Transient Thermal-Hydraulic Analysis of an Overcooling Accident in a Pressurized Water Reactor*

Jeongjin Han
School of Computing & Nuclear and Quantum Engineering
Korea Advanced Institute of Science and Technology (KAIST)
hjj22@kaist.ac.kr

Contents

*Submitted for NQE625: Computational Analysis in Nuclear Systems, KAIST, Fall 2025.

1 Governing Equations

This section summarizes the governing equations used in the MATLAB system code. The modeled system consists of two primary-side channels (hot/average) and a steam generator primary–secondary heat exchange path. One-dimensional, lumped-parameter/finite-volume formulations are employed for mass, momentum, and energy balances. The numerical discretization details (upwind/implicit time marching) are described in Section 2 (Numerical methods).

1.1 Loop momentum balance for total mass flow rate (natural circulation / pump-driven flow)

The total loop mass flow rate is governed by a one-dimensional integral momentum balance. The left-hand side represents the inertia of the circulating fluid, while the right-hand side includes pump head, friction loss, form loss, and gravitational (buoyancy) head.

$$\sum_k \frac{L_k}{A_k}\frac{d\dot{m}}{dt} = \Delta P_{pump} - \Delta P_{fric} - \Delta P_{form} + \Delta P_{grav} \tag{1}$$

where k denotes each hydraulic segment in the loop, L_k and A_k are the segment length and flow area, and $\dot{m}$ is the total mass flow rate. ΔP_{pump} is the pump head contribution (set to zero after pump trip), ΔP_{fric} and ΔP_{form} are frictional and minor (form) losses, and ΔP_{grav} is the net buoyancy head due to density differences along elevation changes. In the present simulation, this equation is activated after the trip to describe natural circulation behavior.

1.2 One-dimensional energy conservation for fluid enthalpy transport (convection with volumetric heat source)

The fluid energy equation is written in terms of specific enthalpy h with one-dimensional axial convection and a heat source term representing wall-to-fluid heat transfer.

$$\rho\frac{\partial h}{\partial t} + G\frac{\partial h}{\partial l} = \frac{A_h q''}{V}$$

where ρ is the fluid density, G is the mass flux, and l is the axial coordinate. A_h is the heat transfer area associated with the control volume V, and q'' is the wall heat flux (positive when heat is transferred to the fluid).

Integrating over a control volume and applying a finite-volume (lumped) approximation yields:

$$\int_V \left(\rho\frac{\partial h}{\partial t} + G\frac{\partial h}{\partial l}\right) dV = \int_V \frac{A_h q''}{V}\, dV \tag{2}$$

$$\rho A_f \Delta l\frac{\partial h}{\partial t} + G A_f (h_{out} - h_{in}) = A_h q'' \tag{3}$$

where A_f is the flow area and Δl is the control-volume length. The convection term is expressed using inlet/outlet enthalpies (h_{in}, h_{out}). The specific discretization choice for the convection term (e.g., first-order upwind) is described in the Numerical methods section.

1.3 Reactivity model (scram insertion + temperature feedback)

The reactivity is modeled as a superposition of (i) scram reactivity insertion after trip and (ii) linear temperature feedback from coolant and fuel temperatures.

$$\rho(t) = \rho_{scram}(t - t_{trip}) + \alpha_w(T_w - T_{w0}) + \alpha_f(T_f - T_{f0})$$

where $\rho_{scram}(t - t_{trip})$ represents the time-dependent scram worth after the reactor trip time t_{trip}, and α_w, α_f are the moderator/coolant and fuel temperature coefficients, respectively. T_w and T_f are

volume-averaged coolant and fuel temperatures, and (T_{w0}, T_{f0}) are reference temperatures at steady state.

$$T_{w0} = \frac{\int_{V_b^{hot}} T_b^{hot}\, dV + \int_{V_b^{avg}} T_b^{avg}\, dV}{V_b^{hot} + V_b^{avg}} \tag{4}$$

$$T_{f0} = \frac{\int_{V_f^{hot}} T_f^{hot}\, dV + \int_{V_f^{avg}} T_f^{avg}\, dV}{V_f^{hot} + V_f^{avg}} \tag{5}$$

Here, $T_b^{hot/avg}$ denotes the bulk coolant temperature in the hot/average channel volumes $V_b^{hot/avg}$, and $T_f^{hot/avg}$ denotes the representative fuel temperature in volumes $V_f^{hot/avg}$.

1.4 Point kinetics with one delayed neutron group

Reactor power is computed using the point kinetics equations with one effective delayed neutron precursor group.

$$\begin{cases} \dot{P} = \dfrac{\rho - \beta}{\Lambda} P + \lambda C \\ \dot{C} = \dfrac{\beta}{\Lambda} P - \lambda C \end{cases} \tag{6}$$

where P is the reactor thermal power (proportional to neutron population), C is the delayed neutron precursor concentration, Λ is the prompt neutron generation time, β is the effective delayed neutron fraction, and λ is the precursor decay constant. The time integration scheme used to update (P, C) is described in the Numerical methods section.

1.5 Pressure drop models (friction, form loss, and gravity head)

The segment-wise pressure losses are evaluated using conventional friction, form-loss, and gravity head terms.

$$\Delta P_{\text{fric}} = f \cdot \frac{L}{D} \frac{G^2}{2\rho}$$

$$\Delta P_{\text{form}} = K_{\text{SE}} \cdot \frac{G^2}{2\rho} \quad \text{or} \quad K_{\text{SC}} \cdot \frac{G^2}{2\rho}$$

$$K_{\text{SE}} = 0.42 \cdot \left(1 - \frac{A_1}{A_2}\right)^2, \quad K_{\text{SC}} = 1 - \frac{A_1}{A_2} \qquad (A_1 < A_2)$$

$$\Delta P_{\text{grav}} = \rho g \Delta h$$

where f is the Darcy friction factor, L is the length, D is the hydraulic diameter, K is the minor-loss coefficient, and Δh is the elevation change. For sudden expansion/contraction, the coefficients $(K_{\text{SE}}, K_{\text{SC}})$ are used as given above.

In the present implementation, the form-loss term uses the local mass flux at the downstream side of the junction (e.g., G_{out}) to ensure consistent momentum loss accounting.

1.6 Fuel/conductor heat conduction with volumetric heat generation (finite-volume form)

The solid temperature field (fuel/cladding) is obtained from a transient heat conduction equation with volumetric heat generation. Heat flux is modeled by Fourier's law with temperature-dependent conductivity $k(T)$.

$$-\nabla \cdot \mathbf{q} + q'''(z) = \rho_s C_{p,s} \frac{\partial T}{\partial t},\ \mathbf{q} = -k(T)\nabla T \Rightarrow\ \nabla \cdot \big(k(T)\nabla T\big) + q'''(z) \quad = \rho_s C_{p,s} \frac{\partial T}{\partial t} \tag{7}$$

Integrating Eq. (7) over a control volume V_i gives the finite-volume (integral) form:

$$\int_{V_i} \rho_s C_{p,s} \frac{\partial T}{\partial t}\, dV = \int_{\partial V_i} k(T)\nabla T \cdot \hat{\mathbf{n}}\, dA + \int_{V_i} q'''(z)\, dV \tag{8}$$

where $\hat{\mathbf{n}}$ is the outward unit normal vector on the control-volume surface.

1.7 Flow distribution model between hot and average channels (pressure-drop matching)

In the two-channel core model, the hot and average channels share the same inlet and outlet plena. Therefore, the pressure drop across the core must be identical for both channels, while satisfying a global mass conservation constraint. For a given total mass flux (or total mass flow rate), the channel-wise mass fluxes (G_{hot}, G_{avg}) are determined by enforcing the pressure-drop matching condition.

$$\begin{cases} \Delta P_{core} = \Delta P_{core}^{hot} = \Delta P_{core}^{avg} \\ \Delta P_{core}^{hot}(G_{hot}) = \Delta P_{LP}^{hot} + \Delta P_{fp}^{hot} + \Delta P_{UP} \\ \Delta P_{core}^{avg}(G_{avg}) = \Delta P_{LP}^{avg} + \Delta P_{fp}^{avg} + \Delta P_{UP} \\ (N_{hot} + N_{avg})\, G_{tot} = N_{hot}\, G_{hot} + N_{avg}\, G_{avg} \end{cases}$$

where ΔP_{LP} and ΔP_{UP} denote the pressure losses in the lower and upper plena, respectively, and ΔP_{fp} denotes the pressure loss in the fuel-pin channel region (including friction, form loss, and gravity head contributions as defined in the previous subsection). N_{hot} and N_{avg} represent the number of parallel subchannels (or rod bundles) corresponding to the hot and average channels. G_{tot} is the given total (area-averaged) mass flux of the core, and (G_{hot}, G_{avg}) are the unknown channel-wise mass fluxes.

In the MATLAB implementation, the mass conservation constraint provides G_{avg} as a function of G_{hot}. Then, G_{hot} is solved by finding the root of the pressure-drop difference function $\Delta P_{core}^{hot}(G_{hot}) - \Delta P_{core}^{avg}(G_{avg}(G_{hot})) = 0$. This procedure yields a consistent flow distribution that equalizes the core pressure drop between the two channels.

1.8 Steam generator node-wise heat flux relation (primary–secondary coupling)

The steam generator (SG) heat transfer is evaluated node-wise by coupling the primary- and secondary-side bulk temperatures through the tube-wall conduction resistance and the convective/boiling heat transfer resistances. Depending on the secondary-side thermodynamic state, two different closures are applied for the wall heat flux q''.

$$T_b^{SG,p} - T_b^{SG,s} = \left(\frac{\Delta x}{K_{ss}} + \frac{1}{h_{DB,p}}\right) q'' + \left(\frac{25^4 q'' \cdot 1^{-6}}{e^{\frac{4P}{6.2}}}\right)^{\frac{1}{4}} \tag{9}$$

$$T_b^{SG,p} - T_b^{SG,s} = \left(\frac{\Delta x}{K_{ss}} + \frac{1}{h_{DB,p}} + \frac{1}{h_{DB,s}}\right) q'' \tag{10}$$

where $T_b^{SG,p}$ and $T_b^{SG,s}$ are the primary- and secondary-side bulk temperatures at a given SG node, respectively. The term $\Delta x/K_{ss}$ represents the tube wall conduction resistance. $h_{DB,p}$ denotes the primary-side convective heat transfer coefficient evaluated by the Dittus–Boelter correlation. The additional nonlinear term in the first relation corresponds to a nucleate boiling wall superheat correlation (Jens–Lottes type) used on the secondary side. In Eq. (9), the factor 10^{-6} is introduced to convert the heat flux unit from $\mathrm{MW/m^2}$, as used in the original Jens–Lottes correlation, to $\mathrm{W/m^2}$ for consistency with the present energy balance formulation.

Here, the quality refers to the secondary-side equilibrium quality at the SG node. When $0 \leq x < 1$, the secondary side is treated as a two-phase boiling condition and the heat flux is obtained by combining: (i) tube-wall conduction, (ii) primary-side forced convection (Dittus–Boelter), and (iii) a secondary-side boiling closure (Jens–Lottes type). When $x > 1$, the secondary side is treated as a superheated vapor condition and forced convection is applied on both sides; i.e., the Dittus–Boelter correlation is used for both $h_{DB,p}$ and $h_{DB,s}$.

In the present implementation, x remains mostly within $0 \leq x \leq 1$ for the transient, and the boiling-mode relation is therefore used for most time steps. The fixed-point iteration procedure used to obtain consistent $(q'',\, T^{SG,s})$ at each time step is described in Section 4.

2 Numerical Methods

2.1 Time step selection based on CFL-type stability limits

$$t^{\omega+1} - t^{\omega} = \Delta t^{\omega} \tag{11}$$

The global time step Δt^{ω} is selected to ensure stability of the explicit first-order upwind update used for the 1-D enthalpy transport equation (see Eq. (15)). Since the system is advanced sector-by-sector (hot channel, average channel, SG primary, and SG secondary), a sector-wise upper bound is first computed and then the minimum value is adopted as the global bound:

$$\Delta t^{\mathbb{UB},\omega} = \min\left(\Delta t^{hot,\omega},\ \Delta t^{avg,\omega},\ \Delta t^{SG,p,\omega},\ \Delta t^{SG,s,\omega}\right), \tag{12}$$

$$\Delta t^{\omega} = 0.8\,\Delta t^{\mathbb{UB},\omega}. \tag{13}$$

Here, each sector bound $\Delta t^{(\cdot),\omega}$ is determined from a CFL-type condition associated with the upwind convection discretization (i.e., requiring the local Courant number to remain in the stable range $0 < C < 1$). In the present implementation, the upwind update is applied independently to each sector rather than to the full closed loop, and the first node in each sector is advanced without a source term while source terms are applied from the second node onward (see Section 4 for the detailed algorithmic flow).

