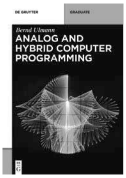
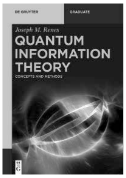
Bernd Ulmann

Analog Computing

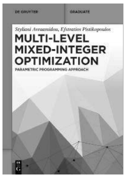
Also of interest



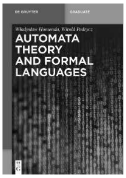
Analog and Hybrid Computer Programming
Bernd Ulmann, 2020
ISBN 978-3-11-066207-8, e-ISBN 978-3-11-066220-7



Quantum Information Theory
Concepts and Methods
Joseph M. Renes, 2022
ISBN 978-3-11-057024-3, e-ISBN (PDF) 978-3-11-057025-0

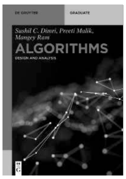


Multi-level Mixed-Integer Optimization
Parametric Programming Approach
Styliani Avraamidou, Efstratios Pistikopoulos, 2022
ISBN 978-3-11-076030-9, e-ISBN (PDF) 978-3-11-076031-6



Automata Theory and Formal Languages
Wladyslaw Homenda und Witold Pedrycz, 2022
ISBN 978-3-11-075227-4, e-ISBN (PDF) 978-3-11-075230-4

High Performance Parallel Runtimes
Design and Implementation
Michael Klemm, Jim Cownie, 2021
ISBN 978-3-11-063268-2, e-ISBN (PDF) 978-3-11-063272-9



Algorithms
Design and Analysis
Sushil C. Dimri, Preeti Malik, Mangey Ram, 2021
ISBN 978-3-11-069341-6, e-ISBN (PDF) 978-3-11-069360-7

Bernd Ulmann

Analog Computing

2nd edition

DE GRUYTER
OLDENBOURG

Mathematics Subject Classification 2010
Primary: 34-04, 35-04; Secondary: 92C45, 92D25, 34C28, 37D45

Author
Prof. Dr. Bernd Ulmann
Schwalbacher Str. 31
65307 Bad Schwalbach
ulmann@analogparadigm.com

ISBN 978-3-11-078761-0
e-ISBN (PDF) 978-3-11-078774-0
e-ISBN (EPUB) 978-3-11-078787-0

Library of Congress Control Number: 2022946448

Bibliographic information published by the Deutsche Nationalbibliothek
The Deutsche Nationalbibliothek lists this publication in the Deutsche Nationalbibliografie; detailed bibliographic data are available on the Internet at http://dnb.dnb.de.

Cover image: Bernd Ulmann
Printing and binding: CPI books GmbH, Leck

www.degruyter.com

To my beloved wife Rikka

Acknowledgments

This book would not have been possible without the support and help of many people. First of all, I would like to thank my wife Rikka Mitsam, who not only did a lot of proofreading and a terrific job in preparing many of the pen-and-ink drawings, but also never complained about being neglected although I lived the last months more or less in seclusion writing this book.

I am particularly grateful for the support and help of Jens Breitenbach, who did a magnificent job at proofreading and provided many suggestions and improvements enhancing the text significantly. He also spotted many LaTeX-sins of mine, thus improving the overall appearance of this book considerably.

In addition to that, I would like to thank Benjamin Barnickel, Arne Charlet, Daniela Koch, Theresa Szczepanski, Dr. Reinhard Steffens and Francis Massen for their invaluable help in proofreading.

Additionally, I would like to thank Tore Sinding Bekkedal, who took the picture shown in figure 8.35, Tim Robinson, who built the incredible Meccano-Differential Analyser shown in figure 2.22, Tibor Florestan Pluto, who took the photo shown on the title page, Robert Limes, who took the picture shown in figure 9.3, and Bruce Baker, who donated the pictures shown in figures 13.60, 13.62, and 13.63, for their permissions to use the aforementioned pictures.

Without the continuous encouragement of Dr. habil. Karl Schlagenhauf and his invaluable suggestions this book might very well not exist at all.

Last but not least, I would like to thank Prof. Dr. Wolfgang Giloi, Prof. Dr. Rudolf Lauber, Dr. Adolf Kley, Prof. Dr. Günter Meyer-Brötz and Manfred Klittich for sharing their memories of the early days of analog computing at Telefunken, etc. Furthermore, Jeroen Brinkmann introduced me to the "Hammer Computer" and Bert Brouwer offered many the insights into solving partial differential equations in conjunction with heat transfer problems.

This book was typeset with LaTeX. Most schematics were drawn using EAGLE and Arno Jacob's wonderful analog computing symbol library, other vector graphics were created with `xfig`.

https://doi.org/10.1515/9783110787740-202

Preface to the 2nd edition

This second edition of “Analog Computing” would not exist if it wasn’t for Dr. Damiano Sacco, acquisition editor at DeGruyter, who asked me to prepare a new edition of “Analog Computing”, which has established itself as a standard textbook since it was first published in 2013.

A lot has happened since then in the world of analog computing. First of all many “new” historic sources, papers, and photographs have become available since 2013. Accordingly, the bibliography of this 2nd edition has been expanded by more then 200 additional sources. Second, and even more importantly, interest in analog computing has grown enormously. This computing paradigm is about to change the world of computing in the near future with a plethora of interesting and commercially important applications ranging from low power computing as required for medical implants to high performance computing, artificial intelligence, and many, many more. It seems even plausible that analog computers might be able to do things typically ascribed to quantum computers while being much simpler to implement, run, and program.

Bringing back analog computers in much more advanced forms than their historic ancestors will change the world of computing drastically and forever. They will not replace the ubiquitous stored-program digital computers but they will complement them, thus making it possible to solve problems that are currently out of reach for standalone digital computers.

The author is especially indebted to Dr. Chris Giles for many valuable discussions and his meticulous proofreading of this 2nd edition and to Nicole Matje for her valuable corrections and suggestions. The author would also like to thank Achim Dassow for information about the QK-329 beam-deflection tube and additional information on early multiplication devices. Rainer Glaschick provided much background information on early differential analysers and on the hyperbolic field tube. Oliver Bach also proofread the book and took special care of the bibliography making sure that it is consistent and correct. Prof. Dr. Dirk Killat contributed figure 12.9. The new cover picture was taken by Tibor Florestan Pluto and shows the author with a GTE-22 analog computer from the late 1960s.

https://doi.org/10.1515/9783110787740-203

Contents

"*An analog computer is a thing of beauty and a joy forever.*"[1]

1 John H. McLeod, Suzette McLeod, "The Simulation Council Newsletter", in *Instruments and Automation*, Vol. 31, March 1958, p. 488.

1 Introduction

1.1 Outline

A book on analog computing and analog computers? You might ask: Isn't that 60 years late? No, it isn't – although the beautiful analog computers of the past are long since history and only few have been preserved in museum collections, the idea of analog computing is still a marvel of elegance and the following chapters will show that it has a bright and fruitful future.

