Towards Formal Verification of Computations and Hypercomputations in Relativistic Physics

Mike Stannett(✉)

Department of Computer Science, University of Sheffield, Regent Court, 211 Portobello, Sheffield S1 4DP, Sheffield, UK
m.stannett@sheffield.ac.uk

Abstract. It is now more than 15 years since Copeland and Proudfoot introduced the term *hypercomputation*. Although no hypercomputer has yet been built (and perhaps never will be), it is instructive to consider what properties any such device should possess, and whether these requirements could ever be met. Aside from the potential benefits that would accrue from a positive outcome, the issues raised are sufficiently disruptive that they force us to re-evaluate existing computability theory. From a foundational viewpoint the questions driving hypercomputation theory remain the same as those addressed since the earliest days of computer science, viz. *what is computation?* and *what can be computed?* Early theoreticians developed models of computation that are independent of both their implementation and their physical location, but it has become clear in recent decades that these aspects of computation cannot always be neglected. In particular, the computational power of a distributed system can be expected to vary according to the spacetime geometry in which the machines on which it is running are located. The power of a computing system therefore depends on its physical environment and cannot be specified in absolute terms. Even Turing machines are capable of super-Turing behaviour, given the right environment.

1 Introduction

The term hypercomputation refers to the study of physical or abstract systems which are potentially capable of behaviours which cannot be simulated by recursive means. The term was introduced by Copeland and Proudfoot ([2]) as a more accurate replacement for the term 'super-Turing' used by Stannett ([13–15]) and Siegelmann ([12]) to describe certain types of putative hypercomputational system. Although no hypercomputer has yet been built (and perhaps never will be), it is instructive to consider what properties any such device should possess, and whether these requirements could ever be met.

Computers are physical devices whose possible behaviours are constrained and described by physical laws. The answers to the questions *what can be computed?* and *what can be computed quickly?* therefore depend on ones theory of physics and the properties of physical materials. Moreover, because physical devices exist in space and time, their computational power can depend both on

J. Durand-Lose and B. Nagy (Eds.): MCU 2015, LNCS 9288, pp. 17–27, 2015.
DOI: 10.1007/978-3-319-23111-2_2

when and where they are located. In particular, spacetime structures can boost the power of computational systems, but can also constrain and reduce their power. Similarly, an algorithm's run-time complexity is not an absolute property but depends on the spacetime trajectory being followed by the machine(s) on which it is running.

1.1 Geometrical Boosting of Computational Power

A well-known strategy for boosting computational power is to exploit the properties of *Malament-Hogarth* (M-H) spacetimes [3]. These are spacetimes containing a point p and a future-pointing semi-infinite worldline w not passing through p, such that every point x of w can be joined to p by a future-pointing timelike path which has finite proper length (Fig. 1). We refer to the pair (w, p) as an *M-H structure* in what follows.

The following lemma shows that all Σ_1^0 and Π_1^0 sets become decidable in M-H spacetime using just two Turing machines, provided they can communicate at least once.

Lemma 1. *Let S be any set in Σ_1^0 or Π_1^0. Then S can be decided in M-H spacetime by a system comprising two computers capable of communicating once.*

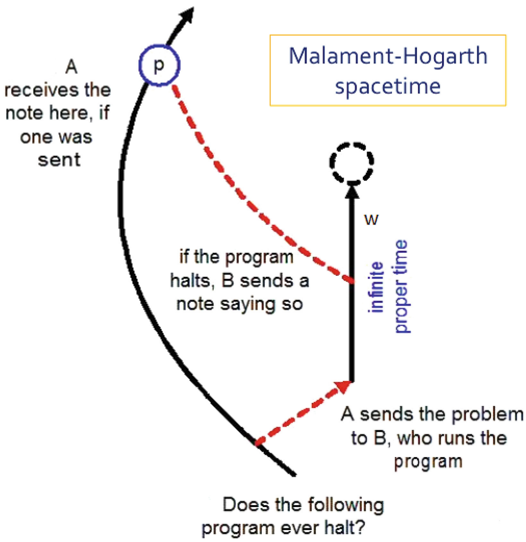

Fig. 1. Temporal structure of a hypercomputation using an M-H structure (w, p). In this example, we solve the Halting Problem in constant time using two communicating Turing machines. Machine A sends the program to machine B, and then travels to the M-H event p. Machine B, moving along w, runs the program and if it ever halts it sends a message to p saying so. On reaching p, A looks for the message. It is present at p if and only if the program halted.

