Modeling of Critically Ill Patient Pathways to Support Intensive Care Delivery

William Trevena[1], Amos Lal[2], Simon Zec[2], Edin Cubro[2], Xiang Zhong[1], *IEEE Member*, Yue Dong[2], and Ognjen Gajic[2]

***Abstract*—The COVID-19 pandemic has exposed long standing deficiencies in critical care knowledge and practice in hospitals worldwide. New methods and strategies to facilitate timely and accurate interventions are needed. A virtual counterpart (digital twin) to critically ill patients would allow bedside providers to visualize how the organ systems interact to cause a clinical effect, offering them the opportunity to evaluate the effect of a specific intervention on a virtual patient before exposing an actual patient to potential harm. This work aims at developing a digital simulation that models the clinical pathway of critically ill patients. Using the mixed-methods approach with the support of multiprofessional clinical experts, we first identify the causal and associative relationships between organ systems, medical conditions, clinical markers, and interventions. We record these relationships as structured expert rules, depict them in a directed acyclic graph (DAG) format, and store them in a graph database (Neo4j). These structured expert rules are subsequently utilized to drive a simulation application that enables users to simulate the state trajectory of critically ill patients over a given simulated time period to test the impact of different interventions on patient outcomes. This simulation model will be the engine driving a future digital twin prototype, which will be used as an educational tool for medical students, and as a bedside decision support tool to enable clinicians to make faster and more informed treatment decisions.**

***Index Terms*—Critical care, Patient pathway, Digital simulation, Graph model.**

I. Introduction

TIMELY and accurate treatment is essential to achieve the best outcomes for many life-threatening conditions. In trauma patients, the opportunity to institute an early and effective treatment is often called the “golden hour” [1], [2]. Similarly, aggressive correction of shock and organ dysfunction during the first day, or “silver day,” was found to decrease patient length of stay and improve health outcomes [3]. Poor outcomes in sepsis, pneumonia, and other common intensive care syndromes were reported to be at least in part due to delayed recognition and management (“failure to rescue”) [4]. To mitigate this situation, diagnostic and therapeutic fidelity in the early hours of critical illness is paramount. Accurate data- and expert-guided treatments and less erroneous “treatment trials” are needed [5]. A virtual counterpart (digital twin) to the critically ill patients with decision support capabilities can potentially address these needs.

Digital twins, since their emergence, have been used to address a variety of diverse challenges in healthcare [6], [7], [8], [9]. A digital twin of patients (or virtual patients) is expected to replicate or simulate what happens during the interaction between a patient and the health care system. Previous works have successfully utilized virtual patient simulations to train medical professionals of various specialties for a variety of clinical settings [10], [11], [12]. Although moderate effectiveness in practice was identified (as shown in [10]), many commonly utilized virtual patient simulation architectures only progress along a limited number of pre-defined, handcrafted pathways. For example, linear text-based scenarios (TBS) and looped, branch serious games (SG) have been widely adopted [13].

In the case of linear TBS virtual patient simulations, the patient progresses along a pre-defined linear pathway that is always the same regardless of the decisions made or actions taken by the student. Students have an extremely limited ability to learn from their mistakes since the simulation progresses the same way regardless of the actions taken by the student. Alternatively, in the looped, branch SG, the patient progresses along one or more handcrafted pre-defined pathways or branches of a decision tree based on the decisions made by the student. Although the simulation progresses differently based on the actions taken by the student, instead of having to select a treatment from a large number of options (or choose to provide no treatment) as necessary in a real clinical setting, in the looped, branch SG, students are only allowed to select a treatment decision from a small pre-defined set of options (i.e., multiple choice). Consequently, the student can only learn from a limited number of potential scenarios and their associated pre-defined clinical outcomes.

To date, we are unaware of any non-linear or non-decision tree based *data-driven computational* simulation models in medical education that allow medical students to dynamically make arbitrary treatment decisions without being restricted to selecting treatment decisions from a limited number of options that are supported by handcrafted pre-defined pathways. Such data-driven computational virtual patient simulation models are necessary to provide medical students with a more realistic or near real-life experience of treating patients in critical care where they will need to make rapid treatment decisions by evaluating a large number of possible treatment options in real time. Such models will be pivotal in transforming medical education by increasing the knowledge, competency, and skill level of medical students, and by helping them become more mindful of the consequences of their actions to reduce the

This work was supported by the National Science Foundation IIS-2123848

[1] Department of Industrial and Systems Engineering, University of Florida, Gainesville, FL 32611, USA

[2] Mayo Clinic, Rochester, MN, 55902, USA

Please send all correspondences to Dr. X. Zhong. xiang.zhong@ise.ufl.edu

likelihood of them making dangerous treatment errors in real-life practice [14]. Furthermore, the majority of current virtual patient simulations are designed for static educational use, whereas computational simulation models can potentially be utilized in an additional capacity as bedside decision support tools. These models have the potential to identify optimal treatments in real time through simulation-based optimization, and they could potentially be utilized within an ensemble approach alongside previously introduced multivariate prescriptive prediction models used to guide personalized precision treatment planning (such as those discussed in [15]).

In this work, we propose a digital twin prototype of critically ill patients simulating the actions and interactions of major organ systems (cardiovascular, neurologic, renal, respiratory, gastrointestinal, inflammatory, and hematology), modeled as autonomous agents based on the causal and associative relationships between organ systems, medical conditions, clinical markets, and interventions. It should be noted that the proposed digital twin is not designed to be used explicitly for diagnosis; rather, it is designed to be used to recreate and predict the future state of the patient of interest based on both the current state of the patient and a set of treatments or interventions that have been administered or are under consideration. The state of the patient (in terms of their clinical markers) and the execution of the expert rules driving the simulation are primarily dependent on the time, medication, and medication doses that are administered at each time step of the simulation (among other factors).

