On the Algebraic Properties of Timeliness

AN IOG TECHNICAL REPORT

Seyed Hossein HAERI
IOG, Singapore
University of Bergen
Norway
hossein.haeri@iohk.io

Peter W. THOMPSON
PNSol, UK
Peter.Thompson@pnsol.com

Peter VAN ROY
Université catholique de Louvain, Belgium
pvr@info.ucl.ac.be

Magne HAVERAAEN
University of Bergen
Norway
Magne.Haveraaen@uib.no

Neil J. DAVIES
PNSol, UK
Neil.Davies@pnsol.com

Mikhail BARASH
University of Bergen
Norway
mikhail.barash@uib.no

James CHAPMAN
IOG, Singapore
james.chapman]@iohk.io

Abstract

Designing distributed systems to have predictable performance under high load is difficult because of resource exhaustion, nonlinearity, and stochastic behaviour. The ΔQ Systems Development paradigm (ΔQSD), developed by PNSol, addresses these difficulties by modelling systems using outcome expressions, which are combinations of basic operations whose behaviour is defined stochastically. This paper defines and proves algebraic properties of these operations when the relevant resource is time (i.e., latency). This is part of an ongoing project to disseminate and build tool support for ΔQSD; for tooling, the ability to simplify outcome expressions without changing the result is essential for managing computational complexity. We show how the ΔQSD operators give rise to different algebraic structures. We prove distributivity of the operators when possible. We prove for the first time the validity of a set of folklore equivalences that are in common usage for ΔQSD. An appendix gives a worked example concerning a memory cache.

1 Introduction

Designing distributed systems to have predictable performance under high load is difficult. At high load, resources such as network, memory, storage, or CPU capacity will be exhausted, which has a dramatic effect on performance. Prediction is difficult because the behavior of system components and their interactions are both nonlinear and stochastic. For over 20 years, a small group of people associated with the company PNSol has worked on diagnosing and designing systems to predict and correct performance problems [6]. PNSol has developed the ΔQ Systems Development paradigm (ΔQSD) as part of this work. ΔQSD has been used in areas as diverse as telecommunications [9] [8] [2], WiFi [4], and distributed ledgers [1]. ΔQSD has been applied to many large industrial systems, including BT, Vodafone, Boeing Space and Defence, and IOG (formerly IOHK).

This paper defines and proves algebraic properties of the ΔQSD operators with respect to timeliness, i.e., when the relevant resource is time. This theoretical work is part of an ongoing project to disseminate and build tool support for ΔQSD, to make it available to the wide community of system engineers. We base our work on the ΔQSD formalisation given in [3], which defines outcome expressions and their semantics, and gives a real-world example of ΔQSD taken from the blockchain domain.

Scope of* Δ*QSD. The ΔQSD paradigm models distributed systems to quantify risk and performance trade-offs. ΔQSD can be used both for design and diagnosis:

- *System Diagnosis.* ΔQSD can analyse the observed performance of a system, to pinpoint anomalous behaviours and fix them. Most past use of ΔQSD by PNSol has been to diagnose and correct problems in large industrial systems.
- *System Design.* ΔQSD can estimate performance trade-offs during the design process. At **every** step, performance of the complete system can be estimated by a computation on the partial design. This computation also determines whether or not the system is feasible, i.e., whether it can or cannot meet the requirements. PNSol has used ΔQSD to design the Shelley block diffusion algorithm which is used in the Cardano blockchain [3].

A pedagogical introduction to ΔQSD is available in a HiPEAC 2022 tutorial [11].

Concepts of* Δ*QSD. Here are three main ΔQSD concepts:

- *Outcome*: Any well-defined system behaviour delimited by observable start and end events. For example, the pair of a database request together with its response defines an outcome.
- *Quality Attenuation (ΔQ)*: An *Improper Random Variable* (IRV) that defines the delay between start and end event in an outcome, as well as its probability of failure. In ΔQSD, timeliness analysis is the process of calculating the respective IRVs of outcomes, by which capturing both delay and rate of failure in the same algebraic

term. For example, the delay between a database request and response, given stochastically through its cumulative distribution function.
- *Outcome Expression*: A combination of outcomes and operators, which can model a whole system (Definition 2.1). Like system components that are composed to form larger components, one composes outcomes using the ΔQSD operators to form larger outcomes. That is transliterated using outcome expressions. The outcome expression is used to compute the ΔQ of the whole system, given the ΔQ of the smaller outcomes.

Contributions of the paper. This paper is about algebraic analysis of outcome expressions, when timeliness of outcomes is the concern. The contributions of this paper are:

- We give the first model theory of resource analysis for systems specified using outcome expressions (Section 3). We specialise that model theory in Section 4 using the timeliness analysis recipe that is commonly in use in ΔQSD (Definition 4.2).
- We prove that the set of outcome expressions forms different algebraic structures with the different ΔQSD operators (Theorems 5.2–5.6).
- We prove four distributivity results about the ΔQSD operators (Lemmata 6.1–6.4).
- We refute the formation of richer algebraic structures by the set of outcome expressions and the current ΔQSD operators (Remarks 5.5, 5.7, and 5.10).
- We provide guidelines for studying the existence of other potential algebraic results (Section 7).
- We prove a dozen equivalences that have been used in the past in the practice of ΔQSD (Section 8).

2 ΔQSD Background

This section is a recapitulation of what is already formalised about ΔQSD [3]. The two cornerstones of ΔQSD that we will detail in this section are quality attenuation and outcome expressions. We will go through each in turn.

Contrary to the typical assumption in computer science, in reality outcomes are never perfect; there is always a possibility of error, delay, and failure. ΔQSD captures that aspect of reality using its concept of quality attenuation, denoted by the symbol ΔQ, which is a measure of how much the quality of an outcome is *attenuated* relative to being perfect. ΔQSD formulates performance management requirements in terms of bounds that are to be maintained on ΔQs [5]. For reasons that are explained in [3], ΔQs are IRVs, incorporating both delay (a continuous random variable) and exceptions/failures (discrete events). IRVs are those random variables the total probability of which may not necessarily reach one [10].

Let ΔQ(x) denote the probability that an outcome occurs in a time $t \leq x$. Then, the *intangible mass* of such an IRV is $1 - \lim_{x\to\infty} \Delta Q(x)$. For a ΔQ, the intangible mass encodes the probability of exceptions or failure occurring over the performance of the respective outcome.

IRVs naturally give rise to a partial order, in which the 'smaller' attenuation is the one that delivers a higher probability of completing the outcome in any given time:

$$\Delta Q_1 \leq \Delta Q_2 \equiv \forall x.\ \Delta Q_1(x) \geq \Delta Q_2(x). \tag{1}$$

We distinguish two outcomes: $\top$ for "perfection" and $\bot$ for "unconditional failure." $\top$ and $\bot$ are the top and bottom elements of the above partial order, respectively.

To formally introduce outcome expressions, we first present the notations we use in this paper.

Let $\mathbb{A}, \mathbb{B}, \mathbb{C}, \ldots$ range over sets of values, and let lower case letters, $a, b, c, \ldots$ range over elements of those sets. Subscripts and priming do not change the syntactic category of a symbol. For example, for a set A, we write $A \ni a$ to indicate that $a, a', a'', \ldots, a_1, a_2, \ldots$ all range over A. For scalar variables, priming is to indicate a possibly different element of the same syntactic category, whereas for functions, priming indicates the derivative of the function.

Definition 2.1. Assume a set $\overline{\mathbb{B}} \ni \beta$ of given base variables. The abstract syntax of outcome expressions is:

$$\begin{array}{llll} \mathbb{O} \ni o & ::= & \beta & \\ & \mid & o \bullet\!\!\rightarrow\!\!\bullet\ o' & \text{sequential composition} \\ & \mid & o \overset{m}{\underset{m'}{\leftrightarrows}} o' & \text{probabilistic choice} \\ & \mid & (o \parallel^{\forall} o') & \text{all-to-finish (a.k.a. last-to-finish)} \\ & \mid & (o \parallel^{\exists} o') & \text{any-to-finish (a.k.a. first-to-finish)} \end{array}$$

Note that m and m' are the weights according to which the choice goes to the left or the right respectively.

3 The Proof System

This section lays a foundation for resource analysis that is particularly geared towards algebraic studies. We begin by stating the compositionality requirements we expect the analysis methods to establish. We give a formal definition of when two formulae related to a common resource can be considered equivalent. Then, we give notation for specifying the formation of an algebraic structure.

Fix a set of resources $\mathbb{H}$ ranged over by ρ. For every $\rho \in \mathbb{H}$, we assume a set $R(\rho)$ over which ρ values range. For example, time values range over $\mathbb{R}$ and number of CPU cycles ranges over $\mathbb{N}$. Call a function

$$m_\rho : \overline{\mathbb{B}} \to R(\rho) \tag{2}$$

a ρ-measuring of $\overline{\mathbb{B}}$. When appropriate, we assume that each ρ is equipped with a unique ρ-measuring (of $\overline{\mathbb{B}}$). As such, we will call that function *the* ρ-measuring (of $\overline{\mathbb{B}}$).

Fix $\mathbb{P} = \{\bullet\!\!\rightarrow\!\!\bullet, \leftrightarrows, \parallel^{\forall}, \parallel^{\exists}\}$. Given a $P \subseteq \mathbb{P}$, take the set $\mathbf{F}|_P$ of P-formulae of $\mathbb{O}$ to be the smallest superset of $\mathbb{O}$ that is closed under P.

Definition 3.1. A P-equation $e \in \mathbb{E}|_P$ is of the form $f_l = f_r$, where $f_l, f_r \in \mathbf{F}|_P$, for some $P \subseteq \mathbb{P}$.

Definition 3.2. Given a binary relation $\mathcal{R}$ on $\mathbf{F}|_P$, an $\mathcal{R}$-instance r is of the form $f_l \ \mathcal{R} \ f_r$, for some $P \subseteq \mathbb{P}$.

A structural extension to a ρ-measuring is a function

$$\hat{\cdot}_\rho : (\overline{\mathbb{B}} \to R(\rho)) \to \mathbb{O} \to R(\rho) \quad (3)$$

which, given a ρ-measuring, provides instructions for how to compositionally measure outcome expressions. Given our uniqueness assumption about the ρ-measuring, we also assume, when appropriate, that each ρ is equipped with a unique extension to (its unique) ρ-measuring. From now on, we will equate the ρ-measuring and its extension.

Definition 3.3. Say a P-equation $f_l = f_r$ holds according to the ρ-measuring when $\hat{m}_\rho(f_l) = \hat{m}_\rho(f_r)$.

Definition 3.4. Given a binary relation $\mathcal{R}$ on $\mathbf{F}|_P$ and its image $m_\rho(\mathcal{R})$ on $R(\rho)$, say an $\mathcal{R}$-instance $f_l \ \mathcal{R} \ f_r$ holds according to the ρ-measuring when $\hat{m}_\rho(f_l) \ m(\mathcal{R}) \ \hat{m}_\rho(f_r)$.

Observing a resource ρ, when a P-equation e holds according to the ρ-measuring, write $\odot\!\odot \ \rho \vDash e$. (Take "$\odot\!\odot$" to be the two eyes of the observer.) When $\odot\!\odot \ \rho \vDash e$ for all resources $\rho \in \mathbb{H}$, write $\vDash e$, as a generalisation. We extend all the notation introduced in the paragraph to $\mathcal{R}$-instances in the trivial way to write $\odot\!\odot \ \rho \vDash r$.

A ρ-theory on $\mathbb{O}$ is a set of P-equations E_P such that $\forall e \in E_P.\ \odot\!\odot\rho \vDash e$, for some $P \subseteq \mathbb{P}$. In that case, write $\odot\!\odot \ \rho \vDash E_P$. The ρ-algebraic properties of $\mathbb{O}$ are those ρ-theories which demonstrate that $\mathbb{O}$ establishes an algebraic structure.

Definition 3.5. For an algebraic structure s, say $(\mathbb{O}, P)$ is an s when observing ρ iff there exists a ρ-theory of P-equations which establishes s. Denote that as $\odot\!\odot \ \rho \vDash (\mathbb{O}, P) : s$.

4 Time

This section utilises the developments of Section 3 to focus on a particular resource, i.e., time, and its analysis, i.e., timeliness analysis. ΔQSD uses ΔQs for timeliness analysis.

Fix a set $\mathbb{I} \ni \iota$ of all IRVs that are differentiable and the values of which are always greater than or equal to zero. Statistically speaking, every ι can be represented both using its PDF or its CDF. The former is the derivative of the latter. As a result, we choose to liberally switch between the two representations as the need rises.

