UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE
ENGENHARIA ELÉTRICA

Previsão do Tempo de Parada para Decantação de Sedimentos e Limpeza das Grades de Proteção das Unidades Geradoras em Usinas Hidrelétricas Utilizando Modelos Ocultos de Markov e Redes Bayesianas

Lênio Oliveira Prado Júnior

Itajubá, Dezembro, 2023

UNIVERSIDADE FEDERAL DE ITAJUBÁ
PROGRAMA DE
ENGENHARIA ELÉTRICA

Lênio Oliveira Prado Júnior

Previsão do Tempo de Parada para Decantação de Sedimentos e Limpeza das Grades de Proteção das Unidades Geradoras em Usinas Hidrelétricas Utilizando Modelos Ocultos de Markov e Redes Bayesianas

Tese de doutorado submetida ao Programa de Engenharia Elétrica como parte dos requisitos para obtenção do Título de Doutor em Ciências em Engenharia Elétrica.

Área de Concentração: Sistemas Elétricos de Potência

Orientador: Prof. Dr. Guilherme Sousa Bastos
Coorientador: Prof. Dr. Edson da Costa Bortoni

Dezembro, 2023
Itajubá

Lênio Oliveira Prado Júnior

Previsão do Tempo de Parada para Decantação de Sedimentos e Limpeza das Grades de Proteção das Unidades Geradoras em Usinas Hidrelétricas Útilizando Modelos Ocultos de Markov e Redes Bayesianas

Tese de doutorado submetida ao Programa de Engenharia Elétrica como parte dos requisitos para obtenção do Título de Doutor em Ciências em Engenharia Elétrica.

Trabalho aprovado. Itajubá, 14 de Dezembro de 2023:

Prof. Dr. Guilherme Sousa Bastos
Orientador

Prof. Dr. Edson da Costa Bortoni
Coorientador

Prof. Dr. André Luis Marques Marcato

Prof. Dr. Agnelo Marotta Cassula

Prof. Dr. Adler Diniz de Souza

Prof. Dr. Mauricio Campos Passaro

Itajubá
Dezembro, 2023

Agradecimentos

Agradeço primeiramente a Deus pela vida, pelas bençãos recebidas e também pela oportunidade de realização do sonho de finalizar o doutorado. Agradeço imensamente ao meu orientador Guilherme Bastos, pelo acompanhamento e paciência, e principalmente pela possibilidade de participação no projeto de P&D da ANEEL junto à Jirau Energia. A todos os funcionários da Jirau Energia, deixo meus sinceros agradecimentos pelo conhecimento compartilhado e pelas muitas reuniões de alinhamento. À minha família, agradeço pela paciência e companhia enquanto trilhava o longo caminho de pesquisa e estudos. À minha esposa Izadora Araújo Garcia e filhos Matheus Braga Prado e Heitor Garcia Prado, que bem sabem o quanto me ausentei para alcançar este objetivo, dedico este trabalho. A presença de vocês me deu força, e nos momentos de dificuldade foi minha fonte de inspiração e energia, necessárias para seguir em frente. À minha mãe Seila Maria Dias Prado e meu pai Lênio Oliveira Prado, agradeço pela vida e pelo incentivo incansável. Aos meus amigos e parceiros de projeto, agradeço pelo constante aprendizado e auxílio durante toda a jornada de P&D e doutorado. Dedico atenção e carinho especiais aos meus amigos Emerson Assis de Carvalho, Ricardo Emerson Julio e Mateus Gabriel Santos, pois vocês foram essenciais para a conclusão deste trabalho, tanto compartilhando conhecimentos técnicos quanto provendo momentos de descanso e lazer. Agradeço de modo particular ao Fábio Júnior Alves por apresentar a mim a possibilidade de participação no projeto de P&D. Agradeço a todos os meus colegas do Instituto Federal, que de uma forma ou de outra me auxiliaram, seja através de revisão de artigos e textos, seja através de cooperação durante aulas e atividades acadêmicas. Por fim, agradeço a todos aqueles que de alguma forma contribuíram para que o objeto final de conclusão do doutorado pudesse ser alcançado.

"Os limites do possível só podem ser definidos indo além do impossível."
(Arthur C. Clarke)

Resumo

Usinas hidrelétricas instaladas na bacia Amazônica são influenciadas negativamente pelo transporte de troncos e sedimentos trazidos pelo leito do rio. Todo o material transportado se acumula nas grades de proteção, que são instaladas nas tomadas d'água das unidades geradores para prevenir a entrada desse material e evitar danos nas turbinas, o que resultaria em prejuízos econômicos. O acúmulo de material nas grades de proteção reduz a queda d'água, e consequentemente a vazão disponível para geração, o que resulta em perdas de carga, impedindo que a unidade de geração seja capaz de operar a plena capacidade. O problema apresentado ocorre na usina hidrelétrica de Jirau, que está situada no Rio Madeira e possui 50 unidades de geração, cada uma com o potencial de geração de 75MW a plena capacidade. Esta usina opera a fio d'água, o que significa que não há um reservatório para armazenamento da água do rio, sendo assim, todo o recurso hídrico deve ser utilizado quando disponível. O fato de ser uma usina a fio d'água, juntamente com o problema de transporte de troncos e sedimentos, traz desafios à operação, pois os sedimentos que se acumulam nas grades de proteção e consequentemente reduzem o potencial de geração requerem o desligamento total das unidades geradoras para que a sujeira acumulada possa decantar, possibilitando assim a retomada da queda d'água, e consequentemente a vazão disponível para geração. A grande quantidade de unidades de geração, juntamente com o fato de que a disposição desses equipamentos em posições diferentes do rio altera o perfil de acúmulo de sedimentos, dificulta a definição de regras que definam qual o tempo necessário de parada de cada unidade para decantação da sujeira e retomada da geração a plena capacidade. Neste cenário, ao invés de contar apenas com a experiência dos operadores, a elaboração de métodos eficientes que sejam capazes de determinar o tempo ideal que cada unidade geradora deve permanecer parada se torna essencial. Neste trabalho são propostos modelos de previsão utilizando Redes Bayesianas e Modelos Ocultos de Markov para estimar o tempo de parada necessário para decantação da sujeira de cada unidade de geração, de modo que esta possa ser utilizada novamente o mais rápido possível. Também são utilizadas técnicas de Big Data e Analytics para coletar e processar o grande volume de dados existentes na usina hidrelétrica. Os resultados obtidos demonstram que os modelos desenvolvidos foram capazes de inferir de modo satisfatório o tempo necessário para decantação dos sedimentos. O modelo resultante possibilita a consulta de informações utilizando várias informações, dentre elas o nível de obstrução no momento da parada de uma unidade, qual o nível de obstrução no momento da retomada, se haviam unidades vizinhas operando e em qual faixa de potência.

Palavras-chaves: Big Data Analytics. Usinas Hidrelétricas. Estados Operacionais de Unidades Geradoras. Modelos Gráficos Probabilísticos, Redes Bayesianas, Modelos Ocultos de Markov.

Abstract

Hydropower plants installed in the Amazon basin are negatively influenced by transporting logs and sediments from the river bed. All transported material accumulates in the protection grids, which are installed in the water intake of the generating units to prevent the entry of this material and avoid damage to the turbines, which would result in economic losses. The accumulation of material on the protection grids reduces the water drop and, consequently, the flow available for generation, which results in load losses, preventing the generation unit from operating at full capacity. The presented problem occurs at the Jirau hydropower plant, located on the Madeira River and has 50 generation units, each with the potential to generate 75MW at total capacity. This plant operates on a run-of-river basis, meaning there is no reservoir for water storage. Therefore, all water resources must be used when available. The fact that it operates on a run-of-river, together with the problem of transporting logs and sediments, brings challenges to the operation of the plant, as the sediments that accumulate in the protection grids and consequently reduce the generation potential require the complete shutdown of the generating units so that the accumulated dirt can decant, thus enabling the resumption of the waterfall, and consequently the flow available for generation. A large number of generation units and the different location dispositions of equipment in the river alter the sediment accumulation profile, making it difficult to define rules that define the necessary stoppage time for each unit to decant the dirt and restart generation at total capacity. In this scenario, instead of relying only on the experience of operators, developing efficient methods capable of determining the ideal stoppage time for each generation unit becomes essential. In this work, prediction models using Bayesian Networks and Hidden Markov Models are proposed to estimate the downtime required for decanting the dirt from each generation unit so that it can be used again for generation in the shortest possible time. Big Data and Analytics techniques are also used to collect and process the large volume of data existing at the hydroelectric plant. The results demonstrate that the developed models could satisfactorily infer the time required for sediment decantation. The resulting model makes it possible to query information using various information, including the obstruction level when a unit stops, the obstruction level at restart time, whether neighboring units are operating, and in which power range.

Keywords: Big Data Analytics. Hydropower Plants. Generating Units Operational States. Probabilistic Graphical Models, Bayesian Networks, Hidden Markov Models.

Lista de figuras

Lista de tabelas

Lista de abreviaturas e siglas

AFD	Autômato Finito Determinístico
AG	Algoritmo Genético
ANEEL	Agência Nacional de Energia Elétrica
API	Application Programming Interface
ARIMA	Autoregressive Integrated Moving Average
BDA	Big Data Analytics
BI	Business Intelligence
CQL	Cassandra Query Language
CPD	Conditional Probability Distribution
CSV	Comma Separated Values
DAG	Directed Acyclic Graphs
DF	Data Frame
ETL	Extract, Transform and Load
GIS	Geographic Information Systems
HDFS	Hadoop Distributed File System
HMM	Hidden Markov Models
IC	Independência Condicional
JSON	JavaScript Object Notation
MDP	Markov Decision Process
ML	Machine Learning
MPO	Manual de Procedimentos da Operação
NILM	Non-Intrusive Load Monitoring
NoSQL	Not Only Structured Query Language

ONS	Operador Nacional do Sistema
PEV	Plug-in Electric Vehicle
PGI	Programa de Gestão Integrada
PLC	Programmable Logic Controller
POO	Programação Orientada a Objetos
P&D	Pesquisa e Desenvolvimento
QBPSO	Quantum-Based Particle Swarm Optimization
RAM	Random Access Memory
RB	Redes Bayesianas
RESTFul	Representational State Transfer
RF	Random Forest
RNA	Redes Neurais Artificiais
SAMUG	Sistema de Apuração de Mudanças de Estados Operativos
SAU	Sistema de Acompanhamento de Usinas
SAX	Symbolic Aggregate Approximation
SCADA	Supervisory Control and Data Acquisition
SG	Smart Grids
SGI	Sistema de Gestão de Intervenções
SIN	Sistema Interligado Nacional
SVM	Support Vector Machine
SQL	Structured Query Language
UG	Unidade Geradora
UHE	Usina Hidrelétrica
XML	Extensible Markup Language

Sumário

1 Introdução

1.1 Motivação

A operação de usinas hidrelétricas (UHEs) é uma tarefa que envolve tecnologias aprimoradas e uma constante busca por métodos que visem o melhor aproveitamento dos recursos hidrológicos, de tecnologias de ponta disponíveis e melhorias contínuas no processo de geração, com a finalidade de manter a estabilidade do sistema elétrico, a segurança da operação e a redução de custos operacionais (LI; WANG; LI, 2018; ASSOCIATION et al., 2018).

Para viabilizar todo o processo envolvido na operação, é comum em UHEs a utilização de sensores para coletar todo tipo de informação, como o nível de reservatório, status dos equipamentos, temperatura, nível de óleo, etc. Tais sensores fornecem informações aos operadores para auxiliar na tomada de decisões e controlar todo o funcionamento da UHE. A leitura das informações é realizada através de aparatos de hardware instalados em diversos locais e equipamentos que precisam ser monitorados. O processamento e tratamento dos dados, a fim de fornecer informações para a operação, ocorre através de sistemas de software (SELAK; BUTALA; SLUGA, 2014).

Dado o grande número de equipamentos a serem monitorados, a quantidade de informações geradas pelas UHEs é enorme, uma vez que os dados são coletados a cada segundo, ou mesmo em instantes menores de tempo. Toda essa informação armazenada em diversos sistemas da UHE contém um enorme valor agregado, e é desejável aplicar técnicas que sejam capazes de extrair informações relevantes deste conjunto de dados para utilização em diversos cenários, dentre eles, o auxílio na tomada de decisões (ZHAN et al., 2014; GHORBANIAN; DOLATABADI; SIANO, 2019).

A possibilidade de extração de informações de um grande volume de dados, coletado de tantas fontes e armazenado em diversos sistemas pela UHE, evidencia a importância do processo analítico. Técnicas para mineração e descoberta de padrões nos dados já são utilizadas em diversas áreas, como indústrias, empresas e provedoras de serviços (CAO et al., 2010), analisando transações, preferências de usuário, correlações nos dados, e outros tipos de aplicações (ZHAN et al., 2014).

Dentro do contexto de gerenciamento de informações, *Big Data*, um paradigma de processamento de grandes quantidades de dados, associado a um conjunto de ferramentas para analisar e extrair informações valiosas desses dados, conhecido como *Big Data Analytics* (BDA), tem sido utilizado nas mais diversas áreas (SAGIROGLU et al., 2016; BHATTARAI et al., 2019; GHORBANIAN; DOLATABADI; SIANO, 2019). O termo *Big*

Data está relacionado às tarefas de coleta, armazenamento, processamento, extração de valor e apresentação de volumes massivos de dados.

Não apenas no contexto de UHEs, mas em toda a estrutura dos sistemas elétricos, desde a geração de energia, transmissão, distribuição e o consumo propriamente dito, a utilização inteligente dos dados coletados tem o potencial de desbloquear inovadoras oportunidades que visam disponibilizar melhorias nos processos, na qualidade do produto ou serviço oferecido, além de ganhos técnicos e econômicos (BHATTARAI et al., 2019).

O setor energético tem enfrentado diversos desafios, como aprimorar a eficiência operacional, controle de custos, confiabilidade, gerenciamento de energia renovável e questões ambientais (AMIN, 2008; ZHOU et al., 2015), e a utilização de BDA pode auxiliar na solução desses desafios (LV et al., 2017). As técnicas relacionadas ao uso de BDA tem o potencial de transformar grandes volumes de dados em ações para melhorias no planejamento e operação do setor elétrico (JIANG et al., 2016). De acordo com trabalhos existentes na literatura, problemas relacionados a *Big Data* têm sido debatidos há alguns anos (AMIN, 2008; JIANG et al., 2016). Embora o assunto não seja tão recente, ainda há questões em aberto, e até mesmo a definição sobre o que é e a partir de qual tamanho se considera *Big Data* ainda é um tema em discussão (ZHOU; FU; YANG, 2016).

Especificamente em UHEs, há uma grande quantidade de dados disponíveis, coletadas por sistemas supervisórios, sistema de acompanhamento, informações adicionadas pelos operadores, registros de manutenção e histórico hidrológico (ZHOU; FU; YANG, 2016). O grande desafio está na integração de dados disponíveis em diversas origens, com taxas de coleta variáveis, formatos diferentes e armazenados em sistemas de diferentes fornecedores, para então dar algum sentido a esses dados.

Dentre as possibilidades trazidas pela aplicação de técnicas de BDA, pode-se destacar a utilização de técnicas de aprendizagem de máquina (*Machine Learning* - ML). ML é um ramo da inteligência artificial, e sistemas dessa categoria tem como característica a modificação do seu comportamento de forma automática à medida que ocorre o aprendizado de novas informações, seja através de novos exemplos, por eventos que ocorrem com o passar do tempo, ou mesmo pela própria experiência do sistema em questão (MOHAMMED; KHAN; BASHIER, 2016).

O termo ML abrange uma enorme quantidade de técnicas, modelos e algoritmos, sendo que cada tipo de cenário se adequa à utilização de um tipo específico de técnica. Existem técnicas para a realização de previsões considerando a incerteza do ambiente objeto de estudo, dentre elas podemos citar aquelas que são capazes de considerar a probabilidade de que algum evento ocorra. A previsão consiste na inferência de valores faltantes em uma base de dados, baseando-se em probabilidades estatísticas ou empíricas, ou em estimar valores futuros baseados em dados históricos (ZHOU et al., 2006).

Determinar um cronograma ideal para as Unidades Geradoras (UGs) disponíveis, a fim de atender a demanda de carga energética, é um problema complexo e que envolve inúmeras variáveis. Inicialmente, é necessário identificar qual é o estado operativo das UGs, para que seja possível definir se esta pode ser utilizada no processo de geração em determinado momento. Realizar o processo de identificação do estado operacional de uma pequena quantidade de UGs pode não ser um processo muito difícil, mas o problema se torna complexo quando inúmeros equipamentos adjacentes estão envolvidos no processo para identificação do estado, ainda mais se houverem várias UGs a serem analisadas. Nos casos onde as UHEs possuem um grande número de UGs, a complexidade na análise de informações se torna um ponto problemático, dada a quantidade de dados que devem ser processados.

A implementação das técnicas e procedimentos descritos neste trabalho foi executada na UHE Jirau, a 4ª maior geradora de energia do Brasil em potência instalada, que está localizada a 120 quilômetros medidos ao longo do Rio Madeira na cidade de Porto Velho, capital do estado de Rondônia. Essa UHE possui 50 turbinas do tipo bulbo operando a plena capacidade, o que significa uma potência instalada total de 3.750 MW.

Um dos aspectos importantes da UHE Jirau está no fato de que ela opera o reservatório a fio d'água, o que exige que a vazão afluente seja turbinada, uma vez que não é possível armazenar o recurso hídrico. Tal aspecto torna a operação ainda mais complexa, exigindo que as UGs estejam disponíveis para geração a maior parte do tempo, tornando o detalhamento de disponibilidade um ponto crucial, o que faz com que a previsibilidade do tempo que as UGs ficam inoperantes um ponto de extrema importância no planejamento da operação.

A característica do ambiente onde a UHE está inserida também adiciona complexidade, dificultando ainda mais a operação. Na UHE Jirau, um fator ligado à natureza do rio interfere diretamente na operação, e consiste em um problema que requer a busca por tecnologias e métodos que possam amenizar o impacto causado. Tal fator refere-se ao transporte de sedimentos e troncos pelo leito do rio, que acabam se acumulando nas grades de proteção presentes em cada uma das UGs. O acúmulo de sedimento reduz o volume da queda d'água disponível, e consequentemente reduz a capacidade de geração energética.

Dado o problema de transporte de sedimentos pelo leito do rio, algumas abordagens são utilizadas pela UHE Jirau para solucionar o problema, dentre elas está a paralisação das UGs que se encontram com muito sedimento acumulado para que ocorra a decantação da sujeira. O tempo exato necessário para que o processo de decantação seja eficaz não é conhecido pelos operadores da UHE, e de tal modo, atualmente é utilizado um tempo aleatório, de acordo com a intuição e conhecimento prévio dos operadores.

Propiciar uma metodologia eficiente para determinar o tempo necessário que uma

UG deve permanecer parada para decantação viabiliza uma melhora significativa na operação da UHE, uma vez que propicia aos operadores uma estimativa de tempo de parada mais realista, permitindo que a UG possa ser alocada na programação de geração com maior confiabilidade. Considerando que a operação de UGs vizinhas àquela que será paralisada interfere no processo de decantação, e que a potência de geração dessas vizinhas também influencia este processo, são necessários métodos que levem em consideração esses fatores, pois alteram o tempo de parada das UGs.

Para alcançar resultados satisfatórios na determinação do tempo de parada necessário, este trabalho realiza uma série de etapas para viabilizar o desenvolvimento de técnicas que possam auxiliar na tomada de decisão pelos operadores da UHE Jirau. Inicialmente deve ser resolvido o problema da grande quantidade de dados relacionados às UGs, coletados e disponibilizados através de sistemas de *hardware* e *software*. Como há uma grande quantidade de dados, um estudo inicial demonstrou a necessidade de realizar etapas de pré-processamento nos dados para que seja possível utilizá-los tanto na identificação dos estados operativos quanto na previsão do tempo de parada das UGs da UHE.

Esta série de etapas de pré-processamento consiste na aplicação de técnicas de *Big Data* e BDA, que visam manipular um grande volume de informações, provenientes de inúmeras fontes de dados, em diversos formatos e com diferentes graus de complexidade para a realização da coleta. O uso de tais técnicas se faz condizente com o cenário existente na UHE, onde existe um alto número de UGs, uma base de dados com informações detalhadas, coletadas de sensores e medidores, além de informações inseridas em sistemas pelos operadores, que foram utilizadas como fonte para o processamento dos dados.

A proposta apresentada nesta tese utiliza Modelos Ocultos de Markov (*Hidden Markov Models* - HMM) e Redes Bayesianas (*Bayesian Networks*) para prever o tempo de parada das UGs para decantação da sujeira acumulada nas grades de proteção. A utilização das referidas técnicas visa propiciar a disponibilização de informações que possibilitem maior confiabilidade na programação de uso das UGs da UHE Jirau, através de uma metodologia de elaboração de modelos utilizando dados reais de operação da UHE Jirau. Para obter uma modelagem que apresente resultados satisfatórios, são necessários dados de entrada confiáveis, formatados de acordo com os requisitos de cada técnica utilizada.

Para viabilizar informações sobre o status de cada uma das UGs, também é proposto neste trabalho uma solução que seja capaz de identificar de modo automático e em tempo real o estado operacional das 50 UGs da UHE Jirau, utilizando nesta tarefa o conceito de Autômatos Finitos Determinísticos (AFD). A identificação do status de cada UG fornece aos operadores da UHE a visão necessária para definir quais UGs estarão disponíveis para a geração energética. Dessa maneira, é possível definir de forma mais consistente qual o volume de geração em determinado instante, fornecendo maior segu-

rança operacional para a UHE. Esse processo de identificação faz uso intenso de inúmeras informações relativas às UGs e seus sistemas adjacentes, e analisar toda essa informação em tempo real sem a realização de um tratamento eficiente nos dados é uma demanda que se demonstrou proibitiva.

Diante do problema da grande quantidade de dados existentes na UHE Jirau, foi idealizada e implementada uma arquitetura capaz de tratar e gerenciar informações através de um conjunto de técnicas de BDA e computação distribuída, realizando a extração, transformação e carregamento das informações, de modo a fornecer informações tratadas e em formatos personalizados. Os dados processados por esta arquitetura foram utilizados para a implementação dos modelos de previsão de tempo de parada propostos neste trabalho, no monitoramento em tempo real para a identificação do estado operacional das UGs e foram disponibilizados aos operadores da UHE através de *dashboards* idealizados e desenvolvidos ao longo do projeto de pesquisa e desenvolvimento (P&D) na qual este trabalho foi desenvolvido. Por fim, os dados processados pela arquitetura também foram utilizados para a integração de diversas fontes de dados da UHE e a entrega de informações em diversos formatos para os inúmeros sistemas existentes, com o objetivo de auxiliar na tomada de decisão pelos operadores.

1.2 Objetivos

O objetivo principal deste trabalho é desenvolver modelos para realizar a previsão do tempo de parada necessário para decantação da sujeira armazenada nas grades de proteção das UGs através de HMM e RBs.

Para alcançar tal objetivo, foram necessárias etapas adicionais, apresentadas a seguir:

Efetuar a identificação dos estados operativos das UGs, utilizando as informações coletadas das unidades e dos sistemas auxiliares, de modo que seja possível fornecer uma visão em tempo real do estado operacional das 50 UGs da UHE Jirau, incluindo o detalhamento dos estados que tornam os equipamentos indisponíveis para uso.

Implementar uma arquitetura para recuperação, tratamento, armazenamento e utilização de grandes quantidades de dados heterogêneos, obtidos de diferentes origens e com variados formatos, utilizando abordagens de sistemas distribuídos para acelerar o processamento das informações, preservando a escalabilidade e performance à medida que o volume de dados aumenta.

Prover meios de recuperação dos dados processados pela arquitetura desenvolvida para viabilizar a construção dos modelos de previsão e para efetuar o monitoramento do estado operacional.

Sistematizar a visualização dos dados através da centralização e sumarização das diversas informações necessárias à operação e planejamento da UHE, em uma plataforma que permita o acesso operacional e gerencial, por meio de conectores de software que sejam adaptáveis às diversas fontes de informação e que possibilitem a visualização imediata dos dados de operação da UHE para aperfeiçoamento do planejamento da operação.

1.3 Questões de Pesquisa

Com base nos desafios apresentados e que motivaram a elaboração deste trabalho, bem como na revisão bibliográfica realizada para direcionar a pesquisa e apresentar o estado da arte relacionado ao tema, foram estabelecidas algumas questões de pesquisa que foram investigadas durante o desenvolvimento deste estudo.

Diante das premissas que (1) a quantidade de informações existentes em UHEs é enorme, com informações obtidas dos mais diversos sistemas, equipamentos e fontes de informação; (2) o cenário de uma UHE que opera a fio d'água, com alto índice de acúmulo de sedimentos e troncos, impõe um dinamismo constante na utilização e limpeza das UGs, o que requer melhorias contínuas no processo de despacho da UHE; (3) em diversas aplicações, o processamento de toda informação disponível na UHE deve ser realizado em um curto intervalo de tempo, sendo necessário selecionar quais as informações mais relevantes para fornecer *insights* e atingir determinado resultado em tempo hábil; (4) para um planejamento de geração mais eficiente, é imprescindível a identificação em tempo real do estados operativos atuais das UGs, e (5) a previsão confiável do tempo de parada necessário para limpeza das grades de proteção, com acúmulo de sujeira proveniente do leito do rio, é de extrema importância para a operação da UHE, as principais questões de pesquisa levantadas, que foram respondidas ao longo deste trabalho são:

- É possível desenvolver uma arquitetura capaz de processar volumes massivos de dados de forma distribuída, permitindo a utilização dos resultados dentro de prazos restritos às demandas operacionais da UHE?
- Essa arquitetura é flexível para acomodar as mudanças e atualizações inerentes ao ambiente de UHEs, bem como o aumento do volume de dados, sem deteriorar o tempo de resposta?
- Através da utilização da arquitetura proposta, é viável extrair informações dos dados brutos que proporcionem melhorias operacionais à UHE em que o trabalho foi aplicado?
- Com as informações extraídas, é possível determinar de forma automática, sistematizada e em tempo real o estado operativo das 50 UGs existentes na UHE?

- É possível desenvolver uma metodologia para realizar a previsão do tempo de parada necessário para decantar a sujeira armazenada nas grades de proteção, considerando ainda os fatores que influenciam na variação desse tempo?

Conforme será apresentado ao longo do trabalho, a arquitetura proposta é capaz de lidar com as tarefas inerentes ao processamento de dados, com limites de tempo impostos pelas rotinas de operação da UHE, possibilitando a utilização das informações resultantes tanto para a identificação dos estados atuais, quanto fornecendo dados de entrada aos modelos de previsão desenvolvidos.

Os modelos de previsão utilizando HMM e RB forneceram resultados satisfatórios, tornando-os uma opção viável para fornecer estimativas de tempo de parada das UGs aos operadores, visando retomar a geração de forma mais eficiente e reduzir o tempo de indisponibilidade na UHE Jirau.

1.4 Metodologia

Neste trabalho é proposta a criação de modelos para a prever qual o tempo de parada necessário para a decantação da sujeira depositada nas grades de previsão das UGs da UHE Jirau. Para a implementação destes modelos, foram utilizadas duas abordagens, uma utilizando HMM e outra utilizando RBs. As técnicas apresentadas são capazes de lidar com incerteza, e apresentam como resultado informações probabilísticas acerca da ocorrência ou não de determinado evento.

A identificação dos estados operativos das UGs foi desenvolvida utilizando uma abordagem com AFD, onde os estados definidos consistem em uma sequência de situações em que a unidade pode estar, e a ocorrência de determinados eventos podem alterar o estado atual. A utilização desta técnica visa possibilitar a apresentação concisa de dados de estado operacional das UGs da UHE Jirau, disponibilizando os resultados em tempo real para auxiliar na tomada de decisão do operador.

Para lidar com a grande quantidade de dados, é proposta a implementação de uma arquitetura para lidar com *Big Data* no contexto de UHEs, utilizando técnicas distribuídas para efetuar o processamento de dados, um conjunto de ferramentas de Extração, Carregamento e Transformação (*Extract - Load - Transform* - ETL), banco de dados distribuído para armazenamento das informações e aplicações de apresentação para exibição das informações resultantes.

A arquitetura proposta foi validada através do uso das informações resultantes em diversas aplicações, mas principalmente nos modelos de previsão, na identificação dos estados operativos, nos *dashboards* desenvolvidos, na disponibilização de dados e integração com outros sistemas existentes na UHE Jirau.

Foi definida uma abordagem experimental para realizar este trabalho, pois esta metodologia é largamente utilizada para avaliar novas soluções para problemas. A metodologia experimental é dividida em duas fases: uma fase exploratória e uma fase de avaliação.

Na primeira fase, conduz-se uma revisão bibliográfica para listar estudos anteriores que abordem e discutam soluções para as questões propostas neste trabalho, incluindo as técnicas e ferramentas geralmente empregadas, bem como as métricas a serem utilizadas para avaliar a qualidade dos resultados obtidos.

Antes da proposição dos modelos de previsão e da arquitetura de processamento de dados apresentados nesta tese, uma pesquisa bibliográfica foi conduzida para investigar modelos de previsão probabilísticos e ferramentas destinadas a lidar com grandes volumes de dados, provenientes de diversas fontes e formatos, incluindo análises dos algoritmos disponíveis para BDA.

Após a fase de pesquisa bibliográfica, durante o desenvolvimento da proposta, foi utilizada uma metodologia iterativa para evolução gradativa da solução. Como toda pesquisa científica, as lacunas e problemas encontrados demandaram novas pesquisas para encontrar soluções inovadoras, dentro de uma abordagem de conceituação e prototipação rápida, necessária ao contexto de UHEs onde este trabalho foi aplicado.

1.5 Estrutura da Tese

A tese foi organizada da seguinte forma: o capítulo 1 apresenta uma introdução às UHEs, as dificuldades e problemas envolvidos em sua operação e quais métodos são propostos neste trabalho para solucionar alguns destes problemas.

O capítulo 2 provê uma visão geral das técnicas utilizadas na organização e processamento de grandes volumes de informações. Soluções envolvendo técnicas de *Big Data* e BDA presentes na literatura são discutidas, bem como questões sobre coleta, armazenamento, comunicação e processamento de dados.

O capítulo 3 apresenta a abordagem utilizada para identificação e monitoramento dos estados operativos das UGs, demonstrando a técnica utilizada em sistemas com eventos discretos. São apresentados detalhes sobre os estados operativos das UGs, bem como as etapas envolvidas na identificação dos estados.

No capítulo 4 é apresentada a teoria sobre modelos gráficos, Cadeias de Markov, Modelos Ocultos de Markov e Redes Bayesianas, e como essas técnicas foram utilizadas neste trabalho para realizar a previsão do tempo de parada das UGs.

O capítulo 5 apresenta o modelo de tratamento e processamento de dados desenvolvido, detalhando a arquitetura proposta para solucionar o problema de gerenciar

grandes quantidades de dados. São apresentadas as etapas realizadas nas fases de coleta, tratamento e processamento dos dados, efetuados pela plataforma proposta, para gerar as informações necessárias para as próximas etapas do trabalho.

Os requisitos para a elaboração dos modelos de previsão do tempo de parada das UGs da UHE Jirau utilizando HMM e RBs são realizadas e apresentadas no capítulo 6, assim como são evidenciados os passos para o desenvolvimento dos modelos.

Os resultados obtidos através da arquitetura de gerenciamento de dados e dos modelos de previsão propostos são apresentados e discutidos no capítulo 7.

As conclusões da realização deste trabalho são apresentadas no capítulo 8, onde também são listados os trabalhos futuros que poderão ser desenvolvidos a partir da proposta apresentada nesta tese.

2 Revisão Bibliográfica

Dada a quantidade de equipamentos existentes em uma usina hidrelétrica, como as UGs e seus componentes adjacentes, os sistemas de resfriamento, os controladores de comportas, os reguladores de pressão, dispositivos atuadores de segurança, proteções do sistema, sensores de temperatura e velocidade, medidores de tensão, sensores de nível do reservatório, reguladores, dentre inúmeros outros dispositivos existentes na UHE, é imprescindível que existam sistemas que viabilizem aos operadores um melhor controle e visualização do status atual desse complexo aparato de equipamentos e dispositivos (GHORBANIAN; DOLATABADI; SIANO, 2019).

Tal necessidade se justifica, por exemplo, pelo fato de que a falha de um sistema de refrigeração que passe despercebida pode ocasionar danos a equipamentos caríssimos, além de ocasionar falhas em cascata em diversos outros mecanismos, que podem inclusive, afetar a segurança dos funcionários e do sistema elétrico como um todo (AMIN, 2008).

A utilização de sistemas PLCs *(Programmable Logic Controller)* para monitoramento e controle de grandes quantidades de dispositivos já é um procedimento difundido e bastante aplicado, não apenas no setor de UHEs, mas também no setor industrial (ORDEAN et al., 2006). Os PLCs atuam diretamente nos equipamentos, provendo a interface de comunicação entre os sensores, atuadores e mecanismos disponibilizados pelos mais diversos fabricantes e os sistemas conhecidos como SCADA *(Supervisory Control And Data Acquisition)*, agindo como o ponto central que possibilita a coleta de informações e ações de controle sobre os equipamentos (VASILIEV; ZEGZHDA; ZEGZHDA, 2016).

Em conjunto, sistemas PLCs e SCADA proporcionam a visualização operacional dos diversos componentes da planta, possibilitando aos funcionários atuar diretamente sobre tais equipamentos, controlando seu modo de utilização. Sistemas SCADA, ou sistemas supervisórios, como também são conhecidos, são utilizados no acompanhamento e visualização de informações, possibilitando o controle e supervisão através da interação com os PLCs, com a finalidade de garantir a eficiência e segurança do processo (KUMAR; SAINI, 2022).

Apesar da utilização de sistemas SCADA ser amplamente adotada, tais sistemas têm a prioridade de requisitar e controlar os dados obtidos dos equipamentos. Mesmo oferecendo ferramentas para criação de alarmes e elaboração de relatórios, esta não é a função principal deste tipo de sistema, o que muitas vezes inviabiliza ou torna mais dispendiosa a elaboração de métodos para a obtenção de informações da maneira desejada (KUMAR; SAINI, 2022).

No caso de UHEs em que há utilização de um sistema SCADA que cumpre efe-

tivamente as tarefas sob sua responsabilidade, ainda se faz necessária uma ferramenta voltada especialmente para a elaboração de relatórios e análises de dados em níveis de operação, supervisão e gerência, mostrando indicadores com base semanal, mensal e até em granularidades de tempo mais específicas.

Um sistema para esta finalidade deve possibilitar a geração de relatórios para análise de dados armazenados em arquivos, bancos de informações e outras fontes de armazenamento, para viabilizar a tomada de decisões. Mesmo que sistemas SCADA disponibilizem mecanismos de exportação de dados e integração com outras fontes de informações, não é uma tarefa trivial elaborar e implementar esse mecanismo nesse tipo de sistema, o que impacta negativamente a integração com os demais sistemas que necessitam de acesso aos dados (KUMAR; SAINI, 2022).

Dado o cenário apresentado, dentre os objetivos deste trabalho está o desenvolvimento de uma arquitetura que seja capaz de lidar com o grande volume de informações coletado dentro do ambiente da UHE, e que provenha extensibilidade e seja robusto para acomodar as mudanças constantes inerentes à operação. A arquitetura deve permitir que os seus utilizadores possam obter as informações que desejam de forma imediata, sem que para isso tenham que acessar diversos sistemas e esperar por horas ou mesmo dias, possibilitando assim que a informação seja usada na tomada de decisões em inúmeras situações dentro da UHE.

2.1 Big Data

O termo *Big Data* refere-se a dados complexos, de grande volume e diversificadas estruturas, cujo gerenciamento e tratamento é difícil ou ineficiente utilizando as aplicações e técnicas tradicionais de processamento e armazenamento. Além disso, as aplicações de visualização de informações não lidam bem com *Big Data*, exigindo uma grande quantidade de servidores e estruturas computacionais para processar paralelamente a maciça fonte de dados (JIANG et al., 2016; LV et al., 2017).

Atualmente, o volume de informações gerado em diversas áreas de conhecimento é enorme. Tais informações são provenientes de sensores, *logs* de aplicações, imagens, mídias sociais, medidores de energia inteligentes, e inúmeras outras fontes. Não é apenas a quantidade de informações que deve ser considerada, mas sim a velocidade com que ela é gerada. É importante destacar outro fator relevante quando se discute sobre *Big Data*: a ausência de padronização na estrutura dos dados provenientes de fontes distintas. (ZHOU; FU; YANG, 2016; SAGIROGLU et al., 2016; SHAQIRI, 2017).

Para gerenciar o armazenamento e processamento de grandes volumes de informações, é necessário o uso de tecnologias capazes de fazê-lo de maneira eficiente, segura e com qualidade. Os gerenciadores de banco de dados relacionais tradicionais falham ao

lidar com dados desestruturados, além de não conseguir alcançar a eficiência necessária para lidar com grandes quantidades de dados em curtos intervalos de tempo (TAHMASSEBPOUR, 2017; GHORBANIAN; DOLATABADI; SIANO, 2019).

Em um contexto histórico, *Big Data* lida com os desafios de manipular grandes conjuntos de dados de modo eficiente, antes que as tecnologias atuais se tornassem disponíveis (GHORBANIAN; DOLATABADI; SIANO, 2019). Embora não haja uma única definição consensual sobre *Big Data*, podendo variar de trabalho para trabalho, algumas definições são comumente encontradas na literatura (SAGIROGLU et al., 2016; GHORBANIAN; DOLATABADI; SIANO, 2019; BHATTARAI et al., 2019). É comum encontrar a relação de *Big Data* com o conceito dos 5V's, e a figura 1 apresenta a informação indicando os 5V's associados: Volume, Velocidade, Variedade, Veracidade e Valor, sendo que alguns trabalhos apresentam variações desses atributos, inserindo valores adicionais ou mesmo ocultando alguns desses atributos.

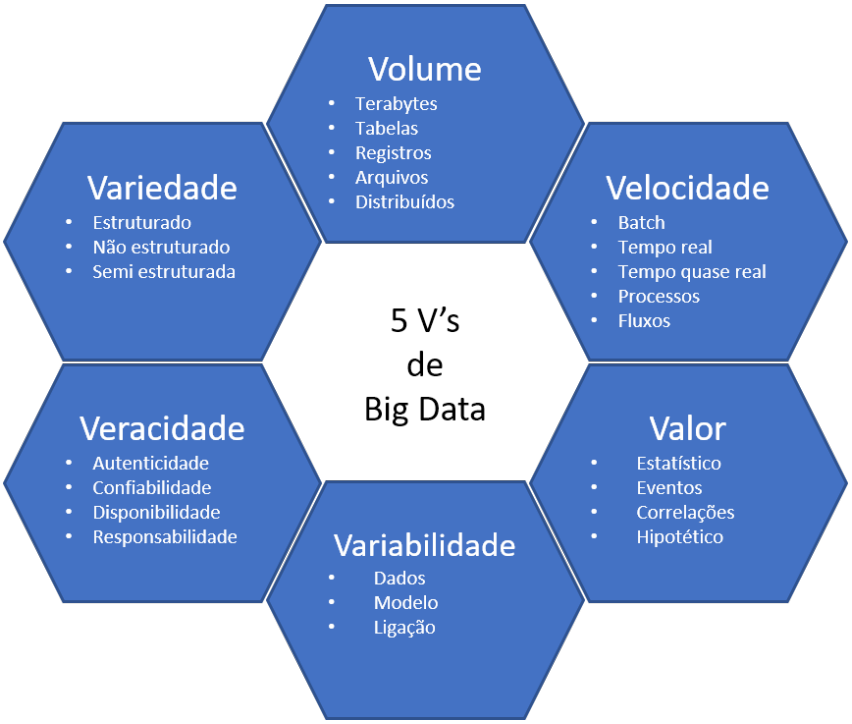

Figura 1 – Big Data - Modelo 5V's. Adaptado de (SHAQIRI, 2017)

O **volume** refere-se ao tamanho dos conjuntos de dados gerados, o número de registros ou armazenamento de grandes volumes de informações. A **velocidade** é a frequência na geração ou transferência de grandes volumes de informações, relacionadas ao tempo de continuidade da coleta, processamento e uso desses dados. A **variedade** está relacionada à heterogeneidade dos tipos de dados, das fontes de informações, aos formatos, à multidimensionalidade e modalidades, como estruturados, semi estruturados e desestruturados. A **veracidade** está relacionada à confiabilidade e qualidade dos dados. Por fim, o **valor** refere-se à extração de percepções, informações e benefícios através do processamento de

dados, assim como a identificação de padrões escondidos através de análises de suas características (BHATTARAI et al., 2019; GHORBANIAN; DOLATABADI; SIANO, 2019).

A quantidade de dados gerada por sistemas relacionados ao setor de energia tem crescido de forma exponencial, e lidar com o processamento desses dados tem se tornado uma preocupação constante desse setor. Gerenciar informações utilizando técnicas tradicionais tem se mostrado ineficiente, uma vez que o tempo necessário se torna impraticável. Isso tem levado ao uso de técnicas de *Big Data*, principalmente em usinas e redes inteligentes (*Smart Grids* - SG). O uso de *Big Data* tem se mostrado eficaz em outras áreas, como sistemas financeiros, biológicos e redes de comunicação sem fio (HE et al., 2015; ZHOU et al., 2016; KUMAR; MOHBEY, 2019; RASHDAN et al., 2019).

Diversos trabalhos tem sido apresentados com a finalidade de contribuir com o estudo e implementação de técnicas de *Big Data*. O trabalho de (HE et al., 2015) apresenta uma arquitetura para SGs baseada na teoria de matrizes aleatórias, que utiliza técnicas estatísticas, especialmente Raio Espectral Médio, para indicar correlação nos dados. Esta técnica evidenciou uma correlação qualitativa com os parâmetros quantitativos de desempenho do sistema. Utilizando a arquitetura proposta, foi possível encontrar o ponto crítico de potência ativa em qualquer nó do barramento, além de possibilitar a detecção de falhas. A arquitetura se mostrou compatível com cálculos usando apenas um pequeno banco de dados regional, assim como em sistemas reais distribuídos de grande escala.

O trabalho apresentado em (WANG et al., 2016) demonstra o uso da técnica *Symbolic Aggregate Approximation* - SAX (NOTARISTEFANO; CHICCO; PIGLIONE, 2013) para redução de dimensionalidade nos dados para simplificar a análise das informações de consumo de energia. O algoritmo *Fast Search and Find of Density Peaks* (RODRIGUEZ; LAIO, 2014) é utilizado para clusterizar os dados de consumo em grupos, em conjunto com uma versão adaptada do algoritmo *K-Means*. Como o trabalho é aplicado em uma quantidade massiva de dados, espalhados em subestações de energia, foi escolhida uma abordagem distribuída ao invés de computação paralelizada, visando reduzir o tráfego de rede. Dada a quantidade de dados analisados, a redução de dimensionalidade em conjunto com a técnica de clusterização obteve resultados satisfatórios na classificação dos perfis de consumo energético, possibilitando agrupar perfis de carga similares e contribuindo para melhores ações de geração de energia.

