Cyberphysical Systems: A Behind-the-Scenes Foundational View

Richard Banach[1] and Wen Su[2⋆]

[1]School of Computer Science, University of Manchester,
Oxford Road, Manchester, M13 9PL, U.K.
banach@cs.man.ac.uk
[2]School of Computer Engineering and Science,
Shanghai University, Shangda Road, Shanghai, China.
wsu@shu.edu.cn

Abstract. Hybrid and cyberphysical systems pose significant challenges for formal development approaches based on pure discrete events. In this essay, after a brief look at the CPS landscape, the foundations of CPS systems are examined from the ground up, with a particular view to aspects rooted in the continuous part of the CPS spectrum. We take a journey starting from the foundations, through a number of ways of addressing the continuous mathematics aspects, to phenomena latent only in the world of physical descriptions, such as the onset of instability due to passing through bifurcation points in the problem parameter space. We argue that such phenomena, that can plague CPS design when optimising for performance metrics, can only be understood by sufficient engagement with the continuous world.

1 Introduction

In today's world of cheap processors, memory, sensors and controllers, the enthusiasm for hybrid [8] and cyberphysical [12] systems (CPS) is veritably exploding. This is increasingly fueling the cost-effectiveness of a smart-everywhere approach to services and systems. New initiatives pour forth at a seemingly ever-increasing rate, in many domains, e.g. health, transport, city infrastructure, communication etc., and their many subdomains.

The presence of control as first class citizen in these systems leads to the impingement of discrete techniques from the computing sphere on the one hand, onto a plethora of techniques from continuous mathematics and the physical systems sphere on the other, lending a highly multi-disciplinary nature to this discipline. It is fair to say, that more than in almost any other multi-disciplinary area, the fundamental role that mathematics plays in all the disciplines that impinge here, means that these disciplines can interact in a deep way, rather than merely providing a distinct view on each other, or offering complementary but still separate families of techniques.

[⋆] Wen Su was supported by the National Natural Science Foundation of China (No. 61602293), and the Science and Technology Commission of the Shanghai Municipality (No. 15YF1403900).

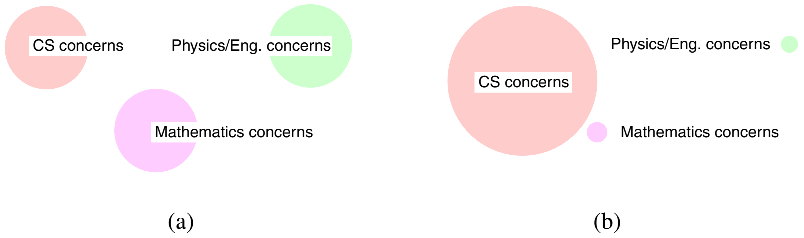


Fig. 1. CPS and computer science: (a) how it should be, with all disciplines involved exerting comparable influence; (b) how it actually is, with overwhelming weight placed on computing perspectives, paying scant regard to the other disciplines.

It is often claimed that completely new formalisms will be needed to reason about systems of this kind, a view that is a little puzzling considering that every component or aspect of such systems comes with a well understood mathematical framework that captures the predictability of its behaviour in engineering contexts. The presence of these, and the *a priori* consistency of mathematics, thus tends to suggest that the underlying mathematical dialogue has not been pursued in sufficient depth. Since Klaus-Dieter Schewe has long had a strong interest in the foundational aspects of cyberphysical systems, it is a pleasure to discuss some of these issues, from a particularly personal perspective, in this festschrift essay.

These days, most design and development of cyberphysical systems is very much rooted in the integration of, and cooperation between, existing tools and techniques from different areas of computer science and different branches of engineering and technology. Overwhelmingly, especially on the computer science side, such tools and techniques are focused on discrete descriptions of system behaviour, and usually pay scant regard to the continuous aspects of physical behaviour. Fig. 1 gives an illustration of the uneven focus.

Unsurprisingly, such approaches are frequently fraught with problems of compatibility and of unpredictable interworking. This arises from a lack of attention to the different semantic foundations of the contributing formalisms, and a lack of precision with which they view issues which are fundamentally continuous. Regarding the latter, frequently, the formalisms in question are unable to speak at all, or can say very little. Since continuous phenomena can display extraordinary subtlety, such a dislocation is evidently undesirable.

In this essay, we look at these issues from the perspective of rigorous model-based system development and verification, but taking a keener interest in the more problematic areas rooted in the continuous world. We will find that we can point to many things which, although perplexing from a conventional discrete/computational perspective, become much clearer when enough notice is taken of what continuous mathematics can tell us. We infer that if we are suitably cognisant of the insights available from *all* the disciplines that contribute to CPS, then most of the foundational problems for CPS melt away, even if the practical problems of constructing large real-world systems both optimally and verifiably, assuredly do not.

The remaining sections of the paper are as follows. In section 2 we briefly survey the most visible features of the CPS world as perceived within the research community. After a few comments, in Section 3 we start from the foundations, reviewing the elements that underpin the mathematical foundations of the theories that contribute to CPS descriptions. The foundational view is important, since a consistent picture must operate across *all* the contributing disciplines, and must connect with the world of discrete mathematics that operates in the computing sphere. This journey through a number of mathematical subdisciplines culminates in Section 4 with the prospects for mechanically supported verification, based on Collins's groundbreaking Cylindrical Algebraic Decomposition, and the possibilities for adapting non-semialgebraic descriptions by using suitable approximations. In Section 5 we use this basis to review a number of phenomena rooted in the continuous world, whose implications are less obvious from a purely discrete perspective. They include: differential-algebraic equations, control issues such as stability and the effects of multi-system descriptions, technical issues in control, delay differential equations, and bifurcations. The section continues by discussing numerical approaches and sampling and quantization issues. Section 6 summarises and concludes.

2 The CPS Landscape

Nowadays, computing devices get ever smaller, more distributed and interconnected, both to each other, and to the physical environment. This enables the construction of systems with a bewildering variety of architectures, required performance characteristics, and interplay with the real world. A very major role is played by simulation in the design of such cyberphysical systems, with popular software suites like SIMULINK [22], 20-sim [2], Modelica [24], much to the fore. Simulation and experimentation are certainly the most appealing ways to realise such systems, since they are so accessible and easily usable, with a relatively modest investment in preparation.

Somewhat more rigorous than pure simulation and experimentation are approaches based on the control aspects of the CPS system. A large literature has grown up around the exploration of appropriate stable control regimes for particular CPS configuration styles and application regimes. Most of this work appears within the wider control systems literature.

As ever though, simulation and experimentation in principle cannot achieve the level of assurance that verification can give (provided, of course, that the models being verified can justify the faith placed in them). Here, the self-evident undecidability of any language expressive enough to describe an interesting set of CPS systems impinges directly on what can be verified and how. The hybrid automaton paradigm (qualified in various ways, as needed) is the default descriptive mechanism in this space. In [8] there is a good survey of well established systems for this, and overwhelmingly, these tend to focus on linear behaviour, because of the tractability of the fragments of arithmetic that are involved.

Another approach takes the analytical descriptions of non-trivial hybrid and cyberphysical systems at face value, and, reflecting centuries-old practice in applied mathematics and physics, engages with them symbolically. The aim is to formalise and to mechanise what can be done via such techniques. Among these approaches we can cite Hybrid CSP [19,32], KeYmaera [26,1] and Hybrid Event-B [5,6].

Invariably, the above sketch is an oversimplification, and there are a large number of variations on these themes to be found in the literature, e.g. [28]. We will have more to say about the connections between some of these styles of approach below.

3 Starting from the Foundations

Cyberphysical systems, by their very name, involve physics (and thus its practical application, engineering). Immediately this implies the involvement of mathematics. They also involve computing, and this too implies the involvement of mathematics. Ideally, all three contributing disciplines, namely physics, mathematics and computing, would play an equally significant role in the development of the subject. However, what is overwhelmingly seen is a very heavy emphasis on the computing aspects, as shown in Fig. 1. The texts [18,3] rather bear this out.

Interestingly, the two mentions of mathematics above refer to very different areas of the subject. In the physical sciences, the mathematics is predominantly continuous, dealing with real valued quantities changing according to physical laws, often expressed via differential equations. Extremely rapid variation in these quantities is idealised as impulsive change, resulting in discontinuities in the real valued behaviour. In the computing sphere, the mathematics is overwhelmingly discrete, with instantaneous change of state being the normal paradigm. The discrete mathematics is overwhelmingly concerned with properties and behaviours expressible via discrete, very often finite, sets.