The intention of this book is twofold: It gives a comprehensive description of the history and technology of classic analog computing but also shows the particular strengths of the analog computation paradigm, which, combined with current state of the art digital circuitry, will find applications in areas such as low power computing, high performance computing and maybe most importantly in *artificial intelligence* (*AI*) and many other fields.

The following chapters first introduce the notion of analog computing before describing the early development of analog computers starting with mechanical analog computers like the Antikythera mechanism, which was built around 100 B.C., and ending with the first analog electronic[1] analog computers developed by Helmut Hoelzer in Germany and George A. Philbrick in the United States.

Next, the basic elements of a typical analog computer are described followed by two chapters showing the anatomy of typical analog computers, ranging from classic systems to more recent implementations, and showing examples of some systems.

The next chapter gives an introduction to analog computer programming[2] followed by a number of practical programming examples ranging from the solution of simple differential equations to the simulation of more complex and non-linear systems.

Hybrid computers (analog computers coupled with stored-program digital computers) are covered in the next chapter. This is followed by a treatment of *Digital Differential Analysers* (digital implementations of analog computers), a chapter on stochastic computing, and a chapter on the simulation of analog computers on classic stored-program digital computers.

The next chapter covers a plethora of classic and current applications of analog computing.

1 The notion of an *analog electronic analog computer* may look like a pleonasm, which it is not, as the following section will show.

2 A much more in-depth treatment of analog and hybrid computer programming can be found in [Ulmann 2020/1].

https://doi.org/10.1515/9783110787740-001

The last chapter of this book covers the decline of analog computing in the late 1970s/early 1980s and the potentially bright future of analog computing in the 21st century.

1.2 The notion of analog computing

First of all it should be noted that the common distinction between *digital* and *analog* computers, based on the way values are represented, is not correct. It is often said that digital computers differ from analog computers by their way of representing numbers as sequences of bits (binary digits), while electronic analog computers work with continuous voltages or currents to represent variables. This erroneous view has even found its way into some encyclopedias.

Apart from the fact that even voltages or currents are not really continuous – eventually an operation like integration boils down to charging a capacitor with discrete electron charges – some analog computers have used a bit-wise value representation and have been implemented using purely digital elements.

If the type of values used in a computation – discrete versus continuous – is not the distinguishing feature, what else could be used to differentiate between *digital* and *analog* computers? It turns out that the difference is to be found in the structure of these two classes of machines: In our modern sense of the word, a digital computer's constituent elements have a fixed structure and it solves problems by executing a sequence (or sequences) of instructions that implement an algorithm. These instructions are read from some kind of memory, thus, a better term for this kind of computing machine would be *stored-program digital computer* since this describes both features of such a machine: Its ability to execute instructions fetched from a memory subsystem and working with numbers that are represented as streams of digits.[3]

An analog computer on the other hand is based on a completely different paradigm: Its internal structure is not fixed – in fact, a problem is solved on such a machine by changing its structure in a suitable way to generate a *model*, an *analog* of the problem.[4] This analog is then used to *analyse* or *simulate* the problem to be solved.[5]

Thus, the structure of an analog computer that has been set up to tackle a specific problem represents the problem itself while a stored-program digital

3 Today these numbers are normally represented by *binary digits*, *bits* for short.

4 [Tse et al. 1964, p. 333] characterized these analogs or analogies as follows: "*The term 'analogy' is defined to mean similarity of relation without identity.*"

5 The path from analogy-making to modelling of a problem is treated comprehensivly by [Care 2008].

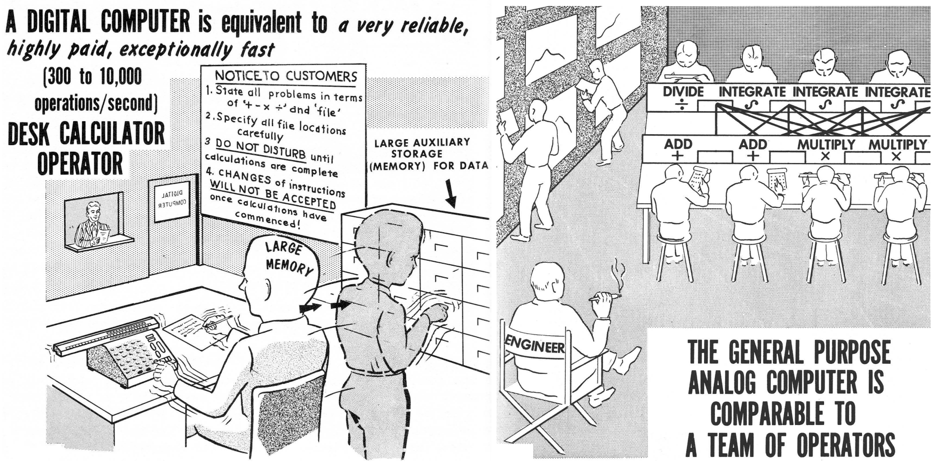


Fig. 1.1. Comparison of the basic structure of stored-program digital computers and analog computers (see [TRUITT et al. 1960, p. 1-40] and [TRUITT et al. 1960, p. 1-41])

computer keeps its structure and only its controlling program changes. This is summarized by CHARLESWORTH[6] as follows:

> "*An analogue computer is a piece of equipment whose component parts can be arranged to satisfy a given set of equations, usually simultaneous ordinary differential equations.*"

Similarly [BERKELEY et al. 1956, p. 75] states that

> "*[a]nalog computers, as the name is intended to imply, compute by means of setups that are analog of the problems to be solved.*"

Figure 1.1 shows this basic difference in architecture and operation between digital and analog computers.

Consequently, it is perfectly possible to build digital analog computers and this has been done in several ways.[7] In fact such machines may play a substantial role in the future when high precision is of the utmost importance and energy efficiency is a secondary consideration.

Employing the techniques of building models, analogs of problems to be solved or analysed, which have been developed through more than 50 years in the context of our current digital technology, can and will lead to systems with exceptional computational power as well as low power consumption.

---

6 See [CHARLESWORTH et al. 1974, p. xi].

7 Cf. sections 10 and 13.15.8.

	Basic technology	
	Analog electronic	Digital electronic
Stored-program control	N/A	stored-program (memory programmed) digital computer
Setting up an analog	traditional analog electronic analog computer	digital differential analyser

Table 1.1. Types of computing machines based on an analog/digital electronic implementation with control based on either a stored-program concept or the implementation of an analog.

Table 1.1 shows the four basic possible combinations of analog/digital implementation technology and stored-program control vs. setting up an analog. Of these combinations only three are of practical interest:

1. The modern stored-program digital computer,
2. the traditional analog electronic analog computer, which will be called an *analog computer* for simplicity in the following text, and, finally,
3. the *Digital Differential Analyser*,[8] which will be described in more detail in chapter 10.[9]

1.3 Direct and indirect analogies

When talking about analogies, it is necessary to distinguish between *direct* and *indirect* analogies. These two terms describe two extremes of abstraction levels in building analogies.

