Article

Not peer-reviewed version

First-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations: Mathematical Framework and Illustrative Application to the Nordheim-Fuchs Reactor Safety Model

Dan Gabriel Cacuci *

Posted Date: 1 August 2024

doi: 10.20944/preprints202407.2613.v1

Keywords: neural ordinary differential equations (NODE); comprehensive adjoint sensitivity analysis methodology for NODE (1st-CASAM-NODE); Nordheim-Fuchs reactor safety model; sensitivity analysis for model features ((1st-FASAM-N); exact sensitivities

Article

First-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations: Mathematical Framework and Illustrative Application

Dan Gabriel Cacuci

University of South Carolina; cacuci@cec.sc.edu

Abstract: This work introduces the mathematical framework of the novel "First-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations (1^{st}-CASAM-NODE)" which yields exact expressions for the first-order sensitivities of NODE decoder-responses to the NODE parameters, including encoder initial conditions, while enabling the most efficient computation of these sensitivities. The application of the 1^{st}-CASAM-NODE is illustrated by using the Nordheim-Fuchs reactor dynamics/safety phenomenological model, which is representative of physical systems that would be modeled by NODE while admitting exact analytical solutions for all quantities of interest (hidden states, decoder outputs, sensitivities with respect to all parameters and initial conditions, etc.). This work also lays the foundation for the ongoing work on conceiving the "Second-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations (2^{nd}-CASAM-NODE)" which aims at yielding exact expressions for the second-order sensitivities of NODE decoder-responses to the NODE parameters and initial conditions while enabling the most efficient computation of these sensitivities

Keywords: neural ordinary differential equations (NODE); comprehensive adjoint sensitivity analysis methodology for NODE (1^{st}-CASAM-NODE); Nordheim-Fuchs reactor safety model; sensitivity analysis for model features ((1^{st}-FASAM-N); exact sensitivities

1. Introduction

Concepts of dynamical systems theory had been frequently used to improve neural network performance [1–3] but Neural Ordinary Differential Equations (NODE) appear to have been formally introduced by Chen et al. [4]. NODE provide an explicit connection between deep feed-forward neural networks and dynamical systems and are considered to provide a bridge between modern deep learning and classical mathematical/numerical modelling. NODE provide a flexible trade-off between efficiency, memory costs and accuracy. The approximation capabilities [5,6] of NODE are particularly useful for time-series modelling [4,7,8], generative models for continuous normalizing flows [4,9] and modeling/controlling physical environments [see, e.g., 10].

Neural ODEs are trained by minimizing a least-squares quadratic scalar-valued "loss function" by computing its gradients with respect to the weights to be optimized using a first-order optimizer such as "stochastic gradient descent" [11,12]). Since ODE solvers (e.g., Runge–Kutta solvers) perform differentiable algebraic operations, the gradients of the loss function can be calculated by the so-called "direct method" which directly backpropagates through the operations performed by the ODE solver. However, when the dynamics are complex, the "direct method" can lead to an arbitrarily large number of function evaluations for adaptive solvers while storing all of the intermediate activations during the "solving" process, so the "direct method" becomes prohibitively memory intensive. A NODE-training method which is less memory intensive is the so-called "adjoint method" [13–15], which solves an ODE (related to the original NODE) backwards in time. The direct method is faster but is more memory intensive than the adjoint method. The one-dimensional definite integrals, which appear when computing gradients via the "adjoint method" are traditionally evaluated by solving them as differential equations, which considerably slows down the training process. Evaluating these one-dimensional definite integrals by using Gauss–Legendre quadrature (rather than solving them as ODEs) has been shown [16] to be faster than ODE-based methods while retaining memory efficiency, thus speeding up the training of NODE.

The gradients of the loss function are often called "sensitivities" in the literature on neural nets and aspects of the optimization/training procedure are occasionally called "sensitivity analysis." But the "loss function" is of interest only for the "training" phase of the NODE and the "sensitivities of the loss function" are driven towards the ideal zero-values by the minimization process while optimizing the NODE weights/parameters. Furthermore, after the NODE is optimized to reproduce the underlying physical system as closely as possible, the responses of interest for the NODE-modeled system is no longer a "loss function" but are various functions of the NODE's "decoder"-output. Since the physical system being modeled by the NODE comprises itself parameters that stem from measurements or computations, they are not perfectly well-known, but are afflicted by uncertainties that stem from the respective experiments and/or computations. Hence, it is important to quantify the uncertainties induced in the NODE decoder-output by the uncertainties that afflict the parameters/weights underlying the physical system modeled by the NODE. The quantification of the uncertainties in the NODE-decoder and derived results (i.e., "NODE-responses") of interest require the computation of the sensitivities of the NODE-decoder with respect to the optimized NODE-weights/parameters. However, a "NODE sensitivity analysis" methodology for computing efficiently exact expressions of decoder-sensitivities with respect to the optimized parameters/weights, including with respect to the initial conditions/encoder, does not seem to be available in the literature.

The scope of this work is to present a novel methodology for computing all of the first-order sensitivities, exactly and exhaustively, of the responses of the post-training optimized NODE-decoder with respect to the optimized/trained weights involved in the NODE's decoder, hidden layers, and encoder. The general mathematical representation of the NODE-network considered in this work is presented in Section 2. As a specific illustrative paradigm application, Section 3 presents the NODE conceptual representation of the Nordheim-Fuchs phenomenological reactor dynamics/safety model [17,18]. This paradigm illustrative model has been chosen because it is representative of typical NODE-applications while admitting closed-form analytical solutions for the quantities of interest, including the functions describing the hidden layers, encoder, decoder, and sensitivities of decoder responses. Section 4 presents the Mathematical Framework of the novel "First-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations (1st-CASAM-NODE)." Section 5 illustrates the application of the 1st-CASAM-NODE methodology to compute all of the first-order sensitivities of Nordheim-Fuchs model responses with respect to the underlying parameters. Specifically, Subsections 5.1 through 5.4, respectively, illustrate the application of the 1st-CASAM-NODE methodology for computing the first-order sensitivities with respect to the underlying model parameters and initial conditions of the following responses: (i) the reactor's flux; (ii) the reactor's energy release; (iii) the reactor's temperature; and (iv) the reactor's thermal conductivity. Using the "energy-released" response as a paradigm, Subsection 5.5 illustrates an alternative path for computing first-order sensitivities by applying the "First-Order Feature Adjoint Sensitivity Analysis Methodology for Nonlinear Systems (1st-FASAM-N)" [19], which is the most efficient procedure for computing first-order sensitivities, but which may require the construction of a dedicated neural net for this purpose.

2. Neural Ordinary Differential Equations (NODE): Basic Properties and Uses

A general mathematical representation of a NODE-network is provided by the following system of so-called "augmented" equations:

$$\frac{d\mathbf{h}(t)}{dt} = \mathbf{f}\left[\mathbf{h}(t);\boldsymbol{\theta};t\right], \; t > 0, \qquad (1)$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

$$\mathbf{h}(t_0) = \mathbf{h}_e(\mathbf{x}, \mathbf{w}), \ at\ t = t_0, \tag{2}$$

$$\mathbf{r}(t_f) = \mathbf{h}_d\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right], \ at\ t = t_f, \tag{3}$$

where:

(i) The quantity t is a time-like independent variable which parameterizes the dynamics of the hidden/latent neuron units; the initial value is denoted as t_0 (which can be considered to be an initial measurement time) while the stopping value is denoted as t_f (which can be considered to be the next measurement time).

(ii) The TH-dimensional vector-valued function $\mathbf{h}(t) \triangleq \left[h_1(t), ..., h_{TH}(t)\right]^\dagger$ represents the hidden/latent neural networks. In this work, all vectors are considered to be column vectors and the dagger "$\dagger$" symbol will be used to denote "transposition." The symbol "$\triangleq$" signifies "is defined as" or, equivalently, "is by definition equal to."

(iii) The TH-dimensional vector-valued nonlinear function $\mathbf{f}\left[\mathbf{h}(t); \boldsymbol{\theta}; t\right] \triangleq \left[f_1(\mathbf{h}; \boldsymbol{\theta}; t), ..., f_{TH}(\mathbf{h}; \boldsymbol{\theta}; t)\right]^\dagger$ models the dynamics of the latent neurons with learnable scalar adjustable weights represented by the components of the vector $\boldsymbol{\theta} \triangleq \left[\theta_1, ..., \theta_{TW}\right]^\dagger$, where TW denotes the total number of adjustable weights in all of the latent neural nets.

(iv) The TH-dimensional vector-valued function $\mathbf{h}_e(\mathbf{x}, \mathbf{w}) \triangleq \left\{h_1^e(\mathbf{x}, \mathbf{w}), ..., h_{TH}^e(\mathbf{x}, \mathbf{w})\right\}^\dagger$ represents the "encoder" which is characterized by "inputs" $\mathbf{x} \triangleq \left[x_1, ..., x_{TI}\right]^\dagger$ and "learnable" scalar adjustable weights $\mathbf{w} \triangleq \left[w_1, ..., w_{TEW}\right]^\dagger$, where TI denotes the total number of "inputs" and TEW denotes the total number of "learnable encoder weights" that define the "encoder."

(v) The TR-dimensional vector-valued function $\mathbf{r}(t_f) \triangleq \left\{r_1\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right], ..., r_{TR}\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right]\right\}^\dagger = \mathbf{h}_d\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right]$ represents the vector of "system responses." The vector-valued function $\mathbf{h}_d\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right] \triangleq \left\{h_1^d\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right], ..., h_{TR}^d\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right]\right\}^\dagger$ represents the "decoder" with learnable scalar adjustable weights, which are represented by the components of the vector $\boldsymbol{\varphi} \triangleq \left[\varphi_1, ..., \varphi_{TD}\right]^\dagger$, where TD denotes the total number of adjustable weights that characterize the "decoder." Each component $r_n\left[\mathbf{h}(t_f); \boldsymbol{\varphi}\right]$ can be epresented in integral form as follows:

$$r_n(\mathbf{h}; \boldsymbol{\varphi}) = \int_{t_0}^{t_f} h_n^d\left[\mathbf{h}(t); \boldsymbol{\varphi}\right] \delta(t - t_f)\, dt; \ n = 1, ..., TR. \tag{4}$$

The weights of the NODE are adjusted/calibrated by "training" the NODE, using gradients of a scalar loss functional, denoted as $L\left[\mathbf{h}(t); \boldsymbol{\theta}; t\right]$ which is designed to represent the deviations/discrepancies between the responses/outputs of the NODE and the "true" values obtained from measurements (or by other means, independently of the NODE). There are several methods for accomplishing this "training," all of which require that the functions underlying the NODE, i.e., $\mathbf{h}(t)$, $\mathbf{f}\left[\mathbf{h}(t); \boldsymbol{\theta}; t\right]$, $\mathbf{h}_e(\mathbf{x}, \mathbf{w})$ and $\mathbf{h}_d\left[\mathbf{h}(t); \boldsymbol{\varphi}\right]$ be differentiable with respect to their arguments. For complex systems, involving many parameters, the so-called "adjoint method" [13–15] offers an optimal compromise between memory requirements and computational intensity. This method computes the required gradients of the loss function by evaluating the following integral:

$$\frac{\partial L}{\partial \boldsymbol{\theta}} = \int_{t_0}^{t_f} \left[\mathbf{a}(t)\right]^\dagger \frac{\partial \mathbf{f}\left[\mathbf{h}(t); \boldsymbol{\theta}; t\right]}{\partial \boldsymbol{\theta}} dt, \tag{5}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

where the so-called "adjoint function" $\mathbf{a}(t) \triangleq \left[a_1(t),...,a_{TH}(t)\right]^\dagger$ satisfies the following "adjoint equation" computed backwards in time:

$$\frac{d\mathbf{a}(t)}{dt} = -\left[\mathbf{a}(t)\right]^\dagger \frac{\partial \mathbf{f}\left[\mathbf{h}(t);\boldsymbol{\theta};t\right]}{\partial \mathbf{h}}, \; t > 0, \tag{6}$$

$$\mathbf{a}(t) = \frac{\partial L\left[\mathbf{h}(t);\boldsymbol{\theta};t\right]}{\partial \mathbf{h}}, \; at \; t = t_f \, . \tag{7}$$

After the "training" of the NODE has been accomplished, the various "weights" will have been assigned "optimal" values which will have minimized the chosen loss functional $L\left[\mathbf{h}(t);\boldsymbol{\theta};t\right]$. These "optimal" values will be denoted using a superscript "zero" as follows: $\boldsymbol{\theta}^0 \triangleq \left[\theta_1^0,...,\theta_{TW}^0\right]^\dagger$ and $\mathbf{w}^0 \triangleq \left[w_1^0,...,w_{TEW}^0\right]^\dagger$. These optimal values are used to compute the optimal values for the system responses, which will be denoted as $r_n^0\left[\mathbf{h}(t_f);\boldsymbol{\varphi}\right]$. However, since the physical parameters and the initial conditions underlying the actual physical system (which is represented by the optimized NODE) are not known exactly (because they are actually subject to uncertainties), it follows that the optimal values obtained for the weights are actually just nominal values that are used to compute the nominal/optimal response-values $r_n^0\left[\mathbf{h}(t_f);\boldsymbol{\varphi}\right]$. The uncertainties in the various weights and initial conditions will induce uncertainties in the system responses, which can be computed deterministically by using the well-known "propagation of errors" methodology, originally proposed by Tuckey [20] and subsequently extended to sixth-order by Cacuci [21].