The discrete sets consist of elements that have no internal structure, and usually, few relationships between them. This contrasts with the world of reals, in which, although the reals are also treated as having no internal structure, we have an enormously rich selection of properties at the disposal of the utiliser, this being due to the fruits of the mathematical analysis that has been created over the last couple of centuries.

3.1 Insights from the Contributing Disciplines

Accepting the broad sweep of issues just mentioned, brings a number of points into prominence:

- Cyberphysical systems are concerned with physical quantities. In physics all quantities are functions of time t, and time is real. Time is also not manipulable (in classical physics). Thus, physics deals with the way that various quantities change over time, but time itself is not one of them.[1]

[1] Formally, this means that time is read-only, and that physical quantities must be defined for all applicable time points (if one interprets normal physical discourse from a formal point of

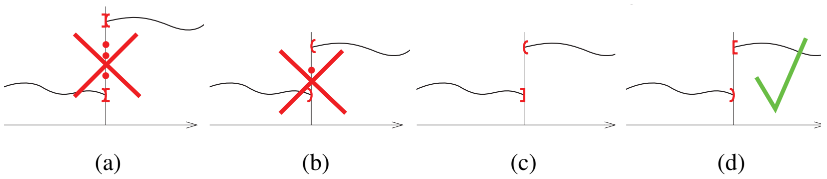


Fig. 2. Piecewise continuous functions of time, and intervals: (a) multivalued functions of time are simply unphysical; (b) individual values at isolated times have no physical significance; (c) left-open right-closed intervals compose nicely, but yield executions without a definite starting point (since there is no earliest point of a left-open interval); (c) left-closed right-open intervals compose nicely and yield a definite starting point.

- Turning to the logical/foundational aspects of the reals, we realise that any real-valued expression, once its parameters are fixed, can only refer to an isolated real number: there is never a ‘next real’, nor indeed a ‘previous real’.
- The computing world re-emphasises the need for discontinuous change. Together with the previous point, a CPS formalism must thus be capable of expressing isolated discontinuous changes in value.
- Typical engineering and physical models require the use of differential equations. So any CPS formalism needs to encompass those.
- Moreover, the normal calculational problem solving techniques used in applied mathematics need to work, otherwise any putative formalism would struggle to achieve anything useful.

3.2 Consequences

We regard the preceding observations as a kind of requirements list that sets out some conditions that a CPS formalism must meet, and we now examine where this leads.[2]

We start with discontinuities. If discontinuities are isolated, then in between, functions of time (that describe values of variables) must be continuous. So we are dealing with piecewise continuous functions, which must therefore be continuous on intervals that can potentially be open or closed at either end. Additionally, in formalisms like the duration calculus [33], multiple state transitions are allowed to take place at a single moment of time. The latter would lead to state functions on intervals that abut at points where the function may be multi-valued — we can dismiss such functions immediately as being unphysical.[3]

view). Moreover, formally, each physical quantity is identified with a particular free variable, because physical discourse is typically much more open-ended than typical formal theories allow.

[2] Many existing CPS formalisms deal with these issues in various *ad hoc* ways. In this esay we take their mathematical consequences at face value, and see where this takes us.

[3] We are not saying that the duration calculus is not useful, merely that it is not useful for the task at hand.

If a right-open interval is followed by a left-open one, then there is a single real value in the middle at which a function that is continuous on those intervals may be defined differently. Such individual point values also have no physical significance since only the integrals of functions over regions (whether large or small) can have an impact physically. So we can disregard this configuration of intervals.

So we are left with left-closed right-open intervals, or alternatively left-open right-closed intervals. Both kinds abut nicely with others of the same kind, making bigger ones. However, left-open right-closed intervals raise an exception at the initial time of an execution, since there is ‘no earliest moment’ of the execution, and a single point does not define a left-open right-closed interval. Thus, left-closed right-open intervals, permitting a definite starting point for an execution, which can then continue over a succession of such intervals, emerge as the winning candidate. Fig. 2 summarises this line of argument.

Next, differential equations (ODEs). Immediately, the discontinuities create a technical issue. If we can anticipate in advance where the discontinuities in functions occur, we can arrange for them to fall exactly at the boundaries of our left-closed right-open intervals. But *if we cannot*, then the right hand side of an ODE may contain discontinuous functions, and it itself may be discontinuous, making the derivative on the left hand side badly defined at such a point.

Here, the instinct of pure mathematicians to imaginatively generalise previously established notions comes to our aid. These days, ODEs are studied assuming their right hand sides are *measurable* over time. Isolated points of discontinuity do not spoil this property, allowing us to use this, the Carathéodory interpretation of ODEs [29], for cases where there might be unanticipated discontinuities on the right.

Of course, if there are no unanticipated discontinuities, we do not need the additional sophistication. However we *do* need a criterion like Lipschitz continuity of the right hand side of an ODE [29], to avoid cases like the non-Lipschitz $\mathcal{D}\,x = x^2 + 1$, whose solution $x(t) = \tan(t)$ explodes at $t = \pi/2$.

Differentiability goes along with *absolute* continuity (rather than unqualified continuity) [31,27]. A function f is absolutely continuous (AC), iff the fundamental theorem of calculus works, iff an increment of f (over an interval) is the integral of its derivative (which exists almost everywhere, over the same interval). This, in tandem with the preceding observations, suggests the world of piecewise AC real functions as a suitable semantic universe for grounding the semantics of CPS formalisms.

As a diversion, Fig. 3 shows a non-absolutely continuous function, the famous Cantor ternary function. It is defined to be flat (with value $\frac{1}{2}$) on the middle third of the unit interval, to be flat (with values $\frac{1}{4}$ and $\frac{3}{4}$) on the middle thirds of the previous outer thirds . . . and so on recursively. All the middle thirds add up to length 1, yet there are enough points left over in the ‘Cantor dust’ that remains to have the cardinality of the entire real line. Though continuous, and flat almost everywhere, this function cannot be the integral of its derivative.

This grounding in a class of functions of time, specifically the piecewise AC real functions of time, permits viewing the two kinds of update needed in CPS systems (discrete and continuous) from a remarkably similar perspective. A discrete update is a pair of valuations of the variables of the system (the before-valuation and the after-

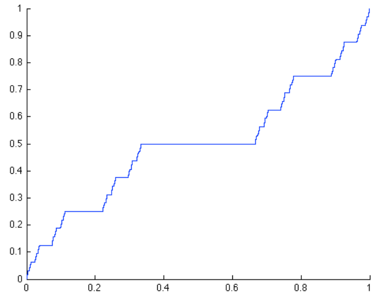


Fig. 3. The Cantor ternary function, a function which is continuous but not absolutely continuous. On all the 'middle thirds' (whose total length is 1) it is flat, so its derivative is zero almost everywhere. Yet it increases in value from 0 to 1, so it is not the integral of its derivative.

valuation) pinned to a particular moment in time (the time of the discontinuity). A continuous update is a time indexed family of pairs of valuations of the variables, with the before-valuation being the valuation at the start of the continuous behaviour and the after-valuation being the valuation at any choice of time subsequently. Viewing time as merely indexing, makes it the only aspect that is different.

Although piecewise AC real functions solve the semantic foundations issue, they permit behaviours that are extremely poorly behaved, if judged by the standards of day-to-day applied mathematics calculations. As an example, consider the function $f(t) = \exp[-\frac{1}{t^2}]$. It is zero at $t = 0$, and so flat there that all its derivatives at $t = 0$ are also zero. This means that the Taylor series derived from these derivatives defines the zero function, quite different from $f(t)$. This implies that Taylor series in general are unreliable when one is dealing with real functions.

We are rescued by a quotation from P. A. M. Dirac: "*A number theory is beautiful, but the complex number theory is more beautiful.*"[4] Its message is that although there are many kinds of 'numbers' explored within mathematics, when we consider complex numbers specifically, and in particular complex analytic functions, a vast array of uniquely powerful properties suddenly burst into vivid relief [17,11]. Here are a few.

Firstly, Taylor's theorem works. A function which is complex differentiable is automatically complex analytic, i.e. Taylor's theorem defines it uniquely. Secondly, there are few awkward issues to worry about: poles of finite order (e.g. $1/(z-a)^k$ for integral k); branch points (e.g. $1/(z-a)^k$ for fractional k); essential singularities. Thirdly, we have unique analytic continuation: knowing complex analytic $f(z)$ precisely in any region, no matter how small, determines $f(z)$ everywhere where it is defined — an incredibly powerful result for practical day-to-day calculations. Fourthly, the authors' own particular favourite, Picard's Great Theorem: *Every analytic function assumes every complex value, with possibly one exception, infinitely often in any neighborhood of an essential singularity.*

[4] Note the exquisite use of the indefinite and definite articles in this quotation. Unfortunately we are not aware of the original source of the quotation.

