%%%% end of new_code %%%%%%%%%%%
331a524,545
> %%%%%%%%%%% new_code %%%%%%%%%%%
>                                     if G_DEBUG
>                                     fprintf(G_FID,'Type 1 Calculations for M give:\n');
>                                     bg_print('M',M);
>                                     fprintf(G_FID,'Type 1 Calculations for pnum give:\n');
>                                     bg_print('pnum',pnum);
>                                     end
>
>     M = cal_M(p,e);
>                                     if G_DEBUG
>                                     fprintf(G_FID,'Type 2 Calculations for M give:\n');
>                                     bg_print('M',M);
>                                     end
>
>     B=M*r; %% This is a better way
>     pnum=transpose(B); %% to calculate pnum
>     add=zeros(1,N+1-numel(pnum));
>     pnum=cat(2,add,pnum);
>                                     if G_DEBUG
>                                     fprintf(G_FID,'Type 2 Calculations for pnum give:\n');
>                                     bg_print('pnum_1',pnum);
>                                     bg_print('pden_1',pden);

```

Sep 08, 11 5:31

diff_residue.txt

Page 4/6

```

>                                     end
> %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
340,347c554,628
<   if (isa (r, "single") || isa (p, "single") || isa (k, "single"))
<       small = max ([max(abs(pden)), max(abs(pnum)), 1]) * eps ("single");
<   else
<       small = max ([max(abs(pden)), max(abs(pnum)), 1]) * eps;
<   endif
<
<   pnum(abs (pnum) < small) = 0;
<   pden(abs (pden) < small) = 0;
---
> %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
> %%%% if (isa (r, "single") || isa (p, "single") || isa (k, "single"))
> %%%%     small = max ([max(abs(pden)), max(abs(pnum)), 1]) * eps ("single");
> %%%% else
> %%%%     small = max ([max(abs(pden)), max(abs(pnum)), 1]) * eps;
> %%%% endif
> %%%%
> %%%% pnum(abs (pnum) < small) = 0;
> %%%% pden(abs (pden) < small) = 0;
> %%%%
> %%%% pnum = polyreduce (pnum);
> %%%% pden = polyreduce (pden);
> %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
>   if (isa (r, "single") || isa (p, "single") || isa (k, "single"))
>       eps_mod = eps ("single");
>   else
>       eps_mod = eps;
>   endif
>   if strcmp(MODE,"MODE_S")
>
>       if G_DEBUG
>           fprintf(G_FID,'MODE   =%s\n',MODE);
>           fprintf(G_FID,'eps_mod   =%+17.10e\n',eps_mod);
>           fprintf(G_FID,"Reduce a and b here\n");
>       endif
>       small = max ([max(abs(pden)), max(abs(pnum)), 1]) * eps_mod/1000;
>       if G_DEBUG
>           fprintf(G_FID,'small =%+17.10e\n',small);
>       end
>       pnum(abs (pnum) < small) = 0;
>       pden(abs (pden) < small) = 0;
>       pnum = polyreduce (pnum);
>       pden = polyreduce (pden);
>   elseif strcmp(MODE,"MODE_Z") || strcmp(MODE,"MODE_Z2")
>       if G_DEBUG
>           fprintf(G_FID,'MODE   =%s\n',MODE);
>           fprintf(G_FID,'eps_mod=%+17.10e\n',eps_mod);
>       end
>       small = max(abs(pnum)) * eps_mod;
>       pnum(abs (pnum) < small) = 0;
>       if G_DEBUG
>           fprintf(G_FID,'Modify pnum using: small =%+17.10e\n',small);
>           bg_print('pnum_2',pnum);
>       end
>       if (isa (r, "single") || isa (p, "single") || isa (k, "single"))
>           large = 1/(min(abs(pden))) / eps ("single");
>       else
>           large = 1/(min(abs(pden))) / eps;
>       endif
>       if strcmp(MODE,"MODE_Z")
>           pden(abs (pden) > large) = 0;
>       elseif strcmp(MODE,"MODE_Z2")
>           pden(abs (pden) < small) = 0;
>       end
>       if G_DEBUG
>           if strcmp(MODE,"MODE_Z")
>               fprintf(G_FID,'Modify pden using: large =%+17.10e\n',large);
>               pden(abs (pden) > large) = 0;
>           elseif strcmp(MODE,"MODE_Z2")
>               fprintf(G_FID,'Modify pden using: small =%+17.10e\n',small);
>               pden(abs (pden) < small) = 0;
>           end
>           bg_print('pden_2',pden);
>           fprintf(G_FID,"Reduce polynominals giving:\n");
>       end
>       pnum = polyreduce (pnum);
>       pden = polyreduce (pden);
>   else
>       fprintf("Error: MODE not equal to \"MODE_S\", \"MODE_Z\", or \"MODE_Z2\\n\");
>       exit;
>   end
>
>       if G_DEBUG
>           bg_print('pnum',pnum);
>           bg_print('pden',pden);
>       end

```

Sep 08, 11 5:31

diff_residue.txt

Page 5/6