Fix a countable set of ΔQ variables $\Delta_v \ni \delta_v$. Let $\Delta = \Delta_v \cup \mathbb{I}$, where $\Delta \ni \delta$. When δ is in its CDF representation, write δ' for its derivative, which is also the respective PDF representation.

Definition 4.1. Call a function $\Delta_\circ[\![.]\!] : \overline{\mathbb{B}} \to \Delta$ a basic assignment when $\Delta_\circ[\![\top]\!] = \mathbf{1}$ and $\Delta_\circ[\![\bot]\!] = \mathbf{0}$, where $\mathbf{1}$ and $\mathbf{0}$ are the functions always returning the constants 1 and 0, respectively.

Definition 4.2. Given a basic assignment $\Delta_\circ[\![.]\!] : \overline{\mathbb{B}} \to \Delta$, define $\Delta Q[\![.]\!]_{\Delta_\circ} : \mathbb{O} \to \mathbb{I}$ such that

$$\begin{aligned}
\Delta Q[\![\beta]\!]_{\Delta_\circ} &= \begin{cases} \mathbf{1} & \text{when } \Delta_\circ[\![\beta]\!] \notin \mathbb{I} \\ \Delta_\circ[\![\beta]\!] & \text{otherwise} \end{cases} \\
\Delta Q[\![o \bullet\!\!\rightarrow\!\!\bullet\, o']\!]_{\Delta_\circ} &= \Delta Q[\![o]\!]_{\Delta_\circ} * \Delta Q[\![o']\!]_{\Delta_\circ} \\
\Delta Q[\![o \overset{m}{\underset{m}{\leftrightarrows}} o']\!]_{\Delta_\circ} &= \tfrac{m}{m+m'} \Delta Q[\![o]\!]_{\Delta_\circ} + \tfrac{m'}{m+m'} \Delta Q[\![o']\!]_{\Delta_\circ} \\
\Delta Q[\![\forall(o \parallel o')]\!]_{\Delta_\circ} &= \Delta Q[\![o]\!]_{\Delta_\circ} \times \Delta Q[\![o']\!]_{\Delta_\circ} \\
\Delta Q[\![\exists(o \parallel o')]\!]_{\Delta_\circ} &= \\
&\quad \Delta Q[\![o]\!]_{\Delta_\circ} + \Delta Q[\![o']\!]_{\Delta_\circ} - \Delta Q[\![o]\!]_{\Delta_\circ} \times \Delta Q[\![o']\!]_{\Delta_\circ}
\end{aligned}$$

where $*$ denotes the convolution of two ΔQs. In the above formulae, the random variables are always represented using their CDFs except sequential composition, where the representation is PDFs on both sides. Note that the PDF of $\top$ is the Dirac $\boldsymbol{\delta}$ function. We denote the set of all basic assignments by $\{\Delta_\circ[\![.]\!]\}$.

Remark 4.3. The equalities of Definition 4.2 can be understood from straightforward probabilistic arguments, as follows, where for any outcome X we define *p[X] = probability of event X* and ΔQ_X(t) = *p[X occurs within time t]* (i.e. the CDF of X).

Sequential composition. If we have two outcomes o and o' that occur sequentially, then the probability that $o \bullet\!\!\rightarrow\!\!\bullet\, o'$ takes a time t is the sum of all probabilities that o takes time $\tau < t$ and o' takes time $t - \tau$, which is the definition of convolution.

Probabilistic choice. If we have a probabilistic choice $o \overset{m}{\underset{m}{\leftrightarrows}} o'$ between two outcomes o and o' with relative weights m and n, then the only ways this event can occur within time t are either: o is chosen and occurs within time t; or o' is chosen and o' occurs within time t. Since these are mutually exclusive events we can simply add their probabilities:

$$\begin{aligned}
\Delta Q_{o \overset{m}{\underset{m}{\leftrightarrows}} o'}(t) = \\
& p[o \textit{ occurs within time } t \cap o \textit{ is chosen}] \\
& + p[o' \textit{ occurs within time } t \cap o' \textit{ is chosen}]
\end{aligned}$$

By the definition of conditional probability this can be written as:

$$\begin{aligned}
& p[o \textit{ occurs within time } t \mid o \textit{ is chosen}] \times p[o \textit{ is chosen}] \\
& + p[o' \textit{ occurs within time } t \mid o' \textit{ is chosen}] \times p[o' \textit{ is chosen}]
\end{aligned}$$

The probability that o occurs within time t given that o is chosen is just ΔQ_o(t), and the probabilities of each side being chosen are the relative weights, so this expression is just the weighted sum of the ΔQs as given in Definition 4.2.

Any and all to finish. If we have two independent outcomes o and o', the probability that both occur (i.e. $p[\forall(o \parallel o')]$) is simply the product of their individual probabilities. For any to finish, consider the probability that *neither* o nor

o' has occurred (by time t):

$$p[\neg o \cap \neg o'] = p[\neg o] \times p[\neg o'] =$$
$$(1 - p[o]) \times (1 - p[o']) = 1 - p[o] - p[o'] + p[o] \times p[o']$$

The probability that either or both of o and o' has occurred is one minus this, so $p[\forall(o \parallel o')] = p[o] + p[o'] - p[o] \times p[o']$.

In what follows, we will drop $\Delta_\circ$ whenever they are fixed throughout a computation and can thus be neglected.

Taking $R(\rho) = \mathbb{I}$ for $\rho = \text{time}$, it is routine to observe that

- according to Equation (2), a basic assignment is a time-measuring, and
- according to Equation (3), a $\Delta\mathrm{Q}\llbracket.\rrbracket_.$ is an extension to every time-measuring.

Armed with that observation, we can start proving properties of time (as a resource) as per Definition 3.3.

Remark 4.4. Note that, according to Definition 4.2, we get $\Delta\mathrm{Q}\llbracket o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2\rrbracket = \Delta\mathrm{Q}\llbracket o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_1\rrbracket$. This may seem counter-intuitive at the first glance because $o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2 \neq o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_1$ and one expects that $\Delta\mathrm{Q}\llbracket.\rrbracket_.$ acts accordingly. Nevertheless, one realises that $\Delta\mathrm{Q}\llbracket o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2\rrbracket = \Delta\mathrm{Q}\llbracket o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_1\rrbracket$ is just fine because, intuitively, $o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2$ is as timely as $o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_1$. So, the timeliness analysis too should give equal results for the two. See the proof of Theorem 5.3 for the mathematical justification of that intuition.

5 Algebraic Structures

This section proves $\mathbb{O}$ with

- probabilistic choice forms a magma (Theorem 5.2);
- sequential composition forms a commutative monoid with $\perp$ as the absorbing element (Theorem 5.3);
- all-to-finish forms a commutative monoid with $\perp$ as the absorbing element (Theorem 5.4);
- any-to-finish forms a commutative monoid with $\top$ as the absorbing element (Theorem 5.6); and
- neither all-to-finish nor any-to-finish nor their combination form richer algebraic structures (Remarks 5.5, 5.7, and 5.10).

The statements of the results that appear hereafter in this paper as well as their proofs have repetitive occurrences of outcome variables and their respective timeliness analyses. In order to prevent repetition in all that, we assume that $\Delta\mathrm{Q}\llbracket o_1\rrbracket = \delta_1$, $\Delta\mathrm{Q}\llbracket o_2\rrbracket = \delta_2$, and $\Delta\mathrm{Q}\llbracket o_3\rrbracket = \delta_3$; and that, unless stated otherwise, δ_1, δ_2, and δ_3 are all in their CDF representations.

Proposition 5.1 is a simple yet handy result that we will use frequently in our proofs.

Proposition 5.1. *Suppose that $o_1 = o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3$. Then, $\odot\!\odot$ time $\vDash \delta_1(t) = \int (\delta_2' * \delta_3')(t)\,\mathrm{d}t$.*

Proof. According to Definition 4.2, for all suitably ranged t,

$$\delta_1'(t) = (\delta_2' * \delta_3')(t).$$

Integrating the two sides, one gets

$$\int \delta_1'(t)\,\mathrm{d}t = \delta_1(t) = \int (\delta_2' * \delta_3')(t)\,\mathrm{d}t,$$

as desired. □

Theorem 5.2. *$(\mathbb{O}, \leftrightharpoons)$ forms a magma when observing time.*

Proof. The only property that is required is closedness of $\mathbb{O}$ under $\leftrightharpoons$, which is already a part of Definition 2.1. □

A magma is the weakest algebraic structure. That is because $\leftrightharpoons$ is not even associative. Despite that, one can still reassociate expressions with two consecutive occurrences of $\leftrightharpoons$. The only thing is that the coefficients will change by such reassociations. Lemmata 8.1 and 8.2 give the exact formulae.

One may notice that Theorem 5.2 avoids the compact notation introduced by Definition 3.5 for specifying the algebraic structure $(\mathbb{O}, \leftrightharpoons)$ forms. That is because, in Definitions 3.3 and 3.4, we are only interested in the algebraic structures established using equational (or otherwise binary relational) theories. Closedness – the only property required by a magma – is not an equational (or otherwise binary relational) property.

Theorem 5.3. *$\odot\!\odot$ time $\vDash (\mathbb{O}, \bullet\!\!\twoheadrightarrow\!\!\bullet) :$ commutative monoid with $\perp$ as the absorbing element.*

Proof. There are four properties to establish:

Associativity. $\odot\!\odot$ time $\vDash o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, (o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3) = (o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2) \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3$ because

$$\begin{aligned}\Delta\mathrm{Q}\llbracket o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, (o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3)\rrbracket &= (\delta_1 * \delta_2) * \delta_3 = \delta_1 * (\delta_2 * \delta_3)\\ &= \Delta\mathrm{Q}\llbracket (o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2) \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3\rrbracket.\end{aligned}$$

The associativity of convolution is due to Strichartz [7, §3.3].

Commutativity. *$\odot\!\odot$ time $\vDash o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2 = o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_1$* because, according to Proposition 5.1

$$\begin{aligned}&\Delta\mathrm{Q}\llbracket o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_2\rrbracket = \\ &\quad\int (\delta_1' * \delta_2')(t)\,\mathrm{d}t = \iint \delta_1'(t)\delta_2'(t-\tau)\,\mathrm{d}t\,\mathrm{d}\tau = \\ &\quad\iint \delta_1'(\tau)\delta_2'(\tau - t)\,\mathrm{d}\tau\,\mathrm{d}t = \int (\delta_2' * \delta_1')(\tau)\,\mathrm{d}\tau \\ &\qquad = \Delta\mathrm{Q}\llbracket o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_1\rrbracket.\end{aligned}$$

Identity Element. Take $e = \top$. Recall that $\Delta\mathrm{Q}\llbracket e\rrbracket = \mathbf{1}$ in the CDF representation and $(\mathbf{1})' = \boldsymbol{\delta}$, where $\boldsymbol{\delta}$ is Dirac's unitary impulse function. Choose an outcome o such that $\Delta\mathrm{Q}\llbracket o\rrbracket = \delta'$ so that δ be in its CDF representation. Then, by Definition 4.2

$$\begin{aligned}\Delta\mathrm{Q}\llbracket e \bullet\!\!\twoheadrightarrow\!\!\bullet\, o\rrbracket = \int (\boldsymbol{\delta} * \delta')(t)\,\mathrm{d}t = \int \delta'(t)\,\mathrm{d}t = \delta \\ = \Delta\mathrm{Q}\llbracket o\rrbracket\end{aligned}$$

implying $\odot\!\odot$ time $\vDash \top \bullet\!\!\twoheadrightarrow\!\!\bullet\, o = o$. Note that, because convolution is commutative, $\top$ is also the right identity element.