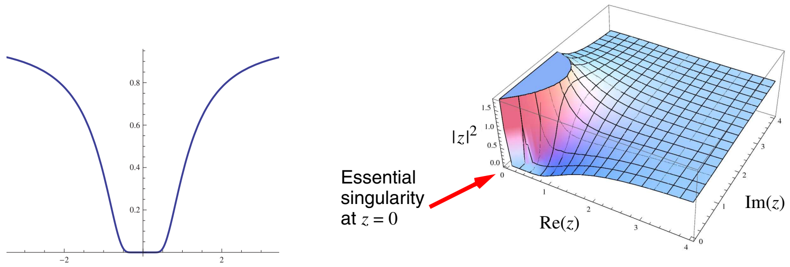


Fig. 4. The real function $f(t) = \exp[-\frac{1}{t^2}]$ on the left. On the right, the magnitude squared of the complex function $f(z) = \exp[-\frac{1}{z^2}]$ which extends it. Picard's Great Theorem implies that $f(z)$ cannot be depicted close to the origin, because of the essential singularity there.

Viewing t as the real part of complex z in our above example, $z = 0$ is an essential singularity of the function $\exp[-\frac{1}{z^2}]$, and this is the source of its wierd behaviour *vis à vis* Taylor's theorem. If we restrict our view of this function to just the real axis, we see nothing untoward, aside from the unnerving degree of flatness; but look one iota away from the real axis in any neighbourhood of the origin, and it's bedlam. Fig. 4 shows both the behaviour on the real line, and the complex behavour away from the real line. Thus, in general, in order to recover good calculational properties, we need to restrict the piecewise AC real functions, to those which arise as piecewise complex analytic functions which are real on the real line.

4 Proof and Verification

Complex analytic functions (even if we just focus on the real part) are fine, but to get the best out of them often requires human ingenuity. This provides food for applied mathematics enterprises in universities the world over. In the computer science world, the desire for automation dictates that we prefer to invest human ingenuity in deriving powerful algorithms that solve broad classes of problems automatically, rather than focusing on individual problems. This sets up a tension between the applied mathematics and computer science perspectives — a tension between *calculation* and *proof*. Let us look briefly at a couple of examples.

Take the case of Propositional Logic (PL). In this case, there are only Boolean values. There is essentially *no* calculation, all manipulation being equivalent to proof. By the time we move to First Order Logic (FOL), calculation starts to play a role: there are constant and function symbols and their interpretations, and expressions formed from them denoting values. Yet handling these is still very generic: Hintikka sets, Herbrand universes and the like, and they lead to the known generic semi-decidability results, etc. Moving on to languages that are expressive enough to plausibly represent CPS systems, generic model theoretic and proof theoretic techniques have long taken a back seat. Whether one can solve a particular CPS system, or prove some property of it, depends almost entirely on the specific constants and function symbols (and on their standard interpretations) that occur in it, and what one knows about them.

Thus, at low levels of formal language expressivity, inference and decidability form the focus, and, to the extent that particular formal languages allow, decidable language fragments are typically defined in terms of connectives that occur, numbers of variables, permitted numbers and alternations of quantifiers, etc. At high levels of expressivity, inference is determined by in-depth investigation of special cases, and typical 'decidable language fragments' (although the phrase is seldom used in these contexts) are often defined by parameterisations of these special cases, making their scope conceptually much narrower than is usual in mathematical logic.

In the traditional applied mathematics sphere, aspects that would usually be attributed to 'logic', were, in the old days, simply done by hand in the meta-level discourse. These days, as automation increases the size and complexity of problems that are tackled, there is a benefit in using automation to manage such aspects, error-prone as they can be. They include: case analysis, completeness of coverage, bound variable scopes, Skolem constant management, SMT-like calls to calculational oracles.

Referring back to the foundations of the CPS world, we must confront the vast gap between the plain set theory of discrete state change on the one hand, and on the other, sophisticated phenomena like essential singularities (that must be avoided if we are to have any hope of calculating anything). Of course the route from simple set theory, via naturals, integers, rationals, reals to complexes, is well known and can be formalised in various ways. The authors have come across a number of such endeavours over the years. When it is attempted for real, it always goes the same way, described as follows.

4.1 Formalising from the Foundations

At the start there is great enthusiasm (the work, if funded, has just been approved). Much enthusiastic hacking of foundations takes place. There are many interleavings of quantifiers to deal with, but morale is high, and progress is made. Work continues, and after a year or two the foothills of applied mathematics slowly start to become visible. As a result of all the valiant struggles with the foundations in the early days of the project, by now, all the foundational issues have been surmounted, and what remains is just hard work.

At this point it is legitimate to ask just how much hard work is at stake. The NIST Handbook of Mathematical Functions [25], a standard bible of results for theoretical physicists, applied mathematicians and their ilk, amounts to almost 1000 pages. The typical foundational effort that has just been described seldom covers more than 50 pages of [25]. Even accepting that 500 or so of those 1000 pages might be regarded as somewhat esoteric for everyday applications, the achievements of the typical formalisation of applied mathematics do not amount to a serious resource for general purpose applied mathematics problem solving.

Going back to our typical foundational endeavour, by the time the point described has been reached, there being no further foundational issues to chew on, the earlier enthusiasm of the foundations enthusiasts has become severely depleted. The endeavour quietly dies.

Of course, there are tools that confront the needs of applied mathematics, as depicted in [25], head on. They are the computer algebra tools such as Mathematica [21], Maple [20], MATLAB [22], etc. Typically, they work at a much higher level of abstraction than the foundations-led outline above. In effect they encode the cases that can be solved, and put great effort into powerful pattern matching routines, so that solvable cases can be discerned as often as possible. The commercial basis of most such tools enables the large amount of work indicated above to be undertaken in a uniformly compatible way, and they are used extensively in real engineering design. These days, despite not starting from the kind of foundations we discussed, the residual risk in the mathematical core of established tools of this kind is vanishingly small, compared with other risks in the design processes during which they are used, in the construction of the systems we rely on every day.

To summarise the above, given some fixed, specific result in mathematics, the complexity of proving it increases dramatically as one descends into increasingly deep foundations in order to insist on basing the proof at that level of axiomatisation. There is a rather splendid parable about this, which the reader may enjoy, following paragraph 7011 in Andrews's admirable book [4].

4.2 Towards Verification

In the computing world, the added value that a formal approach brings (above and beyond the capabilities of conventional development), amounts to the ability to prove properties of a systems model, in a mechanically verified way. The properties in question are predominantly safety properties. In the context of safety properties, choice is always interpreted demonically, since safety demands adequate behaviour under all possible circumstances. However, in the CPS world, control problems are very much to the fore, and in the control world, the *controllability* property is key. Roughly speaking it says: *for every permitted way of setting up the system and system goal,* there exists *a control input that steers the system to the goal*. The emphasis on existential quantification is deliberate, it tends to imply angelic choice, and is unavoidable.[5] However, as often happens when there is an assumed progress property and an angelic property contingent on it, the latter can be wrapped into a safety property (containing the existential quantification) of the assumed progress property, permitting a focus on safety properties in verification after all. (This typically happens for data refinement properties.) The inescapable progress of time, outside the control of the human user, provides a very useful progress property in the context of CPS systems.

A major contributor to the possibility of verification in the CPS arena is Collins's groundbreaking *Cylindrical Algebraic Decomposition* (CAD) [9,7]. This is concerned with semi-algebraic constraints, which are quantified Boolean combinations of formulae of the form $P(x_1 \ldots x_n) \geq 0$, where P is a polynomial expression with real (in practice rational) coefficients.[6] It was Tarski who observed that the disposition of real roots of a real polynomial could be discerned using an extrapolation of classical techniques: Sturm sequences, polynomial GCD calculations, and other results, that used only the coefficients of the polynomial, and which only needed to be manipulated

[5] No conveyance, be it a plane, car, or other vehicle that allows its driver to determine its travel parameters and destination, can prevent its driver crashing it, if the driver so chooses.

[6] Thus, polynomial equations and proper inequalities are subsumed.