3. Illustrative Paradigm Application: NODE Conceptual Modeling of the Nordheim-Fuchs Phenomenological Reactor Dynamics/Safety Model

The Nordheim-Fuchs phenomenological model [17,18] describes a short-time self-limiting power transient in a nuclear reactor system having a negative temperature coefficient in which a large amount of reactivity is suddenly inserted, either intentionally or by accident. The response of such a reactor system can be estimated by considering that the reactivity insertion is sufficiently large and the time-span of the transient phenomena under consideration is of the order of the life-time of prompt-neutrons. For such short times, the effects of delayed neutrons and the local spatial variations of the neutron distribution in the reactor can be neglected, and the heat generated during the transient remains within the reactor. Using the notation of Lamarsh [17], the Nordheim-Fuchs paradigm model describing such a self-limiting power transient comprises the following balance equations:

1. The time-dependent neutron balance (point kinetics) equation for the neutron flux $\varphi(t)$:

$$\frac{d\varphi(t)}{dt} = \frac{k(t)-1}{l_p}\varphi(t), \; t > 0, \tag{8}$$

$$\varphi(0) = \varphi_0, \; t = 0, \tag{9}$$

where l_p denotes the prompt-neutron lifetime, $k(t)$ denotes the reactor's multiplication factor, and φ_0 denotes the initial (i.e., extant flux) prior to initiating the transient at time $t = 0$.

2. The energy production equation:

$$E(t) = \gamma \Sigma_f \int_0^t \varphi(x)\, dx, \tag{10}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

where γ denotes the recoverable energy per fission; $\Sigma_f \triangleq \sigma_f N_f$ denotes the reactor's effective macroscopic fission cross section, where σ_f denotes the reactor's equivalent microscopic fission cross section while N_f denotes the reactor's equivalent atomic number density.

3. The energy conservation equation:

$$c_p\left[T(t)-T_0\right]=E(t), \tag{11}$$

where $E(t)$ denotes the total energy released (per cm³) at time t in the reactor since the onset of reactivity change; c_p denotes the specific heat (per cm³) of the reactor.

4. The reactivity-temperature feedback equation: $k(t)=k_0-\alpha_T k_0\left[T(t)-T_0\right]$, where $k_0 \triangleq k(0)\geq 1$ denotes the changed multiplication factor following the reactivity insertion at $t=0$, α_T denotes the magnitude of the negative temperature coefficient, $T(t)$ denotes the reactor's temperature, and T_0 denotes the reactor's initial temperature at time $t=0$. For illustrating the application of the 1st-FASAM methodology, it suffices to consider the special case of a "prompt critical transient", when the reactor becomes prompt critical after the reactivity insertion, i.e., when $k_0=1$, so that the reactivity-temperature feedback equation takes on the following particular form:

$$k(t)=1-\alpha_T\left[T(t)-T_0\right]. \tag{12}$$

Equations (8)–(12) can be transformed into the following system of nonlinear differential equations:

$$\frac{d\varphi(t)}{dt}=-\frac{\alpha_T}{l_p c_p}E(t)\varphi(t),\ t>0. \qquad \varphi(0)=\varphi_0,\ t=0 \tag{13}$$

$$\frac{dE(t)}{dt}=\gamma\sigma_f N_f\varphi(t),\ E(0)=0, \tag{14}$$

$$\frac{dT(t)}{dt}=\frac{\gamma\sigma_f N_f}{c_p}\varphi(t);\ \ T(0)=T_0. \tag{15}$$

The Nordheim-Fuchs model described by Eqs. (13)–(15) can be solved analytically to obtain exact closed-form expression for the state functions $\varphi(t)$, $E(t)$, and $T(t)$, as follows:

(i) Eliminating the function $\varphi(t)$ from Eqs. (13) and (14) yields a nonlinear differential equation which can be integrated directly to obtain the following relation:

$$\varphi(t)=-\frac{\alpha_T}{2l_p c_p\gamma\sigma_f N_f}E^2(t)+\varphi_0. \tag{16}$$

(ii) Using Eq. (16) in Eq. (14) yields the following nonlinear equation for the released energy $E(t)$:

$$\frac{dE(t)}{dt}=-\frac{\alpha_T}{2l_p c_p}E^2(t)+\varphi_0\gamma\sigma_f N_f,\ E(0)=0. \tag{17}$$

The closed-form solution of Eq. (17) has the following form:

$$E(t)=K_1(\boldsymbol{\alpha})\tanh\left[tK_2(\boldsymbol{\alpha})\right], \tag{18}$$

where:

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

$$K_1(\boldsymbol{\alpha}) \triangleq \left[\frac{2\varphi_0\gamma\sigma_f N_f l_p c_p}{\alpha_T}\right]^{1/2}; \quad K_2(\boldsymbol{\alpha}) \triangleq \left[\frac{\alpha_T\varphi_0\gamma\sigma_f N_f}{2l_p c_p}\right]^{1/2}. \tag{19}$$

(iii) Replacing Eq. (18) into Eq. (16) yields the following closed-form expression for $\varphi(t)$:

$$\varphi(t) = \varphi_0\left\{1 - \tanh^2\left[tK_2(\boldsymbol{\alpha})\right]\right\} = \frac{\varphi_0}{\cosh^2\left[tK_2(\boldsymbol{\alpha})\right]}. \tag{20}$$

(iv) Replacing Eq. (18) into Eq. (11) yields the following closed-form expression for $T(t)$:

$$T(t) = T_0 + \frac{K_1(\boldsymbol{\alpha})}{c_p}\tanh\left[tK_2(\boldsymbol{\alpha})\right]. \tag{21}$$

The typical results of interest (called "model response") for the Nordheim-Fuchs model are as follows:

(i) The neutron flux $\varphi(\tau)$ in the reactor at a "final time" instance denoted as $t=\tau$, after the initiation at $t=0$ of the prompt-critical power transient, which can be defined mathematically as follows:

$$\varphi(\tau) = \int_0^\tau \varphi(t)\delta(t-\tau)\,dt, \tag{22}$$

(ii) The total energy per cm^3, $E(\tau)$, released at a user-chosen "final time" instance denoted as $t=\tau$, after the initiation at $t=0$ of the prompt-critical power transient, which can be defined mathematically as follows:

$$E(\tau) = \int_0^\tau E(t)\delta(t-\tau)\,dt, \tag{23}$$

where $\delta(t-\tau)$ denotes the Dirac-delta functional.

(iii) The reactor's temperature $T(\tau)$ at a "final time" instance denoted as $t=\tau$ after the initiation at $t=0$ of the prompt-critical power transient, which can be defined mathematically as follows:

$$T(\tau) = \int_0^\tau T(t)\delta(t-\tau)\,dt, \tag{24}$$

Comparing the structure of the Nordheim-Fuchs model, cf. Eqs. (13)–(15), to the generic structure of a NODE, cf. Eqs. (1) and (2), indicates the following correspondences:

$$\mathbf{h}(t) \triangleq \left[h_1(t),\ldots,h_{TH}(t)\right]^\dagger \triangleq \left[\varphi(t),E(t),T(t)\right]^\dagger; \quad TH=3; \tag{25}$$

$$\boldsymbol{\theta} \triangleq \left[\theta_1,\ldots,\theta_{TW}\right]^\dagger \triangleq \left(\alpha_T,l_p,c_p,\gamma,\sigma_f,N_f\right)^\dagger; \ \mathbf{x} \triangleq \left[x_1,x_2\right]^\dagger \triangleq \left(\varphi_0,T_0\right)^\dagger; \ TW=6,\ TI=2. \tag{26}$$

$$f_1(\mathbf{h};\boldsymbol{\theta};t) \triangleq -\frac{\alpha_T}{l_p c_p}E(t)\varphi(t) \triangleq -\frac{\theta_1}{\theta_2\theta_3}h_1(t)h_2(t) \tag{27}$$

$$f_2(\mathbf{h};\boldsymbol{\theta};t) \triangleq \gamma\sigma_f N_f\varphi(t) \triangleq \theta_4\theta_5\theta_5 h_1(t); \tag{28}$$

$$f_3(\mathbf{h};\boldsymbol{\theta};t) \triangleq \frac{\gamma\sigma_f N_f}{c_p}\varphi(t) \triangleq \frac{\theta_4\theta_5\theta_6}{\theta_3}h_1(t). \tag{29}$$

The actual values of the components of the vectors $\boldsymbol{\theta}$ and $\mathbf{x}$ are unknown even after having trained the NODE, since the actual values of the parameters underlying the Nordheim-Fuchs model are experimentally-measured and are thus subject to uncertainties. However, the nominal values of these parameters are considered to be known, and are considered to be exactly reproducible by the "trained" NODE; these nominal values will be denoted using a superscript "zero," as follows:

$$\boldsymbol{\theta}^0 \triangleq \left[\theta_1^0,...,\theta_6^0\right]^\dagger \triangleq \left(\alpha_T^0, l_p^0, c_p^0, \gamma^0, \sigma_f^0, N_f^0\right)^\dagger; \ \mathbf{x}^0 \triangleq \left[x_1^0, x_2^0, x_3^0\right]^\dagger \triangleq \left[\varphi_0^0, 0; T_0^0\right]^\dagger. \tag{30}$$

Consequently, the exact values of the functions $\mathbf{h}(t) \triangleq \left[h_1(t), h_2(t), h_3(t)\right]^\dagger \triangleq \left[\varphi(t), E(t), T(t)\right]^\dagger$ are unknown but their nominal values $\mathbf{h}^0(t) \triangleq \left[h_1^0(t), h_2^0(t), h_3^0(t)\right]^\dagger \triangleq \left[\varphi^0(t), E^0(t), T^0(t)\right]^\dagger$ are known after having solved Eqs. (13)–(15) at the nominal values $\left(\boldsymbol{\theta}^0, \mathbf{x}^0\right)$.

The NODE-representations, cf. Eq. (4), of the responses considered in Eqs. (23)–(24)have the following expressions, respectively:

$$r_1(\mathbf{h}) = \int_{t_0=0}^{t_f} h_1(t)\delta\left(t-t_f\right)dt = \varphi\left(t_f\right); \tag{31}$$

$$r_2(\mathbf{h}) = \int_{t_0=0}^{t_f} h_2(t)\delta\left(t-t_f\right)dt = E\left(t_f\right); \tag{32}$$

$$r_3(\mathbf{h}) = \int_{t_0=0}^{t_f} h_3(t)\delta\left(t-t_f\right)dt = T\left(t_f\right). \tag{33}$$

To illustrate the efficient computation of responses involving decoders having their own parameters/weights, the thermal conductivity of the conceptual material of the Nordheim-Fuchs reactor model will be considered to be a "decoder" response having the following expression:

$$\begin{gathered} r_4(\mathbf{h};\boldsymbol{\varphi}) = \int_{t_0}^{t_f} h_4^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right]\delta\left(t-t_f\right)dt; \\ h_4^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right] \triangleq k(T) = \varphi_1 + \varphi_2 h_3(t) + \varphi_3 h_3^2(t) = \varphi_1 + \varphi_2 T(t) + \varphi_3 T^2(t). \end{gathered} \tag{34}$$

4. First-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations (1st-CASAM-NODE): Mathematical Framework

At the optimal/nominal parameter values, the optimal/nominal solution $\mathbf{h}^0(t)$ will satisfy the following forms of Eqs. (1) and (2):

$$\frac{d\mathbf{h}^0(t)}{dt} = \mathbf{f}\left[\mathbf{h}^0(t);\boldsymbol{\theta}^0;t\right], \ t>0, \tag{35}$$

$$\mathbf{h}^0(t_0) = \mathbf{h}_e\left(\mathbf{x}^0, \mathbf{w}^0\right), \ at \ t = t_0. \tag{36}$$

Furthermore, the vector of optimal/nominal response will have components that are obtained by using the nominal values for the respective functions and parameters, i.e.:

$$r_n^0\left(\mathbf{h}^0;\boldsymbol{\varphi}^0\right) = \int_{t_0}^{t_f} h_n^d\left[\mathbf{h}^0(t);\boldsymbol{\varphi}^0\right]\delta\left(t-t_f\right)dt; \ n = 1,...,TR. \tag{37}$$

The known nominal values $\mathbf{x}^0$ of the initial conditions will differ from the true but unknown values $\mathbf{x}$ of the initial conditions by variations denoted as $\delta\mathbf{x} \triangleq \mathbf{x} - \mathbf{x}^0$. Furthermore, the known nominal values $\mathbf{w}^0$ of the weights characterizing the encoder will differ from the true but unknown values $\mathbf{w}$ of the respective weights by variations denoted as $\delta\mathbf{w} \triangleq \mathbf{w} - \mathbf{w}^0$. Similarly, the nominal values $\boldsymbol{\theta}^0$ and $\boldsymbol{\varphi}^0$, respectively, will differ by variations $\delta\boldsymbol{\theta} \triangleq \boldsymbol{\theta} - \boldsymbol{\theta}^0$ and $\delta\boldsymbol{\varphi} \triangleq \boldsymbol{\varphi} - \boldsymbol{\varphi}^0$, respectively, from the corresponding true but unknown values $\boldsymbol{\theta}$ and $\boldsymbol{\varphi}$. Since the forward state functions $\mathbf{h}(t)$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

are related to the weights and initial conditions through Eqs. (1) and (2), it follows that the variations in these weights and initial conditions will induce corresponding variations $\mathbf{v}^{(1)}(t) \triangleq \left[\delta h_1(t),...,\delta h_{TH}(t)\right]^\dagger$ around the nominal solution $\mathbf{h}^0(t)$. In turn, the variations $\delta\boldsymbol{\varphi}$ and $\mathbf{v}^{(1)}(t)$ will induce variations $\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)};\delta\boldsymbol{\varphi}\right)$ in the system's response.