2.2 Numerical update of total mass flow rate for natural circulation

The loop momentum balance given in Section 1.1 (Eq. (1)) is discretized in time to update the total mass flow rate after reactor trip. Using a first-order finite-difference approximation for the inertial term, the governing equation is written as

$$\sum_{k} \frac{L_k}{A_k} \frac{\dot{m}^{\omega+1} - \dot{m}^{\omega}}{\Delta t^{\omega}} = \Delta P_{\text{pump}} - \Delta P_{\text{fric}} - \Delta P_{\text{form}} + \Delta P_{\text{grav}}. \tag{14}$$

Before the reactor trip, the loop is assumed to be in a steady forced-circulation state, such that the pressure balance satisfies

$$\Delta P_{\text{pump}} - \Delta P_{\text{fric}} - \Delta P_{\text{form}} + \Delta P_{\text{grav}} = 0,$$

and the total mass flow rate remains constant. Once the trip occurs, the pump is assumed to stop instantaneously, leading to $\Delta P_{\text{pump}} = 0$. The flow evolution is then governed by the imbalance between frictional/form losses and the buoyancy head, and a new equilibrium corresponding to natural circulation is established.

In Eq. (14), the left-hand side represents the inertial response of the loop, preventing an instantaneous change of the mass flow rate. The gravitational head ΔP_{grav} is treated as a positive driving term, while friction and form losses are always dissipative. Accordingly, the pressure terms are assembled with explicit signs to reflect their physical roles rather than using absolute values.

2.3 Numerical discretization of fluid enthalpy transport (upwind scheme)

The one-dimensional fluid energy conservation equation introduced in Section 1.2 (Eq. (3)) is discretized in time and space using a first-order upwind scheme for the convection term. This choice ensures numerical stability under strong advection-dominated conditions encountered during the transient.

Applying a backward (upwind) differencing for the axial enthalpy gradient and an explicit time integration yields the following update formula for the i-th control volume:

$$
\begin{aligned}
h_i^{\omega+1} &= h_i^{\omega} - \frac{G\,\Delta t^{\omega}}{\rho\,\Delta l}\left(h_i^{\omega} - h_{i-1}^{\omega}\right) + \frac{A_h\,q''\,\Delta t^{\omega}}{\rho\,A_f\,\Delta l} \\
&= h_i^{\omega} - \frac{\dot{m}\,\Delta t^{\omega}}{\rho\,V}\left(h_i^{\omega} - h_{i-1}^{\omega}\right) + \frac{A_h\,q''\,\Delta t^{\omega}}{\rho\,V},
\end{aligned}
\tag{15}
$$

where ω denotes the time index, G is the mass flux, $\dot{m}$ is the mass flow rate, ρ is the fluid density, Δl is the axial length of the control volume, V is the control volume, A_f is the flow area, and A_h is the heat transfer area. The first term on the right-hand side represents axial convection, while the second term accounts for wall-to-fluid heat addition through the heat flux q''.

The upwind scheme is applied independently to four flow paths in the system: the hot channel, the average channel, the steam generator primary side, and the steam generator secondary side. Since the formulation is not loop-based, the first node of each flow path does not include a source term contribution. For the hot and average channels, an energy scaling is applied such that the total heat generation is consistently distributed to downstream nodes. In contrast, for the steam generator primary and secondary sides, the inlet enthalpy is prescribed and the heat source term naturally adjusts through the coupled heat flux calculation, so no additional scaling is required.

The time step Δt^{ω} is constrained by the Courant stability condition associated with the upwind discretization. Specifically, the requirement $0 < C < 1$ for the Courant number determines an upper bound on Δt^{ω} for each flow path, and the final time step is selected as described in Eqs. (12) and (13).

2.4 Numerical evaluation of reactivity feedback terms

The reactivity ρ^{ω} at each time step is evaluated using the temperature feedback model introduced in Section 1.3. In the numerical implementation, the coolant and fuel temperatures appearing in the feedback terms are obtained as volume-averaged quantities over discretized control volumes.

The reference temperatures T_{w0} and T_{f0} are defined by Eqs. (4) and (5), respectively, using the steady-state solution. During the transient, the instantaneous volume-averaged temperatures T_w^{ω} and T_f^{ω} are computed by summing the contributions from each discretized control volume (CX) in the hot and average channels.

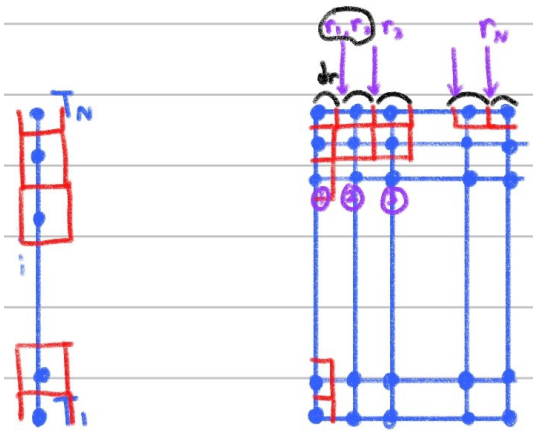


Figure 1: Schematic illustration of volume discretization used for the numerical evaluation of coolant and fuel temperatures. The red dashed lines indicate control-volume (CX) boundaries over which temperature fields are integrated to obtain volume-averaged values for reactivity feedback.

As shown in Fig. 1, both the coolant and fuel regions are discretized into a finite number of control volumes. Within each control volume, temperature is assumed to be spatially uniform, and the volume-averaged temperature is obtained by numerical integration. This discretized treatment enables

the reactivity feedback terms to be evaluated consistently with the finite-volume formulation adopted for the thermal-hydraulic and heat conduction equations.

Using these numerically evaluated temperatures, the reactivity at time level ω is updated explicitly as ρ^{ω} and subsequently used in the point kinetics update described in Section 1.4.

2.5 Semi-implicit time integration of point kinetics equations

The point kinetics equations with one delayed neutron group (Eq. (6)) are integrated in time using a semi-implicit scheme. This approach improves numerical stability while maintaining a relatively simple update structure.

$$\begin{cases} \dfrac{P^{\omega+1}-P^{\omega}}{\Delta t^{\omega}} = \dfrac{\rho^{\omega}-\beta}{\Lambda}P^{\omega+1} + \lambda C^{\omega}, \\ \dfrac{C^{\omega+1}-C^{\omega}}{\Delta t^{\omega}} = \dfrac{\beta}{\Lambda}P^{\omega} - \lambda C^{\omega+1}, \end{cases} \tag{16}$$

where P is the reactor thermal power, C is the delayed neutron precursor concentration, ρ^{ω} is the reactivity evaluated at the current time level, β is the effective delayed neutron fraction, Λ is the prompt neutron generation time, and λ is the precursor decay constant.

In this semi-implicit formulation, the power P is treated implicitly in the prompt neutron term, while the precursor concentration C is treated implicitly in the decay term. All remaining terms are evaluated explicitly at time level ω. As a result, the coupled system can be solved algebraically at each time step without iterative coupling between $P^{\omega+1}$ and $C^{\omega+1}$.

This semi-implicit treatment relaxes the time step restriction associated with the stiff prompt neutron dynamics. Consequently, the global time step Δt^{ω} can be determined solely based on the Courant stability condition of the thermal-hydraulic upwind scheme, as described in Section 2.3, without introducing an additional kinetics-driven upper bound.

2.6 Implicit finite-volume discretization for solid heat conduction

The transient heat conduction equation in the solid region (fuel and cladding), introduced in Section 1.6, is discretized in space using the finite-volume method and integrated in time using a fully implicit scheme.

Starting from the finite-volume integral form given by Eq. (8), the energy balance is written for each axial control volume i. The transient term is evaluated using volume-averaged material properties, while the conductive heat fluxes at the left and right control-volume faces are linearized with respect to the unknown temperature at the new time level $\omega+1$.

After spatial discretization and time integration over Δt^{ω}, the following implicit update equation is obtained:

$$\frac{\int_V \rho C_p dV}{\Delta z}\frac{(T_i^{\omega+1}-T_i^{\omega})}{\Delta t^{\omega}} = -G_{L,i}(T_i^{\omega+1}-T_{i-1}^{\omega+1}) + G_{R,i}(T_{i+1}^{\omega+1}-T_i^{\omega+1}) + q'''(z)A_i$$

$$\frac{\bar{\rho}\overline{C_p}A_i\Delta z}{\Delta z}\frac{(T_i^{\omega+1}-T_i^{\omega})}{\Delta t^{\omega}} = -G_{L,i}(T_i^{\omega+1}-T_{i-1}^{\omega+1}) + G_{R,i}(T_{i+1}^{\omega+1}-T_i^{\omega+1}) + q'''(z)A_i$$

$$T_i^{\omega+1}-T_i^{\omega} = -\widetilde{G}_{L,i}(T_i^{\omega+1}-T_{i-1}^{\omega+1}) + \widetilde{G}_{R,i}(T_{i+1}^{\omega+1}-T_i^{\omega+1}) + q'''(z)\widetilde{A}_i$$

$$\text{where} \quad \widetilde{G}_{L,i} = \frac{G_{L,i}\Delta t^{\omega}}{\bar{A}_V}, \quad \widetilde{G}_{R,i} = \frac{G_{R,i}\Delta t^{\omega}}{\bar{A}_V}, \quad \widetilde{A}_i = \frac{A_i\Delta t^{\omega}}{\bar{A}_V} \quad (\bar{A}_V = \bar{\rho}\overline{C_p}A_i)$$

Here, $G_{L,i}$ and $G_{R,i}$ denote the effective thermal conductances at the left and right faces of control volume i, respectively. The quantities $\widetilde{G}_{L,i}$ and $\widetilde{G}_{R,i}$ are the corresponding dimensionless conduc-

tance coefficients, scaled by the time step and the volumetric heat capacity $\bar{A}_V = \bar{\rho}\,\overline{C_p}\,A_i$. The volumetric heat generation term $q'''(z)$ is treated implicitly through the source coefficient $\widetilde{A}_i$.

Rearranging the discretized equation yields a tri-diagonal linear system for the unknown temperatures at time level $\omega + 1$:

$$(1 + \widetilde{G}_{L,i} + \widetilde{G}_{R,i})T_i^{\omega+1} - \widetilde{G}_{L,i}T_{i-1}^{\omega+1} - \widetilde{G}_{R,i}T_{i+1}^{\omega+1} = T_i^{\omega} + q'''(z)\widetilde{A}_i \tag{17}$$

The resulting linear system is solved at each time step to obtain the updated solid temperature distribution. An implicit scheme is adopted to avoid stability restrictions on the time step size that would otherwise arise from the conduction term, allowing the global time step Δt^{ω} to be determined solely by the convective energy transport constraints.

2.7 Fixed-point iteration for steam generator primary–secondary coupling

The steam generator (SG) is treated as a node-wise coupled heat exchanger between the primary side (SG primary) and the secondary side (SG secondary). At each global time step ω, the primary-side enthalpy/temperature field is advanced using the upwind energy update, while the secondary-side temperature field is obtained by enforcing consistency of the coupled heat flux relation introduced in Section 1.8. In other words, T_{SG}^{p} is updated by time marching, whereas T_{SG}^{s} is obtained through an inner iteration so that the computed heat flux q'' and the secondary-side state become mutually consistent.

At time level ω, the SG coupling is evaluated by a fixed-point iteration (inner loop) as follows:

1. **Freeze the secondary-side temperature field.** Initialize (or take from the previous global time step) the secondary-side bulk temperature at each SG node, $T_{b,i}^{SG,s,(\ell)}$, where ℓ is the inner-iteration index.
2. **Compute the node-wise temperature difference and wall heat flux.** Using the current primary-side bulk temperature $T_{b,i}^{SG,p,\omega}$ and the frozen secondary-side temperature $T_{b,i}^{SG,s,(\ell)}$, evaluate the wall heat flux $q_i''^{,(\ell)}$ from the appropriate closure: Eq. (9) for $0 \le x_i < 1$ and Eq. (10) for $x_i > 1$. Here, the quality x_i is evaluated on the secondary side at node i using the updated secondary-side enthalpy estimate.
3. **Update SG primary-side enthalpy using the computed heat flux.** The primary-side energy source term is constructed from $q_i''^{,(\ell)}$ and applied to the upwind enthalpy update for the primary-side SG nodes to obtain $h_i^{SG,p,(\ell+1)}$ (and the corresponding $T_{b,i}^{SG,p,(\ell+1)}$ if needed for property evaluation).
4. **Update SG secondary-side enthalpy with opposite flow direction.** The same wall heat flux is applied to the secondary side with opposite sign (heat gain on the secondary side) and with reversed node ordering to reflect counter-current flow. This yields the updated secondary-side enthalpy $h_i^{SG,s,(\ell+1)}$ and the updated temperature $T_{b,i}^{SG,s,(\ell+1)}$.
5. **Convergence check and termination.** The inner loop continues until the secondary-side temperature field converges, e.g., $\|T_b^{SG,s,(\ell+1)} - T_b^{SG,s,(\ell)}\|_\infty / \max(1, \|T_b^{SG,s,(\ell)}\|_\infty) < \varepsilon$. Once converged, the final q_i'' and updated SG primary/secondary states are accepted for the global time step $\omega \to \omega + 1$.