2.2 Big Data Analytics - BDA

Dentre os inúmeros desafios provenientes da tarefa de lidar com *Big Data*, como o armazenamento, visualização e busca, talvez uma das mais importantes tarefas a ser desempenhada seja a análise dos dados (ZHOU et al., 2016; BHATTARAI et al., 2019). No contexto de *Big Data*, a análise dos dados significa dar valor e sentido às informações

que estão sendo coletadas e armazenadas.

Tradicionalmente, as aplicações apenas geram relatórios a partir das informações, o que na maioria das vezes são apenas listagens de dados, porém, quando se lida com *Big Data*, tal abordagem não é suficiente. São necessários métodos de ETL para tratar os dados originais coletados e então apresentar as informações de modo que estas tenham um significado, importância e contexto (HE et al., 2015).

O conceito de BDA lida com as etapas de processamento e manipulação das informações, visando realizar o tratamento, filtragem e contextualização de dados de modo a possibilitar uma visão mais significativa, aplicando técnicas avançadas de análises em grandes volumes de dados (LV et al., 2017). De modo geral, o termo BDA é usado para descrever técnicas de análises, tais como: aprendizagem de máquina, mineração de dados e texto, visualização de dados, análise estatística, processamento de linguagem natural, dentre outros (OUSSOUS et al., 2018).

Definir de modo preciso o termo *Big Data* e BDA é uma tarefa um tanto complexa, e até mesmo na literatura encontram-se várias definições sobre estes termos (ZHOU et al., 2016; LV et al., 2017; MAHMUD et al., 2020). O objetivo deste trabalho não é exaurir o significado, nem tão pouco definir, por exemplo, a partir de qual quantidade de dados estamos lidando realmente com *Big Data*. O importante é analisar determinada tecnologia ou ferramenta do ponto de vista prático, de forma que seja possível lidar com grandes quantidades de dados, formatos e origens de maneira consistente, de modo a entregar o resultado dentro de um intervalo de tempo aceitável para o negócio em que a análise esteja sendo aplicada, com um alto grau de confiabilidade e com o menor esforço possível.

Arquiteturas e ferramentas para lidar com *Big Data* normalmente utilizam soluções distribuídas em alguma etapa do processo, uma vez que realizar as etapas de coleta, transformação, armazenamento e visualização de informações em uma abordagem centralizada comumente torna-se proibitiva.

2.3 Ferramentas para Big Data

A implementação de técnicas de *Big Data*, de modo a viabilizar a análise de grandes conjuntos de informações, comumente utiliza uma arquitetura com 4 fases, sendo elas 1) Geração, 2) Aquisição 3) Armazenamento e 4) Processamento. É através dessa abordagem que a informação é transformada de dados brutos à conhecimento útil, que poderá ser utilizado para finalidades de controle, supervisão e gerenciamento (KUMAR; MOHBEY, 2019). A seguir são detalhados os passos envolvidos em cada uma das fases mencionadas.

Geração dos dados: refere-se à maneira pela qual as informações são produzidas ou geradas, que podem ser originadas de diversas fontes e de inúmeros sistemas. Os

dados gerados possuem formatos, tipos e características diferentes. Dentre os exemplos de geração de dados, podem ser enumerados os registros de temperatura lidos de um sensor que monitora determinado equipamento, ou mesmo *logs* de uma página web detalhando os passos que um usuário seguiu durante a navegação, e até mesmo registros lançados manualmente por operadores de equipamentos.

Aquisição: uma vez gerados, os dados precisam ser coletados de seu local de origem para o local onde ocorrerá o processamento. Nesta etapa podem ocorrer agregações e sumarizações para reduzir a quantidade de dados transmitidos. Dada a necessidade de trafegar um grande volume de informações, surge a discussão sobre qual meio de comunicação e técnica de transmissão a serem utilizados para evitar ou reduzir congestionamentos e atrasos. A necessidade de transmissão de dados para o local de processamento é exemplificada por medições efetuadas em lavouras por equipamentos de sensoriamento, aferições de nível e de vazão que deve ocorrer próximo ao rio, e leituras de informações de automóveis, que devem acontecer no próprio veículo (MAHMUD et al., 2020).

Armazenamento: é a etapa onde os dados são salvos após a aquisição, preferencialmente em bancos de dados capazes de gerenciar grandes volumes de informações dentro de um intervalo de tempo restrito, sem degradação do desempenho à medida que o volume de informações cresce. É desejável que os dados estejam o mais próximo possível da central de processamento, a fim de diminuir o tráfego de dados na rede de comunicação durante as análises das informações. Em alguns cenários, existe uma restrição de acesso aos dados, o que impede a utilização de algumas abordagens que necessitem que os dados sejam armazenados fora da instituição, como é o caso de serviços de armazenamento na nuvem (AGARWAL; PRIYUSHA, 2015).

Processamento: É a etapa onde os dados são inspecionados, modelados, filtrados e então transformados de tal maneira que se tornem informações que deem subsídio para tomadas de decisão, apresentem uma tendência, possibilitem a inferência de valores futuros, de modo a trazer um maior valor agregado. Trata-se de um dos pontos cruciais em sistemas de *Big Data* (KUMAR; MOHBEY, 2019).

Para realizar as etapas de coleta, aquisição, armazenamento e processamento dos dados, são apresentadas algumas tecnologias e ferramentas, que juntas compõem a arquitetura proposta neste trabalho e que dão subsídio para o uso das informações nas demais etapas do trabalho.

2.3.1 Bancos de Dados NoSQL

Para lidar adequadamente com *Big Data* é necessário utilizar um gerenciador de banco de dados que não tenha restrições rígidas quanto ao esquema de dados utilizado, ou seja, deve ser possível armazenar e processar dados com diversificadas estruturas de

modo simples (AGARWAL; PRIYUSHA, 2015).

Como normalmente não existe uma única fonte de dados, é comum que existam informações provenientes de variados sistemas e com formatos distintos, e com isso, é necessário utilizar gerenciadores de banco de dados flexíveis aos dados, ao invés de adaptar os dados aos gerenciadores relacionais existentes.

Para atingir tal flexibilidade, são utilizados bancos de dados NoSQL (*Not Only Structured Query Language*). Os bancos de dados NoSQL possuem vantagens sobre bancos relacionais tradicionais, uma vez que não necessitam uma definição dos dados a serem inseridos (SONG et al., 2019). Os dados de entrada podem ser alterados a qualquer momento, sem que seja necessário alterar toda a estrutura de armazenamento desses dados.

Dentre as funcionalidades existentes em bancos NoSQL, destaca-se o gerenciamento de múltiplos servidores e a integração automatizada das informações armazenadas de forma distribuída entre eles. A utilização de inúmeros servidores é nativa na maioria dos gerenciadores de banco NoSQL, e visam fornecer um sistema de armazenamento distribuído robusto e eficiente. É comum classificar os bancos NoSQL em 4 grupos distintos (KUMAR; MOHBEY, 2019):

- Documentos: normalmente utilizados para armazenar arquivos JSON (*JavaScript Object Notation*), com um esquema de dados flexível;
- Grafos: armazena as informações utilizados grafos direcionados, onde os nós representam os dados, e as arestas interligando esses nós representam as relações entre os dados;
- Colunar: diferente de um sistema de gerenciamento de banco de dados relacional, onde os dados são armazenados em linhas, os bancos colunares armazenam informações utilizando colunas que se relacionam entre si através de um identificador ou chave;
- Chave-Valor: similar a um armazenamento de dicionários, comumente utilizados em linguagens de programação, como C#, Java e Python, onde uma chave é utilizada para acessar determinado valor ou conjunto de valores. Pelo uso da chave, é possível armazenar valores relacionados, o que torna possível associar um conjunto de temas equivalentes pelo uso de uma única chave.

Alguns gerenciadores de banco NoSQL fazem uso de armazenamento em cache utilizando memória RAM (*Random Access Memory*), assim como ocorre com processadores, a fim de possibilitar respostas rápidas a consultas recentemente efetuadas, trazendo assim um enorme ganho de performance.

Existem vários bancos de dados NoSQL, e cada um deles possui uma característica de armazenamento diferente (TAHMASSEBPOUR, 2017). Para armazenamento no formato chave-valor, pode-se citar *Woldermot*, *Redis* ou *Dynamo*. Dentre os bancos utilizados para armazenamento de documentos, temos o *MongoDB* e *CouchDB*. Existem bancos projetados para armazenar dados com grandes quantidades de colunas, como *Cassandra* e *HBase*. Por fim, banco de dados gráfico *Neo4j* e *InfoGrid* (SAGIROGLU et al., 2016).

O *Apache Cassandra* (Apache Foundation, 2022b) é um dos bancos de dados utilizados para processamento de grandes volumes de dados (SONG et al., 2019). Inicialmente criado pelo *Facebook*, teve seu código aberto para a comunidade em 2008. Seu projeto foi inspirado pelo *DynamoDB*, da *Amazon*, e seu modelo de dados foi baseado no *BigTable* do *Google*. Atualmente o mantenedor desse sistema é a *Apache Foundation*, que através de colaboradores ao redor do mundo contribuem para implementação de melhorias e correções de *bugs*. Dentre as principais características desse sistema de gerenciamento de banco de dados pode-se destacar:

- É um sistema **distribuído**, onde cada integrante do *cluster* possui a mesma função. Não há um nó mestre, e com isso não há um ponto único de falhas, e cada nó pode atender uma solicitação aleatória. Os dados são distribuídos no *cluster*, de modo que cada nó analise dados diferentes;
- Possui alta **escalabilidade**, possibilitando altas taxas de leitura e gravação, mesmo que novos nós sejam adicionados ao *cluster*.
- É **tolerante a falhas**, permitindo que os dados estejam replicados em diversos *data centers*, o que viabiliza a substituição de um nó com falha sem perda de dados e sem tempo de inatividade.
- Viável para **integração com *Hadoop***, possibilitando assim o uso de *MapReduce*, dentre outros módulos do *Apache*, o que oferece suporte a um grande número de tecnologias.
- Por fim, para interagir com os dados armazenados, o *Apache Cassandra* oferece uma interface para consulta de dados, o CQL *(Cassandra Query Language)*, uma alternativa similar ao SQL *(Structured Query Language)*.

Comparado a um sistema de banco de dados relacional comum, o *Apache Cassandra* oferece um sistema de tabelas onde cada linha pode conter um número diferente de colunas. Trata-se de um sistema híbrido que mescla tabelas com conjuntos de chaves-valores. Dessa forma é possível armazenar dados de diferentes formatos, sem a necessidade de padronização (MAHMUD et al., 2020).

2.3.2 Computação Distribuída

Embora os recursos computacionais para armazenamento, memória RAM e processamento tenham evoluído muito nos últimos anos, proporcionando maior poder computacional, a execução de determinadas tarefas utilizando computação centralizada pode se tornar impraticável (ZHOU et al., 2016).

A computação centralizada normalmente envolve computadores clientes e um ou mais servidores que fornecem serviços específicos para esses clientes, e que em alguns casos, trabalham de forma independente uns dos outros. Já a computação distribuída utiliza um conjunto de computadores, comumente conhecido como *cluster*, para trabalhar de forma conjunta para resolver um problema. O fato de participar de um *cluster* não inviabiliza que os computadores que o integram possam ser utilizados em outras tarefas, o que confere a esse tipo de organização uma maior flexibilidade de uso (SANDHU, 2021).

O uso de um *cluster* segue o seguinte procedimento: uma determinada tarefa é dividida em subtarefas por um computador que precisa realizar um serviço, e essa lista de tarefas é enviada para o *cluster*. Pode haver um nó coordenador do *cluster*, que atribui a cada computador integrante do grupo uma ou mais subtarefas, coordena a execução e retorna o resultado final. Cada subtarefa é executada por um integrante, e o resultado é obtido quando todas as subtarefas são concluídas (DEAN; GHEMAWAT, 2008). O uso de um ambiente distribuído tem como objetivo fornecer ao sistema as seguintes características:

Escalabilidade: o sistema deve ser capaz de se comportar sem atrasos ou interrupções à medida que o volume de informações a serem processados aumenta, ou seja, deve ser possível aumentar a carga de trabalho do sistema distribuído sem que isso aumente exponencialmente o tempo de processamento. Aumentar indefinidamente a carga de trabalho sem a contrapartida de recursos computacionais, incluindo a infraestrutura de rede, levará ao colapso do sistema distribuído, uma vez que a carga de trabalho, em algum momento, superaria os recursos de processamento existentes. Sendo assim, um sistema escalável permanece eficiente com o aumento da carga de trabalho juntamente com o respectivo aumento de recursos de *hardware*.

Extensibilidade: devido à quantidade de tecnologias e ferramentas disponíveis atualmente, deve ser possível adicionar novas funcionalidades ao sistema que não foram inicialmente planejadas, de forma simples e prática, sem que isso exija um processo complexo de desenvolvimento. Também é necessário que o sistema seja capaz de acomodar novas necessidades, como é o caso do setor de energia.

Concorrência: uma vez que o volume de informações a serem processadas tende a crescer, é necessário que as tarefas sejam executadas de forma distribuída. Uma característica de sistemas distribuídos está relacionada ao fato de que os recursos, como por

exemplo os dados sendo processados, podem ser acessados simultaneamente por diversos processos que precisam realizar operações sobre os dados. O processo de gerência de concorrência visa garantir que todas as requisições para uso de um recurso compartilhado sejam efetuadas de tal maneira que possibilite que todos os solicitantes sejam atendidos, enquanto garante a consistência dos recursos acessados concorrentemente.

Robustez: o sistema deve se comportar adequadamente sob situações adversas, tais como aumento expressivo no volume de informações em dado instante, falha de alguns componentes do *cluster* e adição e remoção de novos componentes no sistema. Diferentemente de sistemas centralizados, onde uma falha em um servidor, por exemplo, pode interromper a execução de todos os serviços, em sistemas distribuídos as falhas são parciais, ou seja, enquanto alguns componentes falham, o restante do sistema continua funcionando adequadamente.

Transparência: um sistema é considerado transparente quando consegue oferecer aos usuários serviços que estão distribuídos em um *cluster* de modo que, para o usuário, o fato de que os componentes que realizam o serviço estão distribuídos geograficamente não seja percebido. O sistema deve ser visualizado como um todo, ao invés de componentes espalhados em um conjunto de computadores, sem que o usuário tenha que lidar com qualquer tipo de tarefa para localizar os serviços desejados.

2.3.3 Análise de Dados

Além do armazenamento das informações através de algum mecanismo de banco de dados, também são necessárias algumas ferramentas e métodos capazes de analisar grandes quantidades de dados de maneira eficiente. Na literatura são apresentadas algumas ferramentas, utilizadas para aumentar a eficiência no processo de análise de *Big Data* (LV et al., 2017; KUMAR; MOHBEY, 2019; SANDHU, 2021).

Um primeiro tipo de processamento consiste na leitura e análise de dados estáticos, previamente armazenados, e que permitem um tempo maior para apresentação das informações resultantes. Esse tipo de processamento é conhecido como processamento em lote. É possível utilizar técnicas de aprendizagem de máquina para analisar as informações, de acordo com o domínio dos dados existentes. Algumas bibliotecas e ferramentas para efetuar tal procedimento são: *Hadoop*, *H2O*, *Weka* e *Spark MLlib* (LV et al., 2017).

Um segundo tipo de análise, consiste no processamento de fluxo de informações, onde os dados são processados no momento em que são coletados. Dentre as ferramentas disponíveis para efetuar tal tipo de processamento estão: *Spark Streaming*, *Storm* e *Flink* (KUMAR; MOHBEY, 2019).

Não é possível saber antecipadamente qual a infraestrutura de processamento e armazenamento necessários para lidar com um volume de dados muitas vezes desconhecido

a priori. Há a possibilidade de aumentar o poder computacional de sistemas planejados inicialmente para lidar com pequenas quantidades de informações, porém é comum encontrar limitações a partir de determinado ponto, como por exemplo a quantidade máxima de memória RAM ou o número de processadores que um computador pode suportar.

O cenário onde a escalabilidade costuma se aplicar com melhores resultados são ambientes de computadores de grande porte, como servidores e *mainframes* dedicados, o que usualmente é dispendioso, pois exige profissional ou empresa especializada para gerenciar a infraestrutura, tornando essa abordagem inacessível na maioria das ocasiões (JATOTH; GANGADHARAN; FIORE, 2017).

Um cenário alternativo para habilitar o uso de sistemas distribuídos para processamento de *Big Data*, sem que haja a necessidade de contratação ou aquisição de hardware especializado, é utilizar máquinas comuns, com baixo poder computacional, que estejam conectadas em uma rede de comunicação, para atuar em um ambiente distribuído (DALCIN et al., 2011).

O grande problema nesta abordagem de sistemas distribuídos utilizando computadores comuns é o fato de que o *software* desenvolvido para atuar neste ambiente deve ser cuidadosamente projetado para executar os processos de modo paralelo. O desenvolvimento de um *software* distribuído, escalável e robusto, para uma quantidade não conhecida de dados, além de dispendiosa, é difícil e demorada (DALCIN et al., 2011).

2.3.3.1 Processamento em Lote

Atualmente está disponível um *framework* para processamento paralelo em lote, o *MapReduce*. Desenvolvido pelo *Google* para indexação de páginas web, o conceito do *MapReduce* foi exposto no artigo *"MapReduce: simplified data processing on large clusters"*. (DEAN; GHEMAWAT, 2008). Este *framework* abstrai as etapas de processamento, permitindo que o foco seja dado ao problema de tratamento dos dados, e não nas tarefas subjacentes. O *MapReduce* trabalha com uma função de mapeamento (*Map*) que processa pares de chave/valor para gerar um conjunto intermediário de pares de chave/valor, e após esta etapa, aplica uma função de redução (*Reduce*) que une todos os valores intermediários associados a uma mesma chave.

A principal vantagem oferecida pelo *framework* é a abstração das tarefas de paralelização e execução dos processos em um amplo conjunto de computadores comuns de usuário. O *framework* lida com a comunicação entre os computadores envolvidos no processo, com o agendamento da execução dos processos no conjunto de computadores, com o particionamento dos dados de entrada e com as falhas que ocorrerem nas máquinas durante o processo.

O *MapReduce* é capaz de escalar seu dimensionamento para alocar mais compu-

tadores caso seja necessário processar um volume maior de dados. O intuito é reduzir o tempo de processamento, e o sistema de arquivos distribuído possibilita utilizar milhares de nós, e ao invés de mover os dados até o local de processamento, move os processos para que estes sejam executados próximos de onde os dados se encontram.

Em 2005 um *framework* de código aberto chamado *Hadoop* foi desenvolvido por *Doug Cutting* e *Mike Cafarella* (Apache Foundation, 2022a). O *framework Hadoop* é fornecido pela *Apache Foundation*, que também disponibiliza diversas ferramentas de suporte para análises de *Big Data* usando como subsistema interno o *Hadoop*. O suporte para processamento de grandes volumes de informações é uma característica desse *framework*, que também é capaz de lidar com diferentes formatos de dados, através do sistema de arquivos distribuído, o HDFS (*Hadoop Distributed File System*). O ponto crítico do *Hadoop* é a velocidade de processamento, que é comprometida pelo fato deste *framework* lidar com processamento em lote (SANDHU, 2021).

2.3.3.2 Processamento em Tempo Real

O conceito de processamento em tempo real está ligado à análise e processamento de fluxos de dados no momento em que os dados são coletados e se tornam disponíveis. Também é comum encontrar técnicas de armazenamento por intervalos curtos de tempo para análises em tempo quase real.

O processamento de grandes quantidades de dados em tempo real consiste em um grande desafio, uma vez que impõe restrição de tempo para que o resultado da análise seja apresentado. A restrição de tempo se torna mais desafiadora no contexto de *Big Data*, pois nesse ambiente o volume de informações a ser processado é muito grande (TAHMASSEBPOUR, 2017).

Os dados processados comumente são gravados em um tipo de armazenamento de dados analíticos, otimizado para análise e visualização, que ofereça suporte a altos volumes de leituras e gravações. Há também a abordagem de consumo dos dados de modo direto, através de uma camada de análise e relatórios para *Business Intelligence* (BI) (LV et al., 2017).

O processamento em tempo real é eficaz se os resultados obtidos puderem ser analisados de forma rápida, possibilitando a geração de alertas, ou mesmo disponibilizando painéis que apresentem as informações em tempo real. Baseados na tecnologia disponível, como por exemplo o *Hadoop*, alguns softwares e *plugins* foram desenvolvidos para prover ao *framework* os requisitos tecnológicos capazes de processar e responder consultas de modo rápido (SONG et al., 2019).

Dentre os desafios relacionados ao processamento de tempo real, estão o processamento de mensagens, seja coletando essas mensagens de seu local de origem, seja reali-

zando filtragens, agregações e preparando os dados para análises mais complexas. Quando a informação de entrada é originada de fontes diversas, não há uma padronização no formato dos dados, e com isso é necessário preparar cada conjunto de dados para então utilizá-los (SONG et al., 2019).

A principal razão para aplicar técnicas de processamento em tempo real é disponibilizar os resultados e *insights* dentro de um limite específico de tempo, para que seja possível tomar ações imediatas mediante os resultados apresentados (KUMAR; MOHBEY, 2022).

Como exemplo da utilização dos resultados obtidos na tomada de decisão, podemos destacar as ações preventivas que devem ser executadas caso a análise de dados de equipamentos apresente a informação de que a pressão do óleo de determinado dispositivo está prestes a exceder o limite de segurança, sem que os alarmes e sensores estejam acusando tal informação (MAHMUD et al., 2020).

Outro cenário em que os resultados de análises podem auxiliar na tomada de decisão, é o indicativo de que uma manutenção preventiva deve ser executada imediatamente em dado equipamento, para que danos mais graves sejam evitados, exigindo assim que uma equipe seja prontamente enviada ao local para realização do serviço. Esta informação seria de pouca ou nenhuma utilidade depois que o possível dano ao equipamento já tivesse ocorrido.

2.3.4 Apache Spark

O *Apache Spark* é uma ferramenta que utiliza o sistema de arquivos distribuídos do *Hadoop* para armazenar arquivos de entrada e saída. Desenvolvido com base nos conceitos do *Hadoop*, o *Spark* foi criado para analisar grandes conjuntos de dados e também é utilizado para aprendizado de máquina em computadores de nó único ou *clusters*.

Uma grande vantagem na utilização do *Spark* é que há o suporte para uso de várias linguagens de programação na codificação das análises a serem executadas, tais como: Java, Python, Scala e R. Também há suporte para consultas SQL, oferecendo compatibilidade a desenvolvedores que possuem conhecimento e familiaridade com este tipo de consulta, suporte a fluxos de dados, tais como os obtidos dinamicamente de páginas web, *logs* de sensores e câmeras de vigilância, além de oferecer algoritmos de grafos para análises complexas (HU et al., 2014). A completa infraestrutura suportada pelo Apache *Spark* é exibida na figura 2.

Comparado à plataforma do *Hadoop*, o projeto do *Spark* suporta processamento em memória, o que confere velocidades superiores em comparação às abordagens que utilizam apenas processamento de arquivos, que normalmente estão localizados em unidades de armazenamento com velocidades muito inferiores. Outro ponto de destaque está no

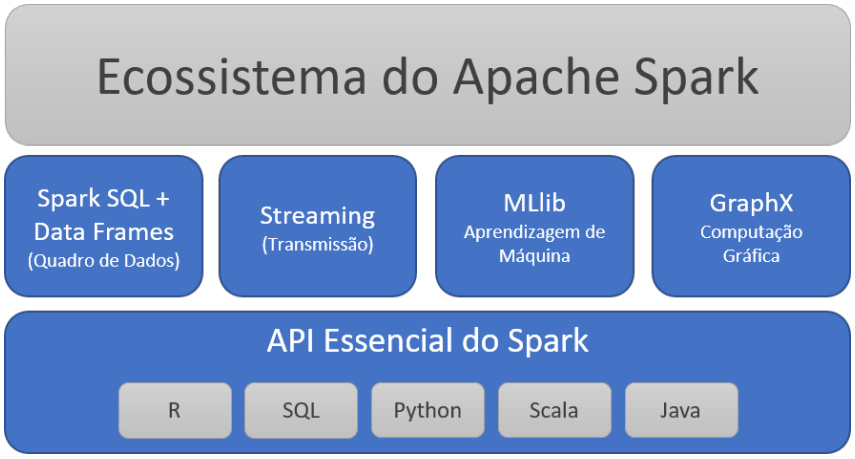

Figura 2 – Ecossistema do *Apache Spark* (DataBricks, 2022)

gerenciamento otimizado do plano de execução das tarefas, definindo de modo automático o número de tarefas a serem alocadas em cada membro do *cluster* para executar o processamento das informações. As tarefas de verificação de falhas de processamento e gerenciamento dos nós ativos no *cluster* também são desempenhadas pelo Apache *Spark* (HU et al., 2014).

O processamento distribuído realizado pelo *Apache Spark*, além de diminuir consideravelmente o tempo gasto durante a execução do processamento em relação ao *MapReduce*, retira do desenvolvedor as tarefas relacionadas à execução de tarefas distribuídas, tais como: gerenciamento de processos, de arquivos, de memória, de sincronização, de alocação de tarefas e detecção e correção de falhas na rede ou em nós do *cluster* (DataBricks, 2022).

Além das linguagens de programação a que oferece suporte, o *cluster* onde o *Apache Spark* comumente está inserido é capaz de trabalhar em conjunto com ferramentas de análises como R e Pandas, permitindo consultas de dados mais complexas, bem como a realização de operações de sumarização e visualização de dados (HU et al., 2014).

Na figura 3 é possível visualizar o suporte a integrações com soluções de código aberto oferecido pelo *Apache Spark*, uma das razões pela qual essa plataforma tem se tornado muito utilizada para processamento de grandes volumes de dados, possibilitando interagir com inúmeras outras soluções já consolidadas, como por exemplo para tratamento de arquivos em bancos como o *MongoDB*, processamento de fluxos de dados de *logs* como o *Kafka*, dados armazenados em bancos de dados relacionais como o *PostgreSQL* e *MySQL*, aplicações que utilizam o sistema de arquivos distribuídos do *Hadoop*, juntamente com o processamento do *MapReduce*, e inúmeras outras aplicações.

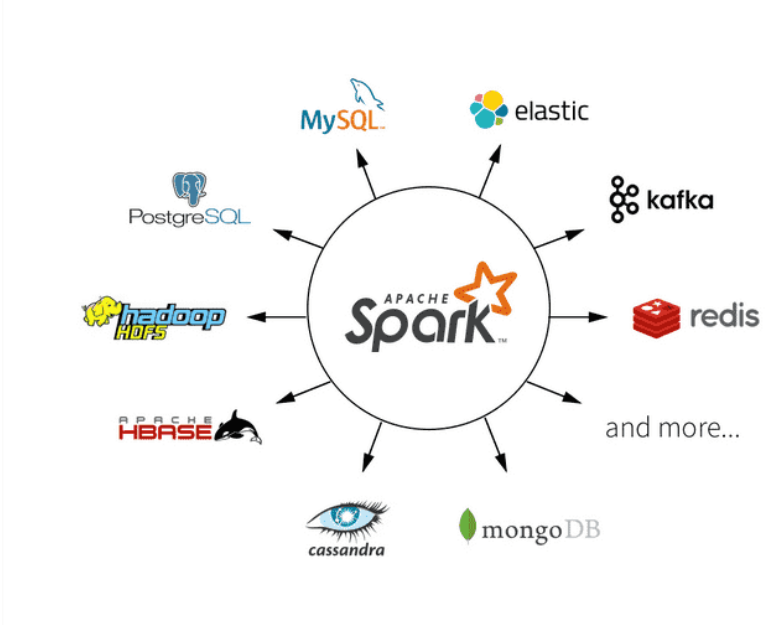

Figura 3 – Integração de Código Aberto do *Apache Spark* (DataBricks, 2022)

2.3.5 Extrair, Transformar e Carregar (*Extract, Transform and Load* - ETL)

A finalidade de um sistema ETL é possibilitar que dados de diversas origens, formatos e tamanhos possam ser processados de modo a serem utilizados por toda a organização de uma maneira mais acessível e uniforme. Dentre as possíveis fontes de dados, é comum não haver uma padronização nos formatos ou mesmo no significado de cada informação disponibilizada em cada sistema que utiliza, armazena e processa essas informações. Além da dificuldade de acesso às bases onde a informação está armazenada, há ainda a tarefa de identificar quais conjuntos de dados são realmente relevantes para processamento e posterior apresentação (CHAUDHARI; MULAY, 2019).

Para a realização de operações de ETL sobre os dados, as etapas comumente realizadas compreendem:

Extração: é a primeira fase do processo, e consiste na obtenção de informações do seu local de origem ou armazenamento. Os dados podem estar em formato cru, como por exemplo *logs*, planilhas, arquivos de texto, dados tabulados ou mesmo serem consumidos através de algum tipo de API *(Application Programming Interface)* que ofereça resultados em XML *(Extensible Markup Language)* ou JSON (SANDHU, 2021). Arquivos do tipo XML são utilizados para troca de informações entre sistemas distintos, auxiliando na transferência de informações através de uma linguagem de marcação com regras, que visa formatar documentos de forma que eles sejam facilmente lidos tanto por humanos quanto por máquinas (BANSAL, 2014). Assim como o XML, arquivos JSON também são utilizados para troca de dados, porém possui uma estrutura mais simples e fácil de ler e escrever, se comparado ao XML (GHORBANIAN; DOLATABADI; SIANO, 2019).

Transformação: nesta fase é efetuada a limpeza nos dados extraídos, removendo valores incorretos ou fora de um intervalo válido, a organização em formatos específicos, seja através de conjuntos de chaves e valores, tabelas sumarizadas ou arquivos, e a validação e o preenchimento de informações faltantes. Outras etapas de transformação realizadas envolvem normalizações, remoção de informações duplicadas, filtragens e agrupamentos. O intuito é preparar os dados, corrigindo erros e informações discrepantes, e aplicar algum tipo de agrupamento e filtragem para que a próxima fase possa ser iniciada (BANSAL, 2014; SANDHU, 2021).

Carregamento: uma vez que os dados estejam em um formato mais apropriado à sua utilização, é necessário disponibilizá-los em um ambiente a partir de onde os dados possam ser consumidos. Esse ambiente pode ser o repositório de dados da empresa, uma solução de BDA, um banco de dados incluído em uma estrutura integrada a demais sistemas, cubos de dados e até mesmo o armazenamento em um *Data Warehouse* (CHAUDHARI; MULAY, 2019).

A figura 4 apresenta os passos realizados nas etapas de ETL.

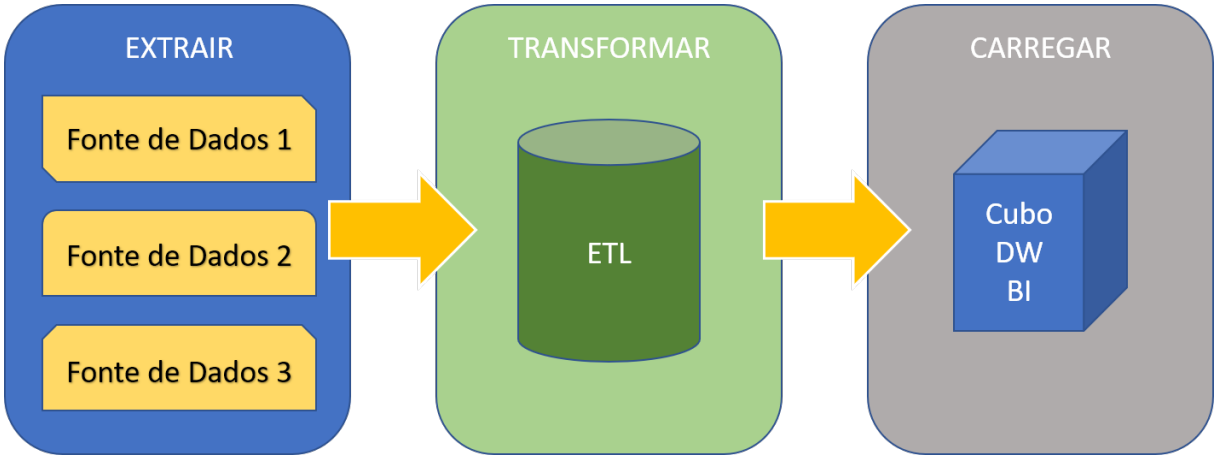

Figura 4 – ETL - Extract, Transform and Load

Existe um grande número de ferramentas que viabilizam a execução do processo de ETL, dentre elas podemos citar: *Pentaho Kettle*, *Microsoft SQL Server Integration Services*, *IBM Infosphere*, *Oracle Warehouse Builder*, *Enterprise Data Integration* e *Talend Open Studio* (BANSAL, 2014; TAHMASSEBPOUR, 2017).

A maioria dessas ferramentas exige um elevado nível de conhecimento técnico para sua utilização, e são aplicações comerciais, sendo necessária a aquisição de licenças com alto custo financeiro. Além disso, apesar dessas soluções disponibilizarem maneiras de acessar inúmeros tipos de bancos de dados e arquivos, ainda é difícil obter métodos de conexão às informações armazenadas por soluções de terceiros, visto que o acesso muitas vezes é restrito ao uso apenas da própria aplicação (GHORBANIAN; DOLATABADI; SIANO, 2019).

Nesses casos, o acesso às informações que a aplicação detém só é possível através dos métodos disponibilizados pela própria aplicação, o que dificulta ou mesmo inviabiliza a integração com ferramentas de terceiros e o uso de módulos de visualização e ETL de outros fornecedores em qualquer etapa do processamento (CHAUDHARI; MULAY, 2019).

2.3.6 Cubo de Dados

A representação de dados com um grande conjunto de atributos não é uma tarefa trivial, uma vez que a visualização de inúmeras variáveis torna difícil a compreensão, além de impor limitações quanto à quantidade de informações que um computador pode processar.

Em um cenário em que as informações estejam armazenadas em um *Data Warehouse* ou central de dados, é necessário que as informações sejam transferidas para o computador ou dispositivo onde se deseja efetuar a visualização. Com isso, há um enorme consumo de recursos computacionais, ocasionando lentidão e grande movimentação de informações dos servidores de dados para os clientes (HU et al., 2014).

Dependendo do contexto da análise dos dados, é possível representá-los utilizando apenas um subconjunto de toda a informação disponível. Para alcançar tal tipo de representação, os cubos de dados são utilizados, de modo a modelar as informações para obter apenas algumas dimensões ou subconjuntos de interesse. As dimensões são também conhecidas como fatos, ou mesmo variáveis que fazem parte de um conjunto completo de dados (BANSAL, 2014).

Para determinada finalidade, é possível utilizar apenas duas ou mais dimensões, seccionando somente os fatos desejados, reduzindo assim a quantidade de dados a serem transferidos, processados e exibidos, o que proporciona maior legibilidade das informações, um processamento mais eficiente e informações resultantes mais direcionadas ao foco desejado (GHORBANIAN; DOLATABADI; SIANO, 2019).

Os cubos de dados eram conhecidos originalmente como vetores multidimensionais, muito conhecidos em diversas linguagens de programação. Dentre as operações possíveis em cubos de dados, destaca-se a extração de subconjuntos, uniões e agregações de dados (SONG et al., 2019).

2.3.7 Big Data no Contexto de Usinas Hidrelétricas

De acordo com (MURDOCK et al., 2020), a geração hidrelétrica representa 15.8% do total de energia produzida no mundo, e 75% de participação no que se refere a fontes renováveis de energia (TURGEON et al., 2021). Embora seja considerada uma fonte de energia renovável, este tipo de geração lida com alguns desafios, como regulações ambi-

entais, restrições de operação e fiscalização por órgãos governamentais, o que dificulta a instalação de novas usinas hidrelétricas. Sendo assim, é importante maximizar o potencial de geração das usinas hidrelétricas existentes, seja através de procedimentos operacionais que resultem na máxima geração de energia possível, seja através da minimização das perdas ocorridas no processo de geração, e através de técnicas que viabilizem uma operação mais eficiente.

Uma maneira de viabilizar uma operação mais eficiente está ligada à disponibilidade de informações que possibilitem uma visão operacional mais ampla e consistente, como por exemplo, uma previsão mais apurada da vazão futura do fluxo de água, um comparativo entre a geração de energia prevista e a geração realizada, informações sobre as operações de manutenção e uma previsão de programações futuras com base no status dos equipamentos, e um monitoramento eficaz dos processos, equipamentos e recursos disponíveis na estrutura das usinas hidrelétricas (RASHDAN et al., 2019).

No contexto energético, a complexidade computacional, a segurança dos dados e também a integração com demais sistemas representam um grande desafio à adoção de *Big Data* no planejamento do sistema (AGARWAL; PRIYUSHA, 2015; ZHOU; FU; YANG, 2016; SHAQIRI, 2017). A utilização de *frameworks* que facilitem a transformação de grandes volumes de informações em ações de planejamento e execução consistem em uma poderosa solução, que em conjunto com técnicas e ferramentas de visualização podem acelerar a transformação desse ambiente através de análises que possam trazer, adicionalmente, retorno financeiro e inovação disruptiva que justifiquem o investimento efetuado para implantação de tais tecnologias (SAGIROGLU et al., 2016; GHORBANIAN; DOLATABADI; SIANO, 2019).

2.3.8 Case - Usina Hidrelétrica de Jirau

A UHE de Jirau está situada no estado de Rondônia, no local denominado Ilha do Padre, a 120 quilômetros ao longo do Rio Madeira, na cidade de Porto Velho. Com a capacidade instalada de 3.750 MW, possui 50 turbinas do tipo Bulbo. A UHE é uma usina à fio d'água e a operação de seu reservatório é realizada conforme curva guia estabelecida na Resolução ANA nº 269/2009 (Agência Nacional de Águas e Saneamento Básico (ANA), 2023). Nos períodos de cheia, a vazão afluente não utilizada para a geração de energia elétrica deve ser liberada através das comportas do vertedouro, ou seja, o recurso hídrico deve ser aproveitado da melhor maneira possível para geração energética, pois a água não pode ser armazenada além do nível máximo permitido para o reservatório (Jirau Energia, 2022).

Jirau é a quarta maior hidrelétrica do Brasil em capacidade instalada e a maior do mundo em número de UGs (Operador Nacional do Sistema Elétrico, 2022a). A elevada quantidade de UGs requer um conjunto complexo de equipamentos, denominados serviços

auxiliares, essenciais para o bom funcionamento, operação e geração de energia na UHE. Esses serviços auxiliares abrangem sistemas de refrigeração, mecanismos de controle de pressão, sistemas de proteção e controle, sensores de temperatura e muito mais. Toda essa imensa estrutura representa desafios operacionais significativos para a UHE.

É importante ressaltar que embora a estrutura da planta seja imensa, a construção da usina foi realizada com cuidados para preservar a biodiversidade e as características naturais do rio, permitindo a passagem das árvores transportadas. A figura 5 fornece uma vista aérea, ilustrando as dimensões físicas da UHE.

Figura 5 – Vista aérea da UHE de Jirau. Margem direita com 28 UGs à esquerda da imagem. Margem esquerda com 22 UGs à direita da imagem (Jirau Energia, 2022)

Uma característica relevante do Rio Madeira, onde a UHE de Jirau está inserida, refere-se à elevada quantidade de sedimentos e troncos transportados em seu curso. Existe na UHE uma infraestrutura para o descarregamento de troncos para a jusante do barramento, composta por linhas de *log booms*, que consiste em uma barreira colocada no rio, projetada para direcionar os troncos para um vertedouro específico. Apesar da existência dessa estrutura, um volume significativo de troncos e sedimentos se acumula nas tomadas d'água das casas de força da usina, o que resulta em restrições de potência decorrentes da obstrução das grades de proteção das UGs.

A UHE Jirau é diretamente afetada por problemas relacionados ao transporte de sedimentos, e o estudo de caso apresentado nesta tese foi realizado nesta usina. A UHE está instalada na bacia do rio Madeira, uma das principais sub-bacias da bacia amazônica, abrangendo uma área de mais de 1,3 milhão de Km^2. A figura 6 apresenta a hidrografia da bacia do rio Madeira, onde é possível observar os principais afluentes desta bacia: os rios Guaporé, Mamoré, Beni, Abunã e Madre de Dios, com as cabeceiras localizada no Brasil, Bolívia e Peru (CARPIO, 2008; CASTRO, 2019).

A bacia amazônica é a maior do mundo, com cerca de 7 milhões de m^2, abrangendo sete países da América do Sul. O território legal da Amazônia está dividido em bacia amazônica ocidental e oriental (FRAPPART et al., 2012).

A bacia amazônica ocidental tem aproximadamente 2.400.000 Km^2. Seus rios mais importantes são os rios Solimões e Madeira, e compartilham características essenciais como grandes dimensões, grandes vazões, baixas declividades e variações significativas de nível e vazão desde secas até períodos de cheias (HEMMING, 1985).

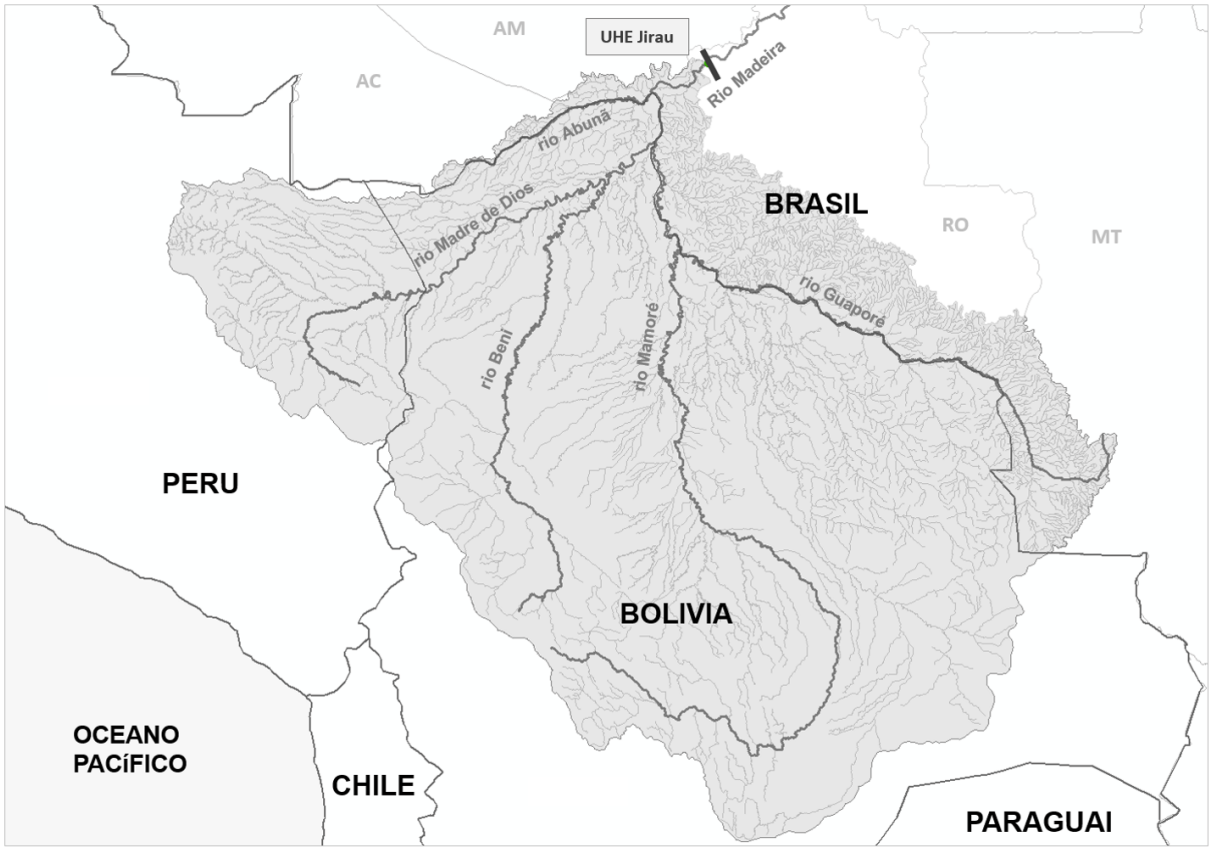

Figura 6 – Hidrografia da bacia do Rio Madeira onde está localizada a UHE Jirau

Seguindo determinação do governo brasileiro, a bacia do rio Madeira possui um minúsculo reservatório de passagem na barragem da UHE com baixa capacidade de regulação. Porém, este reservatório possui alta velocidade de fluxo para transportar muitos materiais, principalmente árvores e sedimentos (SANTOS et al., 2018; SORÍ et al., 2018).

