深蓝学院

shenlonxueyuon.com

第四章作业分享

主讲人 tkalpha

纲要

- 第一部分：计算推导
- 第二部分：ROS作业
- 第三部分：问题与解答

计算推导

深蓝学院 shenlanxueyuan.com

- 求解过程：基本与课件中的步骤一致

- 优化的目标函数
  - $J=\int_0^T g(x,u)dt=\int_0^T(1+u^TRu)dt=\int_0^T(1+a_x^2+a_y^2+a_z^2)dt$
- 状态转移方程
  - $$x=\begin{bmatrix}p_x\\p_y\\p_z\\v_x\\v_y\\v_z\end{bmatrix},u=\begin{bmatrix}a_x\\a_y\\a_z\end{bmatrix},\dot{x}=f(x,u)=\begin{bmatrix}v_x\\v_y\\v_z\\a_x\\a_y\\a_z\end{bmatrix}$$

- 求解
  - 协状态
    $\lambda=\begin{bmatrix}\lambda_1 & \lambda_2 & \lambda_3 & \lambda_4 & \lambda_5 & \lambda_6\end{bmatrix}^T$
  - 哈密顿方程
    $H(x,u,\lambda)=g(x,u)+\lambda^Tf(x,u)$
    $H(x,u,\lambda)=(1+a_x^2+a_y^2+a_z^2)+\lambda^Tf(x,u)$

tkalpha

计算推导

深蓝学院
shenlanxueyuan.com

- 求解过程2

- 求解λ

$$\dot{\lambda} = -\nabla H(x^*, u^*, \lambda) = \begin{bmatrix} 0 & 0 & 0 & -\lambda_1 & -\lambda_2 & -\lambda_3 \end{bmatrix}^T$$

$$\lambda = \begin{bmatrix} 2\alpha_1 \\ 2\alpha_2 \\ 2\alpha_3 \\ -2\alpha_1 t - 2\beta_1 \\ -2\alpha_2 t - 2\beta_2 \\ -2\alpha_3 t - 2\beta_3 \end{bmatrix}$$

- 理想输入

$$u^* = arg\ \min_u H(x^*, u, \lambda) = \begin{bmatrix} \alpha_1 t + \beta_1 \\ \alpha_2 t + \beta_2 \\ \alpha_3 t + \beta_3 \end{bmatrix}$$

- 理想状态

$$\dot{x}^* = f(x^*, u^*)$$

$$x^* = \begin{bmatrix} \frac{1}{6}\alpha_1 t^3 + \frac{1}{2}\beta_1 t^2 + v_{x0} t + p_{x0} t \\ \frac{1}{6}\alpha_2 t^3 + \frac{1}{2}\beta_2 t^2 + v_{y0} t + p_{y0} t \\ \frac{1}{6}\alpha_3 t^3 + \frac{1}{2}\beta_3 t^2 + v_{z0} t + p_{z0} t \\ \frac{1}{2}\alpha_1 t^2 + \beta_1 t + v_{x0} \\ \frac{1}{2}\alpha_2 t^2 + \beta_2 t + v_{y0} \\ \frac{1}{2}\alpha_3 t^2 + \beta_3 t + v_{z0} \end{bmatrix}$$

$$x^*(0) = x(0) = \begin{bmatrix} p_{x0} & p_{y0} & p_{z0} & v_{x0} & v_{y0} & v_{z0} \end{bmatrix}^T$$

计算推导

求解过程3

- 根据末尾状态求解α和β

$$x^*(T) == \begin{bmatrix} p_{xf} & p_{yf} & p_{zf} & v_{xf} & v_{yf} & v_{zf} \end{bmatrix}^T$$

$$记 \begin{bmatrix} \Delta p_x \\ \Delta p_y \\ \Delta p_z \\ \Delta v_x \\ \Delta v_y \\ \Delta v_z \end{bmatrix} = \begin{bmatrix} p_{xf} - v_{x0}T - p_{x0} \\ p_{yf} - v_{y0}T - p_{y0} \\ p_{zf} - v_{z0}T - p_{z0} \\ v_{xf} - v_{x0} \\ v_{yf} - v_{y0} \\ v_{zf} - v_{z0} \end{bmatrix}$$