Taking advantage of clinical expertise at the Mayo Clinic and granular electronic health record (EHR) data, we have developed and validated a simple computational simulation model of sepsis patients [16]. To enable the modeling of general critically ill patients to support both medical student training and decision support at the bedside, this paper introduces a novel directed acyclic graph (DAG)-driven virtual patient simulation architecture powered by a fast and scalable graph database (Neo4j). DAGs serve as flexible, efficient, and interpretable graphical tools that provide a visually friendly way to represent and understand key relationships (e.g., how the organ systems interact to cause a clinical effect). Compared to the “black-box” type of simulation, the DAG approach enables users to view relationships in a graph format to assist with their understanding. However, management of numerous DAGs and using them to efficiently and effectively drive a patient simulation is an imperative challenge. A graph database is therefore adopted to store, query, and analyze the DAGs. By storing the expert rules in a graph database, we strive to realize three benefits: (1) the expert rules driving the simulation can be updated by medical practitioners without the need to change any application code; (2) we reduce the size of the application and the amount of storage space consumed on the client; and, (3) we speed up the application by offloading to the database the determination of which rules are applicable at each time step. These properties are critical to the success of future, large-scale, and adaptive digital twin development.

In what follows, we provide detailed descriptions regarding the expert rules (Section II), the data structure (Section III), the simulation development and validation (Section IV), and the limitations and future directions (Section V).

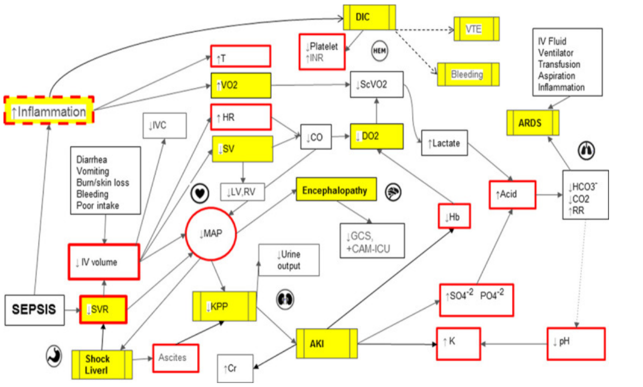


Fig. 1. An example of a directed acyclic graph (DAG) depicting complex pathophysiologic interactions in sepsis-associated multiple organ dysfunction originally published in [16]. AKI = acute kidney injury, ARDS = acute respiratory distress syndrome, CAM-ICU = confusion assessment method for ICU, CO = cardiac output, CO_2 = serum carbon dioxide, DIC = disseminated intravascular coagulation, DO_2 = arterial oxygen delivery, GCS = Glasgow Coma Scale, Hb = serum hemoglobin, HR = heart rate, INR = international normalized ratio, IVC = inferior vena cava, KPP = kidney perfusion pressure, LV = left ventricle, MAP = mean arterial pressure, RR = respiratory rate, RV = right ventricle, $ScVO_2$ = central venous oxygen saturation, SV = stroke volume, SVR = systemic vascular resistance, T = temperature, VO_2 = oxygen uptake, VTE = venous thromboembolism.

II. Expert Rules

Expert rules herein define the effects that variables have on each other (e.g., how changing heart rate impacts blood pressure), and various causes (interventions and interactions) that lead to certain effects on organ systems reflected by clinical markers. We use the term “clinical markers" to describe a patient’s vitals, physiological signs, and biomarkers that can be objectively measured (e.g., heart rate) or observed (e.g., imaging results). Organ systems and their corresponding clinical markers are the same for all patients. However, for different medical conditions, the importance of each clinical marker might differ (e.g., the blood pressure clinical marker would be more important for a patient with hypertension than for a patient without any cardiovascular diseases). Therefore, for a particular medical condition of interest, we prioritize characterizing the most relevant clinical markers that are directly impacted by the medical condition, or that are related to the symptoms used to diagnose the condition to reduce information overload. To systematically define the expert rules, we employed a mixed-methods approach with the support of multiprofessional clinical experts from the Mayo Clinic and the Society of Critical Care Medicine Discovery Research Network. The causal and association relationships were mainly derived using clinical data and available literature supporting the current standard of clinical practice including textbook chapters, critical care board reviews, and original research articles [16].

Because the number and complexity of the expert rules is very high, simulating these rules has historically been computationally prohibitive. To manage the expert rules in an efficient and effective way to enable the real-time simulation of patient pathways, we further depict the relationship of organ

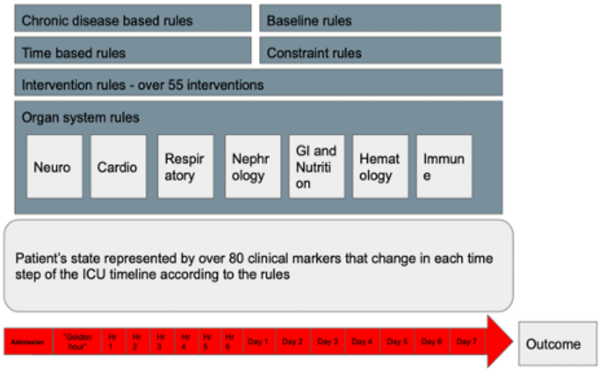


Fig. 2. Overview of the expert rules and patient simulation.

systems, medical conditions, clinical markers, and interventions in a DAG format with nodes and edges representing variables and relationships. For example, Figure 1 demonstrates a DAG depicting complex pathophysiologic interactions in sepsis-associated multiorgan dysfunction. The yellow boxes represent “Concepts," or medical conditions, the boxes with red solid border represent “Actionable clinical points," and the boxes with red interrupted border represent “Semi-actionable clinical points." As evident in Figure 1, a low mean arterial pressure (MAP) will lead to a low kidney perfusion pressure, which will ultimately lead to acute kidney injury (AKI).

