A Case Study in Timed Refinement: A Central Heater

Brendan Mahony
Ian Hayes
University of Queensland
St. Lucia, 4072
Australia

January, 1991

Abstract

The refinement calculus is proving a useful tool for the specification and refinement of sequential processes. In this paper we contend that it is also useful in the timed case. This paper displays the use of the refinement calculus for a small embedded system.

1 Introduction

The refinement calculus [7, 6, 1, 8] extends Dijkstra's weakest precondition program semantics to the realm of specifications. This provides a uniform notation for the entire refinement process. We assume that the reader is familiar with the notion of weakest precondition semantics for programs. Some familiarity with the refinement calculus and real functions will be useful.

The Z notation [2, 10] is used for expressing predicates on states, but a knowledge of Z is not essential to the reading of this paper. Those not familiar with Z may view the schema boxes as a convenient way of naming and expressing predicates. The schema consists of two sections, the first serving merely to introduce the variables discused in the predicate, appearing in the second section.

For sequential processes the refinement calculus sees programs in terms of the state of a system prior to execution and the state after execution of the program. The suitability of this approach relies on three assumptions about the nature of sequential processes.

1 Sequential processes terminate.

2 Sequential processes have sole access to program variables so there can be no interference from other processes.

3 Sequential processes have no timing obligations other than termination.

We are interested in investigating the usefulness of the refinement calculus where these assumptions break down. Our special interest lies in the realm of non-terminating real-time processes, but the process view we describe is equally applicable wherever these assumptions do not apply.

Specification of processes in terms of their initial and final states is not suitable for discussing non-sequential processes, since there may be no final state or intermediate states may be important. We propose two changes to the way in which predicate transformers are understood.

Firstly, the state of a system should be its behaviour over all time. It is essential whenever any of the above assumptions fail that the initial and final state paradigm is replaced by a state history paradigm. Since our interest is in non-terminating processes and we are also interested in real-time, we adopt a history model in which the time domain is represented by the positive real numbers.

Secondly we observe that the initial/final state paradigm is in fact a degenerate case of the history paradigm. A history in which only two state observations are made. Viewed in this light we can see that the the refinement calculus is in fact dealing with the relationship between the structure induced on the system by the process (the post-condition involving initial and final states) and the structure inherent in the system (the precondition involving only the initial states). Thus the process is actually viewed in terms of how it transforms the behaviour of a system that it acts upon. We contend that the natural generalisation of the sequential refinement calculus is based on mapping the desired behaviour of system plus process, the *effect* of the process, to the known behaviour of the system in isolation, the *assumptions* the process may make about the system. We generalise the weakest precondition semantics for sequential processes to a *weakest assumption* semantics for non-sequential processes.

A full justification for the above decisions and a derivation of the resulting non-sequential refinement calculus may be found in [3, 4]. Here we give give only informal motivations for and explanations of the notation we use. The purpose here is to demonstrate the practical usefulness of the techniques.

2 Central Heating

We consider a household central heating system. The intention of the central heater is to maintain the temperature of a house around a certain minimum temperature

$$tmin : TEMPERATURE.$$

We represent the domain of temperatures, $TEMPERATURE$, by the positive real numbers, $\mathbb{R}_+$, being the temperature measured in degree Kelvin.

2.1 Functionality

The house may be modelled as a simple thermodynamic system attached to a heat sink (the cold cold snow) and a heat source (the heater). The state of the house may then be represented by its temperature, and the rates at which heat is going out and coming into the house. The history of the temperature and heat loss and heat gain may be

represented as functions over all time. To avoid pathological, and unrealistic, cases we restrict consideration to topologically continuous functions. A function is topologically continuous if it respects open sets. Formally, the preimage of an open set in the range must be open in the domain. We write $X \twoheadrightarrow Y$ for the total continuous functions from X to Y and $X \nrightarrow Y$ for the partial ones. Strictly speaking, these definitions should be made with respect to particular topologies, but we will assume that each domain has a default topology and omit explicit reference to them. For instance, in the standard topology for the reals, $\mathcal{T}_{\mathbb{R}}$, the open sets are constructed from arbitrary unions of open intervals. An open interval $(x \,..\, y)$ is the set of points $\{z : \mathbb{R} \mid x < z < y\}$. $\mathfrak{I}_{\mathbb{R}}$ is the set of all open intervals.