This fixed-point procedure ensures energy consistency between the SG primary and secondary sides at every time step, while allowing the global time step Δt^{ω} to remain governed by the convective stability constraints used in the loop energy update.

2.8 Numerical evaluation of temperature from enthalpy

In the numerical scheme, the primary and secondary fluid temperatures are not advanced directly. Instead, the specific enthalpy h is first updated using the energy conservation equations, and the corresponding temperature is then recovered from the updated enthalpy and pressure. Because

the thermodynamic relationship between enthalpy and temperature is highly nonlinear and regime-dependent, different numerical treatments are applied depending on the thermodynamic state of the fluid.

Basic assumption

In the subcooled liquid region, the enthalpy is assumed to be primarily governed by temperature rather than pressure. Accordingly, the subcooled enthalpy is approximated as a function of temperature alone, and compressibility effects are neglected. This assumption is consistent with the relatively weak pressure dependence of liquid enthalpy compared to temperature dependence under the present operating conditions.

Region-wise numerical treatment

(1) Subcooled region ($h \leq h_f^{sat}$) When the updated enthalpy satisfies $h \leq h_f^{sat}$, the fluid is treated as subcooled liquid. The temperature is obtained by solving the nonlinear equation

$$f(T) = h_f(T) - h_{\text{target}} = 0$$

using the secant method. The secant update is written as

$$T_{\text{new}} = T_1 - f(T_1)\,\frac{T_1 - T_0}{f(T_1) - f(T_0)},$$

where (T_0, T_1) are the temperatures from the two previous iterations. Unlike the Newton–Raphson method, this approach does not require evaluation of the derivative dh/dT, and thus avoids numerical instability associated with property table differentiation. Convergence is assessed using dual tolerances on enthalpy ($\text{tol}_h = 10^{-3}$) and temperature ($\text{tol}_T = 10^{-8}$).

(2) Two-phase region ($h_f^{sat} < h < h_g^{sat}$) When the enthalpy lies between the saturated liquid and saturated vapor values, the fluid is treated as a two-phase mixture. In this regime, the temperature is determined algebraically as the saturation temperature at the local pressure,

$$T = T_{sat}(P).$$

Because phase change occurs at constant temperature, no iterative numerical procedure is required. The corresponding vapor quality is obtained directly from the enthalpy balance.

(3) Superheated vapor region ($h \geq h_g^{sat}$) When the enthalpy exceeds the saturated vapor enthalpy, the fluid is treated as superheated vapor. The temperature is approximated using first-order linear extrapolation based on the vapor specific heat,

$$T = T_{sat} + \frac{h_{\text{target}} - h_g^{sat}}{C_{p,g}}.$$

To prevent numerical divergence under rapidly varying conditions, the vapor specific heat $C_{p,g}$ is bounded within a prescribed range ($1500 \leq C_{p,g} \leq 3500$ J/kg-K).

Summary of algorithm

The overall temperature recovery procedure can be summarized as follows:

1. Retrieve h_f^{sat}, h_g^{sat}, and T_{sat} at the current pressure from the thermophysical property tables.
2. Determine the thermodynamic region of the updated enthalpy h_{target}.
3. Apply the corresponding numerical method (secant, algebraic assignment, or linear extrapolation) to obtain $T^{\omega+1}$.
4. Use the computed temperature as the initial value for the next time step.

3 Constitutive relations

3.1 Heat transfer correlations

Heat transfer coefficients (HTCs) used in the present analysis are evaluated using standard empirical correlations, depending on the flow regime and thermodynamic state of each region. Different correlations are applied for the core fuel channels and the steam generator (SG) primary and secondary sides.

3.1.1 Single-phase forced convection: Dittus–Boelter correlation

For single-phase turbulent flow in smooth tubes, the Dittus–Boelter correlation is employed to evaluate the Nusselt number:

$$\begin{cases} Nu = 0.023\, Re^{0.8} Pr^{0.4}, & \text{Heating } (T_w > T_b) \\ Nu = 0.023\, Re^{0.8} Pr^{0.3}, & \text{Cooling } (T_w < T_b) \end{cases} \tag{18}$$

The corresponding convective heat transfer coefficient is obtained from $h = \dfrac{Nu\, k}{D_h}$, where k is the thermal conductivity of the fluid and D_h is the hydraulic diameter.

The Dittus–Boelter correlation is applicable for fully developed turbulent flow under the conditions $Re \geq 10^4$, $0.7 \leq Pr \leq 160$, and sufficiently large axial aspect ratios ($L/D \geq 60$). All thermophysical properties are evaluated at the arithmetic mean bulk temperature.

In the present model, the Dittus–Boelter correlation is applied as follows:

- **Core fuel channels (hot and average):** heating condition ($T_w > T_b$).
- **Steam generator primary side:** cooling condition ($T_w < T_b$).
- **Steam generator secondary side (superheated region):** heating condition.

3.1.2 Nucleate boiling heat transfer: Jens–Lottes correlation

In the present study, the steam generator secondary-side thermodynamic state is characterized using the *equilibrium quality*, denoted by x_e. When the equilibrium quality satisfies $0 \leq x_e < 1$, nucleate boiling is assumed and the wall heat flux is evaluated using the Jens–Lottes correlation:

$$q'' = \frac{\exp(4P/6.2)}{25^4} \left(T_w - T_{\mathrm{sat}}\right)^4 \tag{19}$$

where q'' is the wall heat flux [$\mathrm{MW/m^2}$], P is the pressure [MPa], T_w is the wall temperature, and T_{sat} is the saturation temperature corresponding to the local pressure.

This correlation relates the wall superheat to the boiling heat flux and is widely used for subcooled and saturated nucleate boiling conditions in water-cooled reactor applications.

Accordingly, the steam generator secondary side is treated as:

- **Two-phase region ($0 \leq x_e < 1$):** Jens–Lottes correlation.
- **Superheated vapor region ($x_e > 1$):** Dittus–Boelter correlation with heating condition.

3.2 Friction factor correlations

The frictional pressure loss in each flow segment is evaluated using standard correlations for the Darcy friction factor, depending on the local flow regime characterized by the Reynolds number. Laminar and turbulent flow regimes are treated separately.

3.2.1 Laminar flow regime

For laminar flow conditions ($Re < 2100$), the Darcy friction factor is evaluated using the analytical solution for fully developed flow in a circular tube:

$$f = \frac{64}{Re} \tag{20}$$

This expression is applied to all regions where the local Reynolds number falls below the laminar–turbulent transition threshold.

3.2.2 Turbulent flow regime: Colebrook correlation

For turbulent flow conditions ($Re \geq 2100$), the friction factor is evaluated using the Colebrook equation, which implicitly relates the Darcy friction factor to the Reynolds number and relative roughness:

$$\frac{1}{\sqrt{f}} = -2.0 \log_{10} \left(\frac{\varepsilon / D_h}{3.7} + \frac{2.51}{Re\sqrt{f}} \right) \tag{21}$$

where ε is the absolute surface roughness and D_h is the hydraulic diameter. The Colebrook equation is solved iteratively to obtain the friction factor at each axial node.

3.2.3 Reynolds number evaluation

The Reynolds number is evaluated locally at each axial node using the mass-flux-based definition:

$$Re = \frac{G\, D_h}{\mu} \tag{22}$$

where G is the local mass flux, D_h is the hydraulic diameter, and μ is the dynamic viscosity of the fluid. The hydraulic diameter is used consistently for both core channels and steam generator flow paths.

Thermophysical properties, including the viscosity μ, are evaluated at the local bulk temperature T_b at each node. This ensures that temperature-dependent variations in the friction factor are consistently captured within the finite-volume framework.

3.3 Critical heat flux (CHF) model

The critical heat flux (CHF) is evaluated using the Groeneveld 2006 CHF look-up table, which is widely used for CHF prediction in water-cooled reactor applications. The CHF look-up table provides empirically validated CHF values as a function of pressure, mass flux, and equilibrium quality.

The 2006 CHF look-up table is based on a comprehensive database containing more than 30,000 experimental data points and represents CHF behavior for a vertical, uniformly heated 8-mm-diameter water-cooled tube. Compared to earlier versions, the 2006 table incorporates an expanded database, improved screening of experimental data, and enhanced smoothness and accuracy, particularly in the subcooled and limiting quality regions [1].

3.3.1 Applicable range of the CHF look-up table

In the present analysis, the CHF is evaluated within the following ranges of thermodynamic and flow conditions, consistent with the validity domain of the Groeneveld 2006 look-up table:

- Pressure: $14 \leq P \leq 16$ MPa
- Mass flux: $0 \leq G \leq 8000$ kg/m^2 s
- Equilibrium quality: $-0.5 \leq x_e \leq 1.0$

Here, x_e denotes the equilibrium quality evaluated from the local thermodynamic state. The validity ranges listed above define the domain over which the Groeneveld 2006 CHF look-up table can be applied. When the local thermodynamic conditions fall outside this domain, the look-up table interpolation may return undefined values (NaN) in the MATLAB implementation. Such occurrences are monitored explicitly and are discussed in the Results and Discussion sections.

3.3.2 CHF evaluation method and diameter correction

The CHF is evaluated using the direct substitution method (DSM), in which the local CHF is obtained directly from the look-up table using the local flow conditions (P, G, x_e). Specifically, the CHF value corresponding to an equivalent 8-mm tube is first obtained by direct interpolation within the look-up table.

To account for the difference between the actual channel hydraulic diameter $D_{e,ch}$ and the reference diameter of 8 mm, a diameter correction factor is applied as recommended in [1]:

$$\mathrm{CHF} = \mathrm{CHF}_{8\mathrm{mm}} \left(\frac{D_{e,ch}}{0.008} \right)^{-0.5} \tag{23}$$

This correction reflects the experimentally observed dependence of CHF on channel diameter and allows the look-up table to be extended to geometries different from the reference tube.

4 Calculation Algorithm

This section describes the overall calculation algorithm of the MATLAB system code by organizing the models and numerical procedures previously defined through a time-stepping framework. All physical relations used in the algorithm are based on the governing equations introduced in Section 1, while the discrete update rules and stability constraints follow the numerical formulations summarized in Section 2.

Rather than reintroducing individual equations or numerical schemes, the present section focuses on how these labeled components are coupled and executed within each global time step. In particular, the sequence of variable updates, conditional branching associated with steady-state convergence and reactor trip, and the execution order of different solution blocks are summarized in a pseudo-code form to clearly reflect the actual structure of the implementation.

4.1 Definition of logical flags and trip conditions

Before introducing the core calculation loop, the logical flags and trip conditions used in the algorithm are defined for clarity. These flags are introduced to distinguish physical transients from numerical convergence behavior and to ensure consistent activation of reactor kinetics and protection logic.

Steady-state flag (`steady_flag`) The steady-state flag indicates that the system has reached a sufficiently converged steady condition. During the initial convergence process, small numerical variations in temperature and flow rate may occur even in the absence of any imposed transient. To prevent unphysical activation of reactor kinetics during this phase, the point kinetics model and several trip conditions are enabled only after the steady_flag is set to true.

Once the steady-state condition is confirmed, the reference temperatures used in reactivity feedback, including the volume-averaged coolant and fuel temperatures, are fixed and used as baseline values for subsequent transient calculations.

Trip flag (`trip_flag`) The trip flag represents the occurrence of a reactor trip and is activated when any one of the predefined trip conditions is satisfied. The reactor trip conditions are defined as follows.

1. **Power deviation condition**
$$\left| \frac{P - P_{\mathrm{steady}}}{P_{\mathrm{steady}}} \right| > 0.10, \tag{24}$$
where P_{steady} denotes the steady-state reactor power. This condition is evaluated only after the steady_flag is enabled.
2. **Hot-channel exit temperature condition**
$$T_{\mathrm{hot,exit}} \geq T_{\mathrm{hot,exit}}^{\mathrm{steady}}, \tag{25}$$
where $T_{\mathrm{hot,exit}}^{\mathrm{steady}}$ corresponds to the saturation temperature at the nominal primary pressure of 15 MPa.
3. **Departure from nucleate boiling condition**
$$0 < \mathrm{MDNBR} < 1.5. \tag{26}$$
This condition is also evaluated only after the steady_flag is enabled to avoid spurious trip signals during the steady-state convergence process.

The trip_flag is set to true when any of the above conditions is satisfied. Once activated, reactor scram reactivity is inserted, pump head is removed, and the system transitions from forced circulation to natural circulation, as described in the subsequent algorithmic steps.

4.2 Initial and reference conditions

The transient simulation is initialized from a steady reference state. The primary loop is set to a pressure of 15 MPa with a uniform bulk temperature of 543.15 K. The secondary loop is initialized at 4 MPa, with the bulk temperature set equal to the saturation temperature at this pressure, $T_{\mathrm{sec}} = T_{\mathrm{sat}}(4\ \mathrm{MPa})$.

4.3 Transient timeline and event sequence

The transient simulation follows a predefined timeline designed to clearly separate steady-state convergence, reactor kinetics activation, and the imposed steam generator (SG) secondary-side depressurization event.