Absorbing Element. $\circledcirc$ *time* $\vDash \bot \bullet\!\!\to\!\!\bullet\ o = \bot$ because, according to Proposition 5.1,

$$\begin{aligned}\Delta Q[\![\bot \bullet\!\!\to\!\!\bullet\ o]\!] &= \int (\mathbf{0}' * \delta')(t)\,\mathrm{d}t \\ &= \iint 0 \times \delta'(t)\,\mathrm{d}\tau\,\mathrm{d}t = \mathbf{0} = \Delta Q[\![\bot]\!].\end{aligned}$$

Commutativity of $\bullet\!\!\to\!\!\bullet$ implies that $o \bullet\!\!\to\!\!\bullet \bot = \bot$ too. □

Theorem 5.4. $\circledcirc$ *time* $\vDash (\mathbb{O}, \|^{\forall})$: *commutative monoid with* $\bot$ *as the absorbing element.*

Proof. There are four properties to establish:

Associativity. $\circledcirc$ time $\vDash o_1 \|^{\forall} (o_2 \|^{\forall} o_3) = (o_1 \|^{\forall} o_2) \|^{\forall} o_3$ because

$$\begin{aligned}\Delta Q[\![o_1 \|^{\forall} (o_2 \|^{\forall} o_3)]\!] &= (\delta_1 \times \delta_2) \times \delta_3 = \delta_1 \times (\delta_2 \times \delta_3) \\ &= \Delta Q[\![(o_1 \|^{\forall} o_2) \|^{\forall} o_3]\!].\end{aligned}$$

Commutativity. Follows from the commutativity of function multiplication. That is, $\circledcirc$ time $\vDash o_1 \|^{\forall} o_2 = o_2 \|^{\forall} o_1$ because

$$\Delta Q[\![o_1 \|^{\forall} o_2]\!] = \delta_1 \times \delta_2 = \delta_2 \times \delta_1 = \Delta Q[\![o_2 \|^{\forall} o_1]\!].$$

Identity Element. Take $e = \top$. Choose an outcome o such that $\Delta Q[\![o]\!] = \delta$ so that δ be in its CDF representation. Then, by Definition 4.2

$$\Delta Q[\![e \|^{\forall} o]\!] = \mathbf{1} \times \delta = \delta = \Delta Q[\![o]\!],$$

which implies $\circledcirc$ time $\vDash \top \|^{\forall} o = o$. Note that, because function multiplication is commutative, $\top$ is also the right identity element.

Absorbing Element. $\circledcirc$ *time* $\vDash \bot \|^{\forall} o = \bot$ because

$$\Delta Q[\![\bot \|^{\forall} o]\!] = \mathbf{0} \times \delta = \mathbf{0} = \Delta Q[\![\bot]\!].$$

The result follows by the commutativity of $\|^{\forall}$. □

Remark 5.5. It is important to notice that, when observing time, $(\mathbb{O}, \|^{\forall})$ does *not* form a group. That is because, in general, an outcome has no inverse element - intuitively, one can never undo an outcome!

In order to prove that claim formally, suppose otherwise. That is, suppose that there exist a pair of outcomes o_1 and o_2 such that $o_1 \|^{\forall} o_2 = \top$. Then,

$$\Delta Q[\![o_1 \|^{\forall} o_2]\!] = \Delta Q[\![\top]\!] \Rightarrow \delta_1 \times \delta_2 = \mathbf{1} \Rightarrow \delta_2 = \frac{\mathbf{1}}{\delta_1}.$$

However, given that $\delta_1 \leq \mathbf{1}$, we get $\delta_2 \geq \mathbf{1}$. The latter inequality can only be satisfied when $o_1 = \top$. Restricting the application of ΔQSD to perfection is not practical.

Theorem 5.6. $\circledcirc$ *time* $\vDash (\mathbb{O}, \|^{\exists})$: *commutative monoid with* $\top$ *as the absorbing element.*

Proof. There are four properties to establish:

Associativity. Suppose that $\Delta Q[\![o_1]\!] = \delta_1$, $\Delta Q[\![o_2]\!] = \delta_2$, and $\Delta Q[\![o_3]\!] = \delta_3$, all in their CDF representations. Then, $\circledcirc$ time $\vDash o_1 \|^{\forall} (o_2 \|^{\forall} o_3) = (o_1 \|^{\forall} o_2) \|^{\forall} o_3$ because

$$\begin{aligned}\Delta Q[\![o_1 \|^{\exists} (o_2 \|^{\exists} o_3)]\!] &= \delta_1 + (\delta_2 + \delta_3 - \delta_2\delta_3) \\ &\quad - \delta_1(\delta_2 + \delta_3 - \delta_2\delta_3) = \\ \delta_1 + \delta_2 + \delta_3 &- \delta_2\delta_3 - \delta_1\delta_2 - \delta_1\delta_3 + \delta_1\delta_2\delta_3 = \\ (\delta_1 + \delta_2 - \delta_1\delta_2) &+ \delta_3 - (\delta_1 + \delta_2 - \delta_1\delta_2)\delta_3 = \\ \Delta Q[\![(o_1 \|^{\exists} o_2) &\|^{\exists} o_3]\!].\end{aligned}$$

Commutativity. Follows from the commutativity of function multiplication and addition. That is, $\circledcirc$ time $\vDash o_1 \|^{\exists} o_2 = o_2 \|^{\exists} o_1$ because

$$\begin{aligned}\Delta Q[\![o_1 \|^{\exists} o_2]\!] &= \delta_1 + \delta_2 - \delta_1\delta_2 \\ &= \delta_2 + \delta_1 - \delta_2\delta_1 = \Delta Q[\![o_2 \|^{\exists} o_1]\!].\end{aligned}$$

Identity Element. Take $e = \bot$. Choose an outcome o such that $\Delta Q[\![o]\!] = \delta$ so that δ be in its CDF representation. Then, by Definition 4.2

$$\Delta Q[\![e \|^{\exists} o]\!] = \mathbf{0} + \delta - \mathbf{0} \times \delta = \delta = \Delta Q[\![o]\!],$$

which implies $\circledcirc$ time $\vDash \bot \|^{\exists} o = o$. It follows from commutativity of $\|^{\exists}$ that $\bot$ is also the right identity element.

Absorbing Element. $\circledcirc$ *time* $\vDash \top \|^{\exists} o = \top$ because

$$\Delta Q[\![\top \|^{\exists} o]\!] = \mathbf{1} + \delta - \mathbf{1} \times \delta = \mathbf{1} = \Delta Q[\![\top]\!].$$

The result follows by commutativity of $\|^{\forall}$. □

Remark 5.7. Similar to the case for $\|^{\forall}$, it is important to note that, when observing time, $(\mathbb{O}, \|^{\exists})$ does not form a group. Again, it is the lack of an inverse element that is causing the trouble. Here is how:

Suppose that there exist a pair of outcomes o_1 and o_2 such that $o_1 \|^{\exists} o_2 = \bot$. Then,

$$\begin{aligned}\Delta Q[\![o_1 \|^{\exists} o_2]\!] = \Delta Q[\![\bot]\!] &\Rightarrow \delta_1 + \delta_2 - \delta_1 \times \delta_2 = \mathbf{0} \\ &\Rightarrow \delta_2 = \frac{\delta_1}{\delta_1 - 1}.\end{aligned}$$

However, because $\delta_1 \leq \mathbf{1}$, we get $\delta_2 \leq \mathbf{0}$. But, only $\bot$ can satisfy the latter inequality. There is no reason to develop a system all the outcomes of which will fail unconditionally!

Having established that both $(\mathbb{O}, \|^{\forall})$ and $(\mathbb{O}, \|^{\exists})$ form commutative monoids, when observing time, one immediately wonders whether $(\mathbb{O}, \|^{\forall}, \|^{\exists})$ or $(\mathbb{O}, \|^{\exists}, \|^{\forall})$ form semirings. Neither of those are, however, a semiring because they do not distribute over one another.

Proposition 5.8 and Corollary 5.9 help Remark 5.10 demonstrate how the above desirable dstributivities fail. In Proposition 5.8, we write $\circledcirc$ time $\vDash o_1 \leq o_2$ for when $\Delta Q[\![o_1]\!] \leq \Delta Q[\![o_2]\!]$. This is a routine extension of Definition 4.2 w.r.t. Definition 3.4 that is in line with the partial ordering of ΔQs.

Proposition 5.8. $\circledcirc$ *time* $\vDash o_1 \|^{\exists} o_2 \leq o_1$ *and* $\circledcirc$ *time* $\vDash o_1 \|^{\exists} o_2 \leq o_2$*, for every* $o_1, o_2 \in \mathbb{O}$.

Proof. We prove that $\odot\odot$ time $\vDash o_1 \parallel^{\exists} o_2 \leq o_1$. By commutativity of $\parallel^{\exists}$ (Theorem 5.6), it also follows that $\odot\odot$ time $\vDash o_1 \parallel^{\exists} o_2 \leq o_2$.

In order to prove the desired inequality, suppose otherwise. That is, suppose that $\otimes$time $\vDash o_1 \parallel^{\exists} o_2 > o_1$. Then, according to Equation (1):

$$\begin{aligned}\Delta\mathrm{Q}[\![o_1 \parallel^{\exists} o_2]\!] < \Delta\mathrm{Q}[\![o_1]\!] &\Rightarrow \delta_1 + \delta_2 - \delta_1\delta_2 < \delta_1\\ &\Rightarrow \delta_2 - \delta_1\delta_2 < \mathbf{0}\\ &\Rightarrow \delta_2(1-\delta_1) < \mathbf{0}\end{aligned}$$

But, that is impossible because, $\delta_2 \geq 0$ and $\delta_1 \leq 1$. □

Corollary 5.9. $\odot\odot$ *time* $\vDash \exists(o_1 \parallel o_2) = \top$ *implies* $o_1 = \top$ *and* $o_2 = \top$.

Proof. By Proposition 5.8, $\exists(o_1 \parallel o_2) = \top \leq o_1$, which implies $o_1 = \top$. □

Remark 5.10. Neither $(\mathbb{O}, \parallel^{\forall}, \parallel^{\exists})$ nor $(\mathbb{O}, \parallel^{\exists}, \parallel^{\forall})$ form a semiring, when observing time. For either of those two to be the case, $\parallel^{\forall}$ and $\parallel^{\exists}$ need to distribute over one another. Here, we investigate each distributivity.

The first distributivity requirement is

$$o_1 \parallel^{\exists} (o_2 \parallel^{\forall} o_3) \stackrel{?}{=} (o_1 \parallel^{\exists} o_2) \parallel^{\forall} (o_1 \parallel^{\exists} o_3) \tag{4}$$

Now, note that

$$\Delta\mathrm{Q}[\![o_1 \parallel^{\exists} (o_2 \parallel^{\forall} o_3)]\!] = \delta_1 + \delta_2\delta_3 - \delta_1\delta_2\delta_3 \tag{5}$$

and

$$\begin{aligned}&\Delta\mathrm{Q}[\![(o_1 \parallel^{\exists} o_2) \parallel^{\forall} (o_1 \parallel^{\exists} o_3)]\!]\\ &\quad = (\delta_1 + \delta_2 - \delta_1\delta_2)(\delta_1 + \delta_3 - \delta_1\delta_3)\\ &\quad = \delta_1^2 + \delta_1\delta_3 - \delta_1^2\delta_3 + \delta_1\delta_2 + \delta_3\delta_2 - \delta_1\delta_3\delta_2 - \delta_1^2\delta_2 - \delta_1\delta_2\delta_3\\ &\qquad\quad + \delta_1^2\delta_2\delta_3 \end{aligned}\tag{6}$$

In the favour of Equation (4), then, Equating the right-hand-sides of Equations (5) and (6), one gets

$$\delta_1^2 + \delta_1\delta_3 - \delta_1^2\delta_3 + \delta_1\delta_2 - \delta_1^2\delta_2 - \delta_1\delta_2\delta_3 + \delta_1^2\delta_2\delta_3 - \delta_1 = \mathbf{0} \Rightarrow$$
$$\delta_1[\delta_1 + \delta_3 - \delta_1\delta_3 + \delta_2 - \delta_1\delta_2 - \delta_2\delta_3 + \delta_1\delta_2\delta_3 - 1] = \mathbf{0}$$

Hence, either $\delta_1 = \mathbf{0}$ or

$$\delta_1 + \delta_3 - \delta_1\delta_3 + \delta_2 - \delta_1\delta_2 - \delta_2\delta_3 + \delta_1\delta_2\delta_3 = \mathbf{1} \Rightarrow$$
$$(\delta_1 + \delta_3 - \delta_1\delta_3) + \delta_2 - \delta_2(\delta_1 + \delta_3 - \delta_1\delta_3) = \mathbf{1} \Rightarrow$$
$$\Delta\mathrm{Q}[\![(o_1 \parallel^{\exists} o_3) \parallel^{\exists} o_2]\!] = \top.$$

In other words, it follows by Corollary 5.9 that Equation (4) can only hold under the trivial conditions when $o_1 = \bot$ or $o_1 = o_2 = o_3 = \top$.

