Bonsai Brain - AI Solution Spec

Project Start Date**:** 12/2/2019

Authors: [Solution Architect Name]

|  |  |  |
| --- | --- | --- |
| **Customer** | **MineCo Crusher Optimization** | |
| **Project Objective** | *What task/process are you looking to improve using deep reinforcement learning?* | * MineCo mines precious metals from ore.​ * The first and roughest stage of ore processing is crushing the ore, in this case with a gyratory crusher. ​ * Operators manually control supervisory settings for the gyratory crusher.  ​ * The particle size and hardness entering the crusher and the crusher itself, particularly the condition of the liner, vary with an unknown random distribution.  This makes it very difficult to control the crusher well.   ​​ * **The objective of this EAP project is to train a brain(s) to provide supervisory control settings for the underground gyratory crushers at the MineCo South site that maximize the throughput of fragmented ore.**​   **Figure 1**: Crusher Schematic |
| **Business Value** | *What is the business value of improving the control/optimization of this system?* | A 5% improvement in throughput generates a predicted ROI of $10M / year at the mine site in question. |
| **Optimization Goal** | *What Key Performance Indicators (KPI) define the control or optimization of this system?* | |  |  |  | | --- | --- | --- | | **Goal (KPI)** | **Units** | **Description** | | *Maximize* Throughput | tons/hour | The amount of material that passes through the crusher on to high pressure grinding rolls (HPGR) having met the 65mm fineness criteria for particle size. | |
| **Current Methods** | *How do you currently control or optimize the system?* | |  |  |  | | --- | --- | --- | |  | **Method** | **Level** | |  | Human Operator / Engineer | **Supervisory Control**: The fixed plant team currently gaps the crushers once every 24 hours. This leads to sub-optimal results and is time intensive. | |  | Expert System |  | |  | Control Theory (PID, MPC) | **Low-Level Control**: The APC system utilizes both PID and MPC control systems. | |  | Advanced Process Control (APC) | **Low-Level Control**: Supervisory settings are entered into and executed by an APC system. | |  | Optimization Techniques |  | |
| **Limitations of Current Methods** | *What are the challenges and limitations of the current method(s)?* | Metallurgists create and modify recipes that are used by the operators to determine the supervisory setpoints under various conditions.   |  |  |  | | --- | --- | --- | |  | **Limitation** | **Description** | |  | Ability to control well across scenarios / conditions | Human operators find it difficult to manage the changing particle size distribution and the changing hardness distribution of the incoming ore. | |  | Multiple or changing optimization goals |  | |  | Human Operator / Engineer Limitations | |  |  |  | | --- | --- | --- | |  | **Limitation** | **Details** | |  | Difficulty managing many variables and dimensions. |  | |  | Difficulty adapting to changing conditions | Human operators find it difficult to manage the changing particle size distribution and the changing hardness distribution of the incoming ore. | |  | Large performance discrepancy between novice and expert operators | Expert operators gain expertise over many years. | |  | Inconsistency across expert operators |  | | |  | Uncertainty in the measurement of the inputs or the process make it difficult to control or optimize. | The particle size distribution and the hardness distribution of the incoming ore is unknown. | |  | Time to develop control or optimization system is prohibitive |  | |
| **Machine Teaching Strategy** | We use subject matter expertise to help determine the solution architecture of the BRAIN.  **Heuristics**   |  |  |  | | --- | --- | --- | | **When the [environment variable list] trend in this direction or interact in this way,** | **This is what we think it means.** | **This is what you should do (to manipulate control actions).