将α和β作为方程组的自变量

$$\begin{bmatrix} \frac{1}{6}T^3 & 0 & 0 & \frac{1}{2}T^2 & 0 & 0 \\ 0 & \frac{1}{6}T^3 & 0 & 0 & \frac{1}{2}T^2 & 0 \\ 0 & 0 & \frac{1}{6}T^3 & 0 & 0 & \frac{1}{2}T^2 \\ \frac{1}{2}T^2 & 0 & 0 & T & 0 & 0 \\ 0 & \frac{1}{2}T^2 & 0 & 0 & T & 0 \\ 0 & 0 & \frac{1}{2}T^2 & 0 & 0 & T \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} \Delta p_x \\ \Delta p_y \\ \Delta p_z \\ \Delta v_x \\ \Delta v_y \\ \Delta v_z \end{bmatrix}$$

α和β完全可以表示为初始状态、结束状态与T的函数

$$\begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \beta_1 \\ \beta_2 \\ \beta_3 \end{bmatrix} = \begin{bmatrix} -\frac{12}{T^3} & 0 & 0 & \frac{6}{T^2} & 0 & 0 \\ 0 & -\frac{12}{T^3} & 0 & 0 & \frac{6}{T^2} & 0 \\ 0 & 0 & -\frac{12}{T^3} & 0 & 0 & \frac{6}{T^2} \\ \frac{6}{T^2} & 0 & 0 & -\frac{2}{T} & 0 & 0 \\ 0 & \frac{6}{T^2} & 0 & 0 & -\frac{2}{T} & 0 \\ 0 & 0 & \frac{6}{T^2} & 0 & 0 & -\frac{2}{T} \end{bmatrix} \begin{bmatrix} \Delta p_x \\ \Delta p_y \\ \Delta p_z \\ \Delta v_x \\ \Delta v_y \\ \Delta v_z \end{bmatrix}$$

计算推导

求解过程4

- 基于目标函数求得最优的T

$$J = \int_0^T (1 + a_x^2 + a_y^2 + a_z^2)dt$$
$$= T + (\tfrac{1}{3}\alpha_1^2 T^3 + \alpha_1\beta_1 T^2 + \beta_1^2 T) + (\tfrac{1}{3}\alpha_2^2 T^3 + \alpha_2\beta_2 T^2 + \beta_2^2 T) + (\tfrac{1}{3}\alpha_3^2 T^3 + \alpha_3\beta_3 T^2 + \beta_3^2 T)$$

代入α和β的T表示方法可得

$$J = T + \tfrac{4}{T}((v_{xf}^2 + v_{xf}v_{x0} + v_{x0}^2) + (v_{yf}^2 + v_{yf}v_{y0} + v_{y0}^2) + (v_{zf}^2 + v_{zf}v_{z0} + v_{z0}^2)) - \tfrac{12}{T^2}((p_{xf} - p_{x0})(v_{xf} + v_{x0}) + (p_{yf} - p_{y0})(v_{yf} + v_{y0}) + (p_{zf} - p_{z0})(v_{zf} + v_{z0})) + \tfrac{12}{T^3}((p_{xf} - p_{x0})^2 + (p_{yf} - p_{y0})^2 + (p_{zf} - p_{z0})^2)$$

然后求$\frac{dJ}{dT} = 0$时T符合要求的根

$$T^4 - 4T^2((v_{xf}^2 + v_{xf}v_{x0} + v_{x0}^2) + (v_{yf}^2 + v_{yf}v_{y0} + v_{y0}^2) + (v_{zf}^2 + v_{zf}v_{z0} + v_{z0}^2)) + 24T((p_{xf} - p_{x0})(v_{xf} + v_{x0}) + (p_{yf} - p_{y0})(v_{yf} + v_{y0}) + (p_{zf} - p_{z0})(v_{zf} + v_{z0})) - 36((p_{xf} - p_{x0})^2 + (p_{yf} - p_{y0})^2 + (p_{zf} - p_{z0})^2) = 0$$

