On the Exploitation of Automated Planning for Reducing Machine Tools Energy Consumption Between Manufacturing Operations

Simon Parkinson, Andrew Longstaff, Simon Fletcher, Mauro Vallati
University of Huddersfield, UK
firstname.surname@hud.ac.uk

Lukáš Chrpa
Czech Technical University in Prague & Charles University in Prague, CZ
chrpaluk@fel.cvut.cz

Abstract

There has recently been an increased emphasis on reducing energy consumption in manufacturing, driven by the fluctuations in energy costs and the growing importance given to environmental impact of manufactured goods. Lots of attention has been given to the reduction of machine tools energy consumption, as they require large amounts of energy to perform manufacturing tasks.

One area that has received relatively little interest, yet could harness great potential, is reducing energy consumption by planning machine activities between manufacturing operations, while the machine is not in use. The intuitive option – which is currently exploited in manufacturing– is to leave the machine in a normal operating state in anticipation of the next manufacturing job. However, this is far from optimal due to the thermal deformation phenomenon, which usually require an energy-intensive warm-up cycle in order to bring all the components (e.g. spindle motor) into a suitable (stable) state for actual machining. Evidently, the use of this strategy comes with the associated commercial and environmental repercussions.

In this paper, we investigate the exploitability of automated planning techniques for planning machine activities between manufacturing operations. We present a PDDL 2.2 formulation of the task that considers energy consumption, thermal deformation, and accuracy. We then demonstrate the effectiveness of the proposed approach using a case study which considers real-world data.

Introduction

Machine tools are complex mechantronic system used in both subtractive and additive manufacturing. Much of their performance is due to their mechanical rigidity. Machine tools come in a large variety of sizes and configurations, but a common feature is their ability to position their tool in a three-dimensional space relative to the workpiece either to remove (cut, grind, etc.) or add material. Accuracy is often a primary commercial driver in the advancement of machine tools for precision, high-value manufacturing to micrometre-level tolerances. However, maintaining such high levels of accuracy requires strict control of the many factors which can cause a change in accuracy. For example, the effect of temperature change on the machine's structure can have a dramatic impact on the accuracy of the tool.

Energy efficiency is also becoming an increasingly important factor in machine tool development both to reduce manufacturing costs (Draganescu, Gheorghe, and Doicin 2003; Diaz, Redelsheimer, and Dornfeld 2011), as well as reducing environmental impact (Diaz et al. 2010).

The use of machine tools has been identified as the largest consumer of energy *during* the manufacturing of parts. It has been established that machine tools use 63% of the total energy required to manufacture a part (Hesselbach and Herrmann 2011). Additionally, energy consumption occupies over 20% of the operating costs of machine tools per year, in excess of £10,000 for a medium-sized manufacturer. Many researchers have investigated the potential of reducing energy consumption during the manufacturing process itself (Vijayaraghavan and Dornfeld 2010; Liu et al. 2014a; 2015; Diaz, Redelsheimer, and Dornfeld 2011).These works have largely been motivated by the fact that large forces are required to cut material, and any reduction at this stage can therefore be significant. Furthermore, researchers have studied the Job Shop scheduling problem to reduce energy consumption and improve manufacturing throughput (Fang et al. 2013). However, such approaches often ignore the relationship between energy consumption, thermal deformation and the machine tool's accuracy. This is of significance as the desire to improve machining capabilities to a sub-micron level cannot be achieved without thorough thermal analysis.

One area that has received less attention is the consumption of energy *between* manufacturing operations, when the machine is not actively working on a piece, and is therefore nominally idle. In the first instance it may appear that if the machine is idle it will consume no energy. On the contrary, many electrical components of a machine tool will continue to use energy. Furthermore, once the machine is required to operate again, an energy-intensive warm-up cycle is usually required to bring all the subsystems (e.g., spindle motor) into a suitable (stable) state for actual machining. However, a warm-up cycle will only be necessary should the heat-generating subsystem and surrounding structure decrease below an identified temperature. Currently, machine activities between manufacturing operations are not planned, i.e. machines are left on idle, and warm-up cycles are performed before the next operation. This creates an interesting possibility for automated planning. It can automatically decide to keep some of the machine subsystems active, at a reduced level, to generate sufficient heat to maintain the thermal stability of the machine tool's structure and remove the need for a warm-up cycle, thus reducing the overall energy consumption. The number of electrical subsystems, the different operational levels, the current state of the machine tool, and the required initial state of the next manufacturing operation make it challenging to consider all possible options and minimise energy consumption whilst maintaining a desired level of accuracy, that allows the machine tool to effectively starts the next operation.

Although planning techniques have not yet been exploited for planning machine activities between manufacturing operations, previous works (see, e.g., (Parkinson et al. 2012a; 2012b)) demonstrated the potential of using automated planning for optimising different aspects of using machine tools. Remarkably, many researchers have studied the impact of temperature change on the machine's structure (Creighton et al. 2010; Eberspächer, Lechler, and Verl 2015), as well as energy consumption (Peng and Xu 2014; Abele et al. 2011). Other recent work also includes modelling machine tool energy performance (Frigerio et al. 2013; Jeon and Prabhu 2013) and the relationship between energy consumption and temperature generation (Li and Kara 2011; Li, Yan, and Xing 2013; Mayr et al. 2012). Work has also been conducted on using scheduling techniques for manufacturing job which does take some consideration to energy consumption (Wang et al. 2015).

