Published version: https://doi.org/10.1109/SmartGridComm51999.2021.9632332

A World Model Based Reinforcement Learning Architecture for Autonomous Power System Control

Magnus Tarle, Mårten Björkman, Mats Larsson, Lars Nordström, Gunnar Ingeström

***Abstract*—Renewable generation is leading to rapidly shifting power flows and it is anticipated that traditional power system control may soon be inadequate to cope with these fluctuations. Traditional control include human-in-the-loop-control schemes while more autonomous control methods can be categorized into Wide-Area Monitoring, Protection and Control systems (WAMPAC). Within this latter group of more advanced systems, reinforcement learning (RL) is a potential candidate to facilitate power system control facing these new challenges.**

In this paper we demonstrate how a model based reinforcement learning (MBRL) algorithm, which learns and uses an internal model of the world, can be used for autonomous power system control. The proposed RL agent, called the World Model for Autonomous Power System Control (WMAP), includes a safety shield to minimize risk of poor decisions at high uncertainty. The shield can be configured to permit WMAP to take actions with the condition that WMAP asks for guidance, e.g. from a human operator, when in doubt. As an alternative, WMAP could be run in full decision support mode which would require the operator to take all the active decisions.

A case study is performed on a IEEE 14-bus system where WMAP is setup to control setpoints of two FACTS devices to emulate grid stability improvements. Results show that improved grid stability is achieved using WMAP while staying within voltage limits. Furthermore, a disastrous situation is avoided when WMAP asks for help in a test scenario event that it had not been trained for.

***Index Terms*—Decision support systems, Flexible AC Transmission Systems (FACTS), learning, power system control, smart grids.**

I. Introduction

A. Background and motivation

As distributed renewables are replacing traditional centralized power generation, the dynamics of the electrical power grid is changing dramatically. Elevated concerns from this transformation are volatile power flows throughout transmission corridors and poor voltage quality, which increases risk of system splits and even network collapse. As the reduction of inertia and short circuit power continues, shorter time constants and substantial fluctuations are becoming increasingly more demanding. This shift in dynamics are foreseen to require more coordinated, faster and accurate alternatives to the conventional, error-prone and aging human-in-the-loop grid control.

Historically, Flexible AC Transmission Systems (FACTS) devices have successfully been applied to support the grid and increase power loadability through transmission corridors. Engineering a wide area control scheme using several FACTS devices can mitigate the stress on transmission lines by optimizing control reference setpoints [1]. Such wide area control schemes are based on heuristics that are unfortunately tedious to engineer, may need adjustment as the power system evolves and are only applicable to simple network topologies [1]. For more complex network topologies, model based approaches using optimal power flow (OPF) could potentially be used akin to the proposal in [2]. However, such approaches suffer from insufficient update rates and reliability issues due to the dependency on data from Supervisory Control And Data Acquisition (SCADA), the Energy Management System (EMS) infrastructure and OPF convergence.

As an alternative control architecture for complex network topologies, we investigate RL for managing grid control. RL holds the promise to self-learn a coordinated control scheme and continuously update its learning. Before any deployment of RL into the real world however, there are several important aspects that needs to be addressed. One aspect being the conventional naive RL trial and error approach that can be considered orthogonal to safety. Furthermore, many RL algorithms have learning issues since they have to revisit many states an unreasonable amount of times due to poor sample efficiency. In addition, many RL agents tend to overfit to their training environment and generalize poorly to other environments.

In this paper, our goal is to demonstrate an application of an RL based algorithm to power system control to mitigate risk of the above mentioned voltage quality challenges while also confronting RL issues with safety, generalization and sample efficiency. Specifically, we adapt a world model architecture to the power system domain while emphasizing safety.

B. Related Work

The idea of RL as a data driven and learning system to solve power systems control challenges has a long history [3]. Since the advent of deep RL with Deep Q Networks (DQN) [4], power system control problems with continuous state spaces and with discrete as well as continuous actions have been studied. As an application example and in terms of voltage quality, a hybrid RL-optimization control scheme involving converter setpoints and RL controlled switching of

This research has received funding from the Swedish Foundation for Strategic Research under grant agreement ID19-0058 and from Hitachi ABB Powergrids. This work was partially supported by the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

Magnus Tarle is at the division Robotics, Perception and Learning (RPL) at Royal Institute of Technology (KTH), Stockholm, Sweden. Magnus is also at Hitachi ABB Power Grids, Sweden. Mårten Björkman is at the division RPL at KTH, Stockholm, Sweden. Mats Larsson is at Hitachi ABB Power Grids, Switzerland. Lars Nordström is at the division Electric Power and Energy Systems at KTH, Stockholm, Sweden. Gunnar Ingeström is at Hitachi ABB Power Grids, Sweden. Correspondence to Magnus Tarle: <tarle@kth.se>

capacitor banks was investigated in [5]. Furthermore, diverting power using RL was considered in [6] by making discrete topological changes to the grid. In terms of decision support systems, an RL framework provided human operators with suggested voltage setpoints in [7]. Safety was emphasized in [8] where a constrained version of the Soft-Actor-Critic (SAC) [9] algorithm was employed. Finally and although not specifically RL, anomaly detection was studied by [10] in which a Variational Auto-Encoder (VAE) [11] was applied.