计算推导

深蓝学院 shenlanxueyuan.com

- 求解过程技巧

复杂的推导不需要手算，可以借助MATLAB或者Mathematica的数值计算功能完成。

Sym.m

```
syms T Pxf Vxf Vx0 Px0;
DPx = Pxf-Vx0*T-Px0;
DVx = Vxf-Vx0;
A1 = simplify(-12/T^3*DPx+6/T^2*DVx);
B1 = simplify(6/T^2*DPx-2/T*DVx);
Jx1 = simplify(A1^2*T^3/3+A1*B1*T^2+B1^2*T)
```

命令行窗口

```
>> Sym

Jx1 =

(4*(3*Px0^2 - 6*Px0*Pxf + 3*Px0*T*Vx0 + 3*Px0*T*Vxf + 3*Pxf^2 - 3*Pxf*T*Vx0 - 3*Pxf*T*Vxf + T^
```

tkalpha

纲要

- 第一部分：计算推导
- 第二部分：ROS作业
- 第三部分：问题与解答

ROS实现

- [PART I] 轨迹库的生成

直接基于恒加速运动的性质，或者基于数值积分方法（改进欧拉法/龙格库塔法）与状态方程可以推导得到

- 状态转移方程

$$x = \begin{bmatrix} p_x \\ p_y \\ p_z \\ v_x \\ v_y \\ v_z \end{bmatrix}, u = \begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix}, \dot{x} = f(x, u) = \begin{bmatrix} v_x \\ v_y \\ v_z \\ a_x \\ a_y \\ a_z \end{bmatrix}$$

$$v_1 = v_0 + a_0 t$$

$$p_1 = p_0 + v_0 t + 0.5 a_0 t^2$$

```
pos(0) = pos(0) + vel(0)*delta_time + 0.5*acc_input(0)*delta_time*delta_time;
pos(1) = pos(1) + vel(1)*delta_time + 0.5*acc_input(1)*delta_time*delta_time;
pos(2) = pos(2) + vel(2)*delta_time + 0.5*acc_input(2)*delta_time*delta_time;

vel(0) = vel(0) + acc_input(0)*delta_time;
vel(1) = vel(1) + acc_input(1)*delta_time;
vel(2) = vel(2) + acc_input(2)*delta_time;
```

ROS实现

深蓝学院 shenlanxueyuan.com

●［PART II］OBVP求解

1. 核心是求解多项式。作业里使用伴随矩阵法实现。

- 对于多项式$P(x) = x^n + c_{n-1}x^{n-1} + c_{n-2}x^{n-2} + \cdots + c_1 x + c_0$
- 其伴随矩阵为

$$M_x = \begin{bmatrix} 0 & 0 & \cdots & 0 & -c_0 \\ 1 & 0 & \cdots & 0 & -c_1 \\ 0 & 1 & \cdots & 0 & -c_2 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & -c_{n-1} \end{bmatrix}$$