Os volumes de precipitação que o Rio Madeira recebe variam entre 500 e 5.000 mm por ano. A figura 7 apresenta as vazões médias mensais do Rio Madeira (TUCCI, 2007). O governo também regulamenta os níveis de operação da barragem ao longo das diferentes épocas de vazão ao longo do ano.

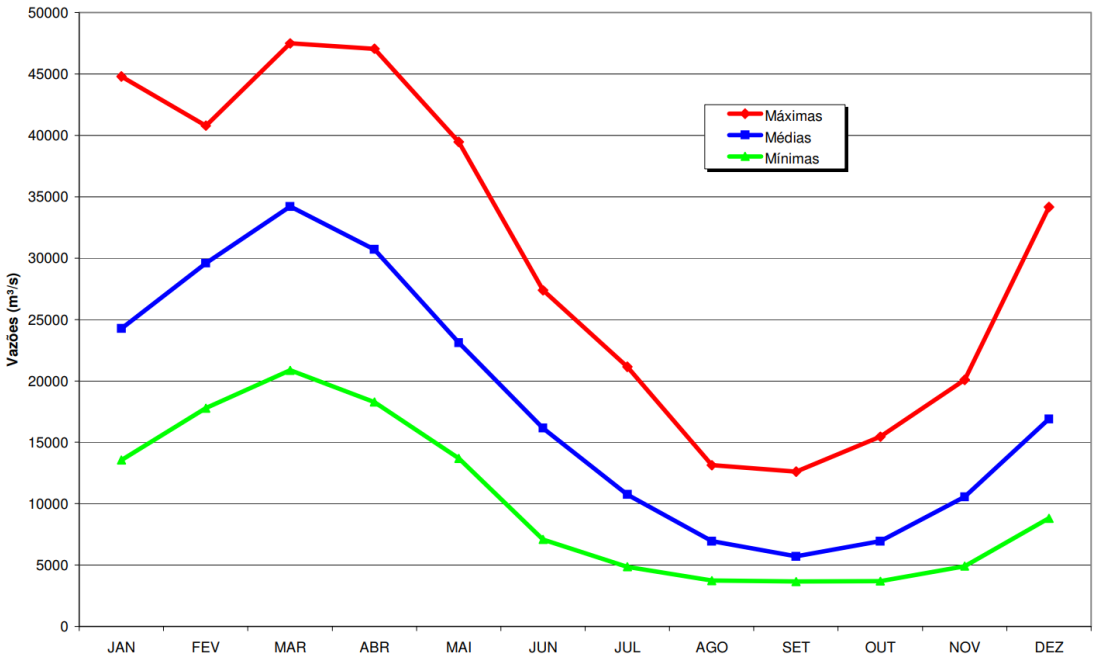

Figura 7 – Fluxos médios mensais do Rio Madeira em Porto Velho

A figura 8 mostra o histórico anual de temperatura de Porto Velho, que varia entre 21 e 34 graus Celsius e raramente é inferior a 18 ou superior a 36 graus.

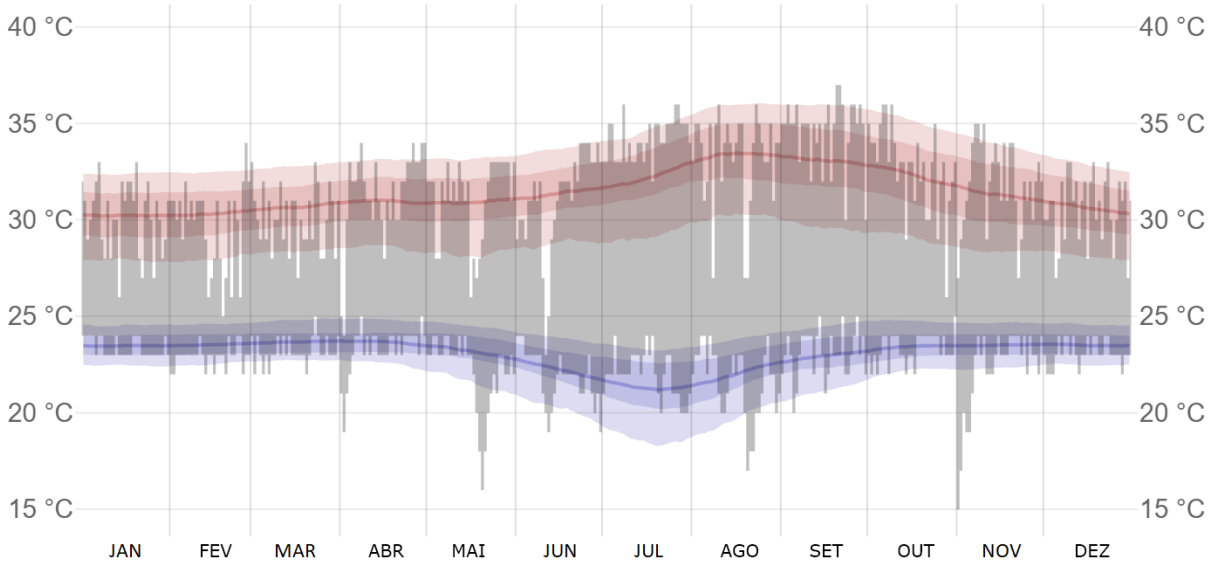

Figura 8 – Histórico Anual de Temperatura de Porto Velho: temperatura na região varia entre 21 e 34 graus Celsius

O problema de entupimento das grades de proteção se agrava no período de cheia do rio, que compreende o intervalo de novembro a maio, onde é mais intenso o transporte

de troncos e sedimentos pelo rio, ocasionando assim maior concentração de sujeira na barragem da UHE, impactando negativamente o desempenho das UGs.

Quando o nível de sujeira alcança o limite máximo, são necessárias atuações para solucionar o problema. O procedimento de limpeza da sujeira é um processo recorrente, porém é necessário o desligamento da unidade geradora para sua realização, o que acarreta perdas financeiras enquanto o equipamento se encontra desligado.

Para um melhor entendimento do funcionamento de uma UHE, algumas informações relacionadas à rotina operacional serão apresentadas. Em UHEs, é comum que o processo de operação seja dividido em três áreas: pré-operação, operação em tempo real e pós-operação. Embora não haja uma separação exata entre quais etapas são realizadas em cada uma dessas áreas, havendo variações em cada usina na organização entre qual área realiza qual etapa, é comum existir a separação de tarefas conforme explicado a seguir (JR et al., 2022).

A pré-operação utiliza previsões de nível de reservatório e medição de nível hidrológico para estimar a quantidade de água disponível para realizar a geração de energia. De acordo com a estimativa do recurso hídrico disponível, é realizada uma estimativa de quanta energia é possível gerar. Diante desta estimativa, é então definido quantas UGs serão necessárias para atender o despacho na usina, liberando as UGs que não serão utilizadas para a realização de manutenções (Operador Nacional do Sistema Elétrico, 2022b).

A operação em tempo real utiliza informações das UGs disponíveis e em manutenção para despachar a quantidade de máquinas necessárias para gerar a quantidade de energia determinada pelos órgãos operadores do setor elétrico, no caso do Brasil, o Operador Nacional do Sistema (ONS). Neste processo, a expertise dos operadores é levada em consideração para decidir quais UGs despachar para atender a demanda energética. As decisões tomadas pela equipe de operação em tempo real são embasadas no planejamento efetuado previamente, porém, há situações não planejadas que requerem ações imediatas, e uma fonte de informação confiável para embasar as decisões é de extrema importância para alcançar uma operação eficiente (Operador Nacional do Sistema Elétrico, 2022a).

A pós-operação é responsável por analisar as ações executadas na usina e comparar com a programação prevista. Dentro desse processo de comparação do previsto versus o realizado, podem ser citadas as ações de análises de geração energética, a realização das manutenções de equipamentos previstas e o volume de recurso hídrico utilizado. A realização das rotinas de comparação viabiliza análises para encontrar possíveis melhorias nos procedimentos realizados na UHE, seja através de mudanças no processo de previsão, utilizando informações de entrada com maior riqueza de detalhes, seja através do esclarecimento dos motivos que impediram a realização de determinada manutenção programada, ou mesmo a execução de manutenções não previstas. Este processo é efetuado na tentativa de identificar quais procedimentos precisam ser acertados para aprimorar e

tornar mais fiel a programação e a execução da operação.

Também é responsável do setor de pós-operação alimentar a pré-operação com informações, além de sugerir melhorias no processo, de modo a aperfeiçoar o planejamento para os próximos dias (BORDIN et al., 2020; Operador Nacional do Sistema Elétrico, 2022b). Um exemplo de informação que pode ser sinalizada pela pós-operação está relacionada às intervenções que estavam agendadas e que não foram realizadas, e que possivelmente terão que ser efetuadas no dia seguinte, tornando indisponível uma UG em um dia diferente do previsto.

O processo realizado por cada uma das áreas citadas pode ser melhorado se os dados históricos da operação da usina forem utilizados para nortear e embasar ações e procedimentos, resultando em um menor custo operacional e maior eficiência. Com o auxílio de informações obtidas através de análises dos dados, cada setor de operação pode atuar de maneira mais justificada e direcionada. Um exemplo dessa melhoria de processo pode ser a pré-operação sugerindo mudanças na escolha de quais UGs utilizar, com a finalidade de operar a usina de modo mais eficiente, minimizando gastos com serviços auxiliares.

O trabalho apresentado nesta tese foi desenvolvido na UHE de Jirau, dentro do contexto de um projeto de Pesquisa e Desenvolvimento (P&D). Tal projeto está direcionado para a otimização da usina, através da minimização das perdas no processo de geração. Dada a grande quantidade de UGs na UHE Jirau, e dado que há o problema de grande acúmulo de troncos e sedimentos trazidos junto com leito do rio, a escolha de quais UGs utilizar no processo de geração para atender a demanda de geração solicitada se torna um problema complexo, dada a grande combinação de máquinas possível . Em muitos cenários, a expertise do operador na sala de controle é o método utilizado durante a seleção das UGs a serem utilizadas.

Para alcançar o objetivo do projeto de P&D, foi realizada a implementação de um sistema de otimização utilizando Processos Decisórios de Markov (*Markov Decision Process - MDP*). De modo resumido, apenas para contextualizar quais requisitos a abordagem utilizando MDP trouxe no âmbito do projeto, é necessário elencar algumas das informações que são utilizadas como entrada para o MDP, para tornar clara a necessidade de uma arquitetura especializada de coleta, armazenamento e processamento de dados. Vale ressaltar que não serão fornecidos detalhes de como a otimização foi efetuada, visto que este não é o foco deste trabalho.

Inicialmente, são selecionados dados de medição do rio para possibilitar os cálculos de valor de vazão, queda líquida e queda bruta disponíveis, e então estimar quanta energia é possível gerar por UG. Em conjunto com essas informações, é necessário estimar o nível de sujeira existente nas grades de proteção, para obter a informação da degradação do desempenho na geração e então chegar ao máximo rendimento alcançável para cada

UG. Outro dado importante utilizado pelo algoritmo refere-se à disponibilidade, sendo necessário detalhar quais UGs estão aptas a serem utilizadas para a geração de energia e quais estão em manutenção, impossibilitando seu uso.

As informações apresentadas acima, necessárias à execução da otimização, representam uma parcela importante dos dados de entrada utilizados pelo MDP, porém inúmeras outras entradas devem ser utilizadas pelo algoritmo. O fato de que para reunir esses dados de entrada é necessário coletar informações espalhadas em diversos sistemas, já demonstra o quão importante foi o planejamento e implementação da arquitetura proposta neste trabalho. Inicialmente, foram identificados alguns desafios no que se refere à utilização dos dados:

É necessário armazenar as informações coletadas pelos PLCs de modo a possibilitar o acesso a esses dados, inclusive com refinamento de informações temporais que permitam detalhar, por exemplo, quando a temperatura de um determinado equipamento alcançou um valor limítrofe, ou mesmo qual era o valor de um relé de proteção em determinado dia e horário, de uma forma que permita uma visualização mais ampla de informações da planta da usina.

Também é um requisito importante definir qual abordagem utilizar para identificar adequadamente os estados operacionais das UGs, de uma forma que seja possível visualizar se o equipamento está apto ou não a ser utilizado em um dado instante. Ainda dentro deste requisito, é preciso evidenciar quais dados são necessários para desenvolver tal abordagem e onde estão localizadas estas informações. Uma vez que os dados utilizados pela abordagem definida estejam disponíveis, e que o procedimento de identificação de estado esteja funcionando, é preciso definir como serão exibidas as informações de estado de cada UG em tempo real, de uma maneira que faça sentido para o operador e que possa ser utilizada para a tomada de decisão.

Inúmeros dados hidrológicos são coletados, armazenados, e posteriormente analisados pelos funcionários da usina, especialistas neste tipo de informação. O desafio é como disponibilizar esse conjunto de dados aos operadores, bem como oferecer métodos para apresentar as informações resultantes das análises, de modo que possam ser utilizadas tanto pelo algoritmo MDP quanto para o planejamento hidrológico para os próximos dias de operação da UHE.

Já no início das etapas de desenvolvimento dos procedimentos de coleta e processamento de dados, para atender todas as propostas apresentadas, incluindo a disponibilização de dados hidrológicos, informações de entrada para o algoritmo de otimização, e apresentação de dados aos operadores, verificou-se que o tempo de resposta não atendia às exigências e requisitos da UHE. A grande quantidade de informações das 50 UGs causou uma lentidão na apresentação dos resultados que inviabilizou a utilização da solução desenvolvida do modo como estava.

Dado este cenário, a pesquisa foi direcionada para solucionar o problema apresentado, e ao longo da revisão bibliográfica foi identificada a necessidade de tratar o problema com soluções de *Big Data*, com a execução de tarefas de forma distribuída. Somente com tecnologias capazes de realizar a manipulação e análise de grandes volumes de dados foi possível apresentar resultados dentro de limites aceitáveis de tempo no contexto da UHE.

A utilização do modelo de execução distribuída exige conhecimentos e habilidades específicas para lidar com os problemas inerentes a este tipo de ambiente. Dentre os detalhes técnicos que precisam ser solucionados quando se utiliza sistemas distribuídos podem ser destacados a execução paralela de tarefas, tolerância a falhas de um ou vários nós, localização de recursos que estão espalhados pela infraestrutura e balanceamento de carga (DEAN; GHEMAWAT, 2008).

O *Apache Spark* lida internamente com a maioria dos problemas apresentados, permitindo que o foco do utilizador da ferramenta seja o de analisar e manipular os dados, e não nos problemas inerentes à sistemas distribuídos. Tal característica é conhecida como transparência de uso (SANDHU, 2021).

O *framework Apache Spark* foi utilizado neste trabalho, dado que possui ampla aplicação em problemas de manipulação de grandes conjuntos de dados, e tem sido implantado em ambientes de *Big Data*. Outra razão para a utilização deste *framework* no projeto refere-se à disponibilidade de integrações com inúmeras outras tecnologias, possibilitando maior acoplamento de novas funcionalidades e acomodação de novas necessidades no decorrer da implementação da solução proposta.

O software *PI System* lida adequadamente com a coleta e armazenamento de informações brutas obtidas dos PLCs, além de disponibilizar um armazenamento de modo temporal desses dados. No contexto do projeto de P&D foi contemplada a aquisição deste software, com o intuito de auxiliar a solucionar os desafios encontrados durante a fase de análise do problema e busca por soluções.

2.3.8.1 PI System

O *PI System* é um sistema de armazenamento de grandes volumes de informações de forma temporal, fornecido pela *AVEVA*. É um portfólio integrado de software que coleta dados dos ativos, sensores e dispositivos instalados na planta da UHE, armazena estes dados com informação de data e hora, possibilitando acesso a dados históricos, permitindo consultas a informações de modo rápido e confiável, de modo a manter as operações críticas funcionando e a equipe gerencial a par da situação atual dos equipamentos da UHE (Aveva OSI Soft, 2022).

O portfólio também disponibiliza métodos para consultar os dados através de softwares de planilha eletrônica, além de oferecer um serviço RESTFul pela qual é possível

consumir os dados através do formato JSON. Além de armazenar os dados da planta coletados pelos PLCs, o *PI System* oferece os meios para analisar os dados e criar fórmulas para avaliar expressões e assim obter um resultado personalizado, através de agrupamentos de valores, verificações lógicas e operações matemáticas.

Os dados que são obtidos de PLCs podem fazer sentido para alguém com conhecimento profundo sobre os equipamentos que estão sendo monitorados, porém podem ser de difícil entendimento para um usuário comum. Para facilitar a visualização dos dados, possibilitar a adição de rótulos, o *PI System* disponibiliza uma ferramenta chamada *PI System Explorer*, pela qual é possível visualizar a infraestrutura de equipamentos e dispositivos de forma centralizada, e com uma organização que faça sentido para pessoas.

A figura 9 oferece uma visão ampla de como o portfólio do *PI System* realiza a coleta, armazenamento e disponibilização dos dados para análises.

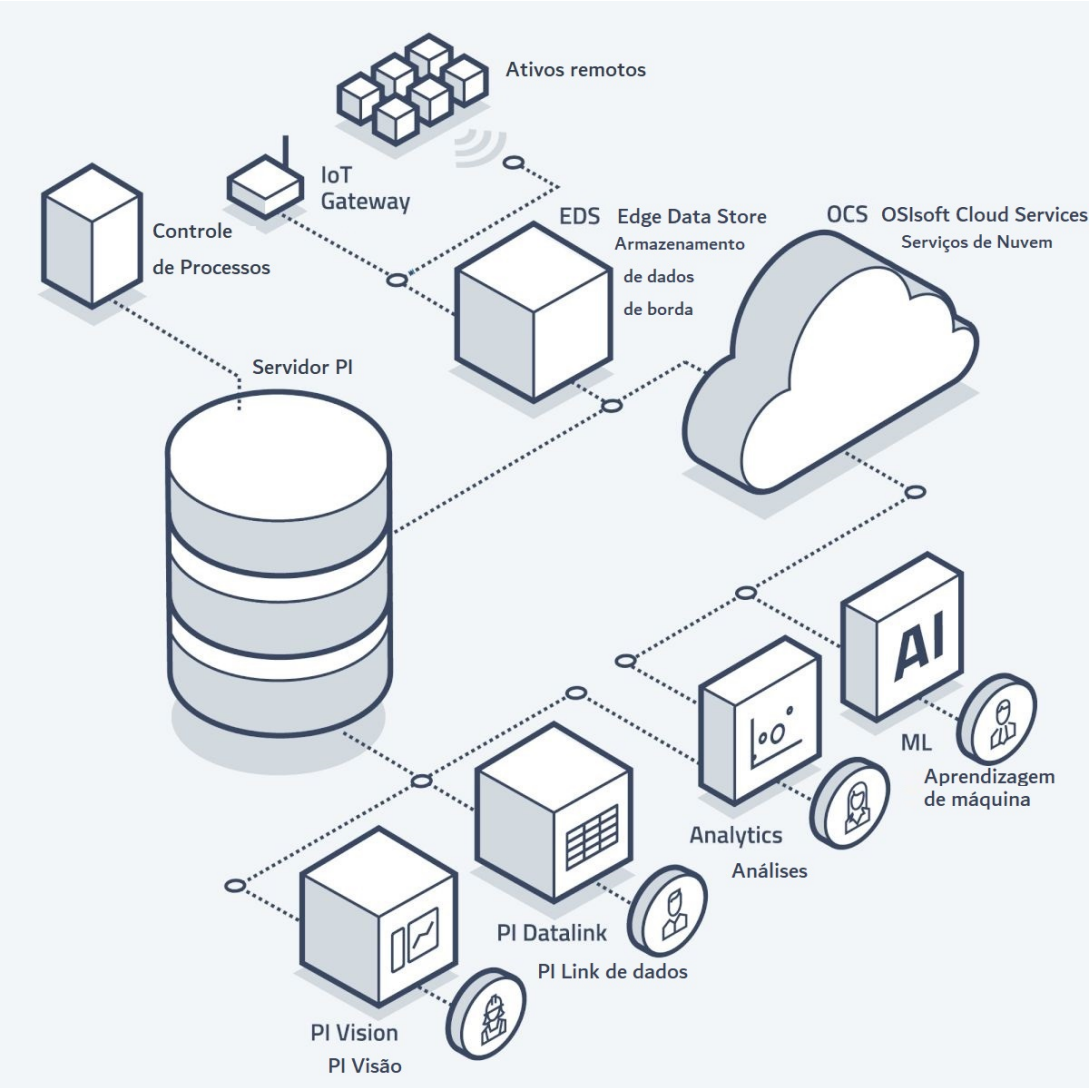

Figura 9 – Arquitetura do *PI System* (Aveva OSI Soft, 2022)

2.3.8.2 Sistema de Acompanhamento de Usinas

Na UHE Jirau é utilizado um sistema para gerenciamento de informações relacionadas às intervenções executadas nos diversos equipamentos da planta. O Sistema de Acompanhamento de Usinas (SAU) gerencia o agendamento de cada intervenção a ser realizada e descreve as atividades executadas em cada procedimento. É também através do SAU que são armazenados os eventos de mudanças de estado operativo, condição operativa e de disponibilidade das UGs.

Os eventos de mudanças de estado operativo são os registros individuais para cada UG, que evidenciam qual o dia e horário determinado estado foi iniciado, com uma descrição adicional associada a esta mudança, e com o registro da intervenção de manutenção associada, caso exista.

Nos casos onde as UGs não conseguem operar na capacidade total, devido a perdas de potência pelo acúmulo de sedimentos nas grades de proteção, são registrados no SAU os valores de perda operativa de cada UG, detalhando o dia e horário de início da restrição, de modo a prover maior detalhamento das ocorrências.

A disponibilidade das UGs é armazenada nesse sistema através de entradas manuais fornecidas pelos operadores, o que exige a consulta da situação de relés, disjuntores, reguladores de velocidade e outras informações em outros sistemas, o que demanda um tempo considerável dos operadores e são possíveis pontos de falha.

Todos os eventos relacionados às mudanças de estado operativo devem ser notificados ao ONS. Existem dois sistemas envolvidos nesta etapa de notificação, sendo um o sistema utilizado pela UHE, e o outro pelo ONS. O SAU armazena e gerencia as informações na UHE, e o ONS utiliza o Sistema de Apuração de Mudanças de Estados Operativos (SAMUG). É através deste sistema que o ONS gerencia as atividades de apuração dos eventos de mudanças de estado operativo.

Para registro e gerenciamento referente às ocorrências de intervenções, o ONS utiliza o Sistema de Gestão de Intervenções (SGI), que visa centralizar e padronizar as solicitações dos diferentes agentes de operação para a realização de intervenções, estabelecendo prioridades que garantam a integridade dos equipamentos e o menor risco para o sistema interligado nacional. No caso da UHE Jirau, o gerenciamento das intervenções é efetuado também pelo SAU, através de entradas manuais efetuadas pelos operadores.

Os procedimentos específicos relacionados ao SAMUG e SGI são detalhados em documentos disponíveis no site do ONS (Operador Nacional do Sistema Elétrico, 2022a; Operador Nacional do Sistema Elétrico, 2022b).

2.4 Otimização de Usinas Hidrelétricas

Um dos objetivos finais que justifica o desenvolvimento da arquitetura de análise de dados proposta é otimizar a operação de UHEs. Para executar essa otimização, diversos desafios precisam ser enfrentados, desde utilizar de forma mais produtiva a vazão de água disponível no rio, através de um despacho que aumente o rendimento das UGs e consequentemente incremente a energia gerada, até melhorias nos processos de planejamento de operação da UHE para os dias seguintes (HUANG et al., 2014).

Um dos planejamentos que impactam na operação de UHEs é o planejamento hidrológico, que consiste em estimar o potencial hídrico que estará disponível para a geração energética, utilizando para tal estimativa dados climáticos, medições efetuadas ao longo do rio através de sensores e a execução de cálculos utilizando dados históricos e a expertise dos operadores (HONG; WHITE, 2009).

Como já mencionado, é comum encontrar uma separação entre os processos envolvidos na operação. A pré-operação, tempo real e pós operação são os três macro processos que, juntos, representam a operação de geração energética em uma UHE. A pré-operação pode ser dividida em programação de longo prazo, médio prazo, curto prazo e em tempo real, respectivamente *long-term*, *medium-term*, *short-term* e *real-time* (BORDIN et al., 2020).

A otimização de UHEs pode ser realizada abordando pontos importantes que representam desafios à operação, e a literatura apresenta diversas metodologias para solucionar os problemas encontrados. Em (CHENG et al., 2015) é apresentada uma solução utilizando Redes Neurais Artificiais (RNA) combinadas à *Quantum-Based Particle Swarm Optimization* (QBPSO) com a finalidade de realizar a previsão de vazão diária, que tem grande impacto no controle da operação de UHEs. Utilizando dados de vazão diária do reservatório do rio, a metodologia foi avaliada e demonstrou uma precisão melhor do que o modelo básico da RNA, enquanto a QBPSO se mostrou eficaz na seleção de parâmetros a serem utilizados.

No trabalho apresentado em (LI et al., 2014), a previsão de geração energética para UHEs pequenas é feita utilizando *Support Vector Machine* (SVM). A escolha por esta abordagem foi baseada na eficácia do método em resolver problemas de reconhecimento de padrões em pequenas amostras, não lineares e de alta dimensão. Para auxiliar na seleção dos parâmetros a serem utilizados pelo modelo de predição, foi escolhido o Algoritmo Genético (AG).

A análise e previsão de entrada de vazão e sedimentos nos reservatórios, utilizando técnicas estatísticas de análises e modelo *Autoregressive Integrated Moving Average* (ARIMA) é realizada no trabalho apresentado em (HAO; QIU; LI, 2017). A verificação da eficácia da proposta é efetuada no Reservatório das Três Gargantas *(Three Gorges Re-*

servoir - TGP), um dos maiores projetos de complexos hidrelétricos do mundo, localizado na China. Foram utilizadas séries hidrológicas de média mensal do período entre 2003 e 2010 para a construção dos modelos, e a análise dos coeficientes de autocorrelação demonstram que o modelo construído consegue prever com alta precisão a vazão e a entrada de sedimentos no complexo da UHE.

A previsão da vazão afluente também tem sido foco de pesquisas, a fim de proporcionar dados mais fidedignos durante a programação da operação e geração energética. O trabalho apresentado em (AHMAD; HOSSAIN, 2019) realiza a previsão de vazão afluente para os próximos sete dias, utilizando RNA de três camadas. Como dados de entrada para o algoritmo foram utilizadas previsões meteorológicas de curto prazo, além de variáveis hidrológicas históricas. A técnica foi validada em 23 barragens dos Estados Unidos, que possuem características hidrológicas e climáticas diversificadas, e demonstrou a possibilidade de maximização da geração através da previsão afluente de maneira eficiente através do algoritmo proposto.

A previsão da energia a ser gerada a curto e longo prazo é realizada no trabalho apresentado em (CERIBASI; CALISKAN, 2019). Utilizando RNAs, os autores realizam a estimativa de geração energética para duas UHEs localizadas na Turquia, utilizando como dados de entrada a energia gerada diária, a queda líquida média e a vazão média diária, obtidos da Diretoria de Operação de Usinas Hidrelétricas da Turquia.

Um modelo não linear auto regressivo em conjunto com RNAs recorrentes é utilizado em (BANIHABIB; BANDARI; PERALTA, 2019) para prever a entrada diária do reservatório por longos períodos, além do intervalo de uma semana. Através de variações dos parâmetros da rede neural, como a função de ativação, o número de entradas e a quantidade de neurônios por camada oculta, foram testadas aproximadamente 1600 alternativas. A precisão do modelo proposto foi comparada com a do modelo ARIMA linear auto regressivo convencional, e os resultados demonstram que o erro quadrático médio da técnica proposta foi cerca de 20% menor do que a do modelo ARIMA.

2.5 Modelos Ocultos de Markov - *Hidden Markov Models (HMM)*

O HMM é um processo estocástico no qual os estados que representam o sistema modelado estão ocultos ou não diretamente observáveis, mas podem ser observados através da sequência de símbolos produzidos como uma sequência de um processo estocástico subjacente. Este tipo de processo é uma técnica comumente usada para lidar com a incerteza. É um modelo probabilístico na qual algum sistema é modelado como um processo de Markov com estados ocultos que não são explicitamente visíveis (RABINER, 1989; STAMP, 2004).

A seguir é apresentada uma revisão apresentando trabalhos que utilizam HMM

para resolver problemas relacionados à produção de energia e seus problemas associados.

A integração massiva de veículos elétricos (*Plug-in Electric Vehicles* - PEVs) nas redes de distribuição de energia afetam diretamente os processos de planejamento, controle e operação. Para contribuir para a compreensão das necessidades de energia desse tipo de veículo, o trabalho de (SUN; YANG; YAN, 2017) apresenta uma abordagem analítica para modelar comportamentos de viagem utilizando PEVs e demandas de carregamento. A simulação de Monte Carlo foi empregada, considerando os propósitos de viagem temporal e estado de carga dos veículos. O modelo de Markov e o HMM formularam a correlação probabilística entre múltiplos estados de PEV e faixas de estado de carga. A técnica foi testada usando uma rede de teste *IEEE 53-bus* com dados de campo, com resultados que demonstram os benefícios da modelagem proposta.

Um método alternativo para classificação de falhas é proposto em (FREIRE et al., 2019), onde um algoritmo HMM é usado para processar sinais elétricos em séries temporais multivariadas. No trabalho é apresentada uma análise comparativa entre a técnica proposta e RNAs, SVM, *K-Nearest Neighbors* e *Random Forest* (RF), e os resultados mostraram que usando uma significância de $\alpha = 5\%$, apenas a RNA e os classificadores utilizando RF oferecem um resultado próximo ao alcançado pelo algoritmo HMM. Outro benefício do algoritmo apresentado é a redução dos custos computacionais em mais de 90% do tempo de processamento em comparação com outros classificadores.

Os sistemas elétricos relacionados às SGs sofrem de problemas de ilhamento, que ocorre quando a rede está desconectada do restante do sistema elétrico. Uma abordagem de algoritmo baseada em HMM é proposta em (KUMAR; BHOWMIK, 2019) para prever a probabilidade de eventos de ilhamento. O processo subjacente mapeia casos padrão ou defeituosos como uma sequência de estados. Embora esses estados não sejam diretamente observáveis, eles seguem um padrão, então a utilização de HMM pode ajudar a detectar esses estados. Esta técnica depende das medições fasoriais (*Phasor Measurements*) obtidas da rede inteligente. Os testes da técnica proposta foram feitos em um sistema de barramento *IEEE 9*, e a análise estatística foi feita para os parâmetros do HMM. Uma RNA treinada é usada para fornecer probabilidades de emissão do HMM, e a previsão de ilhamento é feita usando a probabilidade posterior.

O diagnóstico de falhas de transformadores de potência imersos em óleo usando a análise de gases dissolvidos é um processo comumente utilizado. No trabalho de (JIANG et al., 2019) esta técnica é usada com HMMs para estimar o estado de saúde de transformadores de potência para inferir falhas de operação. Além disso, é proposta uma técnica de predição dinâmica de falhas onde um Modelo de Mistura Gaussiana (*Gaussian Mixture Model*) é usado como um método de agrupamento para extrair características de estado de saúde utilizando para tal um conjunto de dados com 1600 dias de operação. A probabilidade de transição no HMM foi calculada e analisada para relacionar diferen-

tes estados de saúde dos transformadores. Os resultados mostraram a eficácia da solução proposta em fornecer previsão de falhas em uma operação baseada em condições.

Um Monitoramento de Carga Não Intrusiva (*Non Intrusive Load Monitoring - NILM*) é uma técnica usada para identificar corretamente o consumo de dispositivos em um nível desagregado, sem que seja necessária a instalação de equipamentos de medição individuais nos aparelhos. No trabalho de (KONG et al., 2016), um *framework* usando HMM hierárquico é proposto para modelar eletrodomésticos e antecipar características de carga em níveis de baixa tensão e perfis distintos de consumo de energia em dispositivos que possuem múltiplos modos de operação embutidos.

Modelos foram construídos usando uma representação de Rede Bayesiana Dinâmica (*Dynamic Bayesian Network*). A abordagem de maximização da expectativa (*Expectation Maximization*) usando o algoritmo *Forward-Backward* foi aplicada no processo de ajuste do HMM. Testes relacionados à estimativa de desagregação de energia mostraram que a solução proposta usando HMM e uma Rede Bayesiana Dinâmica podem efetivamente lidar com a modelagem de aparelhos com múltiplos modos funcionais.

Uma modelagem de previsão de preços de eletricidade é proposta em (GONZÁLEZ; ROQUE; GARCÍA-GONZÁLEZ, 2005). A técnica é baseada no *Input-Output* HMM e considera a incerteza de algumas variáveis envolvidas, como o comportamento dos concorrentes no mercado de energia, disponibilidade de fontes de energia, afluências hídricas, demanda de energia do sistema e custos relacionados. Os estados do mercado são modelados como estados ocultos e uma matriz de transição de probabilidade condicional é usada para estimar probabilidades quando uma nova sessão de mercado é aberta. Por fim, é feita uma revisão dos modelos de preços de energia elétrica, e trabalhos relacionados utilizando cada tipo de modelo são apresentados, apontando os pontos fortes e fracos de cada trabalho.

Outro trabalho que usa HMM é proposto em (MURTHY et al., 2014). A análise de confiabilidade das unidades de medição fasorial é apresentada. Segundo os autores, a metodologia proposta pode calcular a probabilidade transitória, o que leva a um melhor sistema de monitoramento durante estados transitórios. Essa capacidade permite iniciativas de restauração rápidas e adequadas, fornecendo um método confiável para operar, monitorar e controlar sistemas de medição de grandes áreas.

A penetração de fontes de energia renováveis intermitentes na rede, tais como a energia solar e a eólica, traz novos desafios à operação dos sistemas elétricos. Para enfrentar esses desafios, no trabalho de (BHAUMIK et al., 2018) a potência de um parque eólico é modelada com HMMs discretos, usando dados de medição de várias turbinas, onde os parâmetros são inferidos a partir dos dados e os modelos são usados para estimar a saída de potência agregada individual e total de várias turbinas. A modelagem captura as dependências entre a produção das diferentes turbinas e evidencia que os HMMs podem

reproduzir características essenciais da produção de energia dos parques eólicos.

O trabalho de (ZOHREVAND et al., 2016) propõe uma abordagem para análise de situação e detecção de anomalias usando uma hierarquia de modelos semi-Markov ocultos. A metodologia utilizada modela o comportamento esperado do sistema, para posteriormente detectar anomalias contextuais em sistemas SCADA. O trabalho visa prever e prevenir o risco de ataques que possam interromper ou danificar as estruturas dos sistemas de abastecimento de água ou rede elétrica.

2.6 Redes Bayesianas

RB é uma técnica utilizada para lidar com problemas de incerteza em cenários com aleatoriedade, indeterminismo ou falta de previsibilidade (PEARL, 1988). As RBs são também conhecidas como redes de Bayes, redes de crenças ou redes de decisão.

As RBs têm sido aplicadas em diversas áreas para resolver muitos tipos de problemas, entre eles reconhecimento de fala (ZWEIG; RUSSELL, 1998), sistemas médicos especialistas (SPIEGELHALTER; FRANKLIN; BULL, 2013), avaliação de risco de transporte marítimo (BAKSH et al., 2018), avaliação de risco de descarte de lixo nuclear (LEE; LEE, 2006), sistemas de tomada de decisão baseado em sistemas de informações geográficas (*Geographic Information Systems* - GIS) (STASSOPOULOU; PETROU; KITTLER, 1998) e muitas outras áreas de domínio.

Especificamente nos problemas associados à geração de energia e nas tarefas envolvidas neste processo, alguns trabalhos apresentam abordagens para questões relacionadas à manutenção (LEONI et al., 2019), apoio à decisão de partes interessadas *(stakeholders)* (BARTON et al., 2012), gestão de bacias hidrográficas (BORSUK et al., 2001) e detecção de falhas de usina solar (COLEMAN; ZALEWSKI, 2011).

O trabalho de (CARITA et al., 2013) usa uma RB para detectar falhas em transformadores de potência. A solução utiliza dados históricos no processo de aprendizado para analisar gases dissolvidos em óleo, usando as razões de concentração de gases específicos para identificar deterioração normal e falhas elétricas e térmicas. Comparada à dados da literatura, a solução utilizando RB apresentou alto grau de confiabilidade.

Em (FAHIMAN et al., 2018), uma solução utilizando RB dinâmica é proposta para prever o tamanho da reserva de geração para ambientes de fontes renováveis de energia. A técnica considera a disponibilidade de capacidade dos geradores convencionais, as condições climáticas e os preços de mercado. Além disso, o trabalho propõe uma nova métrica dinâmica para calcular o nível de confiabilidade da rede elétrica, para fornecer uma ferramenta estocástica de suporte à decisão em tempo real. A técnica proposta foi validada usando sete anos de dados históricos fornecidos pelo Operador do Mercado de

Energia Australiano (*Australian Energy Market Operator*). Os resultados mostraram que o risco de redução involuntária de carga pode ser previsto com mais precisão.

Um importante trabalho é apresentado em (YONGLI; LIMIN; JINLING, 2006), onde três modelos baseados em RBs são propostos para estimar a seção defeituosa de um sistema de transmissão de potência. Os modelos são usados dentro de uma área de blecaute para testar se um componente está com defeito. Segundo os autores, o modelo é flexível, pois pode lidar com dados e conhecimentos incertos ou incompletos sobre o diagnóstico do sistema de potência. O modelo usa um algoritmo de retropropagação *(backpropagation)* de erro semelhante ao usado em RNAs. A definição dos priores do modelo requer conhecimento dos especialistas do domínio e modelagem da estrutura da rede.

Outro trabalho que usa RBs e uma abordagem baseada em agente é proposto em (DEHGHANPOUR et al., 2016). Um modelo baseado em agente é apresentado para resolver o problema de licitação estratégica de curto prazo no *pool* de energia de uma empresa de geração. Usando informações públicas incompletas para estimar estratégias de lances ideais, cada agente executa um modelo probabilístico usando RBs dinâmicas, que usa um algoritmo de aprendizado online para treinar o modelo e avaliar a função de lances ótimos para inferir corretamente o estado futuro do mercado. O modelo foi testado no *MATLAB* em duas escalas de tempo diferentes: hora à frente e dia à frente. De acordo com os resultados, os agentes previram o equilíbrio de mercado com antecedência com erros aceitáveis usando dados de informações incompletas.

Um método utilizando RBs com estimativa de probabilidade condicional imprecisa foi usado em (ZHAO et al., 2020) para prever eventos de rampa de geração de energia eólica. O método utiliza o *Maximum Weight Spanning Tree*, um método de busca guloso para ajustar os dados observados com o maior grau, e uma versão modificada do modelo de *Dirichlet* para estimar os parâmetros da rede. Dadas as condições meteorológicas, a solução proposta visa detectar a possibilidade de um evento aleatório de rampa, quantificando a incerteza do evento. Os testes são feitos em um parque eólico real com dados de operação de três anos, mostrando a eficácia dos métodos propostos.

Um sistema de controle usando RBs e outras tecnologias, como RNAs e um sistema Multiagente, é apresentado em (OVIEDO et al., 2014) para apoiar e melhorar o controle automático de usinas solares. O modelo utilizando RB exibe os valores probabilísticos para auxiliar os operadores nas decisões finais relacionadas ao controle remoto da usina solar. Segundo os autores, o sistema resultante fornece uma solução otimizada por meio de tecnologias de inteligência artificial distribuída em sistemas de controle industrial para instalações baseadas em fontes de energia solar fotovoltaica.

O trabalho de (HAO; QIU; LI, 2017) usa abordagens estatísticas para analisar características de escoamento e sedimentos. Os autores usaram a análise de anomalia acumulada, o método de agrupamento ordenado por *Fisher* e a análise espectral de máxima

entropia. Neste trabalho, séries hidrológicas de várias décadas são utilizadas para estudar as variações e prever vazões e cargas sedimentares, tendo como objeto de estudo o TGP na China. A análise de anomalia cumulativa foi usada para estimar a tendência geral de escoamento e sedimento no TGP, e o método de agrupamento ordenado de *Fisher* foi utilizado no estudo para analisar variações abruptas na vazão e carga sedimentar. O modelo *ARIMA* foi usado para construir o modelo de previsão sobre o escoamento médio mensal e o influxo de sedimentos. Os resultados deste trabalho indicaram uma tendência decrescente tanto para o escoamento superficial quanto para o sedimento. Além disso, há pontos de salto nas mudanças de escoamento e sedimentos em 1991 e 2001, respectivamente.

2.7 Análise e Limpeza de Sedimentos e Similares utilizando Técnicas Computacionais

Este trabalho utiliza HMM e RBs para prever o tempo de parada para decantação de sedimentos. Após realizar uma extensa pesquisa bibliográfica, não foram encontrados trabalhos relacionados que lidem diretamente com o problema de acúmulo de sedimento no contexto de UHEs.

Na tabela 1 são apresentados artigos relacionados para evidenciar que existem trabalhos que lidam com limpeza de sedimentos e similares, mas nenhum dos trabalhos é aplicado em usinas hidrelétricas, especificamente para o problema de acúmulo de sedimentos nas grades de proteção.

