

cryo_tank 模型方程与 ODE 求解逻辑

当前版本采用方案二：ODE 直接积分气枕温度 T_{ull} ，氦气质量 m_{He} 和补气流量 $\dot{m}_{He}$ 由恒压纯氦状态关系派生。

1. 模型假设

- 液相区为液氮，气枕区为纯氦气。
- 气枕压力固定为工作压力 P_{work} 。
- 不考虑氮气蒸发、冷凝和气液相间质量传递。
- m_{He} 不是独立状态量，而是维持恒压所需的派生量。

代码位置：tank_model.py:98-214

2. 状态向量

$$\mathbf{y} = [m_{liq}, U_{liq}, T_{ull}]^T$$

- m_{liq} ：液氮质量 [kg]
- U_{liq} ：液相总内能 [J]
- T_{ull} ：气枕区温度 [K]

代码位置：tank_model.py:1-7, 87-89

3. 初始状态方程

$$V_{liq,0} = (1 - f_{ull})V_{total}$$

$$V_{ull,0} = f_{ull}V_{total}$$

$$\rho_{liq,0} = \rho_{LN2}(T_{init}, P_{work})$$

$$m_{liq,0} = \rho_{liq,0}V_{liq,0}$$

$$U_{liq,0} = m_{liq,0}u_{LN2}(T_{init}, P_{work})$$

$$T_{ull,0} = T_{init}$$

$$m_{He,0} = \rho_{He}(T_{init}, P_{work})V_{ull,0}$$

代码位置：tank_model.py:71-85

4. 几何方程

$$A_{cross} = \frac{V_{total}}{H_{tank}}$$

$$D = \sqrt{\frac{4A_{cross}}{\pi}}$$

$$A_{side} = \pi D H_{tank}$$

$$A_{cap} = A_{cross}$$

$$A_{total} = A_{side} + 2A_{cap}$$

$$A_{wet} = A_{cap} + \pi D level$$

$$A_{dry} = A_{cap} + \pi D (H_{tank} - level)$$

其中 level 被限制在 0 到 H_tank 之间。

代码位置 : tank_model.py:41-48, 91-96

5. 派生量方程

5.1 液相派生量

$$u_{liq} = \frac{U_{liq}}{m_{liq}}$$

$$T_{liq} = T_{LN2}(u_{liq}, P_{work})$$

$$\rho_{liq} = \rho_{LN2}(T_{liq}, P_{work})$$

$$V_{liq} = \frac{m_{liq}}{\rho_{liq}}$$

$$level = \frac{V_{liq}}{A_{cross}}$$

$$fill = \frac{level}{H_{tank}}$$

代码位置 : tank_model.py:107-113

5.2 气枕派生量

$$V_{ull} = V_{total} - V_{liq}$$

$$P_{He} = P_{work}$$

$$\rho_{He} = \rho_{He}(T_{ull}, P_{He})$$

$$m_{He} = \rho_{He} V_{ull}$$

$$U_{ull} = m_{He} u_{He}(T_{ull}, P_{He})$$

代码位置 : tank_model.py:115-121

6. 换热方程

$$Q_{liq \rightarrow ull} = h_{conv} A_{cross} (T_{liq} - T_{ull})$$

正值表示热量从液相传给气枕。

$$Q_{leak} = A_{total} q_{mli}$$

默认 MLI 漏热模型。

$$T_{mean} = \frac{T_{inner} + T_{env}}{2}$$

$$Q_{leak} = \frac{A_{total} k_{eff} (T_{mean}) (T_{env} - T_{inner})}{\delta}$$

代码中提供的 Foam 漏热模型。

$$Q_{leak, liq} = Q_{leak} \frac{A_{wet}}{A_{wet} + A_{dry}}$$

$$Q_{leak, ull} = Q_{leak} \frac{A_{dry}}{A_{wet} + A_{dry}}$$

代码位置 : tank_model.py:141-148; heat_leak.py:21-70

7. 液氮质量和液相能量方程

$$\frac{dm_{liq}}{dt} = \dot{m}_{in, LN2} - \dot{m}_{out, LN2}$$

$$h_{in, LN2} = h_{LN2}(T_{in, LN2}, P_{work})$$

$$h_{liq} = h_{LN2}(T_{liq}, P_{work})$$

$$\frac{dU_{liq}}{dt} = \dot{m}_{in, LN2} h_{in, LN2} - \dot{m}_{out, LN2} h_{liq} - Q_{liq \rightarrow ull} + Q_{leak, liq}$$