In this work we investigate the use of AI planning for performing the interval planning task. Specifically, we engineer a PDDL 2.2 planning domain model that takes into consideration the relation between temperature, energy consumption and accuracy. We then demonstrate the effectiveness of the planning-based approach using real-world case studies. Generated plans have been validated by human experts, and their feasibility assessed using the actual manufacturing tools.

Background

In this section, we motivate the importance of considering interval activity, and briefly introduce the language used for encoding the domain model.

Importance of Interval Activity

Figure 1 (coloured) provides a graphical illustration of two manufacturing operations with an interval between. The figure illustrates the relationship between increasing energy consumption (green), heat generation (red), and increasing machine error (blue) through a simplified representation. Note that although the figure is for illustration purposes, the data is a realistic, if simplified, representation of what occurs. In the figure, it can first be seen that energy consumption is at its lowest when the machine is idle, and its highest when a new manufacturing job is started. This is because

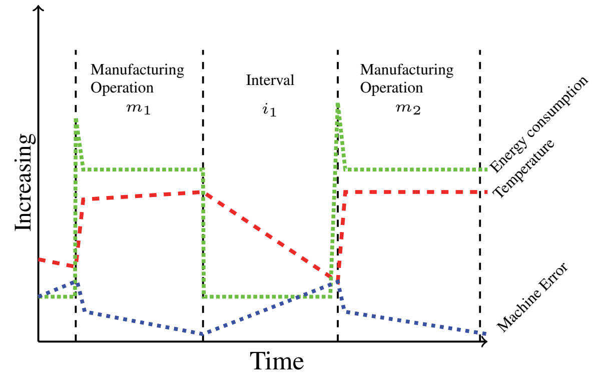

Figure 1: Illustrating how the machine tool energy consumption, structural temperature, and accuracy is changing during manufacturing and interval periods.

a dedicated warm-up cycle is required to stabilise the machine's structure, stabilise the machine error and avoid thermal change during manufacturing. It is then noticeable in the figure that as the energy consumption increases, so does the temperature of the machine's structure. The final relationship presented is that the error of the machine tool increases to a steady-state value and maintained when the temperature is stable. In practice, the number of different operations that occur during machining mean that the energy profile, and resulting temperature and error trends, will display somewhat more complex behaviour.

For the sake of readability, the graphical illustration in Figure 1 shows data about a machine tool as a whole system; however, each subsystem of the machine reacts differently, and has to be considered. A high-speed spindle motor uses significantly more power than a linear axis servo motor. For example, consider manufacturing an aluminium housing with the dimensions of 150mm $\times$ 50mm $\times$ 20mm (Heidenhain 2010). The total energy needed for the machine tool to produce the part is 20.4kW. A total of 4.8kW for the machine tool spindle, and 0.5kW for the three axes' feed drives. Other electrical subsystems (e.g controller, coolant pump, etc.) make up the remainder. As both these components have different levels of energy consumption, they generate different amounts of heat. Moreover, the heat generated by machine subsystems transfers into the machine tool's structure causing distortion. The severity of the effect of changing temperature is dependent on the material from which it is constructed. For example, steel has a high coefficient of thermal expansion (~12μm per °C) compared to carbon fibre (~2μm per °C).

For the purpose of planning machine activity between manufacturing operations, there are two important aspects that have to be considered. First, the relationship between energy consumption, generation of temperature profile, and the effect on machining accuracy. Second, the effect that changing machine tool temperature has on structural deformation of the machine tool. Prior knowledge of these aspects creates the potential to optimise machine tool use between manufacturing operations. For example, in some situations, it may be advantageous to keep the electronic components

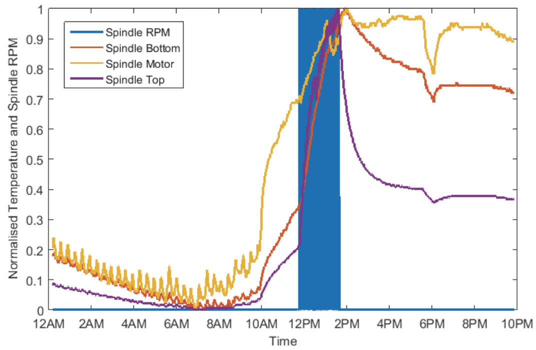

Figure 2: Heat generated by spindle motor during a two-hour heating and cooling cycle. Spindle bottom, top and motor indicate the normalised temperature for three surface temperature sensors mounted around the spindle. Spindle RPM reports the normalised spindle speed in RPM (0 to 9,000).

in use to maintain energy consumption, generate heat, and thus maintain machine accuracy.

Figure 2 demonstrates the heat generated as the spindle speed increases on a three-axis machine tool. The figure shows the normalised spindle speed in RPM (0 to 9,000), and normalised temperature for three surface temperature sensors mounted around the spindle. There are: (1) spindle bottom (21.8°C to 27.4°C), (2) spindle motor (21.6°C to 26.2°C), and (3) spindle top (21.7°C to 33.1°C). The graph illustrates that when the spindle is used at its higher speed, the temperature of the machine tool's structure surrounding the spindle increases rapidly in temperature. Once high speed usage has finished, it can be seen that the structure of the machine tool begins to reduce in temperature.