The temperature of the house at any time depends on the amount of heat that is flowing into and out of the house.

House

$$\theta : TIME \twoheadrightarrow TEMPERATURE$$
$$Hin, Hout : TIME \twoheadrightarrow TEMPERATURE \text{ per } TIME$$

$$\forall t_1, t_2 : TIME \bullet$$
$$\theta(t_2) = \theta(t_1) + \cap_{t_1}^{t_2} Hin(s) - Hout(s) \bullet ds$$

The heat flows are modelled as temperature flows so that the specific heat of the house need not be considered explicitly. The time domain, $TIME$, and the rates of temperature change, $TEMPERATURE$ per $TIME$, are represented by positive reals.

The purpose of the central heater is to regulate the temperature of the house. The temperature must be prevented from falling below $tmin$ for periods longer than

$$\delta : TIME.$$

The standard topology for the reals, $\mathcal{T}_{\mathbb{R}}$, has the useful property that any open set may be covered by a countable[1] disjoint union of open intervals [9]. It is thus possible to uniquely define a function,

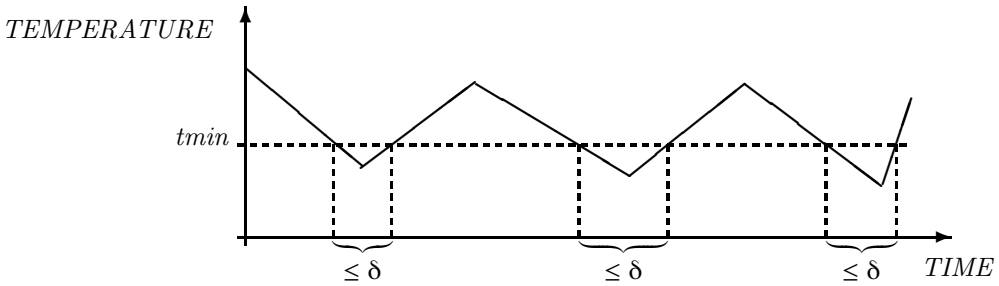
$$OpenCover : \mathcal{T}_{\mathbb{R}} \to \mathbb{P}_{\omega}\, \mathfrak{I}_{\mathbb{R}}$$

$$\forall O : \mathcal{T}_{\mathbb{R}} \bullet$$
$$\bigcup \text{cov}(O) = O$$
$$\forall \Delta, \Delta' : \text{cov}(O) \bullet \Delta \neq \Delta' \Rightarrow \Delta \cap \Delta' = \{\}$$

which decomposes every open set into a countable set of maximal open intervals. In essence any open set of real numbers may be viewed as a (possibly infinite) sequence of intervals.

Using $OpenCover$ we are able to concisely express the requirements for the heater in terms of the length of the time intervals for which the temperature is too low. The times for which the temperature is too low are in the pre-image of the temperature region $(0 \,..\, tmin)$ under the temperature function θ, i.e. $\theta^{-1}((0 \,..\, tmin))$. Using the continuity of θ we know this pre-image is an open set, so $OpenCover$ may be used to find the maximal intervals of time for which the temperature falls below $tmin$.

[1] We extend the Z notation, using $\mathbb{P}_{\omega}\, X$ to represent the countable subsets of X.



Each such interval, Δ, must have duration, $|\Delta|$, less than δ.

WarmHouse

$House$

$\forall \Delta : \mathrm{cov}(\theta^{-1}(\!(0 \mathinner{..} tmin)\!)) \bullet |\Delta| \leq \delta$

This ability to decompose time regions into sets of time intervals adds great expressive power to our specifications.