We further created the following classes of nodes, based on a hierarchical order: (1) measurable/actionable clinical markers; (2) symptoms; (3) medical conditions; and (4) graduated clinical interventions.

1) Measurable/actionable clinical markers. This class includes patients’ vitals, physiological signs, and biomarkers. These clinical markers are typically measurable (objective). To ensure a manageable state space for a patient simulation, clinical markers are color coded as “white,” “yellow,” and “red.” White color indicates a normal value of the clinical marker. Yellow color indicates a disturbance in the clinical marker that needs to be closely monitored, but no immediate action is required. The color red characterizes a disturbance that needs to be acted upon immediately in order to prevent deterioration of the patient’s condition. Examples of this class include respiratory (respiratory rate, oxygen saturation, etc.), cardiovascular (MAP), heart rate, arrhythmia, etc.), neurologic (e.g., Glasgow Coma Scale (GCS)), fluid (pH, electrolytes, etc.), and immune homeostasis (inflammatory biomarkers such as C-reactive protein , white blood cell count, etc.), among other organ system clinical markers.
2) Symptoms. This class of nodes includes patients’ symptoms that are difficult to measure or quantify. Examples include vomiting, diarrhea, and brain swelling.
3) Medical conditions. We define a medical condition as a “concept” (denoted by the yellow boxes in a DAG, see Figure 1) that represents a pattern of clinical markers. These patterns of clinical markers are easily recognized by expert medical practitioners from their extensive clinical experience. Each concept can be affected by various clinical scenarios in our model. Specifically, each clinical scenario represents a situation where one or more of a patient’s organ systems are initially compromised in some way (e.g., a patient is having respiratory problems). Each clinical scenario is defined in terms of the initial abnormal clinical markers that a patient afflicted by the clinical scenario will have upon starting the simulation. These clinical markers will then change as per our expert rules and form state trajectories that will describe the patient’s condition. Examples include severe pneumonia resulting in respiratory failure or acute liver failure resulting in acute brain failure and encephalopathy.
4) Graduated clinical interventions. Interventions are classified into three states: “Not_Given," “Low_Dose," and “High_Dose." For instance, for a critically ill patient with hypertension, we expect that administration of vasopressors would improve the MAP from red (very low) to yellow (low), or from yellow (low) to white (normal) range. Examples of this class include nutrition, medications (including antibiotics, vasoactive agents, cardiac drugs, anesthetics, and sedatives), noninvasive or mechanical ventilation, IV fluids, source control (endoscopic retrograde cholangiopancreatography, abscess drainage, etc.), or blood product transfusion.

Clinically, any effect on the human body (or change in patient states) is considered as either a primary action or a secondary action. Primary actions are the causal and dominant effects which cause a disturbance in clinical markers, and which lead to the more significant consequences or adverse changes in clinical markers. Secondary actions are usually caused by primary actions or are correlated with causal variables. By definition, some clinical markers, such as heart rate, blood pressure, and respiratory rate can either cause or be affected by an expert rule. Low blood pressure, for example, can be changed by other clinical markers (where a change in blood pressure is the effect), but low blood pressure can in other rules be a causal variable that will induce a change in other clinical markers. In addition, an event causing one state transition can trigger a sequence of other events or exert a cascading effect. For example, when modeling a sepsis patient, five homeostatic mechanisms are key to sepsis progression: a) fluid volume & composition, b) acid-base homeostasis, c) clotting and bleeding, d) Oxygen delivery and consumption, and e) inflammation and immunosuppression. In this case (and many others), the nodes in the graph database representing the expert rules form a densely connected network with multiple layers.

III. Graph Database Development

We currently have developed nearly 300 expert rules and 20 DAGs compartmentalized based on organ systems or medical conditions, and we expect to continuously expand and refine these expert rules. Given the high-dimensionality in terms of the number of nodes and edges in the expert rule DAG and the evolving nature of expert rule development (e.g., because of enhanced clinical knowledge, a new clinical discovery, or the emergence of a new medical condition like the coronavirus disease), we need an approach that allows us to quickly query the rules, visualize the relationships (to help reconcile conflicting rules and support human cognition), and also to ensure satisfactory scalability and flexibility (e.g., adding many new rules over time).

For these reasons, we decided to use a graph database to store and represent the expert rules. A graph database stores nodes (vertices) and relationships (edges) instead of tables, or documents. We chose Neo4j as the graph database management system [17]. In Neo4j, information is organized as nodes, relationships, and their properties. Nodes are the entities in the graph. Nodes can be classified into distinct groups by tagging all nodes in each group with a common “label" (e.g., “Intervention"). Furthermore, nodes can be assigned properties (e.g., “Name"). Relationships provide directed connections or edges between two node entities, and like nodes, relationships can be easily classified into distinct groups by defining a distinct relationship “type" for each specific group or class of relationships (e.g., “Impacts"). Relationships always have a direction, a type, a start node, and an end node. Like nodes, relationships can also be assigned properties (e.g., “Impact_Strength"). By using labeled nodes and different types of relationships, it is possible to perform complex queries such as starting at all nodes with a specific label (e.g., “Intervention"), and then traversing all relationships of a specific type (e.g., “Impacts") to obtain all neighboring nodes of another distinct type (e.g., “Clinical_Marker").

The challenge is to develop a formal way to code the expert rules, e.g., what should be considered as nodes or relationships, and, what properties should be defined for each, as the nodes and relationships have various types (e.g., organ system based, interventions, etc.), and could also involve time, probability, and constraints. Below we describe the details regarding how the expert rules are structured and managed in the graph database. This demands a multidisciplinary effort since we need to ensure the fidelity of the expert rules and simultaneously allow them to be efficiently stored and queried.