** | | If incoming particle sizes are larger and / or ore is harder | More compression is required | Choke the crusher | | If incoming particle sizes are smaller and / or ore is softer | This is a higher throughput opportunity | Regulate the crusher | | If the crusher is getting too full, especially with smaller particles. | You may bog (clog up) the crusher in which case it will need to be stopped and manually unclogged. | Slow down the feed into the crusher |   BRAIN(s) can be decomposed into one or more subcomponent concepts as needed. First, we decompose the problem into independent subcomponents, second we determine which technology is best suited to make the decisions of each subcomponent then we determine how to orchestrate the interaction between the concepts.  **Concept Network Decomposition**  The first two heuristics above point to a decomposition that leverages two explainable strategies. One chokes the crusher to maximize compression and works well for larger, harder particles. The second regulates the crusher and works well for smaller, softer particles.  This architecture provides an additional layer of explain-ability so that the BRAIN reports which strategy it is deploying in addition to the control actions.  A close up of a logo  Description automatically generated  **Concept 1**: Choke the Crusher Feed   * This concept learns to keep the crusher full (choke strategy) under various conditions. * This concept outputs the ROM Feeder Speed and Crusher Gap Action.   **Concept 2**: Regulate the Crusher Feed   * This concept learns to drip feed the crusher (regulate strategy) under various conditions. * This concept outputs the ROM Feeder Speed and Crusher Gap Action.   **Selector Concept**: Select Feed Strategy   * This concept decides which crusher strategy to deploy. * This concept outputs the following:   + A binary decision (1 = use ROM Feeder Speed and Crusher Gap Actions from Concept 1, 2 = use ROM Feeder Speed and Crusher Gap Actions from Concept 2)   + The Throat Level action. The Throat Level action accompanies and in some ways determines the crusher feed strategy.   + This concept helps explain the control actions in terms of a human readable strategy.   + This concept also helps discover under which conditions each crusher strategy is most appropriate. | |
| **Control Actions** | *What actions will the BRAIN need to output to control or optimize your system?* | **Figure 2**: Crusher Gap Setting  A close up of a map  Description automatically generated   |  |  |  |  | | --- | --- | --- | --- | |  | **Level** | **Number of Actions** | **Description** | |  | Supervisory | 3 | The BRAIN will provide supervisory set points. | |  | Low-level |  | Low-level control will remain with the APC controllers. |   **Low-Level Control System**: If the brain will provide supervisory control actions, is there a low-level system (APC, MPC, etc.) that must be included in the training loop? If yes, this control system must be integrated with the simulator and documented in the Simulation section below.   |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | | **Name** | **Data Type** | **Units** | **Control Frequency** | **Operating Range [min, max]** | **Description** | | ROM Feeder Speed | Decimal | [units] | [frequency] | [min, max] | The speed of the crusher feeder; this dictates how much ore is coming into the crusher. | | Crusher Gap Setpoint | Decimal | millimeters (mm) | hourly (once per hour) | [90, 165] | The setpoint for the smallest distance (gap) between between the mantle liner and concave liner on the closed side  The crusher gap is currently changed in minimum increments of 10mm. For example, if the gap needs to be changed by 5mm, it will be increased by 15mm and then decreased by 10mm. | | Throat Level | Decimal | percentage | hourly (once per hour) | [min, max] | Vertical distance between the top of the crusher and the crusher gap |   **Delayed Reward Scenario**: It takes a matter of minutes for most ore to move through the crusher and 15 minutes to get an average fragmentation time. We do not have a delayed reward scenario.  *Rule of thumb*: For optimal learning, system should settle to steady state within 1/10th of the control frequency. |
| **Constraints** | *What constraints are placed on the control actions by the system or the process?* | * Engineering limit on the Mantle Height: 25mm to 275mm. Control actions that result in mantle heights of less than 30mm should not be used. * Maximum crusher gap change allowed per hour is 15mm. |