Tabela 1 – Articles Analysis

ID	Ref	Jornal	Técnicas Usadas
1	(SIMEONE et al., 2020)	Sensors	Processamento de Sinais e Regressão usando ML
2	(HAO; QIU; LI, 2017)	Water	ARIMA e análise espectral de entropia máxima
3	(VERAJAGADHESWA et al., 2022)	Expert Systems Applications	Deep Learning, Redes de Classificação e Regressão usando ML
4	(LE et al., 2021)	Ocean Engineering	Reinforcement learning e Rede Neural Convolucional Profunda
5	(YIN et al., 2020)	Sensors	Rede Neural Convolucional Profunda
6	(RAMALINGAM et al., 2018)	Applied Sciences	Deep Convolutional Neural Network e SVM
7	(ESCRIG et al., 2020)	Food Control	Medições Ultrassônicas e ML

O artigo 1 (SIMEONE et al., 2020) trata da limpeza de equipamentos relacionados com a produção de alimentos, o que consome tempo, água, energia e produtos químicos. O trabalho utiliza sensores ópticos e ultrassônicos para monitorar a remoção de incrustações de materiais alimentícios com diferentes propriedades físico-químicas a partir de uma plataforma de bancada. São realizados processamentos de sinais e imagens para monitorar o processo de limpeza, e um modelo de regressão de rede neural é desenvolvido para prever a quantidade de incrustações remanescentes na superfície. Os resultados mostram que os diferentes tipos de materiais de incrustações alimentares investigados foram removidos, e os modelos de redes neurais foram capazes de prever a área e o volume de incrustações presentes durante a limpeza com precisões acima de 97%.

O artigo 2 (HAO; QIU; LI, 2017) efetua a análise e previsão do escoamento e sedimentos em um reservatório, detalhando as características de longo prazo do escoamento e dos sedimentos, incluindo tendência, ponto de salto e ciclo de mudança. São utilizadas no estudo várias abordagens estatísticas, como análise de anomalia acumulada, método de agrupamento ordenado de *Fisher* e análise espectral de entropia máxima (MESA). Foi elaborado um modelo de predição utilizando o método Auto-Regressive Moving Average (ARIMA).

No artigo 3 (VERAJAGADHESWA et al., 2022), é tratado o problema de limpeza de escadas utilizando robô de limpeza comercial, utilizando uma abordagem que permita realizar movimentos ascendentes e descendentes durante a limpeza, visando melhorar o desempenho geral do robô. Foram elaborados um extrator de recursos, uma estrutura de percepção, uma rede de classificação e regressão para gerar uma caixa delimitadora (Bounding Box). Os testes realizados demonstraram que o robô foi capaz de realizar a cobertura autônoma de área enquanto percorre a escada usando a estrutura de percepção desenvolvida.

O artigo 4 (LE et al., 2021) trata da limpeza de cascos corroídos de navios em doca seca. Para tal, é proposta uma plataforma robótica com mecanismo de adesão magnética permanente e auto localização por fusão de sensores para navegar em uma superfície vertical. Para implementar a plataforma, é utilizado um planejamento de caminho de waypoint completo baseado em rede neural convolucional profunda auto-sintetizante. Também foram utilizadas técnicas de aprendizagem por reforço com uma função de recompensa proposta baseada na operação do robô. Os resultados demonstraram que o gasto de energia foi 10% menor e o de água 9% menor.

No artigo 5 (YIN et al., 2020), é proposto um método de limpeza e inspeção de mesas utilizando robô de apoio humano para operação em um ambiente de praça de alimentação. O robô é capaz de realizar a inspeção e limpeza de lixo alimentar na mesa. Para realizar tal tarefa, é utilizada uma rede neural convolucional profunda. Os resultados experimentais mostram que o módulo de detecção de lixo alimentar atinge uma média de 96% de precisão de detecção.

No artigo 6 (RAMALINGAM et al., 2018), é evidenciado um método de detecção e classificação de detritos para limpeza de pisos. O método faz uso de uma rede neural convolucional profunda e máquina de vetores de suporte. Os resultados provam que a técnica proposta pode detectar e classificar com eficiência os detritos no chão, atingindo 95,5% de precisão de classificação.

No artigo 7 (RAMALINGAM et al., 2018), é fornecida uma solução para monitorar a remoção de incrustações de materiais alimentícios em tubos cilíndricos de plástico e metal. Para tal são realizadas medições ultrassônicas e são utilizados métodos de classificação de aprendizado de máquina. O resultado apresentado no trabalho demonstrou o

potencial de utilização de sensores ultrassônicos de baixo custo para monitorar e otimizar processos de limpeza em tubulações.

Conforme apresentado, existem diversas propostas para amenizar ou mesmo solucionar problemas relacionados com limpeza de detritos, sujeira e afins, seja em cascos de navios, escadas, mesas, canos ou equipamentos de produção de alimentos.

Embora existam artigos que tratam de problemas relacionados aos sedimentos de rios, nenhum deles lida diretamente com o acúmulo nas tomadas d'água de unidades geradoras em usinas hidrelétricas, o que torna a metodologia apresentada neste trabalho um ponto de partida inicial para lidar com os problemas resultantes deste acúmulo e causa grandes transtornos para a operação da usina.

3 Identificação e Monitoramento dos Estados Operativos das UGs

As UGs instaladas na UHE de Jirau possuem inúmeros equipamentos auxiliares, além de sensores, alarmes, relés e atuadores associados. A combinação destes elementos forma um conjunto complexo de dispositivos que está diretamente relacionado ao funcionamento e operação da UG. Para que seja possível utilizar esse conjunto de equipamentos para geração de energia, é imprescindível saber o status operacional atual da UG de modo preciso e em tempo real, de modo a identificar se a mesma se encontra disponível, indisponível, e qual a causa da indisponibilidade.

Antes da proposta e implementação da solução para identificação e monitoramento dos estados operacionais das UGs de forma automática, a verificação e acompanhamento das inúmeras informações que permitem identificar o status operacional das UGs era efetuado de maneira não sistematizada pelos operadores da UHE Jirau. Em tal processo não sistematizado, é necessário acessar inúmeros registros, sistemas e informações de controle localizados em sistemas SCADA, aplicações auxiliares e relatórios de manutenção, para verificar os registros de informações que apresentam o status de cada dispositivo relacionado à UG.

De modo mais detalhado, o procedimento não automatizado executado para identificar o estado atual de cada uma das UGs ocorria à medida que os eventos aconteciam. Quando um operador efetua o desligamento de uma UG, ou mesmo quando a UG é desligada automaticamente devido à atuação de algum equipamento de proteção, a informação é anotada em um documento interno, relatando qual o motivo do desligamento, detalhando qual disjuntor ou relé de proteção ocasionou a parada, e quais dispositivos associados foram acionados.

Nesse documento também consta o motivo da parada, se por conveniência operativa, por falhas ou manutenção. Nos casos de falhas intempestivas, onde a UG foi desligada de modo inesperado, o operador não tinha acesso imediato ao dado sobre qual proteção foi ativada e nem detalhes sobre o momento exato em que esta proteção foi ativada, sendo necessária uma consulta manual ao sistema SCADA da UHE para obter tal informação, o que demanda um tempo considerável em um momento em que uma ação rápida é necessária para solucionar o problema apresentado e reestabelecer o funcionamento da UG.

Em outro cenário de desligamento, como para a execução de intervenções, é necessário ter acesso às informações sobre o período de duração do procedimento de manutenção

a ser executado, bem como o detalhamento sobre os horários de início e fim da manutenção, de acordo com o que foi agendado com o ONS. Uma vez que antes de efetivamente realizar o desligamento, é necessário entrar em contato com o ONS para informar sobre a paralisação da UG, o operador deve consultar o número do documento relacionado ao agendamento prévio da manutenção com o ONS, o número do SGI.

Uma vez reunidas todas essas informações, contando ainda com o apoio de analistas e operadores com experiência relacionada ao funcionamento das UGs e seus subsistemas, é realizada a identificação do estado operacional. Adicionalmente, o operador ainda deve lançar manualmente o estado operativo da UG no SAU.

Na sala de operações da UHE, um painel apresenta o status de cada UG, informando se esta se encontra parada, vazia ou gerando. Quando se trata do estado operativo, é necessário identificar com exatidão o horário em que a unidade migrou de estado, relatando a causa dessa mudança, o que requer consultas ao sistema supervisório para obter detalhes de equipamentos e dispositivos relacionados.

Realizar o processo de identificação para um número reduzido de UGs não é um procedimento que demanda tanto tempo, porém quando é necessário identificar e lançar o estado operativo para ilhas, margens ou mesmo para a UHE inteira, o problema se torna imenso. Em cenários específicos onde a UHE sofre uma parada de inúmeras UGs de modo intempestivo, somente após o reestabelecimento de todo o funcionamento dessas unidades é que o estado operativo era lançado a nível de sistema. Dado que a UHE precisa enviar a cada hora para o ONS um relatório do estado de cada UG, atrasos operacionais como os relatados são prejudiciais.

Em situações em que o ONS identifica que os estados apresentados pela UHE não estão condizentes com a realidade, é necessário justificar a razão da divergência. Uma das maneiras pela qual o ONS é capaz de identificar divergências se deve ao fato de que uma única UG parada que esteja identificada como em geração, pode ocasionar uma diferença de 500 m^3 na vazão da UHE, o que resulta em grandes variações no cálculo perante o ONS.

Outra situação recorrente na UHE de Jirau que demanda grande esforço dos operadores para lançamento de estados são aqueles associados às perdas de carga por acúmulo de sedimentos nas grades de proteção. Na época da cheia do rio, onde o volume d'água aumenta significativamente, é comum que as UGs permaneçam vários dias com certo nível de restrição de potência de geração.

Nesses casos onde a UG está em um estado que represente perda de carga, como por exemplo IGPC (Indisponível Gerando com Perda de Carga), ao fim do dia, é necessário finalizar esse estado e iniciá-lo novamente. Esta etapa é necessária para inserir uma sinalizador de que a UG finalizou o dia em um estado de perda de carga, dando início ao

mesmo estado imediatamente. Este procedimento se repete diariamente até que a UG não esteja mais nesta situação. Tal ação informa ao ONS acerca das perdas de geração ocasionadas pelo problema de acúmulo de sedimentos, tanto em base diária como de maneira acumulada.

Devido ao número expressivo de UGs, 50 no caso da UHE Jirau, fica evidente que este processo não sistematizado é dispendioso, consume recursos humanos para coleta e processamento das informações, é passível de erros e atrasos, além de depender do conhecimento especializado de funcionários da usina. A apresentação da informação de estado operativo, sem que fosse necessária alguma interação do operador, não existia na UHE.

Dado o cenário apresentado, o primeiro passo para sistematizar este procedimento foi a realização de um estudo para identificar qual a melhor abordagem para automatizar a verificação do estado operacional em que a UG se encontra em determinado instante, quais os elementos que alteram esse estado, e como identificar, a partir das alterações detectadas, para o novo estado operacional da UG.

Para encontrar a melhor abordagem para a automatização, foi realizado antes o levantamento de todos os elementos que estão envolvidos direta ou indiretamente nas mudanças do estado operacional de uma UG. Nesta tarefa de análise, foram elaborados fluxogramas apontando quais valores nos sensores, atuadores, relés de proteção, entre outros, causam a mudança de estado atual, além de identificar qual é o estado operativo resultante. Ao final desta etapa, todos os estados possíveis foram enumerados, e também foram identificadas as relações entre estado de origem, alterações que causam a mudança estado, e estado de destino para cada estado enumerado.

Finalizada a etapa de identificação dos estados possíveis e dos elementos que estão envolvidos na mudança de estados, o estudo apontou que uma abordagem satisfatória para monitorar e identificar os estados das UGs seria através da utilização da teoria de Autômatos Finitos Determinísticos (AFD). Essa abordagem se mostrou ideal para representar e controlar as mudanças de estados das UGs, uma vez que a teoria possibilita mapear as entidades e seus atributos relacionados como uma máquina de estados.

3.1 Autômatos Finitos Determinísticos - AFD

AFD são baseados no conceito de máquinas de estados finitos, que são modelos matemáticos usados como abstrações para representar tanto circuitos, programas de computador, processos de produção, etc. As máquinas de estados finitos visam capturar e representar partes essenciais de máquinas reais, como por exemplo, a sequência de paradas de um elevador, determinadas pelas requisições efetuadas pelos seus usuários (HOPCROFT; MOTWANI; ULLMAN, 2006).

Em um dado instante, uma máquina de estados está em um determinado estado, sendo possível migrar para um estado diferente, constante em uma lista de estados possíveis. Uma característica importante de uma máquina de estados finitos é que sua memória é limitada ao número de estados em que a máquina pode estar (HOPCROFT; MOTWANI; ULLMAN, 2006).

A organização é dada de acordo com o conceito de estados, e uma máquina abstrata pode estar em um conjunto finito de estados, sendo que, em um dado momento, a máquina só pode estar em um único estado, conhecido como estado atual. A mudança de estado de uma máquina de um estado de origem para um estado de destino é chamada transição. Para que ocorra uma transição, as condições que determinam a mudança de estado devem ser satisfeitas (RABIN; SCOTT, 1959).

Um AFD é uma estrutura matemática contendo três entidades: um conjunto de estados, um conjunto de transições e um alfabeto. O estado inicial é um elemento contido no conjunto de estados, assim como os estados finais, caso existam, que também são elementos do mesmo conjunto (SAKAROVITCH, 2009). Formalmente, AFD são definidos através de uma quíntupla (Q, $\sum$, σ, q_0, F), onde:

- Q é um conjunto finito de um ou mais estados;
- $\sum$ é um conjunto finito de símbolos de entrada, chamado alfabeto;
- σ é uma função de transição (σ: Q x $\sum \rightarrow$ Q);
- q_0 é o estado inicial, sendo um elemento do conjunto de estados ($q_0 \in$ Q);
- F é o conjunto de estados finais, um subconjunto do conjunto de estados (F $\in$ Q).

O conjunto de estados Q é um conjunto finito, e se Q tem n elementos, cada elemento pode ser nomeado como uma sequência, como 1, 2, 3, 4, ou então com um nome mais representativo, como Estado 1, Estado 2, Estado 3 ou Ativo, Inativo, Indefinido, de modo a indicar mais claramente a situação que este estado representa.

A função de transição exibida na definição de AFD mapeia uma relação *estado com símbolo do alfabeto* para um único estado, e é justamente esta a característica que representa o determinismo do AFD. Dado o estado inicial, através de um determinado conjunto de símbolos do alfabeto chamado palavra, uma transição de estado é ativada e faz com que o autômato seja levado para um único estado de destino. Para ativar determinada transição, todos os símbolos da palavra relacionados à máquina que está sendo representada devem coincidir com os símbolos da transição.

Dado um determinado estado e uma palavra, existe apenas um estado do autômato que pode ser alcançado. Em um AFD, o estado seguinte é definido exclusivamente pelo seu estado atual e pela condição de transição.

A figura 10 exibe um autômato com 3 estados (s1, s2 e s3) e um alfabeto com 5 símbolos (A, B, C, D e E).

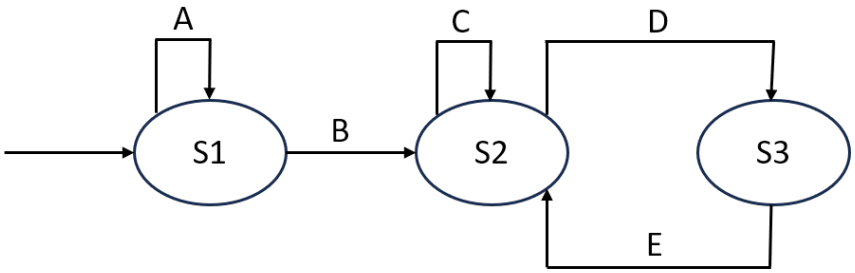

Figura 10 – Exemplo de Autômato

A função de transição pode ser representada visualmente através da matriz de transição, apresentada na tabela 2. A tabela demonstra, por exemplo, que se o estado atual (σ) for *s2* e a transição *C* for ativada, o estado de destino continua sendo *s2*, e se a transição *D* for ativada, o estado de destino será *s3*.

Tabela 2 – Matriz de Transição

σ	A	B	C	D	E
s1	s1	s2	-	-	-
s2	-	-	s2	s3	-
s3	-	-	-	-	s2

3.2 Estados Operativos das Unidades Geradoras

Aplicando a teoria de AFD no problema de identificação de estados das UGs é possível identificar, em um dado instante de tempo, em qual estado operacional a UG se encontra. Para os operadores da usina, os estados representam a informação da disponibilidade ou indisponibilidade dos equipamentos, além de detalhar informações que possibilitam ao operador ter conhecimento se a UG está parada, vazia ou gerando.

As informações dos estados operativos são utilizadas para calcular diversos índices que revelam o desempenho da operação e das rotinas de manutenção, inclusive relacionados ao faturamento da empresa. As informações de estado utilizadas na UHE Jirau são diferentes dos estados utilizados pelo ONS, sendo que este último apresenta menor riqueza de detalhes sobre as UGs, porém, observa-se claramente que esta divergência não representa qualquer diferença entre os dados de disponibilidade apurados nos dois agentes (Operador Nacional do Sistema Elétrico, 2022b).

Quando a unidade está em operação sem conseguir atingir o seu potencial máximo de geração devido à sujeira acumulada nas grades de proteção, essa situação é representada na UHE por um estado operativo que indica que a unidade está com perda de carga.

Além da disponibilidade, caso o estado em que a UG se encontra represente uma situação de manutenção, também é possível determinar a classificação da programação de intervenção junto ao ONS através do SGI. Existem dois tipos de intervenções: as que requerem o desligamento da UG e as que são executadas sem a necessidade de desligamento, logo a identificação do tipo de intervenção é necessário para classificar corretamente a disponibilidade ou indisponibilidade.

Como a UHE de Jirau enfrenta o problema de acúmulo de sedimentos e troncos que requer uma frequente operação de limpeza das grades de proteção, as ações necessárias para tal finalidade frequentemente necessitam que a UG seja desligada.

As intervenções que requerem o desligamento da UG exigem a execução de ações adicionais, independentemente do tipo de operação realizada na unidade. Caso o tempo de desligamento para a realização da intervenção exceda um período de 24 horas, ao término da manutenção, antes da disponibilização dessa UG, será necessário realizar um teste de comprovação de geração. O teste consiste em colocar a UG em modo de geração por um período de no mínimo quatro horas, cujo objetivo é comprovar o potencial de geração na capacidade máxima dadas as condições de vazão existentes.

Durante as quatro horas de execução do teste, a média da potência de geração é contabilizada. Ao final do período, para que o teste seja aprovado e a UG seja considerada disponível, a média da potência de geração obtida deve ser maior ou igual à potência despachável pela UG, que é o valor máximo de geração possível de acordo com a vazão hidrológica disponível. Caso a unidade não consiga alcançar a potência média de geração, há a indicação de que o teste de comprovação de carga falhou, considerando-se assim que a UG está indisponível, mesmo que ainda seja possível realizar a geração.

No total foram enumerados 16 estados que representam o estado operacional das UGs, sendo 4 indicando disponibilidade (DGN, DGT, DV e DP), 3 indicando indisponibilidade com geração (IGP, IGU e IGI), 3 indicando indisponibilidade com equipamento rodando a vazio (IVP, IVU e IVI), 3 indicando indisponibilidade com equipamento parado (IPP, IPU e IPI), e 3 indicando estados com perda de carga (IGPC, IVPC e IPPC). Os estados com o detalhamento da classificação são apresentados na tabela 3. Os estados apresentados mapeiam todos os cenários operativos em que as UGs podem se enquadrar, permitindo aos operadores identificar a situação de cada um dos equipamentos através desses estados.

A caracterização entre gerando, rodando a vazio e parada, se dá da seguinte forma: uma UG em geração está conectada ao SIN (Sistema Interligado Nacional), operando como gerador. Uma UG rodando a vazio está desconectada do SIN, e trata-se de um estado transitório indicativo de que a UG está na velocidade normal, porém não está sincronizada. Uma UG parada está desconectada do SIN, podendo ser solicitada para uso pelo ONS (Operador Nacional do Sistema Elétrico, 2022a; Operador Nacional do Sistema

Tabela 3 – Classificação dos estados operacionais das UGs

Disponibilidade	Status	Classificação	Estado
Disponível	Gerando	Normal	DGN
	Gerando	Teste	DGT
	Vazia	—	DV
	Parada	—	DP
Indisponível	Gerando	Programada	IGP
		Urgente	IGU
		Intempestiva	IGI
	Vazia	Programada	IVP
		Urgente	IVU
		Intempestiva	IVI
	Parada	Programada	IPP
		Urgente	IPU
		Intempestiva	IPI
	Gerando	Perda de Carga	IGPC
	Vazia	Perda de Carga	IVPC
	Parada	Perda de Carga	IPPC

Elétrico, 2022b).

A classificação do estado operacional é efetuada de acordo com 3 níveis. No primeiro nível é definido se a UG está disponível 'D' ou indisponível 'I'. O segundo nível caracteriza a condição operacional da UG, sendo 3 identificações possíveis: gerando 'G', rodando a vazio 'V' ou parada 'P'. O terceiro nível qualifica a disponibilidade ou indisponibilidade de acordo com o 1º nível, indicando normal ou teste, caso a UG esteja disponível, programado, urgente ou intempestivo, caso esteja indisponível.

A indisponibilidade é classificada de acordo com o prazo de solicitação da intervenção junto ao ONS. O prazo refere-se ao período de tempo em que o ONS foi notificado acerca da intervenção antes do início do procedimento. A intervenção programada em regime normal refere-se àquela cuja solicitação for efetuada com antecedência maior ou igual a 48 horas com relação ao horário da intervenção. A intervenção em regime de urgência é aquela cuja solicitação foi efetuada com antecedência menor que 48 horas do horário de início da intervenção. Já a intervenção de emergência é aquela efetuada em equipamento ou instalação com o objetivo de corrigir falha que tenha ocasionado seu desligamento intempestivo, automático ou manual, e que não pôde ser comunicada previamente ao ONS. A tabela 4 apresenta a classificação da indisponibilidade junto ao ONS.

Tabela 4 – Classificação da indisponibilidade

Indisponibilidade	Prazo de notificação ao ONS
Programada	> 48 horas
Urgente	<= 48 horas
Intempestiva	Após a ocorrência da indisponibilidade

3.3 Atributos

De acordo com a teoria de AFD, o alfabeto consiste no conjunto de símbolos, cujos valores determinam a transição de um determinado estado de origem para um único estado de destino.

Uma vez identificado o conjunto de estados operativos possíveis para as UGs, é necessário descrever como ocorrem as mudanças ou transições de estado. Nesta etapa são selecionados os dispositivos da UHE que tem impacto direto na mudança de estado operacional das UGs. Dentre todos os relés, sensores, atuadores e medidores monitorados pelos vários sistemas da UHE, que totalizam milhares de pontos de informação, foram selecionados 11 atributos para viabilizar a identificação das mudanças de estados.

Vale destacar que os atributos escolhidos são resultantes de diversos agrupamentos de informações, saídas de análises e processamentos para limitar a apenas uma quantidade mínima de atributos que ainda assim continue representando as informações vitais da UG. Com o auxílio dos funcionários da usina, foram elaborados fluxogramas com a finalidade de documentar como acontecem as mudanças de estado. Um dos fluxogramas elaborados é exibido na figura 11, onde é possível verificar quais estados podem ser alcançados através do estado de origem 'DGN'.

Na tarefa de identificação de estados, o alfabeto é composto pelos 11 atributos identificados na fase de levantamento através dos fluxogramas de mudança de estados. Na figura 11 os símbolos do alfabeto são representados pelo losango, que consistem nos atributos, que dependendo do valor em que se encontram, levam a UG para determinado estado.

O armazenamento dos valores dos atributos é efetuado no *PI System*, sendo que alguns desses valores são originados dos CLPs da UHE, enquanto outros são gerados pela aplicação de monitoramento de estados desenvolvida, que consulta outros sistemas da UHE, utiliza esses dados durante a verificação e processamento, e salva as informações resultantes nos atributos do *PI System*. Em conjunto, os atributos são avaliados pela aplicação desenvolvida neste trabalho, e caso seja verificado que os valores atuais dos atributos correspondam ao alfabeto de alguma das transições existentes, então esta transição de estados é ativada, o que resulta na migração da UG para um novo estado operacional.

3.4 Transições de Estados

Uma transição de estado leva uma UG de um estado de origem a um determinado estado de destino. Conforme a definição de AFD, uma função de transição mapeia a relação *estado com símbolo do alfabeto* para um único estado. Dado o estado em que a UG se encontra em dado instante, através de um determinado conjunto de símbolos

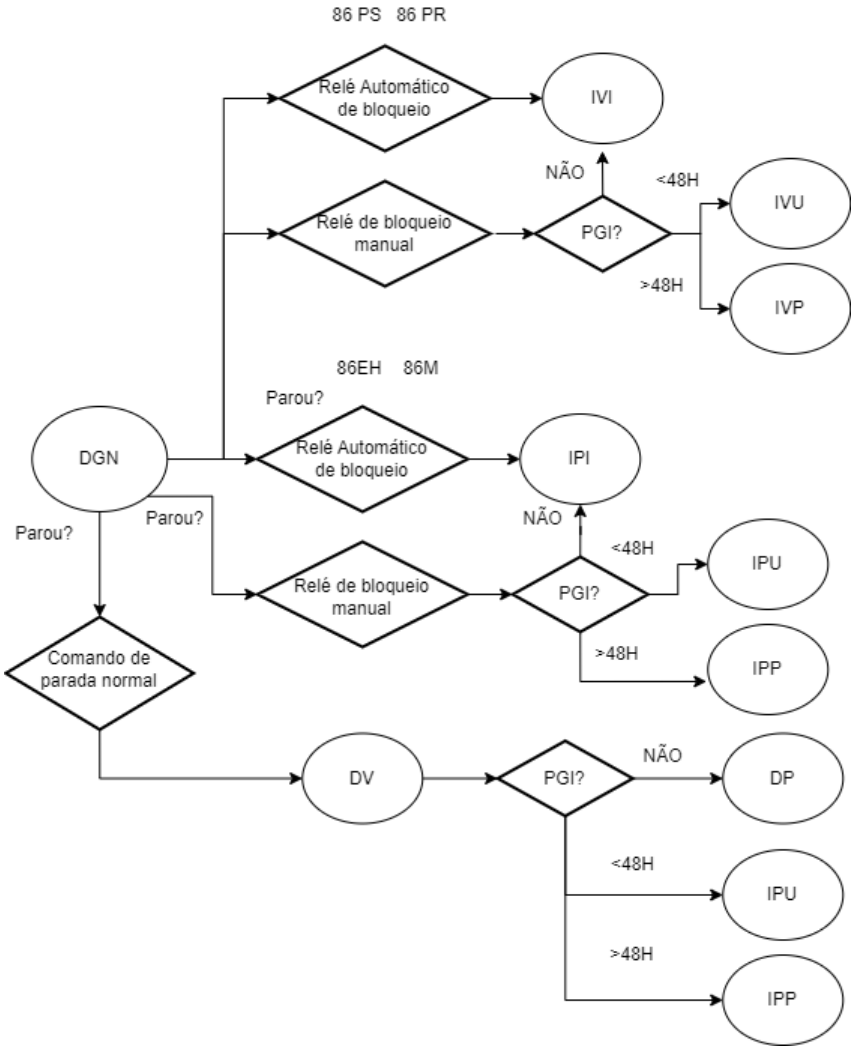

Figura 11 – Fluxograma do estado DGN

do alfabeto, representando os atributos monitorados, uma transição de estado é ativada, levando a UG para um único estado destino. A ativação de uma transição refere-se ao fato de que todos os símbolos do alfabeto desta transição foram satisfeitos.

Na figura 12 é apresentada a regra de transição A, com os respectivos valores de ativação dos atributos. A aplicação de monitoramento verifica alterações nos valores dos atributos, e quando os valores de cada um dos atributos coincidirem com o valor de acionamento constante na regra de transição, a mesma será ativada.

A mesma regra de transição pode ser utilizada para efetuar a transição entre diversos estados de origem e destino, porém, dado o estado atual em que uma máquina de estados se encontra, apenas um único estado de destino pode ser alcançado a partir desta transição. Tal garantia é dada pela relação *estado atual com regra de transição ativada*, resultando em um único estado de destino possível.

A representação visual da função de transição é mostrada como uma matriz na figura 13, que exibe a relação entre estado de origem (1ª linha verde), transições de estados

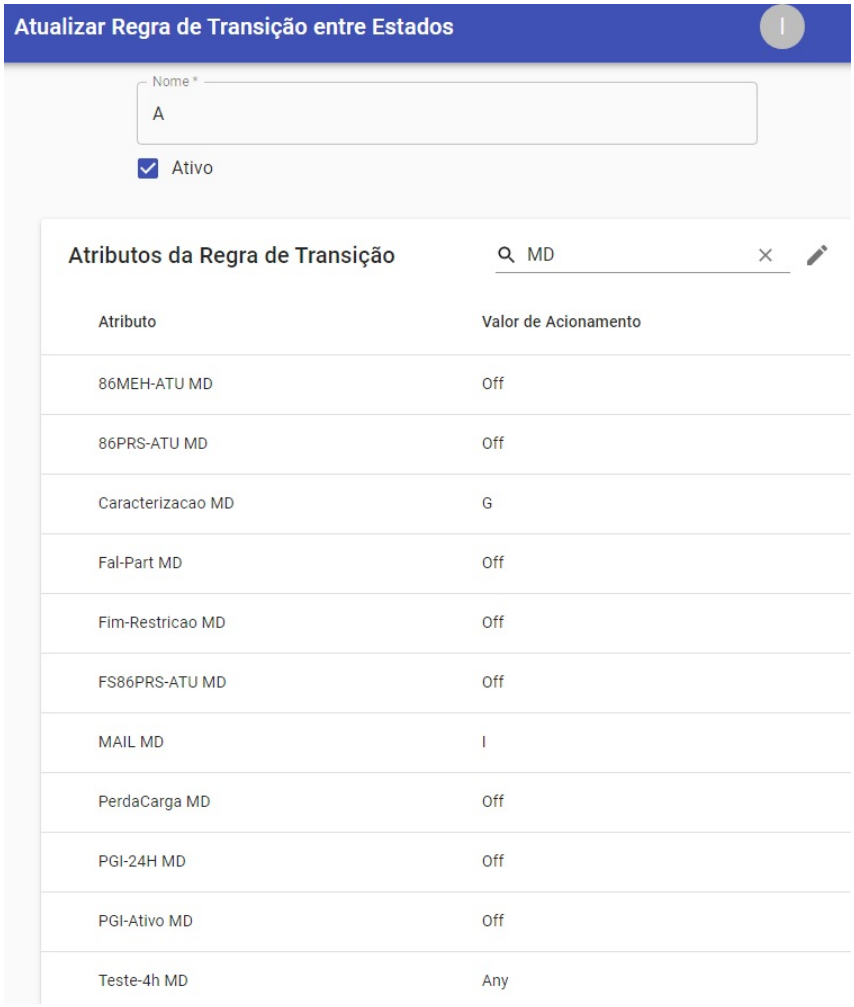

Figura 12 – Tela de modificação das regras de transição entre os estados operativos

(1ª coluna verde) e estado de destino (posição da matriz que relaciona o estado atual da UG com a transição de estado ativada).

Exemplificando, caso a UG esteja no estado atual 'DGN', e a transição 'C' for ativada, o novo estado será 'DV'.

3.5 Análises para Avaliação de Expressões

Além do armazenamento, o *PI System* fornece uma ferramenta para criação de análises, que efetuam cálculos e operações lógicas nos valores dos atributos, e de acordo com as regras elaboradas, permite analisar se o valor do atributo se encontra em uma faixa de valores, se é menor, maior ou igual do que determinando limite, se o valor é considerado válido (verdadeiro ou falso), entre inúmeras outras validações, o que consiste em uma vantajosa ferramenta para verificação de informações.

Em adição às análises, também estão disponíveis fórmulas para a realização de operações lógicas entre atributos do *PI System*. As análises, em conjunto com as fórmulas,

	DGN	DGT	DP	DV	IGI	IGP	IGU	IPI	IPP	IPU	IVI	IVP	IVU
A		IGI	DGN	DGN		DGN	DGN	DGN	DGN	DGN	DGN	DGN	DGN
B	DP			DP	IPI	DP	DP		DP	DP		DP	DP
C	DV		DV		IVI	DV	DV	IVI	DV	DV		DV	DV
D			DGT	DGT	DGT	DGT	DGT	DGT		DGT	DGT	DGT	DGT
E	IGI		IGI	IGI				IGI	IGI		IGI		
F	IVU		IVU	IVU	IVU	IVU	IVU	IVU	IVU		IVU		
G		DGN											
H	IGP		IGP	IGP	IGP		IGP	IGP	IGP	IGP	IGP	IGP	IGP
I	IGU		IGU	IGU	IGU	IGU		IGU	IGU	IGU	IGU		IGU
J			IPI						IPI				
K								DP			DP		
L	IPI			IPI		IPI							
M	IPP		IPP	IPP	IPP	IPP	IPP	IPP		IPP	IPP		IPP
N	IPU		IPU	IPU	IPU	IPU	IPU	IPU	IPU		IPU		IPU
O			IVI	IVI					IVI				
P								DV			DV		
Q	IVI					IVI							
R	IVP		IVP	IVP	IVP	IVP	IVP	IVP	IVP	IVP	IVP		IVP
S					DGN								
T					DP								
U					DV								

Figura 13 – Matriz de transição entre estados operativos das UGs

foram utilizadas para simplificar e reduzir o número de atributos monitorados para a identificação dos estados. Como a UHE de Jirau possui 50 UGs, um número elevado de atributos a serem monitorados, como por exemplo 30 atributos, resultaria em um total de 1500 elementos a serem monitorados (30 atributos x 50 UGs).

Dado que a solução deve apresentar os resultados em tempo real, caso seja necessário efetuar o monitoramento de elevado número de atributos, determinando se os valores atuais de cada um desses atributos devem resultar na ativação de alguma transição de estado, o consumo de recursos de processamento, de armazenamento e de tráfego de dados na rede será tão alto que pode tornar a solução final inviável. Sendo assim, as análises e fórmulas são utilizadas para realizar operações lógicas entre diversos elementos, com a finalidade de disponibilizar a menor quantidade possível de atributos que permita representar os dispositivos de forma agregada.

Na figura 14 é exibida uma análise implementada no *PI System*. A análise apresentada verifica se o teste de comprovação de geração durante as 4 horas foi concluído com êxito, tornando a UG disponível ou indisponível, de acordo com o resultado do teste. As análises do *PI System* utilizam variáveis para armazenar os valores utilizados na lógica. A variável com nome 'Verificacao' apresentada na análise realiza testes lógicos entre

vários atributos de entrada para definir o valor resultante, que é então mapeado para um atributo de saída com o nome 'Teste-4h'.

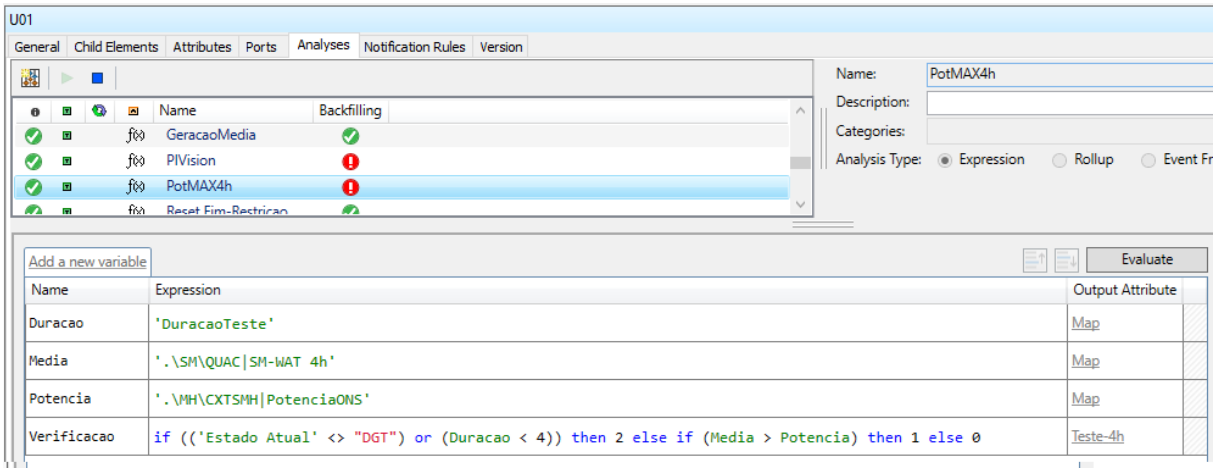

Figura 14 – Análise de verificação do teste de comprovação de carga

A utilização de fórmulas é exibida na figura 15, onde a operação lógica 'OU' é executada entre dois atributos do *PI System* (86EH-ATU e 86M-ATU), que representam relés de proteção. Os relés foram agrupados utilizando o operador lógico 'OU' porque a atuação de qualquer uma dessas proteções causa a parada completa da UG. Sendo assim, a verificação agregada tem o mesmo resultado da verificação individual de cada um desses atributos. O resultado da operação é armazenado no atributo '86MEH-ATU', que é então utilizado no monitoramento de estados para verificação da ativação de algum dos relés de proteção. Apenas no caso apresentado foi possível reduzir pela metade o número de atributos analisados, mantendo ainda a consistência da verificação.

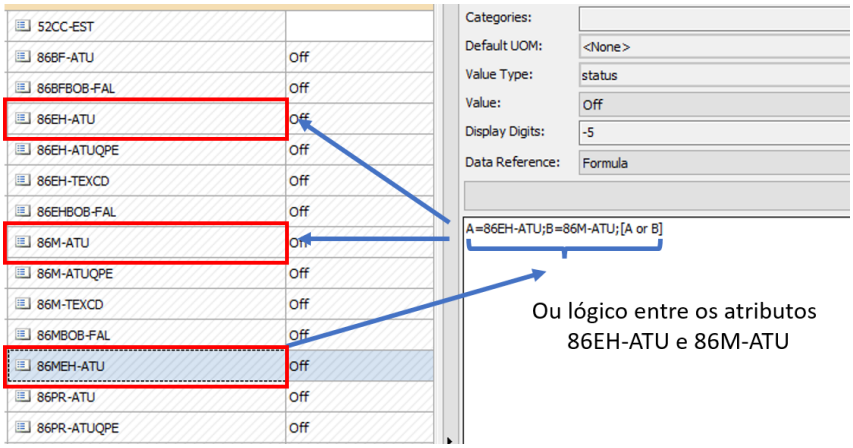

Figura 15 – Fórmula implementando OU lógico entre atributos do PI System

3.6 Monitor Automático de Estados em Tempo Real

Utilizando os símbolos do alfabeto da máquina de estados apresentados na seção 3.3, foi desenvolvida uma solução capaz de monitorar os valores dos atributos, de modo

a determinar qual estado de operação atual de cada uma das UGs e para qual estado de destino a UG deve transitar.

A solução desenvolvida apresenta os resultados em tempo real, disponibilizando imediatamente as informações dos estados atuais das UGs. A identificação de qual transição de estado deve ser efetuada acontece com base nos valores dos atributos e nas listas de transição cadastradas na aplicação. Todas as ações de lançamento de estados das UGs são realizadas de modo automático no SAU, que é o sistema utilizado na UHE para visualização e controle dos estados.

O monitoramento e identificação dos estados são possíveis mediante monitoramento constante de todas as variáveis que afetam ou alteram o estado operacional das UGs. Essa tarefa automatizada adiciona maior confiabilidade no processo, uma vez que não depende da expertise do operador, além de utilizar um procedimento sistematizado para reconhecer as mudanças de estado.

O monitoramento de estados das UGs foi desenvolvido como um serviço executado no sistema operacional *Linux*, instalado no servidor dedicado ao projeto. A aplicação foi dividida em módulos para facilitar a manutenção, possibilitar a disponibilização de recursos de hardware exclusivos por tipo de aplicação, e reduzir o nível de dependência entre os módulos.

A figura 16 apresenta o fluxograma de execução da aplicação de monitoramento. A seguir são descritas as funções de cada módulo desenvolvido, e posteriormente são detalhados os procedimentos executados pelo monitor de estados.

- O módulo 1 recupera os atributos cadastrados no banco de dados da aplicação web, consulta o *PI System* para recuperar os *WebIds*, que são usados para realizar as solicitações de informações, e então salva os valores retornados no banco de dados do monitor de estados. A partir deste momento, o monitor de estados se inscreve para receber notificações sobre atualizações nos valores dos atributos, utilizando os *WebIds* armazenados;

- O módulo 2 efetua o recebimento, leitura e processamento dos e-mails do ONS, armazenando as informações resultantes no banco de dados. Esses e-mails são utilizados para realizar a caracterização da intervenção nas UGs. As tarefas desempenhadas por esse módulo foram implementadas através de um serviço, que tem como funcionalidade exclusiva a leitura e processamento de e-mails, tanto os originados do ONS quanto e-mails de previsão hidrológica enviados pelos funcionários da usina;

- O módulo 3 é responsável pela integração com o SAU, recuperando as informações das intervenções por UG deste sistema e associando com os e-mails do ONS armazenados no banco de dados da aplicação e gerenciados pelo módulo 2, utilizando

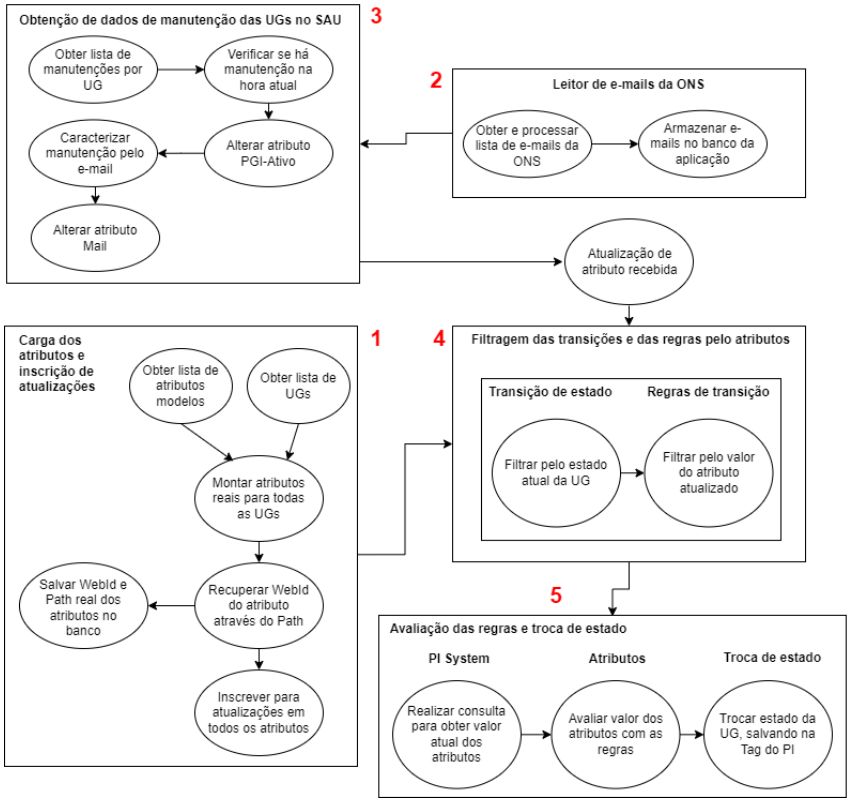

Figura 16 – Fluxograma do Monitor de Estados em Tempo Real

o código da intervenção para efetuar o link entre estes dois conjuntos de informações. Após a recuperação da lista de intervenções, é efetuada a caracterização da manutenção como programada, em regime de urgência ou emergência, alterando os atributos 'PGI-Ativo' e 'Mail' no *PI System* para refletir as manutenções em andamento;

- O módulo 4 realiza o recebimento e processamento das notificações de alterações nos atributos do *PI System*. Utilizando a lista de todas as transições possíveis, obtidas do banco de dados da aplicação, este módulo utiliza o estado atual da UG e os novos valores dos atributos para filtrar as transições disponíveis, verificando se alguma dessas transições foi ativada, para então alterar o estado operativo da UG.