The 1st-CASAM-NODE methodology for computing the first-order sensitivities of the response with respect to the model's weights and initial conditions will be established by following the same principles as those underlying the 1st-CASAM-N methodology [22], which commence by noting that Cacuci [23] has shown that the most general definition of the sensitivity of an operator-valued model response $\mathbf{R}(\mathbf{e})$ with respect to variations $\delta\mathbf{e}$ in the model parameters and state functions in a neighborhood around the nominal functions and parameter values $\mathbf{e}^0$, is given by the 1st-order Gateaux- (G-) variation, which will be denoted as $\delta\mathbf{R}\left(\mathbf{e}^0;\delta\mathbf{e}\right)$ and is defined as follows:

$$\delta\mathbf{R}\left(\mathbf{e}^0;\delta\mathbf{e}\right) \triangleq \left\{\frac{d}{d\varepsilon}\left[\mathbf{R}\left(\mathbf{e}^0+\varepsilon\delta\mathbf{e}\right)\right]\right\}_{\varepsilon=0} \triangleq \lim_{\varepsilon\to 0}\frac{\mathbf{R}\left(\mathbf{e}^0+\varepsilon\delta\mathbf{e}\right)-\mathbf{R}\left(\mathbf{e}^0\right)}{\varepsilon}, \tag{38}$$

for a scalar ε and for all (i.e., arbitrary) vectors $\delta\mathbf{e}$ in a neighborhood $\left(\mathbf{e}^0+\varepsilon\delta\mathbf{e}\right)$ around $\mathbf{e}^0$. The G-variation $\delta\mathbf{R}\left(\mathbf{e}^0;\delta\mathbf{e}\right)$ is an operator defined on the same domain as $\mathbf{R}(\mathbf{e})$ and has the same range as $\mathbf{R}(\mathbf{e})$. The G-variation $\delta\mathbf{R}\left(\mathbf{e}^0;\delta\mathbf{e}\right)$ satisfies the relation: $\mathbf{R}\left(\mathbf{e}^0+\varepsilon\delta\mathbf{e}\right)-\mathbf{R}\left(\mathbf{e}^0\right)=\delta\mathbf{R}\left(\mathbf{e}^0;\delta\mathbf{e}\right)+\Delta(\delta\mathbf{e})$, with $\lim_{\varepsilon\to 0}\left[\Delta(\varepsilon\delta\mathbf{e})\right]/\varepsilon=0$. When the G-variation $\delta\mathbf{R}\left(\mathbf{e}^0;\delta\mathbf{e}\right)$ is linear in the variation $\delta\mathbf{e}$, it can be written in the form $\delta\mathbf{R}\left(\mathbf{e}^0;\delta\mathbf{e}\right)=\left\{\partial\mathbf{R}/\partial\mathbf{e}\right\}_{\mathbf{e}^0}\delta\mathbf{e}$, where $\left\{\partial\mathbf{R}/\partial\mathbf{e}\right\}_{\mathbf{e}^0}$ denotes the first-order G-derivative of $\mathbf{R}(\mathbf{e})$ with respect to $\mathbf{e}$ evaluated at $\mathbf{e}^0$.

Applying the definition provided in Eq. (38) to Eq. (4) yields the following expression for the first-order G-variation $\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)};\delta\boldsymbol{\varphi}\right)$ of the response $r_n(\mathbf{h};\boldsymbol{\varphi})$:

$$\begin{aligned}\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)};\delta\boldsymbol{\varphi}\right)&=\left\{\frac{d}{d\varepsilon}\int_{t_0}^{t_f} h_n^d\left[\mathbf{h}^0(t)+\varepsilon\mathbf{v}^{(1)}(t);\boldsymbol{\varphi}^0+\varepsilon\delta\boldsymbol{\varphi}\right]\delta\left(t-t_f\right)dt;\right\}_{\varepsilon=0}\\ &=\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\delta\boldsymbol{\varphi}\right)\right\}_{dir}+\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)}\right)\right\}_{ind};\quad n=1,...,TR.\end{aligned} \tag{39}$$

where $\mathbf{v}^{(1)} \triangleq \left[v_1^{(1)}(t),...,v_{TH}^{(1)}(t)\right]^\dagger$ and:

$$\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\delta\boldsymbol{\varphi}\right)\right\}_{dir} \triangleq \int_{t_0}^{t_f}\delta\left(t-t_f\right)\left\{\frac{\partial h_n^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right]}{\partial\boldsymbol{\varphi}}\right\}_{\left(\mathbf{h}^0;\boldsymbol{\varphi}^0\right)}\delta\boldsymbol{\varphi}\;dt\;, \tag{40}$$

$$\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)}\right)\right\}_{ind} \triangleq \int_{t_0}^{t_f}\delta\left(t-t_f\right)\left\{\frac{\partial h_n^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right]}{\partial\mathbf{h}(t)}\right\}_{\left(\mathbf{h}^0;\boldsymbol{\varphi}^0\right)}\mathbf{v}^{(1)}(t)\,dt\;. \tag{41}$$

This, the quantity $\left\{\partial h_n^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right]/\partial\boldsymbol{\varphi}\right\}_{\left(\mathbf{h}^0;\boldsymbol{\varphi}^0\right)}$ in Eq. (40) denotes the partial G-derivatives of the response $h_n^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right]$ with respect to the decoder weights $\boldsymbol{\varphi} \triangleq \left[\varphi_1,...,\varphi_{TD}\right]^\dagger$, evaluated at the nominal values $\left(\mathbf{h}^0;\boldsymbol{\varphi}^0\right)$. The quantity $\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\delta\boldsymbol{\varphi}\right)\right\}_{dir}$ is called the "direct-effect term" because it arises directly from parameter variations $\delta\boldsymbol{\varphi}$ and can be computed directly using the nominal values $\left(\mathbf{h}^0;\boldsymbol{\varphi}^0\right)$. The quantity $\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\delta\mathbf{h};\delta\boldsymbol{\varphi}\right)\right\}_{ind}$ is called the "indirect-effect term" because it arises indirectly, through the variations $\mathbf{v}^{(1)}(t)$ in the hidden state functions $\mathbf{h}(t)$. The indirect-effect term can be quantified only after having determined the variations $\mathbf{v}^{(1)}(t)$, which are caused by the variations $\delta\mathbf{x}$, $\delta\mathbf{w}$ and $\delta\boldsymbol{\theta}$.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

The first-order relationships between the variations $\mathbf{v}^{(1)}(t)$, $\delta\mathbf{x}$, $\delta\mathbf{w}$ and $\delta\boldsymbol{\theta}$ are obtained by computing the first-order G-variation of Eqs. (1) and (2), which are obtained, by definition, as follows:

$$\left\{\frac{d}{d\varepsilon}\left[\frac{d}{dt}\left(\mathbf{h}^0+\varepsilon\mathbf{v}^{(1)}\right)\right]\right\}_{\varepsilon=0}=\left\{\frac{d}{d\varepsilon}\mathbf{f}\left[\mathbf{h}^0+\varepsilon\mathbf{v}^{(1)};\boldsymbol{\theta}^0+\varepsilon\delta\boldsymbol{\theta};t\right]\right\}_{\varepsilon=0}, \tag{42}$$

$$\left\{\frac{d}{d\varepsilon}\left[\mathbf{h}^0(t_0)+\varepsilon\mathbf{v}^{(1)}(t_0)\right]\right\}_{\varepsilon=0}=\left\{\frac{d}{d\varepsilon}\left[\mathbf{h}_e\left(\mathbf{x}^0+\varepsilon\delta\mathbf{x},\mathbf{w}^0+\varepsilon\delta\mathbf{w}\right)\right]\right\}_{\varepsilon=0}. \tag{43}$$

Carrying out the operations indicated in Eqs. (42) and (43) yields the following system of equations:

$$\frac{d\mathbf{v}^{(1)}(t)}{dt}-\left\{\frac{\partial\mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial\mathbf{h}}\right\}_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)}\mathbf{v}^{(1)}(t)=\left\{\frac{\partial\mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}\right\}_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)}\delta\boldsymbol{\theta}, \tag{44}$$

$$\mathbf{v}^{(1)}(t_0)=\left\{\frac{\partial\mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial\mathbf{x}}\right\}_{\left(\mathbf{x}^0,\mathbf{w}^0\right)}\delta\mathbf{x}+\left\{\frac{\partial\mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial\mathbf{w}}\right\}_{\left(\mathbf{x}^0,\mathbf{w}^0\right)}\delta\mathbf{w}. \tag{45}$$

where:

$$\frac{\partial\mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial\mathbf{h}}\triangleq\begin{pmatrix}\partial f_1/\partial h_1 & \bullet & \partial f_1/\partial h_{TH}\\ \bullet & \bullet & \bullet\\ \partial f_{TH}/\partial h_1 & \bullet & \partial f_{TH}/\partial h_{TH}\end{pmatrix}_{TH\times TH}, \tag{46}$$

$$\frac{\partial\mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial\boldsymbol{\theta}}\triangleq\begin{pmatrix}\partial f_1/\partial\theta_1 & \bullet & \partial f_1/\partial\theta_{TW}\\ \bullet & \bullet & \bullet\\ \partial f_{TH}/\partial\theta_1 & \bullet & \partial f_{TH}/\partial\theta_{TW}\end{pmatrix}_{TH\times TW}, \tag{47}$$

$$\frac{\partial\mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial\mathbf{x}}\triangleq\begin{pmatrix}\partial h_1^e/\partial x_1 & \bullet & \partial h_1^e/\partial x_{TI}\\ \bullet & \bullet & \bullet\\ \partial h_{TH}^e/\partial x_1 & \bullet & \partial h_{TH}^e/\partial x_{TI}\end{pmatrix}_{TH\times TI}, \tag{48}$$

$$\frac{\partial\mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial\mathbf{w}}\triangleq\begin{pmatrix}\partial h_1^e/\partial w_1 & \bullet & \partial h_1^e/\partial w_{TEW}\\ \bullet & \bullet & \bullet\\ \partial h_{TH}^e/\partial w_1 & \bullet & \partial h_{TH}^e/\partial w_{TEW}\end{pmatrix}_{TH\times TEW}. \tag{49}$$

The system comprising Eqs. (44) and (45) is called the "1st-Level Variational Sensitivity System" (1st-LVSS), and its solution, $\mathbf{v}^{(1)}(t)$, is called the "1st-level variational sensitivity function." Note that the 1st-LVSS would need to be solved anew for each component of the variations $\delta\mathbf{x}$, $\delta\mathbf{w}$ and $\delta\boldsymbol{\theta}$, which would be prohibitively expensive computationally.

The need for solving the 1st-LVSS can be avoided if the indirect-effect term defined in Eq. (41) could be expressed in terms of a "right-hand side" that does not involve the function $\mathbf{v}^{(1)}(t)$. This goal can be achieved by expressing the right-side of Eq. (41) in terms of the solutions of the "1st-Level Adjoint Sensitivity System (1st-LASS)," the construction of which requires the introduction of adjoint operators. Adjoint operators can be defined in Banach spaces but are most useful in Hilbert spaces. For the NODE considered in this work, the appropriate Hilbert space is defined on the domain $\Omega_t\triangleq\left[t_0,t_f\right]$ and will be denoted as $\mathsf{H}_1(\Omega_t)$, so that $\mathbf{v}^{(1)}(t)\in\mathsf{H}_1(\Omega_t)$. In $\mathsf{H}_1(\Omega_t)$, the inner product of two vectors in $\mathbf{u}^{(a)}(t)\in\mathsf{H}_1(\Omega_t)$ and $\mathbf{u}^{(b)}(t)\in\mathsf{H}_1(\Omega_t)$ will be denoted as $\left\langle\mathbf{u}^{(a)},\mathbf{u}^{(b)}\right\rangle_1$, and is defined as follows:

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

$$\left\langle \mathbf{u}^{(a)},\mathbf{u}^{(b)}\right\rangle_1 \triangleq \left\{\int_{t_0}^{t_f} \mathbf{u}^{(a)}(t)\bullet\mathbf{u}^{(b)}(t)\,dt\right\}_{\left(\mathbf{x}^0;\boldsymbol{\theta}^0;\mathbf{w}^0;\boldsymbol{\varphi}^0\right)}, \tag{50}$$

where the "dot" indicates the "scalar product of two vectors" defined as follows: $\mathbf{u}^{(a)}(t)\bullet\mathbf{u}^{(b)}(t) \triangleq \left[\mathbf{u}^{(a)}(t)\right]^\dagger \mathbf{u}^{(b)}(t) \triangleq \sum_{i=1}^{TH} u_i^{(a)}(t)u_i^{(b)}(t) = \left[\mathbf{u}^{(b)}(t)\right]^\dagger \mathbf{u}^{(a)}(t)$.

