Volterra Integral Equations: From Theory to Battery SOH Forecasting

Denis N. Sidorov
Sino-Russian Joint Research Center for Advanced Energy and Power Systems, Melentiev Energy Systems Institute SB RAS, Irkutsk, Russia
Joint work with Dao Minh Hien, Aliona Dreglea, Vlad Byankin (INRTU)

Dynamics in Siberia, Novosibirsk, 2026

Plan of the Talk

1. Volterra operator as amper-hour linear model for SoC
2. Generalization on nonlinear nonstationary case (in term of kernel)
3. Application to autonomous microgrid with energy storage and energy community
4. New classes: Volterra and Hammerstain IE with local and nonlocal loads
5. Volterra layer in ANN helps to build an efficient architecture for various storage systems SoH prediction
6. Robustness to delays and data acquisition defects
7. Concluding remarks and future work

Classic Ampere-hour Model based on the Volterra operator

The direct problem

$$SOC(t) = SOC(0) + \int_0^t \mu(\cdot) i(\tau) d\tau$$

$$\begin{cases} \int_0^t K(t,\tau,x(\tau))\,d\tau = f(t), \;\; 0 \le \tau \le t \le T, \\ v(t) = \int_0^t x(\tau)\,d\tau, \;\; \max\limits_{t\in[0,T]} |v(t)| \le v_{max}, \\ E_{min}(t) \le \int_0^t v(\tau)\,d\tau \le E_{max}(t). \end{cases} \tag{1}$$

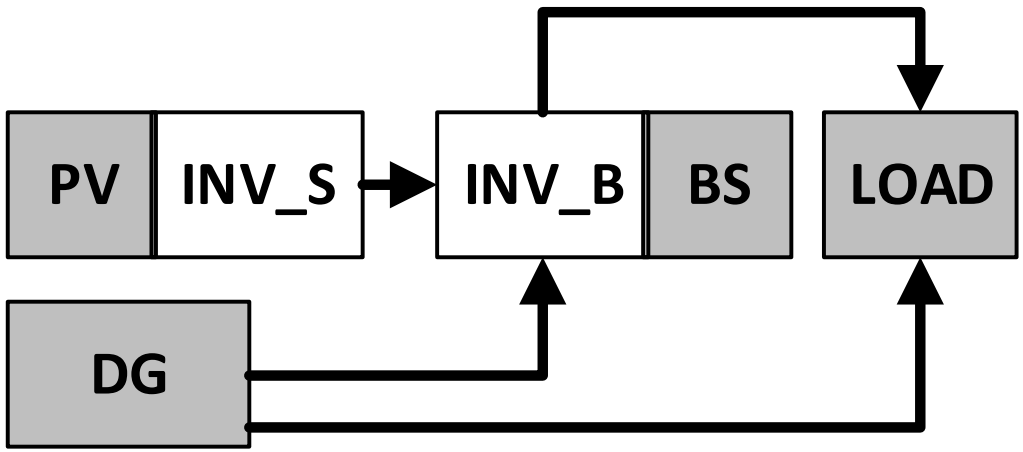
Here source function $f(t)$ is the energy imbalance defined as follows

$$f(t) = f_{PV}(t) - f_{load}(t),$$

where $f_{PV}(t)$ is the PV generation, and $f_{load}(t)$ is the electric load.

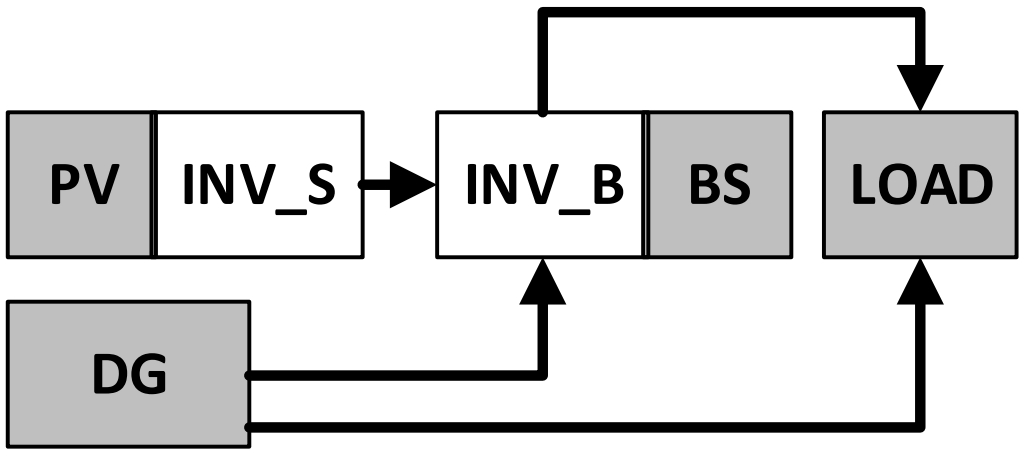
Case Study for Autonomous ES' Dynamical Model

The battery operation modes modeling is demonstrated using the example of an autonomous hybrid energy system. It is assumed that this hybrid system uses: photovoltaic arrays (PV), solar inverters (INV_S), battery inverters (INV_B), battery energy storage (BS), and diesel generator (DG).



Case Study for Autonomous ES' Dynamical Model

The battery operation modes modeling is demonstrated using the example of an autonomous hybrid energy system. It is assumed that this hybrid system uses: photovoltaic arrays (PV), solar inverters (INV_S), battery inverters (INV_B), battery energy storage (BS), and diesel generator (DG).



Sidorov D., Muftahov I., Tomin N., Karamov D., Panasetsky D., Dreglea A., Liu F., Foley A. A Dynamic Analysis of Energy Storage With Renewable and Diesel Generation Using Volterra Equations // IEEE Transactions on Industrial Informatics. 2020. Vol. 16. No. 5. pp. 3451 – s3459.

Installed capacities are as follows: PV is 75 kW, INV_s is 75 kW, INV_b is 72 kW, DG is 2x100 kW and BS is 384 kWh. The maximum load is 47 kW. Modeling the operation of a solar power station is performed using actinometric data recorded in the territory under consideration

Linear Model

The classic discrete model with constant efficiency commonly used in the literature

$$SOC(t) = SOC(t-1) + I_s(t)\Delta t \tag{2}$$

with constrain $I_s(t) \leq r_{BS} Q_{BS}^{max}$ where $SOC(t-1)$ (kW) is battery SOC in time $t-1$, $I_s(t)$ is the alternating power function (kW); r_{BS} is technical restriction on the charge and discharge of battery (from 20% till 40%); Q_{BS}^{max} is installed battery capacity, kWh. If the battery is charged then $I_s(t)$ is multiplied by a constant efficiency of the battery and inverter. This value is assumed to be 0.8 as suggested by Stevens96. Δt is a discrete step (1 hour) to determine the amount of energy released to the battery.

In order to efficiently model the storage operation, the following integral dynamical model with constraints is employed

$$\begin{cases} \int_0^t K(t,\tau,x(\tau))\,d\tau = f(t), \;\; 0 \le \tau \le t \le T, \\ v(t) = \int_0^t x(\tau)\,d\tau, \;\; \max\limits_{t\in[0,T]} |v(t)| \le v_{max}, \\ E_{min}(t) \le \int_0^t v(\tau)\,d\tau \le E_{max}(t). \end{cases} \tag{3}$$

Here source function $f(t)$ is the energy imbalance defined as follows

$$f(t) = f_{PV}(t) - f_{load}(t),$$

where $f_{PV}(t)$ is the PV generation, and $f_{load}(t)$ is the electric load. This imbalance is supposed to be covered by battery storage operated in parallel with diesel generator backup system.

In integral model the alternating function of changing the power $x(t)$ is the desired one. It allows for known maximum speed of the charge v_{max}:

1. to determine $E(t)$, which is the storage SoC under the constraints $E_{min}(t) \leq E(t) \leq E_{max}(t)$ depending on the type of storage;
2. to determine the minimum total capacity of the storage to cover the load shortage;
3. to calculate the number of cycles based on behaviour of the function $E(t)$;
4. the storage lifetime prediction for the specific region.