From $t = 0$ to $t = 120$ s, the system is advanced toward a steady-state solution under constant boundary conditions. During this period, reactor kinetics is disabled and all thermal–hydraulic variables are allowed to converge without triggering transient behavior.

At $t = 120$ s, the reactor kinetics model is activated while maintaining steady boundary conditions for an additional 10 s. This intermediate phase ensures a smooth transition from steady-state convergence to transient calculation without introducing abrupt numerical disturbances.

Starting at $t = 130$ s, the SG secondary-side pressure is linearly reduced from 4 MPa to 1 MPa over a 60 s interval. After reaching 1 MPa at $t = 190$ s, the secondary pressure is held constant for the remainder of the simulation. The calculation is continued until $t = 210$ s.

Reactor trip typically occurs during the depressurization interval ($130 \leq t \leq 190$ s), as discussed in Section 6. Once a trip condition is satisfied, scram reactivity is inserted and the primary coolant pump is simultaneously assumed to stop. As a result, forced circulation is terminated and the system transitions to a natural circulation regime driven by buoyancy effects.

```
[START]
   v
[PHASE 1: flag checker]
   |
   +--> steady_flag checker
   |      update steady_flag, T_f0, T_w0
   +--> trip_flag checker
   |      update trip_flag, trip_t
   |
   v
[PHASE 2: Update React, Decay, and src term]
   |
   +--> for each condition(React update):
   |      steady_falg off => set 0
   |      steady_falg on && trip_falg off => Kinetics
   |      steady_falg on && trip_falg on => Kinetics + scram
   |
   |
   +--> for each condition(Decay update):
   |      steady_falg off => set 1.0
   |      steady_falg on && trip_falg off => ratio P_curr/P_steady
   |      steady_falg on && trip_falg on => ANS73
   |
   +--> update src term based on the Decay
   |
   v
[PHASE 3: Update dt]
   |
   +--> for each item in hot, avg, SG, SG_2nd:
   |      dt_UB_item = min (rho * V / m);
   +--> dt_UB = min dt_UB_item : item in hot, avg, SG, SG_2nd
   +--> update dt (Section 2.1)
   |        dt = 0.8 * dt_UB
   |
   v
[PHASE 4] (cont'd)
```

```
[PHASE 4: Update h -> Tb -> P sequential]
   |
   +--> for each condition1:
   |      exist Src term =>
   |        (hot, avg, SG, SG_2nd case)
   |        Linear update (not cicular loop form)
   |        for each condition2:
   |          dyn Src => (SG, SG_2nd)
   |            loop start: (Section 2.7)
   |              (update both SG, SG_2nd)
   |              given temp diff ->
   |                update qpp for heat removal
   |              upwind scheme (Section 2.3) -> update h
   |              cal. Tb from h (Section 2.8) -> update Tb and P
   |
   |              break cond: (Tb_SG_2nd less chage)
   |            loop end
   |
   |          given. Src => (hot, avg)
   |            upwind scheme (Section 2.3) -> update h
   |            cal. Tb from h (Section 2.8) -> update Tb and P
   |
   |      else =>
   |        (LP, UP, pipe, ... etc)
   |        remain h and Tb, just update P
   |
   v
[PHASE 6: Update Tf (hot, avg)]
   |
   +--> cal. h_DB (heat) based on Tb and qpp
   |
   +--> Update Tf (Section 2.6)
   |        Tf based on h_DB and Tb
   |
   v
[PHASE 6: Update Mass flux]
   |
   +--> for each condition:
   |      (update total mass flux (Section 2.2))
   |      trip_flag on => natural circulation
   |      trip_flag off => const.
   |
   +--> update Mass flow dist btw 2 channels (Section 1.7)
   |      to remove the P_core diff
   |
   v
[PHASE 7: Update RX Kinetics]
   |
   +--> for each condition:
   |      (update Power and Conc. (Section 2.5))
   |      steady_flag on => Kinetics start
   |      steady_flag off => const.
   |
   v
[END]
```

5 MARS Input Description

The MARS input deck is constructed to reproduce the same physical system, boundary conditions, and transient scenario as those implemented in the MATLAB system code, so that a direct comparison between the two simulations is possible. To this end, the nodalization strategy, initial conditions, and reactivity model parameters are matched as closely as permitted by the MARS component structure.

In particular, the same reactivity feedback coefficients (α_w, α_f), scram reactivity insertion model $\rho_{\text{scram}}(t)$, and steady-state reference conditions are used in both simulations. All initial temperatures, pressures, and mass flow rates are aligned with the MATLAB steady-state solution prior to the transient initiation.

System component classification and nodalization

Table 1 summarizes the MARS component types used in the present model. Each component type is selected to represent a specific physical element of the reactor coolant system and steam generator, consistent with the lumped-parameter formulation adopted in the MATLAB code.

Symbol	Component	Example ID
△	**BRANCH**	102, 103
▭	**Heat Structure**	010(hot), 020(avg), 030(SG)
▭	**PIPE**	010(hot), 020(avg), 030(SG p), 040(SG s)
▭	**SNGLVOL**	100, 105, 115, 131
▭	**TMDPVOL**	640(PZR), 120, 125
○	**SNGLJUN**	111, 114, 130, 124
○	**TMDPJUN**	121, 132
○	**VALVE**	032

Table 1: System Component Classification

Figure 2 shows the complete nodalization diagram of the MARS model. The system is composed of two parallel core channels (hot and average), primary loop piping, and a steam generator with explicitly separated primary and secondary sides. The axial discretization of the PIPE components is chosen to be consistent with the control-volume layout used in the MATLAB model, enabling node-by-node comparison of key thermal-hydraulic quantities.

Core and primary loop modeling

The reactor core is modeled using two parallel PIPE components representing the hot and average channels. Each channel is axially discretized and coupled to corresponding heat structures to model fuel-to-coolant heat transfer. This configuration directly mirrors the two-channel representation used in the MATLAB system code.

The primary loop piping (hot leg, cold leg, and connecting segments) is modeled using PIPE and BRANCH components. In order to form a closed loop in MARS, the projection lengths of the hot leg and cold leg are required to be identical. As a result, slight differences in pipe inclination angles are introduced compared to the idealized MATLAB geometry. This geometric adjustment is a modeling constraint of MARS and does not affect the overall pressure balance or natural circulation behavior.

Hydraulic diameter and pressure-drop representation

In the MATLAB code, individual flow channels are explicitly represented, whereas in MARS it is impractical to model each physical channel as a separate PIPE component. Instead, groups of parallel channels are lumped into single PIPE components.

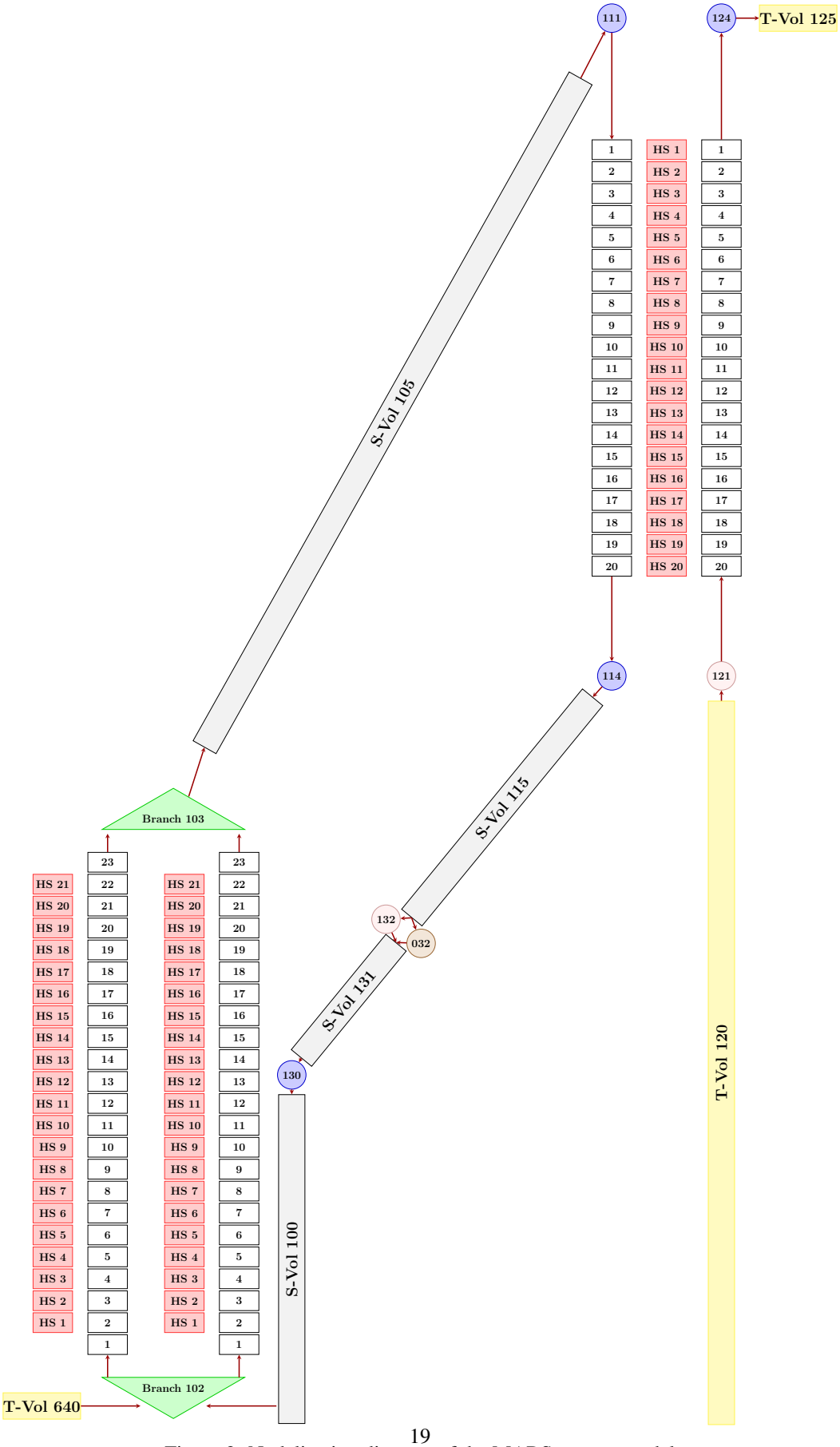


Figure 2: Nodalization diagram of the MARS system model

To ensure consistent pressure-drop behavior, the hydraulic diameter D_e is specified explicitly for each PIPE. The selected D_e corresponds to the representative subchannel hydraulic diameter rather than an area-averaged value. This choice is appropriate because, at a given axial elevation, all parallel channels experience the same pressure drop, and the smallest characteristic hydraulic diameter governs the frictional resistance.

Figure 3 illustrates the conceptual approach used to evaluate pressure losses in the lumped MARS representation.

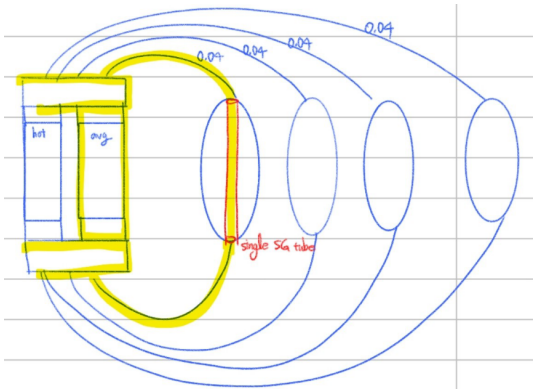


Figure 3: Conceptual pressure-drop evaluation in the lumped MARS model

Steam generator modeling and boundary conditions

The steam generator is modeled by coupling primary- and secondary-side PIPE components through dedicated heat structures. The secondary side pressure and feedwater temperature are imposed using a time-dependent volume (TMDPVOL), which allows direct specification of the pressure transient from 4 MPa to 1 MPa, identical to the MATLAB scenario.

This approach ensures that the primary–secondary heat transfer behavior and boundary conditions are consistent between the two simulations, while respecting the component-based structure of the MARS code.

Overall, the MARS input deck is designed to preserve the same physical assumptions, operating conditions, and transient logic as the MATLAB model, with unavoidable differences arising only from code-specific modeling constraints.

6 Results

For a consistent comparison between the MATLAB and MARS results, the MARS outputs were post-processed and replotted using MATLAB, while the original MARS figures are provided in Appendix A.

In particular, the MDNBR for the MARS case was re-evaluated in MATLAB by extracting the local pressure, mass flux, and equilibrium quality (P, G, x_e) and applying the Groeneveld 2006 CHF look-up table [1], ensuring an identical CHF model for both simulations. The post-processed MDNBR remained above the trip threshold of 1.5 *before the trip*, confirming that the reactor trip was not triggered by the MDNBR criterion. Further discussion on this treatment is given in Section 7.