The next step is to form the inner product of Eq. (44) with a vector $\mathbf{a}^{(1)}(t) \triangleq \left[a_1^{(1)}(t),\ldots,a_{TD}^{(1)}(t)\right] \in \mathsf{H}_1(\Omega_t)$, where the superscript "(1)" indicates "1st-Level", to obtain the following relationship:

$$\left\langle \mathbf{a}^{(1)}(\mathbf{x}),\ \frac{d\mathbf{v}^{(1)}(t)}{dt} - \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \mathbf{h}}\right]_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)} \mathbf{v}^{(1)}(t) \right\rangle_1 = \left\langle \mathbf{a}^{(1)}(\mathbf{x}),\ \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right]_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)} \boldsymbol{\delta\theta} \right\rangle_1. \tag{51}$$

Using the definition of the adjoint operator in $\mathsf{H}_1(\Omega_t)$, the left-side of Eq. (51) is transformed as follows, after integrating by parts over the independent variable t:

$$\begin{aligned} &\int_{t_0}^{t_f} \mathbf{a}^{(1)}(t)\bullet\frac{d\mathbf{v}^{(1)}(t)}{dt}dt - \int_{t_0}^{t_f} \mathbf{a}^{(1)}(t)\bullet\left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \mathbf{h}}\right]_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)} \mathbf{v}^{(1)}(t)dt = \mathbf{a}^{(1)}(t_f)\bullet\mathbf{v}^{(1)}(t_f) \\ &-\mathbf{a}^{(1)}(t_0)\bullet\mathbf{v}^{(1)}(t_0) + \int_{t_0}^{t_f} \mathbf{v}^{(1)}(t)\bullet\left\{-\frac{d\mathbf{a}^{(1)}(t)}{dt} - \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \mathbf{h}}\right]^\dagger_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)} \mathbf{a}^{(1)}(t)\right\}dt. \end{aligned} \tag{52}$$

The last term on the right-side of Eq. (52) is now required to represent the "indirect-effect" term defined in Eq. (41), which is achieved by requiring that the 1st-level adjoint function $\mathbf{a}^{(1)}(t)$ satisfy the following relation:

$$-\frac{d\mathbf{a}^{(1)}(t)}{dt} - \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \mathbf{h}}\right]^\dagger_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)} \mathbf{a}^{(1)}(t) = \left\{\frac{\partial h_n^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right]}{\partial \mathbf{h}(t)}\right\}_{\left(\mathbf{h}^0;\boldsymbol{\varphi}^0\right)} \delta\left(t-t_f\right). \tag{53}$$

The definition of the 1st-level adjoint sensitivity function $\mathbf{a}^{(1)}(t)$ is now completed by requiring it to satisfy (adjoint) "boundary conditions at the final time" $t=t_f$ so as to eliminate the term containing the unknown values $\mathbf{v}^{(1)}(t_f)$ in Eq. (52). This aim is achieved by requiring that

$$\mathbf{a}^{(1)}(t_f) = \mathbf{0}. \tag{54}$$

The system of equations comprising Eqs. (53) and (54) constitute the "1st-Level Adjoint Sensitivity System (1st-LASS)" for the 1st-level adjoint function $\mathbf{a}^{(1)}(t)$. Evidently, the 1st-LASS is independent of parameter variations and needs to be solved just once to obtain the 1st-level adjoint function $\mathbf{a}^{(1)}(t)$. Notably, the 1st-LASS has the same form as the "adjoint equations" used for training the NODE, cf. Eqs. (6) and (7), but with the "response" $\partial h_n^d\left[\mathbf{h}(t);\boldsymbol{\varphi}\right]/\partial\mathbf{h}(t)\delta\left(t-t_f\right)$ being the "source" for the 1st-LASS, whereas the "source" in the "training" of the NODE was the "loss functional" $L\left[\mathbf{h}(t);\boldsymbol{\theta};t\right]/\partial\mathbf{h}\,\delta\left(t-t_f\right)$. Evidently, the 1st-level adjoint sensitivity function $\mathbf{a}^{(1)}(t)$ is the counterpart of the "adjoint function" $\mathbf{a}(t)$ in the "training" of the NODE.

Using the results represented by Eqs. (53), (54), (51), and (41) in Eq. (52) yields the following alternative expression for the "indirect-effect" term, which does not involve the 1st-level variational sensitivity function $\mathbf{v}^{(1)}(t)$ but involves the 1st-level adjoint function $\mathbf{a}^{(1)}(t)$:

$$\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)}\right)\right\}_{ind} = \left\langle \mathbf{a}^{(1)}(\mathbf{x}),\ \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right]_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)} \boldsymbol{\delta\theta} \right\rangle_1 + \mathbf{a}^{(1)}(t_0)\bullet\mathbf{v}^{(1)}(t_0). \tag{55}$$

 doi:10.20944/preprints202407.2613.v1

Using in Eq. (55) the expression provided for $\mathbf{v}^{(1)}(t_0)$ in Eq. (45) yields the following expression for the "indirect-effect" term:

$$\left\{\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)}\right)\right\}_{ind}=\left\langle \mathbf{a}^{(1)}(\mathbf{x}),\ \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right]_{(\mathbf{h}^0,\boldsymbol{\theta}^0)}\delta\boldsymbol{\theta}\right\rangle_1 \\ +\mathbf{a}^{(1)}(t_0)\bullet\left\{\frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial \mathbf{x}}\right\}_{(\mathbf{x}^0,\mathbf{w}^0)}\delta\mathbf{x}+\mathbf{a}^{(1)}(t_0)\bullet\left\{\frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial \mathbf{w}}\right\}_{(\mathbf{x}^0,\mathbf{w}^0)}\delta\mathbf{w}. \tag{56}$$

Replacing the expression obtained in Eq. (55) for the "indirect-effect term" together with the expression of the direct-effect term provided by Eq. (40) into Eq. (39) yields the following expression for the first-order G-variation $\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)};\delta\boldsymbol{\varphi}\right)$ of the response $r_n(\mathbf{h};\boldsymbol{\varphi})$:

$$\delta r_n\left(\mathbf{h}^0;\boldsymbol{\varphi}^0;\mathbf{v}^{(1)};\delta\boldsymbol{\varphi}\right)=\left\{\frac{\partial h_n^d\left[\mathbf{h}(t_f);\boldsymbol{\varphi}\right]}{\partial \boldsymbol{\varphi}}\right\}_{(\mathbf{h}^0;\boldsymbol{\varphi}^0)}\delta\boldsymbol{\varphi}+\mathbf{a}^{(1)}(t_0)\bullet\left\{\frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial \mathbf{w}}\right\}_{(\mathbf{x}^0,\mathbf{w}^0)}\delta\mathbf{w} \\ +\mathbf{a}^{(1)}(t_0)\bullet\left\{\frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial \mathbf{x}}\right\}_{(\mathbf{x}^0,\mathbf{w}^0)}\delta\mathbf{x}+\int_{t_0}^{t_f}\mathbf{a}^{(1)}(t)\bullet\left\{\left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\right]_{(\mathbf{h}^0,\boldsymbol{\theta}^0)}\delta\boldsymbol{\theta}\right\}dt;\quad n=1,\dots,TR. \tag{57}$$

As indicated by the right-side of Eq. (57), the (partial) sensitivities of the response $r_n(\mathbf{h};\boldsymbol{\varphi})$ are provided by the following expressions, all of which are to be evaluated at the nominal values of all functions and parameters/weights:

$$\frac{\partial r_n}{\partial \varphi_i}=\frac{\partial h_n^d\left[\mathbf{h}(t_f);\boldsymbol{\varphi}\right]}{\partial \varphi_i};\ i=1,\dots,TD;\ n=1,\dots,TR; \tag{58}$$

$$\frac{\partial r_n}{\partial w_i}=\mathbf{a}^{(1)}(t_0)\bullet\frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial w_i};\ \ i=1,\dots,TEW;\ \ n=1,\dots,TR; \tag{59}$$

$$\frac{\partial r_n}{\partial x_i}=\mathbf{a}^{(1)}(t_0)\bullet\frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial x_i};\ \ i=1,\dots,TI;\ \ n=1,\dots,TR; \tag{60}$$

$$\frac{\partial r_n}{\partial \theta_i}=\int_{t_0}^{t_f}\mathbf{a}^{(1)}(t)\bullet\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \theta_i}dt;\ i=1,\dots,TW;\ n=1,\dots,TR. \tag{61}$$

5. Illustrative Application of the 1st-CASAM-NODE Methodology to Compute First-Order Sensitivities of Nordheim-Fuchs Model Responses with respect to the Underlying Parameters

The application of the 1st-CASAM-NODE methodology to compute the first-order sensitivities of the responses $r_1(\mathbf{h})$, $r_2(\mathbf{h})$, $r_3(\mathbf{h})$ and $r_4(\mathbf{h})$ with respect to the Nordheim-Fuchs model's parameters and initial conditions will be presented below in Subsections 5.1 through 5.4, respectively. Using the "energy-released" response $r_2(\mathbf{h})=E(t_f)$ as a paradigm, Subsection 5.5 will illustrate an alternative path for computing first-order sensitivities by applying the "First-Order Feature Adjoint Sensitivity Analysis Methodology for Nonlinear Systems (1st-FASAM-N)" [24], which is the most efficient procedure for computing sensitivities, but which may require the construction of a dedicated neural net for this purpose.

5.1. First-Order Sensitivities of the Flux Response $r_1(\mathbf{h})=\varphi(t_f)$

The first-order sensitivity of the response $r_1(\mathbf{h})=\varphi(t_f)$ is provided by the first-order G-differential of the expression in Eq. (31), which is, by definition, obtained as follows:

$$\delta r_1(\mathbf{h};\delta\mathbf{h})=\left\{\frac{d}{d\varepsilon}\left[\int_0^{t_f}\left[\varphi^0(t)+\varepsilon\delta\varphi(t)\right]\delta\left(t-t_f\right)dt\right]\right\}_{\varepsilon=0}=\int_0^{t_f}\delta\varphi(t)\delta\left(t-t_f\right)dt\,. \quad (62)$$

The variation $\delta\varphi(t)$ is the solution of the "First-Level Variational Sensitivity System (1st-LVSS) which is obtained by G-differentiating Eqs. (13)–(15), which yields the following expressions:

$$\left\{\frac{d}{d\varepsilon}\left[\frac{d}{dt}\left(\varphi^0+\varepsilon\delta\varphi\right)\right]\right\}_{\varepsilon=0}=-\left\{\frac{d}{d\varepsilon}\left[\frac{\alpha_T^0+\varepsilon\delta\alpha_T}{\left(l_p^0+\varepsilon\delta l_p\right)\left(c_p^0+\varepsilon\delta c_p\right)}\left(E^0+\varepsilon\delta E\right)\left(\varphi^0+\varepsilon\delta\varphi\right)\right]\right\}_{\varepsilon=0}, \quad (63)$$

$$\left\{\frac{d}{d\varepsilon}\left[\frac{d}{dt}\left(E^0+\varepsilon\delta E\right)\right]\right\}_{\varepsilon=0}=\left\{\frac{d}{d\varepsilon}\left[\left(\gamma^0+\varepsilon\delta\gamma\right)\left(\sigma_f^0+\varepsilon\delta\sigma_f\right)\left(N_f^0+\varepsilon\delta N_f\right)\left(\varphi^0+\varepsilon\delta\varphi\right)\right]\right\}_{\varepsilon=0}, \quad (64)$$

$$\left\{\frac{d}{d\varepsilon}\left[\frac{d}{dt}\left(T^0+\varepsilon\delta T\right)\right]\right\}_{\varepsilon=0}=\left\{\frac{d}{d\varepsilon}\left[\frac{\left(\gamma^0+\varepsilon\delta\gamma\right)\left(\sigma_f^0+\varepsilon\delta\sigma_f\right)\left(N_f^0+\varepsilon\delta N_f\right)}{\left(c_p^0+\varepsilon\delta c_p\right)}\left(\varphi^0+\varepsilon\delta\varphi\right)\right]\right\}_{\varepsilon=0}, \quad (65)$$

$$\left\{\frac{d}{d\varepsilon}\left[\varphi^0(t)+\varepsilon\delta\varphi(t)\right]_{t=0}\right\}_{\varepsilon=0}=\left\{\frac{d}{d\varepsilon}\left(\varphi_0^0+\varepsilon\delta\varphi_0\right)\right\}_{\varepsilon=0}, \quad (66)$$

$$\left\{\frac{d}{d\varepsilon}\left[E^0(t)+\varepsilon\delta E(t)\right]_{t=0}\right\}_{\varepsilon=0}=0, \quad (67)$$

$$\left\{\frac{d}{d\varepsilon}\left[T^0(t)+\varepsilon\delta T(t)\right]_{t=0}\right\}_{\varepsilon=0}=\left\{\frac{d}{d\varepsilon}\left(T_0^0+\varepsilon\delta T_0\right)\right\}_{\varepsilon=0}. \quad (68)$$