Sender	*Receiver*

`y = 0;`
`while (travelling along w) {`
`  while ( R(n,y) ) { y = y+1; }`
`  transmit (result = false) to p;`
`  halt;`
`}`
	`result = true;`
	`travel to p;`
	`wait 1 second;`
	`return result;`

Fig. 2. The programs *Sender* (running on T_S, which is capable of sending at most one message to *Receiver*) and *Receiver* (running on T_R, which is capable of receiving and acting upon at most one message from *Sender*) co-operate to decide the undecidable set S in the context of an M-H structure (w, p). The two machines are initially co-located at some point on the worldline w. The 1-second delay is to avoid ambiguity as to whether *Receiver* returns *result* before or after executing *Sender*'s assignment instruction at p.

Proof. We show that any S in Π^0_1 can be decided in M-H spacetime (the Σ^0_1 case follows by complementarity). Since S is in Π^0_1 we can write $S = \{x \mid \forall y.R(x, y)\}$, where R is recursive. To decide whether $n \in S$, we run the programs *Sender* and *Receiver* shown in Fig. 2.

Suppose $n \notin S$, i.e. $\neg\forall y.R(n, y)$. Then there exists some y for which the test $R(n, y)$ fails. Let $y_{\min}$ be the smallest such y. Then

- The machine T_S travels along w, a trajectory which allows it infinite execution time (since it has infinite proper length). Consequently, *Sender* eventually encounters and fails the test $R(n, y_{\min})$, transmits the instruction "*result* = **false**" to p (along a trajectory of finite proper-length), and terminates.
- *Receiver* sets *result* to **true**, then travels to p where it encounters and executes the instruction sent there by *Sender* setting *result* to **false**. It waits one second and then returns the value of *result*, i.e. **false**.

Now suppose conversely that $n \in S$. Then

- *Sender* never exits the loop testing $R(n, y)$ and never issues the instruction setting *result* to **false**. It runs forever without terminating (its trajectory along w ensures that this is possible).
- *Receiver* sets *result* to **true** and travels to p. After waiting one second it returns the unchanged initial value of *result*, i.e. **true**.

In either case, the system eventually returns a value, and the value returned correctly reports whether or not $n \in S$. □

Lemma 1 shows that spacetime geometries can boost computational power, and that this does not require the introduction of 'unphysical' constructs like infinite precision observations or new types of machine. The machines used for this hypercomputation are simply Turing machines – indeed, *Receiver* is so simple that T_R could arguably be replaced by an essentially trivial 2-state automaton with no loss of power to the system as a whole. Notice, however, that a single machine acting alone cannot exploit the boosting effect of M-H structures, because this relies on splitting the system into two parts, one of which can run forever in a period of time that appears finite to the other. Notice also that spacetime geometries can be considerably more complicated than those considered here, and that structures can be envisaged which allow decidability at all levels of the arithmetic hierarchy [6] and beyond [18].

1.2 Geometrical Reduction of Computational Power

Spacetime geometry can also constrain and reduce computational power. For example, consider a computer traversing a closed timelike curve (CTC) or 'time loop'. Suppose the computer's clock shows that each circuit of the CTC is long enough for it to execute N instructions. Since the computer and all of its components return to their initial spacetime locations (and hence their initial machine states) after every N instructions, the number of steps executable by a CTC-traversing Turing machine is necessarily bounded, and all CTC-located programs must be reversible [16]. Indeed, it is only possible to run a fully controlled program if the temporal length of the CTC is an exact integer multiple of the program's runtime, since it will not otherwise return to its initial state on completion of each circuit.

1.3 Geometrical Effects on Computational Complexity

The possibility of M-H spacetimes also has implications for computational complexity. A simple adaptation of the distributed computation outlined in Lemma 1 allows the result produced by any program to be obtained within a fixed time period, viz. precisely one second longer than it takes *Receiver* to reach p. In M-H spacetimes, all programs have constant run-time complexity. (Similarly, CTCs can be use to transmit results 'into the past', thereby allowing program results to be obtained more quickly than would otherwise be the case.)