In the strict sense of the word direct analogies are models that are based on the same physical principles as the underlying problem to be solved just with a different scaling regarding size or time of a simulation. Well-known examples of such direct analogies are the determination of minimal surfaces using soap films,[10] the evaluation of tensile structures as they are used for roof structures

8 *DDA* for short.

9 It will be shown that DDAs are more capable machines than traditional analog computers since they can deal well with the highly important class of partial differential equations, which analog computers can do only with considerable difficulty and often only by means of discretisation.

10 See [Bild der Wissenschaft 1970] for examples.

like the one built for the Olympic stadium in Munich,[11] wind tunnel models for the evaluation of aerodynamic properties of aircraft and rockets and many more. Other direct analogs employed *electrolytic tanks*. These are reservoirs filled with an electrolytic liquid with embedded electrodes to generate a desired potential distribution within the liquid. Using a two- or three-dimensional sensor carriage, much like an xy-plotter, the potential at any given coordinate within the tank can be determined. Such electrolytic tanks were widely used to solve problems in nuclear engineering, etc.

Over time the notion of direct analog analogies was also applied to computers consisting of networks of passive electronic components such as resistors, capacitors and inductors. [Paschkis et al. 1968, p. 5] defines a direct analog computer as being

> "*based on the identity of the equations describing two or more systems and carrying out measurements on that system which appears most convenient for that purpose [...] In the direct analog, there is a one-to-one relationship between the (passive) components and the physical properties of the several parts of the prime system.*"

Due to their very nature such direct analogs are not very versatile and were often built and employed for a highly specific purpose.[12] Figure 1.2 shows a wonderful example for a direct analogy, a string-weight model used by Antoni Gaudí[13] during the design and construction of the the Colónia Güell church. This model was built in 1908 at a scale of 1:10 with the weights scaled down by a factor of 10^{-4}. Such models are hanging down and simulate the pressure forces acting on pillars and columns by corresponding tensile forces through the maze of strings with their attached weights.[14]

In contrast, indirect analogies exhibit a much higher degree of abstraction and are thus much more versatile. The – mostly analog electronic – analog computers used for setting up indirect analogies are truly universal machines and cover a wide range of possible applications.[15] This higher level of abstraction makes the programming of this class of machines quite challenging since their setup does not

11 The roof structure of the Olympic stadium in Munich was modelled to a large extent using curtain net lace as well as soap bubbles for determining the structure of single roof tiles.

12 More detailed information about this basic class of analogs can be found in [Jackson 1960, pp. 319 ff.], [Paschkis et al. 1968], [Master et al. 1955], [Larrowe 1955] and [Karplus 1958].

13 06/25/1852–06/10/1926

14 See [Krämer 1989] for more details on this particular model. This approach was by no means new even in Gaudí's time. See [Havil 2019, pp. 173 ff.] for a mathematical and historic perspective.

15 One of the earliest publications on the use of electronic analogs to simulate mechanical and acoustical systems was [Olson 1943].

Fig. 1.2. Scale model for the Colonia Güell church

bear any direct resemblance of the problem to be solved. Therefore a thorough mathematical description of the basic problem is required as a precondition for programming an indirect analog computer,[16] as in the case of our modern stored-program digital computers.

Nevertheless, the level of abstraction required for the successful application of analog computers is still relatively small compared with the algorithmic approach of stored-program digital computers. Last but not least, analog computers, be they direct or indirect, are models.

Due to the fact that analog computers work by acting as a model for a given problem that is represented by direct or indirect means, the amount of circuitry necessary for a simulation is determined by the complexity of the underlying problem.

Accordingly, analog computers are not capable of the trade off between time to solution on the one hand and complexity of the underlying problem on the other that is characteristic of stored-program digital computers. This is both a

16 Direct analogs can also be employed in cases where no complete mathematical description of the problem to be solved exists – this may be caused by a principle lack of understanding or by the sheer complexity of the underlying problem. So in some cases direct analogs may even be employed today with success.

curse and a blessing: The curse being that an analog computer consisting of a given number of computing elements cannot solve a problem that requires more computing elements to be implemented. The blessing is that the time to solution on an analog computer is more or less constant and is not related to the size of the underlying problem.

Thus, large problems require large analog computers regardless of the acceptable time to solution – some classic problem areas, especially those found in aerospace and applications in the chemical industry, required well over 1 000 computing elements resulting in substantial, if not giant, analog computers.

In addition to this, a stored-program computer can always exchange compute time for precision – something an analog computer also cannot normally do.[17] The precision of an analog computer is given by its particular implementation and typically does not exceed about three to four decimal places for the variables involved in a computation.

17 This does not hold true for digital differential analysers, cf. section 10.

2 Mechanical analog computers

The earliest analog computers were mechanical in their very nature but were far from being simple. In fact many mechanical analog computers were successfully employed to tackle complex problems ranging from peaceful tide computations to war-time applications like bomb trajectories, fire control, etc. The following sections give a short overview of the era of mechanical analog computers without going too much into detail since mechanical analogs will serve just as a prelude to this book's main theme of electronic analog computers.

2.1 Astrolabes

As early as about 150 B.C. the basics of *astrolabes* were developed. Such devices are basically inclinometers with some additional mechanics to model basic properties of spherical astronomy. Astrolabes are based on the apparent motion of celestial bodies, i. e., the observation that the paths described by stars in the sky are basically circles. Thus, the most common type of astrolabe is the *planispheric astrolabe*, developed in medieval times, which projects the firmament to the equatorial plane.

Using such an instrument it is possible to determine the position of some celestial bodies at a given time. As a navigational tool the planispheric astrolabe is far too imprecise. Nevertheless, it has been used to roughly located the stars for getting navigational fixes. Detailed information about astrolabes can be found in [Dodd 1969] and [J. E. Morrison 2007].

2.2 The Antikythera mechanism

More than 120 years ago, in 1900, sponge divers found a lump of corroded gears in a Roman ship wreck, which carried treasures from Greece dating back to about 100 B. C. It turned out that these were the remains of one of the most complicated mechanical and mathematical devices ever. Due to its location near the Greek island Antikythera (Αντικύθηρα) this impressive machine became known as the *Antikythera mechanism*. Figure 2.1 shows the main fragment of this early analog computer in its current state of preservation.[1]

1 Picture taken by Tilemahos Efthimiadis, protected by the *Creative Commons Attribution 2.0 Generic license.*

https://doi.org/10.1515/9783110787740-002

Fig. 2.1. Main fragment of the Antikythera mechanism as displayed in the National Archaeological Museum, Athens, Greece

Intrigued by this find, Derek de Solla Price[2] started investigating the inner workings of this device and summarized his astonishing discoveries as follows:[3]

> "*It is a bit frightening to know that just before the fall of their great civilization the ancient Greeks had come so close to our age, not only in their thought, but also in their scientific technology.*"

It turned out that the Antikythera mechanism was ahead of its time by at least 1 000 years. It is of such high complexity that recent research using modern X-ray tomography techniques, etc.[4] continues to deliver new insights. New capabilities and details were discovered as late as in 2021/2022.[5] The device modelled the movements of several celestial bodies, even taking into account various anomalies,