The second distributivity requirement is

$$o_1 \parallel^{\forall} (o_2 \parallel^{\exists} o_3) \stackrel{?}{=} (o_1 \parallel^{\forall} o_2) \parallel^{\exists} (o_1 \parallel^{\forall} o_3) \tag{7}$$

Now, note that

$$\begin{aligned}\Delta\mathrm{Q}[\![o_1 \parallel^{\forall} (o_2 \parallel^{\exists} o_3)]\!] &= \delta_1(\delta_2 + \delta_3 - \delta_2\delta_3)\\ &= \delta_1\delta_2 + \delta_1\delta_3 - \delta_1\delta_2\delta_3\end{aligned}\tag{8}$$

and

$$\begin{aligned}&\Delta\mathrm{Q}[\![(o_1 \parallel^{\forall} o_2) \parallel^{\exists} (o_1 \parallel^{\forall} o_3)]\!]\\ &\qquad = (\delta_1\delta_2) + (\delta_1\delta_3) - (\delta_1\delta_2)(\delta_1\delta_3)\\ &\qquad = \delta_1\delta_2 + \delta_1\delta_3 - \delta_1^2\delta_2\delta_3\end{aligned}\tag{9}$$

In the favour of Equation (7), then, Equating the right-hand-sides of Equations (8) and (9), one gets

$$\delta_1\delta_2 + \delta_1\delta_3 - \delta_1\delta_2\delta_3 = \delta_1\delta_2 + \delta_1\delta_3 - \delta_1^2\delta_2\delta_3$$

which implies that Equation (7) only holds trivially when $\delta_1 = \mathbf{1} \wedge \delta_2 \neq \mathbf{0} \wedge \delta_3 \neq \mathbf{0}$, i.e., when $o_1 = \top \wedge o_2 \neq \bot \wedge o_3 \neq \bot$.

6 Distributivity

In this section, we go through a number of distributivity results that we succeeded in establishing (Lemmata 6.1–6.4). In the next section, we will give detail on why we find some distributivity results not to hold **in general**. We work out conditions required for the availability of those distributivity results. Such a discussion is still useful because it helps one to verify, under special circumstances, whether their given IRVs can satisfy the provided conditions.

Here is a syntactic convention that we will adhere to: When, in an equivalence, a couple of $\leftrightarrows$s are used without weights, each on one and only one side of the equivalence, we assume that the weights of those $\leftrightarrows$s are the same. Therefore, we do not bother to repeat those weights. For example, in the lemma below, there are supposed to exist to weights m_2 and m_3 such that $o_2 \overset{m_2}{\underset{m_3}{\leftrightarrows}} o_3$ and $(o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \overset{m_2}{\underset{m_3}{\leftrightarrows}} (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_3)$.

Lemma 6.1. $\odot\odot$ *time* $\vDash o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \leftrightarrows o_3) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightarrows (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_3)$, *where* $o_1, o_2, o_3 \in \mathbb{O}$.

Proof. Fix some coefficients m_2 and m_3. $\odot\odot$ time $\vDash o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \overset{m_2}{\underset{m_3}{\leftrightarrows}} o_3) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \overset{m_2}{\underset{m_3}{\leftrightarrows}} (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_3)$ because

$$\begin{aligned}&\Delta\mathrm{Q}[\![o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \overset{m_2}{\underset{m_3}{\leftrightarrows}} o_3)]\!] = \qquad\qquad \text{(Proposition 5.1)}\\
&\int \left(\delta_1' * \left(\frac{m_2}{m_2+m_3}\delta_2 + \frac{m_3}{m_2+m_3}\delta_3\right)'\right)(t)\,\mathrm{d}t =\\
&\int \left(\delta_1' * \left(\frac{m_2}{m_2+m_3}\delta_2' + \frac{m_3}{m_2+m_3}\delta_3'\right)\right)(t)\,\mathrm{d}t =\\
&\iint \delta_1'(t-\tau)\left(\frac{m_2}{m_2+m_3}\delta_2'(\tau) + \frac{m_3}{m_2+m_3}\delta_3'(\tau)\right)\mathrm{d}\tau\,\mathrm{d}t =\\
&\iint \frac{m_2}{m_2+m_3}\delta_1'(t-\tau)\delta_2'(\tau)\,\mathrm{d}\tau\,\mathrm{d}t +\\
&\qquad \iint \frac{m_3}{m_2+m_3}\delta_1'(t-\tau)\delta_3'(\tau)\,\mathrm{d}\tau\,\mathrm{d}t =\\
&\int \frac{m_2}{m_2+m_3}\int \delta_1'(t-\tau)\delta_2'(\tau)\,\mathrm{d}\tau\,\mathrm{d}t +\\
&\qquad \int \frac{m_3}{m_2+m_3}\int \delta_1'(t-\tau)\delta_3'(\tau)\,\mathrm{d}\tau\,\mathrm{d}t =\end{aligned}$$

$$
\begin{aligned}
&\frac{m_2}{m_2+m_3}\int(\delta_1' * \delta_2')(t)\,\mathrm{d}t + \\
&\quad\frac{m_3}{m_2+m_3}\int(\delta_1' * \delta_3')(t)\,\mathrm{d}t = \qquad \text{(Proposition 5.1)}\\
&\frac{m_2}{m_2+m_3}\Delta\mathrm{Q}[\![o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2]\!]\,\mathrm{d}t + \frac{m_3}{m_2+m_3}\Delta\mathrm{Q}[\![o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2]\!] = \\
&\Delta\mathrm{Q}[\![(o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \underset{m_3}{\overset{m_2}{\leftrightharpoons}} (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_3)]\!].
\end{aligned}
$$

□

Lemma 6.2. *$\odot\odot$ time $\vDash (o_1 \leftrightharpoons o_2) \bullet\!\!\rightarrow\!\!\bullet\, o_3 = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_3) \leftrightharpoons (o_2 \bullet\!\!\rightarrow\!\!\bullet\, o_3)$, where $o_1, o_2, o_3 \in \mathbb{O}$.*

Proof. When observing time,

$$
\begin{aligned}
&(o_1 \leftrightharpoons o_2) \bullet\!\!\rightarrow\!\!\bullet\, o_3 = && \text{(Theorem 5.3)}\\
&o_3 \bullet\!\!\rightarrow\!\!\bullet\, (o_1 \leftrightharpoons o_2) = && \text{(Lemma 6.1)}\\
&(o_3 \bullet\!\!\rightarrow\!\!\bullet\, o_1) \leftrightharpoons (o_3 \bullet\!\!\rightarrow\!\!\bullet\, o_2) = && \text{(Theorem 5.3)}\\
&(o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_3) \leftrightharpoons (o_2 \bullet\!\!\rightarrow\!\!\bullet\, o_3).
\end{aligned}
$$

□

Lemma 6.3. *When observing time, $\|^{\forall}$ distributed over $\leftrightharpoons$ from both left and right.*

Proof. Fix coefficients m_2 and m_3. $\odot\odot$time $\vDash o_1 \|^{\forall} (o_2 \underset{m_3}{\overset{m_2}{\leftrightharpoons}} o_3) = (o_1 \|^{\forall} o_3) \underset{m_3}{\overset{m_2}{\leftrightharpoons}} (o_1 \|^{\forall} o_3)$ because

$$
\begin{aligned}
&\Delta\mathrm{Q}[\![o_1 \|^{\forall} (o_2 \underset{m_3}{\overset{m_2}{\leftrightharpoons}} o_3)]\!] = \\
&\delta_1\left(\frac{m_2}{m_2+m_3}\delta_2 + \frac{m_3}{m_2+m_3}\delta_3\right) = \\
&\frac{m_2}{m_2+m_3}\delta_1\delta_2 + \frac{m_3}{m_2+m_3}\delta_1\delta_3 = \\
&\frac{m_2}{m_2+m_3}\Delta\mathrm{Q}[\![\forall(o_1 \parallel o_2)]\!] + \frac{m_3}{m_2+m_3}\Delta\mathrm{Q}[\![\forall(o_1 \parallel o_3)]\!] = \\
&\Delta\mathrm{Q}[\![(o_1 \|^{\forall} o_3) \underset{m_3}{\overset{m_2}{\leftrightharpoons}} (o_1 \|^{\forall} o_3)]\!].
\end{aligned}
$$

Similar to Lemma 6.2, one concludes that $\odot\odot$ time $\vDash (o_1 \leftrightharpoons o_2) \|^{\forall} o_3 = (o_1 \|^{\forall} o_3) \leftrightharpoons (o_1 \|^{\forall} o_3)$ too. □

Lemma 6.4. *When observing time, $\|^{\exists}$ distributes over $\leftrightharpoons$ from both left and right.*

Proof. Fix coefficients m_2 and m_3. $\odot\odot$time $\vDash o_1 \|^{\exists} (o_2 \underset{m_3}{\overset{m_2}{\leftrightharpoons}} o_3) = (o_1 \|^{\exists} o_3) \underset{m_3}{\overset{m_2}{\leftrightharpoons}} (o_1 \|^{\exists} o_3)$ because

$$
\begin{aligned}
&\Delta\mathrm{Q}[\![o_1 \|^{\exists} (o_2 \underset{m_3}{\overset{m_2}{\leftrightharpoons}} o_3))]\!] = \\
&\delta_1 + \left(\frac{m_2}{m_2+m_3}\delta_2 + \frac{m_3}{m_2+m_3}\delta_3\right) - \\
&\qquad\delta_1\left(\frac{m_2}{m_2+m_3}\delta_2 + \frac{m_3}{m_2+m_3}\delta_3\right) = \\
&\delta_1 + \left(\frac{m_2}{m_2+m_3}\delta_2 + \frac{m_3}{m_2+m_3}\delta_3\right) - \\
&\qquad\left(\frac{m_2}{m_2+m_3}\delta_1\delta_2 + \frac{m_3}{m_2+m_3}\delta_1\delta_3\right) = \\
&\frac{m_2}{m_2+m_3}(\delta_1+\delta_2-\delta_1\delta_2) + \frac{m_3}{m_2+m_3}(\delta_1+\delta_3-\delta_1\delta_3) = \\
&\frac{m_2}{m_2+m_3}\Delta\mathrm{Q}[\![\exists(o_1 \parallel o_2)]\!] + \frac{m_3}{m_2+m_3}\Delta\mathrm{Q}[\![\exists(o_1 \parallel o_3)]\!] = \\
&\Delta\mathrm{Q}[\![(o_1 \|^{\exists} o_3) \underset{m_3}{\overset{m_2}{\leftrightharpoons}} (o_1 \|^{\exists} o_3)]\!].
\end{aligned}
$$

Similar to Lemma 6.2, one concludes that $\odot\odot$ time $\vDash (o_1 \leftrightharpoons o_2) \|^{\exists} o_3 = (o_1 \|^{\exists} o_3) \leftrightharpoons (o_1 \|^{\exists} o_3)$ too. □

7 Potential Distributivity

Recall that out of the four operators of $\mathbb{P}$, three are commutative (i.e., $\bullet\!\!\rightarrow\!\!\bullet$, $\|^{\forall}$, and $\|^{\exists}$) and one is not (i.e., $\leftrightharpoons$). Only when the latter is the operator outside the parentheses, therefore, distributivity from right and left might differ. That gives rise to $2 \times \binom{3}{1} + 2 \times \binom{3}{2} = 12$ possible ways for distributing $\mathbb{P}$ operators over one another.

We established three (Lemmata 6.1–6.4) and refuted two (Remark 5.10). Out of the remaining 7, we have selected 3, for which we have come to condition equations we do not know how to solve in their full generality, if soluble at all. Nevertheless, our understanding is that if spelled out properly, those conditions can help the practising ΔQSD engineer with their specific decision makings if they come to require instances of such distributivities.