In relation to the above, a key aspect that has not yet been explored in RL for power systems is the adoption of world models.

C. *Scope of the paper*

This paper proposes to adapt the world model architecture from [12] to fit a power system application while also adding crucial safety measures.

The main contributions of this paper are three-fold:

- Proposing WMAP, a world model based RL agent architecture for the power system domain. This world model approach is hypothesized to facilitate agent learning and improve generalization.
- By taking advantage of the world model, applying a scheme of safety measures used during operation that allows the agent to ask for assistance whenever uncertain.
- Validating WMAP on the IEEE 14-bus system for an autonomous power converter setpoint control scenario with the objective of improving voltage quality and stability.

The remaining parts of the paper describes WMAP, the case study undertaken on the IEEE 14-bus system and finally concludes with a discussion.

II. A World Model Approach

This section describes the World Model for Autonomous Power System Control (WMAP) and its implementation.

A. *General*

Compared to model free RL (MFRL), MBRL is generally portrayed as having superior sample efficiency and providing the opportunity for the RL agent to plan using its model. As a stepping stone in MBRL performance, the world model architecture which we adapt in this paper, surpassed previous MFRL scores on two benchmark environments at the time of publishing [12]. Using a world model, high dimensional observations can be encoded into a lower dimensional latent space for accelerated learning. Furthermore, the agent can take advantage of a compressed internal dynamics model of the environment to guide its actions.

B. *Safety Benefits*

Before diving into the architecture details, we highlight that WMAP makes use of two safe RL aspects, *Providing Initial Knowledge* and *Ask for Help* as categorized by [13]. We argue that RL agents applying world models have several inherent benefits in relation to these two safe RL categories.

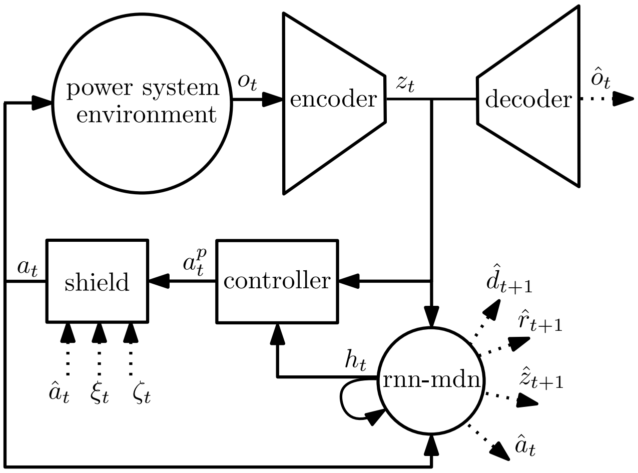

Fig. 1. Illustration of the World Model for Autonomous Power System Control.

First of all, we do not want to cause unnecessary disturbance and possibly hazardous events during agent learning. Contingency scenarios in power systems also occur too seldom and may progress too fast for any agent to learn adequately online. For this reason, we provide initial knowledge by pre-training the RL agent in a simulation model. Contributing to this initial knowledge is that a world model enables accelerated training of the agent due to the lower dimensional latent observation space. Having access to the internal world model also provides the agent with an additional benefit of experiencing data distributions outside the original distribution by increasing the world model state transition uncertainty as in [12].

In regard to ask for help, an internal world model allows us to obtain uncertainty metrics to indicate whether the gap between the real world and the internal model is too wide. We make use of these metrics to ask for help whenever the uncertainty exceeds a certain threshold. To avoid extrapolation errors common to neural networks, the world model is also trained to output a distribution of past actions given an observation, akin to the approach in [14]. In this sense, the agent could be restricted to stay close to previous trusted actions and ask for guidance if the agent wants to attempt a completely new action.

C. *Architectural Overview*

An illustration of WMAP is seen in Fig. 1. WMAP contains five major parts: i) a power system environment simulator, ii) the feature extraction modeled by a VAE encoder and decoder, iii) the environment dynamics realized by a recurrent neural network mixture density network (RNN-MDN) [15], iv) the controller that proposes an action and finally, v) a safety shield.

The WMAP models are listed in Table I and are implemented by three neural networks parameterized by parameters $\boldsymbol{\phi}$, $\boldsymbol{\theta}$ and $\boldsymbol{\psi}$ together with safety shield parameters $\boldsymbol{\upsilon}$.

D. *Main Components*

The power system environment illustrated in Fig. 1, is a simulation environment during training to provide the agent