First, independent expert rules are created by medical practitioners as rows in a spreadsheet. The expert rules are then exported as a CSV file, and by using Neo4j’s LOAD CSV feature, we automatically create nodes and edges representing the rules in the Neo4j graph database. A team of medical practitioners define rules in a shared spreadsheet in the format shown in Table I.

Each rule is first activated by a change in a single “triggering” clinical marker or intervention, and each rule causes a new incremental change in a single “impacted” clinical marker when all conditions for the rule are satisfied. For example, a decrease in MAP will cause an incremental increase in heart rate as long as a patient is not being administered a fentanyl drip. As shown in Table I, each row of the spreadsheet represents a different independent rule. The triggering clinical marker or intervention for each rule is stored in the “Cause” column and the impacted clinical marker is stored in the “Effected_Clinical_Marker” column.

Since a single clinical marker can have a varying effect on the human body depending on how it changes, we need to clearly define the way that the triggering clinical marker must change for each rule to be triggered. For each rule, this is defined by the valid starting state(s) and ending state(s) of the triggering clinical marker. The valid starting state(s) and ending state(s) are stored in the “Previous_State_Of_Cause” and “New_State_Of_Cause” columns, respectively. For interventions which trigger these rules, the valid starting state(s) and ending state(s) indicate whether or not the intervention has been given to the patient, and when applicable, the amount that has been given. When an intervention has not been given to a patient, the state of the intervention will be “Not_Given.”

For each rule, the incremental amount that the impacted clinical marker should change is stored in the “Impact” column. Since our initial model is discrete (e.g., color coded), the effect of each rule on the impacted clinical marker is represented by one of the following integers: (-2,-1,1,2). Here, based on the clinician inputs regarding vital specific normal value ranges, negative integers represent a decrease in the value or level of the impacted clinical marker, and positive integers represent an increase in the value or level of the impacted clinical marker. Here, -2 (resp. 2) represents a decrease (resp. increase) of two levels while -1 (resp. 1) represents a decrease (resp. increase) of only one level. For example, if a rule had an impact value of -2 and the impacted clinical marker was the patient’s GCS score, when this rule is applied, the patient’s GCS score would decrease by two levels, e.g., from yellow high to yellow low in a future state based on the time lapse it needs to be effective. Then, the GCS score will remain in that state unless another triggering event happens.

For a rule to be activated, during the most recent update of the simulation, the triggering clinical marker must have moved from one of the valid starting states to one of the valid ending states defined in the rule. Additionally, any relevant conditions defined in the rule must be satisfied. We divide the additional relevant conditions for each rule into three categories: (1) simple conditions, (2) complex conditions, and (3) timed conditions. We define simple conditions as one or more independent conditions which all must be satisfied for a rule to take effect. Simple conditions are stored in the “Simple_Cond” column. For example, consider the following rule:

- Cause: “Glucose”
- Previous_State_Of_Cause: “White”
- New_State_Of_Cause: “Red_Low”
- Effected_Clinical_Marker: “Glasgow_Coma_Scale (GCS)”
- Impact: -2
- Simple_Cond: {“Glucagon” : “Not_Given”}

Here, the triggering clinical marker is the patient’s glucose level, and this rule is only valid for cases where the patient’s glucose drops from “White” (a healthy level of glucose) to “Red_Low” (a dangerously low level of glucose). The impacted clinical marker in this rule is the patient’s GCS score, and when this rule is applied, the patient’s score is dropped two levels as indicated by the “Impact” value of -2. Here, we have a single simple condition which states that the patient must not be receiving an injection of Glucagon.

We use the term “complex conditions” to describe conditions which are satisfied if at least one of a possible set of conditions is satisfied. For example, a single complex condition could state that at least one of the following must be true: (1) a patient’s MAP is not at an abnormally low level; (2) the patient is being administered norepinephrine.

We use the term “timed conditions” to describe conditions which must be true for a certain amount of time for a rule to be applied. For example, consider a rule which causes an increase in Creatinine and is triggered by a low urine output. Since a patient must have a low urine output for an extended duration before a patient’s Creatinine level is increased, such a rule could have a timed condition which requires that a patient’s urine output is abnormally low for at least 24 hours.

If all of the conditions for a rule are satisfied, we then apply the rule with the probability listed in the “P" column. We utilize this to maintain a level of stochasticity in the simulation model. The probability herein characterizes the chance that a certain change in the human body will occur (e.g., a high probability could represent higher than 80%, a moderate probability is between 30% and 80%, and a low probability is below 30%). The current values are from expert input based on domain knowledge and past observations.

Following the evaluation of all conditions and the probability, we then start a countdown timer based on the time in the “Time_Until_Effect” column (in minutes). This column describes how long the simulation should wait before applying the rule once all conditions have been satisfied. For rules that have timed conditions, the countdown of the time in the “Time_Until_Effect” column only begins after all timed conditions have been satisfied. For example, if a timed condition required that urine output is low for more than 24 hours, once the 24 hours period has passed, then the countdown for the time in the “Time_Until_Effect” would begin. In addition, for rules that do have timed conditions, we leave a value of zero in the “Time_Until_Effect” column.

In summary, in the graph database, the causal and association relationships are featured by the following properties (1) probability (e.g., deterministic vs. probabilistic); (2) onset time (i.e., time until effect); (3) direction and intensity (e.g., decrease or increase 1 or 2 levels); (4) condition (e.g., timed condition); and (5) additional necessary constraints for a rule to take effect. This database structure is carefully crafted, which allows us to capture the majority of the common rules using a systematic format, and enables us to customize each expert rule based on the applicability of each property. This modeling approach ensures the scalability and flexibility of expert rule storage and efficient query by the simulation application.