2 01/22/1922–09/03/1983
3 See [Freeth 2008, p. 7].
4 Cf. http://www.antikythera-mechanism.gr/.
5 See [Freeth et al. 2021] and [Freeth 2022].

which required differential gears[6] and much more complicated epicyclic gearing. This astonishing complexity of the Antikythera mechanism led Mike Edmunds[7] to the following statement:

> "*Nothing as sophisticated and complex is known for another thousand years. This machine rewrites the history of technology. It is a witness to a revolution in human thought.*"

This intricate mechanism allowed the calculation of sun and moon positions at given dates and phases of the moon as well as the prediction of solar and lunar eclipses.[8] The implementation of these functions required more than 30 gears, manufactured with extraordinary precision.[9]

2.3 Slide rules

One of the most common, simplest and well-known analog computers is the *slide rule*,[10] which comes in basically two configurations, *linear*, *circular*, and helical.[11] The basic idea of a slide rule is to reduce the problem of multiplication and division to that of addition and subtraction by employing logarithmically divided scales that may be displaced accordingly to each other in a lateral direction. The analog setup in this case is to mechanize the relation

$$\log(ab) = \log(a) + \log(b)$$

by using two logarithmic scales. Following the development of the logarithm by John Napier[12] and Henry Briggs,[13] who introduced the base 10 for logarithms, it was William Oughtred,[14] who described the principle of the slide rule in his seminal two publications *The Circles of Proportion, and the Horizontall Instru-*

6 Prior to this discovery differential gears were thought to have been invented in medieval times.
7 See [Freeth 2008, p. 9].
8 There are still arguments whether the mechanism also featured indicators for the display of planet positions.
9 A wealth of information about this device may be found in [de Solla Price 1974], [Freeth 2008] and [McCarthy 2009].
10 Also known as a *slipstick*.
11 Additional precision is achieved by this type of slide rule by wrapping extended scales along a helical path. The downside of this is that such slide rules typically feature only two scales.
12 1550–04/03/1617
13 February 1561–01/26/1630
14 03/05/1574–06/30/1660

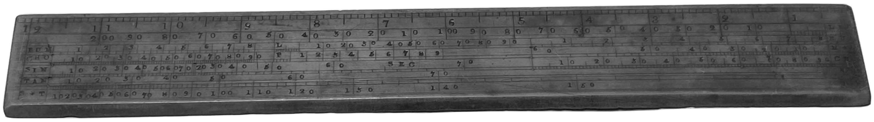


Fig. 2.2. Typical scale (probably 18th century)

ment and *Two rulers of proportion.*[15] An early *scale* is shown in figure 2.2. Working with such a scale required a divider to transfer lengths from one of its engraved scales to another, a tedious and error-prone process that was greatly simplified by the introduction of a slider containing several scales, which led to the then ubiquitous slide rule.

Figure 2.3 shows one of the last, most complex and most versatile slide rules ever built, a Faber Castell 2/83N. Its three main parts are clearly visible:

– The *body*, which consists of the top and bottom *stator* or *stock*. The two stators are held together by two *end braces* or *end brackets*.
– The *center slide*, which can be moved laterally with respect to the body.
– The *cursor*, which slides in grooves of the body.

While simple slide rules only feature a couple of scales, complex ones as the 2/83N have up to 30 and more scales, which implement functions far beyond multiplication and division.[16] Using these scales, trigonometric functions, exponentiation, etc., can be evaluated. Until pocket calculators took over in the 1970s,[17] slide rules were as widely used as they were centuries ago. The following quotation from Josef Vojtěch Sedláček[18] shows this quite strikingly:[19]

15 A comprehensive history of the slide rule may be found in [Cajori 1994] and [Jezierski 2000]. A great introduction to the application and use of slide rules is given in [Hume et al. 2005].

16 If all scales are just on one side of the body, a slide rule is called *simplex*. If scales are found on both sides of the body, it is a *duplex* slide rule. In this case the cursor is also used to transfer partial results from one side of the body to the other.

17 Early pocket calculators such as the Hewlett Packard HP-35 or some models made by Texas Instruments like the SR-10, etc., were explicitly marketed as *electronic slide rules*. The fixed number format featured by early HP calculators that was often set to display only 2 or 4 decimal places also was a reverence for the slide rule.

18 02/24/1785–02/02/1836

19 Cf. [Jezierski 2000, p. 16].

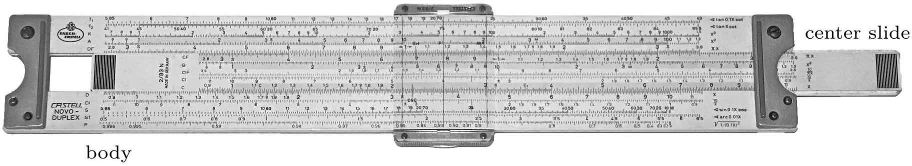


Fig. 2.3. Faber Castell slide rule model 2/83N

> "*It is said that the use of the slide rule in England is so widespread that no tailor makes a pair of trousers without including a pocket just for carrying a 'sliding rule'. During such a time, it is difficult to understand why the slide rule does not enjoy such well-deserved recognition in our own country.*"

Slide rules were essential tools for the scientific and technological progress of the last three centuries ranging from mathematics, civil engineering, commercial applications, electronics, chemistry, life sciences, etc., up to applications in aerospace technology.[20,21]

Although they were rendered more or less obsolete by pocket calculators 50 years ago, there are still areas of application where slide rules are employed regularly. For example, many aviators still use a flight computer like the E-6B, a special form of a circular slide rule, that allows the calculation of ground speed, i. e., the speed of an aircraft corrected for wind effects, and many other crucial parameters.[22]

Figure 2.4 shows a strange special purpose circular slide rule that was deployed in large amounts during the Cold War – a *Nuclear Weapon Effects Computer*, which allowed rough estimates of fatalities and damage should a nuclear air burst occur.

2.4 Planimeters

Planimeters are fascinating instruments. Their purpose, most aptly described by [Henrici 1894, p. 497], is the following:

20 In fact Buzz Aldrin (01/20/1930–) carried a Picket slide rule on the Apollo-11 mission, which was sold on September 20th, 2007 for $77,675.

21 [Kaufmann et al. 1955] describes some interesting electronic circuits, most of which requiring only passive components such as potentiometers and resistors, to implement an *electronic slide rule*.

22 Apart from the ease of use, such specialized slide rules have the advantage of not requiring any electrical power or the like for their operation.