- 伴随矩阵的特征值就是多项式的根

2.除此之外还可以使用费拉里法显式求解，参考代码

3. 助教作业思路的课件里面还有提到Eigen::PolynomialSolver, ZJU-FAST-Lab/Root-Finder等求解库

tkalpha

ROS实现

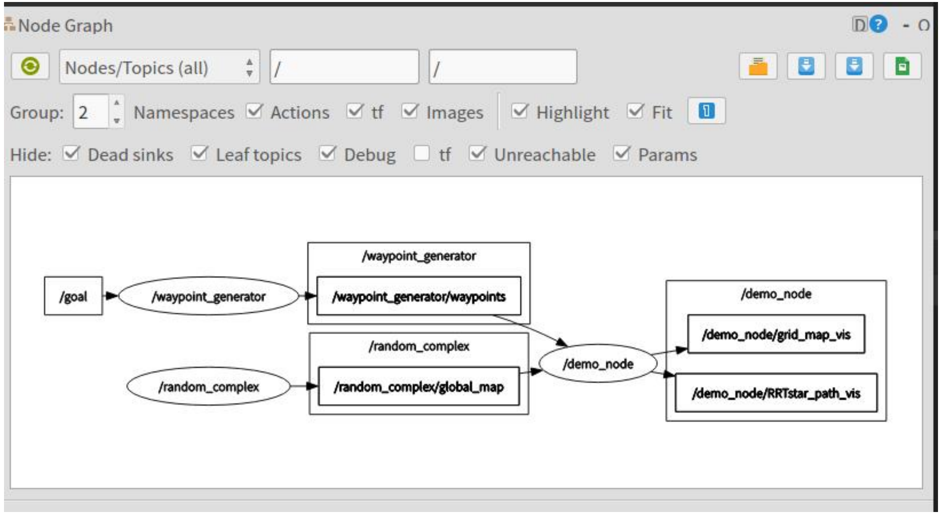
```cpp
double Homeworktool::OptimalBVP(Eigen::Vector3d _start_position,Eigen::Vector3d _start_velocity,Eigen::Vector3d _target_position)
{
    double optimal_cost = 100000;
    // coefficient calculation
    double j_a1 = 4.0*(pow(_start_velocity(0),2)+pow(_start_velocity(1),2)+pow(_start_velocity(2),2));
    double j_a2 = -12.0*((_target_position(0)-_start_position(0))*_start_velocity(0)
                        +(_target_position(1)-_start_position(1))*_start_velocity(1)
                        +(_target_position(2)-_start_position(2))*_start_velocity(2));
    double j_a3 = 12.0*(pow((_target_position(0)-_start_position(0)),2)
                       +pow((_target_position(1)-_start_position(1)),2)
                       +pow((_target_position(2)-_start_position(2)),2));

    Matrix4d t_comp = Matrix4d::Zero(4,4); // matrix generation
    t_comp(1,0) = 1.0; t_comp(2,1) = 1.0; t_comp(3,2) = 1.0;
    t_comp(2,3) = j_a1;
    t_comp(1,3) = 2.0*j_a2;
    t_comp(0,3) = 3.0*j_a3;

    Matrix<complex<double>,Dynamic,Dynamic> t_comp_eig = t_comp.eigenvalues();

    int t_num=t_comp_eig.rows(), kk=0, flag=0; // find a positive real root
    for(; kk<t_num; kk++){
        if(fabs(t_comp_eig(kk,0).imag()-0.0)<1e-8 && t_comp_eig(kk,0).real()>0){
            flag = 1;
            break;
        }
    }
    if(flag==1){ // optimal T and cost calculation
        double t =  t_comp_eig(kk,0).real();
        double j = t + j_a1/t + j_a2/pow(t,2) + j_a3/pow(t,3);
        optimal_cost = j;
    }
    return optimal_cost;
}
```