	Kinetics on		OC start		Trip start	
	MATLAB	MARS	MATLAB	MARS	MATLAB	MARS
Absolute time [s]	120	120	130	130	152.918	162.103
Relative to OC [s]	-10	-10	0	0	22.918	32.103

Table 2: Comparison of key event timings between MATLAB and MARS simulations

6.1 Core power change

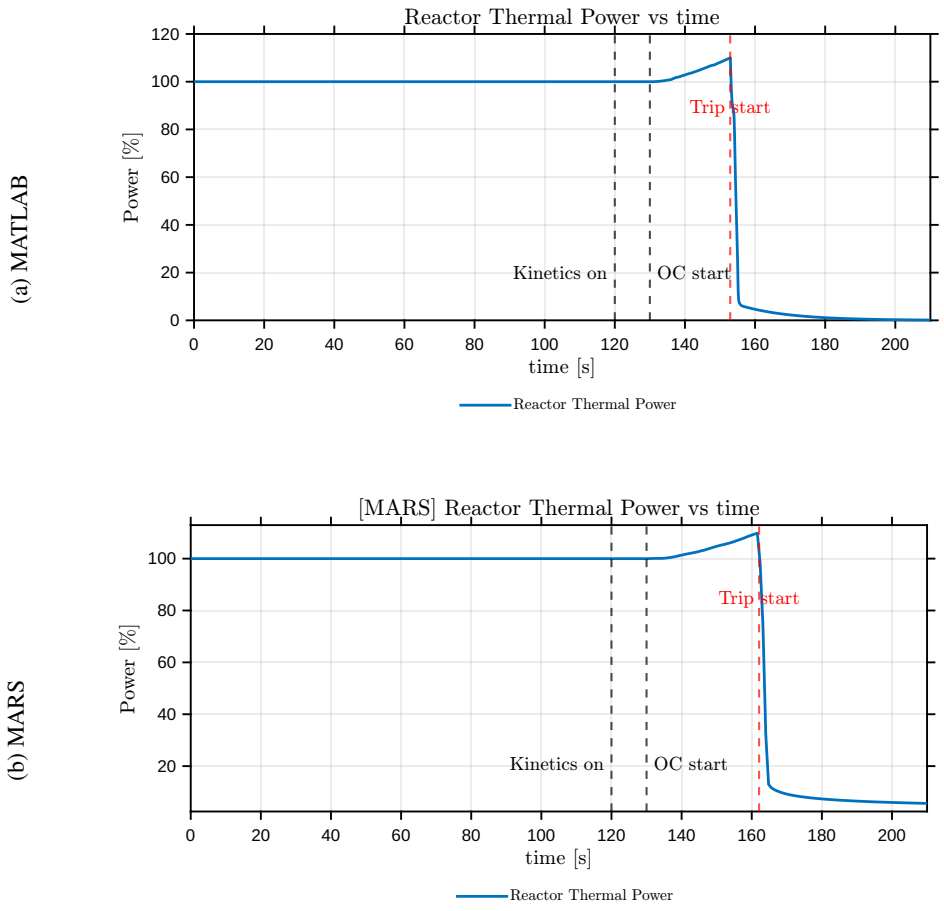
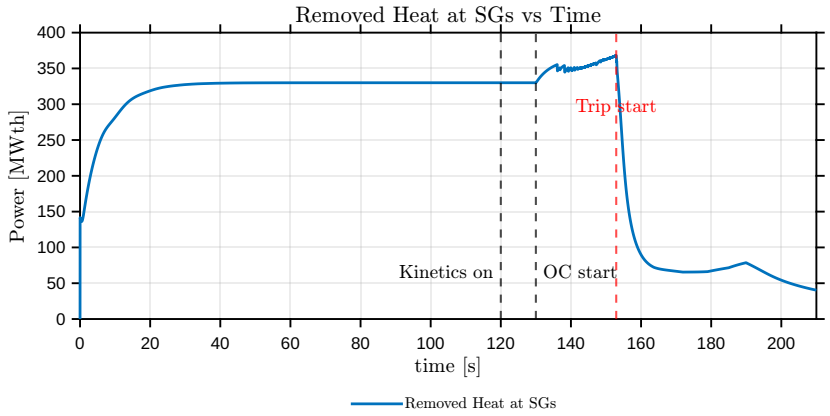


Figure 4: Reactor thermal power as a function of time

6.2 Removed heat via steam generator

(a) MATLAB



(b) MARS

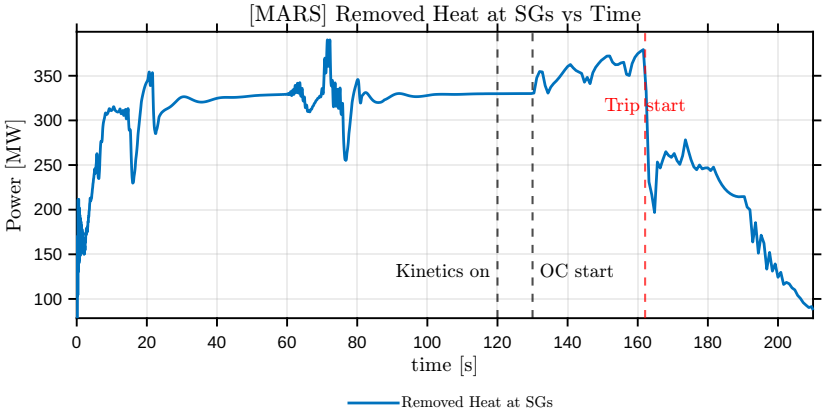
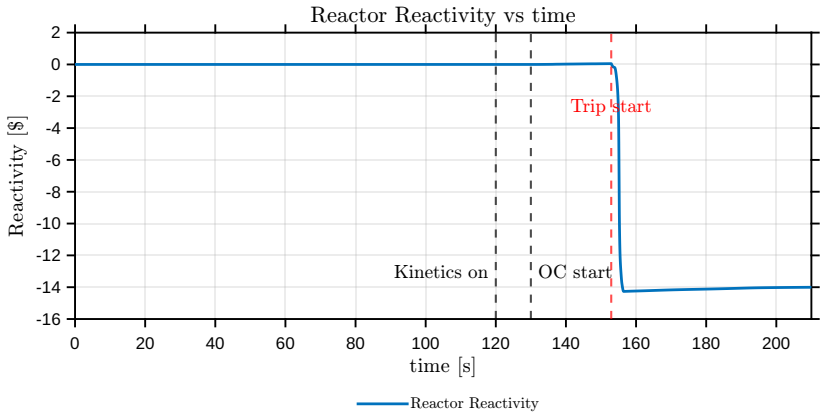


Figure 5: Removed heat via the steam generator (S/G)

6.3 Reactivity change

(a) MATLAB



(b) MARS

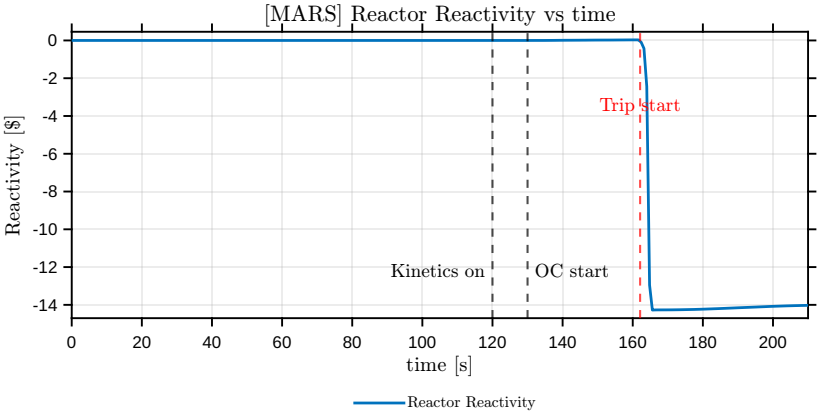
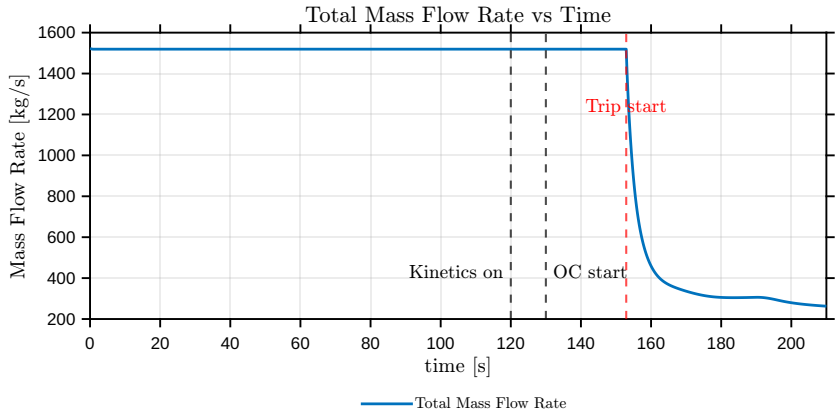


Figure 6: Reactor reactivity

6.4 Total mass flow rate change

(a) MATLAB



(b) MARS

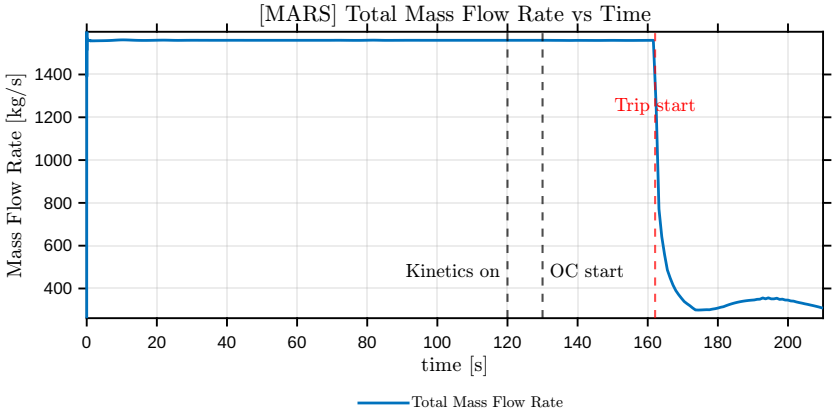
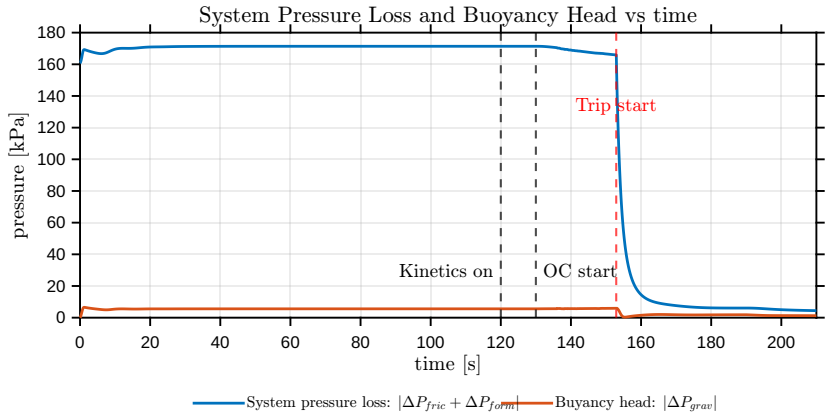


Figure 7: Total mass flow rate

6.5 System pressure losses and system buoyancy head change

(a) MATLAB



(b) MARS

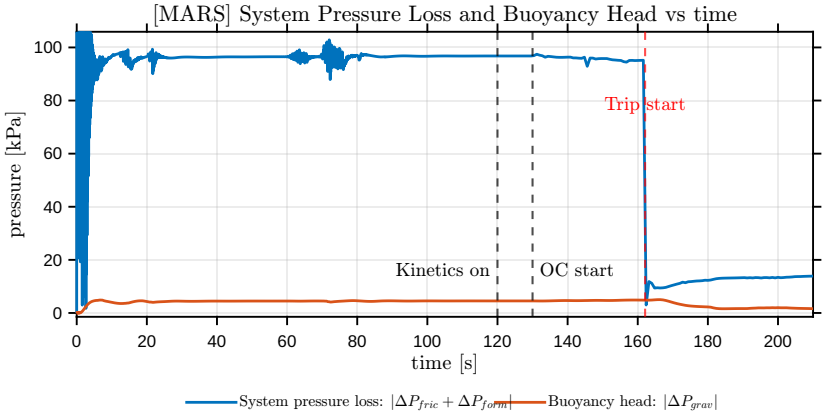
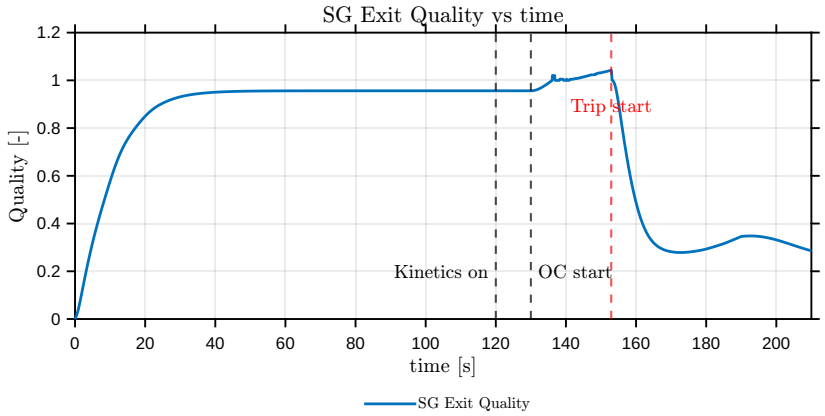