代码位置 : tank_model.py:60-65, 150-156

8. 气枕温度和氦气质量导数

$$m_{He} = \rho_{He}(T_{ull}, P_{work}) V_{ull}$$

$$\frac{dV_{ull}}{dt} = -\frac{\dot{m}_{liq}}{\rho_{liq}}$$

$$Q_{ull, no He} = Q_{liq \rightarrow ull} + Q_{leak, ull}$$

8.1 恒压氦气偏导

$$\rho = \rho_{He}(T_{ull}, P_{work})$$

$$u = u_{He}(T_{ull}, P_{work})$$

$$\rho_T = \left(\frac{\partial \rho}{\partial T}\right)_P$$

$$u_T = \left(\frac{\partial u}{\partial T}\right)_P$$

$$m_T = V_{ull}\rho_T$$

$$m_V = \rho$$

$$U_T = V_{ull}(u\rho_T + \rho u_T)$$

$$U_V = \rho u$$

代码位置 : properties.py:100-109; tank_model.py:165-178

8.2 气枕温度变化率

$$h_{in, He} = h_{He}(T_{in, He}, P_{work})$$

$$\frac{dT_{ull}}{dt} = \frac{Q_{ull, no\ He} - (U_V + P_{work} - h_{in, He}m_V)\frac{dV_{ull}}{dt}}{U_T - h_{in, He}m_T}$$

$$\frac{dm_{He}}{dt} = \dot{m}_{He}$$

$$\dot{m}_{He} = m_T\frac{dT_{ull}}{dt} + m_V\frac{dV_{ull}}{dt}$$

代码位置 : tank_model.py:180-214

9. 完整 ODE 方程组

$$\mathbf{y} = [m_{liq}, U_{liq}, T_{ull}]^T$$

$$\frac{dm_{liq}}{dt} = \dot{m}_{in, LN2} - \dot{m}_{out, LN2}$$

$$\frac{dU_{liq}}{dt} = \dot{m}_{in, LN2}h_{in, LN2} - \dot{m}_{out, LN2}h_{liq} - Q_{liq \rightarrow ull} + Q_{leak, liq}$$

$$\frac{dT_{ull}}{dt} = \frac{Q_{ull, no\ He} - (U_V + P_{work} - h_{in, He}m_V)\frac{dV_{ull}}{dt}}{U_T - h_{in, He}m_T}$$

m_{He} 、 U_{ull} 和 $\dot{m}_{He}$ 是派生量，不是独立 ODE 状态。

代码位置 : tank_model.py:131-214

10. ODE 求解逻辑

$$y_0 = \text{tank.initial_state}()$$

求解器调用 `scipy.integrate.solve_ivp`，右端函数为 `tank.rhs`，积分区间为 `[0, t_end]`，方法为 RK45，最大步长默认为 10 s。

$$\frac{dy}{dt} = f(t, \mathbf{y}) = \text{tank.rhs}(t, \mathbf{y})$$

10.1 液体排空事件

$$g(t, \mathbf{y}) = m_{liq}$$

事件属性：`terminal = True`，`direction = -1`。当液氮质量从正值下降到 0 时终止求解。

代码位置：`solver.py:14-64`

11. 后处理逻辑

求解完成后，`solver.run()` 对每个输出时刻重新执行 `tank.derive(y_i)`，并重新计算换热量和氦气补气流量。

$$\text{info}_i = \text{derive}(\mathbf{y}_i)$$

$$Q_{liq \rightarrow ull}, Q_{leak}, Q_{leak, liq}, Q_{leak, ull}, \dot{m}_{He}$$

`m_He`、`U_ull`、`mdot_He` 均在后处理阶段作为派生结果保存到 `history`。

代码位置：`solver.py:70-124`