Currently, machine operators do not plan any activity between manufacturing operations, and they rely on energy-expensive warm-up cycles. This is also because planning for the machine operator is complicated by the large number of different machine activity actions that can be performed and their potential implications on machine accuracy and energy consumption. For example, each axis and spindle can be moved at different speeds sequentially or concurrently for different periods of time. Moving a single linear axis will transfer heat in the machine's structure surrounding the axis and would result in thermal distortion from that location, whereas moving all three axes simultaneously would transfer heat into more of the machine's structure and potentially result in more symmetrical expansion.

Automated Planning

Automated Planning deals with the problem of finding a totally or partially ordered sequence of actions that transform the environment from a given initial state to some of the goal states (Ghallab, Nau, and Traverso 2004). In general, the environment and action description is represented by a *planning domain model*, and the description of the initial state and the goal conditions is represented by a *planning problem description*. It is worth noting that more planning problem descriptions can be used with a single planning domain model.

In this paper, we use PDDL 2.2 (Edelkamp and Hoffmann 2004), which is an extension of well known PDDL 2.1 (Fox and Long 2003), a language for describing the planning domain model and planning problem descriptions. Notably, the International Planning Competition[1] has resulted in the existence of a significant number of planners able to deal with PDDL 2.2. The environment in PDDL 2.2 is described by *predicates* and *numeric fluents*. *Actions* are specified via their execution *duration*, *preconditions* which are logical expressions that must hold in order to make the action executable, and *effects* which are sets of literals or fluent assignments that take place when the action is executed. In PDDL 2.2, preconditions can take place just before the action is executed, during the action execution or just before finishing execution of the action. Similarly, effects can take place just after the action is executed, or just after finishing action execution (Fox and Long 2003). Therefore, a planning domain model consists of definitions of predicates, numeric fluents and actions. A planning problem description consists of a set of objects, an initial state (a set of grounded predicates and fluent assignments), and a set of goals (logical expressions). In the initial state we can use *time initial literals* that allow facts to become true at predefined timestamps. A *plan* is a set of triples in the form of ⟨timestamp,action,duration⟩ such that executing these actions in the corresponding timestamps for a given duration (it must always be possible) transforms the environment from the initial state to some state where all the goals are satisfied (i.e. a goal state).

The Interval Activity Domain Model

In this section, we introduce and specify the domain model of interval activity planning. In the presented model, two equations are used to determine energy consumption as well as machine tool error (i.e., change of machine tool accuracy).

$$total\ error = total\ error + (duration \times deformation) \quad (1)$$

$$total\ energy = total\ energy + (duration \times power) \quad (2)$$

Equation 1 is used for updating the *total error* fluent by *duration* in minutes, multiplied by the deformation (μm) per minute. Equation 2 updates the *total energy* fluent by the same *duration* (in minutes) multiplied by the a fluent representing *power* (in kW) of a particular component when being used in a predefined mode of operation.

These equations require specific data for each machine component that can be acquired through performing an error mapping and energy monitoring audit. Information about

[1] http://www.icaps-conference.org/index.php/Main/Competitions

energy consumption per time unit, i.e., *power* is widely available for many machine tool electrical subsystems, but that from mechanical interaction (friction) is often less well defined. However, in both cases the amount of heat released into the machine's structure and its affect on machine accuracy needs to be established. This can be acquired by recording the temperature of the machine tool's structure while monitoring the deviation of the machine tool's cutting point. Once all the data has been acquired, Finite Element Analysis can be used to computationally model the relationship between heat generation and *deformation* of the machine tool (Mian et al. 2011; 2013). This model can then be used to derive a series of coefficients that describe the generation of heat with increased energy consumption, and the change in machine tool accuracy from the resulting different thermal gradients.

Many machine subsystems, such as the spindle motor, can be run at any speed between stationary and their maximum RPM. In our domain model, we have adopted five predefined machine tool modes, namely *off, idle, normal, medium* and *high*. Also, the interval of interest (between scheduled jobs) is divided into smaller subintervals such that exactly one activity (predefined mode) per machine tool must be executed in each subinterval. Applying such a discretisation reduces the size of the problem and thus makes it easier to be handled by state-of-the-art planning engines. Although intuition might suggest that sacrificing the "continuous aspect" of the domain model would have detrimental impact on quality of plans, in practice, doing continuous changes to machine tool subsystems (e.g. adjusting the speed of a spindle motor) is not useful. This is because intervals between machine activities are long (tens of minutes) thus such a level of detail is not needed. Moreover, continuous plans would be extremely hard (or even impossible) to automatically execute. Instead, "discrete" plans are easy to generate and execute, and thus making a possible difference between what has been planned and how the plan was executed negligible.

It should also be noted that the length of the interval between manufacturing jobs –for which a plan has to be generated– is known at the start of the interval itself, if not well in advance. Manufacturing companies rely on a highly optimised and very accurate scheduling, in order to maximise the use and exploitation of manufacturing tools and minimise idle times. Intervals are due to some actions that are required to be performed, such as removing a finished workpiece from a machine or performing some routine controls, for which a very strict time window is given.