IV. Patient Simulation

The expert rules are utilized to drive an agent-based simulation application that enables users to simulate the state trajectory of critically ill patients (e.g., urosepsis or pneumonia patients) over a desired time period (see Figure 2). Both a web and mobile (iOS) version of the simulation application are under development. A proof of concept for the interface of the web version is shown in Figure 3. The user will first load a clinical scenario (e.g., a medical condition like pneumonia, kidney failure, etc.). Then, the interface will present the patient’s basic information recorded during the admission. The application will allow users to simulate actions (e.g., clinical interventions) and interactions of major organ systems and their impacts on patient health states in the desired patient timeline (e.g., first 24 hours after admission). We assume that patients are always in one of a finite number of discrete health states, and they move from one state to another according to probabilities that depend on the current state of their overall health and the health care system. A computational patient simulation architecture was developed to integrate the expert rules stored in the graph database with agent-based modeling techniques. Below we briefly describe the major components for agent-based simulation that distinguish our model and allow it to perform more intuitively.

A. Modeling agent behaviors

Organ systems are considered as the autonomous agents. Their states are represented by high priority clinical markers identified as detailed in Section II. The transition of the states (e.g., increased heart rate, decreased urine output, decreased GCS, and dozens of others) will be triggered due to an intervention (e.g., a medication that has been given), jointly determined by the previous and current states of all associated major organ systems, and the baseline health state (evaluated based on variables like age, previous illnesses, alcohol use disorder, and smoking) by executing the expert rules.

We consider a Bayesian network model with relevant clinical markers as nodes in the network, and compute the probability of being at a certain level (e.g., red high or yellow low) of the “affected clinical marker" based on the status of its parent nodes (causes). Most of the rules exhibit the Markovian property, i.e., we mainly track the previous state and the new state of the causes to determine the effect. For those rules that do not possess the memoryless property, we introduce complex conditions and timed conditions. Before computing the state of the clinical marker, we will check all the related conditions to ensure these conditions are satisfied. Then, we further apply the chain rule of probability, or use ad hoc conditional probability tables based on additional input from experts. Finally, we sample the new state based on the calculated probability distribution. The behavior of the whole system of the human body then emerges according to these state transitions.

The patient timeline for the simulation is divided into carefully chosen steps, and all predictions are made for the patient health state in the next step. As we aim to model critically ill patients, after their ICU admission (T_0), the patient timeline is focusing on “golden hours” (every 15 minutes for the first hour, hourly for the first 6 hours, 12-hour interval for the first day and followed by 24 hour intervals thereafter) until 1 week (7 days) after admission (see Figure 2). Also, the timeline will be allowed to reset if there is a significant clinical event that takes place anytime during the treatment.

Cause	Previous _State_ Of_Cause	New _State_ Of_Cause	Effected _Clinical _Marker	Impact	P	Time _Until _Effect	Simple _Cond	Complex _Cond	Timed _Cond
Given_ Glucose	Not_Given	High_ Dose	GCS	2	1.0	15min	{Glucose : Red_Low, Propofol : Not_Given, Fentanyl : Not_Given}	{MAP : White, MAP : Yellow_High, MAP : Red_High, NE : Given}, {Brain_Swell : NA, Mannitol : Given}	NA

TABLE I

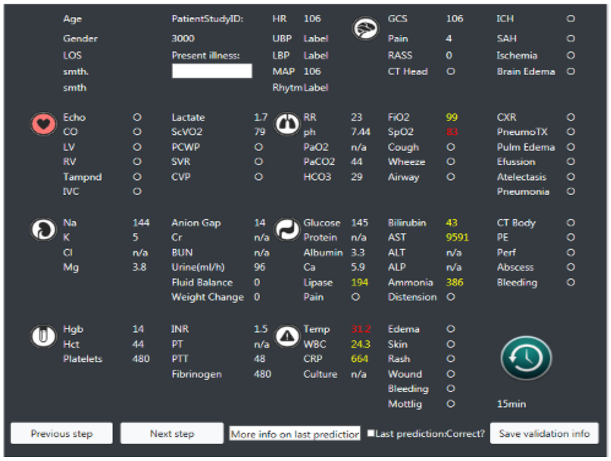
THE DATA STRUCTURE OF THE EXPERT RULES. GCS = GLASGOW COMA SCALE, MAP = MEAN ARTERIAL PRESSURE, NE = NOREPINEPHRINE.

B. Graph database integration

The patient's current state, previous states, and the interventions selected are recorded at each step of the simulation. They are used as arguments when querying the database to determine what expert rules should be applied to "predict" the next states of the simulated patient. An interface between the main simulation application and the graph database has been developed. The "expert rule book" (i.e., the graph database) will be queried at each time epoch to update the agent's states. The input includes the current states and the "delta" (i.e., the change in states between the current and previous time epoch) of the measurable clinical markers, the interventions (if applicable), and other case-specific information. The outputs are the relevant probabilistic relationships that define the interactions and the impact on all organ systems. Based on calculations using the Bayesian network model, the states of the patient will be updated, and the time advances to the next time epoch.

To fully characterize the patient states, joint effects have to be handled, and currently, we are considering different approaches. Many medical conditions happen individually (when they happen simultaneously, e.g., multi-organ failure, they typically correspond to a broader medical condition, e.g., sepsis). For these medical conditions, the DAG already considers the possible interaction of the related organ systems and biomarkers. For rules with complex conditions, it could be possible that only one rule will be effective at one time, or one rule will dominate the others. In rare situations, multiple rules could simultaneously change the same clinical marker, even in different directions. Assuming that all relevant conditions and constraints are satisfied, in the event that multiple rules impacting a single clinical marker are activated simultaneously, we hypothesize that the impact that is applied is the sum of their incremental changes. Note that by defining the impact of each rule as an incremental change, the resolution of the impact is actually low, and the states are changed in an ordinal fashion (e.g., white/normal, yellow/disturbance or red/major disturbance). In reality, the states could represent a range of numerical values (e.g., heart rate between 60 and 100 beats per minute is considered to be normal) and the joint effect might not be simply linear. The current assumption is made to make the model tractable while achieving an acceptable level of fidelity. The team is working to further validate this assumption by analyzing clinical data. Non-conforming and special scenarios will be handled on a case-by-case basis.