TABLE I
WORLD MODEL FOR AUTONOMOUS POWER SYSTEM CONTROL COMPONENTS

Component	Model	Definition
VAE	Representation model	$q_{\phi}(\boldsymbol{z}_t\|\boldsymbol{o}_t)$
	Reconstruction model	$p_{\phi}(\hat{\boldsymbol{o}}_t\|\boldsymbol{z}_t)$
RNN-MDN	Dynamics model	$p_{\theta}(\boldsymbol{h}_{t+1}\|\boldsymbol{z}_t, \boldsymbol{a}_t, \boldsymbol{h}_t)$
	Transition model	$p_{\theta}(\hat{\boldsymbol{z}}_{t+1}\|\boldsymbol{z}_t, \boldsymbol{a}_t, \boldsymbol{h}_t)$
	Reward model	$p_{\theta}(\hat{r}_{t+1}\|\boldsymbol{z}_t, \boldsymbol{a}_t, \boldsymbol{h}_t)$
	Terminal state model	$p_{\theta}(\hat{d}_{t+1}\|\boldsymbol{z}_t, \boldsymbol{a}_t, \boldsymbol{h}_t)$
	Past action model	$p_{\theta}(\hat{\boldsymbol{a}}_t\|\boldsymbol{z}_t)$
Controller	Action proposal model	$p_{\psi}(\boldsymbol{a}_t^p\|\boldsymbol{z}_t, \boldsymbol{h}_t)$
Shield	Observation uncertainty	$p(\xi_t\|\boldsymbol{o}_t, \hat{\boldsymbol{o}}_t)$
	Dynamics uncertainty	$p(\zeta_t\|\boldsymbol{z}_t, \hat{\boldsymbol{z}}_t)$
	Action shield	$p_{\upsilon}(\boldsymbol{a}_t\|\xi_t, \zeta_t, \hat{\boldsymbol{a}}_t, \boldsymbol{a}_t^p)$

with initial knowledge. In deployment, this simulation would be substituted with the real world environment.

Given a power system state $\boldsymbol{s}_t \in \mathcal{S}$ at time t, the VAE encoder encodes an observation $\boldsymbol{o}_t \in \mathcal{O}$ into a latent state $\boldsymbol{\mu}_t + \boldsymbol{\epsilon}\boldsymbol{\sigma}_t = \boldsymbol{z}_t \in \mathcal{Z}$, where $\boldsymbol{\epsilon}$ is a Gaussian distributed sample.

The downstream RNN-MDN receives the compressed representation $\boldsymbol{z}_t$ together with an action $\boldsymbol{a}_t \in \mathcal{A}$ at each time step t. Given a series of such representations and actions, the RNN-MDN learns a model of the dynamics encapsulated in its parameters and hidden state $\boldsymbol{h}_t$, and learns the past action distribution $\hat{\boldsymbol{a}}_t$. Specifically, the RNN-MDN predicts:

- The next latent state distribution $\hat{\boldsymbol{z}}_{t+1}$.
- The next terminal state condition $\hat{d}_{t+1}$.
- The next immediate reward $\hat{r}_{t+1}$.
- The past action distribution $\hat{\boldsymbol{a}}_t$.

The controller is employed as a multilayer perceptron network and is trained with the Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [16] similar to [12]. As input, the controller receives the latent state representation $\boldsymbol{z}_t$ of the VAE as well as the hidden state $\boldsymbol{h}_t$ of the RNN-MDN to be made aware of the dynamics. The controller output is a proposed action $\boldsymbol{a}_t^p \in \mathcal{A}$.

E. Safety Shield

The final component, the shield, outputs an action $\boldsymbol{a}_t$ provided the input proposed action $\boldsymbol{a}_t^p$, the past action distribution $\hat{\boldsymbol{a}}_t$ and uncertainty metrics ξ_t and ζ_t.

Utilizing the Euclidean distance of the reconstruction error, the VAE provides the metric ξ_t to the shield of how well known the current observation is. Furthermore, the mixture density network part in the RNN-MDN provides the metric ζ_t to the shield of how well known the current dynamics are. The RNN-MDN supplied model of the past action distribution $\hat{\boldsymbol{a}}_t$, enables the option to restrict the agent to stay close to more trusted previous actions.

F. Training

In terms of WMAP training, the proposed training scheme is summarized in Algorithm 1. We train each WMAP component at a time, starting with the VAE followed by the RNN-MDN and finally ending with the controller. To accelerate training, the controller is first trained in WMAP's world model and thereafter inside the simulation model. To fine tune WMAP towards the controller's most likely actions, we repeat the training of the WMAP components at least once.

Algorithm 1: Training procedure

```
Initialize simulation environment with training scenario
  and random starting state;
Run simulation environment with random actions and
  add tuple of $(\boldsymbol{o}_t, \boldsymbol{a}_t, \boldsymbol{o}_{t+1}, r_{t+1}, d_{t+1})$ to a dataset $\mathcal{D}$;
Initialize VAE model parameters $\boldsymbol{\phi}$, RNN-MDN
  parameters $\boldsymbol{\theta}$ and controller parameters $\boldsymbol{\psi}$;
for number of loops do training loop
    Train VAE on dataset $\mathcal{D}$ and add VAE encoder
      output $\boldsymbol{\mu}_t$ and $\boldsymbol{\sigma}_t$ to dataset $\mathcal{D}$;
    Train RNN-MDN on the dataset $\mathcal{D}$;
    Train controller inside WMAP's world model;
    Train controller in simulation environment and add
      tuple data to dataset $\mathcal{D}$;
end
```

G. Testing and Deployment

The testing and deployment is summarized in Algorithm 2. At each time step, WMAP receives an observation and begins making use of its shield. WMAP first evaluates if the observation is within its training distribution. If the observation is considered known, the controller proposes an action with a degree of added noise. Any proposed action is also assessed and if WMAP concludes that it has been sufficiently trained for that particular observation and action, the action is taken.

