**SUSTech DSP HW1 2(d)**

图示

描述已自动生成

*本文使用OneHotkey进行公式编辑*

[*RUSRUSHB/OneHotkey: Simplify formula inputs in OneNote & Word, e.g., \a for \alpha. 简化OneNote和Word中的数学公式输入，例如\a代表\alpha*](https://github.com/RUSRUSHB/OneHotkey)

**方法一：时域卷积**

命题：

若：

则有：

证明：

----------------------------------------------------------

此题中：

**方法二： Mason 增益公式**

图示

描述已自动生成

环路增益：

前向路径增益：

系统总行列式：

**方法三：矩阵直接求解**

增广两个自由度（使得总自由度为，这其实和卷积有关）：

要找到一组 ：

**路线一：可暴力求解：**

syms d1 d2 tx0 tx1 tx2 ty0 ty1 ty2

X = [1,d1,d2,0,0;0,1,d1,d2,0;0,0,1,d1,d2]

X =

黑白色的标志

描述已自动生成

Y = [d2,d1,1,0,0;0,d2,d1,1,0;0,0,d2,d1,1]

Y =

文本

描述已自动生成

Tx = [tx0,tx1,tx2];

Ty = [1,ty1,ty2]; % ty0 set to 1

eq = Tx\*X == Ty\*Y;

vars = [tx0,tx1,tx2,ty1,ty2]

vars = 

sol = solve(eq,vars)

sol = 包含以下字段的 struct:

tx0: d2

tx1: d1

tx2: 1

ty1: d1

ty2: d2

**路线二：也可考虑对称性：**文本

描述已自动生成

**关于路线一暴力求解**：下面有个代码可以跑一跑试一下：

clear

syms a0 a1 a2 a3 a4 a5 b0 b1 b2 b3

x=[a0,a1,a2,a3,a4,a5];

y=[b0,b1,b2,b3];

x\_start = 0;

y\_start = 1; % y[n]中h项从h[n-1]开始

[solX, solY]=relaSolve(x, y, x\_start, y\_start)

X =

文本

描述已自动生成

Y =

文本

中度可信度描述已自动生成

solX = 包含以下字段的 struct:

tx0: 0

tx1: b0

tx2: b1

tx3: b2

tx4: b3

solY = 包含以下字段的 struct:

ty0: a0

ty1: a1

ty2: a2

ty3: a3

ty4: a4

ty5: a5

checkRela(solX, solY, x, y, x\_start, y\_start)

LHS =

文本

描述已自动生成

RHS =

文本

描述已自动生成

Check that they are the same:

LHS-RHS=

蓝色的标志

描述已自动生成

function [solX, solY] = relaSolve(x, y, x\_start, y\_start)

if nargin == 2

x\_start = 0; y\_start = 0;

end

x0\_stored = x(1);

idx\_start = min(x\_start, y\_start);

x = [zeros(1, x\_start-idx\_start), x];

y = [zeros(1, y\_start-idx\_start), y];

padded\_len = length(x) + length(y) - 1;

x\_pad = padded\_len - length(x);

y\_pad = padded\_len - length(y);

x = [x, zeros(1, x\_pad)];

y = [y, zeros(1, y\_pad)];

X = sym(zeros(x\_pad+1, padded\_len));

Y = sym(zeros(y\_pad+1, padded\_len));

for k = 1:x\_pad+1

X(k, :) = circshift(x, k-1);

end

for k =1:y\_pad+1

Y(k, :) = circshift(y, k-1);

end

X

Y

Tx = arrayfun(@(i) sym(['tx', num2str(i)]), 0:x\_pad);

Ty = arrayfun(@(i) sym(['ty', num2str(i)]), 0:y\_pad);

% 生成sym

Ty(1)=1; % set to 1 as our wish

eq = Tx\*X == Ty\*Y;

vars = [Tx(:); Ty(:)];

vars = vars(vars ~= 1);

sol = solve(eq,vars);

sol.ty0 = 1;

fields = fieldnames(sol);

idx\_ty1 = find(strcmp(fields, 'ty1'));

new\_fields = [fields(1:idx\_ty1-1); {'ty0'}; fields(idx\_ty1:end-1)];

sol = orderfields(sol, new\_fields);

fields = fieldnames(sol);

% 遍历所有字段，并将每个字段的值乘以x0\_stored以归一化

for i = 1:numel(fields)

field = fields{i};

sol.(field) = sol.(field) \* x0\_stored;

end

idx\_split = find(strcmp(fieldnames(sol), 'ty0'));

fields1 = fields(1:idx\_split-1);

fields2 = fields(idx\_split:end);

solX = rmfield(sol, fields2);

solY = rmfield(sol, fields1);

end

function checkRela(solX, solY, x, y, x\_start, y\_start)

if nargin == 4

x\_start = 0; y\_start = 0;

end

Tx=struct2cell(solX);

Ty=struct2cell(solY);

LHS=Tx\*([zeros(1,x\_start),x])

RHS=(Ty\*([zeros(1,y\_start),y])).'

disp('Check that they are the same:')

disp('LHS-RHS=')

disp(LHS-RHS)

end