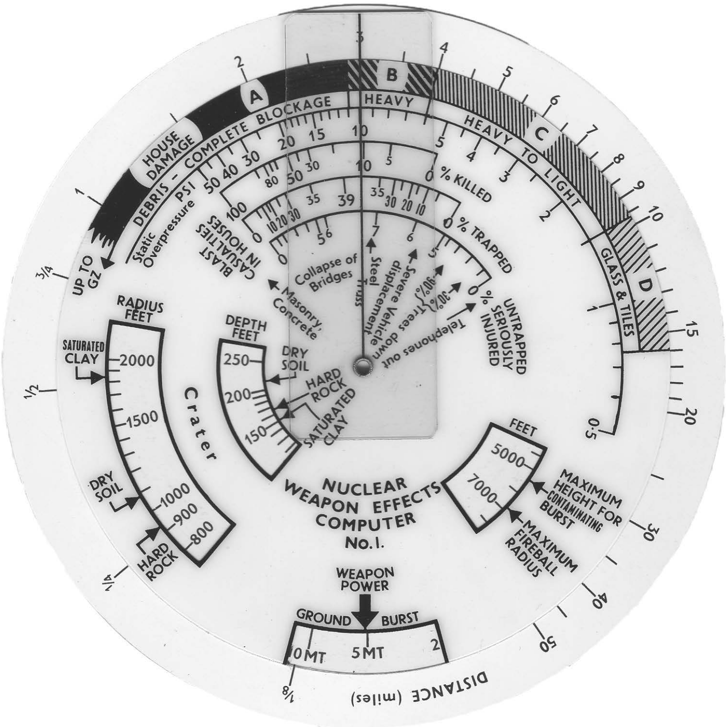


Fig. 2.4. Nuclear Weapon Effects Computer

> “*The object of a planimeter is to measure an area; it has, therefore, to solve a geometrical problem by mechanical means.*”

Measuring areas enclosed by some “good-natured” boundary curve is an important task in many branches of science as well as in commercial applications, registers of real estate and many more. A typical early application was to analyse pressure/volume indicator diagrams[23] as those written by recording steam engine indicators,[24] which requires the determination of the area enclosed by a curve, which is either plotted in a Cartesian or more often a polar coordinate system. A simple and direct method for performing this task is to cut out the area to be determined and weigh the resulting piece of paper yielding quite good results. Although this method is sometimes still used by chemistry students who regularly

23 See [Hütte 1926, pp. 380 f.].

24 The first of these devices was invented by James Watt's[25] assistant John Southern[26] around 1796 (cf. [Miller 2011]).

have to determine integrals over curves generated by spectrometers and the like, it is not really suitable for everyday usage.

As early as 1814 J. M. Hermann,[27] a Bavarian engineer, invented a planimeter that was built, after improvement by Lämmle, in about 1817. Unfortunately, this instrument seems to have gone unnoticed by his contemporaries and had no obvious influence on subsequent developments.[28] In 1824 an Italian professor for mathematics, Tito Gonnella,[29] invented a *wheel-and-cone planimeter* that used a friction-wheel integrator (see section 2.5.3) to perform the necessary integration.[30] The first planimeter that was put into production was a device developed by a Swiss engineer named Johannes Oppikofer,[31] who developed two wheel-and-cone planimeters in 1827 and 1836 and a planimeter based on a friction-wheel rolling on a disk in 1849. In fact, there is a plethora of different planimeter principles and implementation variants.

The most successful type of planimeter is the *polar planimeter* that was developed in 1854 by the Swiss mathematician Jacob Amsler-Laffon.[32] Figure 2.5 shows a typical polar planimeter,[33] which is of a much simpler construction than most of the other instrument types.[34]

The basis of operation for planimeters in general is Green's theorem, which relates a double integral over a closed region, i. e., the area to be determined, to a line integral over the boundary of this region.[35] Thus, a planimeter is a mechanization of

$$\iint \left(\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y}\right) \mathrm{d}A = \oint F \mathrm{d}r.$$

Choosing P and Q in a way that the difference under the left integral equals one yields the area sought. Interestingly, it seems that the first explanation of the operation of planimeters using Green's formula wasn't given until [Ascoli 1947].[36]

27 1785–1841

28 Cf. [Henrici 1894, p. 505].

29 1794–1867

30 [Haeberlin et al.2011] describes this instrument. See also [Henrici 1894, p. 500].

31 09/15/1782–04/21/1864

32 11/11/1823–01/03/1912

33 [Foote et al. 2007] shows how to build a simple polar planimeter.

34 One notable exception is the *Prytz planimeter* (also known due to its physical shape as a *hatchet planimeter*) that was developed by the Danish mathematician and cavalry officer Holger Prytz (who published under the pseudonym "Z") around 1875. His instrument has no moving parts whatsoever but is of very limited precision. A good description of the principle of operation of this instrument can be found in [Foote et al. 2007, pp. 82 ff.].

35 Cf. [Asmar et al. 2018, pp. 177 ff].

36 Explanations of the operation of planimeters with different chains of reasoning can be found in [Henrici 1894, pp. 179 ff.], [Meyer zur Capellen 1949], and [Leise 2007]. Some patents describing interesting planimeter designs are [Coffin 1882], [Snow 1930], etc.

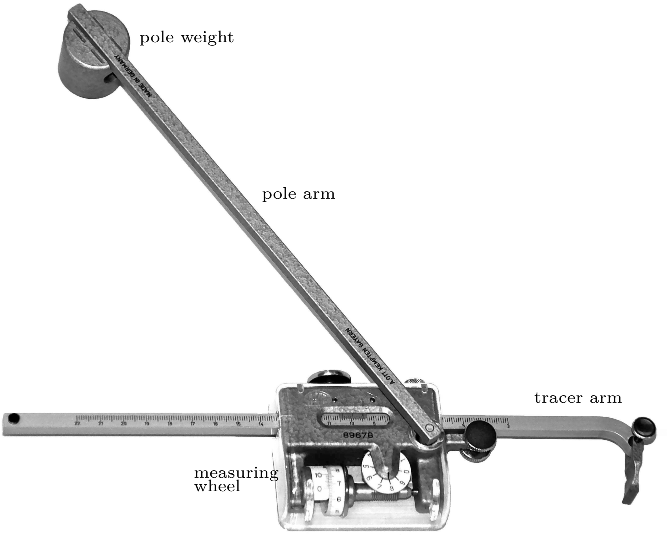


Fig. 2.5. Typical polar planimeter made by A. OTT, Bavaria, Germany

The main parts of a polar planimeter like the one shown in figure 2.5 are two arms connected with an elbow that is restricted to move on the circumference of a circle. The end of one arm, the *pole arm*, is fixed at a point called *pole* by means of a *pole weight* or a needle, while the end of the other arm, the *tracer arm*, can be moved freely by hand. This end is often formed as a needle or a magnifying lens to simplify tracing the boundary curve of the area to be measured. At the elbow is the integrating or measuring wheel.[37]

To measure the area of a circumscribed figure, the pole is fixed either outside or inside the figure, depending on its size. Then the measuring wheel with its vernier and the counting dial are reset with the needle or lens placed on the starting point of the boundary. Then the boundary line is followed manually in clockwise direction. After reaching the starting point again, the area can be read from the counting dial, the wheel and the vernier. If extended precision is necessary, the same procedure can be applied a second time while following the boundary line counterclockwise,[38] which, of course, yields a result with the opposite sign, the complement.