Right-Distributivity of Probabilistic Choice over Sequential Composition. When observing time, for

$$
(o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \underset{m'}{\overset{m}{\leftrightharpoons}} o_3 \stackrel{?}{=} (o_1 \underset{m'}{\overset{m}{\leftrightharpoons}} o_3) \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \underset{m'}{\overset{m}{\leftrightharpoons}} o_3) \tag{10}
$$

to hold, according to Proposition 5.1,

$$
\begin{aligned}
&\Delta\mathrm{Q}[\![(o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \underset{m'}{\overset{m}{\leftrightharpoons}} o_3]\!]\\
&\quad= \frac{m}{m+m'}\int(\delta_1' * \delta_2')(t)\,\mathrm{d}t + \frac{m'}{m+m'}\delta_3\\
&\quad= \frac{m}{m+m'}\iint \delta_1'(\tau)\delta_2'(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t + \frac{m'}{m+m'}\delta_3
\end{aligned} \tag{11}
$$

and

$$
\begin{aligned}
&\Delta\mathrm{Q}[\![(o_1 \underset{m'}{\overset{m}{\leftrightharpoons}} o_3) \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \underset{m'}{\overset{m}{\leftrightharpoons}} o_3)]\!]\\
&\quad= \int\left(\frac{m}{m+m'}\delta_1' + \frac{m'}{m+m'}\delta_3'\right) * \\
&\qquad\qquad\left(\frac{m}{m+m'}\delta_2' + \frac{m'}{m+m'}\delta_3'\right)(t)\,\mathrm{d}t\\
&\quad= \iint\left(\frac{m}{m+m'}\delta_1'(t) + \frac{m'}{m+m'}\delta_3'(t)\right) \times \\
&\qquad\qquad\left(\frac{m}{m+m'}\delta_2'(t-\tau) + \frac{m'}{m+m'}\delta_3'(t-\tau)\right)\mathrm{d}\tau\,\mathrm{d}t
\end{aligned} \tag{12}
$$

For Equation (10) to hold, the right-hand-sides of Equations (11) and (12) need to be equal. That is,

$$\begin{aligned}\frac{m}{m+m'}\iint \delta_1'(\tau)\delta_2'(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t + \frac{m'}{m+m'}\delta_3 = \\ \iint\left(\frac{m}{m+m'}\delta_1'(t)+\frac{m'}{m+m'}\delta_3'(t)\right)\times \\ \left(\frac{m}{m+m'}\delta_2'(t-\tau)+\frac{m'}{m+m'}\delta_3'(t-\tau)\right)\mathrm{d}\tau\,\mathrm{d}t\end{aligned} \tag{13}$$

That is a differential equation we do not know a solution for in its full generality. Given particular values for δ_1, δ_2, and δ_3, however, the ΔQSD practitioner might be able to solve it, if soluble at all.

Distributivity of Sequential Composition over All-to-Finish. We proceed similarly for

$$(o_1 \parallel^{\forall} o_2) \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3 \stackrel{?}{=} (o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3) \parallel^{\forall} (o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3) \tag{14}$$

According to Proposition 5.1,

$$\begin{aligned}&\Delta Q[\![(o_1 \parallel^{\forall} o_2) \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3]\!] \\ &\quad= \int ((\delta_1\delta_2)' * \delta_3')(t)\,\mathrm{d}t \\ &\quad= \int ((\delta_1\delta_2' + \delta_1'\delta_2) * \delta_3')(t)\,\mathrm{d}t \\ &\quad= \iint (\delta_1(t)\delta_2'(t) + \delta_1'(t)\delta_2(t))\delta_3(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t\end{aligned}$$

and

$$\begin{aligned}&\Delta Q[\![(o_1 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3) \parallel^{\forall} (o_2 \bullet\!\!\twoheadrightarrow\!\!\bullet\, o_3)]\!] = \\ &\quad\iint \delta_1'(t)\delta_3'(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t \iint \delta_2'(t)\delta_3'(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t\end{aligned}$$

which together imply that, in order for Equation (14) to hold, solubility of the following differential equation is required:

$$\begin{aligned}&\iint (\delta_1(t)\delta_2'(t) + \delta_1'(t)\delta_2(t))\delta_3(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t = \\ &\quad\iint \delta_1'(t)\delta_3'(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t \iint \delta_2'(t)\delta_3'(t-\tau)\,\mathrm{d}\tau\,\mathrm{d}t.\end{aligned} \tag{15}$$

Right-Distributivity of Probabilistic Choice over All-to-Finish. Timeliness analysis of the following equation

$$(o_1 \parallel^{\forall} o_2) \overset{m}{\underset{m'}{\leftrightharpoons}} o_3 \stackrel{?}{=} (o_1 \overset{m}{\underset{m'}{\leftrightharpoons}} o_3) \parallel^{\forall} (o_2 \overset{m}{\underset{m'}{\leftrightharpoons}} o_3) \tag{16}$$

leads easily to

$$\begin{aligned}&\frac{m}{m+m'}(\delta_1\delta_2) + \frac{m'}{m+m'}\delta_3 = \\ &\quad\left(\frac{m}{m+m'}\delta_1 + \frac{m'}{m+m'}\delta_3\right)\left(\frac{m}{m+m'}\delta_2 + \frac{m'}{m+m'}\delta_3\right)\end{aligned} \tag{17}$$

This is clearly contradictory, so this distributivity does not hold.

Remark 7.1. Intuitively, the three distributivity results investigated in this section all suffer from the same problem. They are **unphysical**! Take Equation (16), for example, depending on how the probabilistic choice goes in reality, there is always a chance for o_3 to be performed twice on the right-hand-side; yet, there is only chance for one performance of it on the left-hand-side. We suspect, therefore, that those equivalences essentially do not hold, even though we do not yet have complete mathematical evidence for this. That is a call for a more elaborate mathematical framework, which we will discuss in Section 9.

8 Other Equivalences Used in Practice

ΔQSD is already in use by its practitioners, who, amongst other usages, simplify outcome expressions according to their timeliness analysis. In particular, Figure 1 distils a list of equivalences that are used in such simplifications. In this section, we will prove those equivalences one by one.

These equivalences provide the basis for rewrite rules that are useful for construction of normal forms, such as expressing a given system as a convolution of probabilistic choices or a probabilistic choice of convolutions. Such rewriting allows for: extraction of common sub-expressions permitting aggregation of failure rates (distinguishing between conditional and non-conditional failure); identifying minimal delays; and, highlighting branching probabilities to identify issues of relative criticality. This is useful for quickly assessing whether a particular outcome decomposition is *feasible* without having to compute the complete ΔQ. For example, if we express each ΔQ as a convolution of a Dirac $\boldsymbol{\delta}$ function with mass at some time t with a distribution starting at zero, then the identities can be used to bring the $\boldsymbol{\delta}$ functions together. Using the identity $\boldsymbol{\delta}(t_1) * \boldsymbol{\delta}(t_2) = \boldsymbol{\delta}(t_1 + t_2)$, the minimum delay can be extracted from this normal form; if this exceeds the maximum delay of the specification then the system is infeasible and the design needs to be revised.

Before we delve into Figure 1, we prove a result about re-associating probabilistic choice. Given an expression with two consecutive probabilistic choices, one of which wrapped inside a pair of parentheses, the ΔQSD practitioner might be interested in wrapping the other two inside a pair of parentheses – re-associating the probabilistic choices, in effect. Lemmata 8.1 and 8.2 give the conditions on the coefficients of those probabilistic choices.

Rather than the notation introduced in Definition 2.1, however, Lemmata 8.1 and 8.2 employ an equivalent notation which is slightly different and more compact:

$$o_1 \overset{[p]}{\leftrightharpoons} o_2$$

shows a probabilistic choice, the probability of which reducing to o_1 is p. One concludes immediately that, in such a case, the probability of the above outcome expression reducing to o_2 is $1 - p$.

The reason why we rather choose this compact notation in this section is that expressing relationships between the coefficients in this notation is considerably less involved, as one sees in Lemmata 8.1 and 8.2.

$$
\begin{array}{llllll}
\bot \leftrightharpoons \bot = \bot & \top \leftrightharpoons \top = \top & \bot \bullet\!\!\rightarrow\!\!\bullet\, o = \bot & o \bullet\!\!\rightarrow\!\!\bullet \bot = \bot & \top \bullet\!\!\rightarrow\!\!\bullet\, o = o & o \bullet\!\!\rightarrow\!\!\bullet \top = o
\end{array}
$$

$$
(o_1 \leftrightharpoons \bot) \bullet\!\!\rightarrow\!\!\bullet\, o_2 = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons \bot \qquad o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \leftrightharpoons \bot) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons \bot \qquad (o_1 \leftrightharpoons \top) \bullet\!\!\rightarrow\!\!\bullet\, o_2 = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons o_2
$$

$$
o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \leftrightharpoons \top) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons o_1 \qquad \bot \overset{[p]}{\leftrightharpoons} (\bot \overset{[q]}{\leftrightharpoons} o) = \bot \overset{[p+(1-p)q]}{\leftrightharpoons} o \qquad o_1 \overset{[p]}{\leftrightharpoons} (o_2 \overset{[q]}{\leftrightharpoons} \top) = o_2 \overset{[q(1-p)]}{\leftrightharpoons} (o_1 \overset{\left[\frac{p}{1-q(1-p)}\right]}{\leftrightharpoons} \top)
$$

Figure 1. Equivalences Already in Use in the Practice of ΔQSD

Lemma 8.1. *$o_1 \overset{[p]}{\leftrightharpoons} (o_2 \overset{[q]}{\leftrightharpoons} o_3) = (o_1 \overset{[p']}{\leftrightharpoons} o_2) \overset{[q']}{\leftrightharpoons} o_3$ iff*

$$
\begin{cases}
p' = \dfrac{p}{1-(1-p)(1-q)} & (18) \\[2ex]
q' = 1-(1-p)(1-q) & (19)
\end{cases}
$$

Proof.

$$
\begin{aligned}
&o_1 \overset{[p]}{\leftrightharpoons} (o_2 \overset{[q]}{\leftrightharpoons} o_3) = (o_1 \overset{[p']}{\leftrightharpoons} o_2) \overset{[q']}{\leftrightharpoons} o_3 \Leftrightarrow \\
&\Delta Q[\![o_1 \overset{[p]}{\leftrightharpoons} (o_2 \overset{[q]}{\leftrightharpoons} o_3)]\!] = \Delta Q[\![(o_1 \overset{[p']}{\leftrightharpoons} o_2) \overset{[q']}{\leftrightharpoons} o_3]\!] \Leftrightarrow \\
&p\delta_1 + (1-p)[q\delta_2 + (1-q)\delta_3] = \\
&\qquad\qquad q'[p'\delta_1 + (1-p')\delta_2] + (1-q')\delta_3
\end{aligned}
$$

which gives rise to a system of three equations with two variables p' and q':

$$
\begin{cases}
p = p'q' & (20) \\
(1-p)q = (1-p')q' & (21) \\
(1-p)(1-q) = 1-q' & (22)
\end{cases}
$$

One gets Equation (19) readily from Equation (22). Then, Equation (18) follows from Equation (20). One can also check the above system of equations for consistency by swapping Equations (18) and (19) into Equation (21). □

Lemma 8.2. *$(o_1 \overset{[p]}{\leftrightharpoons} o_2) \overset{[q]}{\leftrightharpoons} o_3 = o_1 \overset{[p']}{\leftrightharpoons} (o_2 \overset{[q']}{\leftrightharpoons} o_3)$ iff*

$$
\begin{cases}
p' = pq & (23) \\[2ex]
q' = \dfrac{q(1-p)}{1-pq} & (24)
\end{cases}
$$

Proof.

$$
\begin{aligned}
&(o_1 \overset{[p]}{\leftrightharpoons} o_2) \overset{[q]}{\leftrightharpoons} o_3 = o_1 \overset{[p']}{\leftrightharpoons} (o_2 \overset{[q']}{\leftrightharpoons} o_3) \Leftrightarrow \\
&\Delta Q[\![(o_1 \overset{[p]}{\leftrightharpoons} o_2) \overset{[q]}{\leftrightharpoons} o_3]\!] = \Delta Q[\![o_1 \overset{[p']}{\leftrightharpoons} (o_2 \overset{[q']}{\leftrightharpoons} o_3)]\!] \Leftrightarrow \\
&q[p\delta_1 + (1-p)\delta_2] + (1-q)\delta_3 = \\
&\qquad\qquad p'\delta_1 + (1-p')[q'\delta_2 + (1-q')\delta_3]
\end{aligned}
$$

which gives rise to a system of three equations with two variables p' and q':

$$
\begin{cases}
p' = pq & (25) \\
(1-p)q = (1-p')q' & (26) \\
(1-p')(1-q') = 1-q & (27)
\end{cases}
$$

Equation (25) is already Equation (23). Substituting Equation (25) into Equation (26), one gets:

$$
(1-pq)q' = (1-p)q \Rightarrow q' = \frac{q(1-p)}{1-pq}.
$$

One can also check the above system of equations for consistency by swapping Equations (23) and (24) into Equation (27). □

Lemma 8.3. *The equivalences in Fig. 1 are correct.*

Proof. We will go through them one-by-one, starting from the top left to the bottom right.