- Por fim, o módulo 5 recupera do *PI System* os valores atuais dos demais atributos, compara esses valores com as regras cadastradas e, se necessário, efetua a troca de estado das unidades geradoras, salvando a informação do novo estado operacional da UG no *PI System*.

O acesso às informações associadas às UGs é efetuado através de consultas ao *PI System* utilizando o serviço *Representational State Transfer - RESTFul*, que disponibiliza as informações resultantes no formato *JSON*, um formato leve para transportar dados muito utilizado em *APIs Web*, que possibilita fácil integração entre sistemas diferentes.

O *PI System* consiste em uma das principais fontes de informações utilizada no monitoramento em tempo real das UGs, para obtenção dos valores armazenados nesse sistema. Para possibilitar a utilização desse serviço, foi desenvolvida uma *API Web* cliente que possibilita a realização de buscas de informações através de um conjunto de parâmetros, tais como data e hora, nome do atributo, localização deste atributo através do caminho, entre inúmeras outras formas de consulta.

A informação complementar utilizada para a determinação dos estados operativos é obtida do Programa de Gestão Integrada (PGI), um documento que registra as intervenções nas UGs da UHE, informando o horário de início e término da manutenção, uma descrição sobre o procedimento que será realizado e com qual finalidade, e também se o equipamento estará disponível para geração durante a intervenção. O PGI também é utilizado para intervenções não programadas, que alteram o estado operativo da UG de modo a torná-la indisponível. As informações do PGI estão localizadas no banco de dados do SAU, e foi necessário desenvolver uma interface que possibilitasse acessar e coletar os dados desta origem.

Em conjunto com os dados do PGI, é necessário processar os e-mails do ONS que contém o retorno das solicitações de intervenções efetuadas pela UHE. Neste e-mail consta a caracterização da intervenção em relação ao prazo de solicitação, de acordo com o mostrado na tabela 4. As informações dos PGIs e dos e-mails do ONS fornecem a base para determinar o momento em que a intervenção irá acontecer e qual a classificação desta, o que está diretamente ligado ao estado operativo das UGs.

Foi desenvolvida uma aplicação web para possibilitar a configuração do monitor de estados, da ferramenta de otimização de despacho e diversas outras aplicações auxiliares. No que se refere ao monitor de estados, a aplicação web disponibiliza telas específicas por onde são efetuadas as operações de adição, remoção e alteração das informações de estados, das regras de mudança de estados, dos atributos monitorados, e a localização dos atributos no *PI System*, necessárias ao funcionamento do monitor de estados.

A figura 17 apresenta a tela por onde são visualizadas e alteradas as informações de estados utilizadas pela aplicação de monitoramento.

Além dos estados, também é possível adicionar, modificar ou remover as transições de estados. Na figura 18 é exibida a tela onde podem ser realizadas as operações de associação entre estado de origem, estado de destino e regra de transição equivalente. Também podem ser alterados os valores de ativação dos atributos, que farão com que a transição seja selecionada para efetuar a troca de estado. Conforme já mencionado anteriormente, para ativar uma transição de estados é necessário que todos os valores dos atributos monitorados coincidam com os valores previamente cadastrados na regra de transição.

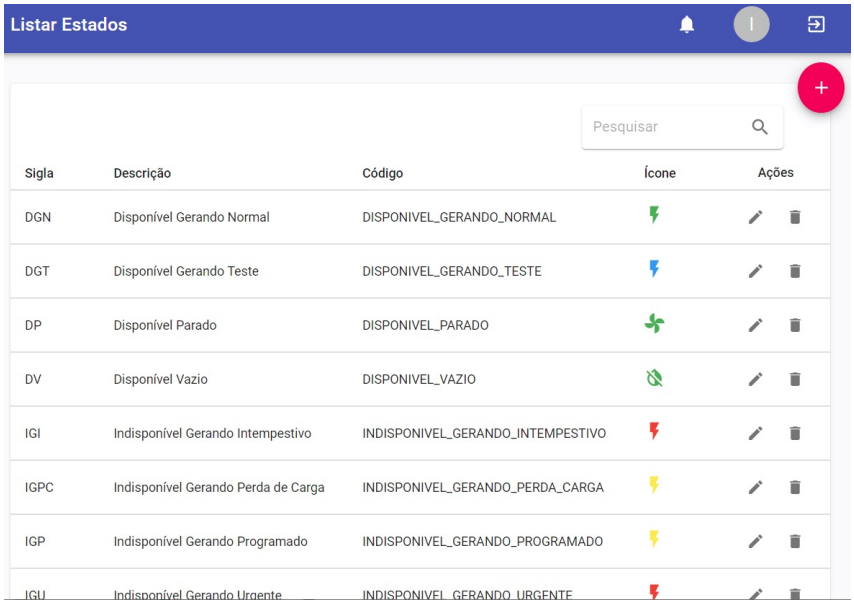

Figura 17 – Tela de modificação dos estados operativos

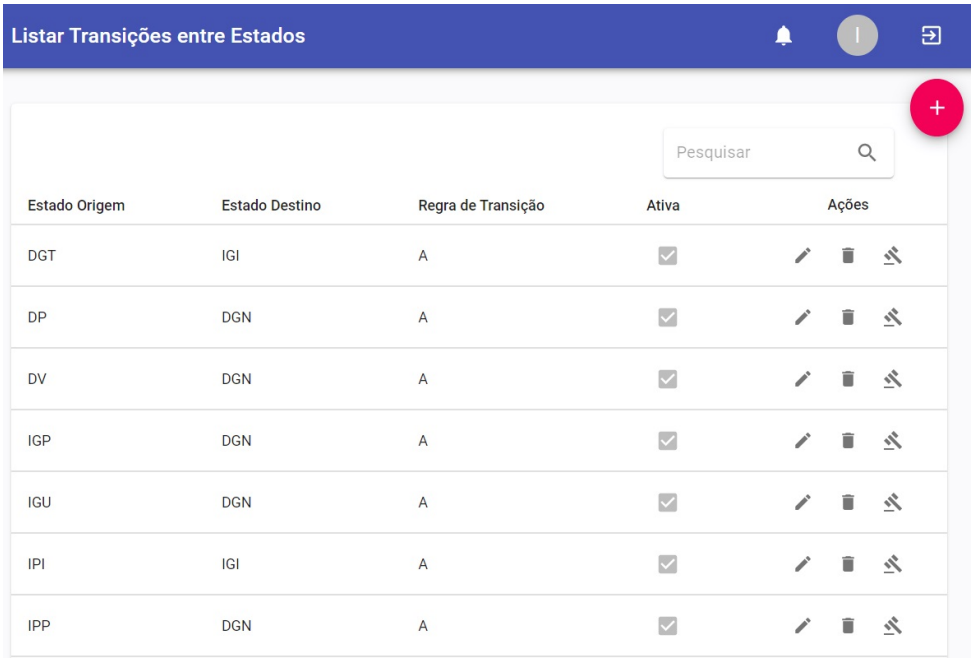

Figura 18 – Tela de cadastro com as opções de seleção do estado de origem, estado de destino e a regra de transição

O serviço que realiza o monitoramento dos estados das UGs é executado a cada minuto, intervalo esse definido pela equipe de funcionários da usina, em consonância com o manual de operações do ONS, pois é o prazo mínimo para que uma mudança de estado possa ser registrada. Esse intervalo de tempo entre as execuções de monitoramento é configurável através da interface disponibilizada na aplicação web desenvolvida.

Quando as rotinas da aplicação de monitoramento são executadas, os passos referentes ao monitor de estados exibidos na figura 16 são efetuados, as verificações de

alterações nos atributos são realizadas e os documentos de intervenção em execução em conjunto com os e-mails do ONS são analisados.

Essas informações são então processadas, e a relação de transições cadastradas são verificadas, filtrando apenas aquelas cujo o estado de origem coincida com o estado atual de cada UG. A partir dessa análise, as alterações de estados identificadas são realizadas, e os resultados das transições de estado são armazenados no *PI System*. Os novos estados das UGs são apresentados imediatamente no painel disponibilizado na aplicação web, conforme mostra a figura 19. Este painel de estados é utilizado na sala de operações da UHE, e auxilia os operadores na decisão de quais UGs utilizar para realizar o despacho energético.

Figura 19 – Painel de UGs utilizado na sala de operações da usina

4 Modelos Gráficos

Os modelos gráficos representam as interações entre um conjunto de variáveis. As variáveis são representadas como vértices de um grafo, enquanto as interações entre esses vértices são apresentadas através de arestas direcionadas. As redes probabilísticas, que são apresentadas através de modelos gráficos, capturam a dependência ou independência associadas com as variáveis representadas. Pares de vértices que não estão conectados em um grafo indicam a independência condicional entre as variáveis que estes vértices representam (WHITTAKER, 2009).

Para possibilitar a compreensão dos modelos gráficos utilizados é necessário a apresentação de conceitos fundamentais, tais como variáveis, grafos direcionados acíclicos, evidência, causalidade e probabilidades.

As variáveis representam o conjunto de eventos de um determinado domínio. Variáveis podem ser discretas, tais como [Azul, Amarelo, Branco, Verde] ou contínuas [-∞, ..., 0.1, 0.2, ..., ∞].

Os grafos direcionados acíclicos (*Directed Acyclic Graphs* - DAG) representam de forma compacta as relações de dependência ou independência entre as variáveis de um dado domínio, de acordo com a presença ou ausência dos vértices. Tais representações são apresentadas através de distribuições de probabilidades conjuntas apresentadas no DAG.

Um grafo é composto por $G = (V, E)$, onde V é um conjunto finito de vértices e $E \in V \times V$ é um conjunto de arestas. Um par ordenado $(u, v) \in E$ indica uma aresta direcionada do vértice u para o vértice v. O vértice u é o pai do vértice v, que por sua vez é filho de u.

4.1 Conceitos Básicos de Probabilidade

Probabilidade é a área da matemática que lida com fenômenos aleatórios, que possuem características de falta de previsibilidade e aleatoriedade. Diz-se que um fenômeno é aleatório quando apresenta resultados diferentes e não previsíveis em uma sucessão de repetições, de modo que não é possível garantir ou predizer o resultado de maneira determinística (WHITTAKER, 2009; IBE, 2014).

A predição de falhas em equipamentos em um intervalo de tempo é um exemplo de evento aleatório, bem como a quantidade de veículos que trafegam por uma determinada avenida em um dado horário. O conceito de probabilidade é fundamental na utilização de Redes Bayesianas (RB), e por tal razão é necessária a introdução desse conceito para esclarecer e embasar o funcionamento desse tipo de técnica (IBE, 2014).

Embora não seja possível prever com precisão qual o resultado de um evento aleatório, é possível extrair estatisticamente informações relacionadas à regularidade de ocorrência de tais eventos, de tal modo que se torna viável prever a frequência de ocorrência dos eventos. Quando se trata de eventos aleatórios, executa-se o experimento foco do estudo por repetidas vezes, com um conjunto de possíveis resultados ou eventos resultantes esperados (KJAERULFF; MADSEN, 2008).

A probabilidade é o valor que representa a frequência de ocorrência de um evento dentro de uma longa série de execuções do experimento. Esse valor de probabilidade pode apresentar números entre 0 e 1, e a soma das probabilidades da série de experimentos deve ser 1. Se um experimento for realizado para verificar a ocorrência de falhas em equipamentos, e o resultado concluir que em 0.75 (ou 75%) das vezes em que as falhas ocorrem, o motivo é superaquecimento, os demais motivos de falhas devem, obrigatoriamente, ser 0.25 (ou 25%), totalizando assim o valor 1 (ou 100%).

Quando são executados os procedimentos de experimentação e verificação de resultados, temos o que é chamado de *espaço amostral*, que se refere aos possíveis resultados de tal experimento, e é expressado como S. Sempre que um experimento é executado, temos que apenas um resultado desse espaço amostral é obtido, e este resultado é chamado *ponto de amostragem*, expressado como a_j, $j = 1, 2, .., T$. Sendo assim, o espaço amostral com todos os T possíveis resultados é expressado como $S = \{a_1, a_2, a_3, ..., a_T\}$.

Um *evento* apresenta resultados dentro do espaço amostral, sendo assim um subconjunto do espaço amostral. Se utilizarmos o exemplo de jogar dados como o evento a ser analisado, o espaço amostral é representado por $S = \{1, 2, 3, 4, 5, 6\}$, e a cada jogada do dado o resultado obrigatoriamente estará dentro do conjunto S. Se forem executados 3 experimentos de jogada de dados, um possível resultado seria $S = \{6, 1, 4\}$.

Quanto ao parâmetro tempo, processos Dado um evento E de um dado espaço amostral S, $P(E)$ representa a probabilidade do evento E de modo que as definições apresentadas a seguir são válidas:

1. A probabilidade de E é representada por um valor dentro do intervalo entre 0 e 1: $0 \leq P(A) \leq 1$.

2. O resultado será um ponto amostral no espaço amostral quando a probabilidade for 1: $P(E) = 1$

3. Em um conjunto de eventos mutuamente exclusivos (E_1, E_2, E_3, ..., E_t) definidos no mesmo espaço, a probabilidade de ao menos um desses eventos acontecer é a soma de suas respectivas probabilidades: $P(E_1 \cup E_2 \cup E_3 ... \cup E_t) = P(E_1) + P(E_2) + P(E_3) + ... + P(E_t)$

4.2 Redes Probabilísticas

As redes probabilísticas são modelos gráficos que relacionam as interações entre um conjunto de variáveis. Essas redes capturam a dependência ou independência condicional associadas às variáveis que estão presentes na rede. Os grafos são capazes de representar de forma intuitiva as relações entre as variáveis, de modo que é muito utilizada para análise em diversos tipos de problemas. Os DAGs são uma classe de grafos que representam de forma robusta as distribuições de probabilidade conjunta que expressam a relação de dependência ou independência (KJAERULFF; MADSEN, 2008).

As representações que as redes probabilísticas podem apresentar são capazes de realizar raciocínio causal e diagnóstico, bem como raciocínio intercausal. O raciocínio causal, também conhecido como preditivo, é aquele indicado pela direção do link que conecta dois vértices, enquanto o raciocínio diagnóstico, também conhecido como abdutivo, é representado na direção contrária do link de conexão (KORB; NICHOLSON, 2010).

Na figura 20 é exibida a representação de uma RB apresentando o problema de câncer de pulmão.

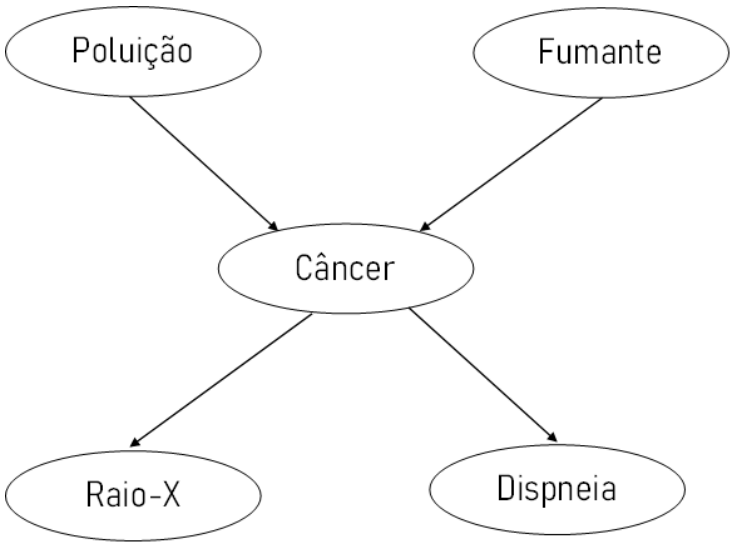

Figura 20 – Rede Bayesiana relacionada ao problema de câncer de pulmão

Utilizando o exemplo da análise do problema de câncer de pulmão, o raciocínio preditivo utiliza novas informações sobre causas para novas crenças sobre efeitos. Sendo assim, o médico que tem conhecimento de que um paciente é fumante, sabe que esse paciente tem maior probabilidade de ter câncer, além de aumentar a crença do médico de que o paciente apresente outros sintomas relacionados ao câncer.

O raciocínio diagnóstico analisa os sintomas para determinar a causa. Um médico cujo paciente relata falta de ar tende a acreditar que o paciente deve ser fumante ou mesmo que tenha câncer.

Embora existam outras metodologias de raciocínio automatizado, é a capacidade de efetuar o raciocínio intercausal que diferencia a inferência em redes probabilísticas das demais técnicas. No exemplo médico utilizado, o raciocínio intercausal é uma forma de pensamento que envolve o raciocínio sobre as causas mútuas de um efeito comum.

Um tipo específico de raciocínio intercausal é chamado de *explaining away*, que pode ser traduzido como "explicada". Neste tipo de raciocínio, causas distintas que possuam um efeito em comum são analisadas. No caso do exemplo sobre câncer, tanto fumante quanto poluição são as causas possíveis para que o paciente tenha câncer.

Embora as causas não estejam diretamente relacionadas entre si, de modo que um sintoma não afete a probabilidade relacionada ao outro sintoma, quando há a comprovação de câncer em um dado paciente, caso seja descoberto que o mesmo vive uma região cujos níveis de poluição são muito altos, isso diminui a probabilidade de que ele seja um fumante.

Sendo assim, a presença de uma causa que explique o efeito, reduz a probabilidade da causa explicativa alternativa, e então se diz que a causa alternativa foi "explicada"(WHITTAKER, 2009).

Na figura 21 são apresentados os tipos de raciocínio discutidos acima.

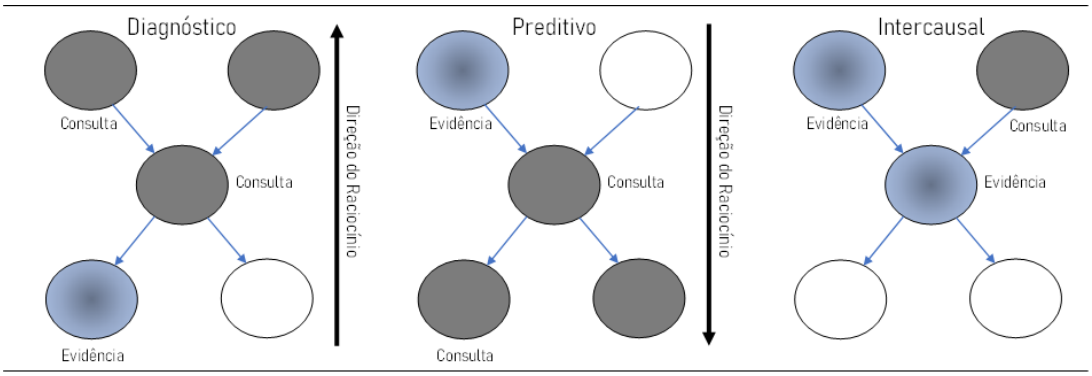

Figura 21 – Tipos de raciocínio utilizados em RBs

4.3 Processos de Markov

Os processos de Markov estão categorizados como processos estocásticos, onde há uma coleção de variáveis aleatórias, sendo estas utilizadas para representar uma característica de interesse. Uma utilização comum para os processos de Markov pode ser observada na modelagem de sistemas que possuem memória limitada do passado (BAUM et al., 1972). Apenas a informação mais recente é relevante e é utilizada para a realização da modelagem e na previsão dos estados futuros que o sistema poderá alcançar.

Ao modelar um sistema em que a máquina se encontra em determinado estado, apenas o estado atual ou os últimos estados mais recentes são considerados na avaliação da probabilidade de atingir algum estado futuro (BAUM; PETRIE, 1966).

Um processo aleatório {X(t) | t ∈ T} é caracterizado como um processo de Markov de primeira ordem se, para qualquer $t_0 < t_1 < ... < t_n$, a função de distribuição cumulativa condicional de $X(t_n)$ para os valores de $X(t_0)$, $X(t_1)$,..., $X(t_n - 1)$, depender apenas de $X(t_n - 1)$ (STAMP, 2004).

Em resumo, dado o presente estado de um processo, o estado futuro independe do passado. Esta propriedade é usualmente chamada de *propriedade de Markov*. Dito de outra forma, um processo é tido como Markoviano se a probabilidade condicional de um evento futuro qualquer é independente do evento passado, dependendo somente do estado presente. Existem processos de Markov de segunda ordem, em que o estado futuro depende tanto do estado atual quanto do estado anterior, e para processos de ordem maior, o mesmo equivale (STAMP, 2004).

A classificação dos processos de Markov é relacionada de acordo com os parâmetros de tempo e quanto à natureza do conjunto de estados. Em ambos os casos, o parâmetro pode ser classificado como discreto ou contínuo. Quanto ao parâmetro tempo, processos de Markov de tempo discreto utilizam valores de tempo enumeráveis, e possuem a característica de que as probabilidades de transição não mudam em relação ao tempo, enquanto processos de Markov de tempo contínuo alteram o estado de acordo com uma variável aleatória exponencial ou, em sua variante, altera o estado de acordo com o menor valor de um conjunto de variáveis aleatórias exponenciais (KUNDU; HE; BAHL, 1989).

Já em relação à natureza dos estados, os processos de Markov de estados discretos são definidos sobre um conjunto enumerável ou finito de estados, enquanto processos de Markov de estados contínuos possuem conjuntos de estados incontáveis. *Cadeias de Markov* referem-se a processos de Markov de estado discreto (KUNDU; HE; BAHL, 1989; LEE et al., 1990).

4.4 Cadeias de Markov

O termo Cadeias de Markov refere-se a um processo estocástico com a propriedade de Markov. Tal processo estocástico está relacionado a uma sequência de variáveis aleatórias. O que difere uma cadeia de Markov de um processo estocástico comum é a falta de memória, ou seja, a propriedade de Markov (STAMP, 2004). Um processo de Markov é tido como uma cadeia de Markov se o tempo for discreto e se para todo *i, j, k,..., m*, a afirmação a seguir for verdadeira:

$$P[X_k = j | X_{k-1} = i, X_{k-2} = n, ..., X_0 = m] = P[X_k = j | X_{k-1} = i] = p_{ijk}$$

onde p_{ijk} é a probabilidade de transição de estados, representando a probabilidade condicional do processo alcançar o estado j no tempo k a partir do estado i no tempo *k-1*. Uma Cadeia de Markov que obedeça à regra apresentada acima é chamada Cadeia

de Markov não homogênea (BAUM et al., 1972). Já a Cadeia de Markov homogênea não depende da unidade de tempo, o que implica na seguinte afirmação:

$$P[X_k = j|X_{k-1} = i, X_{k-2} = \alpha, ..., X_0 = \theta] = P[X_k = j|X_{k-1} = i] = p_{ij}$$

A probabilidade de transição de estados homogênea p_{ij} satisfaz as seguintes condições:

1. $0 \leq p_{ij} \leq 1$
2. $\sum_j p_{ij} = 1, i = 1, 2, ..., n$, ou seja, dado o estado atual, a soma das probabilidades de alcançar os estados de destino possíveis é 1.

As probabilidades de transição são ditas *estacionárias* caso os valores de probabilidades de transição não sejam alterados em relação ao tempo. Cadeias de Markov homogêneas são aquelas em que as probabilidades de transição são constantes em relação ao tempo, conforme apresentado anteriormente.

4.4.1 Matriz de probabilidades de transição de estados

Para representar as probabilidades de transição entre os estados, utiliza-se uma matriz P de n versus n elementos, onde a entrada p_{ij} representa a informação na linha i e coluna j. Para toda e qualquer linha da matriz de probabilidades de transição, a soma das probabilidades da linha deve ser igual a 1.

$$P = \begin{bmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{bmatrix}$$

A representação das probabilidades de transição em uma Cadeia de Markov pode ser feita através de grafos dirigidos, onde as arestas representam o sentido da transição e são rotuladas pelo valor da probabilidade de migrar de um estado de origem para um estado de destino. A figura 22 exibe uma cadeia de Markov com uma representação utilizando grafo.

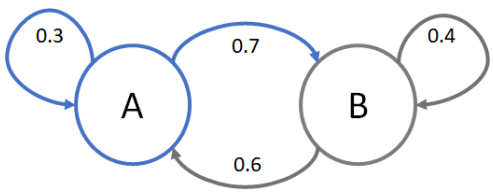

Figura 22 – Representação de uma Cadeia de Markov utilizando grafo

É possível visualizar pelo grafo exibido na figura 22 que a Cadeia de Markov representada possui dois estados A e B. Dado o estado A, é possível continuar no estado A com probabilidade 0.3 e é possível transitar para o estado B com probabilidade 0.7. No estado B é possível continuar no estado B com probabilidade 0.4 e transitar para o estado A com probabilidade 0.6. A matriz de probabilidade de transição referente à Cadeia de Markov apresentada no grafo da figura 22 é exibida a seguir:

$$P = \begin{bmatrix} 0.3 & 0.7 \\ 0.6 & 0.4 \end{bmatrix}$$

Para dar início a uma Cadeia de Markov, é necessário identificar os valores de probabilidades do estado inicial. Esta informação indica qual a probabilidade de o processo iniciar em determinado estado, e sua representação pode ser dada através do uso de um vetor, que tem o tamanho do conjunto de estados possíveis, e a soma das probabilidades desse vetor deve totalizar 1. A representação do vetor de probabilidades de estado inicial é mostrado a seguir:

$$\pi = \begin{bmatrix} 0.5 & 0.5 \end{bmatrix}$$

O que indica uma probabilidade de 50% do processo ser iniciado no estado A e 50% de ser iniciado no estado B.

4.4.2 Classificação de Estados

Os estados utilizados para identificar a situação em que um processo de Markov se encontra possuem algumas características importantes que são utilizadas em sua classificação.

Se for possível a um processo alcançar o estado j a partir de um estado i, tem-se que o estado j é *acessível* através do estado i. Por definição temos:

$i \to j$, se $p_{ij}^{(n)} > 0$ para algum n.

Dois estados i e j são ditos *comunicantes* se eles são *acessíveis* um do outro. A seguir é dada a definição:

$i \leftrightarrow j$ significa que $i \to j$ e $j \to i$.

A comunicação representa uma relação de *equivalência*, sendo que:

- todo estado comunica consigo mesmo, $i \leftrightarrow i$;
- se $i \leftrightarrow j$, então $j \leftrightarrow i$;
- se $i \leftrightarrow j$ e $j \leftrightarrow k$, então $i \leftrightarrow k$.

Os estados de uma Cadeia de Markov podem ser particionados em *classes comunicantes*, de forma que apenas membros de uma mesma classe são comunicantes entre si. A Cadeia de Markov apresentada na figura 23 pode ser particionada nas seguintes classes:

Classe 1: {Estados A, Estado B},

Classe 2: {Estados C, Estado D, Estado E}.

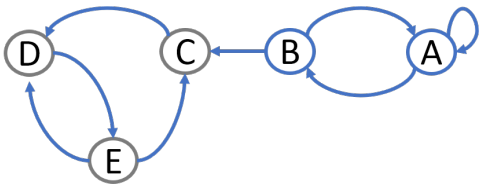

Figura 23 – Diagrama de transição de estados

Uma Cadeia de Markov é dita *irredutível* quando todos os estados se comunicam entre si, ou seja, se existe apenas uma única classe comunicante.

As classes de estados podem ser classificadas como *transientes* e *recorrentes*. Na figura 23 pode-se observar que ao acessar a classe 1, um processo pode alternar sucessivas vezes entre os estados A e B. Em determinado momento, uma vez que o processo atinja a classe 2, não é mais possível retornar à classe 1, ficando restrito aos estados C, D e E.

Os estados da classe 1 são chamados *transientes* pois, uma vez que o processo atinja esses estados, é possível que nunca mais esses estados voltem a ser visitados, logo estes estados serão visitados um número finito de vezes.

Os processos da classe 2 são chamados *recorrentes* pois esses estados com certeza serão visitados novamente, mesmo que não imediatamente, de modo que estes estados serão visitados infinitas vezes durante a execução do processo.

A *periodicidade* é uma propriedade que define que determinado estado de uma Cadeia de Markov será visitado novamente após determinado número de passos.

4.5 Modelos Ocultos de Markov - *Hidden Markov Models (HMM)*

HMMs tem sido utilizados amplamente desde a década de 60, devido à sua rica modelagem matemática e teoria básica para uso em uma variedade de aplicações (RABINER, 1989). Os modelos foram aplicados inicialmente para reconhecimento de fala por (BAKER, 1975), sendo a teoria básica publicada em artigos no final de 1960 e início de 1970 (BAUM; PETRIE, 1966; BAUM et al., 1972).

Nos últimos anos, HMM tem sido aplicado em áreas diversificadas, tais como o reconhecimento de caracteres em textos manuscritos (KUNDU; HE; BAHL, 1989), na

detecção de falhas de sistemas dinâmicos (SMYTH, 1994), na verificação online de assinaturas (YANG; WIDJAJA; PRASAD, 1995) e ainda no reconhecimento de fala (LEE et al., 1990).

HMM é considerado um processo de natureza estocástica, onde um dos processos envolvidos em sua formulação não é observável. Sendo assim, os modelos são utilizados em situações onde os eventos em que se tem interesse estão ocultos, ou seja, não é possível observar tais eventos diretamente. Por exemplo, uma situação onde não é possível observar diretamente um evento é na determinação da temperatura anual média em um passado distante, antes mesmo que os termômetros fossem inventados. No entanto, se for possível determinar uma relação direta entre o tamanho dos anéis de crescimento das árvores e a temperatura de uma determinada região, é possível utilizar a informação do tamanho dos anéis (evento observável) e inferir a temperatura anual daquela região (evento não observável) (STAMP, 2004).

4.5.1 Especificação de Modelos Ocultos de Markov

Os seguintes componentes fazem parte de um HMM:

- T = tamanho da sequência de observação;
- N = número de estados no modelo;
- Q = estados distintos do modelo de Markov {1, 2, ..., N}.Os estados em um determinado tempo são numerados da seguinte forma: $\{q_1, q_2, ..., q_{N-1}\}$;
- M = número de símbolos de observação distintos do modelo;
- V = conjunto de observações possíveis, comumente chamado de símbolos de observação do modelo, denotados por {1, 2, ..., M-1};
- π = distribuição inicial de estados;
- A = probabilidades de transição de estados, de tamanho N x N e representação $\{a_{ij}\}$, conforme definido no item 4.4.1;
- B = matriz de probabilidades de observação, que relaciona os Q estados com os V símbolos de observação do modelo. As linhas da matriz representam os estados, enquanto as colunas representam os símbolos de observações;
- O = sequência de observação $\{O_1, O_2, ..., O_{T-1}\}$, onde $O_i \in$ V para todo i = 1, 2, ..., T-1.

A matriz A é representada por $a_{ij} = \mathrm{P}(\text{estado } q_j \text{ em } t+1 \mid \text{estado } q_i \text{ em } t)$ e pode ser lida como a probabilidade de alcançar o estado q_j no tempo $t+1$ dado que o estado atual no tempo t é q_i. As probabilidades a_{ij} são *invariantes* em relação ao tempo t.

A matriz B é representada por $b_j(k) = \mathrm{P}(\text{observação } k \text{ em } t \mid \text{estado } q_i \text{ em } t)$ e pode ser lida como a probabilidade de observar k no tempo t dado que o estado atual no tempo t é q_i. As probabilidades $b_j(k)$ são *invariantes* em relação ao tempo t.

As matrizes A e B são estocásticas por linha, ou seja, a soma de cada uma das linhas deve ser sempre igual a 1. Por definição, um HMM é definido por A, B e π, sendo estas entidades definidas implicitamente por N (número de estados) e M (número de símbolos de observação), e a representação do modelo é dada por $\lambda = (A, B, \pi)$

4.6 Redes Bayesianas

As RBs são utilizadas para representar dependências causais entre variáveis representadas por nós, onde o grau de dependência entre eles define os valores atribuídos para cada um desses nós, de acordo com a relevância do modelo probabilístico (CARITA et al., 2013), e para modelar sistemas complexos que envolvem incerteza e variabilidade, tais como em finanças, saúde e engenharia. RBs misturam conceitos de grafos e teoria da probabilidade (KORB; NICHOLSON, 2010).

As RBs são modelos gráficos para raciocinar sob incerteza, utilizando os nós para representação das variáveis de um determinado domínio de estudo e os arcos que conectam os pares de nós representam a dependência direta entre as variáveis. As RBs baseiam-se na ideia de que as crenças sobre um determinado evento ou situação podem ser atualizadas à medida que recebemos novas informações. Cada nó da rede representa uma variável aleatória, e os arcos representam as relações probabilísticas entre essas variáveis.

O valor que representa o grau de dependência entre os nós é quantificado pela distribuição de probabilidade condicional (*Conditional Probability Distribution* - CPD) associada a cada nó. As RBs possuem restrição em relação aos arcos, onde não é permitida a existência de ciclos, ou seja, não é possível retornar a um nó caminhando através dos arcos.

Como as RBs representam as relações entre as variáveis através de grafos direcionados, bem como as distribuições de probabilidades conjuntas utilizando grafos direcionados acíclicos, elas podem então ser chamadas de DAGs (*Directed Acyclic Graphs* - DAG) (KJAERULFF; MADSEN, 2008).

Os nós podem ser representados por valores integrais que possuam um intervalo de valores definidos, como por exemplo tempo de uso em anos, variando de 1 até 200. Pode também ser representado por valores ordenados para uma determinada variável de

interesse, como cores $\{vermelho, verde, azul\}$. Outro tipo de representação são os valores lógicos ou booleanos, que informam o estado operacional de determinado componente indicando, por exemplo, se a refrigeração está ligada Sim (*True*) ou Não (*False*).

O desafio na utilização de RBs para lidar com problemas que envolvem incerteza está relacionado à seleção das variáveis de interesse. A escolha de determinadas variáveis dentro de um amplo espaço de possibilidades deve ser feita com cautela, pois a qualidade das variáveis escolhidas determina a qualidade do modelo obtido, e consequentemente das informações extraídas.

As RBs oferecem uma solução poderosa ao lidar com incerteza, permitindo modelar relações complexas entre as variáveis e fazer previsões com um alto grau de precisão. O uso de RBs viabiliza uma compreensão mais profunda dos fatores subjacentes que influenciam os resultados e permite identificar as estratégias mais eficazes para atingir os objetivos desejados (KJAERULFF; MADSEN, 2008; KORB; NICHOLSON, 2010; CARITA et al., 2013).

Visando reduzir o intervalo de valores que uma variável pode assumir, é comum a criação de grupos ou faixa de valores. Para representar a potência de operação de uma UG podem ser utilizadas faixas, como por exemplo *Pot1* para valores entre 25 a 50 MW, *Pot2* para valores entre 50 MW e 75 MW e assim por diante. A utilização de grupos ou faixas deve ser feita de uma maneira que mantenha a representatividade do domínio de conhecimento da variável em análise, porém possibilitando a redução no custo computacional através da redução do espaço de valores existentes.

As RBs são estruturadas de modo a possibilitar a captura das relações qualitativas entre variáveis. Essa relação pode ser observada na RB quando estão presentes arcos conectando dois nós, o que implica que há uma relação causal entre esses nós. Se o nó A possui um arco apontando para o nó B, a direção de A para B representa a relação causal, demonstrando que a variável A afeta a variável B, conforme pode ser observado na figura 24.

Figura 24 – Relação entre Variáveis em Redes Bayesianas

As RBs permitem modelar os efeitos de múltiplas variáveis sobre um determinado resultado, mesmo quando as relações entre as variáveis não são bem compreendidas. Elas auxiliam na identificação dos fatores mais importantes que influenciam um determinado resultado e também possibilitam prever a probabilidade de diferentes cenários com base nos dados disponíveis (KJAERULFF; MADSEN, 2008).

O *Teorema de Bayes* é um conceito fundamental nas RBs. É uma fórmula matemática que descreve a probabilidade de um evento ocorrer com base no conhecimento prévio das condições que podem estar relacionadas com o evento. O teorema tem o nome do reverendo Thomas Bayes, um estatístico do século XVIII, que o formulou pela primeira vez (EDWARDS, 2004).

O teorema de Bayes pode ser expresso na seguinte fórmula:

P(A|B) = P(B|A) * P(A) / P(B)

Onde: P(A|B) é a probabilidade condicional de A dado B, P(B|A) é a probabilidade condicional de B dado A, P(A) é a probabilidade prévia de A, P(B) é a probabilidade prévia de B.

A fórmula pode ser usada para atualizar as crenças sobre a probabilidade de ocorrência de um evento à medida que novas informações são recebidas. Por exemplo, se a probabilidade anterior da ocorrência de um determinado evento for conhecida, e se novas informações relacionadas ao evento forem recebidas, pode-se atualizar a crença sobre a probabilidade de ocorrência do evento.

As RBs também podem ser usadas para identificar as variáveis mais importantes que influenciam um determinado resultado. Analisando as dependências condicionais entre as variáveis, pode-se identificar aquelas que têm a mais forte influência no resultado.

A construção de uma RB envolve várias etapas. Em um primeiro momento, é necessário identificar as variáveis que são relevantes para o problema que está sendo analisado, o que envolve a coleta, preparação e processamento de dados. Após esta etapa, é necessário determinar as relações de dependência condicionais entre as variáveis, para determinar qual variável influencia ou é influenciada por outras variáveis. As ferramentas de software comumente utilizadas para a construção e utilização de RBs são: *Netica*, *Hugin*, *GeNIe* e bibliotecas Python.

Um ponto que deve ser destacado na utilização de RBs é que a rede não é sempre precisa, uma vez que as RBs são baseadas em relações probabilísticas entre variáveis, e há sempre um grau de incerteza envolvido. A rede será tão precisa quanto os dados que são usados para construí-la, e a rede precisa ser atualizadas conforme novos dados se tornam disponíveis, considerando ainda que as relações entre as variáveis podem mudar com o tempo.

Para a construção manual de uma RB é necessário muita habilidade, criatividade e um contato próximo e constante com especialistas no domínio do problema, o que acaba se tornando uma tarefa árdua. Como método alternativo, existem vários algoritmos que podem ser utilizados na construção de RBs, e a escolha sobre qual deles utilizar depende de vários fatores, desde o tamanho da rede, a complexidade do sistema que está sendo modelado e a quantidade de dados disponíveis para uso. Dentro os algoritmos

mais utilizados estão o algoritmo Chow-Liu (SUZUKI, 2012), o algoritmo K2 (LERNER; MALKA*, 2011) e o algoritmo Hill Climbing (SCANAGATTA; SALMERÓN; STELLA, 2019; KOSKI; NOBLE, 2012).

Um aspecto importante a ser considerado ao utilizar algoritmos para a construção de RBs se refere aos recursos computacionais consumidos nesta tarefa, uma vez que o problema é considerado *NP-Hard*. A construção através de algoritmos tem como objetivo escolher o melhor grafo que represente a RB dentro de uma imensa quantidade de candidatos, sendo que à medida que o número de variáveis do modelo aumenta, o número de grafos candidatos aumenta rapidamente. Uma vez que a RB está construída, seja de forma manual ou através do uso de algoritmos, é necessário obter as CPDs (*Conditional Probability Distribution*) associadas à RB, o que pode ser feito mediante análise dos dados disponíveis, através de especialistas no domínio analisado, ou mesmo através de *DataSets* com tais informações.

Uma importante propriedade das RBs é chamada de *Independência Condicional - IC*, que possibilita simplificar as relações entre as variáveis, reduzindo a complexidade da rede e facilitando sua análise e compreensão. A IC viabiliza a realização de inferências em uma RB de forma mais eficiente, pois conhecendo quais variáveis são condicionalmente independentes, é possível evitar o cálculo de probabilidades desnecessárias.

Em uma RB, duas variáveis são condicionalmente independentes se a probabilidade de uma variável não depender do valor da outra, dados os valores das outras variáveis da rede. De modo geral, toda e qualquer variável é independente de seus não-descendentes e de seus não-pais, condicionados em seus pais. A representação da IC é feita usando a notação P(A|B,C), que significa a probabilidade de A dado B e C. Se A é condicionalmente independente de B dado C, então P(A|B,C) = P(A|C) (KJÆRULFF; MADSEN, 2006).

Existem dois tipos de ICs, a IC Global, que ocorre quando dois conjuntos de variáveis em uma RB são independentes um do outro, ou seja, a probabilidade de um conjunto de variáveis não depende dos valores do outro conjunto de variáveis, e a IC Local, que ocorre quando duas variáveis são independentes uma da outra, dados os valores de seus pais. Isto significa que a probabilidade de uma variável não depende do valor da outra variável, dados os valores de seus pais.

Uma tarefa rotineira relacionada às RBs consiste em computar as probabilidades posteriores. Utilizando a forma $P(X|\varepsilon)$, sendo ε a informação ou *evidência* que a rede obteve através de dados externos, na forma de distribuição de probabilidades de uma determinada variável X. Uma evidência pode ser considerada dura *(hard)* ou macia *(soft)*, sendo que a evidência dura possui uma função associada que atribui probabilidade zero a todos os estados, exceto a um deles. Já a evidência macia, também conhecida como evidência virtual, é aquela cujo estado exato da variável não é conhecido, embora existam informações sobre a possibilidade da variável se encontrar em determinado estado

(KOSKI; NOBLE, 2012).

5 Processamento de Dados

Para alcançar o objetivo de prever o tempo de parada para decantação das UGs, é necessária uma grande quantidade de informações para alimentar tanto o modelo de previsão, quanto para identificar os estados operativos atual das UGs. Um dos desafios inerentes à tarefa de lidar com grandes volumes de informações está no fato de que cada fonte de dados apresenta e armazena as informações de uma maneira diferente, como por exemplo, utilizando o gerenciador de banco de dados SQL Server em alguns casos e o gerenciador da Oracle em outros casos.