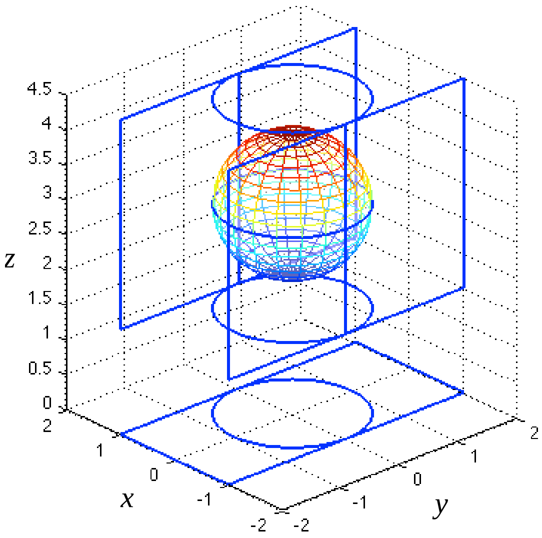
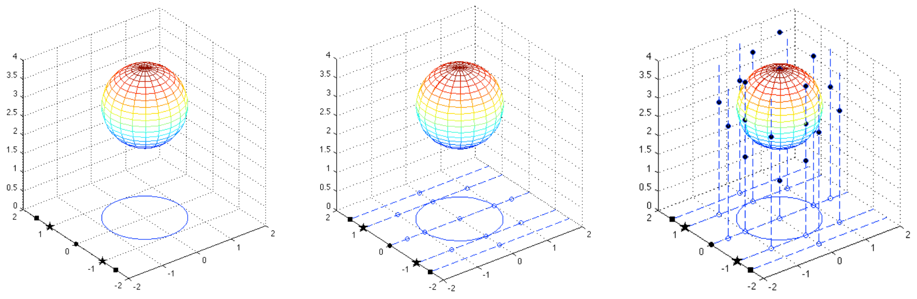


Fig. 5. Three dimensional space, recursively divided into cylindrical regions appropriate to the CAD description of a single sphere of unit radius. Projecting on z produces a circle in a plane; then projecting on y gives an interval bounded by two points, within a line. The two points define two lines in the plane, and these then define two planes in three dimensional space. In general, the decomposition of an $n-1$ dimensional region gives rise to cylinders in n dimensions when the projection is reversed.

symbolically. This led eventually to a decision procedure for semialgebraic constraints that had non-elementary complexity. Collins's great achievement was to reduce this to doubly exponential complexity in the number of variables — making it practical for a moderate number of variables. Collins's result came about by paying close attention to detail, and systematically organising the elimination of variables one at a time until the single variable techniques could be applied, after which the eliminated variables are reintroduced and characterised, again one at a time. We illustrate this fascinating process in Figs. 5, 6, 7.

We take the often used single sphere of unit radius as a running example, specifically, its interior, given by the inequality $P(x, y, z) < 1$ (the negation of the obvious improper inequality), where $P(x, y, z) = x^2 + y^2 + (z-3)^2$. Fig. 5 overviews the process. In the projection phase of CAD, a rather complicated procedure, but one that is nevertheless symbolically computable in terms of semi-algebraic constraints, implicitly identifies the regions of the (x, y) plane where the disposition of the roots of $P(x, y, z) = 0$ (when viewed as a polynomial in z with coefficients in (x, y)) is invariant. In practice this defines the circle $x^2 + y^2 = 1$. This is repeated to project out y, leaving $x^2 = 1$. This is now solved giving $x = \pm 1$.[7]

[7] In general, algebraic numbers, also symbolically computable, are needed to find the roots of an arbitrary real polynomial.



(a) (b) (c)

Fig. 6. Illustrating the CAD extension process. Witness points identifying: the region $x < -1$, the point $x = -1$, the region $-1 < x < 1$, the point $x = 1$, the region $1 < x$ are all shown in (a). In (b), this is extended to the (x, y) plane, identifying regions and points of the plane where the projected constraints have invariant truth value. In (c) this is extended to three dimensional space, systematically dividing it into regions in which the original family of constraints (i.e. $x^2 + y^2 + (z-3)^2 < 1$) have constant truth value.

The CAD process now moves into the expansion phase, illustrated in Fig. 6. The solution points for x identify regions of the x axis and their boundary points. Witness points are chosen in the interiors of the intervals (squares and dot in Fig. 6.(a)); the boundary points are also highlighted (stars in Fig. 6.(a)). These values are substituted into the previously derived semi-algebraic constraints in the (x, y) plane giving lines parallel to the y axis. These lines are subdivided into intervals (with their boundary points) according to how they intersect the solution set of the (x, y) plane's semi-algebraic constraints — and witness points are again found (the small circles in the plane $z = 0$ in Fig. 6.(b)). The procedure is repeated: the (x, y) plane witness point values are substituted into the preceding set of semi-algebraic constraints, giving lines parallel to the z axis. Again, witness points are chosen on these lines in the same manner. This gives the dots depicted in three dimensions in Fig. 6.(c). The essential fact throughout this procedure is that the intervals inside which the witness points are chosen have the property that the truth values of the whole family of semi-algebraic constraints (at the requisite level of projection) are invariant within the region containing the witness point. So it is sufficient to evaluate the constraints at each witness point, and to make suitable logical combinations of the answers, in order to know the truth value of the whole family throughout the whole region.

The procedure as a whole builds a tree, shown for our example in Fig. 7. For instance, the middle quintuplet in the bottom row of Fig. 7 (labelled (x, y, z)) corresponds to the vertical line through $x = y = 0$ in Fig. 6.(c), and represents, respectively: the semi-infinite interval below the sphere, the intersection of the line with the lower hemisphere, the interior of the sphere, the intersection of the line with the upper

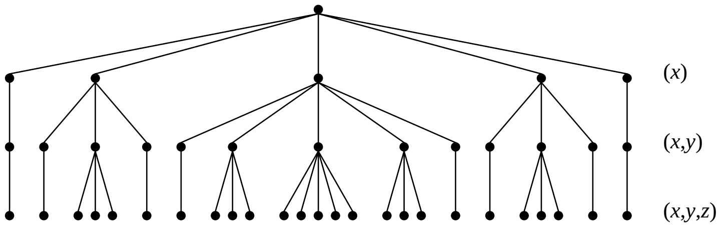


Fig. 7. The tree of witness points for the extension phase of the CAD procedure in the three dimensional sphere example. Descending one level in the tree corresponds to one 'unprojection' in the projection phase.

hemisphere, the semi-infinite interval above the sphere. It is clear that the decomposition along the variable axes generates considerable complexity —after all, there are only three regions of interest in our example: the interior, exterior, and surface of the sphere— yet the tree in Fig. 7 has many nodes. Nevertheless, the procedure is regular and scalable (modulo feasibility considerations), making it widely applicable in practice.

Unfortunately, a decision procedure for semialgebraic constraints does not directly solve the problem of automation of reasoning in CPS systems because, aside from a few rare exceptions, the solutions of CPS systems are not (made out of) polynomials. However, all is not lost. Consider a solution to a CPS system of the form $f(t) = \exp[-\lambda t]$, for which the safety invariant $t \geq 0 \Rightarrow f(t) \leq A$ needs to be established. As a Taylor series, $f(t) = \sum_{k=0}^{\infty}(-\lambda t)^k/k!$. Suppose we did not know that $f(t) = \exp[-\lambda t]$ was monotonically decreasing (making it sufficient to check the value at $t = 0$ to establish the safety invariant). For N big enough, successive terms of the Taylor series are monotonically decreasing in magnitude and alternating in sign. Thus, if N is big enough and odd, successive pairs of terms make a net negative contribution to f so that we can write $f(t) = \mathrm{poly}(t) - |\mathrm{corr}(t)|$, and discarding the correction $\mathrm{corr}(t)$ gives a safe overestimate for f via the polnomial $\mathrm{poly}(t)$. Such an approach allows the safety invariant to be proved within an interval using CAD, if all goes well. (Note that this depends on the safety invariant being given by an inequality, rather than requiring something more precise.)

The example just given was discussed in detail because the general principles apply widely. Useful solutions to (the continuous portions of) CPS systems are asymptotically stable, which means they decrease in magnitude over time. This also means they are given by series whose terms decrease in magnitude and alternate in sign (perhaps in groups). So a technique of judicious grouping of terms can manipulate the series into a polynomial plus a correction that acts in favour of the safety inequality that needs to be established. The MetiTarski tool [23] embodies ideas of this kind.