Operators

In our domain model, there are five operators representing different levels of operation: *off, idle, normal, medium* and *high*. These operators model the servo motor being turned off as well as four discretised levels of utilisation. For example, in terms of percentage utilisation this could be 0% (off), 5% (idle), 25%(normal), 50%(medium) and 100% (high). These operators share the following aspects. For each machine component `?c`, we specify a time unit (the `(time_unit ?c)` fluent) for which each of the operator has to be executed. For the purposes of our analysis,

```
(:durative-action normal
   :parameters (?c - component ?a - axis)
   :duration(= ?duration (time_unit ?c))
   :condition
   (and
    (over all (in_interval))
    (over all (belongs_to ?c ?a))
    (at start (not(in_use ?c)))
    (over all (>=(axis_position ?a)
      (axis_lower_limit ?a)))
    (over all (<=(axis_position ?a)
      (axis_upper_limit ?a)))
   )
   :effect
   (and
    (at start(in_use ?c))
    (at end (not(in_use ?c)))
    (at end (increase(axis_position ?a)
      (*(axis_movement_normal ?c ?a)
      (axis_direction ?a))))
    (at end (increase(looked_at ?c)
      (time_unit ?c)))
    (at end (increase(total_accuracy_axis ?a)
      (*(time_unit ?c)
      (effect_on_accuracy_normal ?c))))
    (at end (increase(total_energy_axis ?a)
      (*(time_unit ?c)
      (energy_consumption_normal ?c))))
))
```

Figure 3: The PDDL encoding of the *normal* operator, where the machine component `?c` on axis `?a` is planned to remain in a normal state of operation for a time unit.

here we consider that a time unit corresponds to a real-world minute. The precondition requires the execution of each of the operators to be performed in the time interval of interest (between scheduled jobs), hence the `in_interval` predicate must be true during the operator execution. Notice that `in_interval` becomes false when another job arrives to the machine, which is known in advance. Also, for a single component only one operator has to be executed at a time. This is ensured by introducing an `(in_use ?c)` predicate such that `(in_use ?c)` must be false prior to the execution of any of the operators handling `?c`, `(in_use ?c)` becomes true when any of of the operators handling `?c` is executed and becomes false again when the execution finishes. As an effect of each operator, *total error* and *total energy* are updated according to equations (1) and (2) respectively. Notice that *power* as well as *deformation* are different for each of the operators. Additionally, to ensure that there will be no "gaps" (i.e., time spans where no operator is executed) we introduce a `(looked_at ?c)` fluent which is incremented by `(time_unit ?c)` after any of the operators (handling `?c`) is executed. Then, in the goal we require for each component `?c`, the value of `(looked_at ?c)` to be equal to the time interval of interest, i.e., between scheduled jobs.

We also encode in our domain model the relationship between the use of electronic components and the physical movement of the machine tool. This feature is available for the normal, medium and high operators. This is essential as a machine may have physical restrictions on its movement which need to be considered to prevent accidental damage. For example, there may be a workpiece currently loaded on the machine tool ready for the next manufacturing job. This workpiece would be located within the machine's working volume and would limit axis movement. To account for this, we introduce a `(belongs_to ?c ?a)` predicate which is used to state which axis a component belongs to and it used to determine which axis' deformation should be updated due to its use. Then, for each axis, we introduce `(axis_lower_limit ?a - axis)` and `(axis_upper_limit ?a - axis)` fluents, and also an `(axis_position ?a - axis)` fluent. All are in millimetre units. Both lower and upper limits are used to determine whether the position of the axis is in range, and the current position is updated by a specified movement speed of the axis depending on the level of component use and type of operator (normal, medium or high). The movement is multiplied by an `(axis_direction ?a - axis)` fluent which represents the current direction of travel (1 for positive, -1 for negative). Figure 3 details the *normal* operator where the machine error, energy usage and axis position are adjusted based on a normal mode of operation.

For handling the change of direction once an axis reaches the limit, we introduce two instant operators. Figure 4 provides the operator which is in charge of changing an axis' direction from forward to reverse. The operator does not contain a precondition to determine whether the axis is at its limit or not as the operation action's precondition (Figure 3) requires that the axis position is within the limits. In our encoding, we decide to leave to the planner the decision of whether a change of direction should be applied or not. A good strategy can also avoid reaching the limits, and maintaining the movements inside the given axis range.

Initial and Goal State

The initial state specifies energy consumption and effect on machine accuracy for each predefined level of operation through the use of numeric fluents. For example, `(power_idle ?c)` and `(deformation_idle ?c)` represent the power and the deformation for a component `?c` in the idle mode. In addition, `(time_unit ?c)` fluent is introduced to specify a predetermined duration of an action that should occur to bring about a change in accuracy and energy consumption. The `(total_error)` and `(total_energy)` fluents are used in the initial state to encode information regarding the machine's current state after finishing manufacturing. In addition, timed initial literals are also used to encode the duration of the interval of interest. Using timed initial literals restricts the makespan to the duration of the interval, overcoming some planner's inability to handle concurrency in durative actions.

The `(belongs_to ?c ?a)` is used to state which axis a component belongs to and it used to determine which axis' deformation should be updated due to its use. The limits of axes' positions, movement speeds (with respect to different operators) and their initial directions are also defined in the initial state.

```
(:durative-action change-to-reverse
   :parameters(?c - component ?a - axis)
   :duration(= ?duration 0)
   :condition
   (and
    (over all(not(in_use ?c)))
    (over all (belongs_to ?c ?a))
    (at start (forward ?a)))
   :effect
   (and
    (at end
      (assign(axis_direction ?a)-1))
    (at end (assign(axis_position ?a)
      (axis_upper_limit ?a)))
    (at end (not(forward ?a)))
    (at end (reverse ?a))))
```

Figure 4: The PDDL encoding of the *reverse* operator, where the direction of a machine's axis is reversed

The goal requires that, for every considered component and every available time unit, an action has been selected. This is achieved using the `(looked_at ?c)` functions.