代码实现

1.生成伴随矩阵
2.选择合适的根
3.求最优的T,cost

tkalpha

ROS实现

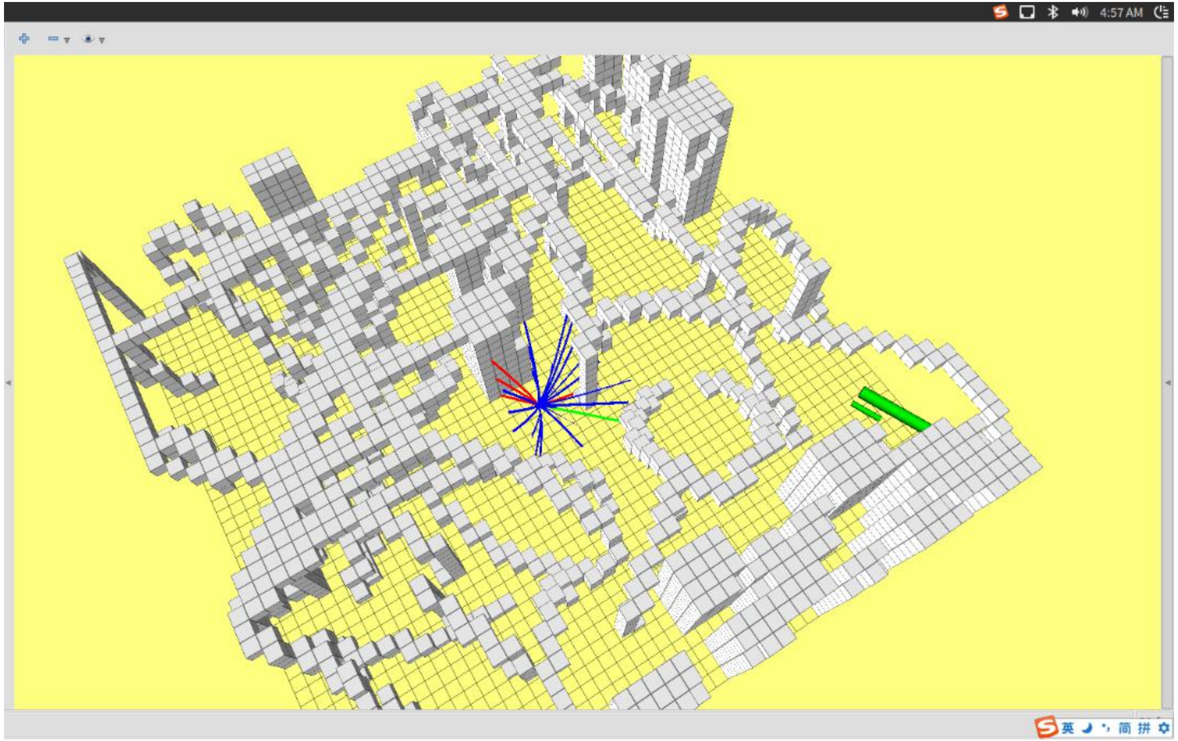


ROS rqt_graph

```
process[rviz-5]: started with pid [3064]
[ INFO] [1629533994.671714908]: [node] receive the planning target
[ INFO] [1629534020.121975636]: [node] receive the planning target
[ INFO] [1629534024.461538702]: [node] receive the planning target
[ INFO] [1629534202.101361454]: [node] receive the planning target
```

ROS_INFO

ROS实现



最终效果

ROS rviz

tkalpha

在线问答 － 其他

深蓝学院 shenlanxueyuan.com

- 群里有关于边界条件符号的讨论

1. 公式推导

深蓝学院 shenlanxueyuan.com

Define the Hamiltonian function:

$$H(s,u,\lambda)=\frac{1}{T}j^2+\lambda^T f_s(s,u)$$
$$=\frac{1}{T}j^2+\lambda_1 v+\lambda_2 a+\lambda_3 j$$

- 由Pontryagin's minimum principle:

$$\dot{\lambda}(t)=-\nabla_s H(s^*(t),u^*(t),\lambda(t))$$

$$\lambda(t)=\frac{1}{T}\begin{bmatrix}-2\alpha\\ 2\alpha t+2\beta\\ -\alpha t^2-2\beta t-2\gamma\end{bmatrix}$$

- 使用边界条件:

$$\lambda(T)=\cancel{-}+\nabla h(s^*(T))$$

- 由于末状态v,a自由，因此h的表达式与T时刻的v和a并不相关
- 因此:

$$\lambda_2(T)=0,\lambda_3(T)=0$$

- 因此 表达式可以化简为:

$$\begin{cases}\beta=-\alpha T\\ \gamma=\frac{\alpha}{2}T^2\end{cases}\qquad \lambda(t)=\frac{1}{T}\begin{bmatrix}-2\alpha\\ 2\alpha(t-T)\\ -\alpha t^2+2\alpha Tt-\alpha T^2\end{bmatrix}$$

Optimal Boundary Value Problem (OBVP)

- Previous slides are about fixed final state problem.
- How about the final state is (partially)-free?
  - Did you notice where is the boundary condition?

$$\lambda(t)=-\nabla h(s^*(t))$$

For fixed final state problem:

$$h(s(T))=\begin{cases}0, & if\ \ s=s(T)\\ \infty, & otherwise\end{cases}$$ Not differentiable

So we discard this condition, and use given $x(T)$ to directly solve for unknown variables

$$s^*(t)=\begin{bmatrix}\frac{\alpha}{120}t^5+\frac{\beta}{24}t^4+\frac{\gamma}{6}t^3+\frac{a_0}{2}t^2+v_0t+p_0\\ \frac{\alpha}{24}t^4+\frac{\beta}{6}t^3+\frac{\gamma}{2}t^2+a_0t+v_0\\ \frac{\alpha}{6}t^3+\frac{\beta}{2}t^2+\gamma t+a_0\end{bmatrix}$$

For (partially)-free final state problem:

$$given\ \ s_i(T),\ i\in I$$

We have boundary condition for other costate:

$$\lambda_j(T)=\frac{\partial h(s^*(T))}{\partial s_j},\ \ for\ j\neq i$$

Then we solve this problem again.