Joint effects of multiple clinical markers might eventually lead a clinical marker to its highest value (typically red high) or lowest value (typically red low). Once a clinical marker reaches one of its extreme ranges, our simulation model will maintain that range in case of any further advance in that (high or low) direction. For example, the red high range for heart rate is between 130 and 200 beats per minute. Once a heart rate reaches this red high range, the model will keep a heart rate in this range even if a change in another clinical marker triggers a rule which attempts to further increase the heart rate. The main program continuously checks the validity of the simulation states to ensure their values will not go beyond the maximum or minimum possible values.



```
Age              PatientStudyID:     HR     106               GCS         106      ICH           O
Gender           3000                UBP    Label             Pain        4        SAH           O
LOS              Present illness:    LBP    Label             RASS        0        Ischemia      O
smth.                                MAP    106               CT Head     O        Brain Edema   O
smth                                 RhytmLabel

Echo      O      Lactate        1.7  RR       23              FiO2        99       CXR           O
CO        O      ScVO2          79   ph       7.44            SpO2        83       PneumoTX      O
LV        O      PCWP           O    PaO2     n/a             Cough       O        Pulm Edema    O
RV        O      SVR            O    PaCO2    44              Wheeze      O        Efussion      O
Tampnd    O      CVP            O    HCO3     29              Airway      O        Atelectasis   O
IVC       O                                                                        Pneumonia     O

Na        144    Anion Gap      14   Glucose  145             Bilirubin   43       CT Body       O
K         5      Cr             n/a  Protein  n/a             AST         9591     PE            O
Cl        n/a    BUN            n/a  Albumin  3.3             ALT         n/a      Perf          O
Mg        3.8    Urine(ml/h)    96   Ca       5.9             ALP         n/a      Abscess       O
                 Fluid Balance  0    Lipase   194             Ammonia     386      Bleeding      O
                 Weight Change  0    Pain     O               Distension  O

Hgb       14     INR            1.5  Temp     31.2            Edema       O
Hct       44     PT             n/a  WBC      24.3            Skin        O
Platelets 480    PTT            48   CRP      664             Rash        O
                 Fibrinogen     480  Culture  n/a             Wound       O
                                                              Bleeding    O
                                                              Mottlig     O        15min
```

Fig. 3. User interface of the patient simulation application.

C. Model verification and validation

Our hypothesis is that the digital twin patient model will accurately represent the response to treatment of critical illness prospectively observed in real patients. The computer simulation model will be verified through a structured process. Each sub-module (cardiovascular, oxygenation and ventilation, neurological, etc.) will be tested independently followed by the entire simulation model. The inputs will be first set to deterministic values to check whether the program functions as the expert rules dictate. Output data will be descriptively summarized using frequencies and percentages for categorical data, and median and interquartile ranges for continuous variables. After the completion of debugging during alpha testing, we will proceed with the silent testing of the model where model outputs will not be known to bedside clinicians.

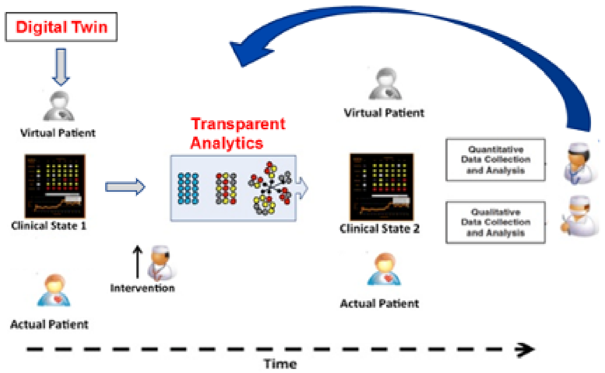


Fig. 4. Validation and refinement of the digital twin patient model through continuous feedback combining quantitative and qualitative approaches.

In our preliminary work [16], we have successfully validated a digital twin model of sepsis patients in the first 24 hours of admission. We expect to extend the previous data collection effort and prospectively validate the performance of this model. Demographic data and clinical data points needed for the study will be abstracted from the electronic medical records or directly from the previous database. Patients' baseline data and clinical data points will be used to define clinical markers' states. Each organ system will be validated individually followed by the validation of the entire virtual patient simulation. Treatment response will be observed in a sample of patients with sepsis and other critical illnesses admitted to the ICU from the emergency department. The response of the virtual patient will be measured against the real patient response (gold standard). For example, for a critically ill patient with hypotension, we expect that administration of vasopressors would improve the mean arterial pressure (MAP) from red (very low) to yellow (low) or from yellow (low) to white (normal) range. If the changes in the virtual patient are found to be concordant with the real patient, this will be considered as a success. However, if the observed output in digital twin varies from the real patient, it would fall under one of the error types (coding error, expert rule error, EHR error, unaccounted error secondary to a known medication, etc.). Agreement statistics (Kappa Coefficient, Bland-Altman), area under receiver operating curve, sensitivity, and specificity will be used as appropriate to determine the accuracy and precision of the computer simulation output compared to actual events prospectively observed in real patients. We are working to set up pilot runs to identify commonly occurring errors and iteratively improve the model fidelity (see Figure 4).

V. Discussions and Future Work

Computational simulation models have been found to be highly effective in healthcare applications such as drug development [18] and sample size determination for optimal treatment prediction models [19]. Furthermore, sophisticated multiscale models integrating human physiology, disease biology, and molecular pathways have been developed and applied to multiple medical problems including the treatment of sepsis [20], [21] and diabetes mellitus [22]. These models are well suited to assist in the development of new pharmacological approaches and medical devices, but their applicability as decision support tools in a clinical context (e.g., critical care delivery) have been limited [22].