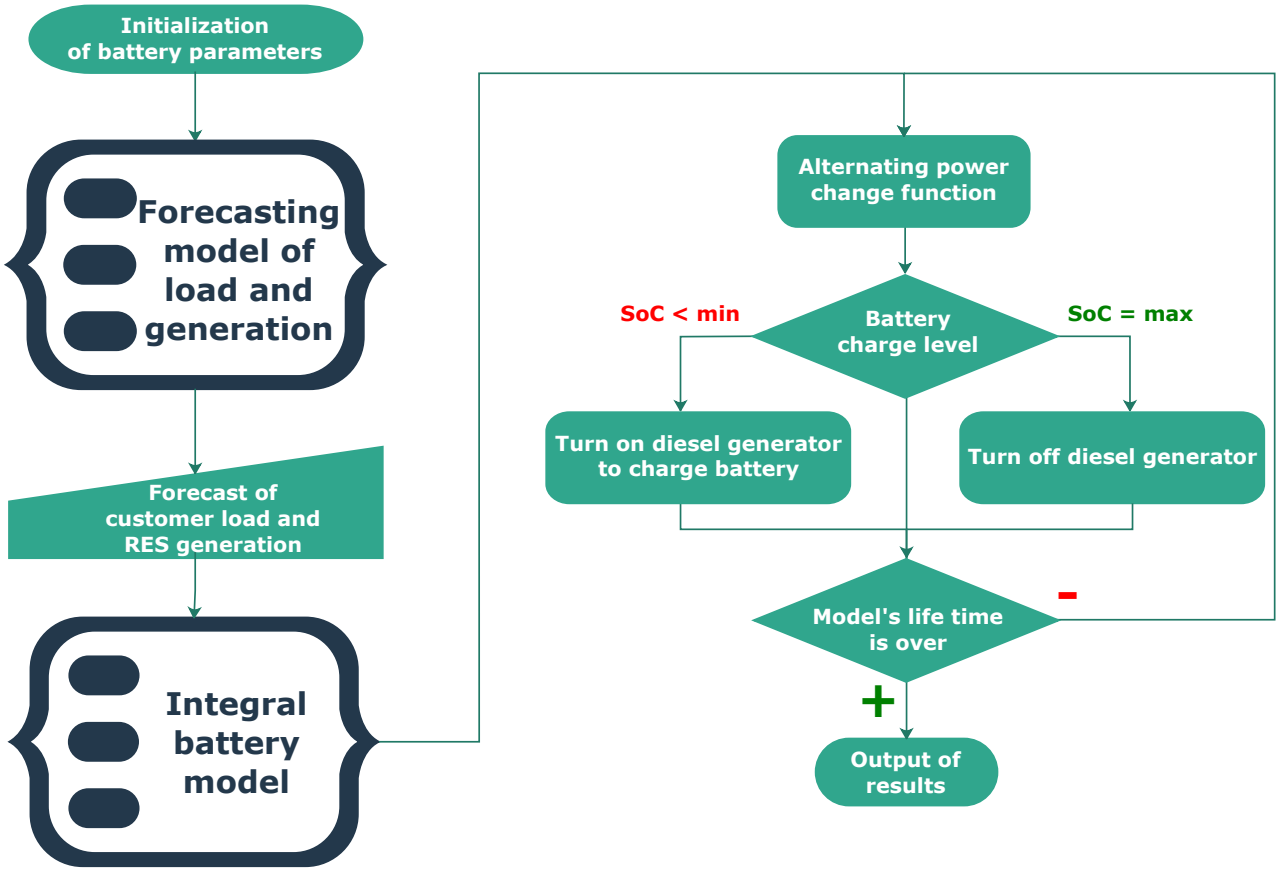


Figure 1: Data flow diagram of the battery storage system

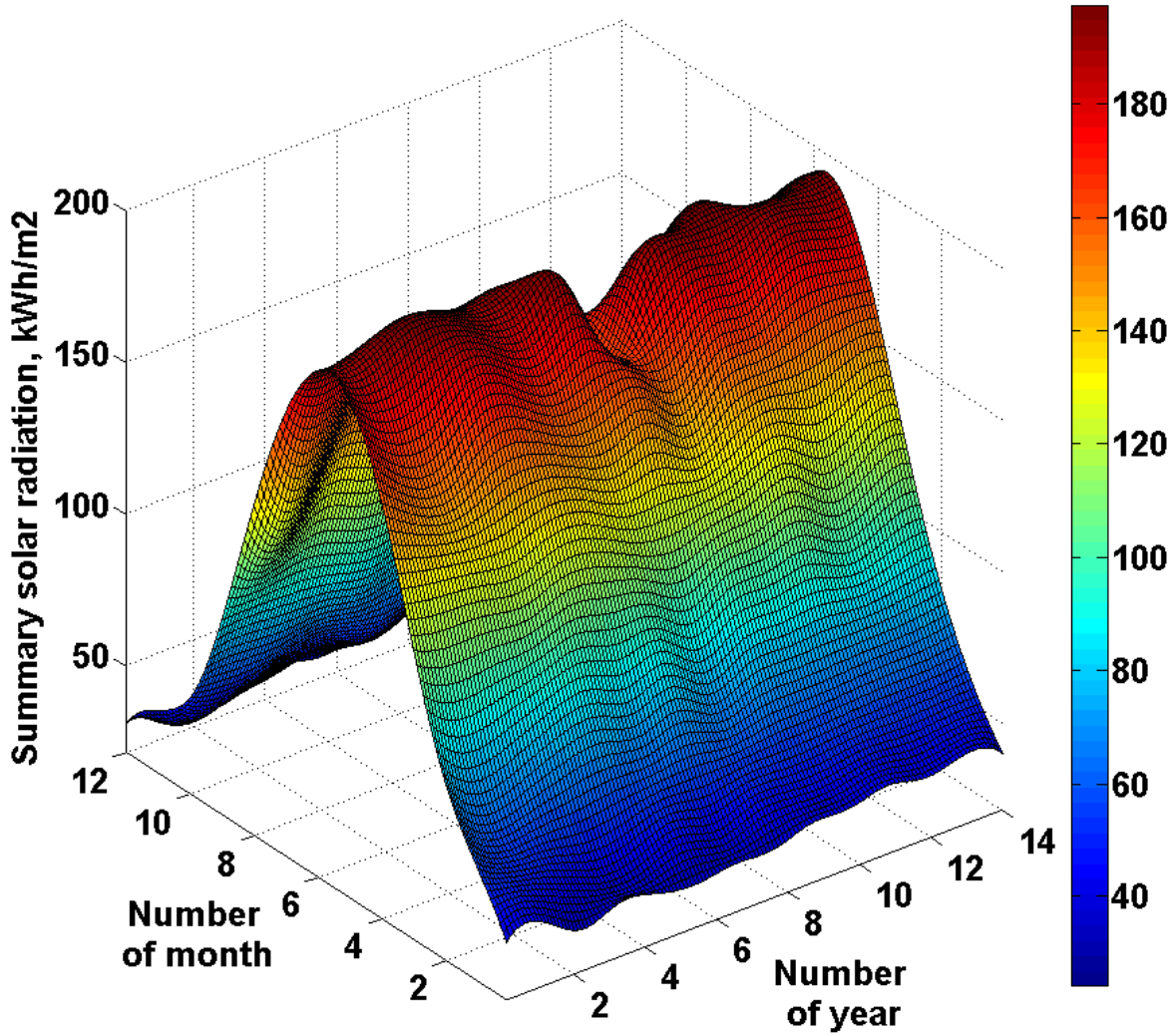


Figure 2: Cumulative solar radiation for the considered region

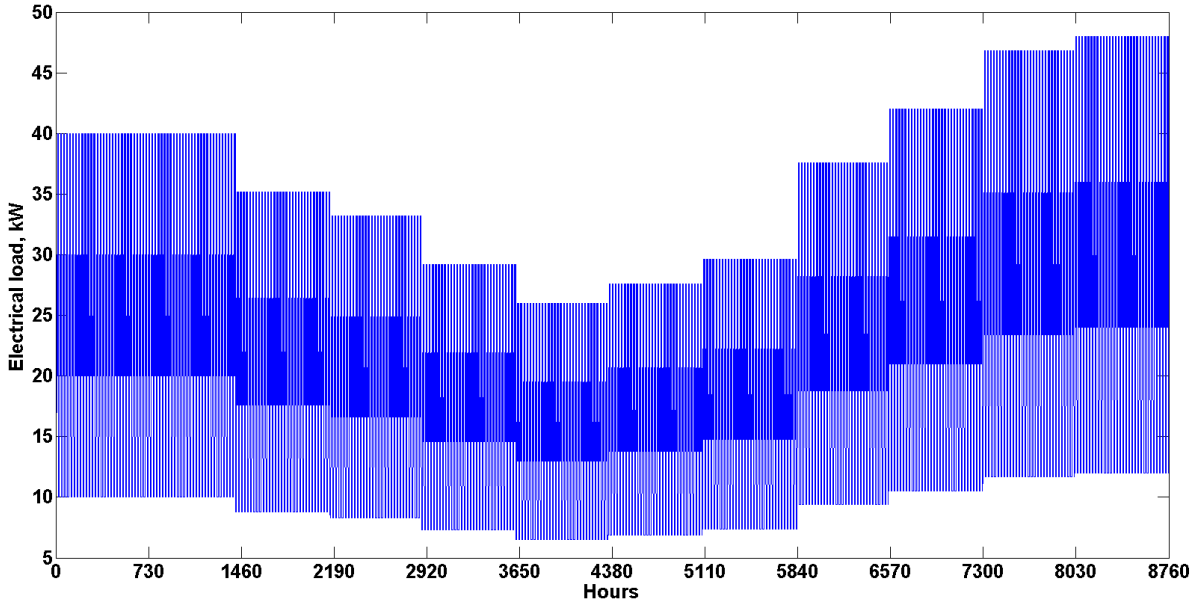


Figure 3: Electrical load during the year

The system is simulated over the entire period of annual meteorological observations with a discrete step of one hour (a total of 122640 steps). After the simulation, the average annual value of the accumulated energy and monthly average values of SoC are calculated.

Lead-carbon batteries adapted to heavy cyclic modes were used in the simulation. The study addresses two cases:

- case 1: the linear model of a battery with a constant efficiency.
- case 2: the Volterra integral model with constant efficiency.