| **Environment States** | *What information do we need to pass to the BRAIN about the system and its environment for the BRAIN to learn to control or optimize the system?* | **Process Variables**   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | **Name** | **Data Type** | **Source** | **Units** | **Measurement Frequency** | **Operating Range [min, max]** | **Description** | | ROM Bin Level | Decimal | ROM Bin Input Predictor | percentage | [frequency] | [min, max] | Percent of capacity that the ROM bin is filled | | ROM Feeder Speed | Decimal | Simulation | [units] | [frequency] | [min, max] | The speed of the feeder delivering uncrushed rock into the crusher​ | | COB Bin Level | Decimal | COB Bin Output Predictor | percentage | [frequency] | [min, max] | Percent of capacity that the Crushed Ore Bin (COB) bin is filled | | COB Low/High Microwave Sensors | Boolean | Simulation | N/A | [frequency] | [min, max] | Sensors for determining if the COB is either approaching full or empty. Will trigger stops for feeder conveyor belts when reached.​ | | COB Feeder Speed | Decimal | Simulation | [units] | [frequency] | [min, max] | The speed of the feeder delivering uncrushed rock out of the COB bin to the main conveyor​ | | Throughput | Decimal | Simulation | tons per hour | [frequency] | [min, max] | The amount of ore that passes through the crusher and goes to HPGR having met the 65mm quality criteria. | | Weight-o-meter | Decimal | Simulation | tons per hour | [frequency] | [min, max] | The amount of ore that leaves the crushed ore bin | | Fragmentation | Decimal | Rock Hardness and Fragmentation Predictor (training), Ore Size Camera (deployment) | millimeters (mm) | [frequency] | [min, max] | Ore size as reported by ore size camera | | Current Gap Setpoint | Decimal | Simulation | millimeters (mm) | [frequency] | [min, max] | The setpoint for the smallest distance (gap) between between the mantle liner and concave liner on the closed side | | Throat Box Sensor | Decimal | Simulation | [units] | [frequency] | [min, max] | Provides an indication of the rock level within the crusher | | Crusher Power | Decimal | Simulation | kWh | [frequency] | [min, max] | Power delivered to operate the crusher | | Mantle Height | Decimal | Simulation | millimeters (mm) | [frequency] | [min, max] | The linear height of the hydraulics controlling the crusher gap width as measured by the mantle linear sensor | | Crusher liner wear | Decimal | Liner Predictor | [units] | [frequency] | [min, max] |  | | Demand predictions | Decimal | [source] | tons | [frequency] | [min, max] | Amount of ore expected to be mined | | Surface Expected | Decimal | [source] | tons | [frequency] | [min, max] | Amount of ore expected on the surface | | Ore Type Prediction | Categorical | [source] | [units] | [frequency] | [min, max] | Type of ore expected | |
| **Deep Reinforcement Learning** | Deep Reinforcement Learning algorithms train agents to make sequential decisions which are assessed for the affect that each decision has on the environment.    For each concept that we will train using Deep Reinforcement Learning, we outline the sequential decision     |  |  |  |  |  | | --- | --- | --- | --- | --- | | **Concept** | **Action** | **State**: How does the Environment change when the control actions are taken? | **Reward** | **Configuration**: What do we need to vary in the training to ensure that the BRAIN works well across scenarios? | | Choke Crusher and Regulate Crusher have the same DRL actions. | Set feeder speed, crusher gap and throat. | Each time a decision is made to feed and set a crusher, the resulting throughput of ore that passes through the particle sieve changes. | Throughput | * Particle Size Distribution * Ore Hardness Distribution | | |
| **Configuration Scenarios** | *What scenarios should the trained BRAIN be able to control across?* | Deep Reinforcement Learning (DRL) can produce BRAIN(s) that control well across a wide range of scenarios and is particularly suitable for situations where the distribution of the variables in the configuration scenarios is unknown and / or non-linear.   |  |  |  | | --- | --- | --- | | **Configuration Variable** | **Range [min, max]** | **Description** | | Particle Size Distribution | [min, max] | [Construct a synthetic set of particle size distributions that represent the variation of input particle sizes experienced at MineCo South during normal and perhaps outside normal operations] | | Ore Hardness Distribution | [min, max] | [We need to construct a synthetic set of particle size distributions that represent the variation of input particle hardness experienced at MineCo South during normal and perhaps outside normal operations] | | Liner Wear | [min, max] | [We need to construct a synthetic set of liner wear scenarios that represent the liner wear experienced at MineCo South during normal and perhaps outside normal operations] |   **Training Episode Length**: [number of control actions included in a training scenario]  An episode represents the number of control actions that comprise a scenario. For example, in an HVAC scenario control actions for an air conditioning unit might be taken 4 times per hour, but multiple hours need to be considered to see a diverse range of building occupancy and the temperature variation during the day. If the training episode is one day, there are 24 x 4 control actions per training episode.  **Benchmark Episode Length**: [number of control actions included in a benchmark scenario]  Sometimes, the benchmark scenario needs to be longer than the training scenario in order to capture the full range of variation of the configuration variables. To extend the example above, benchmarking an HVAC system requires extending the prediction scenario for a trained BRAIN to include seasonality across months. In this case, the benchmark episode length may be 1 year (356 x 24 x 4 control actions). |
| **Success Criteria** | *What criteria will we use to determine the success of the project and how will we measure that success criteria?* | |  |  | | --- | --- | | **KPI** | Throughput (tons/day) | | **Benchmark Comparison** | The BRAIN will be compared to a human operator. | | **Benchmark Scenarios** | |  |  |  |  | | --- | --- | --- | --- | | **Configuration Variable** | **Units** | **Priority** | **Range or Description** | | Particle Size Distribution | millimeters (mm) | 1 | See Configuration Scenarios above. | | Ore Hardness Distribution | [units] | 1 | See Configuration Scenarios above. | | Liner Wear | [units] | 1 | See Configuration Scenarios above. | | | **Benchmark Procedure** | |  |  |  | | --- | --- | --- | |  | **Procedure** | **Duration** | |  | Simulation | [Benchmark Duration in Simulation] | |  | A/B Testing on Live System | [A/B Testing on Live System] | | | **Optimization Improvement** | [success criteria expressed in % improvement over current methods] | | **Return on Investment (ROI)** | [success criteria expressed in return on investment (ROI)] | | **Project Readout and Deliverables** | [expected deliverables besides the BRAIN(s) and a PowerPoint readout report] | |
| **Simulation** |  | |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | | **Readiness** | |  |  | | --- | --- | | **Delivery Date** | [Sim Delivery Date] | | **Validation Date** | [Sim Validation Date] | | **Sim Builder** | [Sim Builder] | | **Integration with Microsoft Machine Teaching Service** | Python SDK3 | | | **Type** | |  |  | | --- | --- | | **Vendor** |  | | **Product (Version)** | N/A, custom sim | | **Software Language**  **API Interface** | Python | | **Speed** | Less than 1 second per iteration | | | **Modeling Method** | |  |  |  | | --- | --- | --- | |  | **Method** | **Description** | |  | Physics Based (First Principles) |  | |  | Discrete Event |  | |  | Surrogate Model |  | |  | Model from Data | The amount (number of rows) of data required to create a simulation model from data varies, but use the following rule of thumb as an absolute minimum: the number of possible states x the number of possible actions. For example, if there are 10 possible actions and 100 possible states you’d need 1,000 rows of data at minimum to build a model.  **Model Accuracy & Robustness**  The model should be validated across the ranges for each of the control actions and environment states listed above. Enter the accuracy of the model for each of the features.   |  |  | | --- | --- | | **Feature** | **Accuracy** | | [One row for each control action and environment state] | [% Error] |   **State Space Completion**   |  |  |  | | --- | --- | --- | |  | **Procedure** | **Rows of Data** | |  | State Space Parameter Sweep | [data volume] | |  | Synthetic State Space Estimation | [data volume] | | | | **Connection** | *Can we exchange messages (input and output) with the simulation model at the simulated control frequency?