AI with constraints for robots

1) Introduction

Apparently some material comes from the universe trough minding on the planet. This is a energy consuming issue there the only target is to expand our routes to other resources than earth. Investing and measuring can bring problems for our atmosphere or disease on the normal functioning human organic substance. Therefore would be better to build up remote environments which are accessible for humans the only thing lacking is the food and oxygen. Therefore we can test the AI in advance (means a HARDWARE created robot with human way of **creating structures**) in certain location. Let's say that the **first constraint** we have is to create (means building structures with the existing resources in a radius area around the best chosen location on the surface). The **second constraint** is to not reproduce itself.

2) Universal hardware execution elements

1) Different configurations of robot can have different constellations. Means the categorisation of elements provided on this hardware.
2) Categorisations
    1) Moving operation
    2) Measuring operation
    3) Mining operation
    4) Constructing operation
3) Model with defined categorisation parts (file extensions obj, stl, step, ...)

3) AI module

The software manipulating this hardware model according the categorisation (reaching the target of operation (#1), for finalising the wanted structure). This is a PC of hardware powerful enough to break down the different cycles (#2) and then break down the cycles in different steps (#3), each cycle is calculated in real time simulations for the movement of the robot (in 3D environment non graphical for avoiding CPU overload) with each used categorisation element for the step.

If the simulation has reached it's goal of the cycle (#2) there should be a **relocation** or **retry according other path** for solving the current cycle.
suppress/not suppress cycle is optional and to set.

In case the hardware execution is error of cycle (#2) we do a **failure handling** (retry for certain amount):
1) Restore procedure
2) Retake the simulation **according the constraints** of the previous hardware execution
3) Perform cycle according new constraints

In case **failure handling** after this fail cycle relocate or retry according other cycle (#2).

Defining the cycles it will happen with **logarithm.** These are steps to be combined according the categorisation elements.

Step 0:
DEFINE REFERENCE LOCATION POINT (SET MARK) before performing the following steps

Step 1:
There are **4 level logarithms calculations for object deconstruction**:
- Level 1: Break down into operations (#1).
- Level 2: Break down into cycles (#2).
- Level 3: Break down into steps (#3).
- Level 4: Case of **categorisation constructing** (additional a combining cycle will be added to the flow, means combining one material to the other in different steps).

The 4 logarithms have to have **different parameters (tolerances, speed, fragmentations) according DNA string** input.
So exceeding the scope of this document is to attach the DNA string to certain parameters into to the logarithm. These **parameters can be seen as constraints** besides the hardware possibility's of the robot.

There is **1 planning logarithm** logarithm takes care of the sequential steps and environment:
- Calculates needed power from current reached cycle to final cycle.
- Calculates power gain over a certain time.
- Calculates needed power for reloading.
- Safety parameters for this issue must be recalculated frequently every time.
- Calculates needed power for reloading must be recalculated frequently cause of time before reloading depends on factors (distance or time, for example solar, for example electricity, …).
- Include the maintenance logarithm.
- Delta safety must be included, mean safety power in case of unforeseen circumstances.

There is **1 maintenance logarithm** to be taken care of the sequential environment:
- Sequential action defined for maintenance. This one is hardware specific. Must be defined according each different hardware robot model.

There is **1 fail over logarithm** to be performed until cycle is ok or dead lock has been reached:
- Restore procedure
- Retake the simulation **according the constraints** of the previous hardware execution
- Perform cycle according new constraints

There is **1 dead lock logarithm** this to go back to the beginning of the current cycle if cycle has reached a dead end or break of material.
- Material check
- Define steps to go back to previous point of the current cycle.
- Evoke maintenance logarithm or not.
- This one include the error handling procedure.
- **Exception: Could go in to the dead lock !!!!!!!!!**

Step 2:
There is **1 Exploring logarithm**:
Step 1 defines the needed materials. This includes that the exploration of the current source is possible to do according the location of this materials in percentage existence on the source.
This means we have to first define the parameters for the planning logaritm.
Also we have to find the resources for step 1.

This means we do in the exploring logarithm a measure period, first to set the planning logarithm parameters in place, second to explore the materials until all resources are found.