5 Gremlins Rooted in the Foundations

The preceding sections took us on a journey that spanned the chasm between the simple discrete set theory of familiar computational state change, and a number of formulations in the continuous world that emphasised existence of semantics, or calculational utility, or potential for automation. Any foundational approach to CPS needs to be able to speak somehow to all these issues, whether explicitly or implicitly.

In the computing world, there is a tendency to the view that once a formal framework has been set up, then 'the rest is programming'. This is largely a consequence of the fact that in the world of discrete set theory and bits, the lack of structure leads to a dearth of generic results. If an arrangement of sets and bits leads to one set of circumstances, it is usually not too hard to find another arrangement of sets and bits which leads to the exact opposite circumstances. In the continuous world this is not the case. The subtle constructions that lead from the discrete foundations to the rich continuous world that we have indicated lead to various non-trivial phenomena that apply to broadly applicable classes of system. In this section we consider a number of topics that are connected with problem solving for CPS systems that arise in this way.

5.1 The Influence of Control Problems

The nature of CPS systems, with their interplay between the physical world and computer control, means that control systems *per se* form a large part of the remit of CPS systems. Control theory has been intensively studied since early in the 20th century, and is by now a large and mature discipline. The point of this brief paragraph is to flag up that while many of the topics in this section are not necessarily directly couched in control terms, their relevance to CPS systems is inevitably because of their impact on control systems.

5.2 Differential-Algebraic Equations

Most formal systems for CPS tend to be monolithic, meaning that the whole of the system under consideration is included in a single model, which is then subjected to whatever analysis is envisaged. There is a good reason for this of course. Having a single model makes the maximum information available to the analysis, maximising its potential power. We illustrate this more concretely below.

However, non-trivial practical systems are made of collections of separate components. This means their global properties emerge indirectly. One very mundane aspect of this is that the separate components of the system are directly connected to each other. Consequently, where a CPS formalism handles the continuous world exclusively via ODEs, this can make for some awkwardness, because the relations expressing direct connections are algebraic; e.g. the voltage and current across an electrical connection between two components are equal, rather than being related via an ODE. Thus the continuous world of complex CPS systems is best described by systems of differential-algebraic equations (DAEs) [16], which combine algebraic relations with ODEs, rather than by just using ODEs.

DAEs evidently offer more possibilities than ODEs alone. Clearly, taking at face value arbitrary algebraic relations involving variables and their derivatives as a specification mechanism forces the adoption of a purely simulation/numerical approach, since there is no hope of an analytic solution in such a setup. This prevents proof of properties in the usual model based way. Restricting to linear DAE specifications gives greater leeway for proof based approaches, but still permits a much wider range of behaviours than we see with just linear ODEs. We illustrate what can happen with some very simple examples. In the following, x, y are state variables, f_1, f_2 are inhomogeneous terms (all of these being potentially time dependent), and $||$ is simultaneous definition. We will assume that there are no other constraints on the state variables than the ones we write.

Among the possibilities that may arise, we have the following: inconsistency, e.g., $x := y || y := x+1$; unique solution with forced initial value, e.g., $x := 1$; unique solution with arbitrary initial value, e.g., $\mathcal{D}\, x = 1$; solution with constrained initial values and given by an unspecified arbitrary function, e.g., $x := y+1$; solution with arbitrary initial values and given by an unspecified arbitrary function, e.g., $\mathcal{D}\, x = y$; solution with forced initial values and involving constrained inhomogeneous functions, e.g., $\mathcal{D}\, x = f_1 || x := f_2$.

In the case of linear systems with constant coefficients (such as all our examples) the Kroneker canonical form of the so-called matrix pencil of coefficients of the system covers all the possibilities that arise. DAE systems are sufficiently complicated that even just relaxing the constant coefficients constraint to allow the coefficients to vary over time is sufficient to materially alter the collection of possibilities available for linear systems.

5.3 Stability Considerations, Multiple Machines

As mentioned above, there is a visible tension between the compelling verification impulse towards monolithic descriptions, yielding global information and maximum power for inference on the one hand, and on the other hand, the pragmatic engineering impulse towards partitioned descriptions, yielding the maximum potential for separate working and thus (optimistically) shorter time to market, but permitting reduced information in the context of each individual component. We give a small illustrative example.

Suppose we have an integrated system containing two variables $x(t)$ and $y(t)$ subject to the dynamical equations $\mathcal{D}\, x = y$ and $\mathcal{D}\, y = -x$. In the context of global information, the solution of this system is familiar: $x(t) = \sin(t)$ and $y(t) = \cos(t)$. These behaviours for $x(t)$ and $y(t)$ are bounded and are (marginally) stable, realising the bounds $|x(t)| \leq 1$ and $|y(t)| \leq 1$.

Now, in the interests of separate working, suppose we are obliged to put $x(t)$ and $y(t)$ into separate constructs, with only partial information available to each about the other. Specifically, let us suppose that in the $x(t)$ construct we only know $|y(t)| \leq 1$ and in the $y(t)$ construct we only know $|x(t)| \leq 1$. Then the locally known versions of the dynamical equations become $\mathcal{D}|x| \leq 1$ and $\mathcal{D}|y| \leq 1$. The worst case of these is

$|x(t)| \leq t$ and $|y(t)| \leq t$. Evidently these behaviours are not stable, so that the loss of information attributable to system partitioning has destroyed certainty about stability.

We can contrast the preceding situation with that in which we have stronger information about stability. Thus instead of the preceding integrated system, suppose that the integrated system's $x(t)$ and $y(t)$ variables have the dynamical equations $\mathcal{D}\, x = y\, \mathrm{e}^{-\lambda t}$ and $\mathcal{D}\, y = -x\, \mathrm{e}^{-\lambda t}$. Now, the integrated solution is $x(t) = \sin(\frac{1-\mathrm{e}^{-\lambda t}}{\lambda})$ and $y(t) = \cos(\frac{1-\mathrm{e}^{-\lambda t}}{\lambda})$. Again we have bounded and stable behaviours, realising the bounds $|x(t)| \leq 1$ and $|y(t)| \leq 1$. In the partitioned system, if we have the same loss of information about $x(t)$ and $y(t)$ as we had before and must replace occurrences of $x(t)$ and $y(t)$ with the bounds, then the best locally known versions of the dynamical equations become $\mathcal{D}\, |x| \leq \mathrm{e}^{-\lambda t}$ and $\mathcal{D}\, |y| \leq \mathrm{e}^{-\lambda t}$. Now, the worst case becomes $|x(t)| \leq \frac{1-\mathrm{e}^{-\lambda t}}{\lambda}$ and $|y(t)| \leq \frac{1-\mathrm{e}^{-\lambda t}}{\lambda}$. This is still stable behaviour.

From the formal verification standpoint, the essential aspect of this version of events is that, even with the degraded knowledge attributable to the partitioning, if the worst case behaviour is nevertheless stable, safety invariants about the behaviours of $x(t)$ and $y(t)$ may still be provable, even if they are suboptimal compared with the global information case.

In section 3.2 we observed the analogy between discrete and continuous updates. This analogy extends to stability considerations. In the discrete world, we often prove termination of a sequential process by proving that each of its steps strictly decreases a *variant expression* which takes values in a well-founded set. The lower bound provided by well-foundedness prevents the sequential process from proceeding indefinitely. By contrast, in the continuous world, we often prove stability by proving that the continuous process (over time) strictly decreases a *Liapunov expression* which takes values in a portion of the reals that is bounded below. As the continuous process forces the Liapunov expression nearer and nearer the lower bound, the dynamics is typically increasingly confined to an asymptotic region, yielding asymptotic stability. Viewed in this light, the termination of the sequential process may also be seen as a kind of stability criterion, since, having terminated, the sequential process is unable to change the value any more.

5.4 Technical Issues in Control

One consequence of the relative longevity and maturity of control theory is that a number of different approaches have been developed to the mathematical analysis of control problems over the years. Given the necessity of relating CPS formulations to foundational issues that we have explored above, the impact of a foundational perspective on different control approaches merits examination.

Most control engineering, as taught in engineering curricula, takes place in the frequency domain, using the Laplace transform (in the case of continuous control) or z transform (for discretized control). Practically useful results are readily obtained, and mathematically, the results in this domain are derived using an L^2 notion of convergence (convergence using mean square error, in less technical language).

An alternative approach, made more popular with the availability of modern simulation based tools, seeks to solve control problems directly in state space. In this domain, some results are also derived using the L^2 formulation, while others are derived in L^∞ (convergence using maximum pointwise deviation). While the L^2 results in the frequency domain and in the state space domain can be related via Plancherel's theorem, they speak about integrated square error rather than pointwise deviation. This makes the results derivable in the two kinds of approach incompatible. A good property derived in either domain does not carry any implication of the analogous property holding in the other domain — in fact quite the opposite is usually the case: a good property derived in one domain spawns a counterexample in the other.