Figure 8: System pressure losses and buoyancy head

6.6 Steam generator exit quality and exit temperature change

(a) MATLAB



(b) MARS

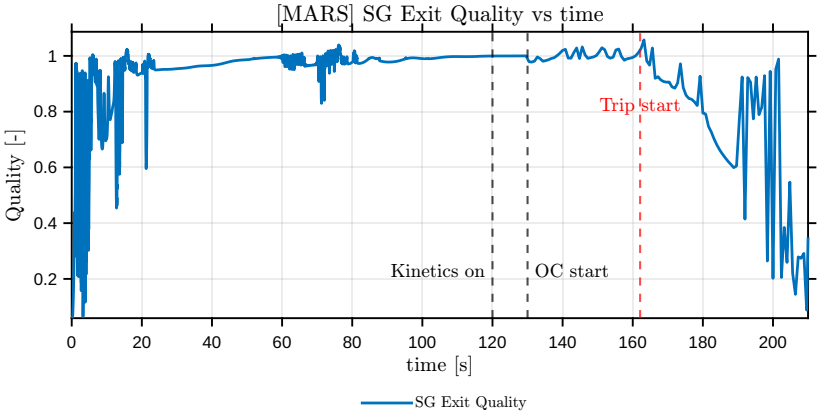
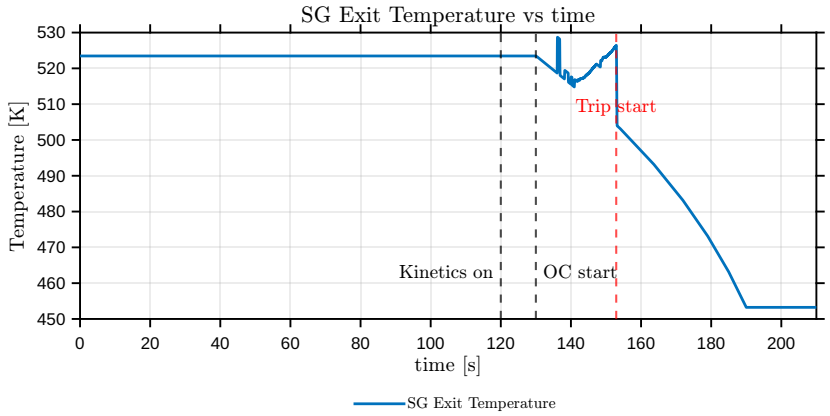


Figure 9: Steam generator exit quality

(a) MATLAB



(b) MARS

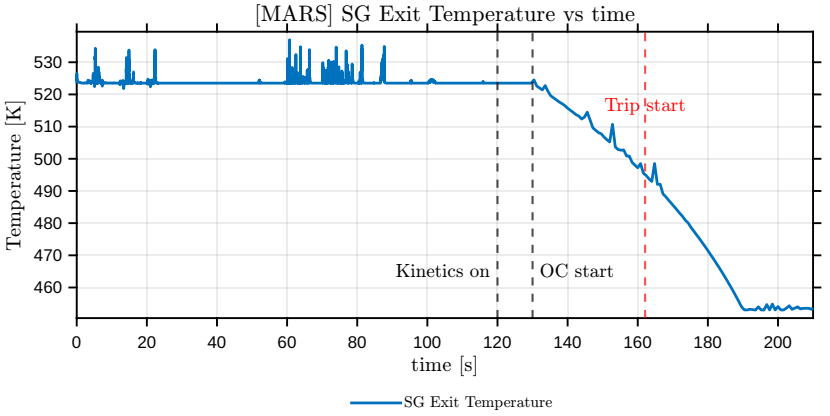
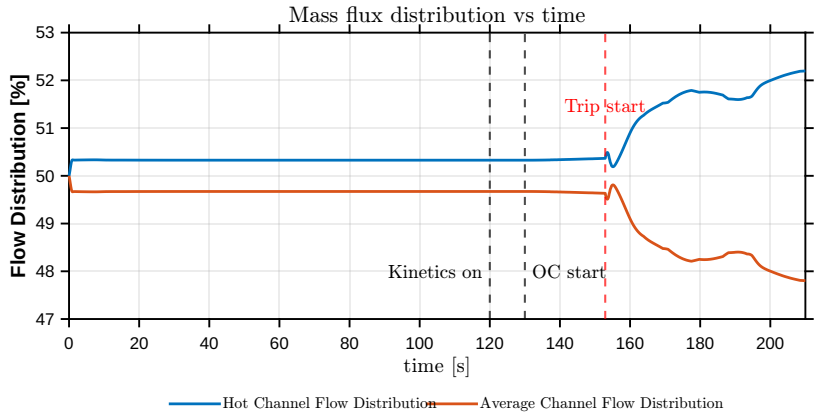


Figure 10: Steam generator exit temperature

6.7 Flow distribution change

(a) MATLAB



(b) MARS

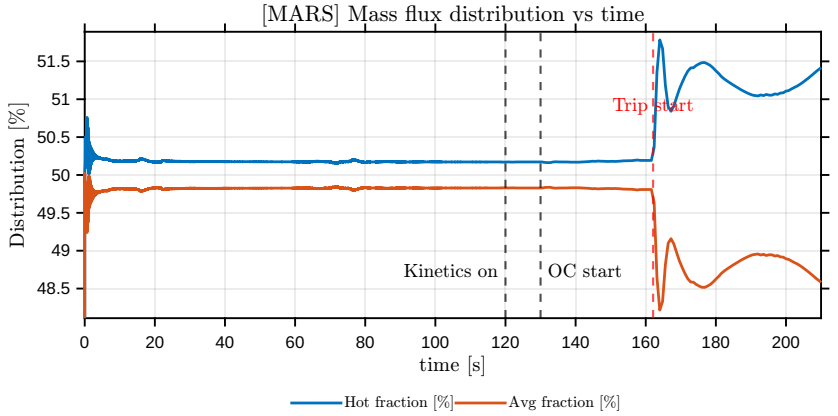
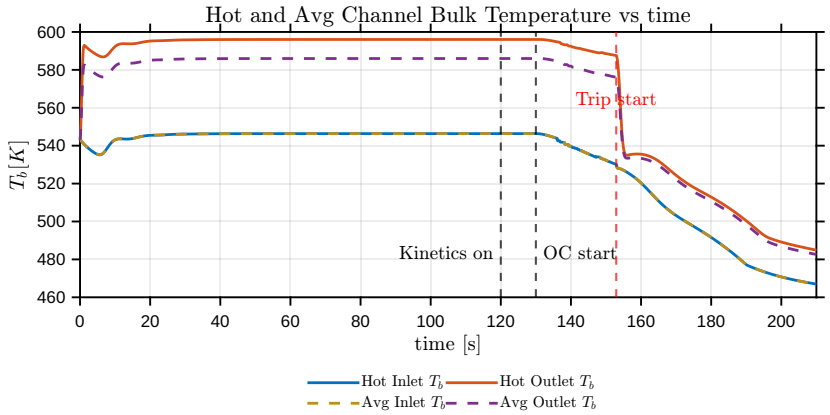


Figure 11: Mass flux distribution between hot and average channels

6.8 Inlet and outlet bulk temperature of core two channels

(a) MATLAB



(b) MARS

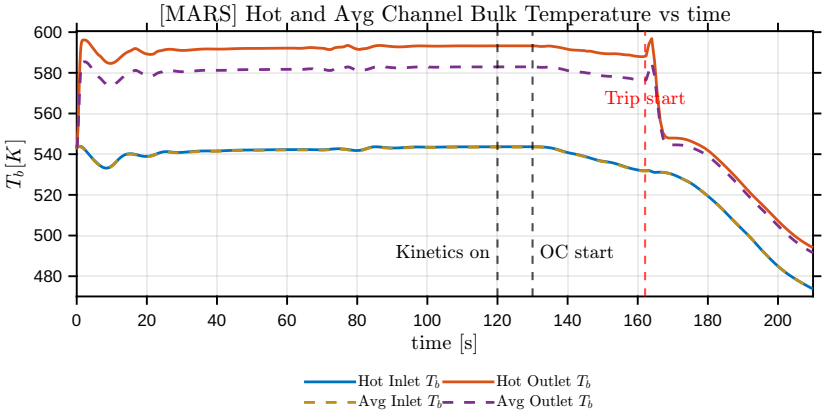
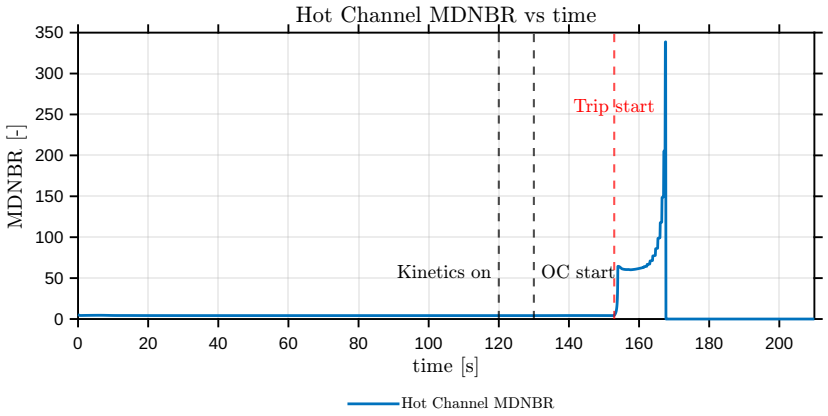


Figure 12: Inlet and outlet bulk temperatures of the hot and average channels

6.9 MDNBR change

(a) MATLAB



(b) MARS

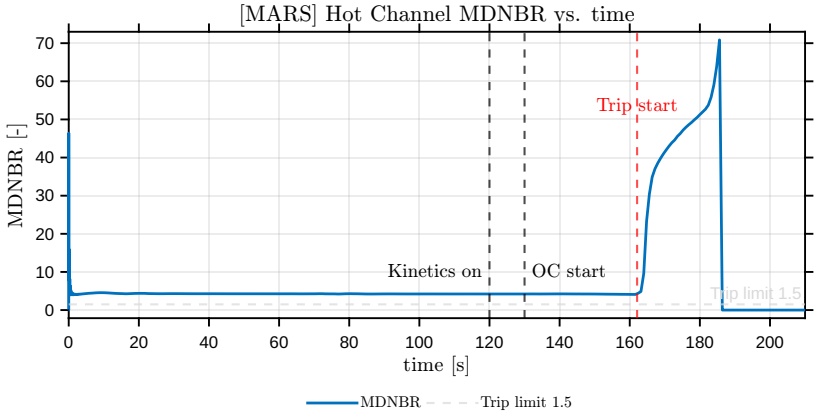
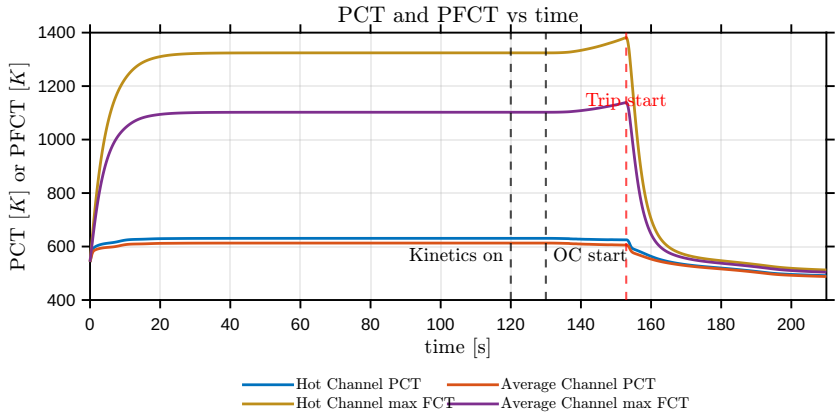


Figure 13: Minimum DNBR (MDNBR) of the hot channel

6.10 PCT change and maximum FCT change

(a) MATLAB



(b) MARS

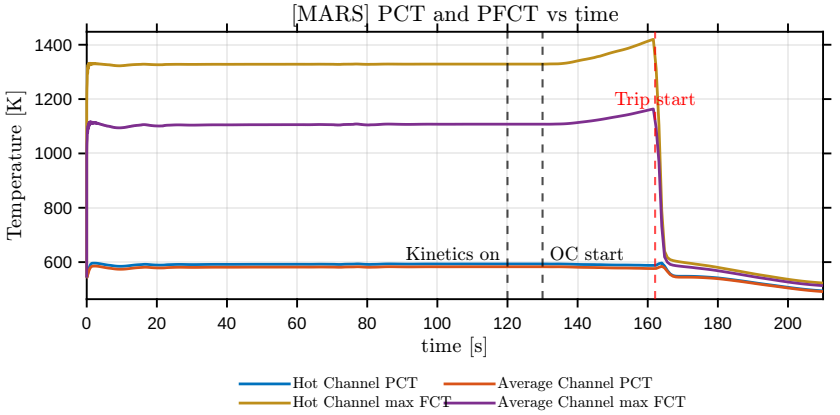


Figure 14: Peak cladding temperature (PCT) and max. fuel centerline temperature (FCT)

7 Discussion

7.1 Discrepancy between MARS internal `htchf` and CHFLUT2006-based MDNBR

Figure 15 compares the minimum departure from nucleate boiling ratio (MDNBR) obtained directly from the MARS internal CHF variable (`htchf`) with the MDNBR re-evaluated using the Groeneveld 2006 CHF look-up table (CHFLUT2006) based on the same MARS thermal–hydraulic outputs.