Performing the operations involving the scalar ε in Eqs. (63)–(68) yields the following expression for the 1st-LVSS:

$$\begin{aligned}&\frac{d}{dt}\delta\varphi(t)+\frac{\alpha_T^0}{l_p^0c_p^0}E^0(t)\delta\varphi(t)+\frac{\alpha_T^0}{l_p^0c_p^0}\varphi^0(t)\delta E(t)\\&=\left[-\frac{\delta\alpha_T}{l_p^0c_p^0}+\frac{\alpha_T^0}{\left(l_p^0\right)^2c_p^0}\delta l_p+\frac{\alpha_T^0}{l_p^0\left(c_p^0\right)^2}\delta c_p\right]E^0(t)\varphi^0(t),\end{aligned} \quad (69)$$

$$\frac{d}{dt}\delta E(t)-\gamma^0\sigma_f^0N_f^0\delta\varphi(t)=\left[\left(\sigma_f^0N_f^0\right)\delta\gamma+\left(\gamma^0N_f^0\right)\delta\sigma_f+\left(\gamma^0\sigma_f^0\right)\delta N_f\right]\varphi^0(t), \quad (70)$$

$$\begin{aligned}&\frac{d}{dt}\delta T(t)-\frac{\gamma^0\sigma_f^0N_f^0}{c_p^0}\delta\varphi(t)\\&=\left[\left(\sigma_f^0N_f^0\right)\delta\gamma+\left(\gamma^0N_f^0\right)\delta\sigma_f+\left(\gamma^0\sigma_f^0\right)\delta N_f-\frac{\gamma^0\sigma_f^0N_f^0}{c_p^0}\delta c_p\right]\frac{\varphi^0(t)}{c_p^0},\end{aligned} \quad (71)$$

$$\left[\delta\varphi(t)\right]_{t=0}=\delta\varphi_0, \quad (72)$$

$$\left[\delta E(t)\right]_{t=0}=0, \quad (73)$$

$$\left[\delta T(t)\right]_{t=0}=\delta T_0. \quad (74)$$

The 1st-LVSS comprising Eqs. (69)–(74) represents the specific form taken on by the general NODE-representation of the 1st-LVSS provided by Eqs. (44) and (45) for the Nordheim-Fuchs model. Comparing Eqs. (69)–(74) to Eqs. (44) and (45) indicates the following correspondences:

$$\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \mathbf{h}} \triangleq \begin{pmatrix} -\frac{\alpha_T^0 E^0(t)}{l_p^0 c_p^0} & -\frac{\alpha_T^0 \varphi^0(t)}{l_p^0 c_p^0} & 0 \\ \gamma^0 \sigma_f^0 N_f^0 & 0 & 0 \\ \frac{\gamma^0 \sigma_f^0 N_f^0}{c_p^0} & 0 & 0 \end{pmatrix}; \ \mathbf{v}^{(1)}(t) \triangleq \begin{pmatrix} \delta\varphi(t) \\ \delta E(t) \\ \delta T(t) \end{pmatrix}; \tag{75}$$

$$\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \boldsymbol{\theta}} \triangleq \begin{pmatrix} \frac{\partial f_1}{\partial \theta_1} & \frac{\partial f_1}{\partial \theta_2} & \frac{\partial f_1}{\partial \theta_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial f_2}{\partial \theta_4} & \frac{\partial f_2}{\partial \theta_5} & \frac{\partial f_2}{\partial \theta_6} \\ 0 & 0 & \frac{\partial f_3}{\partial \theta_3} & \frac{\partial f_3}{\partial \theta_4} & \frac{\partial f_3}{\partial \theta_5} & \frac{\partial f_3}{\partial \theta_6} \end{pmatrix}; \ \delta\boldsymbol{\theta} \triangleq \begin{pmatrix} \delta\alpha \\ \delta l_p \\ \delta c_p \\ \delta\gamma \\ \delta\sigma_f \\ \delta N_f \end{pmatrix}; \tag{76}$$

$$\begin{gathered} \frac{\partial f_1}{\partial \theta_1} \triangleq -\frac{E^0(t)\varphi^0(t)}{l_p^0 c_p^0}; \ \frac{\partial f_1}{\partial \theta_2} \triangleq \frac{\alpha_T^0 E^0(t)\varphi^0(t)}{\left(l_p^0\right)^2 c_p^0}; \ \frac{\partial f_1}{\partial \theta_3} \triangleq \frac{\alpha_T^0 E^0(t)\varphi^0(t)}{l_p^0 \left(c_p^0\right)^2}; \\ \frac{\partial f_2}{\partial \theta_4} \triangleq \sigma_f^0 N_f^0 \varphi^0(t); \ \frac{\partial f_2}{\partial \theta_5} \triangleq \gamma^0 N_f^0 \varphi^0(t); \ \frac{\partial f_2}{\partial \theta_6} \triangleq \gamma^0 \sigma_f^0 \varphi^0(t); \\ \frac{\partial f_3}{\partial \theta_3} \triangleq -\frac{\gamma^0 \sigma_f^0 N_f^0 \varphi^0(t)}{\left(c_p^0\right)^2}; \ \frac{\partial f_3}{\partial \theta_4} \triangleq \frac{\sigma_f^0 N_f^0 \varphi^0(t)}{c_p^0}; \ \frac{\partial f_3}{\partial \theta_5} \triangleq \frac{\gamma^0 N_f^0 \varphi^0(t)}{c_p^0}; \ \frac{\partial f_3}{\partial \theta_6} \triangleq \frac{\gamma^0 \sigma_f^0 \varphi^0(t)}{c_p^0}. \end{gathered} \tag{77}$$

$$\frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial \mathbf{x}} \triangleq \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \ \delta\mathbf{x} \triangleq \begin{pmatrix} \delta\varphi_0 \\ 0 \\ \delta T_0 \end{pmatrix}; \ \frac{\partial \mathbf{h}_e(\mathbf{x},\mathbf{w})}{\partial \mathbf{w}} \triangleq \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}. \tag{78}$$

It is evident that the 1st-LVSS would need to be solved repeatedly in order to compute the 1st-level variational function $\mathbf{v}^{(1)}(t) \triangleq \left[\delta\varphi(t), \delta E(t), \delta T(t)\right]^\dagger$ for every possible variations $\delta\boldsymbol{\theta}$ in the model parameters and variations $\delta\mathbf{x}$ in the initial conditions ('encoder"). This computationally expensive path can be avoided by applying the concepts of the 1st- CASAM-NODE previously outlined in Subsection 4.1, as follows:

1. Consider that the 1st-level variational function $\mathbf{v}^{(1)} \triangleq \left[\delta\varphi(t), \delta E(t), \delta T(t)\right]^\dagger \in \mathsf{H}_1(\Omega_t)$, is an element in a Hilbert space denoted as $\mathsf{H}_1(\Omega_t)$, $\Omega_t \triangleq (0, t_f)$, comprising elements of the form $\mathbf{u}^{(a)}(t) \triangleq \left[u_1^{(a)}(t), u_2^{(a)}(t), u_3^{(a)}(t)\right]^\dagger$, $\mathbf{u}^{(b)}(t) \triangleq \left[u_1^{(b)}(t), u_2^{(b)}(t), u_3^{(b)}(t)\right]^\dagger$, and being endowed with the inner product $\left\langle \mathbf{u}^{(a)}, \mathbf{u}^{(b)} \right\rangle_1$ introduced in Eq. (50), which takes on the following particular form for the Nordheim-Fuchs model:

$$\left\langle \mathbf{u}^{(a)}, \mathbf{u}^{(b)} \right\rangle_1 \triangleq \int_0^{t_f} \mathbf{u}^{(a)}(t) \bullet \mathbf{u}^{(b)}(t)\, dt = \sum_{i=1}^{3} \int_0^{t_f} u_i^{(a)}(t) u_i^{(b)}(t)\, dt . \tag{79}$$

2. Use Eq. (79) to form the inner product of Eqs. (69)–(71) with a yet undefined function $\mathbf{a}^{(1)}(t) \triangleq \left[a_1^{(1)}(t), a_2^{(1)}(t), a_3^{(1)}(t)\right]^\dagger \in \mathsf{H}_1(\Omega_t)$, to obtain the following relation, which is the particular form taken on by Eq. (51) for the Nordheim-Fuchs model:

$$\begin{aligned}&\int_0^{t_f} a_1^{(1)}(t)\left[\frac{d}{dt}\delta\varphi(t)+\frac{\alpha_T^0}{l_p^0 c_p^0}E^0(t)\delta\varphi(t)+\frac{\alpha_T^0}{l_p^0 c_p^0}\varphi^0(t)\delta E(t)\right]dt\\&+\int_0^{t_f} a_2^{(1)}(t)\left[\frac{d}{dt}\delta E(t)-\gamma^0\sigma_f^0 N_f^0\delta\varphi(t)\right]dt\\&+\int_0^{t_f} a_3^{(1)}(t)\left[\frac{d}{dt}\delta T(t)-\frac{\gamma^0\sigma_f^0 N_f^0}{c_p^0}\delta\varphi(t)\right]dt\\&=\int_0^{t_f} a_1^{(1)}(t)\left[-\frac{\delta\alpha_T}{l_p^0 c_p^0}+\frac{\alpha_T^0}{\left(l_p^0\right)^2 c_p^0}\delta l_p+\frac{\alpha_T^0}{l_p^0\left(c_p^0\right)^2}\delta c_p\right]E^0(t)\varphi^0(t)\,dt\\&+\int_0^{t_f} a_2^{(1)}(t)\left[\left(\sigma_f^0 N_f^0\right)\delta\gamma+\left(\gamma^0 N_f^0\right)\delta\sigma_f+\left(\gamma^0\sigma_f^0\right)\delta N_f\right]\varphi^0(t)dt\\&+\int_0^{t_f} a_3^{(1)}(t)\left[\left(\sigma_f^0 N_f^0\right)\delta\gamma+\left(\gamma^0 N_f^0\right)\delta\sigma_f+\left(\gamma^0\sigma_f^0\right)\delta N_f-\frac{\gamma^0\sigma_f^0 N_f^0}{c_p^0}\delta c_p\right]\frac{\varphi^0(t)}{c_p^0}dt.\end{aligned} \tag{80}$$

3. Integrating by parts the terms on the left-side of Eq. (80) yields the following relation

$$\begin{aligned}&\int_0^{t_f} a_1^{(1)}(t)\left[\frac{d}{dt}\delta\varphi(t)+\frac{\alpha_T^0}{l_p^0 c_p^0}E^0(t)\delta\varphi(t)+\frac{\alpha_T^0}{l_p^0 c_p^0}\varphi^0(t)\delta E(t)\right]dt\\&+\int_0^{t_f} a_2^{(1)}(t)\left[\frac{d}{dt}\delta E(t)-\gamma^0\sigma_f^0 N_f^0\delta\varphi(t)\right]dt\\&+\int_0^{t_f} a_3^{(1)}(t)\left[\frac{d}{dt}\delta T(t)-\frac{\gamma^0\sigma_f^0 N_f^0}{c_p^0}\delta\varphi(t)\right]dt=a_1^{(1)}(t_f)\delta\varphi(t_f)-a_1^{(1)}(0)\delta\varphi(0)\\&+a_2^{(1)}(t_f)\delta E(t_f)-a_2^{(1)}(0)\delta E(0)+a_3^{(1)}(t_f)\delta T(t_f)-a_3^{(1)}(0)\delta T(0)\\&+\int_0^{t_f}\mathbf{v}^{(1)}(t)\bullet\left\{\mathbf{A}^{(1)}(\mathbf{h};\boldsymbol{\theta})\mathbf{a}^{(1)}(t)\right\}_{\left(\mathbf{h}^0;\boldsymbol{\theta}^0\right)}dt,\end{aligned} \tag{81}$$

where:

$$\mathbf{A}^{(1)}(\mathbf{h};\boldsymbol{\theta})\mathbf{a}^{(1)}(t)\triangleq-\frac{d\mathbf{a}^{(1)}(t)}{dt}-\left[\frac{\partial\mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial\mathbf{h}}\right]^{\dagger}_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)}\mathbf{a}^{(1)}(t); \tag{82}$$

with

$$\left[\frac{\partial\mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial\mathbf{h}}\right]^{\dagger}_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)}\triangleq\begin{pmatrix}-\alpha_T^0E^0(t)/\left(l_p^0c_p^0\right) & \gamma^0\sigma_f^0N_f^0 & \gamma^0\sigma_f^0N_f^0/c_p^0\\-\alpha_T^0\varphi^0(t)/\left(l_p^0c_p^0\right) & 0 & 0\\0 & 0 & 0\end{pmatrix}. \tag{83}$$

The relation obtained in Eq. (81) is the particular form taken on by Eq. (52) for the Nordheim-Fuchs model.