37 This wheel is extremely delicate – its metal wheel tread should never be touched by hand since it is engraved with microscopic rills, which are vital for the overall precision of the instrument and are easily damaged or cluttered with dirt.

38 Typical errors caused by a wheel with an axis not being perfectly horizontal can thus be compensated to a certain degree.

By setting the position of the elbow, which can normally be shifted along the tracer arm, various scaling values can be set, so areas can be measured in inches as well as in centimeters without the necessity to explicitly convert units.[39]

Determining large areas can be problematic since the counting dial can only represent one significant digit. Thus, as early as 1961 planimeters coupled with electronic counters were built. [Zuse Z80 1961] describes a linear planimeter[40] that was coupled to an electronic counter capable of processing up to 250 000 impulses per second. The pickup from the measuring wheel was done photoelectrically thus allowing even better precision than traditional mechanical instruments.

[Lewin 1972] describes another development that was patented in 1972: Here the position of the tracer arm is sensed by two linear potentiometers generating voltages directly proportional to the current (x, y)-position of the tracer. These voltages in turn control an oscillator and some monostable devices. The integration process is then performed basically by a chain of decade counters. This scheme proved to be too complicated and costly for the market. Another quite recent development is described by [Light 1975]. Here the position of a tracer needle or the like is determined by conductive foils, which are placed under the chart containing the curve to be integrated over.[41]

As old as planimeters are, some companies still manufacture polar and linear planimeters, which achieve accuracies of about 1‰. These instruments are still used regularly for surveying and mapping, for determining the area of furs and fabric, etc.

2.5 Mechanical computing elements

All of the instruments shown in the preceding sections are specialized analog computers, capable only of solving just one distinct problem each. This is obviously caused by their fixed structure – a slide rule can only add lengths, so the only variation possible is that of employing different scales to extend this basic operation to multiplication and division and many more.

In the same sense planimeters are specialized instruments for only determining areas. The following sections will now cover some basic mechanical computing

39 [Palm 2014] gives a thorough treatment of the planimeters and other integrators made by A. Ott.

40 These differ from polar planimeters in so far as the tracer arm is not free to rotate around the elbow connected to the pole arm but is guided in a strictly linear fashion, which is normally implemented by a two-wheel carriage that is dragged behind by the tracer arm while tracing the curve.

41 Although this instrument was not a financial success due to its complexity, its pickup mechanism anticipated the basic techniques used for today's touch screens and the like.

elements that can be used to build true – in the sense of their reconfigurability – analog computers, *differential analysers*.[42]

Typically, mechanical analog computers represent values by rotations of shafts, which interconnect the various computing elements or by positions of linkages. As simple as most mechanical computing elements seem, they are quite powerful tools. Precisions up to 10^{-4} are possible given precise machining of the parts involved and clever scaling of the equations to be solved. In fact, mechanical analog computers even have one advantage over analog electronic analog computers: The former can integrate over every variable whilst the latter can only integrate over time, which then requires some ingenuity to solve partial differential equations[43] and other problems.

2.5.1 Function generation

A common task in simulations is the generation of functions, which are either defined analytically or by measured values. Functions of a single variable are easily implemented using cams driven by a shaft, the angular position of which represents the input variable. A follower measuring the position of the cam's surface yields the desired function value. The position of this pin can be translated into a rotational motion by a rack and pinion arrangement so this output value can be used as input for another element expecting an angular shaft position as its input value.

Another type of generator for a function of one variable is shown in figure 2.6. This is a *squaring cam*, which is used to generate a square function $f(x) = x^2$. If $\mathrm{d}x_1$ denotes the element of rotation of the input shaft, the resulting movement of the string wound around the cone is $r_1\mathrm{d}x_1$ where r_1 denotes the average diameter of the cone at the current position of the string. With r_2 denoting the diameter of the drum, the drum rotation resulting from $\mathrm{d}x_1$ is

$$\mathrm{d}x_2 = -\frac{r_1\mathrm{d}x_1}{r_2}.$$

r_1 is proportional to the angle of rotation x_1 thus

$$\mathrm{d}x_2 = -\frac{kx_1\mathrm{d}x_1}{r_2}$$

with k being a constant of proportionality. Integration finally yields

$$x_2 = -\frac{k}{2r_2}x_1^2,$$

42 Information about mechanical computing elements can be found in [Svoboda 1948], [Meyer zur Capellen 1949], [Willers 1943], and [Bureau of Ordnance 1940].

43 This particular advantage of mechanical analog computers is also exhibited by digital analog computers, see chapter 10, which make these devices quite interesting for future computer architectures.

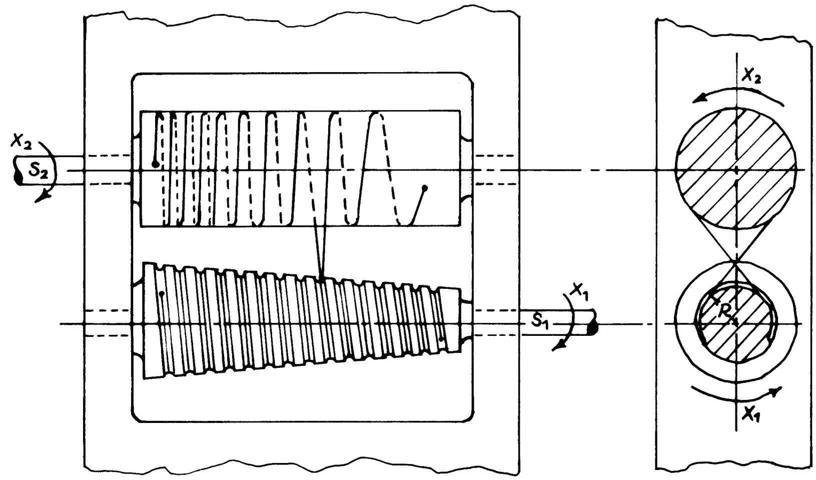


Fig. 2.6. Squaring cam yielding $x_2 = x^2$ (cf. [SVOBODA 1948, p. 22])

which is the desired square function. This mechanism does, of course, not work down to $x_1 = 0$ – if this is necessary, it may be combined with a differential gear as described in section 2.5.2.[44]

In many cases, functions of two variables are necessary – figures 2.7 and 2.8 show a *barrel cam*[45] used to implement a function $x_3 = f(x_1, x_2)$. The value of the input variable x_1 controls the lateral displacement of the three-dimensional barrel cam while the second input variable, x_2, controls the angular position of the cam. The output value is sensed by a follower that gauges the cam's surface. Other implementations feature a movable sensing pin carriage instead of a barrel cam that can be shifted laterally, which allows for a more compact design.