Plan Metric

We introduce three different metrics that can be considered by planners for optimising the quality of generated plans:

1. `(:metric minimize (total_error))`
2. `(:metric minimize (total_energy))`
3. `(:metric minimize`
   `(/(+(total_error)(total_energy))2))`

The first two metrics aim to minimise the values held in the total error and total energy fluent, whereas the third metric is used to minimise the arithmetic mean of both. This creates the potential to perform multi-objective optimisation where both error and total energy consumption are minimised for a given weighting.

Experimental Analysis

In this section, a case study is provided where interval planning is performed for two real-world machine tools when considering different interval scenarios. This includes both a three- and five-axis machine tool. A three-dimension model of the three-axis machine tool is illustrated in Figure 5. Due to the commercial sensitivity of machine tool performance data, we are unable to include a photo (or name and model) of the actual machine used in this experimentation. However, as illustrated in the provided figure, the machine is a typical C-frame three-axis machine tool. Data that can be released and that are relevant for the considered interval planning task –i.e., those related to power and deformation–, are presented in Table 1. As interval duration is in minutes, the

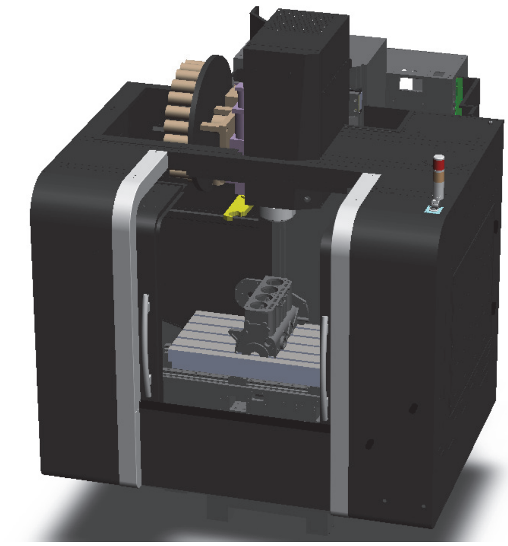

Figure 5: Three-dimensional model of the case study machine tool

Electrical Item	Off kW, µm	Idle kW, µm	Normal kW, µm	Medium kW, µm	High kW, µm
X Servo	0, 0.01	0.02, 0.01	0.01, 0.01	0.02,0.2	0.04, 0.03
Y Servo	0, 0.01	0.02, 0.01	0.01, 0.01	0.02,0.2	0.04, 0.03
Z Servo	0, 0.05	0.03, 0.01	0.01, 0.025	0.05,0.3	0.07, 0.1
Spindle Motor	0, 0.1	0.02, 0.01	0.1, 0.05	0.2,0.1	0.4, 0.2
C Servo	0, 0.1	0.04, 0.04	0.2, 0.02	0.4,0.04	0.08, 0.08
A Servo	0, 0.1	0.05, 0.042	0.3, 0.021	0.5,0.04	0.08, 0.09

Table 1: Case study data for a three-axis machine tool demonstrating power, deformation per item and mode. Power is described in kW, and deformation in µm per minute.

data presented in Table 1 has been normalized into minutes as well. These are kW for the power and the deformation in micrometres per minute. The five-axis machine tool considered in this paper is a modification of the three-axis machine with two additional rotary axes which are located on the machine's table.

The machine-specific data presented in Table 1 is now used in the creation of several PDDL problem files to simulate a number of different interval scenarios. Problem definitions are created with a duration of 30, 60, and 120 minutes. These are typical intervals duration faced by the modelled machine when operating 24 hours per day. Considering the combination of each of these scenarios results in the creation of 6 different problem instances. In addition, each problem instance will be solved using each of the three metrics stated in the domain modelling section, resulting in a total of 18 different PDDL problem definition files. As planning engine we selected LPG-td (Gerevini, Saetti, and Serina 2006), due to its exploitation in real-world planning applications, and its generally good performance. Experiments were run on a quad-core 3.0 Ghz CPU, with 4GB of available RAM. It has been given a 5 CPU-time minute cutoff for solving each problem, and optimise the quality according to the provided metric. The decision to allow a 5 minute time-frame was taken as, in an actual deployment of the planning-based proposed technique, planning for interval machine activities may be required to be done online, i.e. as soon as the machine is paused between manufacturing operations: in this scenario, waiting for more than five minutes would have a negative impact on machine accuracy. On the other hand, in cases where the length of an interval is known in advance, a higher cutoff time can be enforced. LPG-td has been used in "anytime" configuration; it keeps increasing the quality of plan, for a given problem instance, until the available CPU-time is over.

Table 2 provides the total error (Er) and energy consumption (En) values when using plans optimised for the three metrics. Plans were validated by experts, that confirmed they sound and they can be easily executed on the machine tools considered in our case study. As a further validation step, in order to check whether the encoded domain model is a reasonable abstraction of the considered physics dynamics, we executed the generated plans on the case study machines and measured –through available sensors for measuring energy use and positional deviation– the overall energy consumption and final total error. We discovered that as the data used in the domain models was acquired from repeatable measures, the measured results are in line with those shown in Table 2. This confirms that the PDDL 2.2 model has a sufficient level of detail to capture the important aspects of the relation between accuracy, energy consumption and deformation.

From Table 2 it is evident that all the problem instances were solved within the given cutoff time. LPG takes on average 76 and 83 seconds to generate the plans shown in the table, for both 3-axis and 5-axis problem instances. Lower quality plans are generated in less than 5 CPU-time seconds.