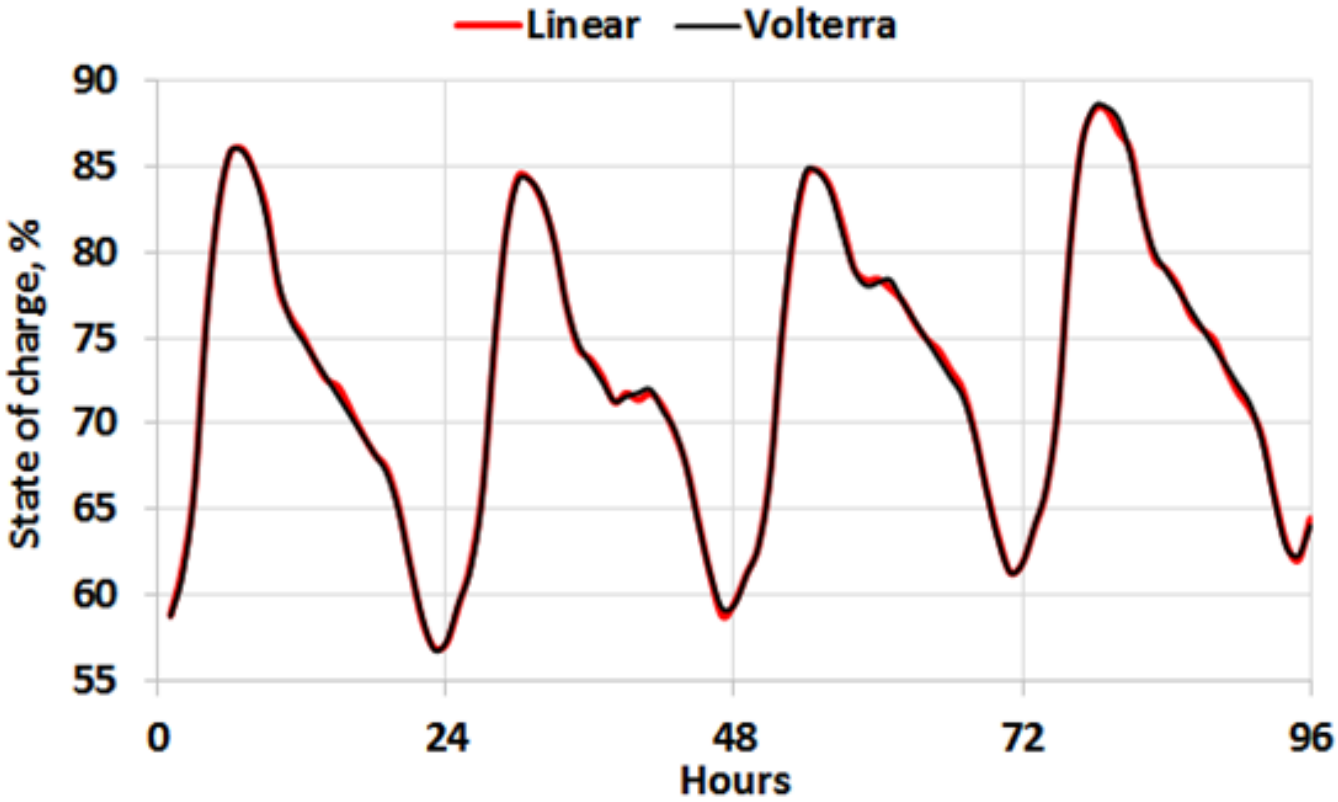


Figure 4: SoC for conventional linear model and the Volterra model

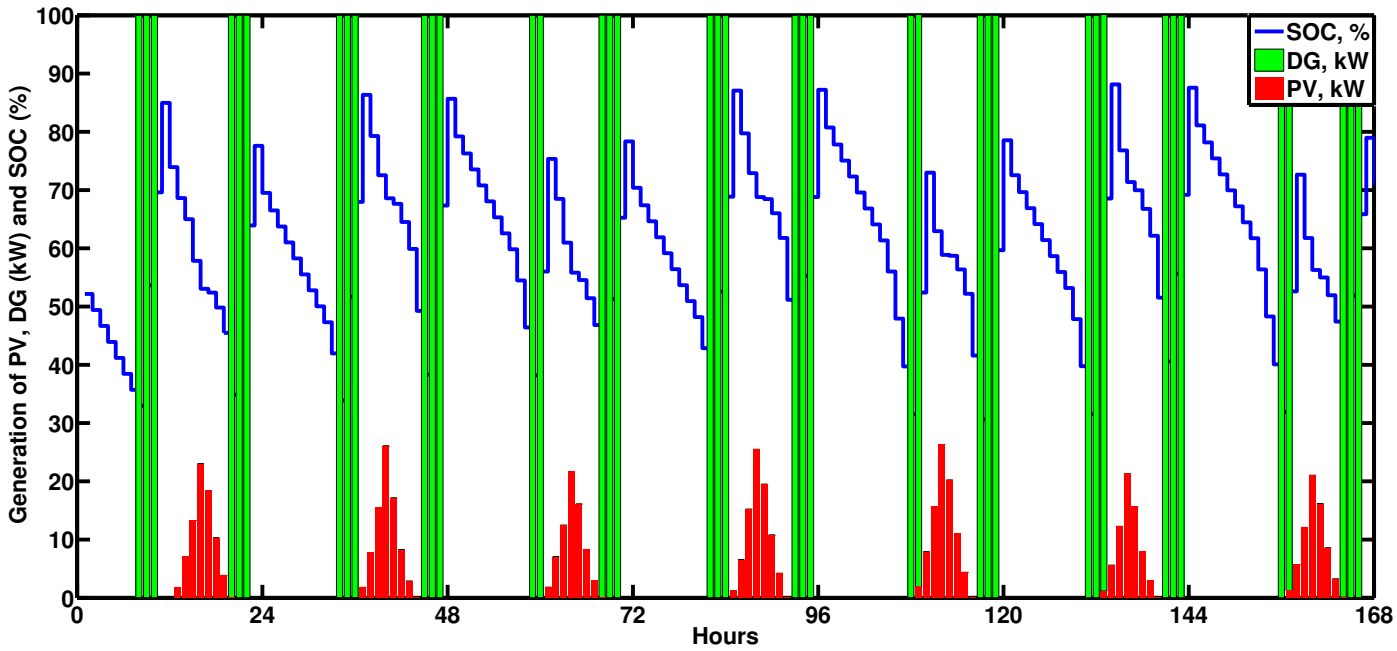


Figure 5: Dynamic analysis of SoC of the battery using the Volterra model with PV and diesel (January)

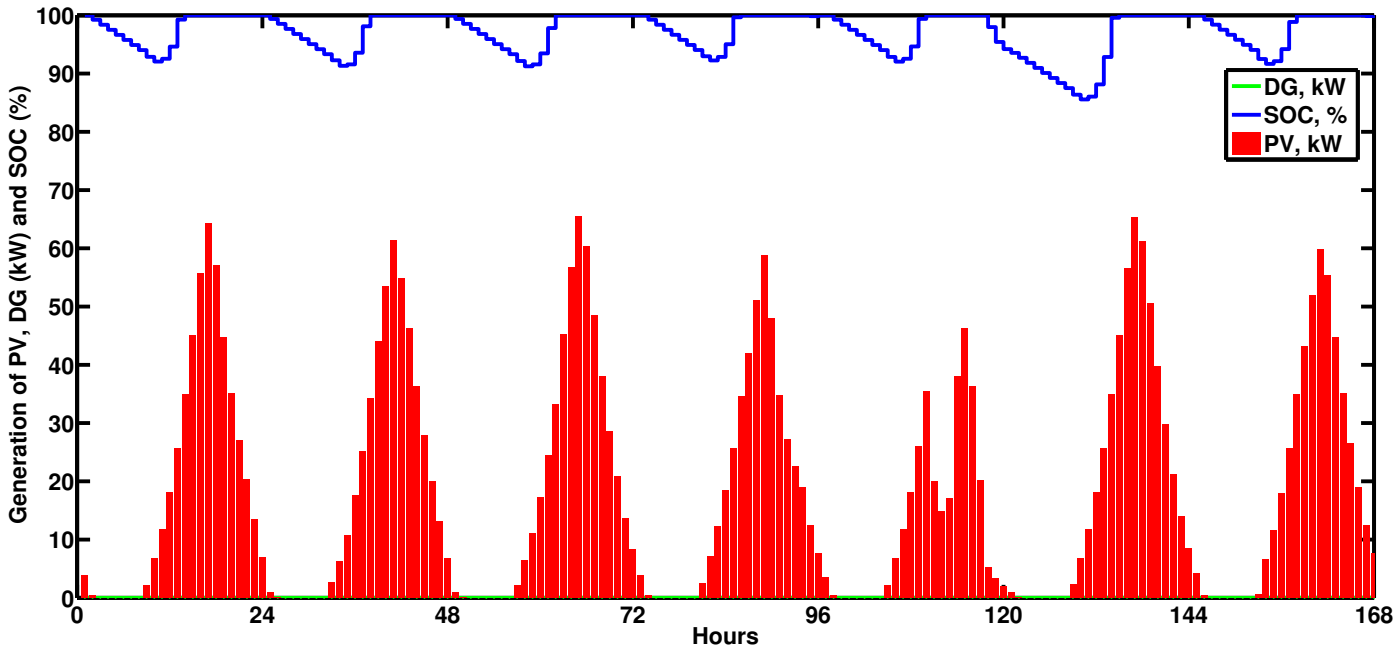
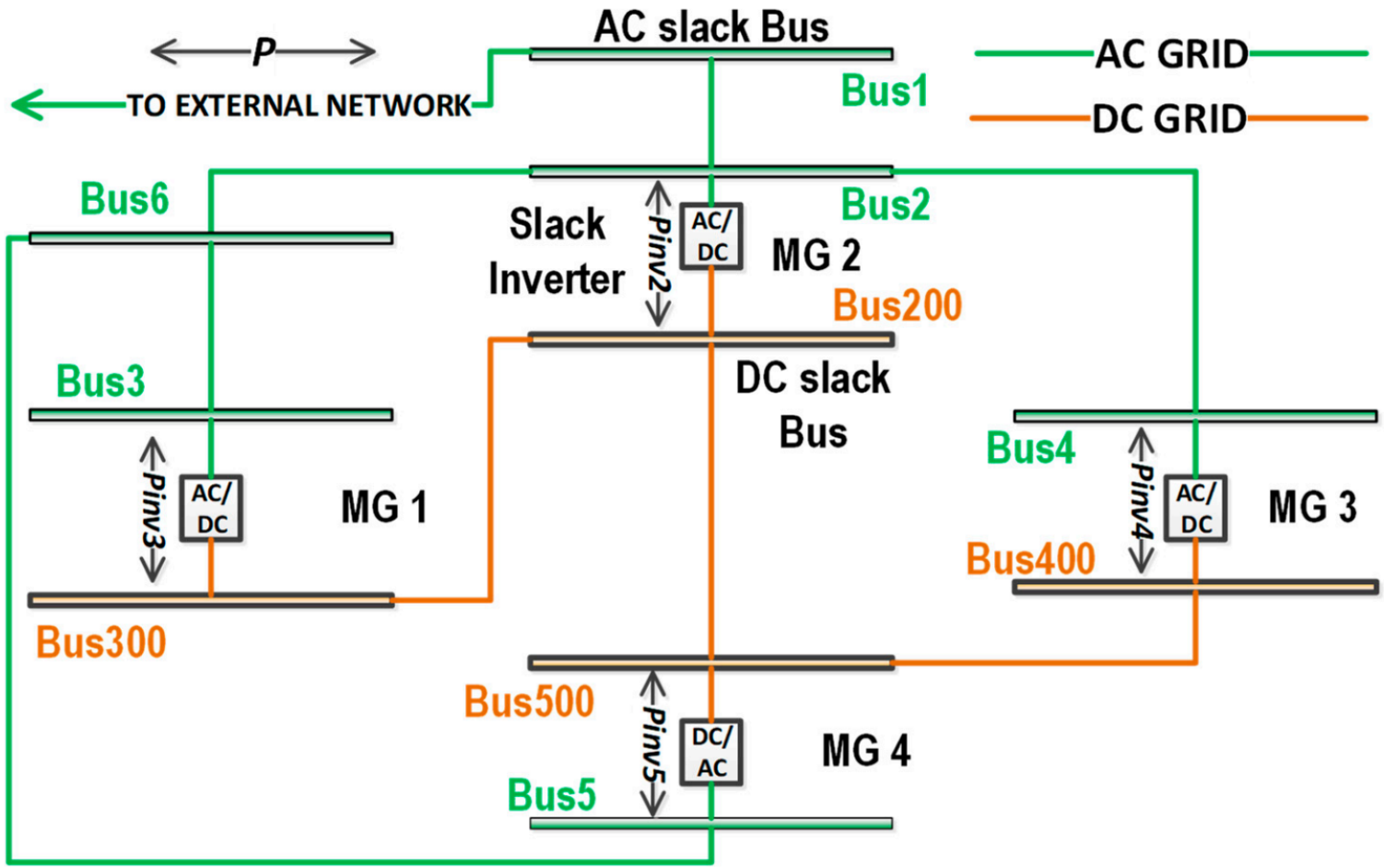
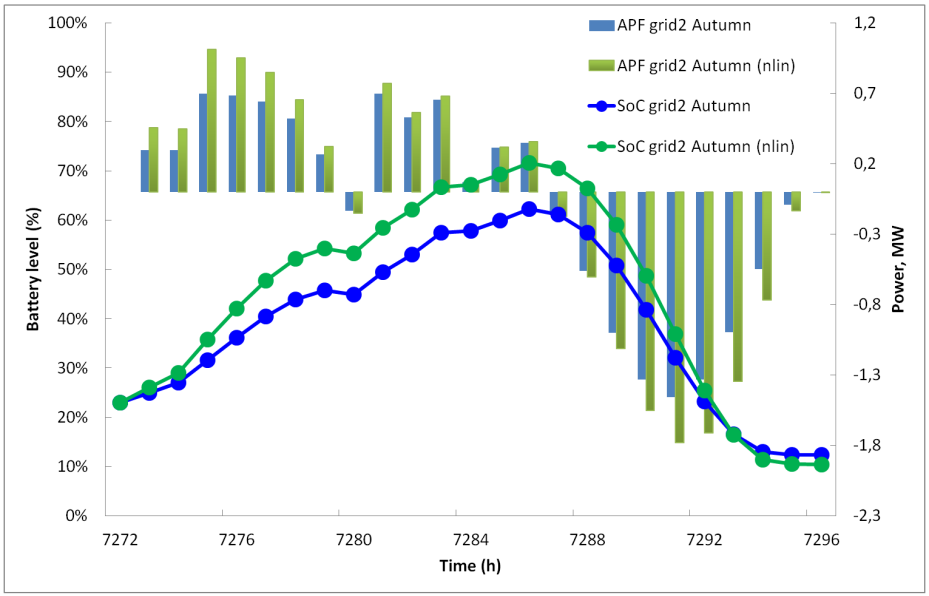