Thus, the various domains of control theory (and others we have not mentioned) tend to exist in separate mathematical silos to some degree, owing to the detailed differences in the rigorous formulations that define them. This being so, it is nevertheless notable that these subtle issue have rather little impact on the day-to-day practice of control engineering. Nevertheless, they *do* have impact on an integration of control issues with model based formal methods techniques, because the latter readily exhibit a sensitivity to the kinds of mathematical subtleties mentioned — and day-to-day control engineering does not. This in turn, is attributable to the fact, as we mentioned above, that model based formal methods are rooted in simple set theory, and so any integration with other concerns has to be sound when based on such set theoretical considerations.

5.5 Delay Differential Equations

The ODEs that we have focused on hitherto provide an excellent framework for describing fundamental physical processes at the microscopic level. However, in system engineering we inevitably deal with finite macroscopic components, and their size and other characteristics mean that it is often the case that there is a delay between the inputs that a component is subject to, and the outputs it can deliver in response. This is especially true if the component in question involves a communication network. Delay differential equations (DDEs) can provide a useful description of some such systems [15,10].

To see the intriguing phenomena that DDEs can lead to, let us consider the simplest possible example: $\mathcal{D}\, x = -K\, x(t - \tau)$. This says that the derivative of x is proportional to a value of x that is τ time units old. In analysing the behaviour of such an equation, is is always most useful to start by linearising, and looking at the stability of small, exponentially varying deviations. In the case of our equation, we start with the simplest possible solution, a steady state solution. In a steady state solution x does not vary with time, and the significance of the delay disappears. It is easy to see that $x^\blacklozenge(t) = 0$ is a steady state solution. If we now add a small exponential perturbation $A\, \mathrm{e}^{\lambda t}$ to $x^\blacklozenge$, we can analyse the tuples of K, A, λ values that yield solutions, and examine the stability (or otherwise) of such solutions. Substituting $A\, \mathrm{e}^{\lambda t}$ into the equation and simplifying yields $\lambda = -K\mathrm{e}^{-\lambda\tau}$. Unfortunately, even in this simplest of problems, the equation is a transcendental equation, and the analysis of the character of its solutions is not trivial.

With K real and positive, if λ is real, it must be negative (because $\mathrm{e}^{-\lambda\tau}$ is then always positive). This is good news, because it implies that $x^\blacklozenge$ is a stable solution.

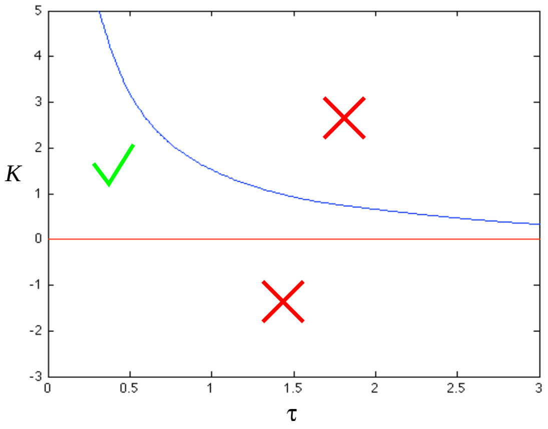


Fig. 8. Regions of solution stability and instability for the simple DDE $\mathfrak{D}\,x = -K\,x(t-\tau)$.

Unfortunately, as we vary the three parameters, we find that there are also solutions to $\lambda = -K\mathrm{e}^{-\lambda\tau}$ with non-real λ, and these lead to oscillatory behaviour around $x^{\blacklozenge}$.

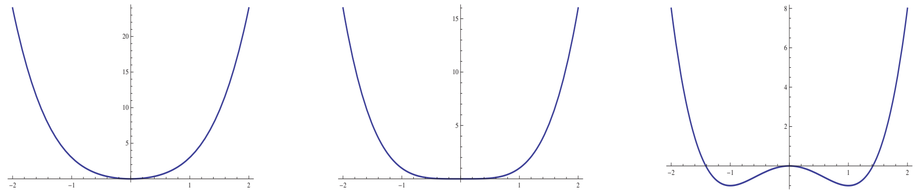
Fig. 8 shows the general characteristics of the parameter space. Evidently, $K < 0$ forces real λ to be positive, yielding instability. Also, for $K > 0$ and τ small enough, we find negative λ and stability. But as τ increases (with K fixed), we eventually cross into another unstable region. What we have outlined in embryonic form is the onset of a steady state bifurcation, in which a seemingly innocuous system, upon being subject to a change in the values of its static parameters, suddenly destabilises and exhibits oscillatory behaviour.

5.6 Bifurcations

Varying parameters to optimise resource utilisation or other performance metrics is grist to the mill in engineering design. Equally, the sudden and unexpected onset of instability and oscillatory behaviour as parameters are varied, is the bane of the engineer's life. Such behaviour is particularly perplexing to the engineer steeped in discrete methods, since there is no possible way to discover the possibility of onset of instability of the kind described, by looking at the system from a purely discrete perspective. Only by non-trivially engaging with the continuous mathematics of the system can one hope to discern the root cause of instabilities of this kind.

The steady state bifurcation we outlined above is merely the simplest example of sudden change in the global characteristics of the solution space of a system as its parameters are varied. Another commonly occurring kind of bifurcation is the *Hopf bifurcation* [13,14].[8] In this, varying the system's static parameters causes a pair of characteristic roots of the stability equation to cross the imaginary axis with non-zero

[8] Historically, Poincaré and Andronov also studied this phenomenon, before Hopf's account made it more widely known.



$\alpha = 2$ $\alpha = 0$ $\alpha = -2$

Fig. 9. Values of the schematic 'energy function' $E = r^4 + \alpha r^2$, where $r^2 = (x^2 + y^2)$ for various values of α.

velocity. As in the preceding case, a previously stable (but not steady state) solution to the system suddenly loses stability and starts to behave in an oscillatory manner.

Figs. 9 and 10 illustrate how a Hopf bifurcation works. We imagine a system containing two dynamical variables x, y (amongst others). For convenience, we suppose that there is circular symmetry in x, y so that we can use $r^2 = (x^2 + y^2)$ and get some simplification. We suppose that there is an 'energy function'[9] for x, y that looks like $E = r^4 + \alpha r^2$ where α is some system parameter that is subject to optimisation. We also assume that the system is dissipative, so that, left to its own devices, the dynamical trajectory would seek a point of minimum 'x, y energy'.

For α positive, the 'energy function' E looks like Fig. 9.(a); the minimum is comfortably at $r = 0$. When $\alpha = 0$ the 'energy function' E looks like Fig. 9.(b); the minimum is still at $r = 0$, but the neighbourhood of $r = 0$ is flatter. But when α is negative, e.g. $\alpha = -2$, the 'energy function' E looks like Fig. 9.(c); the minimum is no longer at $r = 0$ but at a non-zero value.

Typical system trajectories for α values of $+2, 0, -2$ are shown in Fig. 10. In Fig. 10.(a), for $\alpha = 2$, a typical trajectory rapidly sinks to the 'energy' minimum $r = 0$; the system is stable. In Fig. 10.(b), for $\alpha = 0$, a typical trajectory still sinks to the 'energy' minimum $r = 0$, but more slowly because of the absence of the quadratic term in E; the system is still stable because of the quartic term. But for $\alpha = -2$ the quadratic term is negative, and close to $r = 0$ it overcomes the quartic term. The 'energy' minimum jumps away from $r = 0$. There is now a circular limit cycle of minimum 'energy' in the x, y plane. Fig. 10.(c) shows how typical trajectories flee from $r = 0$. For either variable, x or y, the observed behaviour is oscillatory once the dynamics is tracing out the limit cycle.

In fact the 'energy' minimum jumps away from $r = 0$ infinitely fast as a function of α as α crosses the value 0 in a negative direction. Fig. 10.(d) shows the shape of the minimum 'energy' manifold as α varies. The paraboloid on its side in Fig. 10.(d) witnesses the sharp departure from $r = 0$ at the critical point $\alpha = 0$, at which the

[9] Assuming the existence of an 'energy function' is not essential here but it helps to illuminate the example. The 'energy function' does not have to literally be a form of energy — many energy analogues have been identified over the years that share some of the mathematical properties of genuine energy, without actually being forms of energy.