4. The definition of the function $\mathbf{a}^{(1)}(t)$ is now completed by requiring that: (i) the integral term on the right-side Eq.(81) represent the G-differential $\delta r_1(\mathbf{h};\delta\mathbf{h})$ defined in Eq. (62), and (ii) the appearance of the unknown values of the components of $\mathbf{v}^{(1)}(t_f)$ be eliminated from appearing in Eq. (81). These requirements will be satisfied if the function $\mathbf{a}^{(1)}(t)\triangleq\left[a_1^{(1)}(t),a_2^{(1)}(t),a_3^{(1)}(t)\right]^{\dagger}\in\mathsf{H}_1(\Omega_t)$ is the solution of the following "1st-Level Adjoint Sensitivity System (1st-LASS)":

$$\mathbf{A}^{(1)}(\mathbf{h};\boldsymbol{\theta})\mathbf{a}^{(1)}(t)\triangleq-\frac{d\mathbf{a}^{(1)}(t)}{dt}-\left[\frac{\partial\mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial\mathbf{h}}\right]^{\dagger}_{\left(\mathbf{h}^0,\boldsymbol{\theta}^0\right)}\mathbf{a}^{(1)}(t)=\left[\delta\left(t-t_f\right),0,0\right]^{\dagger}; \tag{84}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

$$\mathbf{a}^{(1)}\left(t_f\right) \triangleq \left[a_1^{(1)}\left(t_f\right), a_2^{(1)}\left(t_f\right), a_3^{(1)}\left(t_f\right)\right]^{\dagger} = [0,0,0]^{\dagger}. \tag{85}$$

It is important to note that if the vector-valued function $\mathbf{f}(\mathbf{h};\boldsymbol{\theta})$ is linear in $\mathbf{h}(t)$ (in which case the NODE would be linear), then the 1st-level adjoint sensitivity function $\mathbf{a}^{(1)}(t)$ would not depend on $\mathbf{h}(t)$, so the "forward solution path" would not need to be stored in order to compute $\mathbf{a}^{(1)}(t)$. Otherwise, however, the "forward solution path" $\mathbf{h}(t)$ would need to be stored in order to compute $\mathbf{a}^{(1)}(t)$.

5. Using Eqs. (84), (85), (80), (62), (72), (73) and (74) in Eq. (81) yields the following expression for the first G-differential $\delta r_1(\mathbf{h};\delta\mathbf{h})$ of the response under consideration:

$$\begin{aligned}
\delta r_1(\mathbf{h};\delta\mathbf{h}) = \delta\varphi\left(t_f\right) = \int_0^{t_f} a_1^{(1)}(t)\left[-\frac{\delta\alpha_T}{l_p^0 c_p^0} + \frac{\alpha_T^0}{\left(l_p^0\right)^2 c_p^0}\delta l_p + \frac{\alpha_T^0}{l_p^0\left(c_p^0\right)^2}\delta c_p\right] E^0(t)\varphi^0(t)\,dt \\
+\int_0^{t_f} a_2^{(1)}(t)\left[\left(\sigma_f^0 N_f^0\right)\delta\gamma + \left(\gamma^0 N_f^0\right)\delta\sigma_f + \left(\gamma^0\sigma_f^0\right)\delta N_f\right]\varphi^0(t)\,dt \\
+\int_0^{t_f} a_3^{(1)}(t)\left[\left(\sigma_f^0 N_f^0\right)\delta\gamma + \left(\gamma^0 N_f^0\right)\delta\sigma_f + \left(\gamma^0\sigma_f^0\right)\delta N_f - \frac{\gamma^0\sigma_f^0 N_f^0}{c_p^0}\delta c_p\right]\frac{\varphi^0(t)}{c_p^0}\,dt \\
+a_1^{(1)}(0)\,\delta\varphi_0 + a_3^{(1)}(0)\,\delta T_0 .
\end{aligned} \tag{86}$$

It follows from Eq. (86) that the first-order sensitivities of the response $\varphi\left(t_f\right)$ with respect to the parameters and initial conditions underlying the Nordheim-Fuchs model have the following expressions, all of which are to be evaluated at the nominal values of the respective parameters and functions (but the superscript "zero" is omitted to simplify the notation):

$$\frac{\partial\varphi\left(t_f\right)}{\partial\alpha_T} = -\frac{1}{l_p c_p}\int_0^{t_f} a_1^{(1)}(t)E(t)\varphi(t)\,dt; \tag{87}$$

$$\frac{\partial\varphi\left(t_f\right)}{\partial l_p} = \frac{\alpha_T}{\left(l_p\right)^2 c_p}\int_0^{t_f} a_1^{(1)}(t)E(t)\varphi(t)\,dt; \tag{88}$$

$$\frac{\partial\varphi\left(t_f\right)}{\partial c_p} = \frac{\alpha_T}{l_p\left(c_p\right)^2}\int_0^{t_f} a_1^{(1)}(t)E(t)\varphi(t)\,dt - \frac{\gamma\sigma_f N_f}{\left(c_p\right)^2}\int_0^{t_f} a_3^{(1)}(t)\varphi(t)\,dt; \tag{89}$$

$$\frac{\partial\varphi\left(t_f\right)}{\partial\gamma} = \sigma_f N_f\int_0^{t_f}\left[a_2^{(1)}(t) + \frac{1}{c_p}a_3^{(1)}(t)\right]\varphi(t)\,dt; \tag{90}$$

$$\frac{\partial\varphi\left(t_f\right)}{\partial\sigma_f} = \gamma N_f\int_0^{t_f}\left[a_2^{(1)}(t) + \frac{1}{c_p}a_3^{(1)}(t)\right]\varphi(t)\,dt; \tag{91}$$

$$\frac{\partial\varphi\left(t_f\right)}{\partial N_f} = \gamma\sigma_f\int_0^{t_f}\left[a_2^{(1)}(t) + \frac{1}{c_p}a_3^{(1)}(t)\right]\varphi(t)\,dt; \tag{92}$$

$$\frac{\partial\varphi\left(t_f\right)}{\partial\varphi_0} = a_1^{(1)}(0);\quad \frac{\partial\varphi\left(t_f\right)}{\partial E(0)} = 0;\quad \frac{\partial\varphi\left(t_f\right)}{\partial T_0} = a_3^{(1)}(0). \tag{93}$$

5.2. First-Order Sensitivities of the Energy Released Response $r_2(\mathbf{h}) = E(t_f)$

The first-order G-differential of the response $r_2(\mathbf{h}) = E(t_f)$ defined in Eq. (32) is obtained as follows:

$$\delta r_2(\mathbf{h};\delta\mathbf{h}) = \left\{ \frac{d}{d\varepsilon} \left[\int_0^{t_f} \left[E^0(t) + \varepsilon \delta E(t) \right] \delta(t - t_f) dt \right] \right\}_{\varepsilon=0} = \int_0^{t_f} \delta E(t) \delta(t - t_f) dt \, , \quad (94)$$

where the variation $\delta E(t)$ is the solution of the "First-Level Variational Sensitivity System (1st-LVSS) defined by Eqs. (69)–(74).

The sensitivities of the response $r_2(\mathbf{h}) = E(t_f)$ are determined by following the same procedure as has been outlined in Section 5.2, using an adjoint function denoted as $\boldsymbol{\chi}^{(1)}(t) \triangleq \left[\chi_1^{(1)}(t), \chi_2^{(1)}(t), \chi_3^{(1)}(t) \right]^\dagger \in \mathsf{H}_1(\Omega_t)$. Following the same steps as in Section 5.2 (which are omitted here to avoid undue repetition) leads to the following 1st-LASS for the 1st-level adjoint sensitivity function $\boldsymbol{\chi}^{(1)}(t)$:

$$\mathbf{A}^{(1)}(\mathbf{h};\boldsymbol{\theta}) \boldsymbol{\chi}^{(1)}(t) \triangleq -\frac{d\boldsymbol{\chi}^{(1)}(t)}{dt} - \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \mathbf{h}} \right]^\dagger_{(\mathbf{h}^0,\boldsymbol{\theta}^0)} \boldsymbol{\chi}^{(1)}(t) = \left[0, \delta(t - t_f), 0 \right]^\dagger ; \quad (95)$$

$$\boldsymbol{\chi}^{(1)}(t_f) \triangleq \left[\chi_1^{(1)}(t_f), \chi_2^{(1)}(t_f), \chi_3^{(1)}(t_f) \right]^\dagger = [0,0,0]^\dagger . \quad (96)$$

The sensitivities of $E(t_f)$ with respect to the model parameters and initial conditions have the same formal expressions as shown in Eqs. (87)–(93), but with the components of the 1st-level adjoint sensitivity function $\boldsymbol{\chi}^{(1)}(t)$ replacing the components of $\mathbf{a}^{(1)}(t) \triangleq \left[a_1^{(1)}(t), a_2^{(1)}(t), a_3^{(1)}(t) \right]^\dagger$.

5.3. First-Order Sensitivities of the Temperature Response $r_3(\mathbf{h}) = T(t_f)$

The first-order G-differential of the response $r_3(\mathbf{h}) = T(t_f)$ defined in Eq. (33) is obtained as follows:

$$\delta r_3(\mathbf{h};\delta\mathbf{h}) = \left\{ \frac{d}{d\varepsilon} \left[\int_0^{t_f} \left[T^0(t) + \varepsilon \delta T(t) \right] \delta(t - t_f) dt \right] \right\}_{\varepsilon=0} = \int_0^{t_f} \delta T(t) \delta(t - t_f) dt \, , \quad (97)$$

where the variation $\delta T(t)$ is the solution of the "First-Level Variational Sensitivity System (1st-LVSS) defined by Eqs. (69)–(74).

The sensitivities of the response $r_3(\mathbf{h}) = T(t_f)$ are determined by following the same procedure as has been outlined in Section 5.1, using an adjoint function denoted as $\boldsymbol{\xi}^{(1)}(t) \triangleq \left[\xi_1^{(1)}(t), \xi_2^{(1)}(t), \xi_3^{(1)}(t) \right]^\dagger \in \mathsf{H}_1(\Omega_t)$. Following the same steps as in Section 5.1 (which are omitted here to avoid undue repetition) leads to the following 1st-LASS for the 1st-level adjoint sensitivity function $\boldsymbol{\xi}^{(1)}(t)$:

$$\mathbf{A}^{(1)}(\mathbf{h};\boldsymbol{\theta}) \boldsymbol{\xi}^{(1)}(t) \triangleq -\frac{d\boldsymbol{\xi}^{(1)}(t)}{dt} - \left[\frac{\partial \mathbf{f}(\mathbf{h};\boldsymbol{\theta})}{\partial \mathbf{h}} \right]^\dagger_{(\mathbf{h}^0,\boldsymbol{\theta}^0)} \boldsymbol{\xi}^{(1)}(t) = \left[0, 0, \delta(t - t_f) \right]^\dagger ; \quad (98)$$

$$\boldsymbol{\xi}^{(1)}(t_f) \triangleq \left[\xi_1^{(1)}(t_f), \xi_2^{(1)}(t_f), \xi_3^{(1)}(t_f) \right]^\dagger = [0,0,0]^\dagger . \quad (99)$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

The sensitivities of $T\left(t_f\right)$ with respect to the model parameters and initial conditions have the same formal expressions as shown in Eqs. (87)–(93), but with the components of the 1st-level adjoint sensitivity function $\boldsymbol{\xi}^{(1)}(t) \triangleq \left[\xi_1^{(1)}(t), \xi_2^{(1)}(t), \xi_3^{(1)}(t)\right]^\dagger$ replacing the components of $\mathbf{a}^{(1)}(t) \triangleq \left[a_1^{(1)}(t), a_2^{(1)}(t), a_3^{(1)}(t)\right]^\dagger$.