- $\bot \underset{m_2}{\overset{m_1}{\leftrightharpoons}} \bot = \bot$

$$
\Delta Q[\![\bot \underset{m_2}{\overset{m_1}{\leftrightharpoons}} \bot]\!] = \frac{m_1}{m_1+m_2}\mathbf{0} + \frac{m_2}{m_1+m_2}\mathbf{0} = \mathbf{0} = \Delta Q[\![\bot]\!].
$$

- $\top \underset{m_2}{\overset{m_1}{\leftrightharpoons}} \top = \top$

$$
\Delta Q[\![\top \underset{m_2}{\overset{m_1}{\leftrightharpoons}} \top]\!] = \frac{m_1}{m_1+m_2}\mathbf{1} + \frac{m_2}{m_1+m_2}\mathbf{1} = \mathbf{1} = \Delta Q[\![\top]\!].
$$

- $\bot \bullet\!\!\rightarrow\!\!\bullet\, o = \bot$ and $o \bullet\!\!\rightarrow\!\!\bullet \bot = \bot$ were already established in the proof of Theorem 5.3.
- $\top \bullet\!\!\rightarrow\!\!\bullet\, o = o$ and $o \bullet\!\!\rightarrow\!\!\bullet \top = o$ were already established in the proof of Theorem 5.3.
- $(o_1 \leftrightharpoons \bot) \bullet\!\!\rightarrow\!\!\bullet\, o_2 = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons \bot$
  By Lemma 6.2 and the third equivalence of this lemma:

$$
(o_1 \leftrightharpoons \bot) \bullet\!\!\rightarrow\!\!\bullet\, o_2 = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons (\bot \bullet\!\!\rightarrow\!\!\bullet\, o_2) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons \bot.
$$

- $o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \leftrightharpoons \bot) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons \bot$
  By Lemma 6.1 and the fourth equivalence of this lemma:

$$
o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \leftrightharpoons \bot) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons (o_1 \bullet\!\!\rightarrow\!\!\bullet \bot) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons \bot.
$$

- $(o_1 \leftrightharpoons \top) \bullet\!\!\rightarrow\!\!\bullet\, o_2 = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons o_2$
  By Lemma 6.2 and the fifth equivalence of this lemma:

$$
(o_1 \leftrightharpoons \top) \bullet\!\!\rightarrow\!\!\bullet\, o_2 = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons (\top \bullet\!\!\rightarrow\!\!\bullet\, o_2) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons o_2.
$$

- $o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \leftrightharpoons \top) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons o_1$
  By Lemma 6.1 and the sixth equivalence of this lemma:

$$
o_1 \bullet\!\!\rightarrow\!\!\bullet\, (o_2 \leftrightharpoons \top) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons (o_1 \bullet\!\!\rightarrow\!\!\bullet \top) = (o_1 \bullet\!\!\rightarrow\!\!\bullet\, o_2) \leftrightharpoons o_1.
$$

- $\bot \overset{[p]}{\leftrightharpoons} (\bot \overset{[q]}{\leftrightharpoons} o) = \bot \overset{[p+(1-p)q]}{\leftrightharpoons} o$
  According to Lemma 8.1, $\bot \overset{[p]}{\leftrightharpoons} (\bot \overset{[q]}{\leftrightharpoons} o) = (\bot \overset{[p']}{\leftrightharpoons} \bot) \overset{[q']}{\leftrightharpoons} o$ for some p' and q'. The formulation of p' in terms of p and q does not matter because, according to the first

equivalence of this lemma, $\perp \overset{[p']}{\leftrightharpoons} \perp = \perp$. On the other hand,

$$q' = 1-(1-p)(1-q) = 1-(1-p-q+pq) = p+q-pq = p+(1-p)q.$$

- $o_1 \overset{[p]}{\leftrightharpoons} (o_2 \overset{[q]}{\leftrightharpoons} \top) = o_2 \overset{[q(1-p)]}{\leftrightharpoons} (o_1 \overset{\left[\frac{p}{1-q(1-p)}\right]}{\leftrightharpoons} \top)$.

  Let p_1, p_2, p_3, q_1, q_2, and q_3 be probability values such that

  1. $o_1 \overset{[p]}{\leftrightharpoons} (o_2 \overset{[q]}{\leftrightharpoons} \top) = (o_1 \overset{[p_1]}{\leftrightharpoons} o_2) \overset{[q_1]}{\leftrightharpoons} \top$
  2. $(o_1 \overset{[p_1]}{\leftrightharpoons} o_2) \overset{[q_1]}{\leftrightharpoons} \top = (o_2 \overset{[p_2]}{\leftrightharpoons} o_1) \overset{[q_2]}{\leftrightharpoons} \top$
  3. $(o_2 \overset{[p_2]}{\leftrightharpoons} o_1) \overset{[q_2]}{\leftrightharpoons} \top = o_2 \overset{[p_3]}{\leftrightharpoons} (o_1 \overset{[q_3]}{\leftrightharpoons} \top)$

  The aim is to calculate p_3 and q_3 in terms of p and q. We proceed in a stepwise fashion.

  According to Lemma 8.1,

$$p_1 = \frac{p}{1-(1-p)(1-q)} \tag{28}$$
$$q_1 = 1-(1-p)(1-q). \tag{29}$$

  On the other hand, it is easy to verify that

$$p_2 = 1 - p_1 \tag{30}$$
$$q_2 = q_1. \tag{31}$$

  Finally, by Lemma 8.2,

$$p_3 = p_2 q_2 \tag{32}$$
$$q_3 = \frac{q_2(1-p_2)}{1-p_2 q_2} \tag{33}$$

  Substituting Equations (28) and (29) into Equations (30) and (31), one gets:

$$\begin{aligned} p_2 &= 1 - p_1 = 1 - \frac{p}{1-(1-p)(1-q)} \\ &= \frac{\cancel{p}+q-pq-\cancel{p}}{1-(1-p)(1-q)} = \frac{q(1-p)}{1-(1-p)(1-q)} \qquad (34) \\ q_2 &= q_1 = 1-(1-p)(1-q). \qquad (35) \end{aligned}$$

  Then, substituting Equations (34) and (35) into Equations (32) and (33), one gets:

$$\begin{aligned} p_3 &= p_2 q_2 = \frac{q(1-p)}{\cancel{1-(1-p)(1-q)}} \times \cancel{(1-(1-p)(1-q))} \\ &= q(1-p) \qquad (36) \\ q_3 &= \frac{q_2(1-p_2)}{1-p_2q_2} \\ &= (1-(1-p)(1-q)) \times \frac{1 - \dfrac{q(1-p)}{1-(1-p)(1-q)}}{1-q(1-p)} \\ &= \frac{1-(1-p)(1-q)-q(1-p)}{1-q(1-p)} \\ &= \frac{p+\cancel{q}-\cancel{pq}-\cancel{q}+\cancel{pq}}{1-q(1-p)} = \frac{p}{1-q(1-p)} \qquad (37) \end{aligned}$$

  Equations (36) and (37) imply that

$$o_1 \overset{[p]}{\leftrightharpoons} (o_2 \overset{[q]}{\leftrightharpoons} \top) = o_2 \overset{[q(1-p)]}{\leftrightharpoons} (o_1 \overset{\left[\frac{p}{1-q(1-p)}\right]}{\leftrightharpoons} \top).$$

$\square$

Example usage: It is very useful in practice to be able to make a rapid assessment of a system's likely failure rate. Such rates can be directly extracted from the timeliness analysis. For example, if we write the sequential composition of two outcomes that may fail as $(o_1 \overset{[a]}{\leftrightharpoons} \perp) \bullet\!\!\rightarrow\!\!\bullet (o_2 \overset{[b]}{\leftrightharpoons} \perp)$, so that the failure probabilities are made explicit, we can use the identities of Figure 1 to bring out the failure probability of the combination.

$$\begin{aligned} &(o_1 \overset{[a]}{\leftrightharpoons} \perp) \bullet\!\!\rightarrow\!\!\bullet (o_2 \overset{[b]}{\leftrightharpoons} \perp) = && \text{(Lemmata 6.1 and 8.3)} \\ &((o_1 \overset{[a]}{\leftrightharpoons} \perp) \bullet\!\!\rightarrow\!\!\bullet o_2) \overset{[b]}{\leftrightharpoons} \perp = && \text{(Lemmata 6.2 and 8.3)} \\ &((o_1 \bullet\!\!\rightarrow\!\!\bullet o_2) \overset{[a]}{\leftrightharpoons} \perp) \overset{[b]}{\leftrightharpoons} \perp = && \text{(Lemmata 8.2 and 8.3)} \\ &(o_1 \bullet\!\!\rightarrow\!\!\bullet o_2) \overset{[ab]}{\leftrightharpoons} \perp \end{aligned}$$

Other equivalences can be used to accumulate failure probabilities in more complex outcome expressions. Appropriate use of algebraic rewriting, can allow for symbolic extraction of failure rates depending **only** on the decomposition in terms of outcomes. Creating symbolic relationships between outcome decompositions can permit more principled pruning of the design space.

Remark 8.4. The very last equivalence in Fig. 1 was incorrectly formulated prior to this paper. Thanks to the formalisation developed in this paper, that mistake was corrected.

9 Conclusion and Future Work

This paper lays down model-theoretic foundations for resource analysis à la ΔQSD. On that foundation, it builds particularly for time as a resource consumed by outcomes. In doing so, it enables timeliness analysis via the study of quality attenuation, capturing both delay and failure together. With our focus being on timeliness exclusively, we give proofs for the algebraic structures the ΔQSD operators form with outcome expressions (Theorems 5.2–5.6). We refute the formation of richer algebraic structures by the ΔQSD operators and outcome expressions (Remarks 5.5, 5.7, and 5.10). We prove distributivity results about the ΔQSD operators (Lemmata 6.1–6.4) and provide guidelines for studying the existence of potential distributivity (Section 7). Finally, we prove a dozen of equivalences that have already been used in the practice of ΔQSD over the past few decades (Lemmata 8.1–8.3).

Study of the algebraic properties of other resources à la ΔQSD is our immediate future work. An algebraic categorisation of resources is expected eventually.

A sound theoretical foundation is essential for the construction of robust tool support, which is a prerequisite for wider application of the ΔQSD paradigm. Currently, there is a numerically-based tool prototype. Yet, to deal effectively with large complex systems, this needs to be made more symbolic. The aim is for the expressions to be simplified before calculation, and to be able to represent performance unknowns. Algebraic structures are essential for correctly manipulating and simplifying expressions. This work informs both ongoing practical work and tool development. Conversely, consideration of specific aspects of system design and operation inform the most productive directions for the theoretical developments.

References

[1] D. Coutts, N. Davies, M. Szamotulski, and P. Thompson. 2020. *Introduction to the design of the Data Diffusion and Networking for Cardano Shelley.* Technical Report. IOHK.

[2] N. Davies, P. Thompson, G. Young, J. Newton, B. Teigen, and M. Olden. 2021. Measuring Network Impact on Application Outcomes Using Quality Attenuation. In *Measuring Network Quality for End-Users.* Internet Architecture Board.

[3] S.H. Haeri, P. Thompson, N. Davies, P. Van Roy, K. Hammond, and J. Chapman. 2022. Mind Your Outcomes: The ΔQSD Paradigm for Quality-Centric Systems Development and Its Application to a Blockchain Case Study. *Computers* 11, 3 (2022), 45.

[4] Bjørn Ivar Teigen, Neil Davies, Kai Olav Ellefsen, Tor Skeie, and Jim Torresen. 2022. Quantifying the Quality Attenuation of WiFi. In *2022 IEEE 47th Conference on Local Computer Networks (LCN).* 189–197. https://doi.org/10.1109/LCN53696.2022.9843690

[5] Sergio Leon Gaixas, Jordi Perello, Davide Careglio, Eduard Gras, Miquel Tarzan, Neil Davies, and Peter Thompson. 2016. Assuring QoS Guarantees for Heterogeneous Services in RINA Networks with ΔQ. In *IEEE International Conference on Cloud Computing Technology and Science (CloudCom).* IEEE, 584–589.

[6] Predictable Network Solutions Ltd (PNSol). 2022. http://www.pnsol.com

[7] R. Strichartz. 1993. *A Guide to Distribution Theory and Fourier Transforms.* CRC Press.

[8] P. Thompson. 2022. *TR-452.2 Quality Attenuation Measurements using Active Test Protocols.* Technical Report. The Broadband Forum.

[9] P. Thompson and R. Hernadaz. 2020. *Quality Attenuation Measurement Architecture and Requirements.* Technical Report TR-452.1. Broadband Forum. https://www.broadband-forum.org/download/TR-452.1.pdf

[10] Kishor Shridharbhai Trivedi. 2002. *Probability and statistics with reliability, queuing, and computer science applications* (2 ed.). Wiley, New York, NY, USA.

[11] P. Van Roy, N. Davies, P. Thompson, and S. H. Haeri. 2022. The ΔQSD Systems Development Paradigm, a Tutorial. In *HiPEAC Conference (High-performance Embedded Architecture and Compilation).* Tutorial.