The question now arises of under what condition can we reasonably expect a central heater to be able to do this. Whether it chooses to keep the house warm by restricting the outflow of heat (improved insulation) or by increasing the inflow of heat, the efforts of the heater will be finite in scope. We cannot expect it to be able to compensate for arbitrary rates of heat loss. Therefore we require that there be a bound

$$Insul : TEMPERATURE \text{ per } TIME$$

on the rate at which heat flows out of the house.

For similar reasons to above we cannot expect the heater to heat the house within time δ from an arbitrary starting temperature. Consequently we will assume that the temperature begins in the required range. If desired a terminating process can be used to heat the room up, before turning it over to the standard central heating process.

Consequently we will allow the central heater to assume the house satisfies

InsulatedHouse

$House$

$\theta(0) \geq tmin$

$\forall t : TIME \bullet Hout(t) \leq Insul$

Our specification is then that the heater must cause the house to satisfy $WarmHouse$ as long as the house is known to be an $InsulatedHouse$. We write the specification

$$CentralHeater == [InsulatedHouse, WarmHouse].$$

This notation mimics the specification statements of Morgan [6]. We also adopt the term specification statement for such constructs. In [6] the first predicate is called the

precondition and the second predicate the postcondition. The terms 'precondition' and 'postcondition' are not suitable in the context of non-terminating processes. Instead, we adopt the terms *assumption* and *effect* respectively.

The specification $CentralHeater$ is interpreted as: if the environment in isolation satisfies the first predicate (assumption $CentralHeater$), applying the $CentralHeater$ process to it will cause it to satisfy the further conditions expressed in the second predicate (effect $CentralHeater$).

2.2 The Heater

The central heating system will be implemented using a heating element, with minimum output

$$Output : TEMPERATURE \text{ per } TIME$$

that may be turned on and off to control the house's temperature.

Heater

$$Element : TIME \twoheadrightarrow \{on, off\}$$

The $Element$ function cannot be both continuous and total. Continuous functions into discrete domains must be step functions, with periods of non-definedness between steps. This is a good representation for digital quantities that remain constant until acted on by an event, during the action of which they are undefined.

The heater must act as the heat source of the house.

InstalledHeater

$House$
$Heater$

---

true

$$\forall t : \text{dom } Element \bullet Element(t) = on \Rightarrow Hin(t) \geq Output$$

Here the specification statement appears in a vertical format which is more convenient when the process assumptions are trivial. The syntactic format is inspired by the Z schema, with the first section declaring the process' state variables and their types, the second displaying the assumptions predicate, and the third the effects predicate. The invariants expressed in $House$ and $Heater$ become invariants that are preserved by the process.

When the temperature falls below $tmin$ the heater control reacts by turning on the heater until the temperature has again risen above $tmin$. The heater is turned on within

$$r : TIME$$

of the temperature becoming too low.

ControlHeater

$House$
$Heater$

true

$\forall \Delta : \mathrm{cov}(\theta^{-1}(\!|\{0 \mathinner{..} tmin\}|\!)) \bullet$
$\quad (\!|\inf \Delta + r \mathinner{..} \sup \Delta|\!) \subseteq Element^{-1}(\!|\{on\}|\!)$

The heater will restore the temperature in time as long as

$$\delta.Insul \leq (\delta - r).Output.$$

So we add the assumption

$$Output \geq \frac{\delta}{\delta - r}.Insul$$

These two processes will implement $CentralHeater$ if they are run in parallel. The parallel operator for specification statements [5] is defined

$$[\mathrm{assumption}_1, \mathrm{effect}_1] \parallel [\mathrm{assumption}_2, \mathrm{effect}_2] ==$$
$$\quad [\mathrm{assumption}_1 \wedge \mathrm{assumption}_2, \mathrm{effect}_1 \wedge \mathrm{effect}_2]$$