Figure 6: Dynamic analysis of SoC of the battery using the Volterra model with PV and diesel (June)

The hybrid AC/DC test system



Alternating power functions (APF) in MW with SoC (in %)



Error analysis

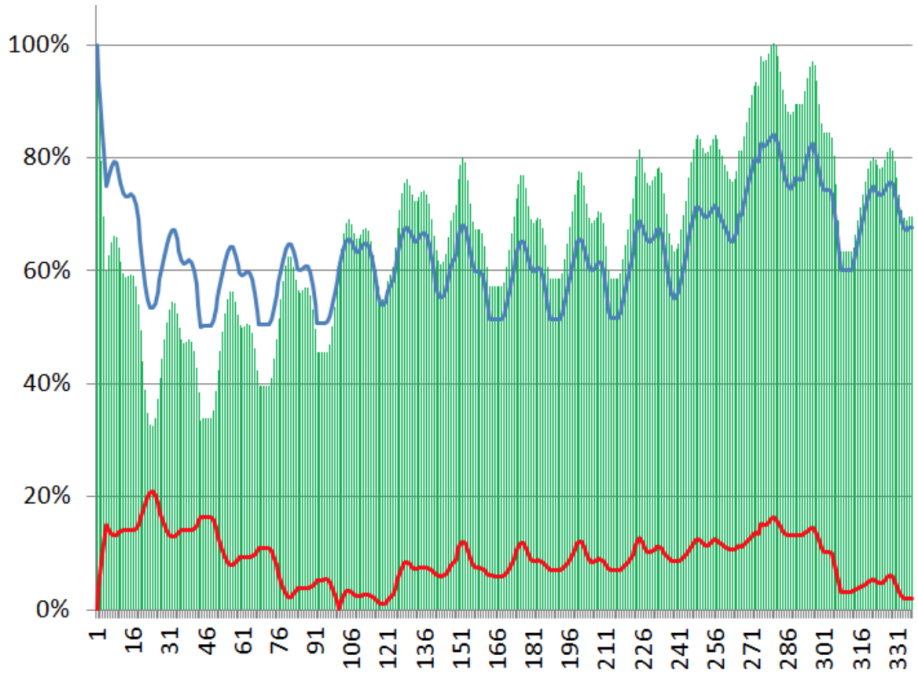


Figure 7: SoC: variable efficiency (green); constant efficiency (blue) of the battery using the Volterra model with PV and diesel (June)

Loaded Volterra Integral Equation

$$a_0(t)x(t)+\sum_{j=1}^{m-1} a_j(t)x(t_j) = \lambda \int_{t_0}^{t} K(t,s)x(s)ds+f(t), \quad t \in \Omega = [t_0, T]$$

where $t_j, (j = 1, 2, \ldots, m-1)$ are from $[t_0, T]$,
$t_0 < t_1 < \cdots < t_{m-1} < t_m = T$; λ is parameter;
$f(t), a_j(t) \in C_{[t_0,T]}, j = \overline{0, m-1}$.

Loaded Volterra Integral Equation

$$a_0(t)x(t)+\sum_{j=1}^{m-1} a_j(t)x(t_j) = \lambda \int_{t_0}^{t} K(t,s)x(s)ds+f(t), \quad t \in \Omega = [t_0, T]$$

where $t_j, (j = 1, 2, \ldots, m-1)$ are from $[t_0, T]$,
$t_0 < t_1 < \cdots < t_{m-1} < t_m = T$; λ is parameter;
$f(t), a_j(t) \in C_{[t_0,T]}, j = \overline{0, m-1}$.

Sidorov N.A., Sidorov D.N. Nonlinear Volterra Equations with Loads and Bifurcation Parameters: Existence Theorems and Construction of Solutions. Differential Equations, 2021, vol. 57, pp. 1640–1651.

Loaded Volterra Integral Equation

Solution is constructed as piecewise - linear approximation

$$a_0(\tau_i)x_i + \sum_{j=1}^{m-1} a_j(\tau_i)x_{v_i} =$$

$$\lambda \sum_{p=1}^{i} \int_{\tau_{p-1}}^{\tau_p} K(\tau_i, s)\left(x_{p-1} + \frac{x_p - x_{p-1}}{\tau_p - \tau_{p-1}}(s - \tau_{p-1})\right)ds + f(\tau_i),$$

where v_i is number of mesh point $\{\tau_i\}_{i=0}^{N}$, same as $t_j,\ j = 1, 2, \ldots, m-1$. Values v_i are defined for specific step size h.

Loaded Volterra Integral Equation

Model Example 1:

$$t \in [t_0, T] = [0, 1];\ m = 3;\ t_1 = \frac{3}{10};\ t_2 = \frac{1}{2};\ \lambda = \frac{1}{4};$$

$$a_0(t) = t^2 + 1;\ a_1(t) = 1 - t^3;\ a_2(t) = t - 2;\ K(t, s) = t - 2s^2;$$

$$f(t) = (t^2 + 1)\cos t + (1 - t^3)\cos\left(\frac{3}{10}\right) +$$

$$+(t - 2)\cos\left(\frac{1}{2}\right) + \frac{t^2}{2}\sin t - \frac{t}{4}\sin t + t\cos t - \sin t.$$