Notice, however, that this requires us to refine our notions of complexity slightly. The program itself may have arbitrarily large complexity, but it is running on the machine *Sender* which is not responsible for reporting the program output. Instead, this is reported by *Receiver* in constant time. In relativistic settings, it is essential to identify carefully which components in a distributed system are deemed responsible for generating the final system output.

2 Modelling Relativity Theory in Isabelle/HOL

Since a spacetime might potentially contain a combination of 'normal' regions, M-H structures and CTCs, the question "what can be computed" has no absolute answer but depends on local and global geometric properties, the number of machines available, their relative spacetime trajectories during computation, and the availability of suitable communication channels. This is a question we would like to investigate in more detail, but we are hampered by the informal yet detailed nature of many proofs in relativity theory (and physics in general). The issue is particularly relevant because the black hole observed at the centre of our own galaxy Milky Way is potentially of the right type to be a habitat for M-H structures [4], and while such structures are obviously beyond our current technological capabilities to exploit, the mere possibility of their existence is enough to warrant a re-evaluation of the extent to which abstract computability and complexity theory give an accurate account of what is actually possible in the physical universe.

In 2012 we joined forces with researchers at the Rényi Institute of Mathematics in Budapest, who have spent many years developing versions of relativity theory expressed in first order logic – our goal is to express the Hungarian theories in Isabelle/HOL [9] so as to allow machine-assisted investigation of various key hypotheses concerning the possibilities for computation and hypercomputation in relativistic physics [17]. In this section we briefly describe the Hungarian approach, and show how it can be translated with relative ease into machine-readable form.

2.1 First-Order Relativity Theory

The approach adopted by Andréka, Németi and the Hungarian team is to formulate a collection of related relativity theories in first-order logic (FOL), using axioms that are as simple and transparent as possible [1]. Our own starting point is the translation of the Hungarian axioms and theorems into machine-readable format suitable for use with the Isabelle/HOL proof assistant [9].

For example, special relativity is represented as a theory SpecRel based on just four physical axioms:

- AxPh (Photon Axiom)
  Each inertial observer considers the speed of light to be positive, and the same in every spatial direction. Moreover, photons can be emitted in or arrive from any spatial direction.
- AxEv (Event Axiom)
  All observers inhabit the same universe, i.e. they consider the same events to take place (though possibly at different locations or times).
- AxSelf (Self Axiom)
  Inertial observers consider themselves to be stationary.
- AxSym (Symmetry Axiom)
  Whenever observers consider two events to be simultaneous, they agree as to the spatial distance between those two events – this allows observers to calibrate their rulers relative to one another.

The underlying theory has two basic sorts: *quantities* and *bodies*. Quantities are used to express distances and times, and are assumed to satisfy the axioms of a field. Bodies in SpecRel include *inertial observers* and *photons*, which are identified by predicates, e.g. $IObs(b)$ is *true* if and only if body b is an inertial

```
class Lines = Quantities + Vectors + Points
begin

  ...

  fun space2 :: "('a Point) ⇒ ('a Point) ⇒ 'a" where
    "space2 u v
      = (xval u - xval v)*(xval u - xval v)
      + (yval u - yval v)*(yval u - yval v)
      + (zval u - zval v)*(zval u - zval v)"

  fun time2 :: "('a Point) ⇒ ('a Point) ⇒ 'a" where
    "time2 u v = (tval u - tval v)*(tval u - tval v)"
```

Fig. 3. Spatial and temporal distances are defined as properties of lines, and are used to calculate the speeds needed to move from one spacetime location to another. The class `Lines` is one of several classes bundled together to form the background context class `SpaceTime` which defines the geometrical structures needed to describe spacetime. These include quantities, vectors, points, cones, straight lines and planes.

observer, and likewise $Ph(b)$ indicates whether b is a photon. Central to all of the Hungarian versions of first-order relativity theory is the *worldview relation*, W, where $W(m, b, x)$ means that observer m considers body b to be present at location x.

These constructs are generally sufficient to allow the axioms to be specified. For example, we can use the field axioms to define functions $space^2$ and $time^2$ giving the (squared) spatial and temporal distances between two spacetime events (Fig. 3). Recalling that $IOb(m)$ means "m is an inertial observer", these in turn let us write AxPh as

$$
\begin{aligned}
IOb(m) \rightarrow (\exists v.((v > 0) \wedge (\forall xy.(& \\
&(\exists p.(Ph(p) \wedge W(m, p, x) \wedge W(m, p, y))) \\
&\quad \leftrightarrow (space^2\ xy = (v * v) * (time^2\ xy))))))
\end{aligned}
$$

In words: each inertial observer is associated with a positive speed v with the property that whenever any photon is considered by m to pass through two spacetime locations x and y, the (squared) speed associated with the straight line joining these points is v^2.