Barrel cams like this were often used in mechanical fire control systems but were also used in some cases in electronic analog computers since the generation of functions of two variables is a complicated task for such a machine. In this case the input shafts for x_1 and x_2 are controlled by electronic servo mechanisms while the output sensing pin is connected to a potentiometer, which delivers a voltage proportional to the desired function value.[46]

An important topic regarding cams and barrel cams is that their surface has to conform to several mechanical constraints to ensure that the sensing pin can follow the surface smoothly and that the pin can never block the cam in its movement due

44 Cf. [SVOBODA 1948, p. 22].
45 Three-dimensional cams like this have also been called *camoids*, cf. [SVOBODA 1948, p. 23].
46 See section 4.5.7.

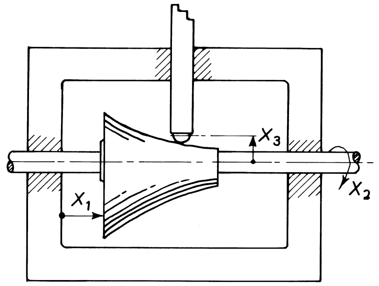


Fig. 2.7. Three-dimensional cam as a function generator for a function of two variables $f(x, y)$ (cf. [Svoboda 1948, p. 23])

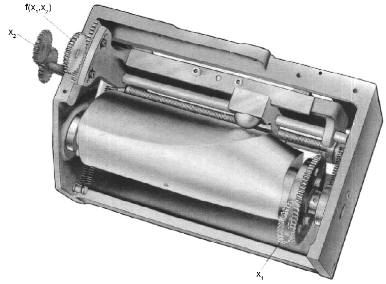


Fig. 2.8. The barrel cam – a practical implementation of a function generator yielding $f(x_1, x_2)$ (cf. [Bureau of Ordnance 1940, p. 54])

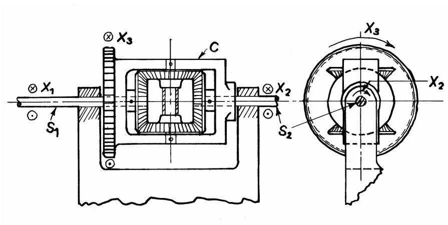


Fig. 2.9. Structure of a bevel-gear differential (cf. [Svoboda 1948, p. 6])

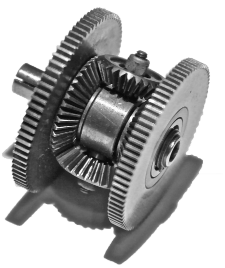


Fig. 2.10. A differential gear from the dead reckoning computer PHI-4A-2 used in Starfighter jets

to steep slopes or grooves. This might sometimes conflict with the mathematical requirements of the computer setup. In such cases, functions may be generated by solving suitable differential equations, which normally requires many more additional computing elements than a straightforward barrel cam implementation.

2.5.2 Differential gears

A barrel cam function generator like this could, in principle, be used to add or subtract two variables but a simpler, cheaper and more precise mechanism to implement this basic operation exists in form of the *bevel-gear differential*[47] shown in figure 2.9. This device adds two shaft rotations x_1 and x_2 with some scaling

47 Such building blocks are also called *additive* or *linear* cells (cf. [Svoboda 1948, p. 6]).

parameters s_1 and s_2, which result from the actual design of the differential, yielding $x_3 = s_1 x_1 + s_2 x_2$.

A practical example of such a differential gear is shown in figure 2.10. This was used in the dead reckoning computer PHI-4A-2 of a Starfighter jet.[48]

2.5.3 Integrators

In contrast to most other machines, integration is a fundamental as well as natural operation for an analog computer. Using mechanical computing elements, the operating principles of integration are remarkably simple.[49] The first mechanical integrators were developed in the early 19th century: In 1814 JOHANN MARTIN HERMANN developed an integrator consisting of a cone with a wheel rolling on its surface.[50] The position of the wheel on the envelope of the cone represents the values of the function to be integrated while the rotation of the cone corresponds to the variable of integration.[51]

Basically an integrator mechanizes the calculation of integrals like

$$x_3 = \int x_1 \, \mathrm{d}x_2$$

where x_2 is represented by the rotation of the integrator disk (or cone) while the values of x_1 control the position of the friction-wheel rolling on the disk surface. Figure 2.11 shows the structure of such an integrator.

Obviously it is $\mathrm{d}x_3 = kx_1 \mathrm{d}x_1$ where $k = 1/r$ denotes the radius of the friction-wheel thus yielding

$$x_3 = \int_{x_{20}}^{x_2} \frac{1}{r} x_1 \, \mathrm{d}x_2.$$

Figure 2.12 shows a friction-wheel integrator from the *Oslo analyser*, which was built from 1938–1942 – at its time one of the largest, most precise and most

48 The central element carrying the bevel gears is called *spider block*.

49 Far from being trivial is the implementation of integrators due to the necessary high precision in order to minimise the accumulation of errors within a simulation.

50 See [PETZOLD 1992, p. 26].

51 A similar mechanism was developed later by TITO GONNELLA and used as the basis for his planimeter.

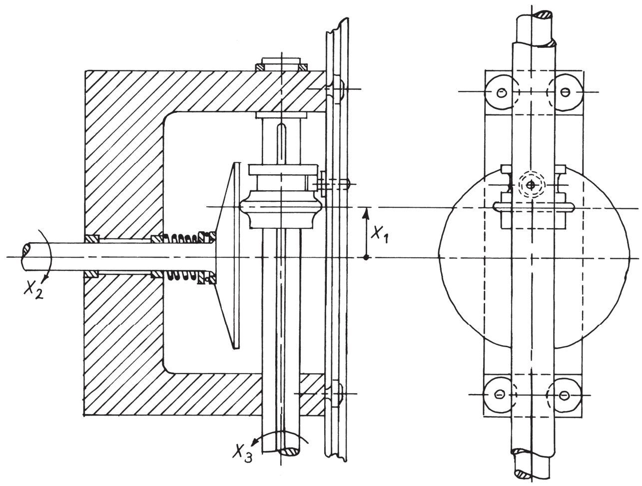


Fig. 2.11. A friction-wheel integrator (cf. [Svoboda 1948, p. 24])

powerful differential analysers in the world.[52] On the right hand side[53] the rotating disk S and the friction-wheel h with its associated shaft are visible. The wheel position on the surface of the disk with respect to the disk center corresponds to $F(x)$.

In 1876, James Thomson,[54] the brother of William Thomson[55] – later Lord Kelvin – developed an integrator that replaced the friction-wheel with a steel sphere that runs in a movable cage (controlled by x_1).[56] This sphere provides a frictional connection between the rotating disk and a cylinder that acts as a pickup for the integration result x_3. Figure 2.13 shows a variation of this implementation – two stacked balls, instead of a single sphere, are guided in a movable cage. Such integrators are known as *double-ball integrators*.[57]

52 This particular machine featured twelve integrators of this type and was used until 1954, see section 2.8.

53 On the left side a *torque amplifier* and a *frontlash unit* can be seen. These are typical devices in a mechanical differential analyser but outside the scope of this book. Refer to [Willers 1943, pp. 236 ff.], [Robinson] or [Fifer 1961, p. 672] for more information about these units.