31

群友 @mengchaoheng：

根据相关参考文献，左下角的课件PPT才是对的。不过不影响作业的结果。

$$p_N=\nabla g_N.$$

We will assume that the constraint sets U_k are convex, so that we can apply the optimality condition

$$\sum_{k=0}^{N-1}\nabla_{u_k}J(u_0^*,\ldots,u_{N-1}^*)'(u_k-u_k^*)\geq 0,$$

for all feasible $(u_0,\ldots,u_{N-1})$ (see Appendix B). This condition can be decomposed into the N conditions

$$\nabla_{u_k}J(u_0^*,\ldots,u_{N-1}^*)'(u_k-u_k^*)\geq 0,\qquad \text{for all } u_k\in U_k,\ k=0,\ldots,N-1. \tag{3.42}$$

We thus obtain:

3.4 EXTENSIONS OF THE MINIMUM PRINCIPLE

We now consider some variations of the continuous-time optimal control problem and derive corresponding variations of the Minimum Principle.

3.4.1 Fixed Terminal State

Suppose that in addition to the initial state $x(0)$, the final state $x(T)$ is given. Then the preceding informal derivations still hold except that the terminal condition $J^*(T,x)=h(x)$ is not true anymore. In effect, here we have

$$J^*(T,x)=\begin{cases}0 & \text{if } x=x(T),\\ \infty & \text{otherwise.}\end{cases}$$

Thus $J^*(T,x)$ cannot be differentiated with respect to x, and the terminal boundary condition $p(T)=\nabla h(x^*(T))$ for the adjoint equation does not hold. However, as compensation, we have the extra condition

$$x(T):\text{given},$$

thus maintaining the balance between boundary conditions and unknowns.

If only *some* of the terminal states are fixed, that is,

$$x_i(T):\text{given},\qquad \text{for all } i\in I,$$

where I is some index set, we have the partial boundary condition

$$p_j(T)=\frac{\partial h(x^*(T))}{\partial x_j},\qquad \text{for all } j\notin I,$$

for the adjoint equation.

Example 3.4.1

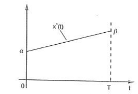


Figure 3.4.1 Optimal solution of the problem of connecting the two points $(0,\alpha)$ and (T,β) with a minimum length curve (cf. Example 3.4.1).

Example 3.4.2 (The Brachistochrone Problem)

In 1696 Johann Bernoulli challenged the mathematical world ...

$$\dot{x}=f(x,u,t)\qquad (n \text{ differential equations}), \tag{2.5.4}$$

$$\dot{\lambda}=-\left(\frac{\partial f}{\partial x}\right)^T\lambda-\left(\frac{\partial L}{\partial x}\right)^T\qquad (n \text{ differential equations}), \tag{2.5.5}$$

$$\left(\frac{\partial H}{\partial u}\right)^T=\left(\frac{\partial f}{\partial u}\right)^T\lambda+\left(\frac{\partial L}{\partial u}\right)^T=0\qquad (m \text{ algebraic equations}), \tag{2.5.6}$$

$$x_k(t_o)\text{ given or }\lambda_k(t_o)=0\ \ k=1,\ldots,n\qquad (n \text{ boundary conditions}), \tag{2.5.7}$$

$$\lambda^T(t_f)=\left(\frac{\partial\phi}{\partial x}+\nu^T\frac{\partial\psi}{\partial x}\right)_{t=t_f}\qquad (n \text{ boundary conditions}), \tag{2.5.8}$$

$$\psi[x(t_f),t_f]=0\qquad (q \text{ side conditions}). \tag{2.5.9}$$

The stationarity conditions (2.5.6) determine the m-vector $u(t)$. The $2n$ differential equations (2.5.4) and (2.5.5), with the $2n$ boundary conditions (2.5.7) and (2.5.8), form a two-point boundary-value problem with q parameters ν to be found in (2.5.8) so that the q side conditions (2.5.9) are satisfied.

tkalpha

在线问答

深蓝学院
shenlanxueyuan.com

Q&A

tkalpha

深蓝学院
shenlanxueyuan.com

感谢各位聆听！

Thanks for Listening