$x(t) = \cos(t)$

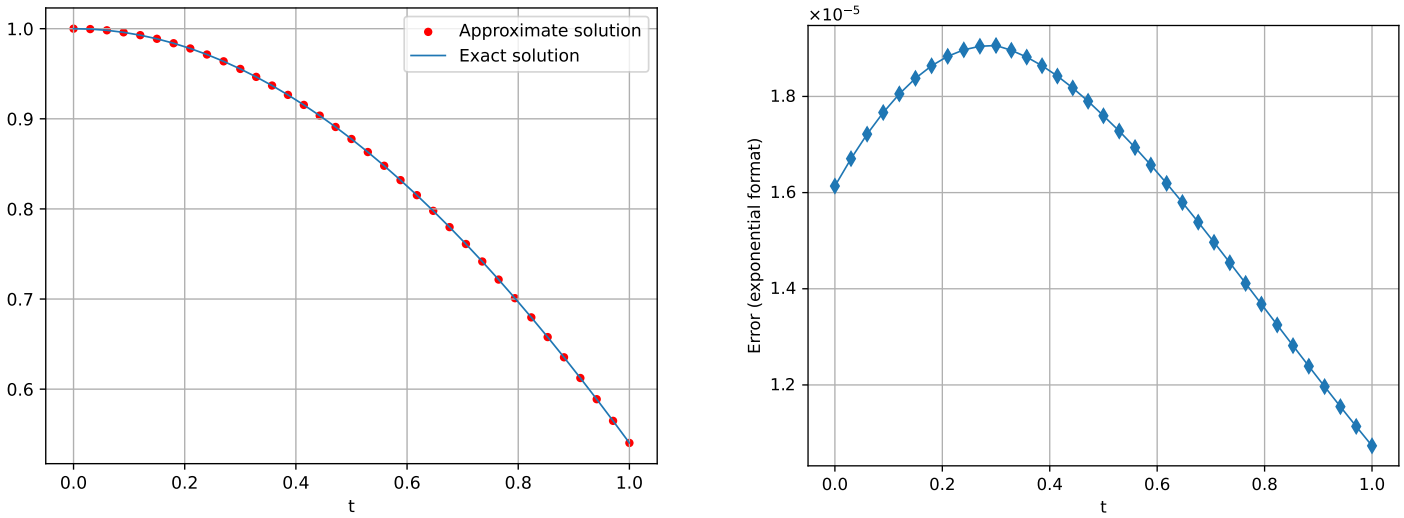
Loaded Volterra Integral Equation

$\varepsilon_h = \|X_N(t) - x(t)\|_{C_{[t_0,T]}}$ error, and $r = \dfrac{\ln \frac{\varepsilon_{h_{k-1}}}{\varepsilon_{h_k}}}{\ln \frac{h_{k-1}}{h_k}}$ is convergence order.

$\boldsymbol{h}$	$1/8$	$1/16$	$1/32$	$1/64$	$1/128$	$1/256$
ε_h	2.20E-04	6.99E-05	1.91E-05	5.04E-06	1.30E-06	3.30E-07
r	—	1.65	1.88	1.92	1.96	1.98

$\boldsymbol{h}$	$1/512$	$1/1024$	$1/2048$	$1/4096$	$1/8192$	$1/16484$
ε_h	8.31E-08	2.09E-08	5.23E-09	1.31E-09	3.27E-10	8.18E-11
r	1.99	1.99	1.93	2.07	2.00	2.00

Loaded Volterra Integral Equation



(a) **(b)**

Figure 8: Comparative analysis.

Loaded Volterra Integral Equation

Example 2:

$$f(t) = 2\sin(t)te^{-t^2} + \frac{3e^t e^{-\frac{9}{4}}}{2} + \\ + \frac{3te^{-\frac{9}{25}} + 24\cos(t)e^{-\frac{36}{25}} + 12(t^4+t)e^{-\frac{144}{25}}}{5} + \\ + \frac{19(5^t + 100t)e^{-\frac{361}{100}}}{10} - \\ - \frac{(t^3e^{t^2} + \sqrt{\pi}\mathrm{erf}(t)e^{t^2} - t^3 + t^2 - e^{t^2} - 2t + 1)e^{-t^2}}{16}$$

$t \in [t_0, T] = [0, 3];\ m = 6;\ t_1 = \frac{6}{10};\ t_2 = \frac{12}{10};\ t_3 = \frac{15}{10};\ t_4 = \frac{19}{10};\ t_5 = \frac{24}{10};$
$\lambda = \frac{1}{8};\ K(t,s) = 2s - s^2 + t^3;$
$a_0(t) = 2\sin(t);\ a_1(t) = t;\ a_2(t) = 4\cos(t);\ a_3(t) = e^t;\ a_4(t) = 5^t + 100t;\ a_5(t) = t^4 + t;$
Exact Solution $x(t) = te^{-t^2}$

Loaded Volterra Integral Equation

$\boldsymbol{h}$	$1/8$	$1/16$	$1/32$	$1/64$	$1/128$
ε_h	8.54E-05	3.22E-05	9.96E-06	2.76E-06	7.21E-07
r	—	1.41	1.69	1.85	1.93

$\boldsymbol{h}$	$1/256$	$1/512$	$1/1024$	$1/2048$
ε_h	1.85E-07	4.68E-08	1.18E-08	2.95E-09
r	1.97	1.98	1.99	1.93

Byankin, V., Tynda, A., Sidorov, D., Dreglea, A. (2025). Numerical solution of locally loaded Volterra integral equations. Computation, 13(5), 121.

Loaded Hammerstain Integral Equation

$$y(t) = \int_a^b K(t,s)\Psi(s,y(s))ds + f(t).$$

Loaded HIE

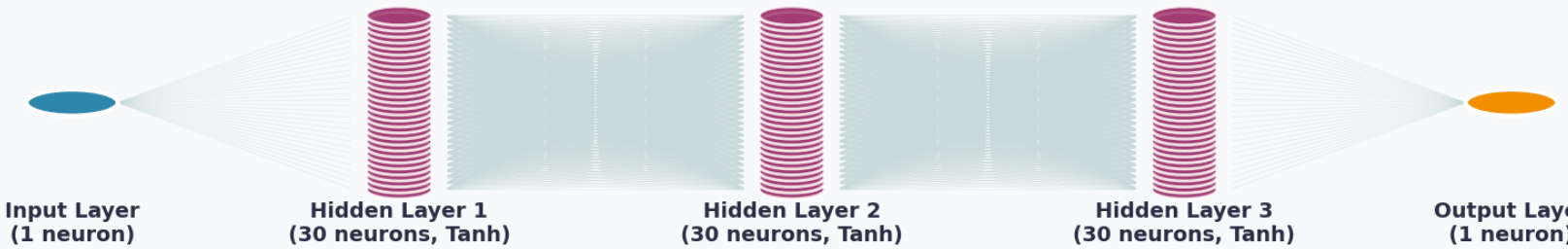
$$y(t) = \int_a^b G(t,s) \sum_{i+k\geq 2}^{\infty} f_{ik}(s,\lambda)y(s)^i y_\alpha^k ds + f(t,\lambda)y_\alpha \qquad (4)$$

where $t \in [a,b],\ \lambda \in R,\ y_\alpha = \int_a^b \alpha(t)y(t)dt$ and $y_\alpha = y(\alpha)$ such as $\alpha \in [a,b]$.

Loaded Hammerstain Integral Equation

HAM-RaNN Framework for Hummerstein Integral Equations

RaNN Architecture: [1 → 30 → 30 → 30 → 1] with Adam Optimizer (Fully Connected)



HAM-RaNN Solution Process

Step 1: HAM Solution
$x_{HAM}(t) = \sum_{n=0}^{m} x_n(t)$

Step 2: Compute Residual
$R(t) = x_{RaNN}(t) - \mathcal{H}[x_{RaNN}](t)$

Step 3: Train RaNN
$u(t) = \mathrm{RaNN}(t, \theta)$ learns residual
Epochs: 5000, lr: 0.001, Adam

Step 4: HAM-RaNN Solution
$x_{RaNN}(t) = x_{HAM}(t) + u(t, \theta^*)$
Enhanced accuracy by HAM-RaNN

Mathematical Formulation

Hammerstein Equation: $x(t) = \int_a^b K(t,s)\, g(x(s))\, ds + f(t)x(\alpha)$

HAM Approximation: $x_{HAM}(t) = \sum_{n=0}^{m} x_n(t)$

RaNN Correction: $u(t, \theta) = \mathrm{RaNN}(t, \theta)$

Combined Solution: $x_{RaNN}(t, \theta) = x_{HAM}(t) + u(t, \theta)$

Residual Function: $R(t; \theta) = x_{RaNN}(t, \theta) - \mathcal{H}[x_{RaNN}](t)$

Loss Function: $\mathcal{L}(\theta) = \frac{1}{N} \sum_{i=1}^{N} [R(t_i; \theta)]^2$

Optimization: $\theta^* = \arg\min_{\theta} \mathcal{L}(\theta)$ (Adam, 5000 epochs)

Final Solution: $x_{RaNN}(t) = x_{HAM}(t) + u(t, \theta^*)$

Loaded Hammerstain Integral Equation

Stochastic arithmetic CESTAC was used to control accuracy.
RaNN was used:

$$Y(x) \approx \sum_{i=1}^{N} \beta_i \, \sigma(W_i x + b_i),$$

$$R(t,\theta) = \Big(U_{\mathsf{HAM}}(t) + \mathsf{RaNN}(t,\theta) - \mathcal{H}[U_{\mathsf{HAM}} + \mathsf{RaNN}](t,\theta) \Big)^2.$$

J. Vignes. A stochastic arithmetic for reliable scientific computation. Mathematics and Computers in Simulation. Volume 35, Issue 3, September 1993, Pages 233 – 261.

Loaded Hammerstain Integral Equation

Example

$$y(t) = \int_0^1 tsy^3(s)ds + f(t)y(\alpha),$$

where $f(t,\lambda) = 1 - \frac{11191}{9240}t + t^3$, and exact solution is $x(t) = t^3 - t + 1$.