From analysing the results presented in Table 2, it is also noticeable that optimising for a single metric is often at the expense of the other. In addition, when optimising the energy the total consumption is 0. This is because the planner is able to identify a plan where the machine is `off`. However, as the components are off and the machine's structure begins to cool, the deformation is higher, resulting in extremely high machine error values. Intuitively this may seem of no benefit to a manufacture; however, it is worth highlighting that not all manufacturers are operating to such tight tolerances and may be content with such error values. Typically, for the machine tools considered in our case-study, an accuracy of at most 30 µm is required.

Table 2 also provides *de facto* standard "strategies" which

Instance	Standard En(kW)	Standard Er(µm)	Metric: Error En(kW)	Metric: Error Er(µm)	Metric: Energy En(kW)	Metric: Energy Er(µm)	Metric: $\overline{Er + En}$ En(kW)	Metric: $\overline{Er + En}$ Er(µm)
3-30	6.6	4.0	9.0	0.0	0.0	5.5	1.8	0.5
3-60	7.5	4.6	14.9	0.1	0.0	11.0	3.6	1.1
3-120	9.2	5.7	31.6	0.6	0.0	22.0	7.1	2.1
5-30	8.4	5.9	11.5	0.1	0.0	6.5	2.1	0.6
5-60	9.6	6.7	18.7	0.2	0.0	12.0	3.9	1.2
5-120	11.9	8.3	34.3	1.0	0.0	22.0	6.9	2.1

Table 2: Energy and error values for the considered scenarios when exploiting the standard strategy, and the plans generated using the proposed approach while optimising for the three different metrics. Problem instances are in the format of a the number of axes and the interval duration. E.g 3-30 represents a 3 axis machine with a 30 minute interval.

are constructed based on current industry standard practises. Note that such strategies consist of having all the machine's electrical subsystems in `idle`, and then running the warm-cycle if needed. Given the considered time window, an intensive warm-up cycle –constituting a 10 minute period– is always performed, in order to keep all the subsystems at the required temperature. Remarkably, the warm-up cycle effect is different, according to the length of the interval: the longer the interval, the lower the accuracy at the start of the subsequent manufacturing operation.

When comparing the results from the automatically constructed plans with the de facto standard strategies, it is easy to see that the improvement in energy and error given by using the multi-object plan is on average of 54% and 79%, respectively. For example, consider the 5-30 instance, the generated plans will reduce the energy consumption and increase the accuracy of 66% and 90%, respectively, with regards to the current industry practise. Clearly, a dramatic reduction in energy consumption can be achieved when optimising for the energy metric only, and a similar figure can be observed when focusing on accuracy. However, the multi-objective metric results in plans that show a very valuable compromise. It should be noted that the machine error meets the threshold requirements for the considered machine tools in all automatically generated plans –and should be emphasised that for manufacturing purposes, it is pivotal to keep accuracy within the given threshold: further accuracy reduction is appreciated, but not critical. The comparison results demonstrate the suitability and potential impact of utilising automated planning for considering interval activity. Results presented in Table 2 clearly indicate that the proposed planning-based approach is useful for stabilising the machine error whilst reducing estimated energy consumption, and shows a significant improvement over the current standard strategies exploited in the manufacturing field.

Figure 7 demonstrates the `total-energy` and `total-error` fluents throughout the produced plans for the 3-30 problem instance when optimising for the three different metrics. The provided graphs show the cumulative evolution of error and energy of the 3 components of the machine. The results clearly demonstrate how the values are changing depending on the scheduled action. The first graph (left) illustrates that when optimising for energy consumption, the planner selects to switch all the electrical components off resulting in a continuous increase of machine error. Conversely, when optimising for error (middle), it can be seen that the planner selects actions which have a minimal impact on the error at the expense of energy. As already observed in Table 2, the multi-object approach (right) is interesting as it demonstrates a combination whereby a series of actions is selected to keep both metrics at a reduced value.

```
0:(IDLE X_SERVO X)          [1.0000]
0:(OFF Z_SERVO Z)           [1.0000]
1:(IDLE X_SERVO X)          [1.0000]
1:(OFF Z_SERVO Z)           [1.0000]
2:(IDLE X_SERVO X)          [1.0000]
2:(OFF Z_SERVO Z)           [1.0000]
3:(IDLE X_SERVO X)          [1.0000]
3:(HIGH Z_SERVO Z)          [1.0000]
4:(NORMAL X_SERVO X)        [1.0000]
4:(OFF Z_SERVO Z)           [1.0000]
```

Figure 6: Plan excerpt generated by considering two components involved in the 3-30 instance. The plan is optimised for the trade-off between energy consumption and overall error. For the sake of conciseness, only the first four minutes are shown, for two of the three subcomponents.