Simple Cache

January 19, 2023

[1]:
```
import DeltaQ.Workbench
import Text.Printf
import Data.Maybe
```

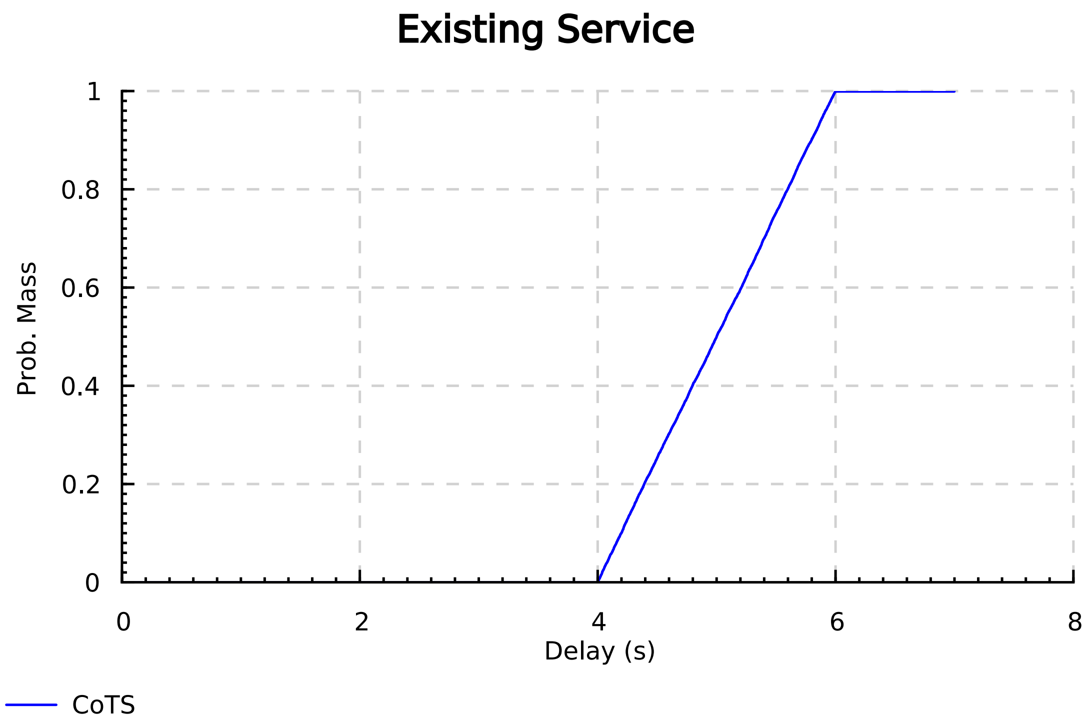
1 Simple 'Cache' performance optimisation example

Time is but an illusion - Lunch Time doubly so. Choose your scaling factor as needed

You have an existing service ('off the shelf') that takes between 4s and 6s and has an (inherent) failure ratio of 0.1% (99.9% success).

[2]:
```
x1 = (□) (999/1000) perfection bottom □ uniform 4 6 :: DeltaQ
printf "Simplified expression: %s" (show x1)
printf "CoTS reliabilty: %f" $ 1 - lossprob x1
plotDQs "Existing Service" [("CoTS", x1)]
```

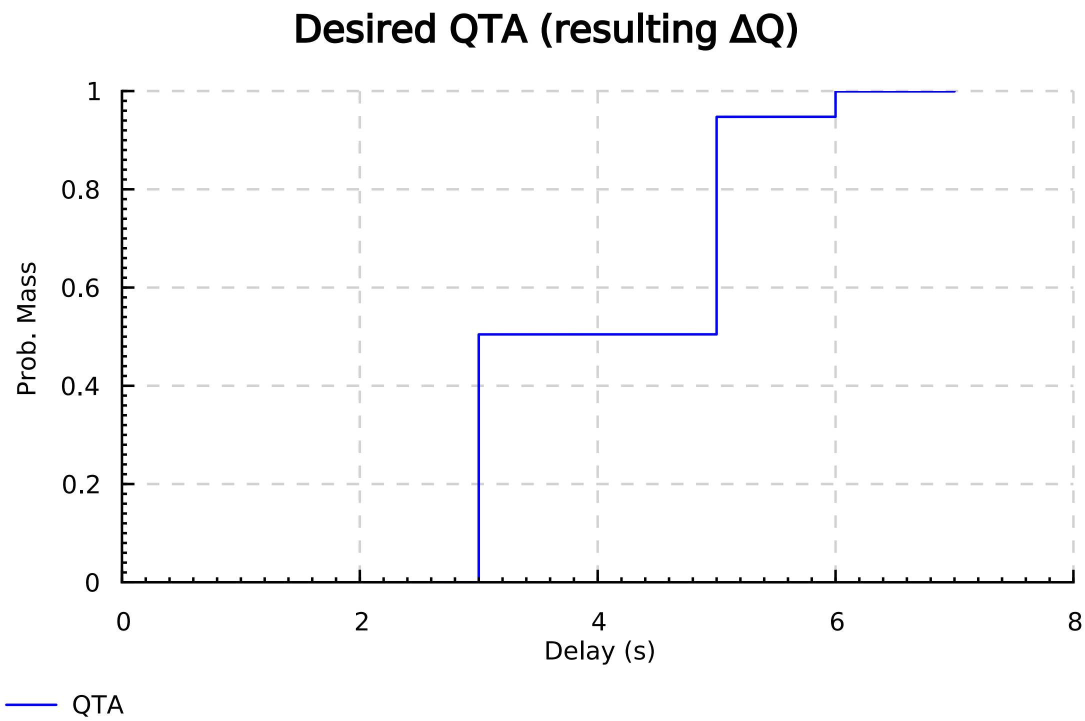
```
Simplified expression:  (999 1)    4.0   [0,2.0]

CoTS reliabilty: 0.999
```



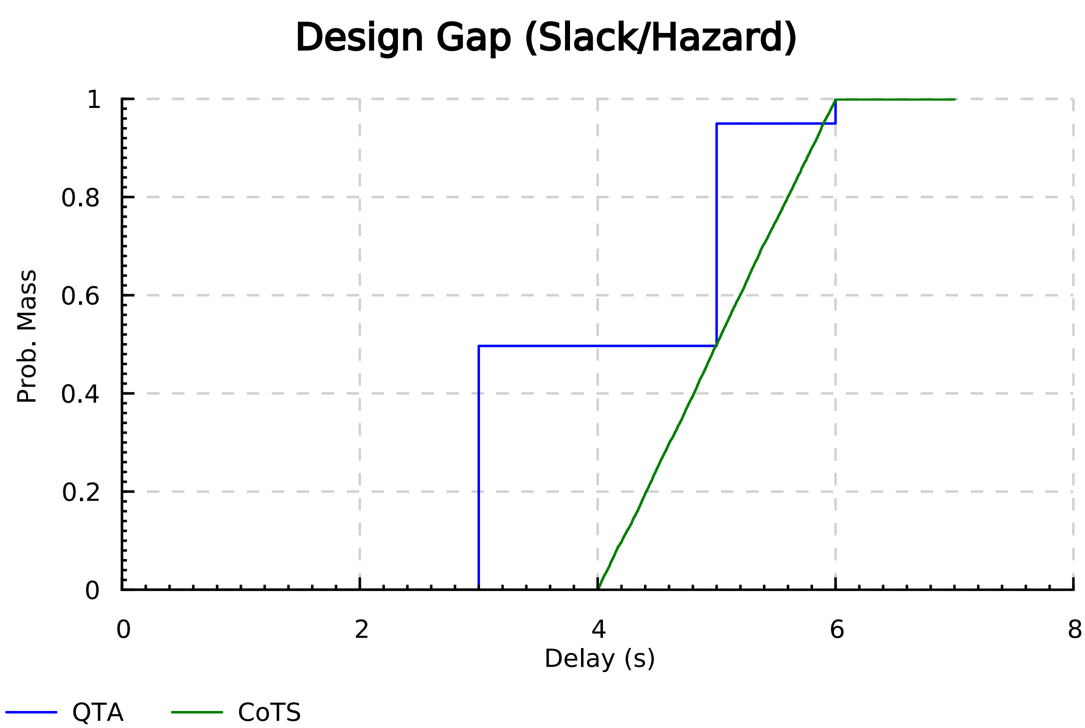
The desire is to create an optimised service that delivers 50% of its outcomes within 3s, 95% of them within 5s and 99.95% within 6s.

[4]:
```
qta1 = fromQTA [(0.5, 3), (0.95, 5), (0.9995, 6)] :: DeltaQ
printf "Minimum acceptable reliablity: %f" $ 1 - lossprob qta1
plotDQs "Desired QTA (resulting  Q)" [("QTA", qta1)]
```

```
Minimum acceptable reliablity: 0.9995
```




```
[5]: plotDQs "Design Gap (Slack/Hazard)" [("QTA", qta1), ("CoTS", x1)]
```



As can be seen, this is (almost) entirely “in hazard”. Can caching help? How much performance is needed in the cache? What sort of cache hit rates are needed?

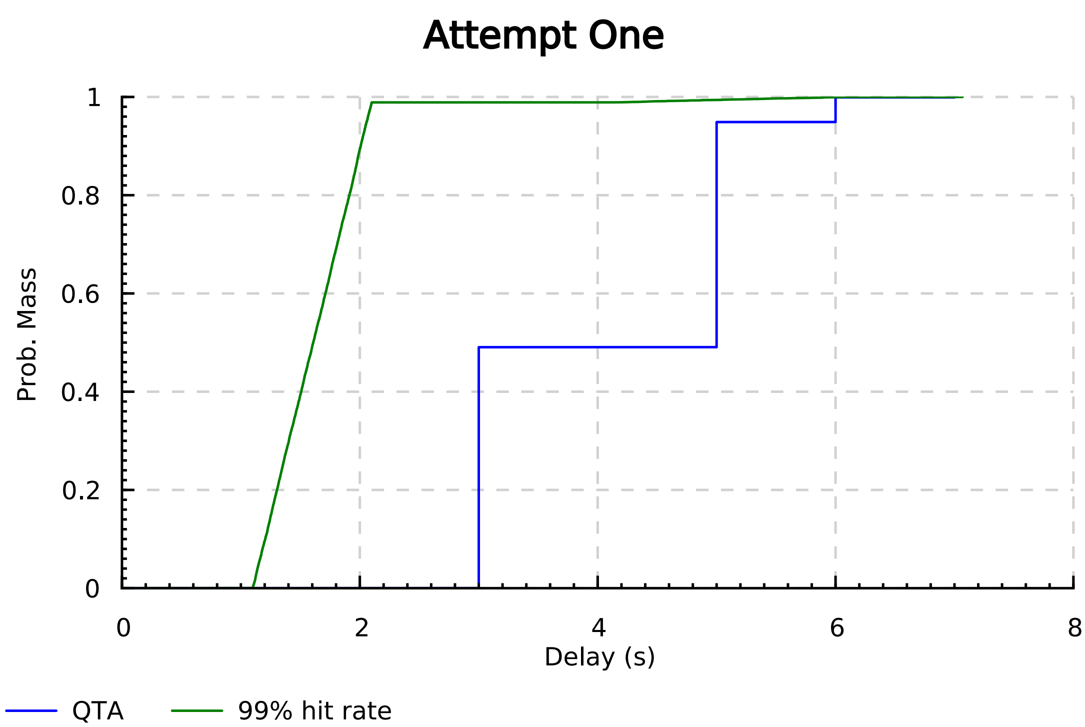
Cache lookups take time (say 0.1s), we will also assume that the CoTS system is relatively precious/expensive so we will not make concurrent lookups against it.