The translation into Isabelle/HOL format is now straightforward, viz.

```
class AxPh = WorldView +
assumes
  AxPh:"IOb(m)
        ⟹ (∃v. ( (v > (0::'a)) ∧ ( ∀x y . (
            (∃p. (Ph p ∧ W m p x ∧ W m p y))
                  ⟷ (space2 x y = (v * v)*(time2 x y))
           ))))"
```

```
record Body =
  Ph :: "bool"
  IOb :: "bool"

class WorldView = SpaceTime +
fixes
  (* Worldview relation *)
  W :: "Body ⇒ Body ⇒ 'a Point ⇒ bool" ("_ sees _ at _")
   ...
```

Fig. 4. A body can be a photon and/or an inertial observer. We do not require that the body should only be one or the other, because this is a theorem that can be proven from the axioms. The worldview relation is a predicate defined on two bodies and one location, and introduces the notation `a sees b at x` as a more intuitive rendition of `W a b x`. It inherits basic definitions from the class `SpaceTime`.

This is an essentially verbatim translation of AxPh. It assumes that various `WorldView` constructs of Fig. 4 are in place, including the inherited definitions of `space2` and `time2`.

Two other first-order variants of relativity theory are also relevant here. The theory AccRel represents a kind of halfway-house: bodies can be accelerated (non-inertial), but we do not as yet include Einstein's Equivalence Principle relating acceleration to gravity. Adding an axiom representing the latter leads to GenRel, the first-order theory of general relativity. The use of the `record` construct in Isabelle/HOL is especially useful in this context, as it allows us to extend some of our definitions very easily. When reasoning in SpecRel, for example, we assume that bodies are either photons or inertial observers. When we come to define AccRel we can simply extend the `Body` record to include a third predicate for non-inertial observers, without having to re-work our earlier proof that bodies cannot be both photons and inertial observers. (Alternatively, as long as we avoid introducing a fourth type of body we can identify non-inertial observers semantically – they are bodies `b` for which `IOb b` and `Ph b` are both `false`.)

Choosing the axioms as simple as possible allows us to investigate the extent to which different axioms can be weakened without losing physical realism. For example, while AxPh says that each observer considers the speed of light to be constant, there is no assumption that different observers agree as to what this speed is (this is instead proven as a theorem). Similarly, there is no axiom declaring the sets of photons and inertial observers to be disjoint; this is another theorem. On the other hand, the drive for simplicity is not without cost. For example, the reader may be wondering why AxPh refers to the *squared* speed of light. This is because FOL is not powerful enough to characterise the field $\mathbb{R}$ of real numbers; there are fields which satisfy precisely the same first order theorems as $\mathbb{R}$ but which admit infinite values and infinitesimals [5,10]. Similarly, $\mathbb{R}$ satisfies various additional field axioms that are not always needed for the theorems we wish to prove; in particular we do not generally assume the

Euclidean axiom (that all positive quantities have square roots) because, as AxPh shows, we can redefine concepts using squared values instead. The question naturally arises, which number fields can be used when modelling relativity theory? Madarász and Székely argue that the answer depends on the underlying axiom system used to capture each particular version of relativity theory, and have demonstrated that an axiom system for special relativity can even be defined over the field $\mathbb{Q}$ of rationals [7]. Taking such considerations into account can add significantly to the work involved in stating theorems and developing their proofs.

Nonetheless, the approach has several advantages from a computational point of view. Consider, for example our Isabelle/HOL description of basic spacetime constructs. This is a 836-line file giving definitions, axioms and proofs relating to quantities, vectors, points, lines, planes and cones. This file took approximately 4 person-weeks to construct and verify, but now that it is in place the sparse nature of our assumptions and constructs means that relatively little additional work is required when moving from the special (SpecRel) to the accelerated (AccRel) or general (GenRel) first-order theories of relativity. The main difficulty lies not in translating the underlying axioms and theorems, but in generating verifiable proofs.