No que se refere à apresentação das informações, em alguns dos sistemas de onde as informações precisam ser coletadas, é possível extrair a informação no formato JSON, um formato de troca de dados entre sistemas, independente da linguagem de programação, e que facilita bastante a leitura e processamento das informações. Já em outros casos, foi necessário utilizar relatórios disponíveis nos sistemas existentes na UHE para exportação de informações. Estes relatórios tiveram que ser pré-processados para que fosse possível a leitura e processamento. Ainda foram utilizadas informações em formato *Comma Separated Values* (CSV) em casos em que o sistema utilizado pela UHE oferecia essa opção.

O uso de técnicas de Big Data e *Analytics* envolve grandes quantidades de dados, em vários formatos e em volumes diferentes. Para o cenário utilizado no contexto desse trabalho, essas características estão presentes, e representam desafios para a obtenção e organização de toda informação necessária para prever o tempo de parada para decantação e também os estados operativos das UGs.

Para lidar com estes desafios, de modo a possibilitar a utilização das informações para alcançar o objetivo apresentado, neste trabalho é proposto uma arquitetura que seja ao mesmo tempo robusta, adaptável, extensível e escalável, e que seja capaz de lidar com toda a complexidade apresentada no cenário de UHEs, no caso de estudo deste trabalho, a UHE de Jirau.

5.1 Arquitetura Proposta

Para organizar a obtenção, carregamento e processamento de informações é proposta uma arquitetura extensível, que seja capaz de acomodar a inserção de novas fontes de dados, novos algoritmos de manipulação de dados, e principalmente novos métodos para visualizar as informações após o processamento. Além disso, essa arquitetura deve ser confiável, uma vez que decisões operacionais e gerenciais são tomadas levando em consideração os resultados apresentados através dela. Sendo assim, as etapas devem ser

realizadas com especial atenção nas rotinas executadas, para prover a confiabilidade necessária aos dados resultantes. Nesta etapa, os funcionários da UHE estiveram presentes para efetuarem a validação necessária nos resultados obtidos.

O problema detectado logo no início dos trabalhos está relacionado à coleta de dados. Como acessar e recuperar dados disponíveis em sistemas que foram projetados por empresas diferentes, que utilizam métodos de armazenamento distintos e em formatos e volumes diferentes? Um ponto mais importante do que os apresentados previamente está relacionado à realidade de que estes sistemas de onde as informações são coletadas são dinâmicos, o que significa que o modo como a informação é armazenada hoje pode mudar, bem como o formato como estas informações são exportadas desses sistemas.

Levando em consideração o dinamismo do ambiente e também dos sistemas existentes, a arquitetura proposta lida com o problema utilizando acoplamentos de conectores de dados. O acoplamento de conectores possibilita adicionar ou alterar as fontes de dados sem que isso acarrete mudanças na utilização dos dados. Foi desenvolvida uma camada separando as informações disponíveis na fonte de dados do uso efetivo desses dados na aplicação. Esta camada adicional faz a conexão entre a origem e o uso da informação, de modo que se for necessário alterar o modo como a informação é coletada, não há implicações na utilização dada para os dados, simplificando assim alterações no modelo de busca.

A figura 25 apresenta a camada de acoplamento onde o conector está inserido. Na figura, o conector está identificado como Integrador, que é a entidade responsável por lidar com a conexão entre os dados e a camada de aplicação que faz uso destes dados. Após desenvolver a camada de acoplamento de conectores, vários conectores foram elaborados, cada um para recuperar as informações de determinado sistema da usina. Neste tipo de abordagem, à medida que os conectores estavam sendo desenvolvidos, já foi possível entregar dados para utilização de forma gradativa, pois cada conector é independente dos demais, o que confere maior robustez para a solução.

5.1.1 Armazenamento e Processamento de Dados

Toda a informação coletada das diversas origens através do acoplamento de conectores precisa estar disponível em uma infraestrutura de onde possa ser processada. Como o volume de informações é alto, foi selecionado um modelo de armazenamento utilizando um banco de dados distribuído, o Apache Cassandra. Este modelo foi adotado para viabilizar o processamento paralelo de diversas fontes de informação.

O Apache Cassandra é um sistema gerenciador de banco de dados No-SQL, que possui as características desejáveis para trabalhar com grandes volumes de dados, pois é um sistema distribuído que comporta a utilização de vários membros dentro de um

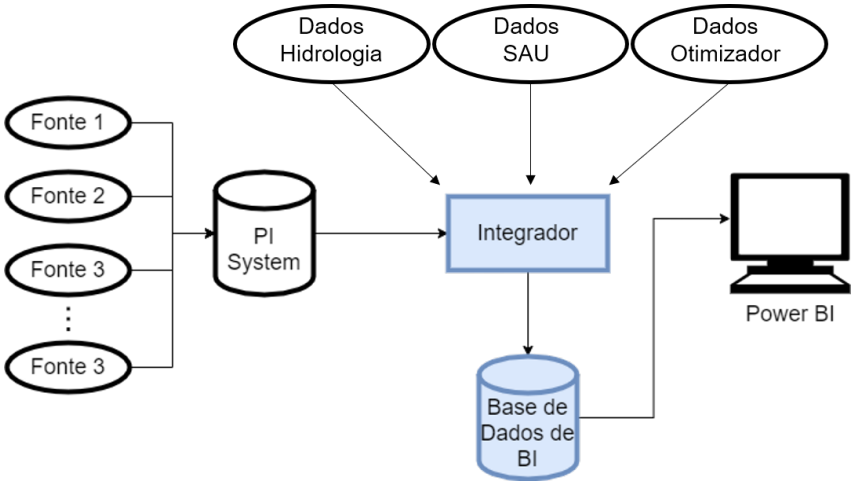

Figura 25 – Camada de acoplamento de dados. A camada lida com integrações de diversas fontes de informação.

conjunto ilimitado de *clusters*, sem a necessidade de um controlador mestre que pode se tornar um ponto central de falhas.

Além disso, esse sistema de gerenciamento é escalável, de modo que o tempo de leitura e gravação de dados aumente de forma linear à medida que a quantidade de membros nos *clusters* aumenta. Outras características de destaque são a existência de uma grande comunidade de desenvolvedores colaborando para solucionar falhas, o fato de que esse sistema é gratuito e é muito utilizado em empresas como Apple, Cisco, Facebook e Netflix, o que demonstra a capacidade e confiabilidade do sistema. Estas características possibilitaram encontrar ajuda para muitos problemas durante o desenvolvimento do trabalho, utilizando especialistas e seus conhecimentos publicados em fóruns e centrais de ajuda.

Uma vez solucionado o problema de acesso aos dados e escolhido o método de armazenamento mais adequado para lidar com as características do ambiente da UHE, outro ponto foi levantado: como possibilitar o processamento dos dados de modo que seja possível adicionar novas maneiras de manipular esses dados com o menor impacto possível na utilização desses dados?

Diversos *softwares* comerciais realizam com eficiência esta tarefa, porém a maioria deles não disponibiliza interfaces para que seja possível integrações com softwares de terceiros, e quando existe um modo de integração simples, este ainda exige iterações manuais. Além do problema exposto, existe a questão do licenciamento. Os melhores softwares para manipulação de dados através de rotinas de ETL são pagos, e os valores costumam ser na casa dos milhões para uso em um ambiente como o de UHEs.

Como solução para o problema apresentado, foi utilizado um conjunto de *softwares* e bibliotecas para criar um ambiente de coleta de dados, armazenamento e processamento distribuídos, escalável e adaptável. O processamento de dados é realizado utilizando-se o

Apache Spark, e a análise de dados se dá pela utilização da linguagem de programação Python juntamente com as bibliotecas PySpark, NumPy e Pandas.

O *DataFrame* do Apache Spark oferece inúmeras operações passíveis de serem executadas nos dados, e as rotinas de ETL foram implementadas utilizando todo esse conjunto de softwares. Como o Apache Spark realiza as tarefas de modo distribuído, utilizando ainda a memória do computador, isso faz com que o processamento seja muito mais rápido se comparado às alternativas que fazem uso de sistemas de arquivos, como por exemplo o MapReduce.

A utilização da linguagem Python possibilita o uso de classes, herança e polimorfismo, características do paradigma de Programação Orientada a Objetos (POO), que organiza o design de software nos chamados objetos, ao invés de métodos e funções. Dessa maneira, é possível desenvolver uma estrutura base de manipulação de dados, e utilizar subclasses para elaboração de estruturas mais complexas e específicas no processamento de informações.

Os conceitos de herança e polimorfismo utilizados no desenvolvimento das técnicas de ETL permitem que novos processadores de dados sejam criados sob demanda e adicionados à plataforma. Com tal característica a solução se torna extensível, permitindo a elaboração de novas transformações ou a adequação das transformações existentes, para ajustar os dados às necessidades de uso da UHE.

A figura 26 apresenta o conceito de herança aplicado às funções de ETL utilizadas na plataforma proposta.

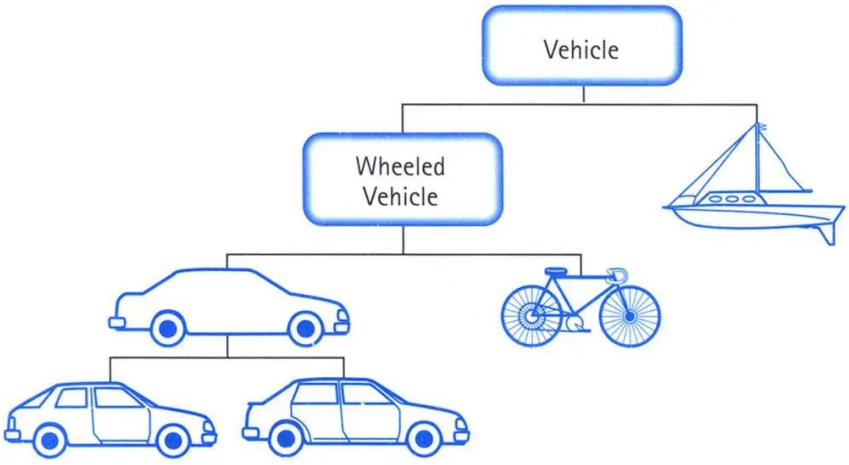

Figura 26 – POO - Herança utilizada nas funções de ETL da plataforma

Escolhida a metodologia para estruturar as ferramentas e métodos para processamento dos dados, foram utilizados os módulos e bibliotecas disponíveis na linguagem Python e no *framework* do Apache Spark para efetuar as transformações necessárias nos dados.

O módulo Spark SQL é utilizado para executar consultas SQL para leitura e processamento de dados, utilizando interfaces que fornecem um conjunto maior de informações sobre a estrutura dos dados e sobre a computação que está sendo executada. Essas informações sobre os dados possibilitam ao módulo aplicar otimizações extras durante a execução, para minimizar o tempo de processamento.

Como um dos objetivos do *framework* desenvolvido é fornecer extensibilidade, é possível utilizar módulos do Spark e Python desejados para implementar novos processadores de dados, sem que isso interfira no uso que se dá aos dados. Essa característica é importante pois, uma vez que os resultados das manipulações de dados são salvos em um banco de BI específico, que é integrado às ferramentas de visualização utilizadas na UHE, é desejável que a maneira como as informações são entregues sejam consistentes e padronizadas.

Nas figuras 27, 28 e 29 são apresentados trechos de códigos Python. A figura 27 apresenta o trecho de código onde o processamento é iniciado. Este arquivo verifica qual a fonte de informação, para então proceder com as validações necessárias, e então iniciar o processamento. A figura 28 apresenta um trecho do código utilizado como base para todas as rotinas de processamento. Conforme explicado anteriormente, o conceito de herança é utilizado para estender as funcionalidades aproveitando as rotinas base. Na figura 29 é apresentado um trecho de código especializado, que realiza um tratamento específico para tratar dados de cada UG da UHE. Este código estende o código base, e adiciona rotinas detalhadas para atender a necessidade de tratamento por UG.

A grande vantagem da extensibilidade é que é possível utilizar várias maneiras de interagir com os dados, e independentemente do método escolhido para interação, ao calcular um resultado, o mesmo mecanismo de execução é utilizado. Isso possibilita que ao implementar um novo modelo de tratamento das informações, seja possível alternar entre os diversos métodos disponíveis para interagir com os dados, de modo a utilizar a maneira mais natural de expressar uma transformação, sem que haja diferenças na performance de execução.

Os Data Frames (DF) do Spark são coleções distribuídas de dados organizados em colunas nomeadas. Conceitualmente, os DF são equivalentes às tabelas de um banco de dados relacional, porém com consideráveis otimizações de performance. Os DF são muito dinâmicos, e possibilitam sua construção a partir de uma ampla variedade de fonte de dados, tais como bancos de dados externos, arquivos estruturados, arquivos CSV e XML e DF existentes.

Em conjunto com as funções disponíveis na linguagem Python, os DFs representam uma ferramenta poderosa que possibilita a execução de inúmeras transformações nos dados, de modo simples e eficiente. A figura 30 apresenta os dados contidos no DF, e a figura 31 apresenta o Schema, que representa as informações sobre a estrutura do DF.

```
class DataProcessing:

    debug = False

    def __init__(self, args):
        self.arguments = self.getArguments(args)
        jsonData = self.readJsonFile(self.arguments.settingsFilename)
        sparkMaster = jsonData["SparkMaster"]

        spark = self.initSpark(self.arguments.sparkAppName, sparkMaster)

        self.cassandra = cc.CassandraConnector(spark, jsonData["Cassandra"], 'query')
        self.sqlServer = ss.SqlServerConnector(spark, jsonData["SqlServer"])

    def initSpark(self, appName, sparkMaster):
        return SparkSession.builder.appName(appName).master(sparkMaster).config("spark.driver.host", "1
        #.config("spark.driver.extraClassPath","mssql-jdbc-8.4.1.jre8.jar")

    def getArguments(self, arguments):
        requiredArguments = ['Caminho do arquivo de configuracoes', 'Nome da tabela', 'Modo de Gravacao

        if(len(arguments) <= 5):
            print("Informar parametros como argumentos:")
            for i, required in enumerate(requiredArguments):
                print("{}: {}".format(i, required))
            raise ValueError("Numero de argumentos invalido.")

        if(self.debug):
            print("Argumentos fornecidos:")
            for i, arg in enumerate(arguments):
                print("Argumento {}: {}".format(i, arg))

        return Arguments(arguments)
```

Figura 27 – Código Python que inicia o processamento, de acordo com a origem dos dados

Uma vez realizadas as etapas de ETL nos dados, utilizando os módulos do Apache Spark em conjunto com os módulos desenvolvidos utilizando Python, as informações processadas se encontram ainda em memória, sendo necessário disponibilizar as informações em um ambiente onde os usuários da usina poderão acessar e utilizar para elaboração de documentos e relatórios. A estrutura do Spark possibilita a utilização de inúmeros gerenciadores de banco de dados para armazenamento de informações pós processadas. A figura 32 apresenta as opções mais comumente utilizadas.

Para atender esta demanda, a plataforma proposta apresenta a opção de salvar informações em um gerenciador de banco de dados, de acordo com a necessidade e disponibilidade. Para o caso específico da UHE de Jirau, as informações foram disponibilizadas no banco de dados SQL Server, uma vez que a usina possui licença e já utiliza este gerenciador. Caso for necessário utilizar um outro gerenciador de banco de dados, é necessário realizar pequenas alterações para que a plataforma possa armazenar em um gerenciador diferente.

5.1.2 Exportação Automática e Visualização de Dados

A fim de viabilizar meios para que todo o processo de coleta, processamento e análise dos dados sejam realizados sem a necessidade de intervenção manual, foram disponibilizadas ferramentas para automatizar a tarefa. Na aplicação web, por onde são realizadas as configurações e a visualização do status das ferramentas desenvolvidas, também

```python
import pyspark.sql.functions as f
from pyspark.sql.types import DoubleType
from IPython.display import display, HTML

class ETLBase(object):

    def __init__(self):
        pass


    def processData(self, inputDF):
        df = inputDF

        self.columns = self.getAttributesNames(df)

        df = self.processPathField(df)
        df = self.pivotWithoutAgg(df)
        df = self.extractDateTimeFields(df)
        df = self.replaceColumnsNamesCharacter(df, ".", "")

        df = df.sort("equipamento", "timestamp")

        return df

    if __name__ == "__main__":
        print("Funcoes de ETL dos dados")
```

Figura 28 – Modelo base de tratamento de dados

```python
from ETLBase import ETLBase

class ETLAggregate(ETLBase):

    def __init__(self):
        super(ETLAggregate, self).__init__()

    def processData(self, df):
        groupByColumns = ['equipamento','ano','mes','dia','hora','minuto']
        aggregationFunction = "avg"

        df = super(ETLAggregate, self).processData(df)
        df = self.groupAndApplyFunction(df, self.columns, groupByColumns, aggregationFunction)
        df = self.renameColumns(df, aggregationFunction)
        df = df.sort(groupByColumns)

        return df
```

Figura 29 – Modelo específico que estende o modelo base de tratamento de dados

foram desenvolvidas telas por onde é possível definir quais dados devem ser exportados, a frequência de exportação e o modelo de processamento que deve ser utilizado.

Na figura 33 são apresentadas as exportações cadastradas, que serão efetuadas conforme agendamento realizado para cada tarefa. É possível incluir novos agendamentos, editar, excluir e parar o processo de exportação.

Além de utilizar planilhas eletrônicas para lidar com as informações resultantes do processamento efetuado pela arquitetura proposta, são utilizados *dashboards*, que utilizam gráficos e componentes visuais e textuais para exibir informações, para tornar a visualização mais amigável. O desenvolvimento dos *dashboards* foi efetuado utilizando o Power BI, porém qualquer ferramenta com esta finalidade pode ser utilizada. A escolha

path	timestamp	nome	valor
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T03:00:00Z	SimulaPotencia	69,40240786951023
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T03:24:00Z	SimulaPotencia	69,32152031330268
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T03:48:00Z	SimulaPotencia	69,35570183627993
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T04:12:00Z	SimulaPotencia	69,39717167862494
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T04:36:00Z	SimulaPotencia	69,38136009917744
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T05:00:00Z	SimulaPotencia	69,34510001465357
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T05:24:00Z	SimulaPotencia	69,3986935007773
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T05:48:00Z	SimulaPotencia	69,33355775691334
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T06:12:00Z	SimulaPotencia	69,37917534605057
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T06:36:00Z	SimulaPotencia	69,3563907932298
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T07:00:00Z	SimulaPotencia	69,35187134322793
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T07:24:00Z	SimulaPotencia	69,374250180314
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T07:48:00Z	SimulaPotencia	69,34996667662665
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T08:12:00Z	SimulaPotencia	69,40977326405658
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T08:36:00Z	SimulaPotencia	69,33907592653526
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T09:00:00Z	SimulaPotencia	69,40285848958558
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T09:24:00Z	SimulaPotencia	69,45030123869704
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T09:48:00Z	SimulaPotencia	69,34679850826836
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T10:12:00Z	SimulaPotencia	69,36401082595636
\\ESBR-UNIFEI\ESBR-nova\ESBR\U03	SimulaPotencia	2022-01-01T10:36:00Z	SimulaPotencia	69,31010779196707

Figura 30 – Visualização dos dados do Data Frame

```
etlDF.printSchema()
```

```
root
 |-- path: string (nullable = false)
 |-- timestamp: string (nullable = true)
 |-- nome: string (nullable = true)
 |-- valor: string (nullable = true)
```

Figura 31 – Schema / Estrutura do Data Frame

por esta opção foi motivada pelo fato de que a equipe de funcionários na UHE Jirau está familiarizada com esta ferramenta, além de possuírem um ambiente integrado onde os funcionários podem compartilhar e interagir dentre os *dashboards* disponíveis.

A figura 34 exibe um *dashboard* detalhando as informações de manutenção divididos por UG. De tal maneira é possível visualizar quais UGs estarão indisponíveis por período, auxiliando o operador no processo de seleção das UGs para efetuar o despacho. Ainda é possível filtrar por intervalo de datas ou por UG. Abaixo do nome da UG são exibidos os documentos de intervenção existentes para o período escolhido.

Dentre as muitas informações relacionadas ao desempenho da UHE Jirau que devem ser acompanhadas constantemente, a potência de geração alcançada pelas UGs é um dado de extrema importância. Para trazer essa informação ao conhecimento dos operadores de forma amigável, o *dashboard* apresentado na figura 35 foi desenvolvido, onde é possível visualizar as informações de potência realizada (SW-WAT), ou seja, a potência efetivamente gerada, a potência sugerida pelo otimizador desenvolvido no contexto do projeto, e a melhor potência, que representa o valor de potência que resultaria na melhor eficiência global da UHE.

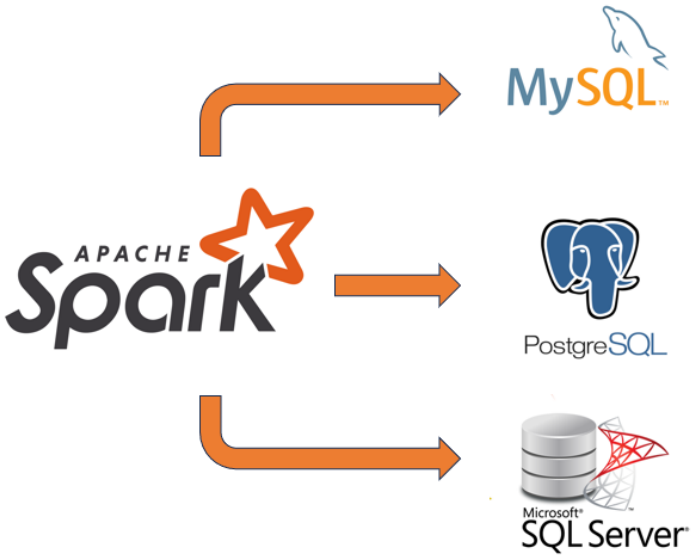

Figura 32 – Bancos de dados utilizados para armazenar informações pelo Apache Spark

Figura 33 – Tela de listagem dos agendamentos de exportações de BI

De posse das informações de medição de nível do rio e demais informações hidrológicas, é possível obter a média de queda bruta obtida. Com essa informação de média de queda bruta é possível estimar a potência máxima possível de ser obtida na UHE. A figura 36 apresenta as seguintes informações relacionadas a um determinado período: a média de geração realizada na UHE, a média de queda bruta em metros e a potência esperada dado o cenário hidrológico. Esse *dashboard* serve como guia para a operação, fornecendo uma visão geral do desempenho da UHE.

Como já mencionado anteriormente, a UHE Jirau sofre do problema de acúmulo de sedimentos, que acarretam a diminuição da queda líquida disponível, e consequentemente reduz o potencial de geração das UGs. Para mensurar o quanto esse acúmulo interfere no processo de geração, as informações apresentadas na figura 37 demonstram a geração realizada na UHE em comparação com a geração que seria possível caso não houvesse perda de carga. Essa informação possibilita justificar por qual motivo não foi possível alcançar a máxima geração dado o cenário hidrológico. No relatório apresentado é possível

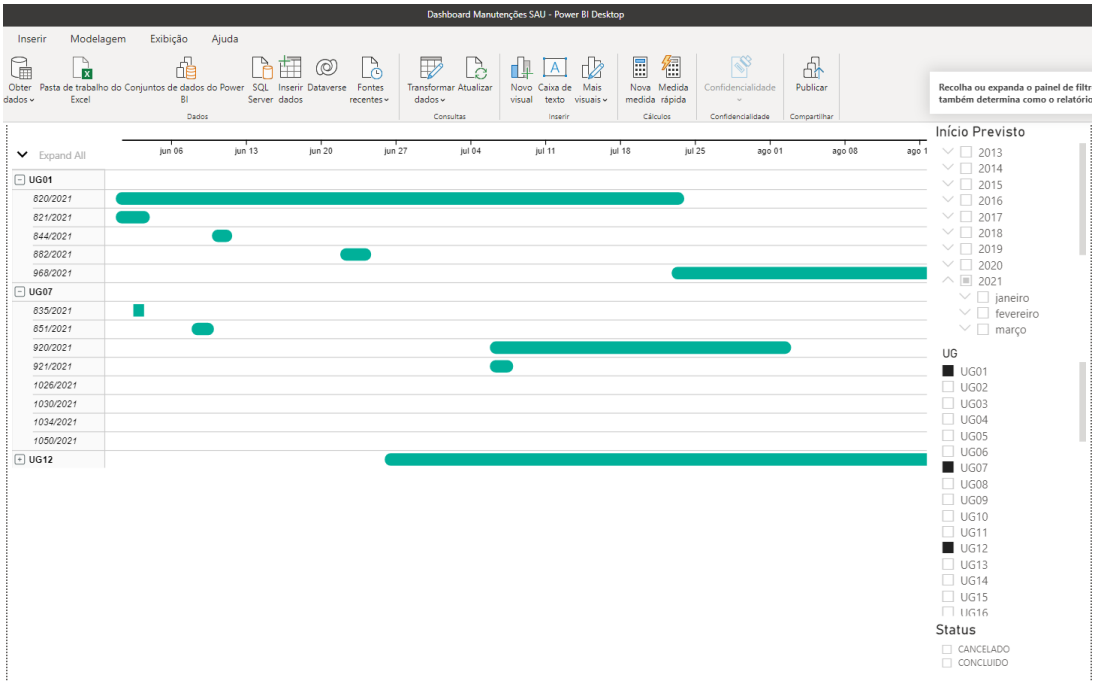

Figura 34 – Dashboard de manutenções por UG

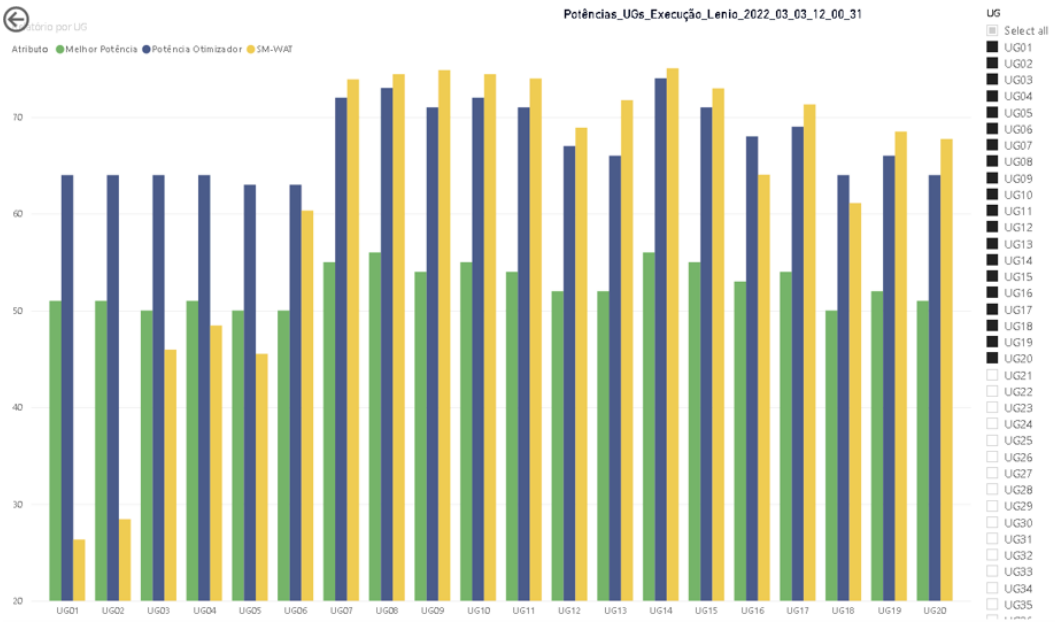

Figura 35 – Dashboard comparativo de potências

visualizar a informação relacionada ao número de UGs utilizadas para alcançar a geração apresentada e o número de UGs necessárias, ou seja, pela análise efetuada e pelo gráfico apresentado, seria possível obter o mesmo índice de geração utilizando 3 UGs a menos, caso não houvesse problemas relacionados à perda de carga. Tal informação evidencia a necessidade de atuar constantemente no problema relacionado ao acúmulo de sedimentos, visto que tais ações possibilitam uma melhora operacional.

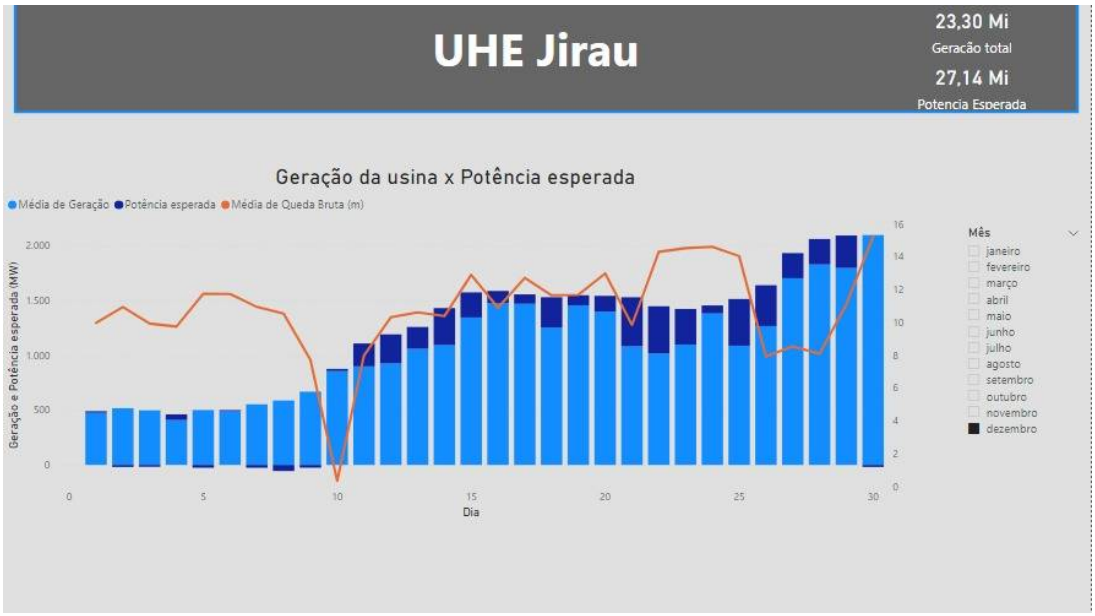

Figura 36 – Dashboard exibindo a geração da usina em comparação com a potência esperada

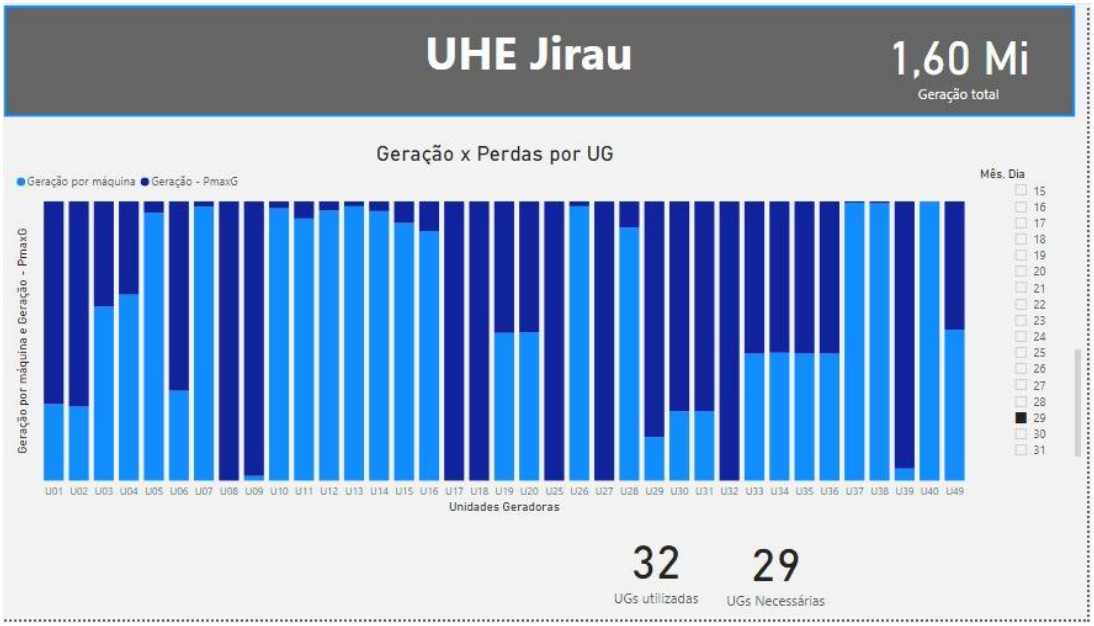

Figura 37 – Dashboard exibindo a geração realizada na usina e a geração possível caso não houvessem perdas de carga

6 Previsão do Tempo de Parada das UGs

Neste capítulo são apresentadas as informações sobre os modelos de previsão desenvolvidos com o intuito de realizar a previsão do tempo de parada necessário para que o sedimento depositado nas grades de proteção das UGs possa decantar, de modo que seja possível reestabelecer a geração energética em seu potencial máximo o mais breve possível.

Para a elaboração dos modelos de previsão foram utilizados dados coletados das UGs através dos PLCs e dos diversos sistemas da UHE Jirau. Dentre esses dados estão as informações de potência e nível de entupimento da grade (K). Somente enquanto a UG está em modo de geração é possível estimar o quão entupida a grade de proteção está; sendo assim, é imprescindível conhecer o estado operacional das UGs para saber o ponto exato em que a UG deve ser considerada parada ou em geração, para que sejam realizados os cálculos que estimam o nível de entupimento. É no momento em que a UG migra para um estado indicativo de parada da unidade que é armazenada a informação do nível de sujeira atual, para que quando a UG retome à geração, seja verificado qual é o estado posterior de sujeira estimado da unidade.

Durante a operação da UG, são realizadas medições de nível antes e depois das grades, e a diferença entre essas medições representa a perda de carga em metros. Análises de regressão utilizando as informações de potência e a queda observada da UG são realizadas para determinar a vazão de cada UG. Esta regressão é baseada na curva colina da UG. O valor de vazão obtido pela regressão e as informações de perda de carga em metros previamente registradas são usados para calcular o K, representando o fator de obstrução das grades de proteção. Um aumento no fator K indica maior acúmulo de sedimentos depositados nas grades. De acordo com a ABNT NBR 11213 (2001) a definição do fator de entupimento (K) que representa a perda de carga na passagem da grade é calculada através da equação de Kirschmer, apresentada em (6.1).

$$\Delta h_{gra} = K_{gra} . \frac{\left(\frac{q_i}{2}\right)^2}{2.A_{gra}^2.g} \tag{6.1}$$

onde:

Δh_{gra} = Perda de carga na grade [m]

K_{gra} = Coeficiente adimensional relativo a grade;

A_{gra} = Área da seção transversal da grade;

g = Gravidade;

q_i = Vazão turbinada na unidade i [m^{3}s].

Os modelos utilizam dados coletados das 50 UGs da UHE Jirau. A base de dados abrange três meses, de novembro de 2021 a janeiro de 2022, com intervalo de amostragem de 10 minutos. Na primeira tentativa de análise, foram utilizados inúmeros atributos das UGs. Devido à grande quantidade de dados relacionados aos equipamentos, utilizar todas as informações disponíveis em qualquer técnica de previsão é praticamente inviável. Essa limitação decorre do tempo necessário para o processamento dos dados e dos recursos computacionais consumidos na execução de tais tarefas.

Visando reduzir o número de atributos utilizados para a construção dos modelos, foi aplicada a técnica de correlação de Pearson. Esta técnica de correlação, ao passo que possibilita reduzir a quantidade de atributos utilizados, mantém a representatividade dos dados ao identificar os atributos de interesse mais relevantes, evidenciando o grau de dependência entre as variáveis analisadas. O coeficiente de correlação quantifica a relação entre as variáveis, com valores variando entre -1, indicando uma forte relação negativa, 1, indicando uma forte relação positiva; e zero indicando nenhum relacionamento. A fórmula do coeficiente de correlação de Pearson é expressa em (6.2).

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{6.2}$$

Após aplicar a técnica de correlação de Pearson, um conjunto reduzido de elementos foi obtido. Os atributos resultantes da análise foram: queda líquida, potência despachada, eficiência, vazão calculada e perda na grade (perda de carga ocasionada pelo acúmulo de sedimento nas grades de proteção), sendo que o último atributo representa a perda de potência de geração relacionada aos sedimentos nas acumulados nas tomadas de água. A informação de correlação entre os atributos pode ser visualizada através do mapa de calor mostrado na figura 38. É possível visualizar uma forte correlação entre queda líquida e potência despachada e vazão calculada e potência despachada e perceber que a relação entre queda líquida e vazão calculada é fraca. No entanto, a vazão calculada está fortemente correlacionada com a perda na grade.

6.1 Hidden Markov Model

Os elementos necessários para o funcionamento do HMM foram implementados, resultando assim no vetor de probabilidade inicial, a matriz de probabilidade de transição e a matriz de probabilidade de emissão. A representação dos estados utilizados pelo HMM é mapeada usando o fator K, e quatro faixas de valores são definidas para seu uso no modelo. Esses intervalos incluem o intervalo mais limpo S1, onde K varia de 0 a $1 * 10^{-6}$,

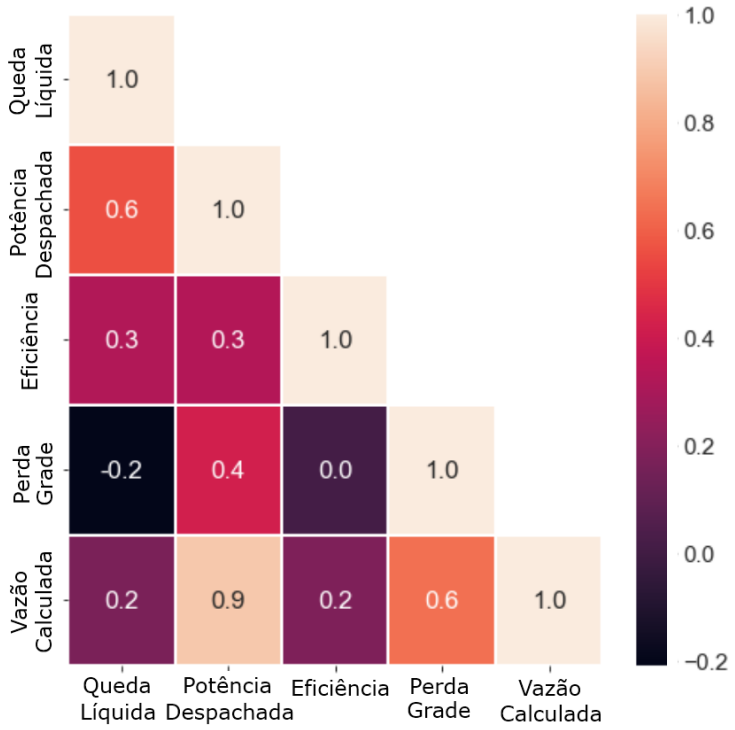

Figura 38 – Mapa de Calor de Correlação de Atributos

o intervalo S2 de $1 * 10^{-6}$ a $4 * 10^{-6}$. o S3 varia de $4 * 10^{-6}$ a $5 * 10^{-6}$ e o intervalo mais obstruído S4 para valores acima de $5 * 10^{-6}$.

Dada a perda de carga observada quando a UG foi parada, o HMM é aplicado para prever o fator de obstrução das UGs após um determinado tempo parada. Como resultado, é possível detectar uma relação de dependência entre o nível de obstrução da UG e o tempo de decantação necessário para viabilizar a retomada na geração. A aplicação de HMM permite a extração desta relação.

Durante o período de operação da UG, pode-se calcular o fator K e a perda de carga em metros. Uma vez que o nível de obstrução nas grades de lixo atinja o valor máximo suportado, o operador para a unidade para decantação.

Quatro intervalos foram criados para mapear o tempo de inatividade das UGs: intervalo H1 de 1 a 4 horas de inatividade, intervalo H2 de 4 a 8, intervalo H3 de 8 a 12 e intervalo H4 acima de 12 horas.

Dado o nível de obstrução no momento da parada da unidade, o HMM apresenta as probabilidades da UG de estar em cada faixa de nível de obstrução mapeado ao longo do tempo. É importante ressaltar que o vetor de probabilidade, as matrizes de transição e de emissões foram derivadas de dados históricos da UHE para que os resultados do modelo reflitam a realidade da parada da UG para decantação.

Os dados históricos são divididos em dois conjuntos, um representando o conjunto de treinamento e outro o conjunto de teste. A separação é necessária para avaliar o

modelo utilizando um grupo diferente daquele utilizado no treinamento, evitando assim o *overfitting* dos dados.

Na UHE é impossível determinar o nível atual de obstrução da UG após algumas horas com a UG parada para decantação. Portanto, o HMM é utilizado neste cenário para estimar o nível de obstrução da UG através de probabilidades para determinar se é possível reiniciar a operação da unidade. Por tal razão, o nível de obstrução é considerado como o estado oculto não observável do HMM. O decorrer do tempo em que a UG fica parada é tratado como a emissão, que é observável, e a relação entre tempo de parada e nível futuro de obstrução é o resultado desejado ao se utilizar o HMM.

Para a criação do modelo, as seguintes informações são inferidas dos dados históricos: o vetor de probabilidade anterior ou inicial, a probabilidade de transição e as matrizes de probabilidade de emissão.

O vetor de probabilidade inicial denota a probabilidade da UG estar em um estado de obstrução inicial específico, servindo para determinar o estado inicial mais provável para a UG. As probabilidades iniciais são definidas com base na relação entre o número de paradas para decantação e o nível de obstrução quando a UG foi parada. Consequentemente, o vetor de distribuição de estado inicial obtido, apresentado na Tabela 5, confirma a observação esperada de que as paradas para decantação da UG são mais frequentes quando o nível de obstrução é maior.

Tabela 5 – Vetor de Distribuição de Estado Inicial

$$\pi = \begin{array}{cccc} \text{S1} & \text{S2} & \text{S3} & \text{S4} \\ \hline 0.08 & 0.12 & 0.30 & 0.50 \end{array}$$

A matriz de probabilidade de transição representa a probabilidade da UG fazer a transição de um estado para outro. À medida que o tempo de inatividade da UG aumenta, há uma probabilidade maior de transição de um estado de obstrução maior para um estado menor. Como o nível de obstrução atual não é diretamente observável, é considerado um estado oculto. A matriz de probabilidade de transição resultante pode ser vista na Tabela 6.

Tabela 6 – Matriz de Probabilidade de Transição

$$A = \begin{array}{c|cccc} & \text{S1} & \text{S2} & \text{S3} & \text{S4} \\ \hline \text{S1} & 0.80 & 0.11 & 0.08 & 0.01 \\ \text{S2} & 0.48 & 0.44 & 0.01 & 0.07 \\ \text{S3} & 0.43 & 0.42 & 0.10 & 0.05 \\ \text{S4} & 0.15 & 0.44 & 0.06 & 0.34 \end{array}$$

Por outro lado, a matriz de probabilidade de emissão corresponde à informação observada relativa ao nível de obstrução atual da UG. Esta informação é o tempo de

decantação decorrido desde que a UG foi parada. Com o passar do tempo, as grades da UG tendem a ficam mais limpas, o que atende às expectativas. Assim, o HMM utiliza o vetor de probabilidade e matrizes para estimar o nível de obstrução da UG após um tempo aleatório. A tabela 7 apresenta a matriz de probabilidade de emissão resultante.