In the case of either the observation or proposed action being considered to be outside the training distribution, WMAP instead asks for expert guidance. The thresholds that would determine when to trust an observation or action are parameterized by the shield. Finally, to ensure incremental learning, the interactions with the environment are stored in the dataset that is used for later retraining.

III. CASE STUDY

This section describes the case study undertaken on the IEEE 14-bus system to evaluate WMAP.

A. Evaluation

WMAP was assessed against three other main methods in the case study:

- Fixed control reference setpoints referred to as *Constant*.
- Emulation of an "expert" decision maker, implemented as a Sobol and Gaussian Process algorithm using the Python package *Ax*. The algorithm, referred to as *Expert 5*, had full access to the environment model and the future state given an action. Two variants of this expert were added with the restriction to only take new actions every sixth

Algorithm 2: Test and deployment procedure

```
Initialize simulation environment with test scenario or
  real power system interface;
Initialize WMAP model and its shield parameters $\boldsymbol{\upsilon}$;
for each time step $t$ do
    Encode observation $\boldsymbol{o}_t$ into latent state $\boldsymbol{z}_t$;
    if uncertainty $\xi_t$ or $\zeta_t$ above threshold $\upsilon_s$ then
        ask expert and take expert action;
    end
    else
        Get controller proposed action and add noise;
        if action outside action distribution threshold
          $\upsilon_a$, action is considered extrapolated then
            Ask expert to verify action and reward;
            if reward is below reward threshold $\upsilon_r$ then
                Save verified action data to dataset $\mathcal{D}$;
                Ask expert and take expert action;
            end
            else
                Take action;
            end
        end
        else
            Take action;
        end
    end
    Get new hidden state $\boldsymbol{h}_{t+1}$ and predict $\hat{\boldsymbol{z}}_{t+1}$;
    Save tuple $(\boldsymbol{o}_t, \boldsymbol{a}_t, \boldsymbol{o}_{t+1}, r_{t+1}, d_{t+1})$ to dataset $\mathcal{D}$;
end
```

or third time step, referred to as *Expert 30* and *Expert 15* respectively.

- A standard MFRL algorithm, implemented as a Twin Delayed Deep Deterministic Policy Gradient (TD3) [17] using the Python package *Stable Baselines 3*.

We also assessed different variants of the WMAP components to study their effect. First of all, four different VAE configurations were studied, referred to as:

- *WMAP-M* applying a standard VAE [11] with the modification of adopting a scheduler to vary the balance of the VAE loss terms as in [18]. Note that all VAE variants applied such a scheduler and used an encoding $\boldsymbol{z} \in \mathbb{R}^{16}$.
- *WMAP-C* having an auxiliary classification loss, enforcing latent spaces that contain crucial information about the network state.
- *WMAP-G* employing the graph encoder part from [19] to take advantage of the additional information related to the electrical system graph structure.
- *WMAP-S* adding an auxiliary self-supervised loss [20] to impose conditions on the VAE latent space encodings.

Secondly, two variants of the RNN-MDN were set up in which the *WMAP-B* applied a bistable recurrent cell [21] to gain memory capacity, while all other WMAP configurations applied a gated recurrent unit (GRU) [22]. Thirdly, two controller configurations were evaluated, the standard CMA-ES trained feed forward network controller and a more sophisticated TD3 controller. Finally, WMAP configurations were also evaluated with and without the shield. For example, *WMAP-C* is without shield, *WMAP-C-E* denotes shield without caring about the past action distribution, while *WMAP-C-EA* employs full use of the shield.

For the evaluation, the test procedure in Algorithm 2 was undertaken and the average total cumulative reward was compared at the end of the test scenario episode for five different seeds. All agents were allowed to interact with the training scenario for 132 thousand time steps, corresponding to 458 days of operation, before being evaluated in the test scenario.

B. Electrical System

An environment model was developed utilizing *Pandapower* [23]. As Pandapower is a power flow calculation software, a limitation is that low level controllers and power system dynamics are not represented. For the case study in question, the assumption is that fast power system dynamics are kept stable through low level controllers not considered in the employed power flow model. This is motivated by having sufficient time separation between the change of reference setpoints and faster acting low level controllers.

The original IEEE 14-bus system in Fig. 2 was modified by removing all synchronous condensers as a means to increase the sensitivity of the voltages to the operating point. Furthermore, the generator at node 2 was converted into a PQ node. Assumptions on the nominal voltages and on the line and transformer ratings were made. In addition, a simplified first order system thermal model was applied to each line and transformer. Moreover, a line between bus 6 and 13 was added and the line between bus 2 and 3 was modified to provoke overload situations.

Separate training and test scenarios were used. The load profiles were synthetically modeled using *Enlopy*, which is a library to generate, analyze and process energy time series. Generator profiles were obtained from publicly available aggregated wind generation profiles. Via the profiles, each scenario contained 8000 specific time steps with a time step interval of five minutes. The test scenario included a large disturbance in which the generator at node 1 initially drops 10 percent in voltage magnitude, followed by a temporary disconnection of the line between buses 3 and 4.