2.2 Generating Verifiable Proofs

Automated theorem provers are extremely useful tools, but they are also unforgiving. For example, in our proof of Lemma 1 we wrote *the* Σ_1^0 *case follows by complementarity*, assuming that the reader would have sufficient mathematical competence to infer the following argument:

- if S is a Σ_1^0 set, it can be written $S = \{x \mid \exists y.R(x,y)\}$ for some recursive predicate R.
- this can be rewritten $S = \{x \mid \neg\forall y.\neg R(x,y)\}$.
- this is the complement of the set $S' = \{x \mid \forall y.\neg R(x,y)\}$.
- the predicate $R' \equiv \neg R(x,y)$ is recursive because R is recursive.
- consequently $S' = \{x \mid \forall y.R'(x,y)\}$ is a Π_1^0 set.
- consequently (as proven) S' is decidable in M-H spacetime.
- and hence $S \equiv \mathbb{N} \setminus S'$ is decidable in M-H spacetime.

Seen in this way, it is clear that the phrase *follows by complementarity* conceals a significant amount of detailed reasoning, and all of this reasoning would need to be expressed in machine-readable form if we were to attempt a machine-verification of our proof.

As our machine verification of the SpecRel theorem "no observer can travel faster than light" reveals, this problem of abbreviated reasoning is just as pronounced when discussing proofs relating to physical theories. Indeed, the bulk of the work involved choosing sensible descriptions of what we mean by geometrical terms like *line*, *plane* and *cone*. For example, while a mathematician would accept that two lines that are both parallel to a third line must be parallel to each other, this required detailed proof within Isabelle/HOL (Fig. 5).

```
lemma lemParallelTrans:
  assumes "lineA \<parallel> lineB"
    and   "lineB \<parallel> lineC"
    and   "direction lineB \<noteq> vecZero"
  shows   "lineA \<parallel> lineC"
proof -
  have case1: "direction lineA = vecZero \<longrightarrow> ?thesis" by auto
  have case2: "direction lineC = vecZero \<longrightarrow> ?thesis" by auto

  {
    assume case3: "direction lineA \<noteq> vecZero \<and> direction lineC \<noteq> vecZero"

    have exists_kab: "\<exists>kab.(kab \<noteq> (0::'a) \<and> direction lineB = kab**direction lineA)"
      by (metis parallel.simps assms(1) case3 assms(3))
    then obtain kab where kab_props: "kab \<noteq> 0 \<and> direction lineB = kab**direction lineA" by auto

    have exists_kbc: "\<exists>kbc.(kbc \<noteq> (0::'a) \<and> direction lineC = kbc**direction lineB)"
      by (metis parallel.simps assms(2) case3 assms(3))
    then obtain kbc where kbc_props: "kbc \<noteq> 0 \<and> direction lineC = kbc**direction lineB" by auto

    def kac \<equiv> "kbc * kab"
    have kac_nonzero: "kac \<noteq> 0" by (metis kab_props kac_def kbc_props no_zero_divisors)
    have "direction lineC = kac**direction lineA"
      by (metis kab_props kbc_props kac_def lemScaleScale)
    hence ?thesis by (metis kac_nonzero parallel.simps)
  }
  from this have "(direction lineA \<noteq> vecZero \<and> direction lineC \<noteq> vecZero) \<longrightarrow>
  ?thesis" by blast

  thus ?thesis by (metis case1 case2)
qed
```

Fig. 5. Isabelle/HOL proof that if two lines are both parallel to a third line, then they are also parallel to each other.

Having constructed all of the ‘background’ theory, translating the Hungarian proof that observers cannot travel faster than light into Isabelle/HOL form became a relatively straightforward – though still extremely time consuming – process of writing down the major steps in the proof, and then carefully filling in every possible gap in the reasoning until complete verification was achieved.

3 Next Steps

Although we have had promising results modelling SpecRel, including the first known machine verified proof of the statement “no observer can travel faster than light”, the time involved in constructing these proofs means we have yet to make comparable progress developing Isabelle/HOL verification systems for theorems in AccRel or GenRel. Our ultimate goal is to provide indisputable proof of the conjectures:

Conjecture 1. *Computation in standard Euclidean spacetime means Turing computation.*

Conjecture 2. *Computation in M-H spacetimes verifiably includes super-Turing computation.*

However, verifying these conjectures formally adds an additional layer of complexity, because they introduce a new factor not normally considered when discussing relativity theory, namely the nature of computers and computations.