5.4. First-Order Sensitivities of the Thermal Conductivity Response $r_4(\mathbf{h};\boldsymbol{\varphi}) = k(T_f;\boldsymbol{\varphi})$

The first-order G-differential of the response $r_4(\mathbf{h};\boldsymbol{\varphi}) = k(T_f;\boldsymbol{\varphi})$ defined in Eq. (34) is obtained as follows:

$$\begin{aligned}
&\delta r_4(\mathbf{h};\boldsymbol{\varphi};\delta\mathbf{h};\delta\boldsymbol{\varphi}) = \delta k(T;\boldsymbol{\varphi};\delta T;\delta\boldsymbol{\varphi}) = \left\{\frac{d}{d\varepsilon}\int_0^{t_f}\left[\varphi_1^0 + \varepsilon\delta\varphi_1\right]\delta\left(t-t_f\right)dt\right\}_{\varepsilon=0} \\
&+\left\{\frac{d}{d\varepsilon}\int_0^{t_f}\left[\left(\varphi_2^0+\varepsilon\delta\varphi_2\right)\left(T^0+\varepsilon\delta T\right)+\left(\varphi_3^0+\varepsilon\delta\varphi_3\right)\left(T^0+\varepsilon\delta T\right)^2\right]\delta\left(t-t_f\right)dt\right\}_{\varepsilon=0} \\
&=\left\{\delta k(T;\boldsymbol{\varphi};\delta\boldsymbol{\varphi})\right\}_{dir} + \left\{\delta k(T;\boldsymbol{\varphi};\delta T)\right\}_{ind},
\end{aligned} \tag{100}$$

where the direct-effect and the indirect-effect terms, respectively, are defined as follows:

$$\left\{\delta k(T;\boldsymbol{\varphi};\delta\boldsymbol{\varphi})\right\}_{dir} \triangleq \delta\varphi_1 + \delta\varphi_2\int_0^{t_f} T^0(t)\,\delta\left(t-t_f\right) + \delta\varphi_3\int_0^{t_f}\left[T^0(t)\right]^2\delta\left(t-t_f\right); \tag{101}$$

$$\left\{\delta k(T;\boldsymbol{\varphi};\delta T)\right\}_{ind} \triangleq \int_0^{t_f}\left[\varphi_2^0 + 2\varphi_3^0 T^0(t)\right]\delta T(t)\,\delta\left(t-t_f\right)dt. \tag{102}$$

The direct-effect term yields the following sensitivities which can be evaluated immediately:

$$\frac{\partial k(T_f)}{\partial\varphi_1} = 1;\quad \frac{\partial k(T_f)}{\partial\varphi_2} = T^0\left(t_f\right);\quad \frac{\partial k(T_f)}{\partial\varphi_3} = \left[T^0\left(t_f\right)\right]^2. \tag{103}$$

The indirect-effect term can be evaluated only after determining the variational function $\delta T(t)$, which is the solution of the 1st-LVSS defined by Eqs. (69)–(74). The need for solving (repeatedly) the 1st-LVSS can be circumvented by applying the principles of the 1st-CASAM-NODE, as previously outlined. Thus, following the same procedure as detailed in Section 5.1 leads to the following 1st-LASS for the 1st-level adjoint sensitivity function, denoted as $\boldsymbol{\psi}^{(1)}(t) \triangleq \left[\psi_1^{(1)}(t), \psi_2^{(1)}(t), \psi_3^{(1)}(t)\right]^\dagger \in \mathsf{H}_1(\Omega_t)$, for computing the sensitivities stemming from the indirect-effect term $\left\{\delta k(T;\boldsymbol{\varphi};\delta T)\right\}_{ind}$:

$$\mathbf{A}^{(1)}(\mathbf{h};\boldsymbol{\theta})\,\boldsymbol{\psi}^{(1)}(t) = \left[\left[\varphi_2^0 + 2\varphi_3^0 T^0(t)\right]\delta\left(t-t_f\right), 0, 0\right]^\dagger; \tag{104}$$

$$\boldsymbol{\psi}^{(1)}\left(t_f\right) \triangleq \left[\psi_1^{(1)}\left(t_f\right), \psi_2^{(1)}\left(t_f\right), \psi_3^{(1)}\left(t_f\right)\right]^\dagger = [0,0,0]^\dagger. \tag{105}$$

It is important to note that all of the following 1st-Level Adjoint Sensitivity Systems, enumerated in items (i) through (iv), below:

(i) the 1st-LASS defined by Eqs. (84) and (85), which are solved for obtaining the corresponding 1st-level adjoint sensitivity function needed for computing the sensitivities of the component $h_1(t) \triangleq \varphi(t)$ of the state function $\mathbf{h}(t)$;

(ii) the 1st-LASS defined by Eqs. (95) and (96), which are solved for obtaining the corresponding 1st-level adjoint sensitivity function needed for computing the sensitivities of the component $h_2(t) \triangleq E(t)$ of the state function $\mathbf{h}(t)$];

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

(iii) the 1st-LASS defined by Eqs. (98) and (99), which are solved for obtaining the corresponding 1st-level adjoint sensitivity function needed for computing the sensitivities of the component $h_3(t) \triangleq T(t)$ of the state function $\mathbf{h}(t)$, and

(iv) the 1st-LASS defined by Eqs. (104) and (105), which are solved for obtaining the corresponding 1st-level adjoint sensitivity function needed for computing the sensitivities stemming from the indirect-effect term $\{\delta k(T;\boldsymbol{\varphi};\delta T)\}_{ind}$;

…have the same structures/operators on their left sides, and the respective adjoint sensitivity function all satisfy the same final-time conditions; only the source terms on the right-sides of the respective 1st-LASS differ from each other. Consequently, the same numerical procedures and/or neural nets can be used for computing the respective 1st-level adjoint sensitivity functions.

Since the NODE is a first-order ODE, the corresponding 1st-LASS is solved "backwards" in time, starting at the final time-step $t = t_f$, as indicated by the general 1st-CASAM-NODE methodology presented in Section 4. If the NODE is linear in the state function (dependent variable) $\mathbf{h}(t)$, then the 1st-LASS will be independent of $\mathbf{h}(t)$, so the "forward solution path" would not need to be stored in order to compute the 1st-level adjoint sensitivity functions. In contradistinction, if the NODE is nonlinear in the state function (dependent variable) $\mathbf{h}(t)$, then the 1st-LASS will depend on $\mathbf{h}(t)$, so the "forward solution path" would need to be stored in order to compute the respective 1st-level adjoint sensitivity functions.

Furthermore, the same formal expressions are obtained for the sensitivities of the responses considered. Thus, the respective 1st-level adjoint sensitivity functions differ from each other according to the response considered, but the quadrature-schemes needed to evaluate the integrals defining the respective sensitivities are the same. Therefore, the same numerical procedures and/or neural nets can be used for computing the respective integrals that define the 1st-order sensitivities, while using the appropriate/corresponding 1st-level adjoint sensitivity functions. If the decoder-response depends on parameters/weights, additional sensitivities arise from the respective non-vanishing "direct-effect term."

If simple relations can be obtained among the responses of interest, such as Eqs. (11) and (16) for the illustrative paradigm example, then the sensitivities of the various responses can be obtained by using these relationships, but this is seldom the case in practice.

5.5. Most Efficient Computation of First-Order Sensitivities: Application of the 1st-FASAM-N

In most, if not all, practical situations, the equations modeling the physical system under consideration can be recast to suit the computation of the response under consideration and, consequently, the computation of the response sensitivities with respect to the underlying model parameters. For example, the response $r_2(\mathbf{h}) = E(t_f)$ involves just the function $E(t)$; hence, this response would be ideally computed, together with its sensitivities to parameters, by using an equation containing as few as possible dependent variables other than the ones [e.g., $E(t)$] needed for computing the response. Such an equation was obtained in Eq. (17), which contains just the dependent variable $E(t)$, so it would be more advantageous to us it for the sensitivity analysis of $r_2(\mathbf{h}) = E(t_f)$ rather than use the entire system of equations underlying the Nordheim-Fuchs model, as was done, for illustrative purposes, in Section 5.1. Furthermore, the form of Eq. (17) indicates that the "features" (i.e., functions) of model parameters characterizing this balance equation can be chosen as follows:

$$F_1(\mathbf{p}) \triangleq \frac{\alpha_T}{2l_p c_p}; \; F_2(\mathbf{p}) \triangleq \varphi_0 \gamma \sigma_f N_f; \; \mathbf{F}(\mathbf{p}) \triangleq \left[F_1(\mathbf{p}), F_2(\mathbf{p})\right]^\dagger, \tag{106}$$

where the vector of *primary* model parameters is defined as follows:

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

$$\mathbf{p} \triangleq \left[p_1, \ldots, p_7\right]^{\dagger} \triangleq \left[\alpha_T, l_p, c_p, \gamma, \sigma_f, N_f, \varphi_0\right]^{\dagger}. \tag{107}$$

Note that the vector $\mathbf{p}$ includes the initial condition φ_0.

In terms of the "feature function" $\mathbf{F}(\mathbf{p}) \triangleq \left[F_1(\mathbf{p}), F_2(\mathbf{p})\right]^{\dagger}$, Eq. (17) can alternatively be written as follows:

$$\frac{dE(t)}{dt} = -F_1(\mathbf{p}) E^2(t) + F_2(\mathbf{p}), \;\; E(0) = 0. \tag{108}$$

In terms of the feature function $\mathbf{F}(\mathbf{p}) \triangleq \left[F_1(\mathbf{p}), F_2(\mathbf{p})\right]^{\dagger}$, the solution of Eq. (108) has the following form:

$$E(t) = \left[\frac{F_2(\mathbf{p})}{F_1(\mathbf{p})}\right]^{1/2} \tanh\left[tG(\mathbf{p})\right]; \;\; G(\mathbf{p}) \triangleq \sqrt{F_1(\mathbf{p}) F_2(\mathbf{p})}. \tag{109}$$

Of course, a specific NODE would need to be constructed to model Eq. (108).

The form of Eq. (108) is suitable for applying the "nth-Order Features Adjoint Sensitivity Analysis Methodology for Nonlinear Systems (nth-FASAM-N)" [24], which is the most efficient methodology for computing sensitivities, particularly for sensitivities of second- and higher-order. This methodology considers the specific "features" of model parameters, such as the function $\mathbf{F}(\mathbf{p}) \triangleq \left[F_1(\mathbf{p}), F_2(\mathbf{p})\right]^{\dagger}$, to compute sensitivities with respect to model parameters more efficiently than by considering directly the respective primary parameters.

For the computation of 1st-order sensitivities, the 1st-FASAM-N commences by constructing the 1st-Level Variational Sensitivity System (1st-LVSS) for the variational function $\delta E(t)$ by applying the definition of the first-order G-differential to Eq. (108), which yields:

$$\frac{d}{d\varepsilon}\left\{\frac{d\left[E^0(t) + \varepsilon \delta E(t)\right]}{dt} + \left[F_1^0 + \varepsilon \delta F_1\right]\left[E^0 + \varepsilon \delta E\right]^2 - \left[F_2^0 + \varepsilon\left(\delta F_2\right)\right]\right\}_{\varepsilon=0} = 0, \tag{110}$$

$$\frac{d}{d\varepsilon}\left\{\left[E^0(t) + \varepsilon \delta E(t)\right]_{t=0}\right\}_{\varepsilon=0} = 0. \tag{111}$$

Performing the operations indicated in Eqs. (110) and (111) yields the following expression for the 1st-LVSS satisfied by the variational function $\delta E(t)$:

$$\left[\frac{d}{dt} + 2F_1 E(t)\right]\delta E(t) = -\delta F_1 E^2(t) + \delta F_2, \;\; t > 0, \tag{112}$$

$$\delta E(0) = 0, \;\; t = 0. \tag{113}$$

The 1st-LVSS represented by Eq. (112) is to be solved at the nominal values for the parameters and the state function $E(t)$ but the superscript "0" (which indicates "nominal values") has been omitted to simplify the notation.

Numerically, the 1st-LVSS would need to be solved anew for the various variations δF_1, δF_2, in the components of the feature function $\mathbf{F}(\mathbf{p})$. This need for repeatedly solving the 1st-LVSS can be avoided by constructing the corresponding 1st-Level Adjoint Sensitivity System (1st-LASS). The Hilbert space appropriate for the construction of the 1st-LASS corresponding to Eq. (112) is endowed with the following particular form of Eq. (79):

$$\left\langle u^{(a)}(t), u^{(b)}(t)\right\rangle_1 \triangleq \int_0^{t_f} u^{(a)}(t) u^{(b)}(t)\, dt. \tag{114}$$

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

Using Eq. (114) to form the inner product of Eq. (112) with a yet undefined function $\omega^{(1)}(t)$ yields the following relation:

$$\int_0^{t_f} \omega^{(1)}(t)\left[\frac{d}{dt}+2F_1E(t)\right]\delta E(t)\,dt = -(\delta F_1)\int_0^{t_f}\omega^{(1)}(t)E^2(t)\,dt + (\delta F_2)\int_0^{t_f}\omega^{(1)}(t)\,dt\,. \quad (115)$$

Integrating by parts the left side of Eq. (115) yields the following relation:

$$\int_0^{t_f} \omega^{(1)}(t)\left[\frac{d}{dt}+2F_1E(t)\right]\delta E(t)\,dt = \omega^{(1)}(\tau)\delta E(\tau) - \omega^{(1)}(0)\delta E(0)$$
$$+\int_0^{t_f}\delta E(t)\left[-\frac{d\omega^{(1)}(t)}{dt}+2F_1E(t)\omega^{(1)}(t)\right]dt. \quad (116)$$

Identifying the integral on the right-side of Eq. (116) with the G-differential $\delta E(\tau)$ of the response $E(\tau)$ obtained in Eq. (32) and eliminating the unknown value $\delta E(\tau)$ from the right-side of Eq. (116) by setting $\omega^{(1)}(\tau)=0$ yields the following 1st-Level Adjoint Sensitivity System (1st-LASS) for the 1st-level adjoint sensitivity function $\omega^{(1)}(t)$:

$$\left[-\frac{d}{dt}+2F_1E(t)\right]\omega^{(1)}(t) = \delta(t-t_f),\ \ t>0\,, \quad (117)$$

$$\omega^{(1)}(t_f)=0, \quad t=t_f\,. \quad (118)$$

The 1st-LASS represented by Eqs. (117) and (118) is independent of variations in the feature functions (and/or parameters) so it would need to be solved only once, numerically. In the present case, the 1st-LASS can be solved analytically to obtain the following closed-form expression for the 1st-level adjoint sensitivity function $\omega^{(1)}(t)$:

$$\omega^{(1)}(t) = H(t_f-t)\left\{\frac{\cosh[tG(\mathbf{p})]}{\cosh[t_fG(\mathbf{p})]}\right\}^2, \quad (119)$$

where $H(t-t_f)$ denotes the Heaviside functional.