Based on power flow simulations and the training scenario profiles, two FACTS devices were modeled and added to the system. The devices consist of a Static Synchronous Compensator (STATCOM), referred to as SVS in Fig. 2 and a Thyristor Controlled Series Capacitor (TCSC), referred to as SC in Fig. 2. Actions allowed by the agent were to control the reference setpoints of these two devices, i.e. a voltage reference v_{ref} for the shunt device and a current reference i_{ref} for the series device.

C. Observation space

The observation space of the electrical system is defined as a graph $\mathcal{G} = (\mathcal{H}, \mathcal{E}, \boldsymbol{g})$, where $\mathcal{H}$ is a set of N nodes, $\mathcal{E}$ is a set of M directional edges and $\boldsymbol{g}$ is a vector containing global features. In our case, nodes are representing power system buses, edges are representing lines and transformers while global features represent abnormal system conditions. A remark is that the choice of defining directional edges is to enable information about edge source and destination features.

Let each node $n \in \mathcal{H}$ be represented by a vector $\boldsymbol{h}_n \in \mathbb{R}^4$ with the following properties:

$$\boldsymbol{h}_n = (v_n, \delta_n, p_n, q_n) \tag{1}$$

where the voltage magnitude is denoted v_n and δ_n is the voltage phase information for each node n. The sum of the injected and outgoing active and reactive powers, for each node n, are represented by p_n and q_n respectively.

Furthermore, let each directional edge $e \in \mathcal{E}$ between a source node $k \in \mathcal{H}$ and a destination node $l \in \mathcal{H}$ be represented by a vector $\boldsymbol{e}_{kl} \in \mathbb{R}^5$ with the following properties:

$$\boldsymbol{e}_{kl} = \left(p^s_{kl}, q^s_{kl}, p^d_{kl}, q^d_{kl}, T_{kl}\right) \tag{2}$$

where the active and reactive power transmitted across the edge, measured at the source node k, are represented by p^s_{kl} and q^s_{kl} for each edge e. The active and reactive power transmitted across the edge, measured at the destination node l, are represented by p^d_{kl} and q^d_{kl}. The estimated temperature of the edge is denoted by T_{kl}.

Finally, let global system properties be represented by the binary vector $\boldsymbol{g} = (g_1, g_2, ..., g_5) \in \{0,1\}^5$ where g_1 and g_2 encode undervoltage, g_3 and g_4 overvoltage and g_5 overload conditions. Note that only the TD3 RL agent had direct access to $\boldsymbol{g}$ while *WMAP-C* was trained to correctly classify this global feature vector from its latent space $\boldsymbol{z}_t$.

D. Objective and Reward Shaping

We want to reinforce actions that increase voltage stiffness by rewarding the agent with a metric r^q_t on qv-sensitivity:

$$r^q_t = \frac{1}{N_q} \sum_{j \in \mathcal{H}_q} \sqrt{\left(\frac{q^j_{t-1} - q^j_t}{v^j_{t-1} - v^j_t}\right)^2} = \frac{1}{N_q} \sum_{j \in \mathcal{H}_q} \eta^j_t \tag{3}$$

where the injected reactive power at node j at time step t is denoted q^j_t and v^j_t is the voltage magnitude. To avoid including stiff voltage sources, the set $\mathcal{H}_q \subseteq \mathcal{H}$ is the set of N_q nodes which have loads connected but where η^j_t in (3) is less than a predefined parameter β_{q_2}.

To further promote voltage stiffness, the agent is rewarded by r^v_t when staying close to nominal node voltages v^j_{nom}:

$$r^v_t = -\frac{1}{N_v} \sum_{j \in \mathcal{H}_v} \left(v^j_{nom} - v^j_t\right)^2 \tag{4}$$

where $\mathcal{H}_v \subseteq \mathcal{H}$ is the set of N_v nodes which have loads connected but lack a connected stiff or controlled voltage source.

Moreover, keeping line and transformer temperatures below maximum allowed temperatures is also rewarded with r^o_t:

$$r^o_t = -\frac{1}{M_o} \sum_{j \in \mathcal{E}_o} \sigma\left(T^j_t - \beta^j_{o_2}\right) \tag{5}$$

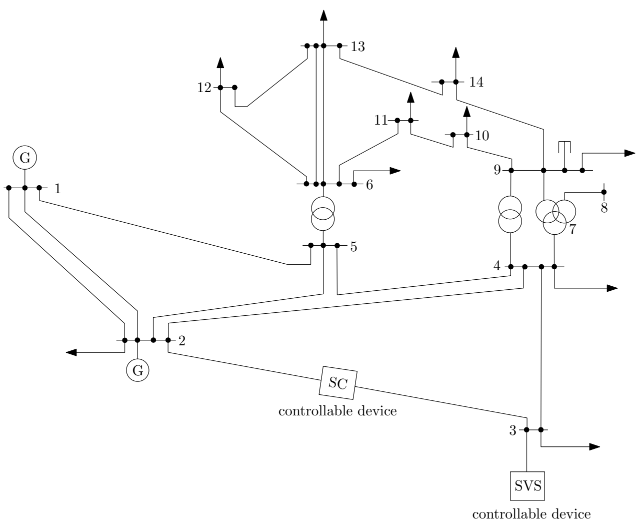

Fig. 2. Case study performed on a modified IEEE 14-bus system with two added FACTS devices.

where σ is the sigmoid function and T^j_t is the temperature at time t at edge j. The set $\mathcal{E}_o \subset \mathcal{E}$ is the set of M_o lines and transformers within the system. The maximum permissible temperature is characterized by the parameter $\beta^j_{o_2}$.