The motivation for this comparison arose from the observation that the MDNBR predicted by MARS reaches zero earlier than that obtained from the standalone MATLAB simulation. To identify the source of this discrepancy, the MDNBR was recalculated using the local pressure, mass flux, and equilibrium quality extracted from the MARS results, while applying the CHFLUT2006 correlation consistently.

As shown in Fig. 15, a clear difference is observed between the MDNBR evaluated using the MARS internal `htchf` and that computed using CHFLUT2006, even though both are nominally based on the same CHF correlation. In particular, the `htchf`-based MDNBR exhibits an earlier and more abrupt decrease, reaching zero while the CHFLUT2006-based MDNBR remains finite.

This discrepancy indicates that the CHF handling within MARS may differ from a direct application of the CHFLUT2006 look-up table, motivating a closer examination of the thermodynamic conditions and validity ranges used in the internal CHF evaluation.

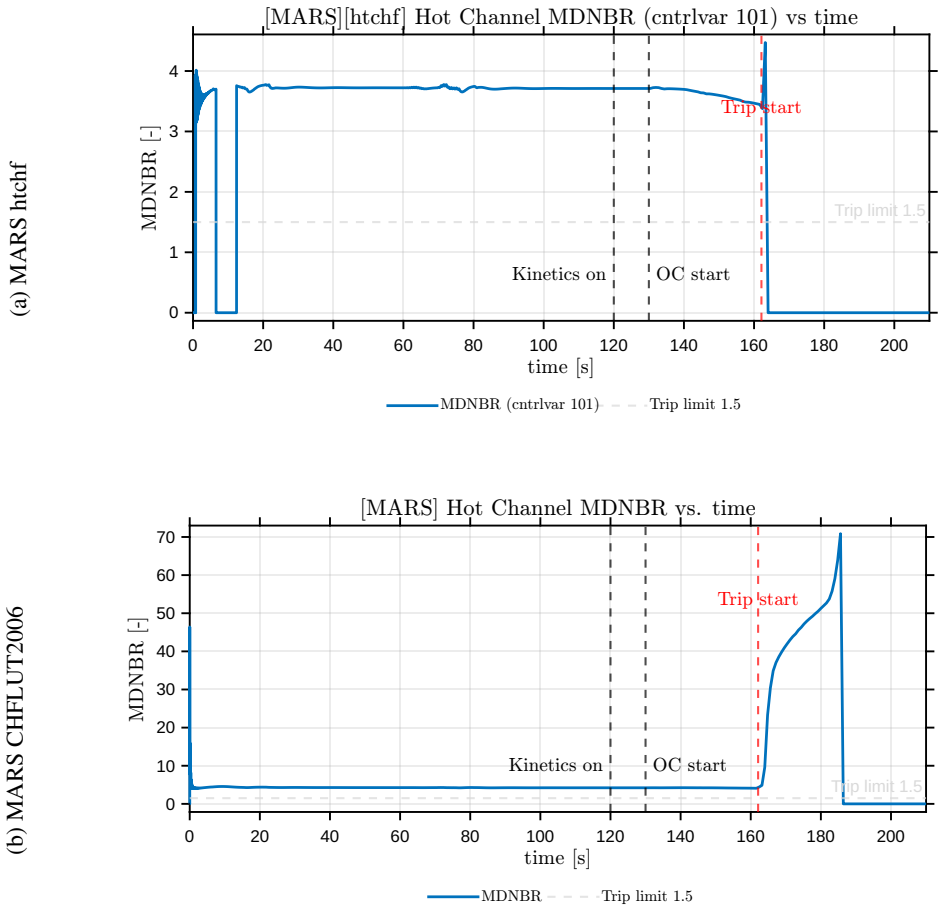


Figure 15: Reactor thermal power as a function of time

7.2 Validity range of CHF evaluation and origin of zero `htchf`

To identify the origin of the discrepancy discussed in the previous subsection, the thermodynamic variables used for the CHF evaluation in MARS were examined in detail. According to the TA, MARS employs the same Groeneveld CHF look-up table for critical heat flux prediction. Based on this explanation, the local equilibrium quality x_e and pressure P, which are key input variables for the CHF evaluation, were extracted from the MARS results and analyzed.

Figures 16 and 17 show the time histories of the hot-channel equilibrium quality and pressure obtained from the MARS simulation, respectively. The regions in which the CHFLUT2006 correlation is well defined are highlighted in green. Despite remaining within these nominal validity ranges, the CHF value computed by the MARS internal variable `htchf` is observed to collapse to zero.

This observation indicates that the zero-valued CHF predicted by `htchf` cannot be attributed solely to an excursion beyond the formal applicability limits of the CHFLUT2006 look-up table. Instead, it suggests that additional internal treatment or limiting logic within the MARS CHF implementation may be responsible for the observed behavior.

For this reason, in the result comparison presented in Section 6, the MDNBR obtained directly from the MARS internal CHF output was not used. Instead, the MDNBR was consistently re-evaluated based on the extracted $(P,\ G,\ x_e)$ using the CHFLUT2006 correlation, allowing a more controlled and transparent comparison between the MATLAB and MARS simulations.

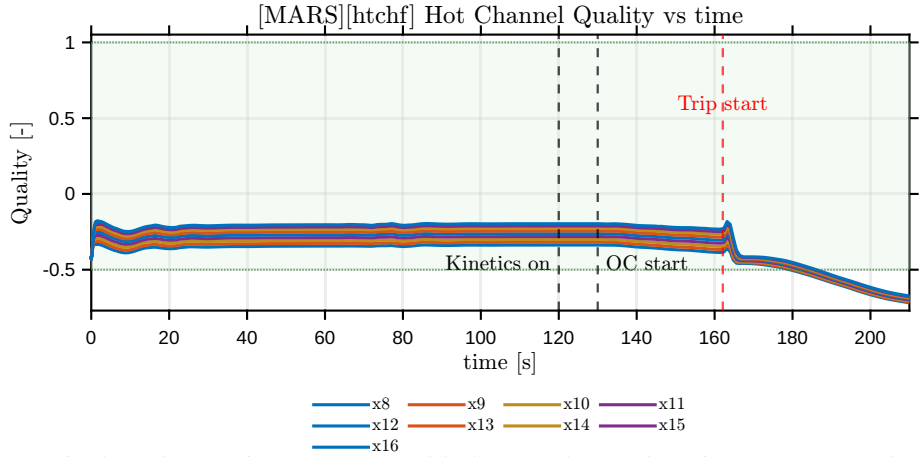


Figure 16: Time history of hot-channel equilibrium quality obtained from the MARS simulation.

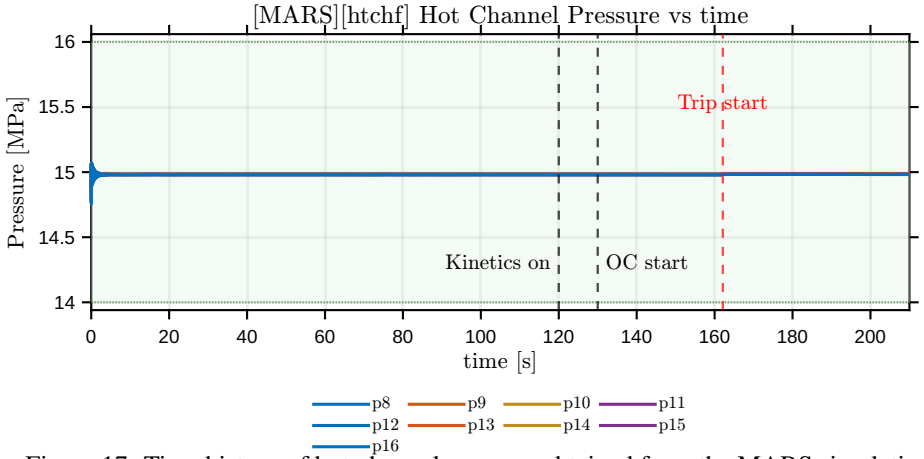


Figure 17: Time history of hot-channel pressure obtained from the MARS simulation.

7.3 Physical interpretation of the transient behavior

Before directly comparing the MATLAB and MARS results, it is necessary to first understand the physical mechanisms driving the observed transient behavior. The analysis therefore focuses on the period following the activation of reactor kinetics and the onset of the overcooling (OC) transient.

After reactor kinetics is activated at $t = 120$ s, the system is held under steady boundary conditions for an additional 10 s. At $t = 130$ s, the OC transient begins as the steam generator secondary-side pressure is reduced, leading to enhanced heat removal. As a result, the bulk coolant temperature T_b decreases at both the inlet and outlet of the core.

The reduction in coolant temperature causes the wall temperature T_w to fall below its steady-state reference value T_{w0}, producing a positive reactivity insertion through the moderator temperature feedback described in Section 1.3. Consequently, reactor power increases according to the point kinetics model summarized in Section 1.4. Despite this power increase, the cooling effect associated with the OC transient dominates, and the bulk coolant temperature continues to decrease, resulting in a net increase in total reactivity.

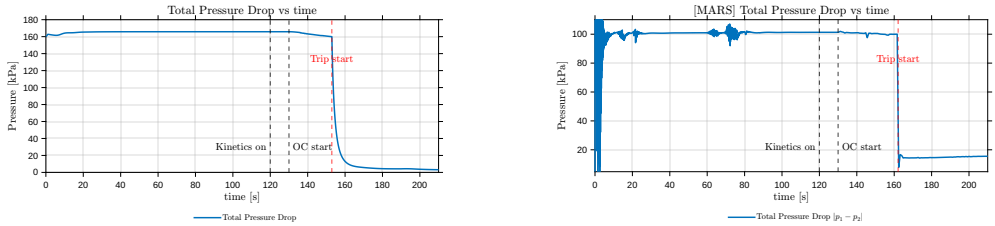
An important feature of this phase is that the fuel temperature T_f increases even as T_b decreases. The increase in T_f is driven by the rising reactor power. However, the contribution of fuel temperature feedback to reactivity is weaker than that of the coolant temperature feedback, as reflected by the reactivity coefficients $\alpha_w = -0.4 \times 10^{-2}\beta$ and $\alpha_f = -0.03 \times 10^{-2}\beta$. As a result, the decrease in T_b outweighs the increase in T_f, leading to a net positive reactivity insertion.

Reactor power continues to rise until it reaches the trip threshold of 110% of the steady-state value, at which point the reactor trip is initiated. Following the trip, scram reactivity is inserted and the primary pump is assumed to stop simultaneously, causing the system to transition to natural circulation.

The insertion of scram reactivity leads to a rapid reduction in total reactivity, followed by a decrease in reactor power as described by the point kinetics model. As power decreases, both the bulk coolant temperature T_b and the fuel temperature T_f decrease. The scram reactivity becomes constant after approximately 3.5 s, after which a gradual increase in reactivity is observed due to temperature feedback effects, as shown in Fig. 6.

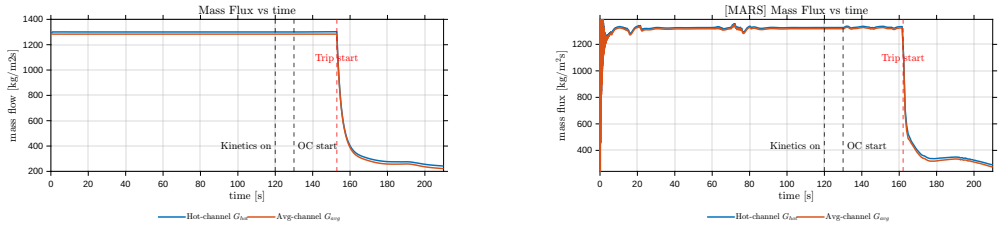
7.4 Additional hydraulic information

Before comparing the MATLAB and MARS results in detail, additional hydraulic quantities are presented to provide context for the system response during the transient. Figures 18 and 19 summarize the time evolution of the total pressure drop and mass flux, respectively, as predicted by the two simulations. These quantities characterize the transition from forced circulation to natural circulation and form the hydraulic basis for the subsequent thermal and safety-related comparisons.



(a) MATLAB (b) MARS

Figure 18: Total pressure drop as a function of time.



(a) MATLAB

(b) MARS

Figure 19: Mass flux as a function of time.