Figure 6 provides a plan excerpt where the metric is to minimise the arithmetic mean of both the total error and energy. This excerpt is interesting as it illustrates how the actions can be scheduled to maintain a desired level of heat generation while reducing energy consumption. It can be noticed how initially the electrical subsystem `x_servo` is turned `idle` to save energy, and as there is enough residual heat within the machine's structure, stability of the machine is maintained. It is then noticeable how gradually the electrical subsystem is turned back on and left to `normal`,

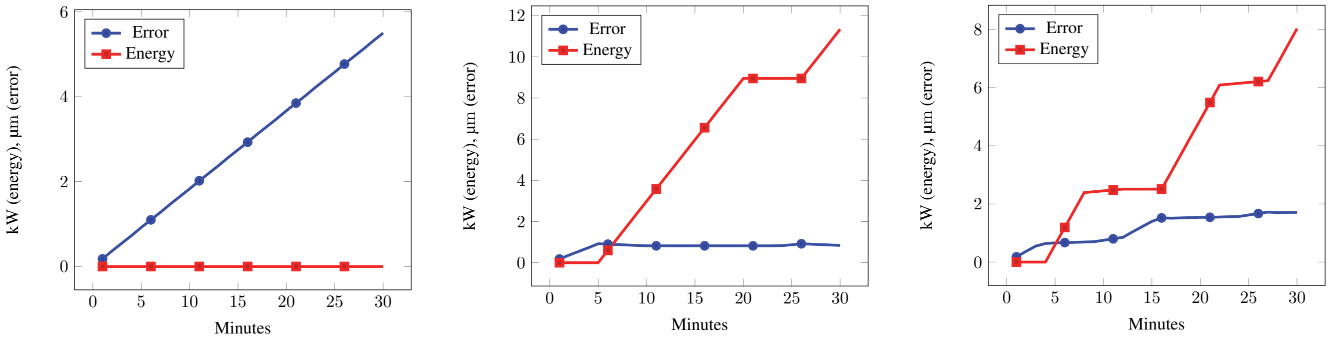

Figure 7: (Coloured) Evolution of error (µm) and Energy consumption (kW) when executing the plans generated by the proposed approach for the 3-30 instance, optimised according to the introduced metrics: minimising energy consumption (left), minimising error (centre), and minimising both aspects at the same time (right).

allowing heat to be generated. The plan then demonstrates that the planner decided to exploit a different approach for maintaining the temperature of the `z_servo` subsystem: the `spindle` is initially set at a `off` level as is cooling, until the third minute where it is run at a `high` level, and then turn `off` again. Optimising the same problem instance for energy would result in a plan where the machine is left `off`, and when optimising for error, result in a plan where the machine's motors are kept `normal` to maintain a minimal level of error.

Is it possible to assess the financial impact of exploiting the generated plans, instead of the de facto standard, for a manufacturing company? Assuming that the machine will be operating 24 hours a day, it is reasonable to state that there would be six 3-30 intervals, six 3-60 intervals, and two 3-120 intervals during shift changes. This corresponds to the average usage figure of a typical C-frame three-axis machine tool. These values are influenced from research into optimising job throughput on machine tools (Lee and Kim 2012; Liu et al. 2014b). Using automatically constructed interval plans would result in a total annual energy saving of 340KWh. If we assume the energy price to be 0.15£, the annual saving would be £52. If we also assume that the manufacturer is of medium size and has 50 machine tools, the total energy reduction would be £2,600 This is significant as not only is the energy usage and cost reduced, the improvement in machine accuracy will also minimise the probability of manufacturing defective parts, resulting in additional cost reduction.

Conclusion

This paper presented the use of automated planning to plan for machine tool activity between manufacturing operations. This is a novel application with a great potential to reduce energy consumption in manufacturing. We introduced a PDDL 2.2 model of the task, that allows to consider at the same time accuracy and energy consumption. Three different metrics have been introduced, that can be used for generating plans optimised for accuracy, energy consumption, or both at the same time.

In order to test the proposed approach, we designed a case-study using real-world data of typical three-axis and five-axis machine tools. Plans have been generated for three different intervals length, which corresponds to common lengths in manufacturing companies. We then compared the quality of the plans generated by our approach with the de facto standard. The results of the comparison shows that the proposed approach can effectively be used for planning the activities between planning operations, and the different introduced metrics can result in very different plans. In general, the multi-objective metric –that considers at the same time accuracy and energy consumption– leads to valuable plans, with an average improvement of 54% (79%) in terms of accuracy (energy consumption).

We see several avenues for future work. We plan to investigate the use of the introduced planning model on different manufacturing machines. We are also interested in assessing the impact of the granularity and temperature-accuracy discretisation on plans' quality and solvability. We envisage an analysis of the performance of different planning engines: recent International Planning Competitions have fostered the development of efficient domain-independent planners. Finally, further work should be spent to remove reliance on the machine operator for providing information (tolerance requirements, etc.) and for converting the PDDL output into code which can be directly executed on the machine tool's controller. This would increase the usability of the presented technique and increase its impact in the manufacturing community.

Availability. The full domain model and problem instances are available at: http://selene.hud.ac.uk/scomsp2/interval.zip It should be noted that in the provided version, some details about the case study machine tools have been modified due to the commercial sensitivity of the corresponding information.

References

Abele, E.; Sielaff, T.; Schiffler, A.; and Rothenbücher, S. 2011. Analyzing energy consumption of machine tool spindle units and identification of potential for improvements of efficiency. In *Glocalized Solutions for Sustainability in Manufacturing*. Springer. 280–285.

Creighton, E.; Honegger, A.; Tulsian, A.; and Mukhopadhyay, D. 2010. Analysis of thermal errors in a high-speed micro-milling spindle. *International Journal of Machine Tools and Manufacture* 50(4):386–393.

Diaz, N.; Helu, M.; Jayanathan, S.; Chen, Y.; Horvath, A.; and Dornfeld, D. 2010. Environmental analysis of milling machine tool use in various manufacturing environments. In *Sustainable systems and technology (ISSST), 2010 IEEE international symposium on*, 1–6. IEEE.