In particular, as we saw in Sect. 1.1 we need to capture within Isabelle/HOL a first-order theory representing distributed computation occurring within M-H spacetimes, and we envisage having to capture a localised variant of a theory at least as complex as the π-calculus [8, 11], since we need to discuss the properties of systems comprising multiple spatially-separated mobile components. Moreover, given the reliance of the schemes presented here upon the properties of M-H structures like those occurring in certain types of spacetime singularity, we will presumably also need to model what it means for a spacetime to contain a black hole, what it means for that black hole to be rotating, what it means for that rotation to be slow, and what it means for an entity to cross the event horizon. These are all new concepts in the world of Isabelle/HOL proof construction, and while we recognise that the task will require years rather than months to complete, we remain ever hopeful of eventual success.

References

1. Andréka, H., Madarász, J.X., Németi, I.: Logical analysis of relativity theories. In: Hendricks, et al. (eds.) First-Order Logic Revisited, pp. 1–30. Logos Verlag, Berlin (2004)
2. Copeland, J., Proudfoot, D.: Alan Turing's forgotten ideas in computer science. Sci. Am. **280**(4), 99–103 (1999)
3. Etesi, G., Németi, I.: Non-turing computations via Malament-Hogarth space-times. Int. J. Theor. Phys. **41**, 341–370 (2002)
4. Genzel, R., Schoedel, R., Ott, T., Eckart, A., Alexander, T., Lacombe, F., Rouan, D., Aschenbach, B.: Near-infrared flares from accreting gas around the supermassive black hole at the Galactic Centre (2003). arXiv:astro-ph/0310821
5. Goldblatt, R.: Lectures on the Hyperreals: An Introduction to Nonstandard Analysis. Graduate Texts in Mathematics, vol. 188. Springer-Verlag, Heidelberg (1998)
6. Hogarth, M.: Deciding arithmetic using SAD computers. Br. J. Philos. Sci. **55**, 681–691 (2004)
7. Madarász, J.X., Székely, G.: Special relativity over the field of rational numbers. Int. J. Theor. Phys. **52**(5), 1706–1718 (2013)
8. Milner, R.: Communicating and Mobile Systems: The Pi Calculus. Cambridge University Press, Cambridge (1999)
9. Nipkow, T.: Programming and proving in Isabelle/HOL, August 2014. http://www.cl.cam.ac.uk/research/hvg/Isabelle/dist/Isabelle/doc/prog-prove.pdf
10. Robinson, A.: Non-Standard Analysis. Princeton University Press, Princeton (1996)
11. Sangiorgi, D., Walker, D.: The Pi-Calculus: A Theory of Mobile Processes. Cambridge University Press, Cambridge (2003)
12. Siegelmann, H.: Computation beyond the Turing limit. Science **268**(5210), 545–548 (1995). http://www.dx.doi.org/10.1126/science.268.5210.545
13. Stannett, M.: Super-Turing computation. Seminar presentation, Department of computer science, University of Sheffield (1990). http://www.researchgate.net/publication/258848388_1990_Super-Turing_Computation
14. Stannett, M.: X-machines and the halting problem: building a super-Turing machine. Formal Aspects Comput. **2**(1), 331–341 (1990). http://www.dx.doi.org/10.1007/BF01888233

15. Stannett, M.: An introduction to post-Newtonian and super-Turing computation. Technical report CS-91-02, Department of Computer Science, University of Sheffield (1991). http://www.researchgate.net/publication/236852111_An_Introduction_to_post-Newtonian_and_super-Turing_computation
16. Stannett, M.: Computation and spacetime structure. Int. J. Unconv. Comput. **9**(1–2), 173–184 (2013)
17. Stannett, M., Németi, I.: Using Isabelle/HOL to verify first-order relativity theory. J. Autom. Reasoning **52**(4), 361–378 (2014)
18. Welch, P.: The extent of computation in Malament-Hogarth spacetimes (2006). arXiv:gr-qc/0609035

http://www.springer.com/978-3-319-23110-5

Machines, Computations, and Universality
7th International Conference, MCU 2015, Famagusta, North Cyprus, September 9-11, 2015, Proceedings
Durand-Lose, J.; Nagy, B. (Eds.)
2015, XX, 199 p. 41 illus., Softcover
ISBN: 978-3-319-23110-5