*  [Can this software connect for input and output on the inner loop?] | | **Configuration** | *Can we input the configuration scenarios programmatically into the simulation model?*  [Can we input configuration scenarios programmatically into the simulation model?] | | **Parallelization (Licensing)** | *Can we run 10, 100 or 1000 copies of your simulation in the Azure cloud?*  [Can we run the simulation in the Azure Cloud?] | | **Simulation to Reality** | *Has the simulation model been used to design a control system, an optimization system or used by human operators to control the system.*  No.  *What is the error percentage that describes the accuracy between the simulation model and the real system across all scenarios and equipment that will be controlled by the BRAIN?*  [% error] | |
| **Supplementary Decision Models** | Will Machine Learning (ML) models or other decision-making technology be used to supplement the environment state from the simulator? | |  |  |  |  | | --- | --- | --- | --- | | **Type** | **Training Data** | **Model Accuracy** | **Description** | | Liner Predictor | [training data] | [model accuracy] | [description] | | ROM Bin Input Predictor | [training data] | [model accuracy] | [description] | | COB Bin Output Predictor | [training data] | [model accuracy] | [description] | | Rock Hardness Fragmentation Predictor | [training data] | [model accuracy] | [description] | | Maximum Safe Crusher Gap Predictor | [training data] | [model accuracy] | [description] |  |  |  | | --- | --- | | **Delivery Date** | [Delivery Date] | | **Validation Date** | [Validation Date] | | **Model Builder** | MineCo | | **Integration with Microsoft Machine Teaching Service** | Python SDK2 | |
| **Deployment** | *How will the BRAIN interface with your system? (select & respond to one or multiple options below)* | |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | |  | **Decision Support** | *Human engineers, operators or analysts will continue to control and automate my system augmented by BRAIN decisions.*  **Cloud Deployment**  **Edge Deployment**  **Embedded Deployment**  Integration with OT environment will be through existing IoT/OPC Gateway  Edge infrastructure enabled to host containers, within the IT environment.   |  |  |  | | --- | --- | --- | |  | **Decision Delivery Mechanism** | **Description** | |  | Decision Support UI |  | |  | Spreadsheet or other mechanism |  | |  | Integration with current reporting system |  | | |  | **Direct Control** | *The BRAIN will connect to the system directly to automate the control or optimization.*  **Cloud Deployment**  **Edge Deployment**  **Embedded Deployment**  For Edge and Embedded Deployments:   |  |  | | --- | --- | | **Device Type** | [Device Type] | | **Number of Devices** | [Number of Devices] | | **Device Lifecycle** | [Device Lifecycle] | | **Docker Support** | [Docker Support] | | **Processor** | [Processor] | | **Connection Protocol** | [Connection Protocol] | | **Integrator** | [Integrator] | | **Integration Delivery Date** | [Integration Delivery Date] |   [Diagram of control] | |
| **Team** |  | |  |  | | --- | --- | | **Executive Sponsor** | [Executive Sponsor Name] | | **Machine Teacher** | [Machine Teacher Name] | | **Data Scientist (Optional)** | [Data Scientist Name] | | **Subject Matter Expert** | [Subject Matter Expert Name] | | **Simulation Expert** | [Simulation Expert Name] | | **Deployment Expert** | [Deployment Expert Team] | | **IT** | [IT Contact Name] | | **Project Team** | [Are there any other members of the MineCo team that should be included weekly status updates and any other regular project communication in addition to the members listed here?] | | **Services Partner** | [Services Partner Team] | | **Microsoft Applied AI Engineer** | [Project Applied AI Engineer] | | **Microsoft Technical Program Manager** | [Project Technical Program Manager] | | **Microsoft Account Team** | [Account Executive, Account Technical Strategist Names] | | **Microsoft CSA** | [CSA Name] | |
| **Azure Infrastructure** |  | |  |  | | --- | --- | | **Azure Subscription** | [Subscription ID] | | **Other Azure Services Required** | [List of Azure Services] | |