To encourage actions that make power flow solutions converge, the agent is given a constant reward r^c_t at each time step. Non-convergent solutions however terminate the training or testing and are penalized by a negative constant reward r^n_t.

The total scalar reward r_t provided to the agent is given as:

$$r_t = \beta_o r^o_t + \beta_v r^v_t + \beta_q r^q_t + \beta_c r^c_t + \beta_n r^n_t \tag{6}$$

where β_o, β_v, β_q, β_c and β_n are reward function weight parameters obtained via empirical observations of the rewards. The weights were chosen so that each reward component would have a comparable influence on the total reward.

E. Results

Results from the case study are illustrated in Fig. 3. The best performing WMAP agent with the highest total cumulative mean reward was evaluated to be the agent with the auxiliary classification loss and full shield activated, i.e. the *WMAP-C-EA*. With the aid of the shield, it overcame the test scenario disturbance event and surpassed the reward of the other WMAP variants. Only *Expert 5*, which had full access to the environment achieved a higher mean reward. Prior to the disturbance event, *WMAP-C-EA* also showed the highest reward with the exception of *Expert 15* and *Expert 5*. Perhaps surprisingly, because of the graph nature of the electrical system, the *WMAP-G* utilizing a graph encoder performed the worst out of the WMAP configurations.

We underline that without the shield, Pandapower failed to converge during the disturbance event, indicating a voltage collapse in the power system. The exception was the *WMAP-C* and two experts. They succeeded to complete the scenario using at least one of the seeds, which explains the large standard deviation in the results. As a remark, the action part

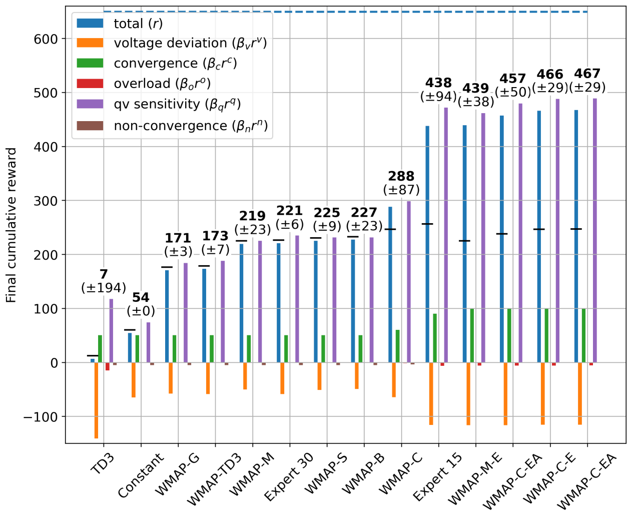

Fig. 3. Results on the case study test scenario. The values within the bar plot refer to the mean value of the final total cumulative reward while the standard deviation is given inside the parenthesis. The dotted horizontal blue line represents the *Expert 5* final total cumulative mean reward. The black dashes in front of, or on top of the vertical total reward bars, represent the total cumulative reward prior to the large disturbance event.

of the shield caused a larger performance difference between *WMAP-M-E* and *WMAP-M-EA* when compared to *WMAP-C-E* and *WMAP-C-EA*. As we used the same shield parameters for all agents, we expect that variations in exploration, model parameter weights and performance between agents may cause such dissimilarities.

Another observation is that TD3 was learning but appeared to get stuck in local optima. In comparison to *WMAP-TD3*, the latter had a higher mean reward and was more inclined towards the same local optima across the seeds.

To further highlight the difference between having the shield activated and de-activated during the disturbance event, an example is visualized in Fig. 4. As the initial voltage drop occurred, the shield uncertainty metric increased to the point that WMAP asked for guidance in the case of the shield being activated. The emulated expert decision maker, provided aid by suggesting actions for both the voltage and current references to ensure that the environment model still converged. The initial advice was to push the current reference to a maximum with an increase in the voltage reference. We interpret this as a plausible action considering the initial voltage drop and the ensuing impedance increase in the system.

To illustrate the differences in reward, the cumulative mean reward over time is shown for a selection of agents in Fig. 5. We observe that *WMAP-C-EA* tended to favour overload and voltage deviation while *Expert 5* was inclined to favor more the qv-sensitivity reward. We hypothesize that the ability to predict the future state is of substantial importance for obtaining a high qv-sensitivity reward and that it is the main driver behind the difference. Additionally, *WMAP-C-EA* agent exceeded the reward of *Expert 30* with approximately one standard deviation during the test scenario. We further note that keeping the setpoints constant resulted in poor fulfillment of the objective. As a final remark, the variance of the standard TD3 agent was excessively larger compared to the other agents.