Tabela 7 – Matriz de Probabilidade de Emissão

$B =$

	H1	H2	H3	H4	H5	H6
S1	0.11	0.066	0.198	0.077	0.022	0.527
S2	0.088	0.099	0.198	0.187	0.033	0.396
S3	0.053	0.105	0.105	0.211	0.053	0.474
S4	0.279	0.131	0.158	0.153	0.049	0.23

A implementação do HMM é realizada utilizando o Pomegranate, um pacote Python para modelos probabilísticos (Jacob Schreiber, 2023). A construção do modelo envolveu as três entidades: π, A e B. Uma vez que o modelo é treinado, dada uma sequência de observações O, o modelo determina uma pontuação para a sequência observada usando o chamado *Forward Algorithm*, ou α-*pass*, usando o conceito de programação dinâmica.

Após obter a pontuação da sequência observada, o próximo passo é revelar a sequência de estados mais provável dadas as observações apresentadas. Dado o tempo decorrido da parada da UG, o algoritmo *Viterbi* é usado para expor os estados ocultos, representando o fator K atual. O algoritmo *Viterbi* gera a sequência mais provável de estados ocultos para uma determinada lista de observações, usando programação dinâmica para gerar a sequência de saída recursivamente.

6.2 Redes Bayesianas - RBs

As RBs oferecem a possibilidade de representar um problema de determinado domínio através de uma estrutura gráfica composta por nós que compreendem um conjunto de variáveis de domínio aleatórias. Nesta estrutura gráfica, os arcos conectam os nós em pares, o que representa a dependência direta entre as variáveis. A distribuição de probabilidade condicional de cada nó associado governa a força do relacionamento entre as variáveis.

A utilização de RBs neste trabalho permite representar as variáveis que afetam diretamente o tempo de parada necessário para diminuir o nível de obstrução das grades de proteção das UGs, através do processo de decantação.

As seguintes variáveis são consideradas na modelagem da RB: fator K antes da unidade parar, a indicação se a UG esquerda, a direita ou ambas as vizinhas estão em operação durante o tempo de parada da UG analisada, e a potência de operação das

unidades vizinhas, se for o caso. O diagrama da RB resultante, projetado para refletir informações sobre as UGs, é mostrado na figura 39.

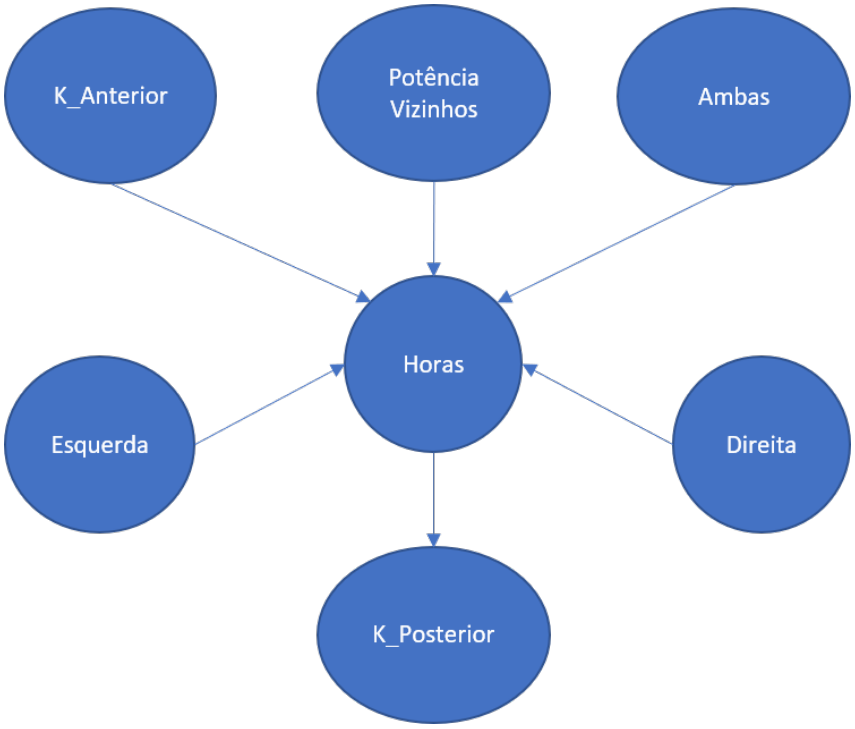

Figura 39 – Diagrama da Rede Bayesiana

O modelo da RB pode ser consultado por uma ou mais variáveis que fazem parte da modelagem, obtendo-se a probabilidade condicional de acordo com as entradas fornecidas.

Para contabilizar o tempo de operação das vizinhas enquanto a UG analisada está parada, todos os intervalos de cada uma das vizinhas em funcionamento são somados para obter a relação entre o tempo de funcionamento dos vizinhos pelo tempo de parada da UG analisada. Por exemplo, se a UG analisada fica parada por 12 horas, e a vizinha direita opera por 3 horas, então a vizinha opera por 25% do tempo de parada da UG analisada. Portanto, o tempo de operação das UGs vizinhas foi dividido em quatro faixas, a faixa T1 até 25% do tempo, T2 de 25% a 50%, faixa T3 de 50% a 75%, e faixa T4 de 75% a 100%.

A média é usada para calcular a potência de operação das vizinhas durante as horas de uso da UG esquerda ou direita. Se ambas os vizinhas estiverem operacionais, será utilizada a potência de geração média para cada vizinho. A informação de potência foi discretizada em 4 faixas, sendo a faixa Pot1 até 20 MW de potência, a faixa Pot2 de 20 MW a 40 MW, a faixa Pot3 de 40 MW a 55 MW e a última faixa Pot4 acima de 40 MW.

Finalmente, os mesmos quatro intervalos de tempo (T1, T2, T3 e T4) são usados para inferir as informações necessárias sobre o tempo de parada da UG.

As distribuições de probabilidades condicionais (CPD) relacionadas a cada variável do modelo são obtidas através do aprendizado de parâmetros usando os dados fornecidos e a estrutura do modelo. O algoritmo Maximum Likelihood Estimation (MLE) é usado neste trabalho para extração de CPD usando um conjunto de dados (KJAERULFF; MADSEN, 2008; KOLLER; FRIEDMAN, 2009).

O modelo Bayesiano e as CPDs fazem inferências utilizando diversos cenários para validar a técnica proposta e comparar os resultados obtidos do modelo com os dados operacionais da planta.

7 Resultados e Discussões

7.1 Resultados

Após a elaboração dos modelos propostos utilizando as técnicas apresentadas, são realizados estudos de casos para verificar seu desempenho. Nesta seção são apresentados os resultados obtidos. Inicialmente serão evidenciados os resultados utilizando HMM, e posteriormente serão apresentados os resultados com o uso de RBs.

Com o intuito de facilitar a análise dos resultados, é apresentada na Tabela 8 listando as faixas valores para cada atributo utilizado.

Tabela 8 – Faixas de valores para os atributos utilizados

Tempo de Inatividade	Intervalo
H1	1 a 4 horas
H2	4 a 8 horas
H3	8 a 12 horas
H4	Acima de 12 horas

Tempo de Operação	Intervalo
T1	Até 25%
T2	25 a 50%
T3	50 a 75%
T4	75 a 100%

Potência de Operação	Intervalo
Pot1	Até 20 MW
Pot2	20 a 40 MW
Pot3	40 a 55 MW
Pot4	Acima de 55 MW

Nível de Obstrução	Intervalo
S1	0 a $1 * 10^{-6}$
S2	$1 * 10^{-6}$ a $4 * 10^{-6}$
S3	$4 * 10^{-6}$ a $5 * 10^{-6}$
S4	Acima de $5 * 10^{-6}$

7.1.1 Hidden Markov Models

Através da aplicação do HMM os resultados obtidos para UG 2 são apresentados na Tabela 9 e fornecem uma representação abrangente das probabilidades associadas a estados ocultos específicos. Estes estados ocultos correspondem a vários níveis de obstrução causados pela acumulação de sedimentos. A probabilidade da UG estar em estados

ocultos distintos pode ser verificada analisando os dados dentro de cada intervalo horário observado.

Tabela 9 – Resultados do HMM para a UG 2: Probabilidade de estado após determinado tempo decorrido. As colunas representam os níveis de obstrução e as linhas representam o tempo decorrido

	S1	S2	S3	S4
H1	48.4	38.5	42.1	41.0
H2	26.9	35.2	31.6	31.1
H3	17.2	18.7	15.8	13.1
H4	7.5	7.6	10.5	14.8

Sempre que a UG está no estado *S4*, o que significa que possui o maior nível de obstrução, a probabilidade de transição para um estado menos obstruído torna-se evidente somente após o intervalo de tempo *H3*. Este resultado está alinhado com a prática operacional da UHE, que costuma manter a unidade parada por longos períodos quando o nível de obstrução atinge um grau mais elevado.

De maneira alternativa, se a mesma UG possui o nível de obstrução atual *S2* e a UG permanece parada pelo intervalo de tempo *H3*, há uma probabilidade significativamente maior (18,7%) de que esta permaneça nesse estado. A transição da UG para um estado mais limpo que o *S2* exige um tempo de parada mais prolongado devido às características do processo de sedimentação, na qual materiais mais densos demoram mais para assentar.

No nível de obstrução *S1* a unidade pode permanecer no mesmo estado, ou às vezes foi identificada a evolução para um nível de obstrução superior, passando para *S2*. Este evento pode ocorrer devido à operação de UGs vizinhas, o que contribui para a movimentação de sedimentos, migrando material para a UG parada.

Em um cenário ideal para a operação da UHE, uma UG no nível de obstrução mais alto deverá permanecer parada até atingir o nível de sujeira mais baixo, quando poderá retornar à atividade. Um estudo de caso foi realizado para obter a probabilidade da UG migrar do estado inicial *S4* para *S1* como seu estado final. O resultado é apresentado na Tabela 10.

É possível notar que o cenário fornecido tem maior probabilidade de ocorrer somente após o *H3*, com probabilidade de 24,99%, e é mais provável, com 30,20%, em *H4*. Como esperado, não é comum chegar a *S1* a partir de *S4* após os períodos *H1* ou *H2*, correspondentes a intervalos de 1 a 4 horas e de 4 a 8 horas, respectivamente.

Para analisar as diferenças de tempo de parada entre as diferentes UGs, a Tabela 11 apresenta informações relativas ao nível de obstrução e ao tempo de inatividade para as UGs 1 e 3, respectivamente.

As seguintes considerações podem ser feitas através da análise dos dados e uti-

Tabela 10 – Probabilidade da UG 2 migrar do estado inicial *S4* para *S1* como seu estado final

	S4 → S1
H1	7.80
H2	18.25
H3	24.99
H4	30.20

Tabela 11 – Resultados do HMM para as UGs 1 e 3: Probabilidade de estado após determinado tempo decorrido. As colunas representam os níveis de obstrução e as linhas representam o tempo decorrido

	UG 1				UG 3			
	S1	S2	S3	S4	S1	S2	S3	S4
H1	62.0	34.1	27.3	28.1	59.0	37.5	28.9	31.1
H2	32.4	24.5	26.9	25.9	35.2	26.4	26.4	28.1
H3	3.7	22.9	25.6	23.4	4.1	15.1	23.9	20.9
H4	1.9	18.5	20.2	22.6	1.7	21.0	20.8	19.9

lizando o mesmo estudo de caso realizado na UG 2, para obter a probabilidade da UG migrar do estado inicial *S4* para *S1* como seu estado final:

Para a UG 1, a probabilidade de permanecer no estado *S4* mostra uma dispersão equilibrada ao longo do tempo. Este padrão indica situações onde a UG transita para um estado mais limpo mesmo dentro do intervalo *H1*. Por outro lado, há casos em que um período de tempo maior, como *H4*, é necessário para a transição. Esta variação pode ser atribuída à proximidade da UG 1 com a margem do rio, o que representa uma potencial contribuição para a acumulação de sedimentos, visto que é uma área onde ocorre uma maior concentração de sujeira.

Como observações relativas à análise efetuada na UG 3, a probabilidade de persistir no estado *S4* durante os intervalos *H1* e *H2* é maior, registando valores de 31,1% e 28,1%, respetivamente. A tendência predominante é que a UG 3 faça a transição para um estado mais limpo somente após o intervalo *H3*.

Os comportamentos diferentes observados entre as UGs 2 e 3 podem ser atribuídos aos seguintes fatores: a UG 1 absorve sedimentos da margem do rio e pode, consequentemente, transferir sedimentos para a UG 2, explicando porque a UG 2 muda para um estado mais limpo somente após um tempo de parada mais prolongado. Por outro lado, a UG 3 não é afetada pelo mesmo problema devido a sua maior distância da UG 1, ilustrando o impacto das UGs vizinhas no processo de decantação de sedimentos.

Os resultados de probabilidade para as UGs 31 e 32 são mostrados na Tabela 12.

O comportamento das UGs 31 e 32 difere daqueles apresentados para as UGs 1 a 3. Essa diferença pode ser atribuída ao fato dessas UGs estarem em margens diferentes,

Tabela 12 – Resultados do HMM para as UGs 31 e 32: Probabilidade de estado após decorrido o tempo. As colunas representam os níveis de obstrução e as linhas representam o tempo decorrido

	UG 31				UG 32			
	S1	S2	S3	S4	S1	S2	S3	S4
H1	35.4	38.5	42.1	41.0	46.9	44.3	45.7	43.2
H2	26.9	35.2	31.6	31.1	32.7	31.9	36.1	34.0
H3	27.2	18.7	15.8	13.1	15.2	17.6	11.1	12.7
H4	10.5	7.6	10.5	14.8	5.2	6.2	7.1	10.1

separadas por quilômetros, e à curvatura do rio apresentada na margem esquerda onde essas UGs estão instaladas. É possível observar certa semelhança entre as probabilidades para as UGs 31 e 32, com pequenas variações no tempo necessário para mudança entre os estados. Geralmente há migração entre estados somente após o intervalo *H3*, o que pode estar associado ao tipo de material acumulado nas grades de proteção.

Quando as UGs apresentam um grau de obstrução notavelmente alto, surge uma tendência predominante: a limpeza substancial das grades de proteção ocorre somente após um tempo prolongado de inatividade da UG. Especificamente, se a UG for parada por um curto período de tempo e logo em seguida retomar sua operação, espera-se que uma quantidade considerável de material ainda obstrua de modo persistente as grades de proteção.

As observações revelam que as unidades posicionadas perto da margem do rio apresentam um acúmulo de sedimentos notavelmente maior, levando a uma obstrução mais pronunciada das grades de proteção. As UGs adjacentes também sofrem um efeito residual desta acumulação de sedimentos, embora com um impacto menor.

Outra UG que apresenta alto índice de acúmulo de material depositado é a UG 29, que fica na ombreira direita da casa de força 2 da UHE. A sua localização, cujo o escoamento do rio tem menor velocidade, favorece o acúmulo acentuado de sedimentos, troncos, árvores e macrófitas nessa unidade. Um comparativo entre a UG 29 e sua vizinha, a UG 30, é apresentado na tabela 13.

Tabela 13 – Resultados do HMM para as UGs 29 e 30: Probabilidade de estado após determinado tempo decorrido. As colunas representam os níveis de obstrução e as linhas representam o tempo decorrido

	UG 29				UG 30			
	S1	S2	S3	S4	S1	S2	S3	S4
H1	36.1	29.9	29.9	32.4	43.9	39.1	39.4	46.1
H2	28.4	25.1	25.1	29.3	31.8	30.2	38.2	32.7
H3	21.2	23.1	23.1	21.1	16.2	17.4	18.7	17.3
H4	14.3	21.9	21.9	17.2	8.1	13.3	3.7	3.9

Pelo resultado apresentado, é possível perceber que a UG 29 apresenta probabilidades distribuídas, o que significa que mesmo com o decorrer do tempo, não há garantias de mudanças no nível de acúmulo de sedimentos, e consequentemente limpeza das grades.

Caso a UG 29 esteja no nível de sujeira *S4*, a probabilidade de migrar para um estado mais limpo só é considerável a partir de *H4*, com o valor de 17,2%. A distribuição de probabilidade para os intervalos *H1* até *H3* não são tão distantes se comparados com o mesmo cenário para a UG 30, onde a partir de *H3* há maiores chances de migrar para um estado mais limpo, e a partir de *H4* é muito provável que esta mudança de estado ocorra.

Outra informação obtida do resultado apresentado, em um cenário em que a UG 29 esteja no nível de sujeira *S1* é a de que é possível que a UG mude para um estado mais sujo mesmo após um tempo de parada *H3*, com o valor de probabilidade de 21,2%. Tal situação pode ocorrer devido à proximidade da UG de uma área onde há grande acúmulo de sujeira, de modo que mesmo com o passar do tempo, a sujeira permanece nas proximidades das grades de proteção.

A probabilidade de que a UG 30 migre de um estado mais limpo *S1* para um estado mais sujo *S2*, o que é um estado indesejável para a operação da usina, é menor, com valores de 16,2% após *H3*, e 8,1% após *H4*.

Por fim, para os estados de sujeira *S2* e *S3*, a UG 29 apresenta pouca variação na probabilidade de mudança a medida que o tempo decorre, o que demonstra que a operação desta UG é bastante comprometida devido ao acúmulo de sedimentos. As probabilidades de migrar do estado *S2* para outro estado a partir de *H1*, *H2* e *H3* são 29,9%, 25,1% e 23,1% respectivamente. Vale lembrar que não é possível afirmar, embora provável, que a migração ocorra obrigatoriamente para um estado mais limpo.

Na tabela 14 são apresentadas as informações comparando os resultados para as UGs 30 e 31. Este comparativo visa destacar que, assim como ocorre com a UG 3, que está distante da UG 1, a UG 31 é pouco influenciada pelo grande acúmulo de sedimentos que ocorre na UG 29.

Tabela 14 – Resultados comparativos do HMM para as UGs 30 e 31: Probabilidade de estado após determinado tempo decorrido. As colunas representam os níveis de obstrução e as linhas representam o tempo decorrido

	UG 30				UG 31			
	S1	S2	S3	S4	S1	S2	S3	S4
H1	43.9	39.1	39.4	46.1	35.4	38.5	42.1	41.0
H2	31.8	30.2	38.2	32.7	26.9	35.2	31.6	31.1
H3	16.2	17.4	18.7	17.3	27.2	18.7	15.8	13.1
H4	8.1	13.3	3.7	3.9	10.5	7.6	10.5	14.8

No comparativo percebe-se que a partir dos estados de sujeira *S3* e *S4*, nos tempos

de parada *H1*, *H2* e *H3*, há muita semelhança nas probabilidades das UGs 30 e 31.

Um ponto importante a ressaltar é que a UG 30, que sofre influência direta da UG 29, tem probabilidades que indicam uma migração para um estado mais limpo a partir de *H4* para os estados *S3* e *S4*, se comparado à UG 31. Tal indicativo pode ser atribuído ao fato de que a operação da UG 29 provavelmente realiza a movimentação dos sedimentos acumulados na sua vizinha, contribuindo ainda mais para manter a UG 29 suja, ao mesmo tempo que acaba limpando a UG 30. Esta relação depende de como foi o funcionamento das vizinhas enquanto a UG estava parada para decantação.

É fundamental ressaltar que a técnica HMM não considera se as UGs vizinhas estão em operação, nem contabiliza a potência de operação dessas UGs ou o tempo em que estavam em modo de geração enquanto determinada UG está parada para decantação. Sendo assim, fazendo uso de HMM, os resultados para uma mesma UG pode sofrer variações nos resultados, dependendo da configuração das UGs vizinhas durante o período de inatividade da UG em análise.

Embora os operadores da planta estejam cientes de que tais fatores influenciam diretamente no nível de sujeira das UGs, os resultados deste trabalho ajudam a compreender e revelar a importância desta influência de acordo com a UG ou condições de operação de cada margem.

Por esse motivo, as RBs são utilizadas para considerar os fatores que afetam diretamente a operação e consequentemente alteram o fluxo de sedimentos durante o tempo de parada da UG. RBs separadas são criadas para cada unidade com o intuito de refletir as especificidades de cada uma e também o modo de operação em cada situação.

7.1.2 Redes Bayesianas

A seguir são apresentados os resultados obtidos para diferentes tipos de consultas de RBs. As informações demonstram a probabilidade de migração das UGs entre os níveis de sujeira de acordo com o decorrer do tempo.

São apresentados cenários em que a UG esquerda, direita ou ambas estão em operação, qual o nível de sujeira no momento em que a UG foi parada, e também qual o nível de sujeira esperado quando a UG retomar a operação. Por fim, também são especificadas as potências de operação das UGs vizinhas àquelas que estão sendo analisadas.

A Tabela 15 mostra as distribuições de probabilidades condicionais (CPDs) para UG 2.

Na tabela pode ser visualizado que a partir do nível de obstrução *S1*, após o intervalo de tempo *H1* é provável que a UG permaneça neste estado, com probabilidade de 81,3%, ao passo que a partir do intervalo *H3* esta probabilidade é de apenas 5,3%. Quando

Tabela 15 – Resultados da RB para UG 2: Probabilidade de estado após determinado tempo decorrido. As colunas representam os níveis de obstruçãoe as linhas representam o tempo decorrido

	S1	S2	S3	S4
H1	81.3	75.3	59.0	51.0
H2	12.1	17.0	25.5	23.2
H3	5.3	5.4	10.5	16.0
H4	1.3	2.3	5.0	9.8

a UG se encontra no nível de obstrução *S4*, portanto o nível mais sujo, a probabilidade de que após *H1* ainda permaneça neste estado é de 51,0%, sendo mais provável que a partir de *H3* a UG migre para um estado mais limpo, com o valor de 16,0%.

A utilização de modelos derivados de RBs oferece uma vantagem significativa devido às suas capacidades inerentes de consulta. O modelo possibilita que consultas sejam efetuadas informando quaisquer atributos utilizados e que estejam mapeados na rede, facilitando assim a previsão de valores posteriores. Especificamente, estes modelos permitem a previsão do nível de obstrução para cada intervalo de tempo de inatividade distinto.

Esta capacidade preditiva aumenta a capacidade de estimar e antecipar a progressão dos níveis de obstrução durante vários tempos de inatividade operacional. Em essência, as RBs permitem uma exploração abrangente dos atributos da rede, permitindo a geração de insights valiosos sobre o comportamento esperado do sistema ao longo do tempo.

Utilizando a RB mostrada na figura 39, é possível estimar o nível de obstrução resultante utilizando um determinado cenário, para verificar quais parâmetros influenciam mais o processo de decantação.

Abaixo são listados os casos de testes utilizados, apresentando os valores inseridos para os parâmetros da RB.

- A = [K_Anterior: *S4*]
- B = [K_Anterior: *S4*, Direita: *T3*]
- C = [K_Anterior: *S4*, Direita: *T3*, Esquerda: *T1*]
- D = [K_Anterior: *S4*, Ambas: *T4*]

Foram utilizados dados referentes a UG 2. Em todos os cenários considera-se que a UG está no nível mais alto de obstrução, *S4*.

A figura 40 apresenta os casos A e B, utilizando o diagrama da rede Bayesiana.

O cenário A é parametrizado apenas com a informação de obstrução *S4*. O cenário B é configurado com a informação adicional 'Direita', cujo valor definido é *T3*, que compreende o valor que varia de 50% a 75% do tempo.

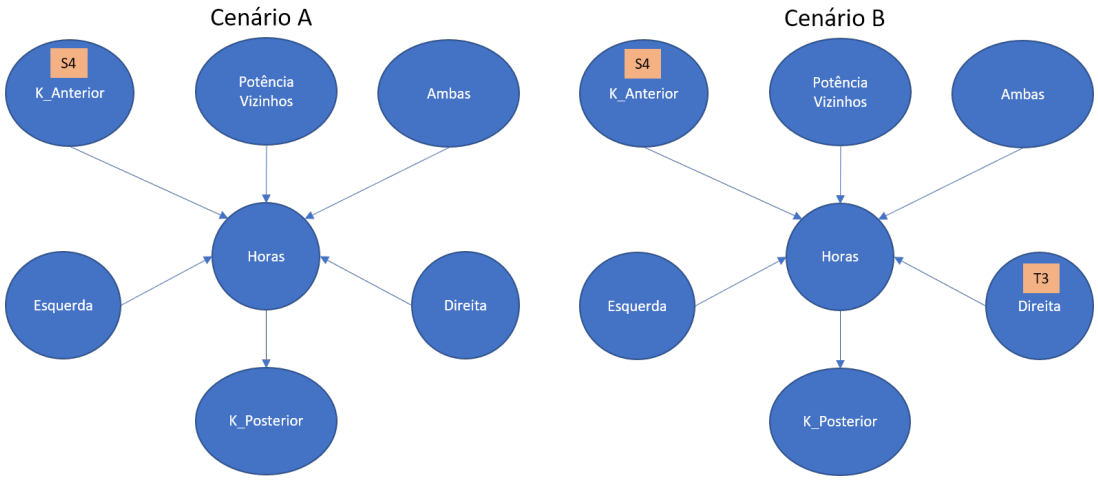

Figura 40 – Diagrama da Rede Bayesiana para os casos A e B

As informações 'Direita', 'Esquerda' ou 'Ambas' referem-se ao percentual de tempo que a unidade adjacente operou quando a UG foi parada para decantação. Neste caso do cenário B, a UG analisada é a 2, e a UG vizinha 'Direita' é a 3.

Os casos C e D são apresentados na figura 41, também utilizando a representação da rede Bayesiana.

O cenário C está configurado com o mesmo valor para o parâmetro 'Direita': *T3*. Além disso, a informação 'Esquerda' é definida como *T1*, que compreende o valor até 25% do tempo.

O cenário D está configurado com o valor do parâmetro 'Ambas' igual a *T4*, o que significa que ambas as UGs, 'Direita' e 'Esquerda', neste caso, 3 e 4, respectivamente, operaram durante o intervalo de tempo compreendendo os 75% a 100% do tempo em que a UG 2 ficou parada para decantação.

Os resultados com as CPDs para os cenários analisados são apresentados na Tabela 16.

Tabela 16 – Resultados da RB para UG 2: Probabilidades obtidas para os cenários A, B, C e D, utilizando *S4* como nível de obstrução inicial

	A	B	C	D
S1	9.4	8.0	40.1	48.4
S2	38.8	32.3	27.0	35.2
S3	32.9	30.8	25.9	42.1
S4	18.8	28.9	6.9	14.8

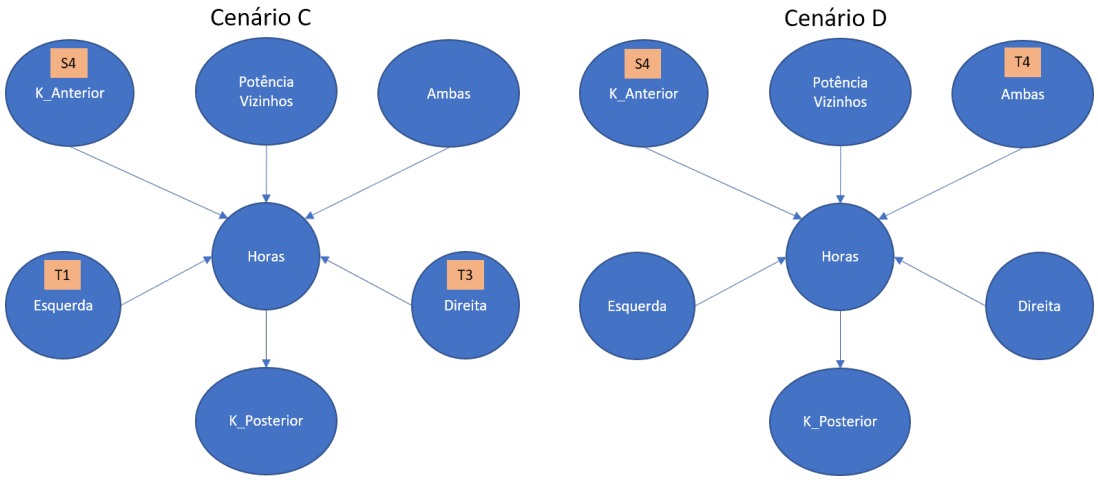

Figura 41 – Diagrama da Rede Bayesiana para os casos C e D

No cenário A, a probabilidade da UG 2, parada no pior nível de obstrução, retomar a operação nos níveis *S2* e *S3* é de aproximadamente 38,8% e 32,9%, respectivamente. No cenário B, esses valores estão próximos, 32,3% e 30,8%, respectivamente. A operação da UG vizinha, no caso, a UG 3, não impactou significativamente no nível de obstrução da UG 2.

No cenário C foram obtidos os resultados de limpeza mais favoráveis, com 40,1% de probabilidade da UG retornar no nível de obstrução mais limpo *S1*. A provável explicação para tal comportamento pode ser que a operação da UG esquerda, no caso, a UG 1, tenha puxado o sedimento da UG 2, migrando a UG mais rapidamente para um nível de obstrução mais baixo.

Finalmente, no cenário D, ambas as vizinhas estiveram em operação durante todo o tempo em que a UG 2 esteve parada. Os resultados demonstram uma probabilidade mais uniformemente distribuída entre os níveis S1, S2 e S3, com valores de 48,4%, 35,2% e 42,1%, respectivamente.

Os resultados mostram que o processo de decantação quando a UG está parada é significativamente influenciado pelas UGs vizinhas. Essa relação muda dependendo do tempo em que as UGs vizinhas estavam operando e do nível de obstrução quando a UG parou.

A RB foi parametrizada para apresentar as saídas do modelo para cada nível de obstrução final ao retomar a operação da UG para todos os tempos de parada disponíveis, permitindo uma visão mais abrangente dos dados, incluindo resultados de probabilidade mais completos. Os resultados obtidos são mostrados na Tabela 17.

Dado que a UG foi parada no nível de obstrução mais alto *S4*, o seguinte comportamento pode ser percebido pelas análises da tabela, para diversos cenários: após o tempo de parada *H1*, a maior probabilidade é que a UG retome a operação ainda em nível

Tabela 17 – Resultados da RB para UG 2: Para cada cenário de A a D, e para cada intervalo de tempo de *H1* a *H4*, as probabilidades da UG retornar à operação em níveis de sujeira de *S1* a *S4* são apresentados, usando *S4* como nível de obstrução inicial

Horas	K Posterior	A	B	C	D
H1	S1	5.6	3.3	2.9	3.9
H1	S2	5.6	3.3	2.9	5.9
H1	S3	13.9	8.3	7.2	9.7
H1	S4	16.7	10.0	8.7	11.7
H2	S1	1.8	1.9	1.7	1.9
H2	S2	1.9	2.0	1.7	1.9
H2	S3	9.4	9.6	8.3	9.6
H2	S4	11.3	11.5	9.9	11.5
H3	S1	1.9	2.8	2.4	2.3
H3	S2	9.6	13.9	12.0	11.4
H3	S3	1.9	2.8	2.4	2.3
H3	S4	3.8	5.6	4.8	4.6
H4	S1	9.5	14.3	20.0	14.5
H4	S2	4.9	7.9	2.1	8.1
H4	S3	1.0	1.0	10.5	1.0
H4	S4	1.2	1.8	2.5	1.8

S4. Para o tempo *H3*, o reinício deve ocorrer no nível *S2*, e por fim, a parada durante o período de tempo *H4* aumenta a probabilidade de retomada no nível *S1*.

Somente no intervalo de tempo de parada *H2* esse padrão não se mantém. Em vez de retomar no nível *S3*, a UG permanece no nível *S4*, demonstrando que parar a UG por curtos intervalos de tempo não influencia tão fortemente o nível de obstrução.

A seguir são definidos outros cenários para analisar o comportamento das UGs 37 e 38.

- E = [K_Anterior: *S3*]
- F = [K_Anterior: *S3*, Esquerda: *T1*]
- G = [K_Anterior: *S3*, Esquerda: *T2*, Direita: *T3*]
- H = [K_Anterior: *S3*, Ambas: *T2*, Potência Vizinhos: *Pot1*]
- I = [K_Anterior: *S3*, Ambas: *T4*, Potência Vizinhos: *Pot3*]

A figura 42 apresenta os casos E e F, utilizando o diagrama da rede Bayesiana.

O cenário E é parametrizado com a informação de obstrução *S3*. Além desta informação, o cenário F também define o valor 'Esquerda' igual a *T1*, que compreende o valor até 25% do tempo.

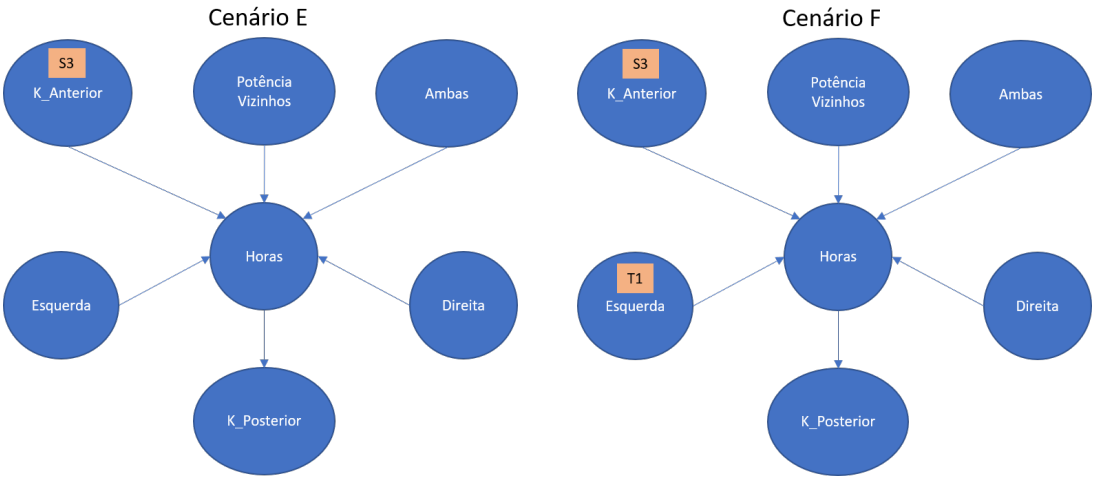

Figura 42 – Diagrama da Rede Bayesiana para os casos E e F

Como são utilizadas duas UGs adjacentes, quando a UG 37 está sendo analisada, a 'Direita' compreende a UG 38, e no momento da análise da UG 38, a 'Esquerda' compreende a UG 37. Esse cenário visa verificar os resultados probabilísticos de modo a evidenciar a influência das UGs vizinhas em operação, enquanto determinada UG está parada para decantação. Os cenários compreendem horários de análise diferentes, ou seja, momentos distintos em que uma ou outra das UGs analisadas estava parada.

Os casos G e H são apresentados na figura 43, também utilizando a representação da rede Bayesiana.

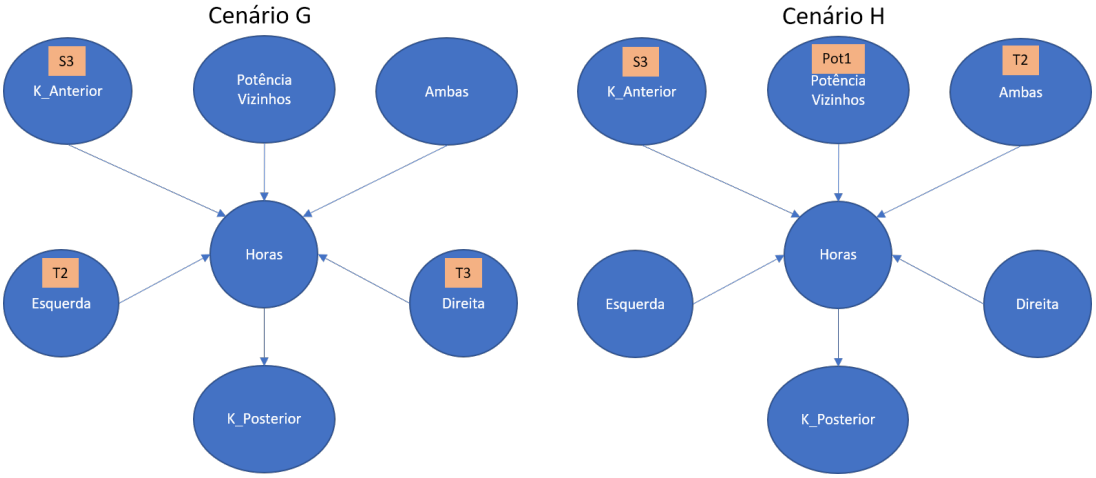

Figura 43 – Diagrama da Rede Bayesiana para os casos G e H

O cenário G está configurado com o parâmetro 'Direita' igual a *T3*, que compreende o valor entre 50% e 75% do tempo, e a informação 'Esquerda' como *T2*, que compreende o valor entre 25% e 50% do tempo.

O cenário H está configurado com o valor do parâmetro 'Ambas' igual a *T2*, o que significa que tanto a UG 'Direita' quanto a 'Esquerda' operaram durante o intervalo de

tempo em que a UG em análise ficou parada para decantação.

Por fim, a figura 44 apresenta o caso I, configurado com o valor do parâmetro 'Ambas' igual a *T4* e 'Potência Vizinhos' igual a *Pot3*, o que significa que a média de potência de operação durante o período está entre 40 a 55 MW.

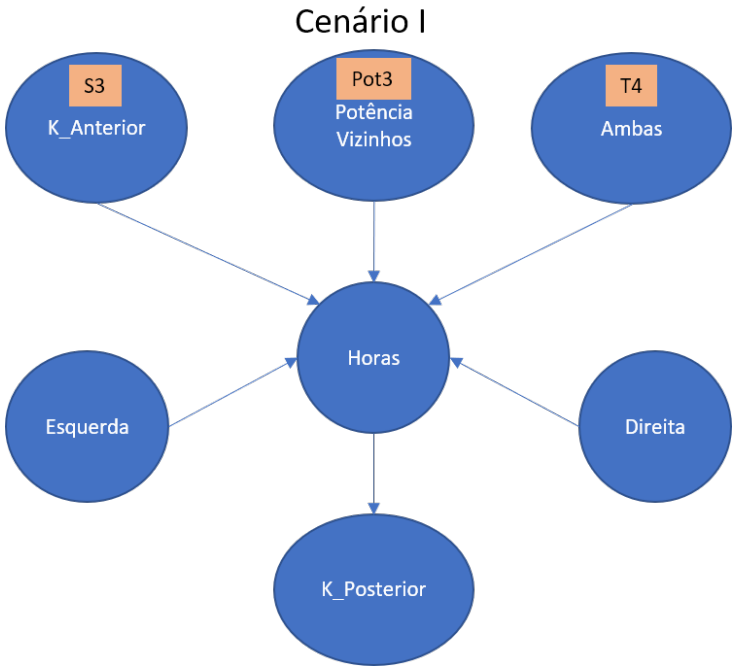

Figura 44 – Diagrama da Rede Bayesiana para os caso I

A imagem 45 apresenta as informações de vizinhança quanto a UG 37 e 38 estão sendo analisadas.

Figura 45 – UGs Vizinhas em Relação à UG em Análise

Os resultados com as CPDs para os cenários analisados são apresentados na Tabela 18.

Analisando o cenário E, a probabilidade da UG 37, que foi parada para decantação no nível *S3*, retomar a operação no nível e *S2* é de aproximadamente 51,5%. No entanto, é pouco provável que a UG retome a operação no nível *S1*, com o valor de 9,4%. Já no cenário F, onde apenas a UG vizinha direita está em operação, a probabilidade de retomada no nível *S2* é um pouco maior, com o valor de 59,3%.

Tabela 18 – Resultados da RB para UG 37: Probabilidades obtidas para os cenários E, F, G, H e I, utilizando *S3* como nível de obstrução inicial

	E	F	G	H	I
S1	9.4	10.3	28.0	19.1	34.1
S2	51.5	59.3	49.1	51.4	46.4
S3	38.1	28.9	20.9	27.2	17.3
S4	1.0	1.4	1.9	2.2	2.2

No cenário G, onde ambas as vizinhas estão em operação, os valores obtidos revelam um comportamento interessante. Embora a retomada no nível *S2* apareça com um valor de probabilidade um pouco mais baixo, 49,1%, a probabilidade de retomada em um nível mais limpo *S1* aumentou consideravelmente para 28,0%. Ainda é possível notar que a probabilidade da UG se manter no nível *S3* também é menor do que nos cenários anteriores, com o valor de 20,9%, comparados com o valor para o cenário E (38.1%) e F (28,9%).

A definição do valor de potência de operação das UGs vizinhas é efetuada no cenário H, sendo que o valor definido para este parâmetro foi *Pot1*, que compreende um valor de até 20 MW. Para que as UGs operem nessa faixa de potência, o que não é comum de acontecer na usina, é provável que o nível de obstrução esteja elevado.

Dito isso, e analisando os resultados obtidos, é possível observar que a retomada da UG no nível *S2* apresenta o valor de 51,4%, muito próximo ao cenário E, porém com uma probabilidade maior de retomada no nível *S1*, com o valor de 19,1%, em relação ao mesmo cenário E, com o valor de 9,4%. Esta observação demonstra que mesmo operando com baixa potência, é provável que o funcionamento das vizinhas movimente o sedimento da UG que está parada, efetuando assim a limpeza das grades.

Finalmente, no cenário I, temos ambas as vizinhas em operação durante todo o tempo em que a UG 37 esteve parada, agora com a média de potência de operação entre 40 e 55 MW, um valor expressivo se comparado aos 20 MW do cenário anterior.

Os resultados demonstram a maior probabilidade de retomada da UG no nível *S1*, com o valor de 34,1% se comparado com os demais cenários. Além disso, a probabilidade de retomada no nível *S2* também é bem expressiva, com o valor de 46,4%, o que demonstra que é muito provável que a UG esteja mais limpa quando retomar à operação. Novamente, os resultados mostram que o processo de decantação quando a UG está parada é significativamente influenciado pelas UGs vizinhas.

Na Tabela 19, são apresentados os resultados com as CPDs para os cenários E a I para a UG 38.

Os resultados para a UG 38 são similares aos da UG 37, porém com alguns pontos de destaque. No cenário H é perceptível um aumento de 9,3% na probabilidade de que a

Tabela 19 – Resultados da RB para UG 38: Probabilidades obtidas para os cenários E, F, G, H e I, utilizando *S3* como nível de obstrução inicial

K Posterior	E	F	G	H	I
S1	8.1	13.2	31.8	18.9	42.1
S2	52.9	56.1	47.1	60.7	46.3
S3	38.4	28.6	17.4	17.1	10.1
S4	0.6	2.1	3.7	3.3	1.5

UG 38 retome a operação no nível *S2*, com o valor de 60,7% se comparado ao valor de 51,4% da UG 37.