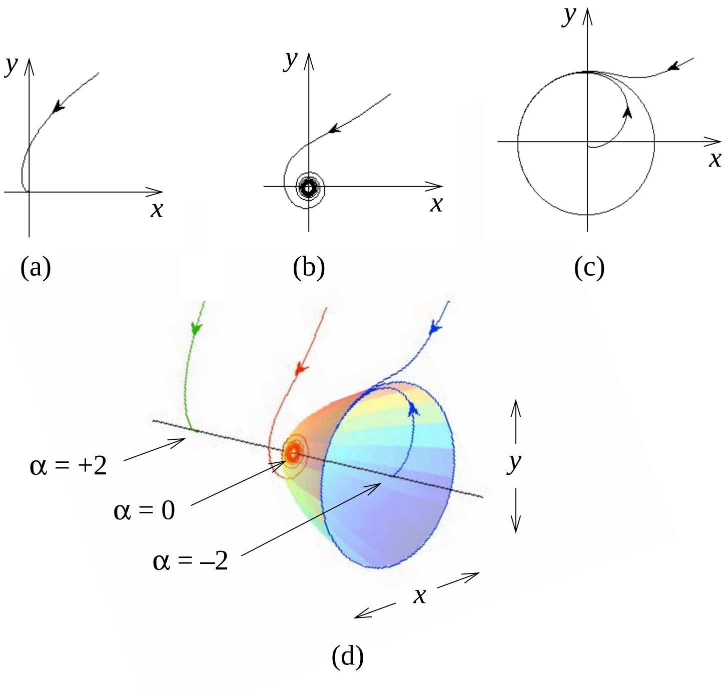


Fig. 10. Typical trajectories for the dissipative 'energy functions' of Fig. 9. In (a) $\alpha = +2$; in (b) $\alpha = 0$; in (c) $\alpha = -2$ and trajectories flee from $r = 0$. The minimum 'energy' manifold consists of the non-negative α axis together with the paraboloid on its side for negative α.

minimum energy manifold switches from a single zero value to the circle of radius $\sqrt{-\alpha/2}$. The various example trajectories are superposed on it.

Of course these examples are just the best known instances of change in the global characteristics of the solution space, and they provide an inroad into a wealth of increasingly complicated phenomena that can be increasingly difficult to discern from the standard way in which problems are posed, in terms of differential and algebraic equations. In all cases though, an approach based purely on discrete techniques is doomed to never make contact with the underlying causes of the loss of stability, and to leave designers who rely solely on such techniques perplexed by the instabilities that they encounter.

5.7 Numerical Approaches

From the perspective of gaining the greatest possible insight into the behaviour of a system, being able to attack it analytically (as we have done in the examples above) is evidently the ideal. But this ideal approach is available in an acutely small number of cases. The case of linear systems is familiar and provides the basis for most exact results in the kinds of case we have discussed, but beyond linear systems, the pickings for analytically based approaches are sparse indeed. Unfortunately, most realistic descriptions of physical systems contain nonlinearities, immediately putting them beyond the reach of analytical approaches.

Under the circumstances, numerical approaches come to the fore. Here we can distinguish two kinds of problem tackled numerically. The first kind concerns the calculation of numerical values for situations where there is an underlying analytical model. A typical such case would be a linear equation, $\mathcal{D}\, x = A\, x$ say, for which we have an analytic solution $\mathrm{e}^{A\,t}$, and we need a specific value, e.g. at $t = 5$, so we need to calculate e^{5A}. For such problems, there are typically power series (e.g. the familiar one for e^{t}), which we can use to home in on the value needed. It is important to note that in such cases there is a large amount of group theoretic machinery doing a lot of the heavy lifting behind the scenes, which allows us to focus on just the value $t = 5$. Such numerical problems are relatively easy. (Evidence for the implicit presence of the group theory is given by such familiar things as the multiplication laws for the exponential, e.g. $\mathrm{e}^{a}\mathrm{e}^{b} = \mathrm{e}^{a+b}$, or the addition laws for trigonometric or hyperbolic functions, and so on.)

The second, much tougher kind, are situations in which there is no discernible group theoretic machinery around (aside from very generic results, e.g. concerning the flow semigroup of a dynamical system, that tend not to lead to calculational techniques). Then, if the dynamics starts at $t = 0$ and we are interested in what happens at $t = 5$, there is no longer a convenient formula to have recourse to, which captures the initial conditions at $t = 0$, and into which we can just plug the value $t = 5$ to get the answer. Now, we must grasp the differential equation (or other dynamical system) and must integrate it by brute force, inching along until we reach $t = 5$. This is much more difficult for the following reason. Let $\mathcal{D}\, x = \phi(x)$ be our ODE. Assuming we know the value of x at $t = t_0$, we are interested in the value at $t = t_0 + h$. For this we will need (sooner or later) the values of $\mathcal{D}\, x$ at $t = t_0$ and $t = t_0 + h$, the latter of which depends on the value of $\phi(x)$ at $t = t_0 + h$, which depends on the value of x at $t = t_0 + h$, which is what we are trying to find! An enormous literature has arisen around this issue, because of both its technical challenge and its enormous practical utility [13,14]. The fact that there are readily available existence theorems that assure us that all these things are well defined [29] only adds to our chagrin, since they do not easily translate into efficient numerical algorithms.

A further, related point concerns the kind of results that *are* available regarding this kind of numerical integration of differential equations. Typically, they state that in the limit of the step size h going to zero, such and such an algorithmic procedure converges to values which are on the solution trajectory of the ODE. However, of more interest to the engineer is a result which quantifies the closeness of the approximation to the solution, in terms of the step size needed to achieve it. Such results are more rare, especially since the estimates used in proving the convergence tend to be suboptimal, to improve the perspicacity of the proof.

Given the somewhat pessimistic drift of the last paragraphs, we can reasonably ask whether there is much scope for the kind of analytically based approaches that we have, rather implicitly, been advocating. More specifically, how much can proof and verification based approaches contribute to the development of CPS systems? The answer is twofold. Firstly, there is the bespoke option. Although many systems are not solvable analytically, mathematical ingenuity can often rigorously elicit specific facts about the solution space of the problem, which can help improve what can be deduced numerically. Such work goes case by case, and helps sustain applied mathematics departments in universities the world over, and will continue to do so for the forseeable future. Secondly, there is the fact that engineers need to be able to predict how the artifacts they design will behave. If linear systems give the only route to routine predictability, then engineers will, overwhelmingly, tend to use linear techniques in their designs, using them in designs built out of linear pieces, combined in suitable ways to give overall behaviour which is not linear in the large. The first option offers a significant challenge to verification approaches since each problem will demand its own verification strategy, but the second is much more tractable, since it just requires the flexible combination of pieces which can be handled analytically.

5.8 Sampling, Aliasing, Quantization

Although we have focused rather heavily on phenomena latent in the continuous world in the preceding sections, in practice, system development inevitably descends to the world of discrete implementations. In this world, the smoothly changing phenomena of the continuous world dissolve into jumpy broken-up phenomena, whose behaviour is not always a simple retraction of their continuous counterparts. Of course, the desire in performing a discretization step is for the result *to be exactly* a simple retraction of the continuous version, but, as often happens, the desire for the design to optimise certain performance measures may make the discretization step cross a boundary between relatively faithful reflection of an earlier continuous design and a more chaotic regime.

The study of the correspondence between continuous and discretized worlds is perhaps most highly developed in the signal processing world. Real signals are continuous, whereas the processing of them is done almost exclusively in the digital sphere nowadays. The digitization process involves, firstly, the choice of a suitable sampling rate, and at each sampling point, the conversion of the value of the continuous signal to one of a finite number of discrete values: quantization. The converse applies when a digital signal has to be put back into the continuous world.

Much has been learned about these processes over recent decades [30]. The rule of thumb *sine qua non* in this sphere, is the Nyquist Sampling Theorem. This states that provided the sampling rate is at least twice the highest frequency present in the signal to be processed, discretization will not introduce false harmonics into a reconstructed signal. In fact, the onset of false harmonics in a reproduced signal can be seen as a kind of bifurcation in the underlying detailed dynamics. Thus, as a parameter (the sampling rate) is steadily decreased, a threshold is crossed and the dynamics bifurcates to allow the presence of not only the correct frequencies, but of the false ones too. The false frequencies give rise to what is known as aliasing. These phenomena become apparent when we precisely model what is going on in the discretization process in the continuous sphere. Of course, what works in the time domain (i.e. as regards sampling rate) works equally well in the value domain. Thus, too coarse a quantization strategy can be as damaging to a discretization process as too low a sampling rate.