The design for the central heating system is then

$$CentralHeater_1 == (InstalledHeater \parallel ControlHeater).$$

Theorem 2.1

$$CentralHeater \sqsubseteq CentralHeater_1$$

Proof: The definition of refinement for specification statements [6, 5] means we must show that

$$\mathrm{assumption}\ CentralHeater \Rightarrow$$
$$\quad \mathrm{assumption}\ CentralHeater_1 \wedge$$
$$\quad \mathrm{effect}\ CentralHeater_1 \Rightarrow \mathrm{effect}\ CentralHeater$$

Since assumption $CentralHeater_1$ is **true** this reduces to

$$\mathrm{assumption}\ CentralHeater \wedge \mathrm{effect}\ CentralHeater_1 \Rightarrow$$
$$\quad \mathrm{effect}\ CentralHeater$$

Writing this out in full, we find that we must show that

$$\begin{array}{ll}
\forall\, t : TIME \bullet & (InsulatedHouse) \\
\quad Hout(t) \leq Insul & \\
\quad \theta(0) \geq tmin & \\
\forall\, t : TIME \bullet & (\text{effect } InstalledHeater) \\
\quad Element(t) = on \Rightarrow Hin \geq Output & \\
\forall\, \Delta : \mathrm{cov}(\theta^{-1}(\!|(0 \mathinner{..} tmin)|\!)) \bullet & (\text{effect } ControlHeater) \\
\quad (\!|\inf \Delta + r \mathinner{..} \sup \Delta|\!) \subseteq Element^{-1}(\!|\{on\}|\!) & \\
\Rightarrow & \\
\forall\, \Delta : \mathrm{cov}(\theta^{-1}(\!|(0 \mathinner{..} tmin)|\!)) \bullet & \\
\quad |\Delta| \leq \delta & (WarmHouse)
\end{array}$$

Case (Δ finite)

Suppose $\Delta \in \mathrm{cov}(\theta^{-1}(\!|(0, tmin)|\!))$ is finite in length, then as θ is continuous and $\theta(0) \geq tmin$

$$\theta(\inf \Delta) = \theta(\sup \Delta) = tmin,$$

so from the invariant on $House$,

$$0 = \cap_{\inf \Delta}^{\sup \Delta} Hin(s) - Hout(s) \bullet ds.$$

Using the hypotheses about the heater and the maximum heat loss this gives

$$\begin{array}{ll}
\cap_{\inf \Delta}^{\sup \Delta} Insul \bullet ds & \\
\geq \cap_{\inf \Delta}^{\sup \Delta} Hout(s) \bullet ds & \text{(insulation hypothesis)} \\
= \cap_{\inf \Delta}^{\sup \Delta} Hin(s) \bullet ds & \\
\geq \cap_{\inf \Delta + r}^{\sup \Delta} Output \bullet ds & \text{(element hypothesis)} \\
\geq \cap_{\inf \Delta + r}^{\sup \Delta} \dfrac{\delta}{\delta - r} Insul \bullet ds &
\end{array}$$

Thus

$$|\Delta| \,.Insul \geq (|\Delta| - r).\frac{\delta}{\delta - r}.Insul$$

$$\text{i.e. } (\delta - r).|\Delta| \geq (|\Delta| - r).\delta$$

$$\text{i.e. } \delta.|\Delta| - r.|\Delta| \geq |\Delta|.\delta - r.\delta$$

$$\text{i.e. } \delta \geq |\Delta|$$

as required.

Case (Δ infinite)

A similar proof will show that Δ must be finite.

□

The specification $InstalledHeater$ can be implemented by the simple expedient of placing the heating element inside the house. All control logic may therefore be restricted to the process $ControlHeater$, and we can restrict further consideration to that process.

2.3 Avoiding Rapid Oscillations

It would be undesirable if the heater were to turn off immediately the temperature climbed to $tmin$, since this would mean the temperature was usually below $tmin$. To avoid this we strengthen $ControlHeater$ to ensure that the heater remains on until the temperature reaches $tmax : TEMPERATURE$, where $tmax > tmin$.