Critical illness offers a number of advantages for model developers, such as the availability of large quantities of quantitative data. This has resulted in an increasing effort to develop data-driven clinical decision support tools [23], [24], [25]. However, the performance of these models are limited by the amount of available data, and clinicians are weary of these "black-box" models without clearly understanding the underlying rules that guide the model outputs. As such, these purely data-driven models might underperform in the live clinical setting and struggle to reach the brink of clinical utility at the bedside. Without explicit consideration of known causal pathways (based on biological and physiological understanding informed by experts), the model output can lead to results at best counterintuitive or uninterpretable, at worst inaccurate and detrimental. More importantly, even accurate prognostic information (classifying patients who will require renal replacement therapy or die during critical illness) is of limited value to the bedside clinician [26]. Predictive information — predicting the risk vs. benefit of a particular treatment — is of greater clinical value. For example, will my patient benefit from a red cell transfusion, or continuous versus intermittent renal replacement? Models with poor interpretability are unlikely to deliver transformative change to clinical medicine and predictive enrichment requires innovation [27].

A. *Path forward: causal AI models and digital twins*

A fundamental difference between "black-box" prediction models and human intelligence is the human ability to recognize and reduce uncertainty through inquiry and observation of response to treatment. To overcome the shortcomings of purely data-driven models, an alternative approach, "causal AI" is emerging. Causal AI models represent a class of statistical learning models that are developed using relevant and timely patient data, strongly informed by expert rules that define a causal structure or other clinically informed structure to the data. These models will be developed with a deep understanding of the underlying causal pathways, therefore providing greater physiological homology and eventually translating into better performance. For instance, we can consider the use of graph neural networks, neural networks based on graph architectures, to predict patient outcomes and to identify optimal treatments [28]. We are striving to develop a digital twin prototype, which can be characterized as a special type of computer simulation that implements a causal AI model, combining current data from the patient/subject with its model algorithm explicitly designed based on the causal principles.

B. *Digital twins: challenges and opportunities*

The availability of high-resolution quantitative patient data and a relatively short trajectory of critical illness to a stable outcome makes the ICU an ideal environment for development and testing of EHR data augmented digital twin patient models. In our current practice, the expert rules are entirely based on expert knowledge. In the future, we expect to build causal Bayesian network [29] and dynamic Bayesian network [30] using EHR data with structure informed, in part, by expert rule DAGs. Currently, raw EHR data are not optimized for human or computer decision making. To attain this target, it is of vital importance that the most "meaningful data" is separated effectively from the noise in the EHR. In addition, the causal AI digital twin will need to be validated prospectively using near-real time EHR data feeds, simulating application in an environment that may pose unique challenges and insights not captured retrospectively. The temporal relationship of interventions and outcomes (e.g., the time and pace of disease progression and associated effects) will need to be studied with a higher degree of accuracy and reliability. These models will require the development of computationally efficient statistical learning approaches using mainly observational data, which is still a cutting-edge research area [31]. Furthermore, a systems engineering framework for prospective refinement of the digital twin model will serve as the corner stone for future implementation. We envision causal AI digital twins that will simultaneously deliver predictive and prognostic information as a critical leap forward in support of more efficient medical education and less error-prone bedside decision making.

Acknowledgment

We would like to thank John M. Litell, D.O. and Nika Zorko Garbajs, M.D. for their support in developing expert rules.

References

[1] K. Fassbender, C. Balucani, S. Walter, S. R. Levine, A. Haass, and J. Grotta, "Streamlining of prehospital stroke management: the golden hour," The Lancet Neurology, vol. 12, no. 6, pp. 585–596, 2013.

[2] Y. Heled, M. Rav-Acha, Y. Shani, Y. Epstein, and D. S. Moran, "The "Golden Hour" for Heatstroke Treatment," Military Medicine, vol. 169, no. 3, pp. 184–186, 03 2004.

[3] O. Blow, L. Magliore, J. A. Claridge, K. Butler, and J. S. Young, "The golden hour and the silver day: Detection and correction of occult hypoperfusion within 24 hours improves outcome from major trauma," Journal of Trauma and Acute Care Surgery, vol. 47, no. 5, 1999.

[4] E. Rivers, B. Nguyen, S. Havstad, J. Ressler, A. Muzzin, B. Knoblich, E. Peterson, and M. Tomlanovich, "Early goal-directed therapy in the treatment of severe sepsis and septic shock," New England journal of medicine, vol. 345, no. 19, pp. 1368–1377, 2001.

[5] N. H. G. Holford, H. C. Kimko, J. P. R. Monteleone, and C. C. Peck, "Simulation of clinical trials," Annual Review of Pharmacology and Toxicology, vol. 40, no. 1, pp. 209–234, 2000, pMID: 10836134.

[6] H. Laaki, Y. Miche, and K. Tammi, "Prototyping a digital twin for real time remote control over mobile networks: Application of remote surgery," IEEE Access, vol. 7, pp. 20 325–20 336, 2019.

[7] C. Pizzolato, D. J. Saxby, D. Palipana, L. E. Diamond, R. S. Barrett, Y. D. Teng, and D. G. Lloyd, "Neuromusculoskeletal modeling-based prostheses for recovery after spinal cord injury," Frontiers in Neurorobotics, vol. 13, 2019.

[8] K. Bruynseels, F. Santoni de Sio, and J. van den Hoven, "Digital twins in health care: Ethical implications of an emerging engineering paradigm," Frontiers in Genetics, vol. 9, 2018.

[9] Y. Liu, L. Zhang, Y. Yang, L. Zhou, L. Ren, F. Wang, R. Liu, Z. Pang, and M. J. Deen, "A novel cloud-based framework for the elderly healthcare services using digital twin," IEEE Access, vol. 7, pp. 49 088–49 101, 2019.