Using Eqs. (116)–(118) in Eq. (115) yields the following expression for the first-order total G-differential $\delta E(t_f)$ of the response $E(t_f)$ in terms of the 1st-level adjoint function $\omega^{(1)}(t)$:

$$\delta E(t_f) = -(\delta F_1)\int_0^{t_f}\omega^{(1)}(t)E^2(t)\,dt + (\delta F_2)\int_0^{t_f}\omega^{(1)}(t)\,dt\,. \quad (120)$$

It follows from Eqs. (120), (119) and (109) that the two sensitivities of the response $E(t_f)$ with respect to the two components of the feature function $\mathbf{F}\triangleq(F_1,F_2)^\dagger$ have the following expressions:

$$\frac{\partial E(t_f)}{\partial F_1} = -\int_0^{t_f}\omega^{(1)}(t)E^2(t)\,dt = \frac{1}{2}\left[\frac{F_2(\mathbf{p})}{F_1(\mathbf{p})}\right]^{1/2}\left\{\frac{t_f}{\cosh^2[t_fG(\mathbf{p})]} - \frac{\tanh[t_fG(\mathbf{p})]}{G(\mathbf{p})}\right\}; \quad (121)$$

$$\frac{\partial E(t_f)}{\partial F_2} = \int_0^{t_f}\omega^{(1)}(t)\,dt = \frac{1}{2G(\mathbf{p})}\tanh[t_fG(\mathbf{p})] + \frac{t_f}{2\cosh^2[t_fG(\mathbf{p})]}\,. \quad (122)$$

The above expressions are to be evaluated at the nominal parameter values but the superscript "zero" has been omitted, for simplicity. The expressions obtained in Eqs. (121) and (122) can be verified by differentiating the expression provided in Eq. (109), evaluated at a user-chosen time $t=t_f$ within the interval $0<t_f<\infty$.

Preprints.org (www.preprints.org) | NOT PEER-REVIEWED | Posted: 1 August 2024 doi:10.20944/preprints202407.2613.v1

The sensitivities of the response $E(t_f)$ with respect to the model parameters and initial condition are obtained by using the following "chain-rule" relationship:

$$\frac{\partial E(t_f;F_1;F_2)}{\partial p_i} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_1(\mathbf{p})}{\partial p_i} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2(\mathbf{p})}{\partial p_i};\ i=1,...,7. \tag{123}$$

The explicit expressions for the specific sensitivities of the response $E(t_f)$ with respect to the parameters underlying the feature functions are obtained using Eq. (123) in conjunction with Eqs. (121) and (122) while recalling the definitions of the feature functions $F_1(\mathbf{p})$ and $F_2(\mathbf{p})$ defined in Eq. (106). The detailed expressions of these sensitivities are as follows:

$$\frac{\partial E(t_f)}{\partial \alpha_T} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_1}{\partial \alpha_T} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2}{\partial \alpha_T} = \frac{1}{2l_p c_p}\frac{\partial E(t_f)}{\partial F_1}; \tag{124}$$

$$\frac{\partial E(t_f)}{\partial l_p} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_1}{\partial l_p} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2}{\partial l_p} = -\frac{\alpha_T}{2(l_p)^2 c_p}\frac{\partial E(t_f)}{\partial F_1}; \tag{125}$$

$$\frac{\partial E(t_f)}{\partial c_p} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_1}{\partial c_p} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2}{\partial c_p} = -\frac{\alpha_T}{2(c_p)^2 l_p}\frac{\partial E(t_f)}{\partial F_1}; \tag{126}$$

$$\frac{\partial E(t_f)}{\partial \gamma} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_1}{\partial \gamma} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2}{\partial \gamma} = \varphi_0 \sigma_f N_f \frac{\partial E(t_f)}{\partial F_2}; \tag{127}$$

$$\frac{\partial E(t_f)}{\partial \sigma_f} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_1}{\partial \sigma_f} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2}{\partial \sigma_f} = \varphi_0 \gamma N_f \frac{\partial E(t_f)}{\partial F_2}; \tag{128}$$

$$\frac{\partial E(t_f)}{\partial N_f} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_1}{\partial N_f} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2}{\partial N_f} = \varphi_0 \gamma \sigma_f \frac{\partial E(t_f)}{\partial F_2}. \tag{129}$$

$$\frac{\partial E(t_f)}{\partial \varphi_0} = \frac{\partial E(t_f)}{\partial F_1}\frac{\partial F_2}{\partial \varphi_0} + \frac{\partial E(t_f)}{\partial F_2}\frac{\partial F_2}{\partial \varphi_0} = \gamma \sigma_f N_f \frac{\partial E(t_f)}{\partial F_2}; \tag{130}$$

Notably, the application of the 1st-FASAM-N requires one "large-scale" computation to solve the 1st-LASS, cf. Eq. (117) and (118), which is a single ODE, to obtain the 1st-level adjoint function $\omega^{(1)}(t)$, which is a scalar-valued function. However, solving the forward model, cf. Eq. (17), and the corresponding 1st-LASS, comprising Eq. (117) and (118), would require the construction of a separate (albeit simpler) NODE. The 1st-level adjoint function $\omega^{(1)}(t)$ is subsequently used in performing two integrals (quadrature) for obtaining the two sensitivities of the response $E(t_f)$ with respect to the two components $F_1(\mathbf{p})$ and $F_2(\mathbf{p})$ of the feature function $\mathbf{F}(\mathbf{p}) \triangleq (F_1, F_2)^\dagger$. Subsequently, all of the response sensitivities with respect to the model's primary parameters are obtained analytically by using the chain-rule to differentiate the components of the feature function with respect to the underlying model parameters and initial conditions.

In contradistinction, if one wishes to compute directly the sensitivities of the response with respect to the model parameters and initial conditions, it has been shown in Subsections 5.1–5.4 that the original NODE can be used to solve (backward in time) the 1st-LASS, which comprises a system of three coupled ODEs (rather than a single ODE if the 1st-FASAM is used) for obtaining the 1st-level adjoint function, which is a vector-valued function comprising three components, cf.

 doi:10.20944/preprints202407.2613.v1

$\boldsymbol{\chi}^{(1)}(t) \triangleq \left[\chi_1^{(1)}(t), \chi_2^{(1)}(t), \chi_3^{(1)}(t)\right]^{\dagger}$ for the response $E(t_f)$. The respective vector-valued 1st-level adjoint function is subsequently used in computing six (rather than two, if the 1st-FASAM is used) integrals (quadrature) for obtaining the six sensitivities of the respective response with respect to the six model parameters.

Equations similar to Eq. (17) can be derived for the reactor-flux and reactor temperature responses, so the 1st-FASAM can be applied in a similar fashion to compute the first-order sensitivities of these responses. Using the sensitivities with respect to the reactor temperature response would readily provide the first-order sensitivities of the reactor thermal conductivity response. However, corresponding to each of these responses, a specific NODE would need to be constructed. Of course, any of these specific NODE would have much simpler structures than the NODE for solving simultaneously the system of coupled ODEs presented in Subsections 5.1 through 5.4.

6. Discussion and Conclusions

This work has introduced the mathematical framework of the novel "First-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations (1st-CASAM-NODE)" which yields exact expressions for the first-order sensitivities of NODE decoder-responses to the NODE parameters, including encoder initial conditions, while enabling the most efficient computation of these sensitivities. The application of the 1st-CASAM-NODE has been illustrated by using the Nordheim-Fuchs reactor dynamics/safety phenomenological model, which is representative of physical systems that would be modeled by NODE while admitting exact analytical solutions for all quantities of interest (hidden states, decoder outputs, sensitivities with respect to all parameters and initial conditions, etc.). It has also been shown that if the equations underlying the physical model can be re-arranged so as to group the parameters/weights into functional "features" of several parameters, then the "First-Order Feature Adjoint Sensitivity Analysis Methodology for Nonlinear Systems (1st-FASAM-N)" can be advantageously applied to compute the response sensitivities with respect to the feature functions (which are by definition fewer than the number of parameters). The response sensitivities with respect to the primary parameters are subsequently obtained analytically by using the chain-rule to differentiate the components of the feature function with respect to the underlying model parameters and initial conditions. Applying the 1st-FASAM-N, however, would require the construction of a specific NODE for this purpose.

This work has also laid the foundation for the ongoing work on conceiving the "Second-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Neural Ordinary Differential Equations (2nd-CASAM-NODE)" which aims at yielding exact expressions for the second-order sensitivities of NODE decoder-responses to the NODE parameters and initial conditions while enabling the most efficient computation of these sensitivities.

Funding: This research received no external funding.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Haber, E.; Ruthotto, L. Stable architectures for deep neural networks. *Inverse problems*, **2017**, 014004.
2. Lu, Y; Zhong, A.; Li, Q.; Dong, B. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. *International Conference on Machine Learning*, PMLR, **2018**, 3276–3285.
3. Ruthotto, L.; Haber, E. Deep neural networks motivated by partial differential equations. *Journal of Mathematical Imaging and Vision*, **2018**, 352–364.
4. Chen, R.T.Q.; Rubanova, Y.; Bettencourt, J.; Duvenaud, D.K. Neural ordinary differential equations. In *Advances in Neural Information Processing Systems*, *31*, **2018,** pp. 6571–6583. Curran Associates, Inc., 2018. arXiv:1806.07366v5 [cs.LG] 14 Dec 2019
5. Dupont, E.; Doucet, A.; The, Y.W. Augmented neural odes. *Advances in Neural Information Processing Systems*, *32*, **2019**, 14–15.
6. Kidger, P.; On Neural Differential Equations, arXiv e-prints (2022), arXiv:2202.02435.
7. Kidger, P.; Morrill, J.; Foster, J.; Lyons, T.; Neural controlled differential equations for irregular time series, *Advances in Neural Information Processing Systems*, **2020**, *33*, 6696–6707
8. Morrill, J. Salvi, C.; Kidger, P.; Foster, J. Neural rough differential equations for long time series, *International Conference on Machine Learning*, PMLR, **2021**, 7829–7838.
9. Grathwohl, W., Chen, R. T. Q., Bettencourt, J., Sutskever, I., and Duvenaud, D. Ffjord: Free-form continuous dynamics for scalable reversible generative models. *International Conference on Learning Representations*, **2019**.
10. Zhong, Y. D., Dey, B., and Chakraborty, A. Symplectic ode-net: Learning Hamiltonian dynamics with control. In *International Conference on Learning Representations*, **2020**.
11. Tieleman, T.; Hinton, G. Lecture 6.5—RMSProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, **2012**.
12. Kingma, D. P.; Ba, J. Adam: A method for stochastic optimization. *International Conference on Learning Representations*, **2015**.
13. Pontryagin, L.S. *Mathematical Theory of Optimal Processes*. CRC Press, Boca Raton, FL, USA, 1987.
14. LeCun, Y.; Touresky, D.; Hinton, G.; Sejnowski, T. A theoretical framework for back-propagation. In *Connectionist Models Summer School*, **1988**.
15. LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, *86(11)*, **1998**, 2278–2324.
16. Norcliffe, A.; Deisenroth, M.P. Faster training of neural ODEs using Gauss–Legendre quadrature. *Transactions on Machine Learning Research*, 08/**2023**. Code available at: https://github.com/a-norcliffe/torch_gq_adjoint
17. Lamarsh, J.R. *Introduction to Nuclear Reactor Theory*, Adison-Wesley Publishing Co., Reading MA, USA; 1966; pp. 491-492.
18. Hetrick, D. L. *Dynamics of Nuclear Reactors*, American Nuclear Society, Inc., La Grange Park, IL., USA, 1993; pp. 164-174.
19. Cacuci, D.G. Computation of high-order sensitivities of model responses to model parameters. II: Introducing the Second-Order Adjoint Sensitivity Analysis Methodology for Computing Response Sensitivities to Functions/Features of Parameters," *Energies*, **16**, 2023, 6356. https://doi.org/10.3390/en16176356
20. Tukey, J.W. (1957) *The Propagation of Errors, Fluctuations and Tolerances*; Technical Reports No. 10–12; Princeton University. Princeton, NJ, USA, 1957.
21. Cacuci, D.G. *The nth-Order Comprehensive Adjoint Sensitivity Analysis Methodology (nth-CASAM): Overcoming the Curse of Dimensionality in Sensitivity and Uncertainty Analysis, Volume I: Linear Systems*. 362 pages, Springer Nature Switzerland, Cham, 2022; https://doi.org/10.1007/978-3-030-96364-4.
22. Cacuci, D.G. The Fourth-Order Comprehensive Adjoint Sensitivity Analysis Methodology for Nonlinear Systems (4^{th}-CASAM-N): I. Mathematical Framework. *Journal of Nuclear Eng*., **2022**, *3(1)*, 37–71. https://doi.org/10.3390/jne3010004jne-1568378.
23. Cacuci, D.G. (1981a). Sensitivity theory for nonlinear systems: I. Nonlinear functional analysis approach. *J. Math. Phys*., **1981**, *22*, pp. 2794–2812.
24. Cacuci, D.G. Introducing the n^{th}-Order Features Adjoint Sensitivity Analysis Methodology for Nonlinear Systems (n^{th}-FASAM-N): I. Mathematical Framework," *Am. J. Comp. Math*, **2024**, *14*, 11-42. https://doi.org/10.4236/ajcm.2024.141002