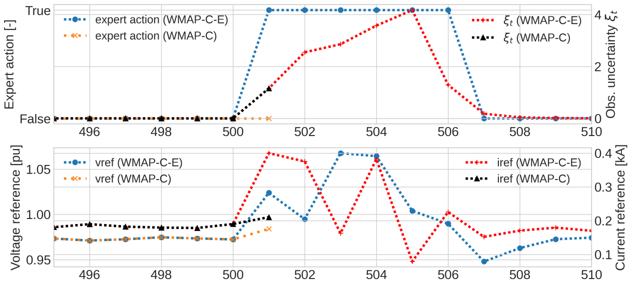

Fig. 4. Example of observation uncertainty metric ξ_t and the actions during a large disturbance event. *WMAP-C*, which does not have any shield active, fails early with a non-convergent power flow. *WMAP-C-E*, which has the shield active, asks for advice and overcomes the event it had not been trained for.

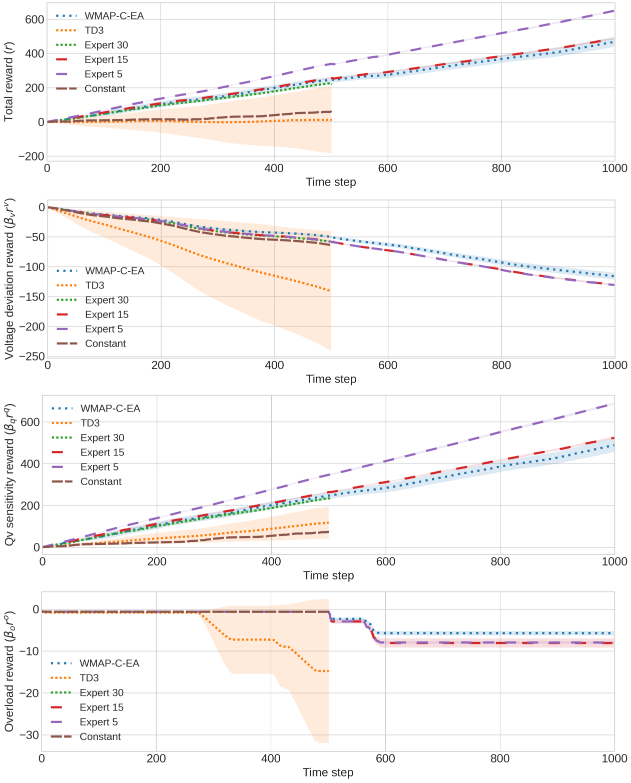

Fig. 5. Cumulative mean reward over time for a selection of agents where shaded areas represent a standard deviation of one. Besides the cumulative total reward r, the voltage deviation $\beta_v r^v$, the qv-sensitivity $\beta_q r^q$ and the overload $\beta_o r^o$ parts of the total reward are illustrated. The convergence $\beta_c r^c$ and non-convergence $\beta_n r^n$ portions of the total reward are not shown as these rewards are based on simple constants.

IV. Discussion

Firstly, WMAP is shown to provide significant improvements of voltage stiffness and ensures that voltage limits are met, specifically compared to setpoints that are not updated regularly as was represented by the constant actions in Fig. 3. *WMAP-C* even outperforms an expert decision maker that does not update the setpoints at short enough time intervals. The added auxiliary classification loss of the VAE in *WMAP-C* is hypothesized to contribute to an improved latent space representation for the case study objective.

Secondly, including the safety shield in the world model based architecture resulted in a higher total reward. With some minor exceptions, the WMAP agents that did not apply the shield failed on the large disturbance event which they had not been trained on. As such, the shield acts as a safety layer in case that the agent has not been trained well enough for the environment it will operate in.

Thirdly, with other choices of input parameters, actions and reward function, the basic idea could be applicable also to other control problems in power systems. This will be investigated in future work.

Finally, compared to the TD3 out-of-the-box RL agent, WMAP generalizes better to the test environment in terms of reward given a limited amount of environment interactions. Despite having tried different hyperparameters, it should be noted however that a fair comparison can not be made without a more extensive hyperparameter search.

V. Conclusion

In this paper we presented WMAP, a world model based RL agent architecture, and its potential to solve a power system control problem. To adapt to the real world, the world model's capability of estimating uncertainty was used to add a vital layer of safety. Importantly, the architecture allows the agent to take actions while asking for help if uncertainty is too high.

The case study results on the IEEE 14-bus system showed that WMAP provided improved performance with increased voltage stiffness compared to common fixed control setpoints and non-frequent expert setpoint updates. It also outperformed a standard MFRL algorithm given a limited amount of training.

Compared to traditional methods and while not demonstrated in this paper, WMAP is a learning method expected to respond fast and continuously improve over time. Following assisted decisions, the response to those actions is added to the training dataset, such that uncertainty is reduced should similar operating conditions be encountered again. Thus, the need for expert support can be expected to decrease over time. Similar to other neural network based algorithms, the model does suffer from lack of explainability. Although due to the world model's modularized nature, individual components can be inspected and we believe that appropriate testing procedures will act as a counterweight to the need of explainability.

Future work is considered to further focus on safety, benchmarking against state of the art methods, continual learning and scalability.