[10] A. A. Kononowicz, L. A. Woodham, S. Edelbring, N. Stathakarou, D. Davies, N. Saxena, L. T. Car, J. Carlstedt-Duke, J. Car, and N. Zary, "Virtual patient simulations in health professions education: Systematic review and meta-analysis by the digital health education collaboration," Journal of Medical Internet Research, vol. 21, no. 7, p. e14676, Jul. 2019.

[11] Q. Wu, Y. Wang, L. Lu, Y. Chen, H. Long, and J. Wang, "Virtual simulation in undergraduate medical education: A scoping review of recent practice," Frontiers in Medicine, vol. 9, 2022.

[12] C. Y. Lee and S. W. H. Lee, "Review: Impact of the educational technology use in undergraduate pharmacy teaching and learning – a systematic review," Pharmacy Education, vol. 21, p. p. 159–168, Jun. 2021.

[13] J. Berger, N. Bawab, J. De Mooij, D. Sutter Widmer, N. Szilas, C. De Vriese, and O. Bugnon, "An open randomized controlled study comparing an online text-based scenario and a serious game by belgian and swiss pharmacy students," Currents in Pharmacy Teaching and Learning, vol. 10, no. 3, pp. 267–276, 2018.

[14] R. Hester, W. Pruett, J. Clemmer, and A. Ruckdeschel, "Simulation of integrative physiology for medical education," Morphologie, vol. 103, no. 343, pp. 187–193, 2019.

[15] Z. D. Cohen and R. J. DeRubeis, "Treatment selection in depression," Annual Review of Clinical Psychology, vol. 14, no. 1, pp. 209–236, 2018, pMID: 29494258.

[16] A. Lal, G. Li, E. Cubro, S. Chalmers, H. Li, V. Herasevich, Y. Dong, B. W. Pickering, O. Kilickaya, and O. Gajic, "Development and verification of a digital twin patient model to predict specific treatment response during the first 24 hours of sepsis," Critical care explorations, vol. 2, no. 11, 2020.

[17] "Neo4j," Oct 2021. [Online]. Available: https://neo4j.com/

[18] H. Wang, O. Milberg, I. H. Bartelink, P. Vicini, B. Wang, R. Narwal, L. Roskos, C. A. Santa-Maria, and A. S. Popel, "<i>in silico</i> simulation of a clinical trial with anti-ctla-4 and anti-pd-l1 immunotherapies in metastatic breast cancer using a systems pharmacology model," Royal Society Open Science, vol. 6, no. 5, p. 190366, 2019.

[19] A. Luedtke, E. Sadikova, and R. C. Kessler, "Sample size requirements for multivariate models to predict between-patient differences in best treatments of major depressive disorder," Clinical Psychological Science, vol. 7, no. 3, pp. 445–461, 2019.

[20] T. Eissing, L. Kuepfer, C. Becker, M. Block, K. Coboeken, T. Gaub, L. Goerlitz, J. Jaeger, R. Loosen, B. Ludewig et al., "A computational systems biology software platform for multiscale modeling and simulation: integrating whole-body physiology, disease biology, and molecular reaction networks," Frontiers in physiology, vol. 2, p. 4, 2011.

[21] M. McDaniel, J. M. Keller, S. White, and A. Baird, "A whole-body mathematical model of sepsis progression and treatment designed in the biogears physiology engine," Frontiers in Physiology, vol. 10, 2019.

[22] D. M. Eddy and L. Schlessinger, "Archimedes: a trial-validated model of diabetes," Diabetes care, vol. 26, no. 11, pp. 3093–3101, 2003.

[23] T. J. Pollard and L. A. Celi, "Enabling machine learning in critical care," ICU management & practice, vol. 17, no. 3, p. 198, 2017.

[24] T. J. Pollard, A. E. Johnson, J. D. Raffa, L. A. Celi, R. G. Mark, and O. Badawi, "The eicu collaborative research database, a freely available multi-center database for critical care research," Scientific data, vol. 5, no. 1, pp. 1–13, 2018.

[25] M. Komorowski, L. A. Celi, O. Badawi, A. C. Gordon, and A. A. Faisal, "The artificial intelligence clinician learns optimal treatment strategies for sepsis in intensive care," Nature medicine, vol. 24, no. 11, pp. 1716–1720, 2018.

[26] N. Tomašev, X. Glorot, J. W. Rae, M. Zielinski, H. Askham, A. Saraiva, A. Mottram, C. Meyer, S. Ravuri, I. Protsyuk et al., "A clinically applicable approach to continuous prediction of future acute kidney injury," Nature, vol. 572, no. 7767, pp. 116–119, 2019.

[27] M. Singer, C. S. Deutschman, C. W. Seymour, M. Shankar-Hari, D. Annane, M. Bauer, R. Bellomo, G. R. Bernard, J.-D. Chiche, C. M. Coopersmith et al., "The third international consensus definitions for sepsis and septic shock (sepsis-3)," Jama, vol. 315, no. 8, pp. 801–810, 2016.

[28] Feb 2022. [Online]. Available: https://neptune.ai/blog/graph-neural-network-and-some-of-gnn-applications

[29] J. Yu, V. A. Smith, P. P. Wang, A. J. Hartemink, and E. D. Jarvis, "Advances to bayesian network inference for generating causal networks from observational biological data," Bioinformatics, vol. 20, no. 18, pp. 3594–3603, 2004.

[30] Z. Ghahramani, "Learning dynamic bayesian networks," International School on Neural Networks, Initiated by IIASS and EMFCSC, pp. 168–197, 1997.

[31] N. Ramanan and S. Natarajan, "Causal learning from predictive modeling for observational data," Frontiers in big Data, p. 34, 2020.