This includes the categorisations of the moving and the measuring. According a radius of the certain position (means initialisation place).

Interesting is the distances of search locations, we got one a radius, 2 the distance of different measure points, 3 heights/depths. Must be recalculated until all resources are found, this means after current plane not containing all resources, extend the factor and recalculate but subtract current plane from the reference point.

- Surface measurements
- Below surfaces measurements
- Atmosphere measurements

We will perform this exploring logarithm until all resources are found according the quality percentage of the structure defined in the object file (means for example 60 – 100 percent correct). Each of the measurement points will be saved according the reference point and stocked into the memory.

Step 4:

The source map is the locations of the founded resources.
Could be on different locations.

Step 5:

Execution (Performance) … .
- Do the Exploring logarithm included with the maintenance logarithm (indicator is the hardware).
- Do the planning logarithm according source map.

→ **Here we go;-)**

- Execute the planning logarithm.

4) **AI used logic**

Wanted structure

.OBJ, .dae, … data for input into the AI environment

TARGET

- Following parameters included
  - Material
  - Sizes
  - (optional) contruction order

Used hardware

.OBJ, .dae, … data for input into the AI Environment with the hardware categorisations defined on the object. This hardware categorisations are set on each functional part of the robot.

Subdefine the categorie in units for executing a certain movement.

Executing categories

Executing the category is according the universal interface, this as a queue to translate the standard step to a certain command for the hardware part who is able to perform this category action.

Each hardware part has a certain factory library to connect to for manipulation of the device. There fore we have a queue interface box which can Plug-in the library to translate the AI categorized command to a certain routine. This routine can be preformed as programmed with the output parameters (coordinates or the real time simulation area according zero point of the location where the device is connected, and again with the additional coordination system to the reference point (equals initialisation point of STEP 0).

The universal interface has some commands which can be defined with feedback.
- HW-DEVICES (General config for connecting):
  - **MOVE** (Moving command)
  - **MESU** (Measuring command, means camera, electricity, atmosphere, ...)
  - **SUBS** (Minding devices)
  - **ASSE** (Assemble a material)

- Each HW device has some commands general commands:
STATE;EXECUTE=0/1;WAIT=0/1;
START;X=123,123;Y=543,879;Z=634,9864
ABORT;(Abort the procedure)
PROCE;ProcedureName=Procedure_01;KeyName_01=AAA;KeyName_02=BBB;KeyName_03=AAA;KeyName_04=AAA(Optional command)
CONN;USER;PASS(Connecting)
DISS;(Disconnecting)
MAINT;

ERROR;VALUE='#' (queue, FIFO, and removed when readed)
//TIMEOUT ms
TIMEOUT:VALUE='3000'
//ENUM METRIC=1, Volt=2, Ampere=3, Herz=4, Mol=3, (queue, FIFO, and removed when readed)
//Moliculaire thickness to compare with mass of material to provide the thickness
RESULT:RETURN_PROCEDURE='0/1';RETURN_UNIT='1';RETURN_NUMBER='';RETURN_STRING='';

- Each HW device has some settings according config file:
UNIVERSAL_DEVICE_PARAM_MAINTENANCE_SECURITY_RATE
…...
SPECIFIC_DEVICE_PARAM_PARAMNAME
……

Example config file:

```
<CONFIG>
<UNIV_CONNECTION_PARAMETERS>
<!-- JBUS=1; MODBUS=2; SERIAL_DB9=3; SERIAL_DB25=4; SOKETS=5;MYSQL=6; →
<connection_type>2</connection_type>
<DB_CONNECTION_STRING><DB_CONNECTION_STRING>
<IP></IP>
<SERIAL_PORT_NUMBER></SERIAL_PORT_NUMBER>
<UNC_PATH></UNC_PATH>
<FTP></FTP>
<USB></USB>
</UNIV_CONNECTION_PARAMETERS>
<SPECIFIC_CONNECTION_PARAMETERS>
<TAGNAME_01></TAGNAME_01>
<TAGNAME_02></TAGNAME_02>
….
</SPECIFIC_CONNECTION_PARAMETERS>
<PROCEDURES>
<PROCEDURE ProcedureName=""><![CDATA[
Sub Main

        Dim Machine As HardwareDevice
        Set Machine = WcdSystem.ThisDevice

        Dim ID As String
        ID = Machine.Id
```