A Q model of the cache - takes some time, has some success probabilty then either responds from cache or falls back on the CoTS service. We are assuming that the cache has a reliablity of 99.999 (five nines)

[6]:
```
cacheModel cacheHitRate hitDq missDq = (□) 0.99999 ( 0.1 □ (□) cacheHitRate hitDq missDq) bottom
```

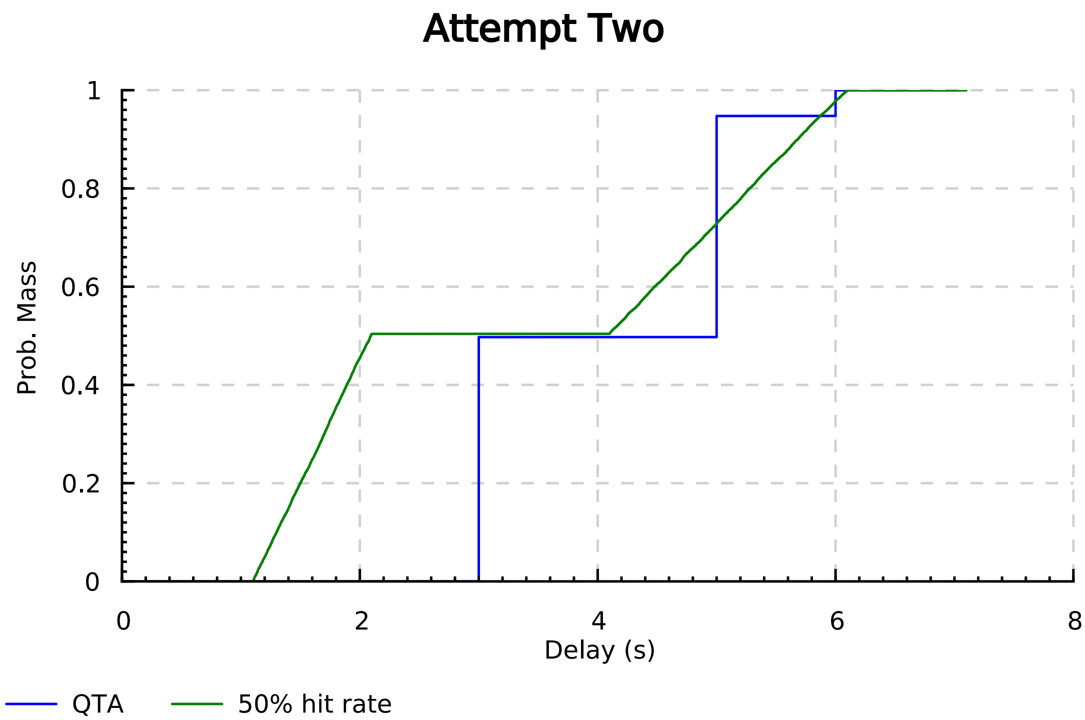
[8]:
```
try1 = cacheModel (99/100) (uniform 1 2) x1 :: DeltaQ
printf "System reliablilty: %f" $ 1 - lossprob try1
plotDQs "Attempt One" [("QTA", qta1), ("99% hit rate", try1)]
```

```
System reliablilty: 0.9999800001
```



So it works, but (for the sake of argument) that sort of cache is ‘expensive’. There is plenty of ‘slack’ in the proposed solution - could we live with a lower hit rate? Budget would suggest a 50% hit rate was more affordable.

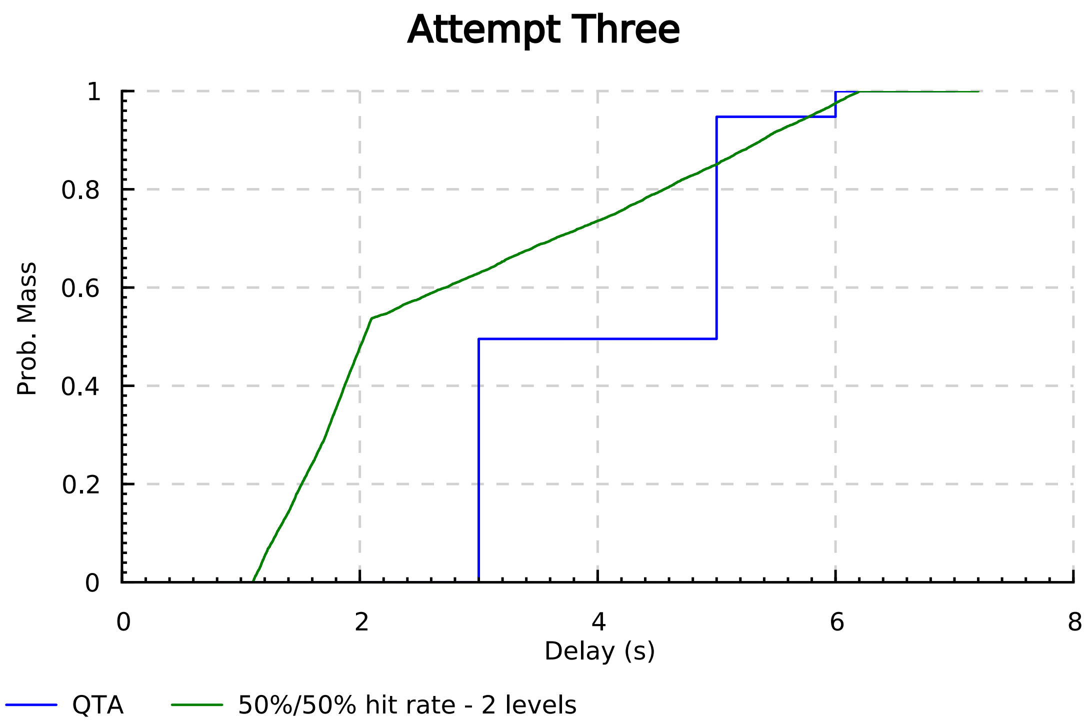
[9]:
```
try2 = cacheModel (50/100) (uniform 1 2) x1 :: DeltaQ
plotDQs "Attempt Two" [("QTA", qta1), ("50% hit rate", try2)]
```



This is looking promising - but we're not meeting the QTA (between about 5s and 6s); could we introduce a two level cache? The second level cache is slower, it takes between 1.5s and 4s to yeild a result, however it is cheaper, let's size it for 50% hit rate.

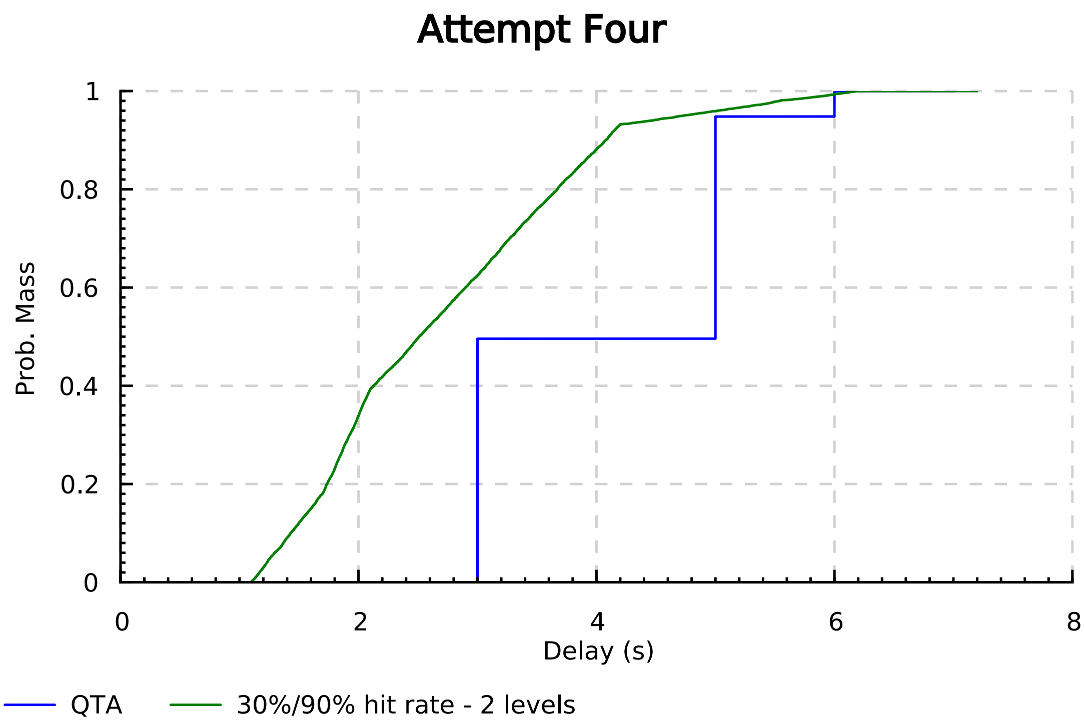
We'll assume that we sequentially check each cache before reaching out to the underlying service.

[10]:
```
try3 = cacheModel (50/100) (uniform 1 2) (cacheModel (50/100) (uniform 1.5 4)␣
↪x1)
plotDQs "Attempt Three" [("QTA", qta1), ("50%/50% hit rate - 2 levels", try3)]
```



This is nearly there. However the cost differential between the L1 and L2 cache is large, for every 1% of reduction in L1 we can have 2.5%-3/0% increase in L2 for the same price. Perhaps we can do a bit better?

[11]:
```
try4 =  cacheModel (30/100) (uniform 1 2) (cacheModel (90/100) (uniform 1.5 4)␣
 ↪x1)
plotDQs "Attempt Four" [("QTA", qta1), ("30%/90% hit rate - 2 levels", try4)]
```

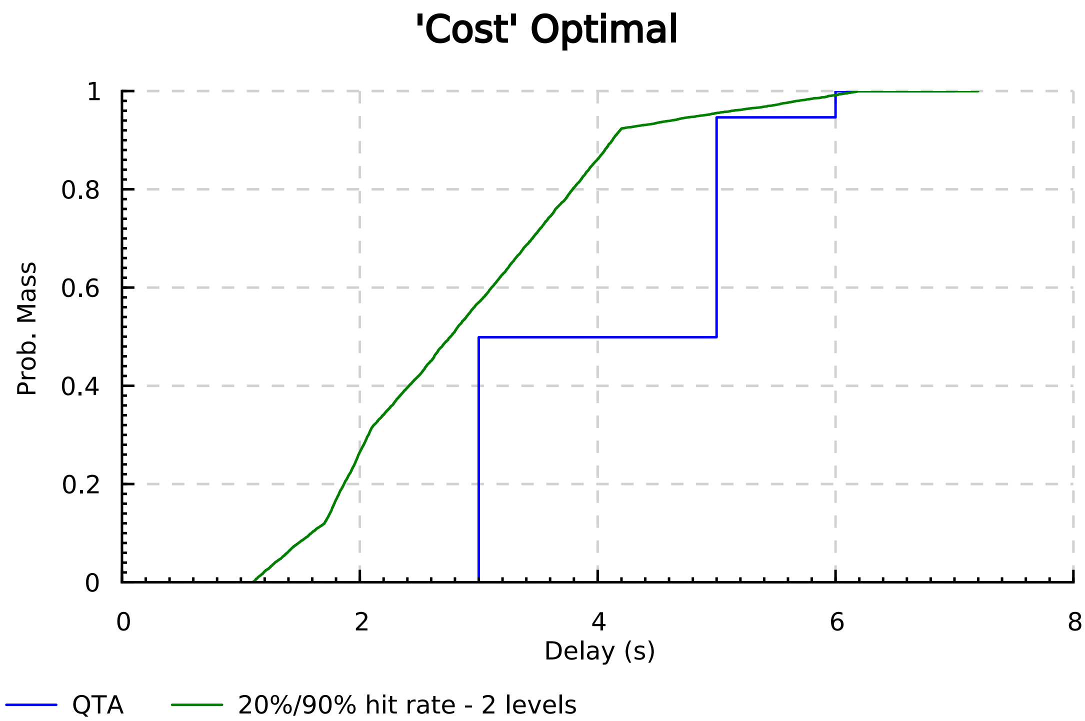


Of course, the budget is everything and anything we can do shrink that L1 cache frees up some monetary slack....

[12]:
```
try5 =  cacheModel (20/100) (uniform 1 2) (cacheModel (90/100) (uniform 1.5 4)␣
  ↪x1) :: DeltaQ
printf "System Reliability: %0.7f" $ 1 - lossprob try5
try5' =  fromJust $ mmmv try5
printf "Min/max response Time: %0.3f/%0.3f, mean/stddev: %0.3f/%0.3f" (mmmvMin␣
  ↪try5') (mmmvMax try5') (mmmvMean try5') (sqrt $ mmmvVar try5')
plotDQs "'Cost' Optimal" [("QTA", qta1), ("20%/90% hit rate - 2 levels", try5)]
```

```
System Reliability: 0.9999020
```

```
Min/max response Time: 1.100/6.200, mean/stddev: 2.860/0.647
```



We have investigated (to some level of fidelity) a design space that takes a CoTS, adds a front-end cache to create a performance (and 'cost' optimal) solution.

Not only were we able to optimise with respect to performance and cost, we were also able to ensure that the resulting system reliablity exceeded the target (*Note:* the cache component reliability has to be at least four nines - but that is just a parameter in the model).

We have done this exploration graphically, but that was just for pedagogical purposes - all the underlying properties for automation are present.