References

[1] M. Larsson, C. Rehtanz, and D. Westermann, "Improvement of Cross-border Trading Capabilities through Wide-area Control of FACTS," in *Bulk Power Syst. Dyn. Control VI*, 2004.

[2] T. Zabaiou, L. A. Dessaint, and I. Kamwa, "A comparative study of VSC-OPF techniques for voltage security improvement and losses reduction," *IEEE Power Energy Soc. Gen. Meet.*, vol. 2014-Octob, no. October, 2014.

[3] M. Glavic, R. Fonteneau, and D. Ernst, "Reinforcement Learning for Electric Power System Decision and Control: Past Considerations and Perspectives," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6918–6927, 2017.

[4] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, S. Petersen, C. Beattie, A. Sadik, I. Antonoglou, H. King, D. Kumaran, D. Wierstra, S. Legg, and D. Hassabis, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[5] Q. Yang, G. Wang, A. Sadeghi, G. B. Giannakis, and J. Sun, "Two-Timescale Voltage Control in Distribution Grids Using Deep Reinforcement Learning," *IEEE Trans. Smart Grid*, vol. 11, no. 3, pp. 2313–2323, 2020.

[6] T. Lan, J. Duan, B. Zhang, D. Shi, Z. Wang, R. Diao, and X. Zhang, "AI-based autonomous line flow control via topology adjustment for maximizing time-series ATCs," *IEEE Power Energy Soc. Gen. Meet.*, vol. 2020-Augus, 2020.

[7] R. Diao, Z. Wang, D. Shi, Q. Chang, J. Duan, and X. Zhang, "Autonomous Voltage Control for Grid Operation Using Deep Reinforcement Learning," *IEEE Power Energy Soc. Gen. Meet.*, vol. 2019-Augus, 2019.

[8] W. Wang, N. Yu, Y. Gao, and J. Shi, "Safe Off-Policy Deep Reinforcement Learning Algorithm for Volt-VAR Control in Power Distribution Systems," *IEEE Trans. Smart Grid*, vol. 11, no. 4, pp. 3008–3018, 2020.

[9] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," *35th Int. Conf. Mach. Learn. ICML 2018*, vol. 5, pp. 2976–2989, 2018.

[10] R. Zheng and J. Gu, "Anomaly detection for power system forecasting under data corruption based on variational auto-encoder," *IET Conf. Publ.*, vol. 2019, no. CP764, pp. 1–6, 2019.

[11] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *2nd Int. Conf. Learn. Represent. ICLR 2014 - Conf. Track Proc.*, no. Ml, pp. 1–14, 2014.

[12] D. Ha and J. Schmidhuber, "Recurrent world models facilitate policy evolution," *Adv. Neural Inf. Process. Syst.*, vol. 2018-Decem, no. C, pp. 2450–2462, 2018.

[13] J. Garc and F. Fern, "A Comprehensive Survey on Safe Reinforcement Learning," *J. Mach. Learn. Res.*, vol. 16, pp. 1437–1480, 2015.

[14] S. Fujimoto, D. Meger, and D. Precup, "Off-policy deep reinforcement learning without exploration," *36th Int. Conf. Mach. Learn. ICML 2019*, vol. 2019-June, pp. 3599–3609, 2019.

[15] A. Graves, "Generating Sequences With Recurrent Neural Networks," *arXiv Prepr. arXiv1308.0850*, pp. 1–43, 2013.

[16] N. Hansen and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies." *Evol. Comput.*, vol. 9, no. 2, pp. 159–195, 2001.

[17] S. Fujimoto, H. Van Hoof, and D. Meger, "Addressing Function Approximation Error in Actor-Critic Methods," *35th Int. Conf. Mach. Learn. ICML 2018*, vol. 4, pp. 2587–2601, 2018.

[18] H. Fu, C. Li, X. Liu, J. Gao, A. Celikyilmaz, and L. Carin, "Cyclical annealing schedule: A simple approach to mitigating KL vanishing," *NAACL HLT 2019 - 2019 Conf. North Am. Chapter Assoc. Comput. Linguist. Hum. Lang. Technol. - Proc. Conf.*, vol. 1, pp. 240–250, 2019.

[19] X. Bresson and T. Laurent, "A Two-Step Graph Convolutional Decoder for Molecule Generation," *arXiv Prepr. arXiv1906.03412*, 2019.

[20] J. Zbontar, L. Jing, I. Misra, Y. LeCun, and S. Deny, "Barlow Twins: Self-Supervised Learning via Redundancy Reduction," *arXiv Prepr. arXiv2103.03230*, 2021.

[21] N. Vecoven, D. Ernst, and G. Drion, "A bio-inspired bistable recurrent cell allows for long-lasting memory," *PLoS ONE 16(6) e0252676.*, 2021.

[22] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using RNN encoder-decoder for statistical machine translation," *EMNLP 2014 - 2014 Conf. Empir. Methods Nat. Lang. Process. Proc. Conf.*, pp. 1724–1734, 2014.

[23] L. Thurner, A. Scheidler, F. Schafer, J. H. Menke, J. Dollichon, F. Meier, S. Meinecke, and M. Braun, "Pandapower - An Open-Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems," *IEEE Trans. Power Syst.*, vol. 33, no. 6, pp. 6510–6521, 2018.