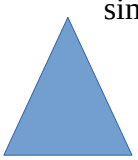
```
		Debug.Print  "Machine ID=";ID
		If ( Left(ID,6) <> "ID_GFM" ) Then
			' Not for me
			Exit All
		End If

		'
		' Check the command , we are waiting for open door command
		'
		If ( Machine.LastCommand = "RU" ) Then
		Debug.Print  "LAST COMMAND RU"
		Debug.Print LCase(Machine.LastParameters(0))
			If ( LCase(Machine.LastParameters(0)) = "wp_load.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_load_mach1_its148.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_load_mach1_its72v.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_load_mach2_its148.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_load_mach2_its72v.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_unload_mach1_its148.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_unload_mach1_its72v.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_unload_mach2_its148.xmlj" Or
LCase(Machine.LastParameters(0)) = "wp_unload_mach2_its72v.xmlj" Or
LCase(Machine.LastParameters(0)) = "tool_load_mach1.xmlj" Or
LCase(Machine.LastParameters(0)) = "tool_load_mach2.xmlj" Or
LCase(Machine.LastParameters(0)) = "tool_unload.xmlj" ) Then
		'If ( LCase(Machine.LastParameters(0)) = "loadworkpiece.xmlj" Or
LCase(Machine.LastParameters(0)) = "unloadworkpiece.xmlj" Or
LCase(Machine.LastParameters(0)) = "loadelectrode.xmlj" Or LCase(Machine.LastParameters(0))
= "unloadtool.xmlj" ) Then

				ID = ID + "_BP"
			Debug.Print  ID
				Dim SisterMachine As HardwareDevice
				Set SisterMachine = HardwareDevices.GetHardwareDevice(ID)

				Dim pars(1) As String
				pars(0) = "LOADING"

				SisterMachine.ExecuteCommand("OP",pars)
			Debug.Print "Command executed"

			End If

		End If

	End Sub
]]></PROCEDURE>
</PROCEDURES>
</CONFIG>
```

Simulation of steps according category, this simulation will provide the best fit parameters which can be transferred to the Hardware procedure. Simulations area has object sizes and the skeletons which can simulate the movement of the hardware.

Category 1: A to B

- Known sources: deploy location
- Known sources: A ref location HW center according deploy location
- Known sources: B target location HW center according deploy location

Route defined from explore logarithm (scanned area) after real time simulation.



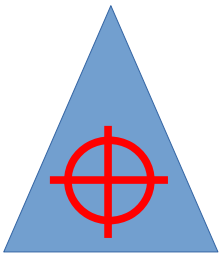
Categorie 1

A

B

Category 2: Measure operation

Measure operation:
- Measure the target with specific device
- Measure result transfer over definition in unit and value
-



Category 3: Mining operation

- Mining target defined
- Recalculate cycle if not ended or delta is exeeded
- Known sources: B target location HW center according deploy location

Route defined from explore logarithm (scanned area) after real time simulation.

Category 4: Constructing operation

483d3fb739e75980b0c1f62681f64a1bb4b3612f2d9fa972b8c18fbfae055d66

1) **Target goal of accuracy according used material**

Delta predefined, how much we can go of limits according the to build object for the material.
Delta predefined parameter in compliance with previous one, action radius.

2) **Step simulation (REAL TIME in 3D environment one graphical)**

Before executing a cycle we have to do a 3D simulation with or without graphical support. This happen in a real time environment. Means defining the location of the to perform steps which have been planned. The 3D model of the robot will say which positions to take and which paths to take to reach with the hardware the goal for the current step. This includes the angels and length, with, height, of the hardware to finalise the step.

3) **Used units for the Hardware devices**

- Time should be digital
- Distance: …, km, m, dm, cm, mm, …
- Measurements: Molecularity measurements