```

> %%%%%%%%% end of new_code %%%%%%%%%
349,350c630
<   pnum = polyreduce (pnum);
<   pden = polyreduce (pden);
---
> endfunction
351a632,682
> %%%%%%%%% new_code %%%%%%%%%
> function M = cal_M(p,e)
>   N = length(p); %% Define N as the number of poles
>   C = cell(N,1); %% Define cell C to contain N vectors
>   %%%%%%%%% PART 1: Calculation of the initial values for vectors C(2) through C(N)
>   x=[1 -p(1)];
>   if (e(2)==1)
>       C{2}=x;
>   else
>       C{2}=[1];
>   endif
>   for i=3:N
>       x=conv( x,[1 -p(i-1)]);
>       if (e(i)==1)
>           C{i}=x;
>       else
>           C{i}=C{i-(e(i)-1)};
>       endif
>   endfor
>   %%%%%%%%% PART 2: Calculation of the final values for vectors C(N) through C(1)
>   x=[1 -p(N)];
>   if (N>2)
>       C{N-1}=conv(C{N-1},x);
>   else
>       C{N-1}=x;
>   end
>   for i=N-2:-1:2;
>       x=conv(x,[1 -p(i+1)]);
>       C{i}=conv(C{i},x);
>   endfor
>   if (N>2)
>       x=conv(x,[1 -p(2)]);
>       if (e(1)>1)
>           x=deconv(x,[1 -p(N)]);
>       end
>       C{1}=x;
>   end
>   %%%%%%%%% PART 3: Calculation of matrix M using vectors C(1) through C(N)
>   M = zeros(N,N);
>   for j=1:N
>       x=C{j};
>       len_x=length(x);
>       offset=N-len_x;
>       for i=1:offset
>           M(i,j)=0;
>       endfor
>       for i=1:N-offset
>           M(i+offset,j)=x(i);
>       endfor
>   endfor
>   %%%%%%%%% End of Calculation of matrix M
352a684
> %%%%%%%%% end of new_code %%%%%%%%%
357c689,693
<   [r, p, k, e] = residue (b, a);
---
> %%----- old_code -----
> %%                                     [r, p, k, e] = residue (b, a);
> %%----- new_code -----
> [r, p, k, e] = bg_residue (b, a);
> %%----- end of new_code -----
365c701,705
<   [br, ar] = residue (r, p, k);
---
> %%----- old_code -----
> %%                                     [br, ar] = residue (r, p, k);
> %%----- new_code -----
> [br, ar] = bg_residue (r, p, k);
> %%----- end of new_code -----
368c708,712
<   [br, ar] = residue (r, p, k, e);
---
> %%----- old_code -----
> %%                                     [br, ar] = residue (r, p, k, e);
> %%----- new_code -----
> [br, ar] = bg_residue (r, p, k, e);
> %%----- end of new_code -----
375c719,723
<   [r, p, k, e] = residue (b, a);

```

Sep 08, 11 5:31

diff_residue.txt

Page 6/6

```

---
> %%----- old_code -----
> %%                                     %! [r, p, k, e] = residue (b, a);
> %%----- new_code -----
> %! [r, p, k, e] = bg_residue (b, a);
> %%----- end of new_code -----
381c729,733
< %! [br, ar] = residue (r, p, k);
---
> %%----- old_code -----
> %%                                     %! [br, ar] = residue (r, p, k);
> %%----- new_code -----
> %! [br, ar] = bg_residue (r, p, k);
> %%----- end of new_code -----
390c742,746
< %! [b, a] = residue (r, p, k, e);
---
> %%----- old_code -----
> %%                                     %! [b, a] = residue (r, p, k, e);
> %%----- new_code -----
> %! [b, a] = bg_residue (r, p, k, e);
> %%----- end of new_code -----
393c749,753
< %! [rr, pr, kr, er] = residue (b, a);
---
> %%----- old_code -----
> %%                                     %! [rr, pr, kr, er] = residue (b, a);
> %%----- new_code -----
> %! [rr, pr, kr, er] = bg_residue (b, a);
> %%----- end of new_code -----
403c763,767
< %! [r, p, k, e] = residue (b, a);
---
> %%----- old_code -----
> %%                                     %! [r, p, k, e] = residue (b, a);
> %%----- new_code -----
> %! [r, p, k, e] = bg_residue (b, a);
> %%----- end of new_code -----
409c773,777
< %! [br, ar] = residue (r, p, k);
---
> %%----- old_code -----
> %%                                     %! [br, ar] = residue (r, p, k);
> %%----- new_code -----
> %! [br, ar] = bg_residue (r, p, k);
> %%----- end of new_code -----
411a780,798
> %%----- new_code -----
> %!test
> %! z1=+7.0372976777e+06;
> %! p1=-3.1415926536e+09;
> %! p2=-4.9964813512e+08;
> %! r1=-(1+z1/p1)/(1-p1/p2)/p2/p1 ;
> %! r2=-(1+z1/p2)/(1-p2/p1)/p2/p1 ;
> %! r3=(1+(p2+p1)/p2/p1*z1)/p2/p1 ;
> %! r4=z1/p2/p1 ;
> %! r=[r1;r2;r3;r4];
> %! p=[p1;p2;0;0];
> %! k=[];
> %! e=[1;1;1;2];
> %! b=[1,z1];
> %! a=[1,-(p1+p2),p1*p2,0,0];
> %! [br, ar] = bg_residue (r, p, k, e);
> %! assert ((abs (br - b) < 1e-8
> %!         && abs (ar - a) < 1e-8));
> %%----- end of new_code -----

```