Inevitably, what holds in the signal processing world applies directly to the control problem world of CPS, since both the input and output of a control process are themselves signals. The subtlety here is that, in essence, all the results derived in the signal processing world regarding sampling and quantization are derived in the L^2 sphere. This contrasts with the model based and verification approach to CPS systems which is much more concerned with results in the L^∞ sphere. This is because the model based approach works with the current state (at the current moment in time) and fidelity to some desired notion of acceptable behaviour is based on the pointwise deviation between actual and desired state, as time proceeds.

As pointed out in Section 5.4, the L^2 and L^∞ worlds are, strictly speaking, mathematically incompatible. Therefore, fully understanding the complex phenomenon of the discretization process, with its many opportunities for bifurcation arising from the additional presence of the discretization parameters remains a considerable challenge.

6 Summary and Conclusions

The kind of mathematics that might conceivably have an impact on the formulation and behaviour of CPS has been in development for at least 400 years. During this time an enormous amount of relevant knowledge has been accumulated, from the applied mathematics that strives to accurately quantify the behaviour of components and systems, to the pure mathematics that underpins the foundations of differential equations and connects these to the kind of discrete formulations familiar in the computational world.

In this essay we started by arguing for a world of piecewise absolutely continuous real functions of time, since these include both the absolutely continuous functions within which the modern theory of differential equations resides, and the discontinuous changes needed by computational frameworks and impulsive physical control. This world allows the semantics of typical syntactic frameworks for CPS to be formulated fairly easily. However it offers few guarantees regarding calculation.

Within this world we identified the functions that were (restrictions of) piecewise holomorphic functions that were real on the real line, as offering dramatically better prospects for calculation. However, despite this, they offer limited prospects for automation, since mathematical creativity is often required to get the best out of this world.

To maximise the possibilities for automation, we pointed at the functions characterised by semialgebraic properties, for which relatively more recent advances in algebraic geometry have created decision procedures based on cylindrical algebraic decomposition. Implemented in modern tools, these procedures have led to a vast surge in the mechanically supported design of complex engineering systems, characterised using semialgebraic properties.

It is only fair to point out that none of the preceding involves noise, noise being a property that every physical system exhibits to some degree. Since, in many cases, the noise can be regarded as negligible, constructing a framework that disregards it is not a waste of time. However, in many other cases noise is not negligible, and for such cases, we would need a stochastic extension of the preceding theory. Removing the 'absolutely' qualifier from our first world gives us a playground in which a stochastic calculus extension of the ideas discussed here may find a semantics. For convenience, we regard all this as outside the scope of the present essay.

Around this main thread, a number of other ideas swirl. In Section 5 we pointed out a number of them: DAEs, control issues, DDEs, bifurcations, numerical issues and quantization. Although all of them can make a difference to the description and behaviour of CPS systems, it is perhaps the bifurcations that have the most visible and dramatic impact: a system, hitherto quite well behaved. suddenly loses stability and starts to oscillate wildly, in response to an innocent looking adjustment to some static parameters.

In reality, the above constitutes a rather demanding sweep of theoretical techniques to take on board, and almost all approaches to CPS focus on one or other fairly narrow portion of this spectrum. The narrowness of focus is, of course, unfortunate, since it precludes locating the source of some difficulty that arises in the design of a system in the correct way, if the cause of the difficulty lies in some part of the spectrum unfamiliar to a particular individual.

This phenomenon is particularly prevalent in the computational world. The somewhat understandable inclination from the computational point of view to relegate the non-discrete aspects of CPS to a rather distant world of continuous behaviours, belies their ability to dramatically affect overall system behaviour in a manner that is essentially impossible for a discrete system formulation to engage with. Particularly when discussing bifurcations, we have indicated just how dramatic the effects of this can be. If this essay is to serve any useful purpose at all, it would be to help highlight the need for a deeper appreciation of just how wide the spectrum of ideas that impact CPS behaviour actually is, and thus to help stifle poorly judged views about the inadequacy of the theoretical and foundational framework for CPS. The fact of the matter remains, that *all* elements of CPS systems have well established mathematical descriptions that can help explain their behaviour. It remains the responsibility of CPS designers to appreciate the implications of all of them.

References

1. KeYmaera, http://symbolaris.com
2. 20-sim: http://www.20sim.com/
3. Alur, R.: Principles of Cyberphysical Systems. MIT Press (2015)
4. Andrews, P.: An Introduction to Mathematical Logic and Type Theory: To Truth Through Proof. Academic (1986)
5. Banach, R., Butler, M., Qin, S., Verma, N., Zhu, H.: Core Hybrid Event-B I: Single Hybrid Event-B Machines. Sci. Comp. Prog. 105, 92–123 (2015)
6. Banach, R., Butler, M., Qin, S., Zhu, H.: Core Hybrid Event-B II: Multiple Cooperating Hybrid Event-B Machines. Sci. Comp. Prog. 139, 1–35 (2017)
7. Basu, S., Pollack, R., Roy, M.F.: Algorithms in Real Algebraic Geometry. Springer (2006)

8. Carloni, L., Passerone, R., Pinto, A., Sangiovanni-Vincentelli, A.: Languages and Tools for Hybrid Systems Design. Foundations and Trends in Electronic Design Automation 1, 1–193 (2006)
9. Caviness, B., Johnson (eds.), J.: Quantifier Elimination and Cylindrical Algebraic Decomposition. Springer (1998)
10. Erneux, T.: Applied Delay Differential Equations. Springer (2009)
11. Gamelin, T.: Complex Analysis. Springer (2001)
12. Geisberger, E., Broy (eds.), M.: Living in a Networked World. Integrated Research Agenda Cyber-Physical Systems (agendaCPS) (2015), http://www.acatech.de/fileadmin/user_upload/Baumstruktur_nach_Website/Acatech/root/de/Publikationen/Projektberichte/acaetch_STUDIE_agendaCPS_eng_WEB.pdf
13. Hairer, E., Norsett, S., Wanner, G.: Solving Ordinary Differential Equations I Nonstiff Problems. Springer (1993)
14. Hairer, E., Wanner, G.: Solving Ordinary Differential Equations II Stiff and Differential-Algebraic Problems. Springer (1996)
15. Hale, J., Verduyn Lunel, S.: Introduction to Functional Differential Equations. Springer (1993)
16. Kunkel, P., Mehrmann, V.: Differential-Algebraic Equations: Analysis and Numerical Solution. European Mathematical Society (2006)
17. Lang, S.: Complex Analysis. Springer (2008)
18. Lee, E., Shesha, S.: Introduction to Embedded Systems: A Cyberphysical Systems Approach. LeeShesha.org, 2nd. edn. (2015)
19. Liu, J., Lv, J., Quan, Z., Zhao, H., Zhou, C., Zou, L.: A Calculus for Hybrid CSP. In: Ueda (ed.) Proc. APLAS-10. vol. 6461, pp. 1–15. Springer, LNCS (2010)
20. Maple: http://www.maplesoft.com
21. Mathematica: http://www.wolfram.com
22. MATLAB and SIMULINK: http://www.mathworks.com
23. MetiTarski: https://www.cl.cam.ac.uk/~lp15/papers/Arith/
24. Modelica: https://www.modelica.org/
25. Olver, F., Lozier, D., Boisvert, R., Clark, C.: NIST Handbook of Mathematical Functions. Cambridge University Press (2010)
26. Platzer, A.: Logical Analysis of Hybrid Systems: Proving Theorems for Complex Dynamics. Springer (2010)
27. Royden, H., Fitzpatrick, P.: Real Analysis. Pearson (2010)
28. Su, W., Abrial, J.R., Zhu, H.: Formalising Hybrid Systems with Event-B and the Rodin Platform. Sci. Comp. Prog. 94, 164–202 (2014)
29. Walter, W.: Ordinary Differential Equations. Springer (1998)
30. Widrow, B., Kollar, I.: Quantization Noise: Roundoff Error in Digital Computation, Signal Processing, Control, and Communications. Cambridge University Press (2008)
31. Wikipedia: Absolute continuity.
32. Zhan, N., Wang, S., Zhao, H.: Formal Modelling, Analysis and Verification of Hybrid Systems. In: Proc. UTPFM-13. vol. 8050, pp. 207–281. Springer LNCS (2013)
33. Zhou, C., Hoare, T., Ravn, A.: A Calculus of Durations. Inf. Proc. Lett. 40, 269–276 (1991)