A diferença está intimamente ligada à redução na probabilidade de que a UG 38 retome no nível *S3* na qual foi parada para decantação. A UG 37 apresenta probabilidade de retomar em *S3* com o valor de 27,2%, enquanto a UG 38 apresenta o valor de 17,1%.

Isso pode indicar uma decantação mais rápida devido ao tipo de material acumulado, bem como pode estar relacionado ao momento em que a UG vizinha esquerda ou direita deu início à operação, fazendo com que o material acumulado seja movimentado para longe das grades de proteção, na direção de sua vizinha esquerda ou direita.

O cenário abaixo é utilizada com a finalidade de comparar as probabilidades de retomada da operação das UGs em um nível específico de obstrução.

- J = [K_Anterior: *S3*, Ambas: *T4*, Potência Vizinhos: *Pot3*, K_Posterior: *S1*]

A validação é executada utilizando-se as UGs 36, 37, 38 e 39. A figura 46 apresenta o caso J utilizando o diagrama da rede Bayesiana.

O cenário J é parametrizado com a informação de obstrução inicial *S3* e retomada de operação no nível *S1*. O valor do parâmetro 'Ambas' foi definido como *T4*, e por fim o parâmetro 'Potência Vizinhos' igual a *Pot3*.

Na Tabela 20, são apresentados os resultados com as CPDs para o cenário J para as UGs 36, 37, 38 e 39.

Tabela 20 – Resultados da RB para as UGs 36 a 39: Probabilidades obtidas para o cenário J, utilizando *S3* e *S1* como nível de obstrução inicial e final, respectivamente

Horas	UG36	UG37	UG38	UG39
H1	7.5	12.6	11.9	10.3
H2	12.9	16.5	18.3	13.4
H3	29.7	28.7	27.9	29.9
H4	49.9	42.3	41.9	46.3

Os resultados demonstram que mesmo utilizando os mesmos parâmetros na RB, o comportamento de limpeza das UGs apresentam diferenças suaves. Enquanto a probabili-

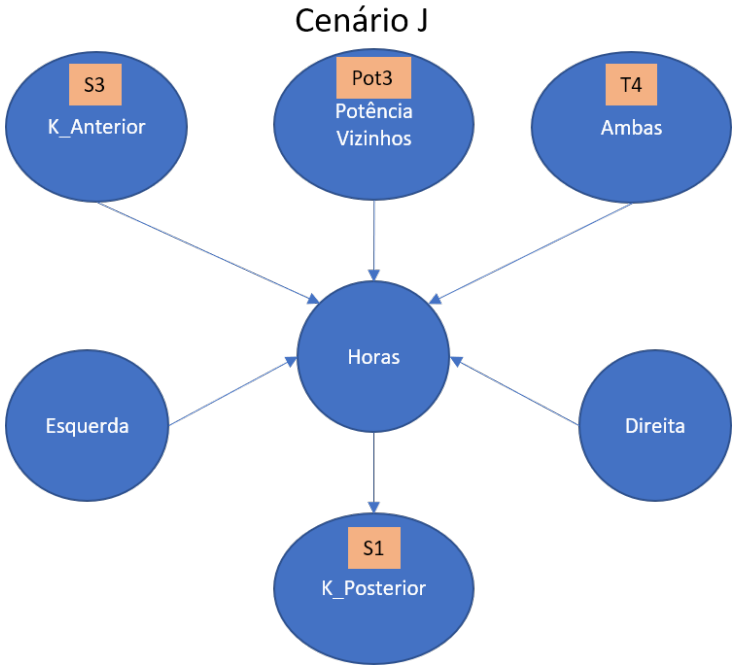

Figura 46 – Diagrama da Rede Bayesiana para o caso J

dade de que a UG 36 migre para o nível *S1* no tempo *H4* com o valor de 49,9%, a UG 38 apresenta o valor de 41,9% uma diferença de 8,0%. A partir do tempo *H3*, não houve uma grande variação nos valores de probabilidade das UGs, sendo que a UG 39 apresentou o maior valor, 29,9% e a UG 38 o valor de 27,9%, uma diferença de apenas 2,0%.

7.2 Discussões

A principal característica dos HMMs é a sua adequação para uso com dados sequenciais, onde a ordem das observações é essencial. Neste trabalho, os HMMs capturaram dependências temporais e transições entre diferentes estados, o que evidenciou a relação entre o nível de obstrução e o tempo decorrido quando as UGs são paradas para decantação. A flexibilidade dos HMMs permitiu seu uso com dados de entrada de séries temporais enquanto os níveis de obstrução são mapeados como estados no modelo.

As duas principais vantagens dos HMMs estão relacionadas à modelagem probabilística e à incorporação de estados ocultos. A primeira característica capta as incertezas, o que se enquadra nos objetivos deste trabalho: mapear a proporção de sedimentos acumulados e o tempo de inatividade necessário para sedimentação deste material. A segunda mapeia os processos subjacentes não observados de níveis de obstrução como estados ocultos, permitindo a estimativa do assentamento de sedimentos de acordo com o tempo decorrido.

Por outro lado, as limitações dos HMMs neste modelo são: uma vez que as probabilidades de transição são influenciadas pela UG vizinha, a Propriedade de Markov é

diretamente afetada e pode não se manter. A Propriedade de Markov também pode ter dificuldade para capturar dependências de longo alcance de maneira eficaz.

Outra limitação está relacionada ao espaço de estados fixados: o número de estados ocultos foi determinado antecipadamente e pode não representar o melhor cenário possível, além da dificuldade em determinar os intervalos limites das faixas de representação de cada nível de obstrução. A escolha do número apropriado de estados e da faixa compreendida entre cada nível foi um desafio, pois poderia afetar o desempenho do modelo. Foi necessário um esforço significativo para garantir que o espaço de estados selecionado refletisse a melhor opção.

Os resultados mostraram que a expressividade da técnica HMM por si só é limitada, uma vez que os modelos podem não capturar eficazmente relações complexas entre variáveis. A complexidade do treinamento representa um gargalo de tempo, pois um novo ciclo de treinamento deve ser realizado a cada nova variação de parâmetro e mapeamento de estado.

As vantagens das Redes Bayesianas baseiam-se no fato de que as RBs fornecem uma forma natural e intuitiva de modelar incertezas e dependências em dados obtidos da operação real da UHE. Como a RB é um framework probabilístico, foi possível inferir informações mesmo quando algumas variáveis pareciam não relacionadas.

A inferência causal das RBs permitiu compreender como o uso da UG vizinha afetou outras variáveis do modelo. Esse recurso foi valioso para a tomada de decisão sobre o uso das UGs enquanto as vizinhas estão paradas para decantação de sedimentos. Com a ajuda de especialistas no domínio do problema da UHE Jirau, foi possível validar crenças prévias e relações causais obtidas pela modelagem resultante.

As RBs permitiram análises exploratórias e inferências eficientes relacionadas ao comportamento dos sedimentos, o que ajudou a descobrir padrões ocultos que não eram imediatamente aparentes nos dados brutos, e a calcular probabilidades de diferentes cenários, fornecendo evidências de consulta. Como os dados operacionais reais da UHE estavam disponíveis, foi possível utilizar a RB para aprender os parâmetros de probabilidades condicionais a partir desses dados, o que torna o modelo consistente com a realidade da usina.

Redes Bayesianas apresentam algumas limitações, como a forte dependência da estrutura do gráfico. Foi um desafio corrigir a especificação do design, o que exigia conhecimentos especializados no domínio, uma vez que o modelo poderia não capturar de maneira eficaz as relações corretas sem a contribuição de especialistas.

O tempo de treinamento necessário foi um gargalo para perceber variações de parâmetros durante a modelagem, porque o elevado número de UGs representa um problema de complexidade computacional. Finalmente, as relações de correlação e causalidade re-

presentaram um desafio porque assumir a causalidade baseada na correlação pode por vezes ser perigoso.

7.3 Validação do Modelo

Para validar a qualidade dos modelos obtidos, foram utilizadas algumas métricas que visam identificar a eficiência na representação do modelo.

O escore de correlação visa pontuar o quão bem a estrutura do modelo representa as correlações nos dados. Os modelos obtidos apresentaram um escore de aproximadamente 92.3%, o que é considerado um resultado positivo e satisfatório.

A pontuação de probabilidade de registro *(Log Likelihood Score)* mede o log de verossimilhança, o que determina o quão bem o modelo especificado descreve os dados. A interpretação direta desta pontuação é difícil, mas pode ser usada para comparar o ajuste de dois ou mais modelos. A pontuação é calculada com base nos parâmetros estimados, e uma pontuação mais alta significa um melhor ajuste.

Ao observar a pontuação obtida pelo modelo resultante (-1027421.57), pode-se confirmar que é um valor suficiente para que o modelo represente de modo fiel as informações resultantes apresentadas.

Como uma validação adicional para o modelo, o conjunto de dados foi seccionado em partes distintas na proporção 80/20, onde 80% do conjunto de dados foi utilizado durante o treinamento do modelo e 20% foi selecionado como conjunto de testes. Tal separação visa averiguar se o modelo é capaz de inferir informações que não estavam no conjunto de dados utilizados durante o treinamento do modelo.

Em aproximadamente 97.5% dos casos o modelo foi capaz de inferir o resultado adequadamente, sendo que as validações do resultado apresentado foram confirmadas através do conjunto de testes.

7.4 Ganhos Operacionais

Nesta seção serão abordados os casos analisados após inferência realizada pelos modelos apresentados. Estes casos visam identificar possíveis melhorias operacionais na UHE de Jirau, que possam trazer ganhos operacionais, e consequentemente, ganhos financeiros devido a implementação de regras operacionais extraídas através da análise dos resultados.

Como as faixas de tempo foram discretizadas para utilização no modelo, há uma variação de 4 horas entre cada faixa de tempo em que a UG fica parada, sendo a faixa *H1*

de 1 a 4 horas, a faixa *H2* de 4 a 8 horas, a faixa *H3* de 8 a 12 horas e a faixa *H4* acima de 12 horas.

Observando os resultados obtidos através dos modelos, sabe-se que uma UG que ficou parada pelo tempo *H3* ficou de 8 a 12 horas parada, porém, ao analisar os dados, é possível saber exatamente qual o período em que a UG ficou parada, qual o nível de sujeira em que a UG foi parada e o nível de obstrução em que a UG foi restabelecida.

Há nos dados vários cenários onde as UGs foram paradas no nível *S4* e retornaram no nível *S1*, com variações de até 3 horas no tempo em que a UG ficou parada. O entendimento dos fatores relacionados a esta diferença de tempo para alcançar o mesmo nível de obstrução final representa a possibilidade de elaborar regras que visam o ganho operacional da UHE.

Analisando o caso de diferentes UGs em ocasiões distintas temos que, quando parada no nível de obstrução mais acentuado, ou seja, *S4* e retomando em *S1*, houve uma diferença de até 2 horas no tempo de parada nas seguintes condições:

- Cenário 1: Não haviam UGs vizinhas em operação durante o tempo em que a UG em análise esteve parada.
- Cenário 2: Apenas 1 das vizinhas operava durante o tempo em que a UG em análise esteve parada.
- Cenário 3: Ambas as vizinhas operavam durante o tempo em que a UG em análise esteve parada.

Foi possível verificar que no cenário 1, o tempo de decantação necessário é aproximadamente 2 horas médias maior. Esta situação se comprovou para todas as UGs, exceto as UGs 1, 28 e 50, pois estão próximas às margens. A média de horas apresentada para o cenário 1 é de aproximadamente 14,41 horas.

O cenário 2 considerou situações em que uma das vizinhas esteve operando, seja a esquerda ou direita. Nos casos analisados, a média obtida foi de 13,14 horas, ou seja, 1,3 horas a menos em comparação com o cenário 1.

Para contabilizar que a UG vizinha esteve operando, foram selecionados apenas casos onde o período de operação da vizinha ultrapassou 3 horas. Foi identificado que a operação de uma UG vizinha por período inferior a 3 horas não alterou o tempo gasto para alcançar o nível de obstrução desejado.

Por fim, o cenário 3 foi o mais promissor, no que se refere ao tempo necessário para alcançar o nível de obstrução desejado. Quando as duas UGs vizinhas operam enquanto a UG em análise está parada, o tempo médio foi de 12,23 horas. Esse resultado é quase 2 horas a menos do que o alcançado no cenário 1, e 1 hora a menos do que o cenário 2.

Nesse cenário foi considerado apenas os casos em que a soma de tempo de operação das 2 UGs vizinhas foi maior do que 3 horas, mesmo que as 2 vizinhas não tenham operado 100% do tempo em que a UG analisada esteve parada.

A utilização das técnicas apresentadas neste trabalho apresentou informações que permitem ao operador tomar a decisão baseada em modelos que foram elaborados com base nos próprios dados de operação da UHE Jirau.

Como no período de cheia do rio o recurso hídrico deve ser utilizado para a geração ou então ser vertido pelos vertedouros, uma redução de 1 ou 2 horas no tempo em que as UGs permanecem paradas para decantação já representa ganhos financeiros.

8 Conclusão

Esta tese propõe técnicas para estimar o tempo ideal de parada das UGs da UHE Jirau utilizando Modelos Ocultos de Markov (Markov Hidden Models) e Redes Bayesianas como métodos de inferência. Dados operacionais de campo são utilizados para obtenção dos modelos apresentados, no entanto, para viabilizar a disponibilidade desses dados para a implementação dos modelos, foi necessário antes lidar com problemas relacionados a Big Data e Big Data Analytics.

Dada a grande quantidade de fontes e formatos de informação, o volume de dados gerados pelos mais diversos aparatos de hardware, software e lançamentos efetuados por funcionários, coletar, reunir, processar e analisar toda essa massa de dados representou um grande desafio do trabalho. Foram necessários estudos e implementações para viabilizar meios para que a informação pudesse ser utilizada nos modelos de previsão apresentados neste trabalho.

Ainda foi necessário desenvolver métodos capazes de identificar de maneira automática e em tempo real o estado operativo das 50 UGs, sendo que o esforço relacionado à esta tarefa demandou tempo e dedicação, além de validações constantes entre pesquisador e funcionários da UHE de Jirau.

O resultado obtido possibilita a apresentação do estado operacional aos funcionários da planta sem que seja necessário o acesso e consulta a diversos sistemas, conferindo maior precisão e velocidade ao processo de identificação, além de viabilizar que as informações de estado pudessem ser inseridas nos modelos de previsão.

Os resultados obtidos pelos modelos elaborados demonstram consistência com a operação diária da planta, permitindo a utilização do modelo na tomada de decisão do operador, auxiliando na operação do elevado número de UGs existentes na UHE de Jirau.

Uma vantagem essencial da metodologia apresentada é que ela permite meios sistematizados e baseados em dados para modelar a inferência de informações, possibilitando regras de operação mais consistentes na UHE.

Um ponto forte do trabalho apresentado é a união do HMM com RBs, que permitiu aproveitar as principais características de cada uma das técnicas. A robustez do modelo permite a extração de informações de probabilidade. Isso trouxe à tona detalhes relacionados ao comportamento da obstrução nas grades de proteção, incluindo o impacto da UG vizinha no comportamento de sedimentação.

Como a UHE de Jirau é uma usina a fio d'água e não permite armazenamento de recursos hídricos, a proposta apresentada neste trabalho oferece métodos que permitem

utilizar um modelo robusto no planejamento da operação da usina sob diversos cenários possíveis, extraindo a probabilidade resultante sob cada perspectiva analisada.

Esta proposta inovadora visa trazer maior clareza e padronização à extração de regras de operação nas UHEs cujo acúmulo de sedimentos na grade pode influenciar negativamente as operações diárias, especialmente aquelas da Bacia Amazônica.

A metodologia aplicada neste trabalho pode ser utilizada em outras UHEs, tanto para extração de regras de operação quanto para modelagem de HMM e RB. Além disso, o modelo resultante pode ajudar a identificar fatores que alteram a eficiência operacional das UGs, fornecendo ferramentas e métodos para operar a planta de forma mais eficiente.

Por se tratar de um trabalho pioneiro, que aborda o problema de obstrução das grades de proteção com consequente impacto na operação de usinas hidrelétricas, estudos adicionais ainda precisam ser realizados para referência contextual e comparação. Esse problema relacionado ao acúmulo de sedimentos é específico das UHEs localizadas na Bacia Amazônica. No caso da usina de Jirau, o desafio das altas taxas de transporte de sedimentos surge apenas durante determinados períodos do ano. Por este motivo, são utilizados apenas dados referentes às épocas de cheias.

Embora os modelos gerados utilizem dados operacionais da UHE, o treinamento foi offline. Como trabalho futuro, propõe-se integrar dados da planta para treinamento de modelos online, apresentando informações em tempo real para auxiliar os operadores na tomada de decisões e minimizar o tempo de inatividade das UGs.

Outras modificações que podem ser incluídas em trabalhos futuros são a quantidade de estados ocultos mapeados para uso no HMM, a definição de novas faixas de tempo e de nível de obstrução utilizados nos modelos e a análise e inclusão de variáveis adicionais para composição das RBs, tais como a sequência de parada das UGs para decantação dos sedimentos.

Referências

AGARWAL, S.; PRIYUSHA, M. A transformation from relational databases to big data. *International Journal Of Engineering And Computer Science*, 2015. 30, 31, 42

Agência Nacional de Águas e Saneamento Básico (ANA). *ANA*. 2023. Disponível em: <http://arquivos.ana.gov.br/resolucoes/2009/269-2009.pdf?144140>. 42

AHMAD, S. K.; HOSSAIN, F. A generic data-driven technique for forecasting of reservoir inflow: Application for hydropower maximization. *Environmental Modelling & Software*, Elsevier, v. 119, p. 147–165, 2019. 53

AMIN, M. Challenges in reliability, security, efficiency, and resilience of energy infrastructure: Toward smart self-healing electric power grid. In: IEEE. *2008 IEEE Power and energy society general meeting-conversion and delivery of electrical energy in the 21st century*. [S.l.], 2008. p. 1–5. 17, 25

Apache Foundation. *Apache Hadoop*. 2022. Disponível em: <https://hadoop.apache.org/>. 36

Apache Foundation. *Open Source NoSQL Database*. 2022. Disponível em: <https://cassandra.apache.org/_/index.html>. 32

ASSOCIATION, I. H. et al. *Hydropower status report: Sector trends and insights 2018*. [S.l.]: London: Author, 2018. 16

Aveva OSI Soft. *PI System*. 2022. Disponível em: <https://www.osisoft.pt/pi-system/>. 8, 49, 50

BAKER, J. The dragon system–an overview. *IEEE Transactions on Acoustics, speech, and signal Processing*, IEEE, v. 23, n. 1, p. 24–29, 1975. 85

BAKSH, A.-A. et al. Marine transportation risk assessment using bayesian network: Application to arctic waters. *Ocean Engineering*, Elsevier, v. 159, p. 422–436, 2018. 56

BANIHABIB, M. E.; BANDARI, R.; PERALTA, R. C. Auto-regressive neural-network models for long lead-time forecasting of daily flow. *Water Resources Management*, Springer, v. 33, p. 159–172, 2019. 53

BANSAL, S. K. Towards a semantic extract-transform-load (etl) framework for big data integration. In: IEEE. *2014 IEEE International Congress on Big Data*. [S.l.], 2014. p. 522–529. 39, 40, 41

BARTON, D. N. et al. Bayesian networks in environmental and resource management. *Integrated environmental assessment and management*, Wiley Online Library, v. 8, n. 3, p. 418–429, 2012. 56

BAUM, L. E. et al. An inequality and associated maximization technique in statistical estimation for probabilistic functions of markov processes. *Inequalities*, v. 3, n. 1, p. 1–8, 1972. 81, 83, 85

BAUM, L. E.; PETRIE, T. Statistical inference for probabilistic functions of finite state markov chains. *The annals of mathematical statistics*, JSTOR, v. 37, n. 6, p. 1554–1563, 1966. 81, 85

BHATTARAI, B. P. et al. Big data analytics in smart grids: state-of-the-art, challenges, opportunities, and future directions. *IET Smart Grid*, Wiley Online Library, v. 2, n. 2, p. 141–154, 2019. 16, 17, 27, 28

BHAUMIK, D. et al. Hidden markov models for wind farm power output. *IEEE Transactions on Sustainable Energy*, IEEE, v. 10, n. 2, p. 533–539, 2018. 55

BORDIN, C. et al. Machine learning for hydropower scheduling: State of the art and future research directions. *Procedia Computer Science*, Elsevier, v. 176, p. 1659–1668, 2020. 47, 52

BORSUK, M. et al. Stakeholder values and scientific modeling in the neuse river watershed. *Group Decision and Negotiation*, Springer, v. 10, n. 4, p. 355–373, 2001. 56

CAO, L. et al. Combined mining: discovering informative knowledge in complex data. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, IEEE, v. 41, n. 3, p. 699–712, 2010. 16

CARITA, A. J. Q. et al. Bayesian networks applied to failure diagnosis in power transformer. *IEEE Latin America Transactions*, IEEE, v. 11, n. 4, p. 1075–1082, 2013. 56, 87, 88

CARPIO, J. M. Hidrologia e sedimentos. *Águas Turvas: alertas sobre as conseqüências de barrar o maior afluente do Amazonas/Glenn Switkes*, 2008. 44

CASTRO, N. P. d. Avaliação de indicadores de alteração hidrológica na bacia hidrográfica do rio madeira: grandes obras hidráulicas, sedimentos e os possíveis impactos na dinâmica fluvial. Universidade Estadual Paulista (Unesp), 2019. 44

CERIBASI, G.; CALISKAN, M. Short-and long-term prediction of energy to be produced in hydroelectric energy plants of sakarya basin in turkey. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, Taylor & Francis, p. 1–16, 2019. 53

CHAUDHARI, A. A.; MULAY, P. Scsi: real-time data analysis with cassandra and spark. *Big Data Processing Using Spark in Cloud*, Springer, p. 237–264, 2019. 39, 40, 41

CHENG, C.-t. et al. Daily reservoir runoff forecasting method using artificial neural network based on quantum-behaved particle swarm optimization. *Water*, MDPI, v. 7, n. 8, p. 4232–4246, 2015. 52

COLEMAN, A.; ZALEWSKI, J. Intelligent fault detection and diagnostics in solar plants. In: IEEE. *Proceedings of the 6th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems*. [S.l.], 2011. v. 2, p. 948–953. 56

DALCIN, L. D. et al. Parallel distributed computing using python. *Advances in Water Resources*, Elsevier, v. 34, n. 9, p. 1124–1139, 2011. 35

DataBricks. *Sobre o Apache Spark*. 2022. Disponível em: <https://databricks.com/spark/about>. 8, 38, 39

DEAN, J.; GHEMAWAT, S. Mapreduce: simplified data processing on large clusters. *Communications of the ACM*, ACM New York, NY, USA, v. 51, n. 1, p. 107–113, 2008. 33, 35, 49

DEHGHANPOUR, K. et al. Agent-based modeling in electrical energy markets using dynamic bayesian networks. *IEEE Transactions on Power Systems*, IEEE, v. 31, n. 6, p. 4744–4754, 2016. 57

EDWARDS, A. [the reverend thomas bayes, frs: A biography to celebrate the tercentenary of his birth]: Comment. *Statistical Science*, JSTOR, v. 19, n. 1, p. 34–37, 2004. 89

ESCRIG, J. et al. Monitoring the cleaning of food fouling in pipes using ultrasonic measurements and machine learning. *Food Control*, Elsevier, v. 116, p. 107309, 2020. 58

FAHIMAN, F. et al. Data-driven dynamic probabilistic reserve sizing based on dynamic bayesian belief networks. *IEEE Transactions on Power Systems*, IEEE, v. 34, n. 3, p. 2281–2291, 2018. 56

FRAPPART, F. et al. Surface freshwater storage and dynamics in the amazon basin during the 2005 exceptional drought. *Environmental Research Letters*, IOP Publishing, v. 7, n. 4, p. 044010, 2012. 44

FREIRE, J. C. A. et al. Transmission line fault classification using hidden markov models. *IEEE Access*, IEEE, v. 7, p. 113499–113510, 2019. 54

GHORBANIAN, M.; DOLATABADI, S. H.; SIANO, P. Big data issues in smart grids: A survey. *IEEE Systems Journal*, IEEE, v. 13, n. 4, p. 4158–4168, 2019. 16, 25, 27, 28, 39, 40, 41, 42

GONZÁLEZ, A. M.; ROQUE, A. S.; GARCÍA-GONZÁLEZ, J. Modeling and forecasting electricity prices with input/output hidden markov models. *IEEE Transactions on Power Systems*, IEEE, v. 20, n. 1, p. 13–24, 2005. 55

HAO, C.-F.; QIU, J.; LI, F.-F. Methodology for analyzing and predicting the runoff and sediment into a reservoir. *Water*, MDPI, v. 9, n. 6, p. 440, 2017. 52, 57, 58, 59

HE, X. et al. A big data architecture design for smart grids based on random matrix theory. *IEEE transactions on smart Grid*, IEEE, v. 8, n. 2, p. 674–686, 2015. 28, 29

HEMMING, J. *Change in the Amazon Basin: The frontier after a decade of colonisation*. [S.l.]: Manchester University Press, 1985. v. 2. 44

HONG, Y.-S. T.; WHITE, P. A. Hydrological modeling using a dynamic neuro-fuzzy system with on-line and local learning algorithm. *Advances in Water Resources*, Elsevier, v. 32, n. 1, p. 110–119, 2009. 52

HOPCROFT, J. E.; MOTWANI, R.; ULLMAN, J. D. Automata theory, languages, and computation. *International Edition*, v. 24, n. 2, p. 171–183, 2006. 63, 64

HU, H. et al. Toward scalable systems for big data analytics: A technology tutorial. *IEEE access*, IEEE, v. 2, p. 652–687, 2014. 37, 38, 41

HUANG, S. et al. Monthly streamflow prediction using modified emd-based support vector machine. *Journal of Hydrology*, Elsevier, v. 511, p. 764–775, 2014. 52

IBE, O. *Fundamentals of applied probability and random processes*. [S.l.]: Academic Press, 2014. 78

Jacob Schreiber. *Pomegranate - A Python package for probabilistic models*. 2023. Disponível em: <https://pomegranate.readthedocs.io/en/latest/index.html>. 107

JATOTH, C.; GANGADHARAN, G.; FIORE, U. Evaluating the efficiency of cloud services using modified data envelopment analysis and modified super-efficiency data envelopment analysis. *Soft Computing*, Springer, v. 21, p. 7221–7234, 2017. 35

JIANG, H. et al. Energy big data: A survey. *IEEE Access*, IEEE, v. 4, p. 3844–3861, 2016. 17, 26

JIANG, J. et al. Dynamic fault prediction of power transformers based on hidden markov model of dissolved gases analysis. *IEEE Transactions on Power Delivery*, IEEE, v. 34, n. 4, p. 1393–1400, 2019. 54

Jirau Energia. *Conheça a UHE - Jirau Energia*. 2022. Disponível em: <https://www.jirauenergia.com.br/conheca-a-uhe/>. 8, 42, 43

JR, J. B. et al. Hydropower operation optimization using machine learning: A systematic review. *AI*, MDPI, v. 3, n. 1, p. 78–99, 2022. 46

KJÆRULFF, U. B.; MADSEN, A. L. Probabilistic networks for practitioners-a guide to construction and analysis of bayesian networks and influence diagrams. *Department of Computer Science, Aalborg University, HUGIN Expert A/S*, 2006. 90

KJAERULFF, U. B.; MADSEN, A. L. Bayesian networks and influence diagrams. *Springer Science+ Business Media*, Springer, v. 200, p. 114, 2008. 79, 80, 87, 88, 109

KOLLER, D.; FRIEDMAN, N. *Probabilistic graphical models: principles and techniques*. [S.l.]: MIT press, 2009. 109

KONG, W. et al. A hierarchical hidden markov model framework for home appliance modeling. *IEEE Transactions on Smart Grid*, IEEE, v. 9, n. 4, p. 3079–3090, 2016. 55

KORB, K. B.; NICHOLSON, A. E. *Bayesian artificial intelligence*. [S.l.]: CRC press, 2010. 80, 87, 88

KOSKI, T. J.; NOBLE, J. A review of bayesian networks and structure learning. *Mathematica Applicanda*, Polskie Towarzystwo Matematyczne, v. 40, n. 1, 2012. 90, 91

KUMAR, D.; BHOWMIK, P. S. Hidden markov model based islanding prediction in smart grids. *IEEE Systems Journal*, IEEE, v. 13, n. 4, p. 4181–4189, 2019. 54

KUMAR, K.; SAINI, R. A review on operation and maintenance of hydropower plants. *Sustainable Energy Technologies and Assessments*, Elsevier, v. 49, p. 101704, 2022. 25, 26

KUMAR, S.; MOHBEY, K. K. A review on big data based parallel and distributed approaches of pattern mining. *Journal of King Saud University-Computer and Information Sciences*, Elsevier, 2019. 28, 29, 30, 31, 34

KUMAR, S.; MOHBEY, K. K. A review on big data based parallel and distributed approaches of pattern mining. *Journal of King Saud University-Computer and Information Sciences*, Elsevier, v. 34, n. 5, p. 1639–1662, 2022. 37

KUNDU, A.; HE, Y.; BAHL, P. Recognition of handwritten word: first and second order hidden markov model based approach. *Pattern recognition*, Elsevier, v. 22, n. 3, p. 283–297, 1989. 82, 85

LE, A. V. et al. Reinforcement learning-based optimal complete water-blasting for autonomous ship hull corrosion cleaning system. *Ocean Engineering*, Elsevier, v. 220, p. 108477, 2021. 58, 59

LEE, C.-J.; LEE, K. J. Application of bayesian network to the probabilistic risk assessment of nuclear waste disposal. *Reliability Engineering & System Safety*, Elsevier, v. 91, n. 5, p. 515–532, 2006. 56

LEE, K.-F. et al. Speech recognition using hidden markov models: a cmu perspective. *Speech Communication*, Elsevier, v. 9, n. 5-6, p. 497–508, 1990. 82, 86

LEONI, L. et al. Developing a risk-based maintenance model for a natural gas regulating and metering station using bayesian network. *Journal of Loss Prevention in the Process industries*, Elsevier, v. 57, p. 17–24, 2019. 56

LERNER, B.; MALKA*, R. Investigation of the k2 algorithm in learning bayesian network classifiers. *Applied Artificial Intelligence*, Taylor & Francis, v. 25, n. 1, p. 74–96, 2011. 90

LI, G. et al. Short-term power generation energy forecasting model for small hydropower stations using ga-svm. *Mathematical Problems in Engineering*, Hindawi, v. 2014, 2014. 52

LI, W. K.; WANG, W. L.; LI, L. Optimization of water resources utilization by multi-objective moth-flame algorithm. *Water Resources Management*, Springer, v. 32, p. 3303–3316, 2018. 16

LV, Z. et al. Next-generation big data analytics: State of the art, challenges, and future research topics. *IEEE Transactions on Industrial Informatics*, IEEE, v. 13, n. 4, p. 1891–1899, 2017. 17, 26, 29, 34, 36

MAHMUD, M. S. et al. A survey of data partitioning and sampling methods to support big data analysis. *Big Data Mining and Analytics*, TUP, v. 3, n. 2, p. 85–101, 2020. 29, 30, 32, 37

MOHAMMED, M.; KHAN, M. B.; BASHIER, E. B. M. *Machine learning: algorithms and applications*. [S.l.]: Crc Press, 2016. 17

MURDOCK, H. E. et al. Renewables 2020-global status report. 2020. 41

MURTHY, C. et al. Reliability analysis of phasor measurement unit using hidden markov model. *IEEE Systems Journal*, IEEE, v. 8, n. 4, p. 1293–1301, 2014. 55

NOTARISTEFANO, A.; CHICCO, G.; PIGLIONE, F. Data size reduction with symbolic aggregate approximation for electrical load pattern grouping. *IET Generation, Transmission & Distribution*, Wiley Online Library, v. 7, n. 2, p. 108–117, 2013. 28

Operador Nacional do Sistema Elétrico. *ONS - Manual de Procedimentos da Operação*. 2022. Disponível em: <http://www.ons.org.br/paginas/sobre-o-ons/procedimentos-de-rede/mpo/>. 42, 46, 51, 66, 67

Operador Nacional do Sistema Elétrico. *ONS - Procedimentos de Rede*. 2022. Disponível em: <http://www.ons.org.br/paginas/sobre-o-ons/procedimentos-de-rede/vigentes/>. 46, 47, 51, 65, 66, 67

ORDEAN, M. et al. Scada systems-support for the maintenance management of hydro power plants. In: IEEE. *2006 IEEE International Conference on Automation, Quality and Testing, Robotics*. [S.l.], 2006. v. 1, p. 238–242. 25

OUSSOUS, A. et al. Big data technologies: A survey. *Journal of King Saud University-Computer and Information Sciences*, Elsevier, v. 30, n. 4, p. 431–448, 2018. 29

OVIEDO, D. et al. Multiple intelligences in a multiagent system applied to telecontrol. *Expert systems with applications*, Elsevier, v. 41, n. 15, p. 6688–6700, 2014. 57

PEARL, J. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. [S.l.]: Morgan kaufmann, 1988. 56

RABIN, M. O.; SCOTT, D. Finite automata and their decision problems. *IBM journal of research and development*, IBM, v. 3, n. 2, p. 114–125, 1959. 64

RABINER, L. R. A tutorial on hidden markov models and selected applications in speech recognition. *Proceedings of the IEEE*, Ieee, v. 77, n. 2, p. 257–286, 1989. 53, 85

RAMALINGAM, B. et al. Cascaded machine-learning technique for debris classification in floor-cleaning robot application. *Applied Sciences*, MDPI, v. 8, n. 12, p. 2649, 2018. 58, 59

RASHDAN, A. Y. A. et al. *Method and application of data integration at a nuclear power plant. Light Water Reactor Sustainability Program report*. [S.l.], 2019. 28, 42

RODRIGUEZ, A.; LAIO, A. Clustering by fast search and find of density peaks. *science*, American Association for the Advancement of Science, v. 344, n. 6191, p. 1492–1496, 2014. 28

SAGIROGLU, S. et al. Big data issues in smart grid systems. In: IEEE. *2016 IEEE International Conference on Renewable Energy Research and Applications (ICRERA)*. [S.l.], 2016. p. 1007–1012. 16, 26, 27, 32, 42

SAKAROVITCH, J. *Elements of automata theory*. [S.l.]: Cambridge University Press, 2009. 64

SANDHU, A. K. Big data with cloud computing: Discussions and challenges. *Big Data Mining and Analytics*, TUP, v. 5, n. 1, p. 32–40, 2021. 33, 34, 36, 39, 40, 49

SANTOS, R. E. et al. The decline of fisheries on the madeira river, brazil: The high cost of the hydroelectric dams in the amazon basin. *Fisheries management and ecology*, Wiley Online Library, v. 25, n. 5, p. 380–391, 2018. 44

SCANAGATTA, M.; SALMERÓN, A.; STELLA, F. A survey on bayesian network structure learning from data. *Progress in Artificial Intelligence*, Springer, v. 8, p. 425–439, 2019. 90

SELAK, L.; BUTALA, P.; SLUGA, A. Condition monitoring and fault diagnostics for hydropower plants. *Computers in Industry*, Elsevier, v. 65, n. 6, p. 924–936, 2014. 16

SHAQIRI, B. *Exploring techniques of improving security and privacy in big data*. Tese (Doutorado) — Ph. D. thesis, University of Information and Technology-Ohrid, 2017. 8, 26, 27, 42

SIMEONE, A. et al. Intelligent industrial cleaning: a multi-sensor approach utilising machine learning-based regression. *Sensors*, MDPI, v. 20, n. 13, p. 3642, 2020. 58

SMYTH, P. Hidden markov models for fault detection in dynamic systems. *Pattern recognition*, Elsevier, v. 27, n. 1, p. 149–164, 1994. 86

SONG, J. et al. Haery: a hadoop based query system on accumulative and high-dimensional data model for big data. *IEEE transactions on knowledge and data engineering*, IEEE, v. 32, n. 7, p. 1362–1377, 2019. 31, 32, 36, 37, 41

SORÍ, R. et al. The atmospheric branch of the hydrological cycle over the negro and madeira river basins in the amazon region. *Water*, MDPI, v. 10, n. 6, p. 738, 2018. 44

SPIEGELHALTER, D. J.; FRANKLIN, R. C.; BULL, K. Assessment, criticism and improvement of imprecise subjective probabilities for a medical expert system. *arXiv preprint arXiv:1304.1529*, 2013. 56

STAMP, M. A revealing introduction to hidden markov models. *Department of Computer Science San Jose State University*, p. 26–56, 2004. 53, 82, 86

STASSOPOULOU, A.; PETROU, M.; KITTLER, J. Application of a bayesian network in a gis based decision making system. *International Journal of Geographical Information Science*, Taylor & Francis, v. 12, n. 1, p. 23–46, 1998. 56

SUN, S.; YANG, Q.; YAN, W. A novel markov-based temporal-soc analysis for characterizing pev charging demand. *IEEE Transactions on Industrial Informatics*, IEEE, v. 14, n. 1, p. 156–166, 2017. 54

SUZUKI, J. The bayesian chow-liu algorithm. In: *The sixth european workshop on probabilistic graphical models*. [S.l.: s.n.], 2012. p. 315–322. 90

TAHMASSEBPOUR, M. A new method for time-series big data effective storage. *Ieee Access*, IEEE, v. 5, p. 10694–10699, 2017. 27, 32, 36, 40

TUCCI, C. E. Análises dos estudos ambientais dos empreendimentos do rio madeira. *Instituto Brasileiro de Meio Ambiente–IBAMA*, 2007. 45

TURGEON, K. et al. Empirical characterization factors to be used in lca and assessing the effects of hydropower on fish richness. *Ecological Indicators*, Elsevier, v. 121, p. 107047, 2021. 41

VASILIEV, Y. S.; ZEGZHDA, P.; ZEGZHDA, D. Providing security for automated process control systems at hydropower engineering facilities. *Thermal Engineering*, Springer, v. 63, p. 948–956, 2016. 25

VERAJAGADHESWA, P. et al. A novel autonomous staircase cleaning system with robust 3d-deep learning-based perception technique for area-coverage. *Expert Systems with Applications*, Elsevier, v. 194, p. 116528, 2022. 58, 59

WANG, Y. et al. Clustering of electricity consumption behavior dynamics toward big data applications. *IEEE transactions on smart grid*, IEEE, v. 7, n. 5, p. 2437–2447, 2016. 28

WHITTAKER, J. *Graphical models in applied multivariate statistics*. [S.l.]: Wiley Publishing, 2009. 78, 81

YANG, L.; WIDJAJA, B.; PRASAD, R. Application of hidden markov models for signature verification. *Pattern recognition*, Elsevier, v. 28, n. 2, p. 161–170, 1995. 86

YIN, J. et al. Table cleaning task by human support robot using deep learning technique. *Sensors*, MDPI, v. 20, n. 6, p. 1698, 2020. 58, 59

YONGLI, Z.; LIMIN, H.; JINLING, L. Bayesian networks-based approach for power systems fault diagnosis. *IEEE Transactions on Power Delivery*, IEEE, v. 21, n. 2, p. 634–639, 2006. 57

ZHAN, J. et al. Study of the key technologies of electric power big data and its application prospects in smart grid. In: IEEE. *2014 IEEE PES Asia-Pacific Power and Energy Engineering Conference (APPEEC)*. [S.l.], 2014. p. 1–4. 16

ZHAO, Y. et al. Bayesian network based imprecise probability estimation method for wind power ramp events. *Journal of Modern Power Systems and Clean Energy*, SGEPRI, v. 9, n. 6, p. 1510–1519, 2020. 57

ZHOU, A. et al. Prediction-based population re-initialization for evolutionary dynamic multi-objective optimization. In: *EMO*. [S.l.: s.n.], 2006. v. 4403, p. 832–846. 17

ZHOU, D. et al. Distributed data analytics platform for wide-area synchrophasor measurement systems. *IEEE Transactions on Smart Grid*, IEEE, v. 7, n. 5, p. 2397–2405, 2016. 28, 29, 33

ZHOU, K.; FU, C.; YANG, S. Big data driven smart energy management: From big data to big insights. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 56, p. 215–225, 2016. 17, 26, 42

ZHOU, K. et al. Energy conservation and emission reduction of china's electric power industry. *Renewable and Sustainable Energy Reviews*, Elsevier, v. 45, p. 10–19, 2015. 17

ZOHREVAND, Z. et al. Hidden markov based anomaly detection for water supply systems. In: IEEE. *2016 IEEE International Conference on Big Data (Big Data)*. [S.l.], 2016. p. 1551–1560. 56

ZWEIG, G.; RUSSELL, S. Speech recognition with dynamic bayesian networks. In: *Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*. [S.l.]: University of California, Berkeley, 1998. p. 173–180. 56

Apêndices

APÊNDICE A – Trabalhos Resultantes

Durante a realização do doutorado foram obtidos resultados oriundo das pesquisas efetuadas. Dentre os resultados estão a publicação em revistas, apresentação de trabalhos em eventos e participação em projeto de P&D. A seguir são apresentados os resultados obtidos:

- Publicação de capítulo em livro intitulado “Decision Making Applications in Modern Power Systems”, disponível no endereço: <https://www.sciencedirect.com/book/9780128164457/decision-making-applications-in-modern-power-systems>. O capítulo de minha autoria é o 19: "Modeling and processing of smart grids big data: study case of a university research building".
- Participação no projeto de P&D da ANEEL durante o período entre 08/2020 até 12/2022. Título do projeto: “Otimização da operação de usinas hidrelétricas através da minimização das perdas no processo de geração”, executado na usina hidrelétrica de Jirau.
- Apresentação de trabalho no evento "XV SEPOPE: Simpósio de Especialistas em Planejamento da Operação e Expansão de Sistemas de Energia Elétrica". Título do trabalho: "Ferramenta para detecção e monitoramento automáticos de estados operativos de Unidades Geradoras de Usinas Hidrelétricas".
- Apresentação de trabalho no evento "XXVI SNPTEE: Seminário Nacional de Produção e Transmissão de Energia Elétrica". Título do trabalho: "Aquisição de dados de usina hidroelétrica e integração com sistemas de Business Intelligence".
- Publicação de artigo na revista AI: “Hydropower Operation Optimization Using Machine Learning: A Systematic Review”, disponível no endereço: <https://www.mdpi.com/2673-2688/3/1/6>.
- Dentre as soluções entregues durante a execução do projeto de P&D, foi desenvolvido um framework de Big Data Analytics utilizando técnicas de Business Intelligence, que foi implantado e está em uso na usina hidrelétrica de Jirau.
- Outra solução entregue, foi a ferramenta de identificação automática e em tempo real dos estados operativos das 50 unidades geradoras da usina hidrelétrica de Jirau.
- Publicação de artigo na revista Energies, “Special Issue - Power System Analysis Control and Operation” intitulado “Forecast of Operational Downtime of the Generating Units for Sediment Cleaning in the Water Intakes”.