Diaz, N.; Redelsheimer, E.; and Dornfeld, D. 2011. Energy consumption characterization and reduction strategies for milling machine tool use. In *Glocalized Solutions for Sustainability in Manufacturing*. Springer. 263–267.

Draganescu, F.; Gheorghe, M.; and Doicin, C. 2003. Models of machine tool efficiency and specific consumed energy. *Journal of Materials Processing Technology* 141(1):9–15.

Eberspächer, P.; Lechler, A.; and Verl, A. 2015. Control-integrated consumption graph-based optimisation method for energy reduction of machine tools with automated parameter optimisation. *International Journal of Computer Integrated Manufacturing* 1–10.

Edelkamp, S., and Hoffmann, J. 2004. PDDL2.2: The Language for the Classical Part of the 4th International Planning Competition. Technical Report 195, Albert-Ludwigs-Universitat Freiburg, Institut fur Informatik.

Fang, K.; Uhan, N. A.; Zhao, F.; and Sutherland, J. W. 2013. Flow shop scheduling with peak power consumption constraints. *Annals of Operations Research* 206(1):115–145.

Fox, M., and Long, D. 2003. PDDL2.1: an extension to PDDL for expressing temporal planning domains. *Journal of Artificial Intelligence Research* 20:61–124.

Frigerio, N.; Matta, A.; Ferrero, L.; and Rusinà, F. 2013. Modeling energy states in machine tools: an automata based approach. In *Re-Engineering Manufacturing for Sustainability*. Springer. 203–208.

Gerevini, A.; Saetti, A.; and Serina, I. 2006. An approach to temporal planning and scheduling in domains with predictable exogenous events. *Journal of Artificial Intelligence Research* 25:187–231.

Ghallab, M.; Nau, D.; and Traverso, P. 2004. *Automated Planning Theory and Practice*. Elsevier Science.

Heidenhain. 2010. Aspects of energy efficiency in machine tools. Technical report.

Hesselbach, J., and Herrmann, C. 2011. Globalized solutions for sustainability in manufacturing. In *Proceedings of the 18th CIRP International Conference on Life Cycle Engineering*, 2071–1050.

Jeon, H. W., and Prabhu, V. V. 2013. *Advances in Production Management Systems. Sustainable Production and Service Supply Chains: IFIP WG 5.7 International Conference*. Springer Berlin Heidelberg. chapter Modeling Energy Performance of Manufacturing Systems Using Gi/M/1 Queues, 127–134.

Lee, J.-Y., and Kim, Y.-D. 2012. Minimizing the number of tardy jobs in a single-machine scheduling problem with periodic maintenance. *Computers & Operations Research* 39(9):2196–2205.

Li, W., and Kara, S. 2011. An empirical model for predicting energy consumption of manufacturing processes: a case of turning process. *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 225(9):1636–1646.

Li, L.; Yan, J.; and Xing, Z. 2013. Energy requirements evaluation of milling machines based on thermal equilibrium and empirical modelling. *Journal of cleaner production* 52:113–121.

Liu, Y.; Dong, H.; Lohse, N.; Petrovic, S.; and Gindy, N. 2014a. An investigation into minimising total energy consumption and total weighted tardiness in job shops. *Journal of Cleaner Production* 65:87 – 96.

Liu, Y.; Dong, H.; Lohse, N.; Petrovic, S.; and Gindy, N. 2014b. An investigation into minimising total energy consumption and total weighted tardiness in job shops. *Journal of Cleaner Production* 65:87–96.

Liu, Y.; Dong, H.; Lohse, N.; and Petrovic, S. 2015. Reducing environmental impact of production during a rolling blackout policy–a multi-objective schedule optimisation approach. *Journal of Cleaner Production* 102:418–427.

Mayr, J.; Jedrzejewski, J.; Uhlmann, E.; Donmez, M. A.; Knapp, W.; Härtig, F.; Wendt, K.; Moriwaki, T.; Shore, P.; Schmitt, R.; et al. 2012. Thermal issues in machine tools. *CIRP Annals-Manufacturing Technology* 61(2):771–791.

Mian, N. S.; Fletcher, S.; Longstaff, A. P.; and Myers, A. 2011. Efficient thermal error prediction in a machine tool using finite element analysis. *Measurement Science and Technology* 22(8):085107.

Mian, N. S.; Fletcher, S.; Longstaff, A. P.; and Myers, A. 2013. Efficient estimation by FEA of machine tool distortion due to environmental temperature perturbations. *Precision engineering* 37(2):372–379.

Parkinson, S.; Longstaff, A.; Crampton, A.; and Gregory, P. 2012a. The application of automated planning to machine tool calibration. In *Proceedings of the twenty-second international conference on automated planning and scheduling (ICAPS)*.

Parkinson, S.; Longstaff, A. P.; Fletcher, S.; Crampton, A.; and Gregory, P. 2012b. Automatic planning for machine tool calibration: A case study. *Expert Systems with Applications* 39(13):11367–11377.

Peng, T., and Xu, X. 2014. Energy-efficient machining systems: a critical review. *The International Journal of Advanced Manufacturing Technology* 72(9-12):1389–1406.

Vijayaraghavan, A., and Dornfeld, D. 2010. Automated energy monitoring of machine tools. *CIRP Annals-Manufacturing Technology* 59(1):21–24.

Wang, S.; Lu, X.; Li, X.; and Li, W. 2015. A systematic approach of process planning and scheduling optimization for sustainable machining. *Journal of Cleaner Production* 87:914–929.