After application of HAM with $\hbar = -1$ for one iteration $(n = 5)$, we get an approximate solution

$$y_{HAM}(t) = 1 - 1.01754t + t^3$$

Table 1: Comparative Analysis: HAM vs HAM-RaNN

Metric	**HAM**	**HAM-RaNN**	**Improvement%**
Max Error	1.754000E-02	4.587471E-05	99.74
MAE	8.77E-03	2.09E-05	99.76
MSE	1.03E-04	5.34E-10	100.00
R^2	0.9926	1.0000	0.74

Example 2:

$$y(t) = \int_0^1 ts\,\lambda\,[y(s)]^2[y(\alpha)]^2\,ds + f(t)y(\alpha), \tag{5}$$

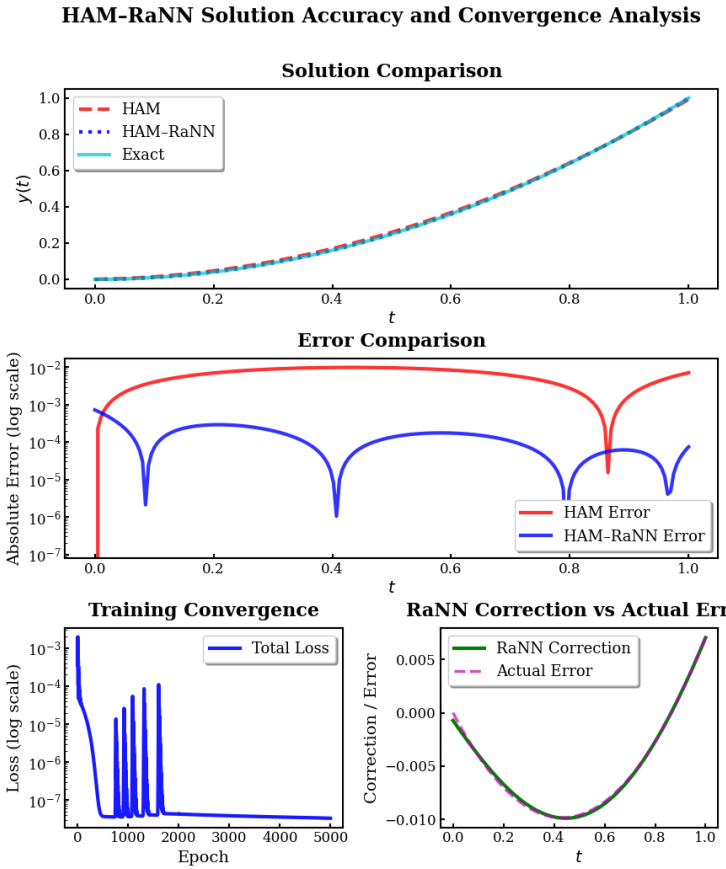
where $\alpha = 1.$, $\lambda = 2$ and $f(t) = (-\frac{t}{3} + t^2)$ with exact solution $y(t) = t^2$.
After one iteration $\hbar = -1,55$ $(n = 3)$, we get

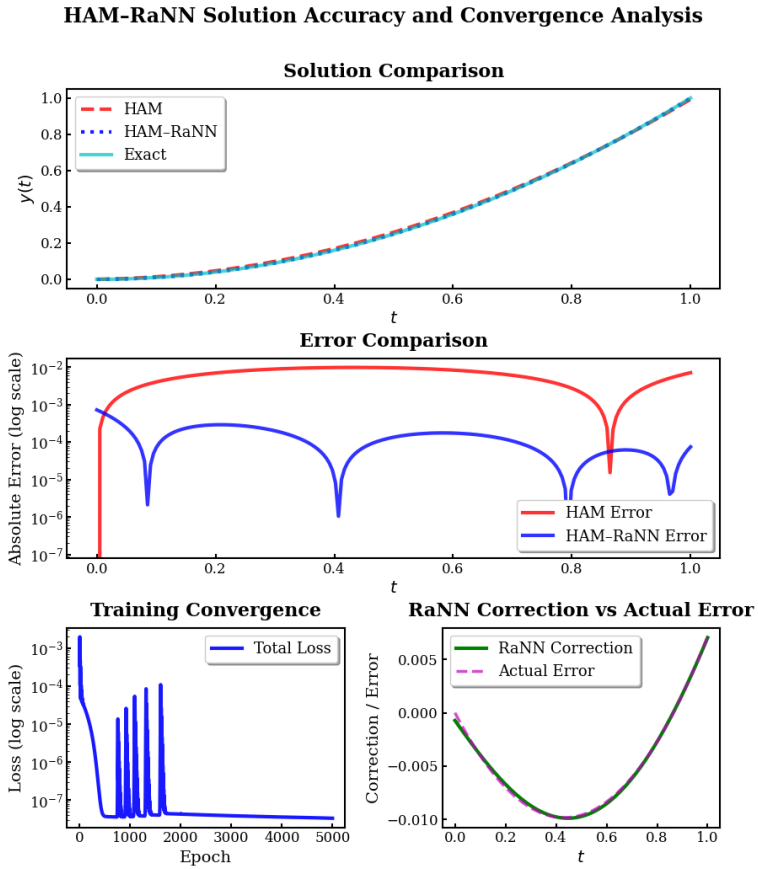
$$y_{HAM}(t) = 0,0455044t + 0,947373t^2$$

Table 2: HAM vs HAM-RaNN

Metric	**HAM**	**HAM-RaNN**	**Improvement%**
Max Error	9.836444E-03	7.237657E-04	92.64
MAE	6.12E-03	1.39E-04	97.73
MSE	4.67E-05	3.33E-08	99.93
R^2	0.9995	1.0000	0.05

Loaded Hammerstain IE





Volterra layer in ANN

$$\int_0^t K(\tau, \sigma(t-\tau))y(t-\tau)\,d\tau, \quad t \in [0, M]$$

where

$$K(\tau, \sigma) = \beta \exp(-\gamma\tau)\left[1 + \mathbf{w}^T f(\sigma)\right]$$

Applications

Challenges

- Li-ion battery aging is highly nonlinear
- Depends on operational history
- LSTM lacks physical interpretability
- Poor generalization under varying conditions

Our Solution

- TKAN-Volterra Hybrid Model
- Physics-informed approach
- History-dependent modeling
- Adaptive fusion mechanism

Applications: Electric Vehicles, Grid Energy Storage, Portable Electronics

Mathematical Foundations: TKAN

Kolmogorov-Arnold Representation Theorem

$$f(x_1, ..., x_n) = \sum_{q=1}^{2n+1} \Phi_q \left(\sum_{p=1}^{n} \phi_{q,p}(x_p) \right), \tag{6}$$

Any continuous multivariate function can be expressed as superposition of univariate functions

TKAN Key Features:

B-spline Activation
Learnable functions for nonlinear mapping

RKAN Layers
Recurrent KAN for sequential data

LSTM-style Gating
Forget, input, output gates for memory

RKAN Layer: $s_{l,t} = W_{l,x}\tilde{x}_t + W_{l,h}\tilde{h}_{l,t-1}$ — **Output:** $o_t = \sigma(W_0 \cdot r_t + b_0)$

Mathematical Foundations: Volterra Model

Volterra Integral Equation for Battery Degradation

$$y(t) = y_0(t) - \int_0^M K(\tau, \sigma(t-\tau)) \cdot y(t-\tau)\, d\tau \tag{7}$$

$$K(\tau, \sigma) = \beta \cdot \exp(-\gamma \cdot \tau) \cdot \left[1 + \mathbf{w}^T \cdot f(\sigma)\right] \tag{8}$$

Kernel Components:

$\beta \cdot \exp(-\gamma \cdot \tau)$
Exponential decay
(Fading memory)

$[1 + \mathbf{w}^T \cdot f(\sigma)]$
Load modulation

$f(\sigma) = [|\sigma_1|, |\sigma_2|, |\sigma_3|]$
Stress factors
(V, I, T)

Physical Principle: Harsh operating conditions $\rightarrow$ Accelerated capacity fade

Physics-Informed Feature Engineering

17 Features Extracted → 14 Features Used (after data leakage prevention)

Internal Resistance Estimation (3 Methods)

IR-Drop	Ohmic	Dynamic
$R = \frac{\Delta V}{\Delta I}$	$R = \frac{\Delta V_{inst}}{\Delta I}$	$R(t) = \frac{V(t) - V_0 - \Delta V}{\Delta I}$

Feature Groups:

Voltage	Current	Thermal	Cycle
mean, std, min, max, dV/dt	mean, std, power indicators	temp mean/max, stress factors	phase (early/mid/late)

Data Leakage Prevention: Excluded capacity, resistance_avg, soh_degradation_rate

Normalization: RobustScaler for outlier handling

State of Health (SoH) Definition

Capacity-Based SoH (70%)

$$SoH_C = \frac{C_{current}}{C_{initial}} \times 100\% \quad (9)$$

EoL threshold: 30% capacity fade

Power-Based SoH (30%)