7.5 Comparison between MATLAB and MARS results

A direct comparison between the MATLAB and MARS results reveals several systematic differences in both the timing of the reactor trip and the post-trip thermal–hydraulic response.

As summarized in Table 2, the reactor trip occurs earlier in the MATLAB simulation than in the MARS simulation. This difference is attributed to a faster reactivity update in the MATLAB point kinetics model, which leads to a more rapid increase in reactor power and an earlier satisfaction of the trip criterion.

After the reactor trip, notable differences are observed in the MDNBR, steam generator heat flux, and outlet quality. As shown in Fig. 13, the post-trip MDNBR reaches values on the order of 350 in the MATLAB simulation, whereas the corresponding maximum value in MARS remains around 70. This discrepancy indicates a much more rapid reduction of the steam generator heat flux q''_{SG} in the MATLAB model. Because the MDNBR is defined as the ratio of the critical heat flux to the local heat flux, the abrupt decrease in q''_{SG} directly results in an artificially large MDNBR.

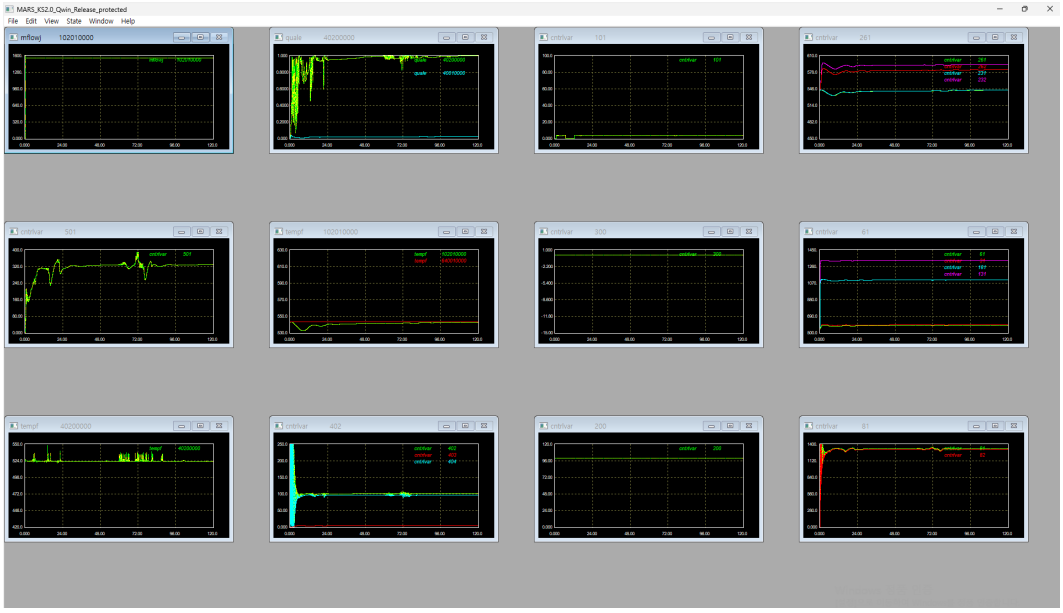
In the MATLAB implementation, the steam generator heat flux is updated primarily based on temperature differences without explicitly modeling thermal inertia. As a result, q''_{SG} follows the reactor power almost instantaneously after the trip, leading to the rapid decrease observed in Fig. 5. A similar trend is observed for the steam generator outlet quality, as shown in Fig. 9. In contrast, the MARS simulation exhibits a more gradual response due to its more detailed treatment of system dynamics.

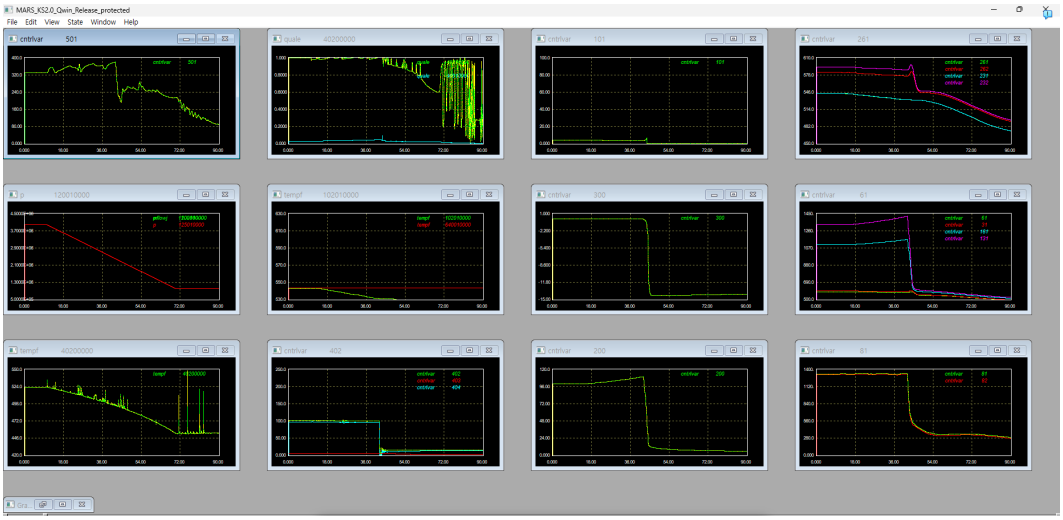
Differences are also observed in the mass flux distribution following the trip. In the MATLAB model, the inertia of the entire primary loop is accounted for explicitly, resulting in a relatively smooth and gradual decrease in mass flux. In the MARS model, the primary pump is represented by a time-dependent pressure boundary condition, and pump trip is implemented using a valve model. This treatment leads to a more abrupt initial change in the loop momentum, producing stronger early-time fluctuations. As shown in Fig. 19, the mass flux decreases more rapidly in MARS immediately after the trip, whereas the MATLAB result exhibits a slower decay. The lower final mass flux predicted by MATLAB is consistent with its larger total pressure drop, as shown in Fig. 18.

References

[1] D. C. Groeneveld, J. Q. Shan, A. Z. Vasić, L. K. H. Leung, A. Durmayaz, J. Yang, S. C. Cheng, and A. Tanase. The 2006 chf look-up table. *Nuclear Engineering and Design*, 237(15–17): 1909–1922, 2007. doi: 10.1016/j.nucengdes.2007.02.014. URL `https://doi.org/10.1016/j.nucengdes.2007.02.014`.

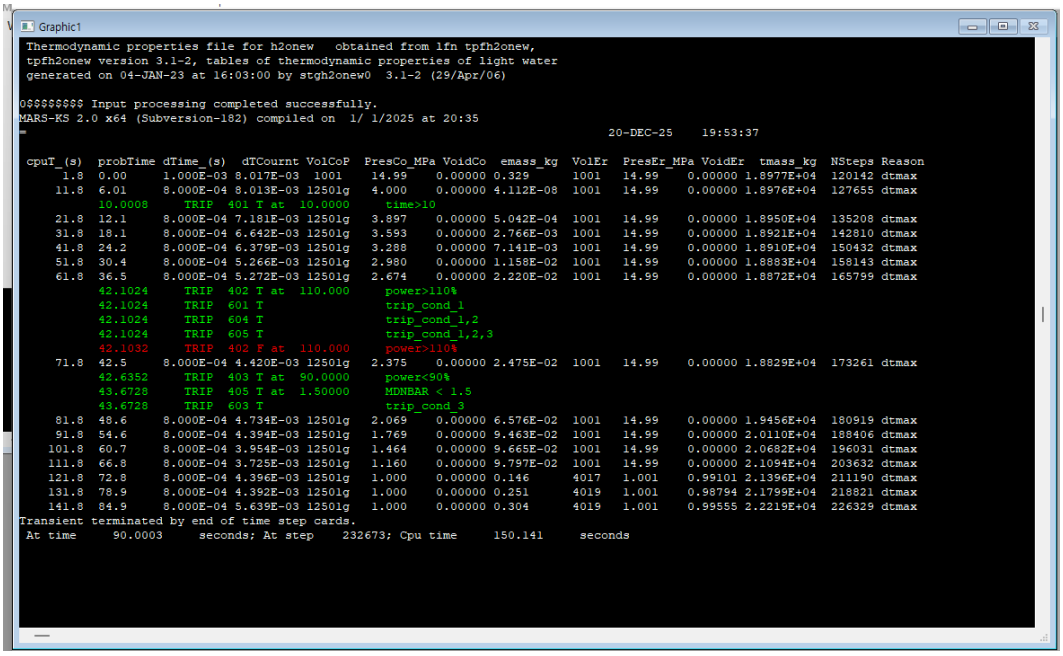
A Original MARS Results





B Trip Results

```
=== TRIP CONDITION CHECK ===
Power            : 3.6302e+08 W
Power (steady)   : 3.3000e+08 W
ΔPower / P0      :   10.006 %  --> cond1 = 1
T_hot_exit       :   587.59 K
T_hot_exit (ss)  :   615.18 K  --> cond2 = 0
MDNBR            :  4.213     --> cond3 = 0
>>> TRIP = 1 <<<
============================
```



```
Graphic1
 Thermodynamic properties file for h2onew   obtained from lfn tpfh2onew,
 tpfh2onew version 3.1-2, tables of thermodynamic properties of light water
 generated on 04-JAN-23 at 16:03:00 by stgh2onew0  3.1-2 (29/Apr/06)

 0$$$$$$$$ Input processing completed successfully.
 MARS-KS 2.0 x64 (Subversion-182) compiled on  1/ 1/2025 at 20:35
                                                                    20-DEC-25    19:53:37

  cpuT_(s)  probTime dTime_(s)  dTCournt VolCoP  PresCo_MPa VoidCo  emass_kg  VolEr  PresEr_MPa VoidEr  tmass_kg  NSteps Reason
       1.8  0.00     1.000E-03 8.017E-03  1001    14.99    0.00000 0.329     1001   14.99    0.00000 1.8977E+04  120142 dtmax
      11.8  6.01     8.000E-04 8.013E-03 12501g   4.000    0.00000 4.112E-08 1001   14.99    0.00000 1.8976E+04  127655 dtmax
            10.0008     TRIP  401 T at  10.0000     time>10
      21.8  12.1     8.000E-04 7.181E-03 12501g   3.897    0.00000 5.042E-04 1001   14.99    0.00000 1.8950E+04  135208 dtmax
      31.8  18.1     8.000E-04 6.642E-03 12501g   3.593    0.00000 2.766E-03 1001   14.99    0.00000 1.8921E+04  142810 dtmax
      41.8  24.2     8.000E-04 6.379E-03 12501g   3.288    0.00000 7.141E-03 1001   14.99    0.00000 1.8910E+04  150432 dtmax
      51.8  30.4     8.000E-04 5.266E-03 12501g   2.980    0.00000 1.158E-02 1001   14.99    0.00000 1.8883E+04  158143 dtmax
      61.8  36.5     8.000E-04 5.272E-03 12501g   2.674    0.00000 2.220E-02 1001   14.99    0.00000 1.8872E+04  165799 dtmax
            42.1024     TRIP  402 T at  110.000     power>110%
            42.1024     TRIP  601 T               trip_cond_1
            42.1024     TRIP  604 T               trip_cond_1,2
            42.1024     TRIP  605 T               trip_cond_1,2,3
            42.1032     TRIP  402 F at  110.000     power>110%
      71.8  42.5     8.000E-04 4.420E-03 12501g   2.375    0.00000 2.475E-02 1001   14.99    0.00000 1.8829E+04  173261 dtmax
            42.6352     TRIP  403 T at  90.0000     power<90%
            43.6728     TRIP  405 T at  1.50000     MDNBAR < 1.5
            43.6728     TRIP  603 T               trip_cond_3
      81.8  48.6     8.000E-04 4.734E-03 12501g   2.069    0.00000 6.576E-02 1001   14.99    0.00000 1.9456E+04  180919 dtmax
      91.8  54.6     8.000E-04 4.394E-03 12501g   1.769    0.00000 9.463E-02 1001   14.99    0.00000 2.0110E+04  188406 dtmax
     101.8  60.7     8.000E-04 3.954E-03 12501g   1.464    0.00000 9.665E-02 1001   14.99    0.00000 2.0682E+04  196031 dtmax
     111.8  66.8     8.000E-04 3.725E-03 12501g   1.160    0.00000 9.797E-02 1001   14.99    0.00000 2.1094E+04  203632 dtmax
     121.8  72.8     8.000E-04 4.396E-03 12501g   1.000    0.00000 0.146     4017   1.001    0.99101 2.1396E+04  211190 dtmax
     131.8  78.9     8.000E-04 4.392E-03 12501g   1.000    0.00000 0.251     4019   1.001    0.98794 2.1799E+04  218821 dtmax
     141.8  84.9     8.000E-04 5.639E-03 12501g   1.000    0.00000 0.304     4019   1.001    0.99555 2.2219E+04  226329 dtmax
 Transient terminated by end of time step cards.
  At time     90.0003     seconds; At step    232673; Cpu time     150.141     seconds
```