$ControlHeater_1$

$House$
$Heater$

---

true

$$\forall \Delta : \mathrm{cov}(\theta^{-1}(\!|\{0 \mathinner{..} tmin\}|\!)) \bullet$$
$$\quad \exists \Delta' : \mathrm{cov}(\theta^{-1}(\!|\{0 \mathinner{..} tmax\}|\!)) \bullet$$
$$\quad\quad \Delta \subseteq \Delta' \land$$
$$\quad\quad \{\inf \Delta + r \mathinner{..} \sup \Delta'\} \subseteq Element^{-1}(\!|\{on\}|\!)$$

Since $ControlHeater_1$ is gained by strengthening of the effect of $ControlHeater$ it is clear that it is a refinement thereof.

Theorem 2.2

$$ControlHeater \sqsubseteq ControlHeater_1$$

The parallel operator is monotonic with respect to refinement [5]. This means that refining one of the sub-specifications yields a refinement of the combined specification.

Corollary 2.3

$$CentralHeater \sqsubseteq (InstalledHeater \| ControlHeater_1)$$

3 Conclusion

We have demonstrated that non-terminating real-time processes may be developed using rigorous specification and refinement techniques. The techniques in this paper decompose specifications into assumptions a process may make and the effect required of it. In this way non-terminating processes can also be associated with a predicate transformer semantics. We have demonstrated that notions from the sequential refinement calculus, such as refinment (Theorems 2.1,2.2) and monotonic specification combinators (Corollary 2.3) are applicable to real-time, non-terminating and parallel processes. Whilst our case study lies in the realm of embedded, real-time systems, the techniques demonstrated are applicable to arbitrary non-sequential processes.

Further the techniques we have demonstrated are a generalisation of the refinement calculus for sequential processes. Sequential processes may be considered as a special

case in the assumption/effect view of processes. The predicate transformer calculus derived in [1] remains the theoretical grounding for our non-sequential refinement calculus.

Acknowledgements

This paper is the result of work begun while I was visiting the Programming Research Group, Oxford University, under funding from the Australian Commonwealth Postgraduate Research Award Scheme. I would like to thank the Programming Research Group for their hospitality.

I am in debt to Carroll Morgan for his inspiring work.

Lastly I would like to acknowledge the contribution of the English weather in motivating this case study.

References

[1] R. J. R. Back and J. von Wright. Refinement calculus, part I: Sequential nondeterministic programs. In J. W. de Bakker, W. P. de Roever, and G. Rozenberg, editors, *Stepwise Refinement of Distributed Systems: Models, Formalism, Correctness*, LNCS 430, pages 42–66. Springer Verlag, 1990.

[2] I. J. Hayes, editor. *Specification Case Studies*. Prentice Hall International, 1987.

[3] B. P. Mahony. The refinement calculus and process construction. Submitted for publication to Acta Informatica, 1990.

[4] B. P. Mahony. *The Specification and Refinement of Real-time Processes*. PhD thesis, University of Queensland, 1991.

[5] B. P. Mahony and I. J. Hayes. Generalising the specification statement to real-time. Working Paper, 1990.

[6] C. C. Morgan, K. A. Robinson, and P. Gardiner. On the refinement calculus. Technical Monograph PRG-70, Oxford University Programming Research Laboratory, 1988.

[7] C.C. Morgan. The specification statement. *ACM Trans. Prog. Lang. and Sys.*, 10(3), July 1988. Reprinted in [6, pp. 7–30].

[8] J. M. Morris. A theoretical basis for stepwise refinement and the programming calculus. *Science of Computer Programming*, 9:287–306, 1987.

[9] H. L. Royden. *Real Analysis*. Macmillan Publishing Co., Inc., second edition, 1968.

[10] J. M. Spivey. *The Z Notation: A Reference Manual*. Prentice Hall International, 1989.