$$SoH_P = \frac{R_{eol} - R}{R_{eol} - R_{init}} \times 100\% \quad (10)$$

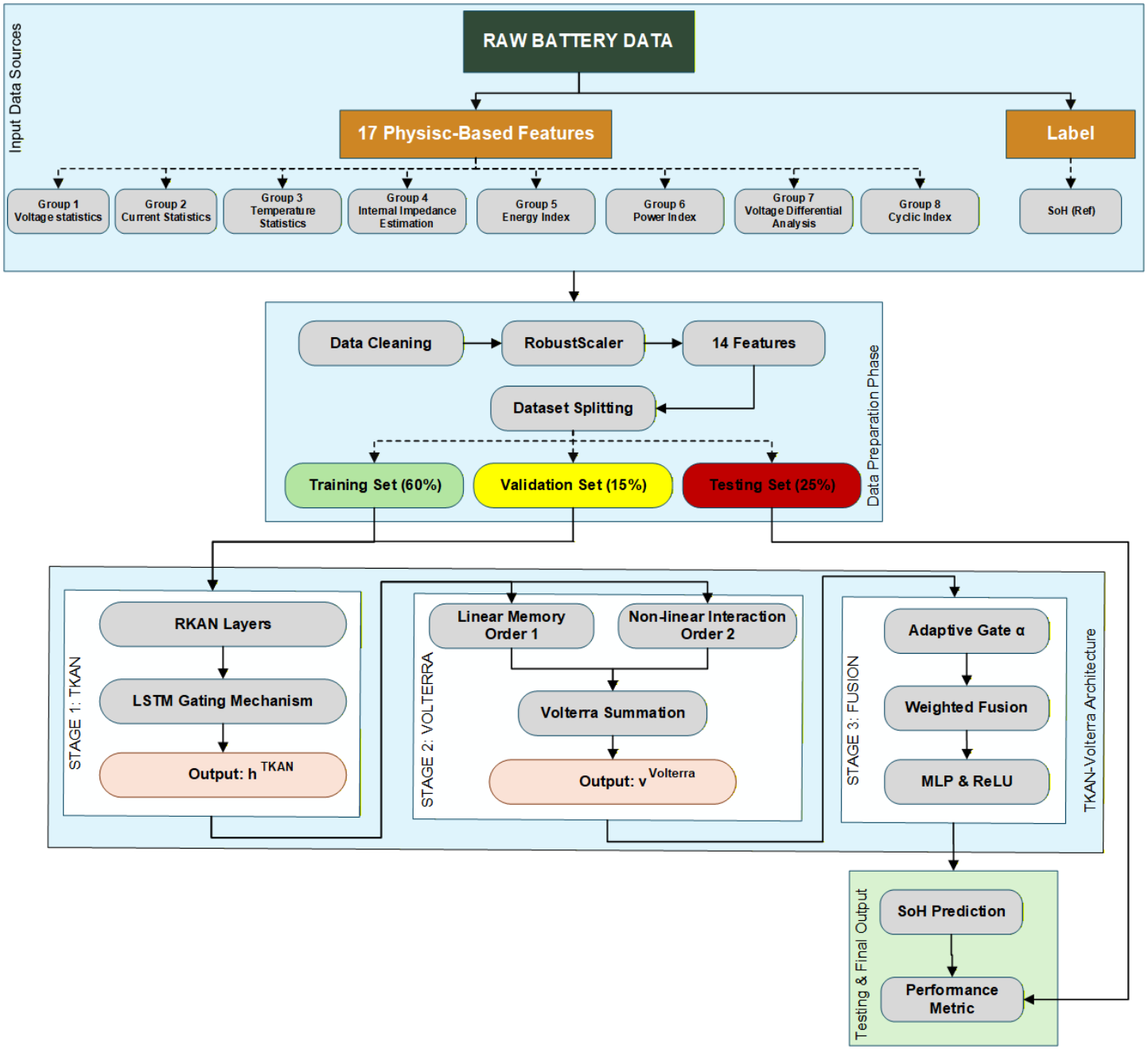
$R_{eol} \approx 2 \times R_{initial}$ for Li-ion

$$SoH_{combined} = 0.7 \times SoH_C + 0.3 \times SoH_P$$

Physical Constraints:

- Monotonic decrease: SoH cannot increase over cycles
- Bounded range: $0\% \leq SoH \leq 100\%$
- History-dependent: Requires long-term sequential data

TKAN-Volterra Architecture



Dataset: NASA Li-ion Battery (PCoE)

Battery	Temp (C)	Condition	Cycles	SoH Range
B05	32.8	Normal	158	0.64–0.93
B07	32.4	Normal	158	0.70–0.94
B18	31.1	Normal	122	0.67–0.91
B33	38.0	High	187	0.10–0.94
B34	39.2	High	187	0.63–0.91
B46	9.1	Cold	62	0.00–0.72
B47	8.5	Cold	62	0.00–0.70
B48	8.0	Cold	62	0.00–0.71

Data Configuration

Data Split: Train 60% — Validation 15% — Test 25% (Chronological, No Shuffle)

Performance Comparison

Model	Avg R^2	RMSE	MAE	MAPE (%)
TKAN-Volterra	**0.811**	0.0085	**0.0057**	**0.92**
PI-LSTM	0.786	**0.0081**	0.0058	0.93
Standard LSTM	0.649	0.0279	0.0114	1.92

Key Findings:

- ✓ TKAN-Volterra achieves highest $R^2 = 0.811$
- ✓ Wins on 4/8 batteries (50%)
- ✓ Positive R^2 on ALL 8 batteries

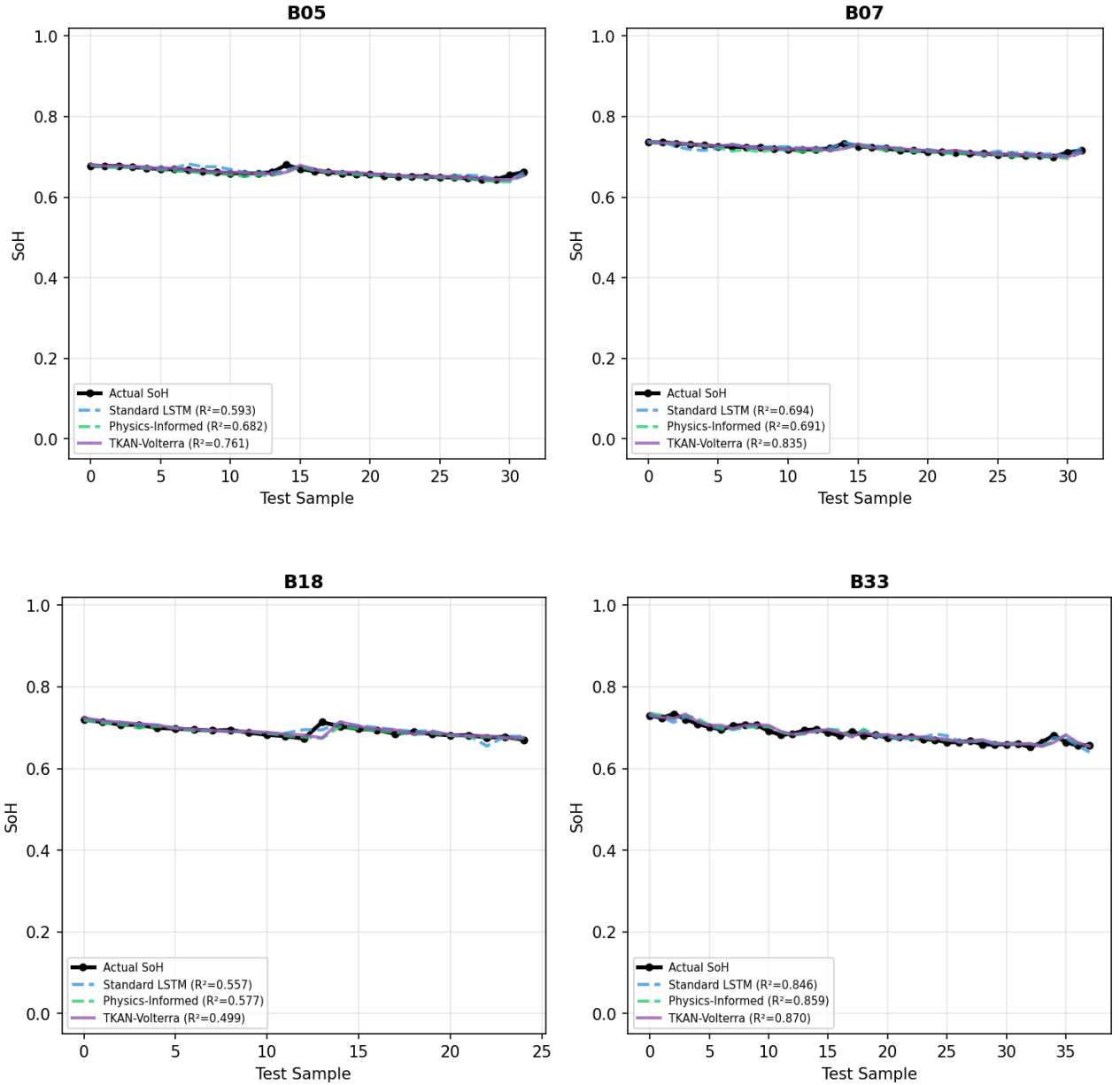
Per-Battery R^2 Comparison

Battery	Temp	TKAN-Volterra	PI-LSTM	Std LSTM
B05	32.8C	**0.761**	0.682	0.593
B07	32.4C	**0.835**	0.691	0.694
B18	31.1C	0.499	**0.577**	0.557
B33	38.0C	**0.870**	0.859	0.846
B34	39.2C	**0.536**	0.490	0.532
B46	9.1C	0.995	0.998	**0.999**
B47	8.5C	0.994	0.995	**0.997**
B48	8.0C	**0.999**	**0.999**	–

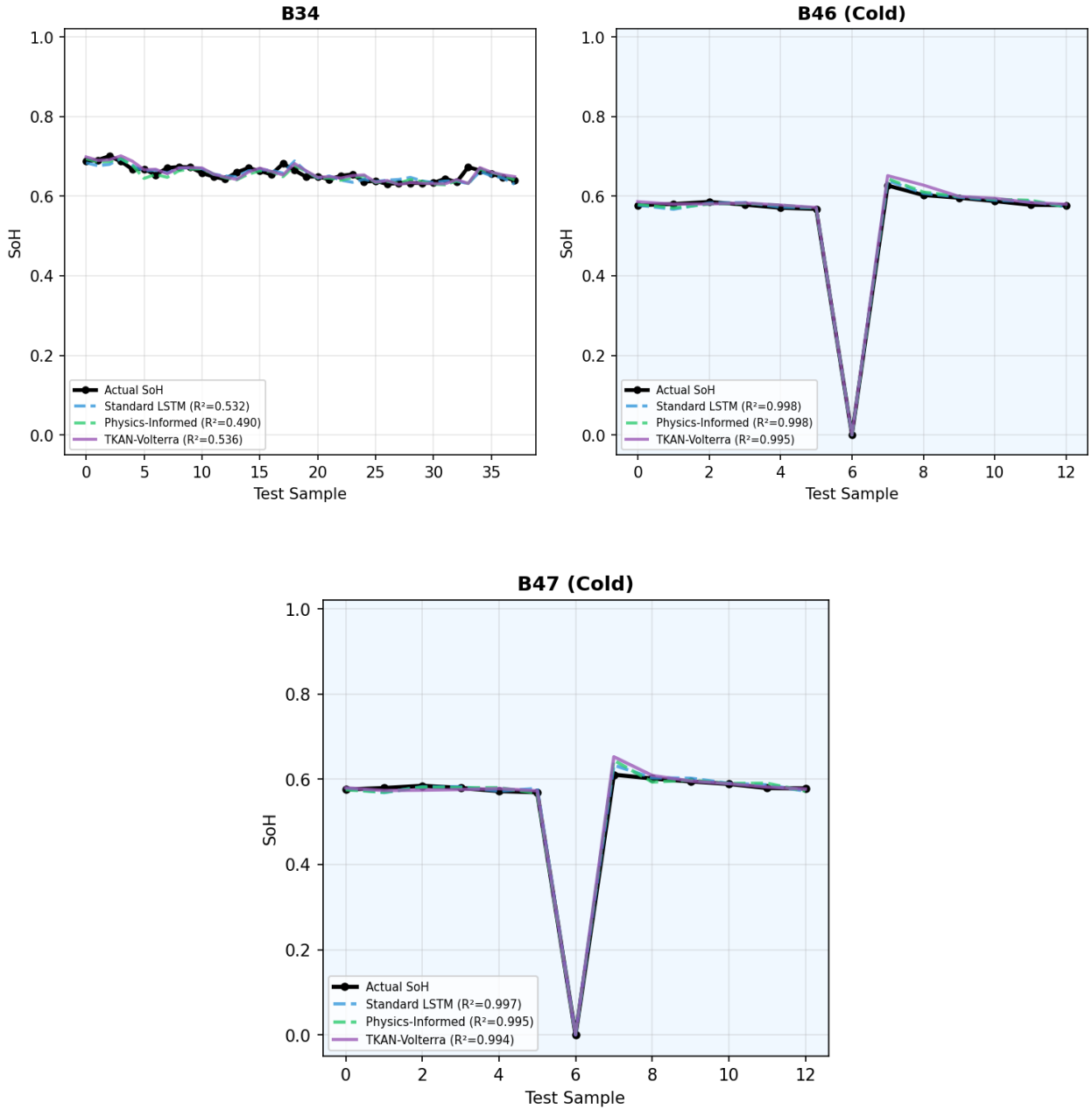
Summary by Temperature:

Normal (31–33C): TKAN avg $R^2 = 0.700$ — High (38–39C): TKAN avg $R^2 = 0.703$ — Cold (8–9C): All models $R^2 > 0.99$

SoH prediction comparison on B05-B33



SoH prediction comparison on B34, B46, B47



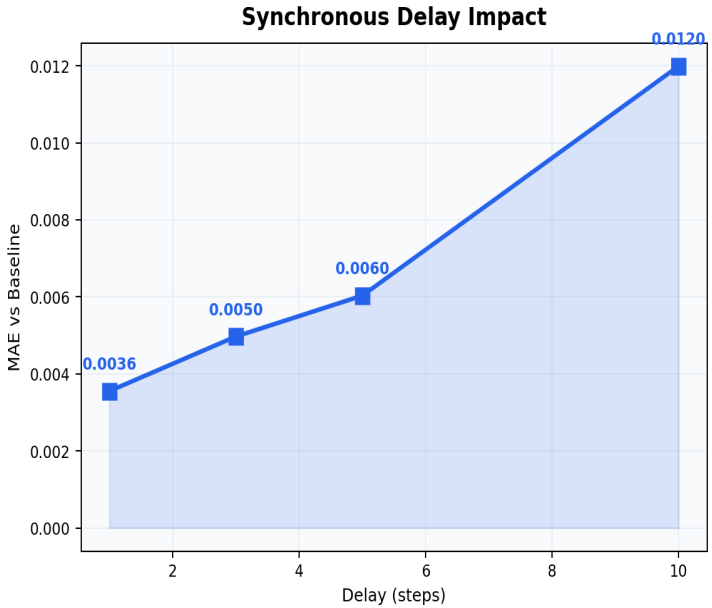
Robustness: Data Acquisition Delays

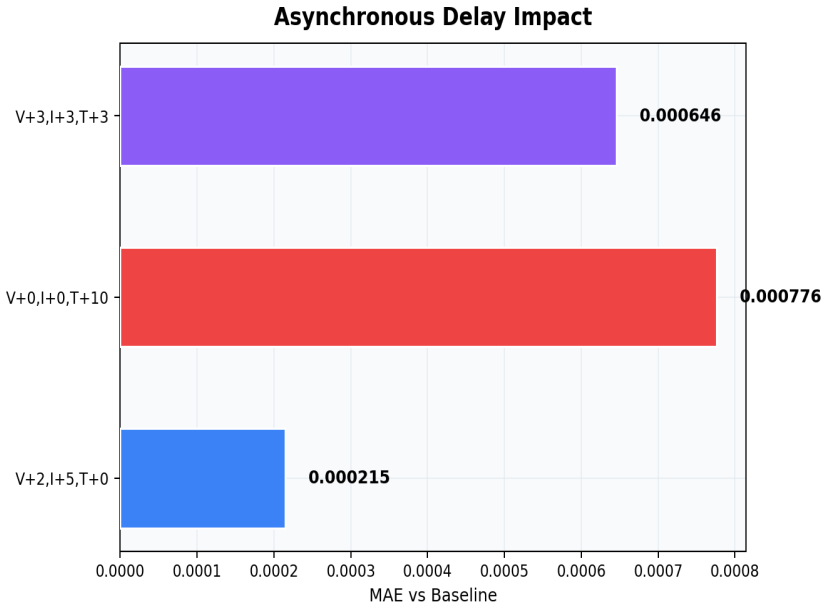
The TKAN-Volterra model shows strong stability:

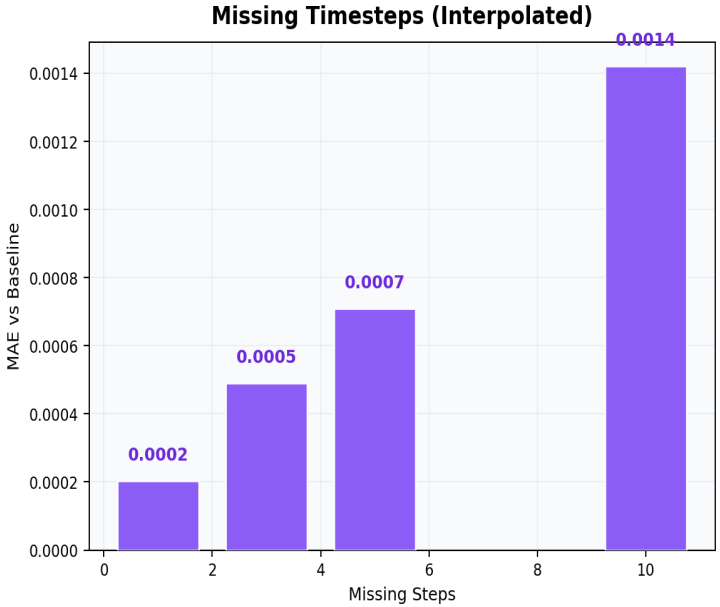
- **Synchronous delay**: When all signals are delayed together, MAE grows slowly from 0.0036 (1-step) to 0.012 (10-step). The model degrades gradually, not suddenly.
- **Asynchronous delay**: When voltage, current, and temperature arrive at different times, MAE stays below 0.001.
- **Missing timesteps**: The model handles up to 10 missing samples with MAE of only 0.0014, with no errors or crashes.

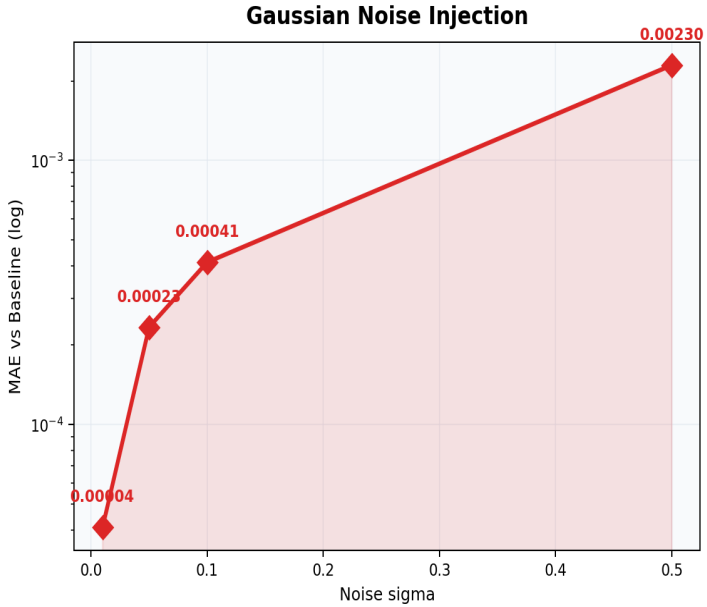
These results confirm that TKAN-Volterra works well in real battery systems where perfect real-time data is not always available.

Robustness: Data Acquisition Delays









Conclusion & Future Work

Summary

- Novel TKAN-Volterra hybrid architecture for SoH prediction
- Physics-informed: 17 features (14 used), composite SoH (70% C + 30% P)
- Best performance: $R^2 = 0.811$, wins 4/8 batteries
- Positive R^2 on all 8 batteries (unlike Standard LSTM)
- Robust under data delays up to 10 sampling intervals

Future Directions

- Validation on different battery chemistries
- Bayesian uncertainty quantification
- Attention mechanisms & Federated learning
- Integration with Remaining Useful Life (RUL) prediction

THANK YOU

dsidorov@isem.irk.ru
www.compvis.ru

This work was funded within State Assignment of MSHE of RF, No. FZZS-2024-0003