54 02/16/1822–05/08/1892

55 06/26/1824–12/17/1907

56 See [Thomson 1912, pp. 452 ff.].

57 Such double-ball integrators were used well into the second half of the 20th century. This was mainly due to their high precision – [Librascope 1957] mentions a repeatability of 10^{-4} – quite remarkable for a mechanical device – as well as to their robustness. Accordingly, such integrators were used quite often in aerospace applications like dead reckoning computers, etc.

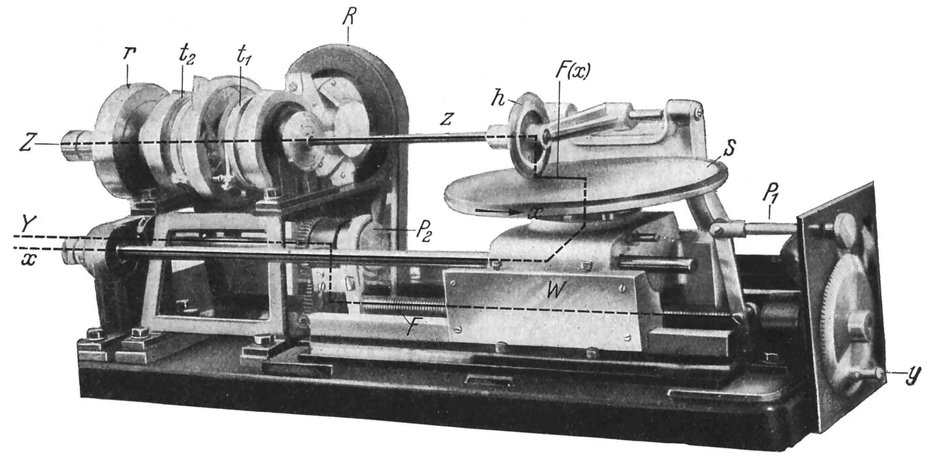


Fig. 2.12. Integrator from the Oslo differential analyser, see section 2.8 ([WILLERS 1943, p. 237])

Based on this integrator developed by his brother, Lord KELVIN devised the idea of a machine suitable for solving differential equations. The basic concept was to start with the highest derivative in the differential equation to be solved and to generate all of the necessary lower derivatives by repeated integration. Using these derivatives, the highest derivative – that was the starting point for this procedure – can now be synthesized by combining the lower derivatives in an appropriate way.[58] The discovery of this method is described in [THOMSON 1876] as follows:

> "*But then came a pleasing surprise. Compel agreement between the function fed into the double machine and that given out by it [...] The motion of each will thus be necessarily a solution of [the equation to be solved]. Thus I was led to the conclusion, which was unexpected; and it seems to me very remarkable that the general differential equation of the second order with variable coefficients may be rigorously, continuously, and in a single process solved by a machine.*"

Interestingly, KELVIN did not build a practical machine based on this insight, which is all the more puzzling since a (double) ball integrator can transfer sufficient torque to allow a small number of such devices to be chained. Trying to implement this scheme using friction-wheel integrators would have required torque amplifiers, as these integrators can only transmit tiny amounts of torque. Consequently, this brilliant idea of setting up an analog to represent differential equations fell into oblivion. This happened again 45 years later when UDO KNORR[59] published a

58 This *classical differential analyser technique*, as [KORN et al. 1964, p. 1-5] puts it, is described in more detail in section 7.2.
59 04/20/1887–07/10/1960

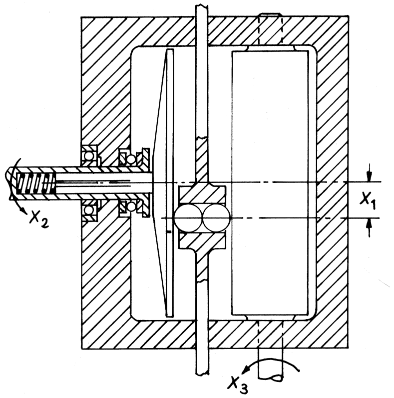


Fig. 2.13. Principle of operation of a double-ball integrator (see [SVOBODA 1948, p. 25])

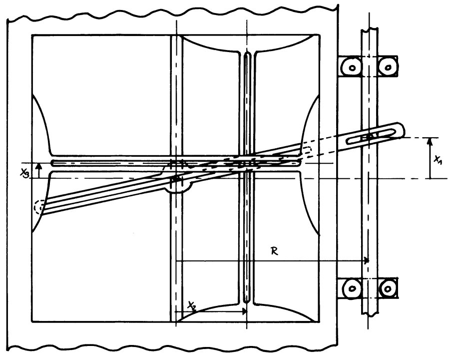


Fig. 2.14. Basic structure of a slide multiplier (cf. [SVOBODA 1948, p. 12])

similar idea in 1921.[60] KNORR explicitly noted that coupled integrators can be used to solve a large group of differential equations of high degree.[61]

2.5.4 Multipliers

Multiplication is a quite difficult task for a mechanical analog computer. While multiplication with a constant can be easily performed with an appropriate gear mechanism or with a friction-wheel integrator where the position of the friction-wheel corresponds to the multiplier, the generalized operation where both input variables may vary is much more difficult to implement.

One common implementation is based on the product rule known from calculus. From $(uv)' = u'v + uv'$ it follows by integration that

$$uv = \int u\,\mathrm{d}v + \int v\,\mathrm{d}u. \tag{2.1}$$

Thus, the multiplication of two variables can be implemented using two interconnected integrators.[62]

60 Cf. [PETZOLD 1992, p. 33].

61 See [WALTHER et al. 1949, p. 200].

62 This requires that the integrators are not restricted with respect to the variable of integration. Only mechanical analog computers and DDAs (cf. section 10) fulfill this requirement. Analog electronic analog computers can only use time as the variable of integration, so this class of machines requires different multiplication schemes.

Figure 2.14 shows the structure of a *slide multiplier*. It consists of two carriages sliding vertically and horizontally respectively and a lever that rotates around a center pin that is fixed to the enclosure. Both carriages and the lever are coupled with another pin that runs in grooves of these three elements. The input variables are x_1, represented by the displacement of the rotating lever's end, and x_2, which corresponds to the horizontal displacement of the second slider. The multiplication result $x_3 = kx_1x_2$ where k denotes a scaling factor that depends on the dimensions of the multiplier, is then available as the vertical displacement of the first slider.

2.6 Harmonic synthesizers and analysers

Although Lord Kelvin did not attempt to build a true general purpose mechanical analog computer, he did build some quite complex special purpose analog computers[63] to predict tides by means of harmonic synthesis.[64] Even the earliest sailors had a genuine interest in tide prediction since accurate predictions lead to fewer lay days in harbours. A good description of the term *tide* is given by Kelvin himself:[65]

> "*The tides have something to do with motion of the sea. Rise and fall of the sea is sometimes called a tide; but I see, in the Admiralty Chart of the Firth of Clyde, the whole space between Ailsa Craig and the Ayrshire coast marked 'very little tide here'. Now, we find there a good ten feet rise and fall, and yet we are authoritatively told there is very little tide. The truth is, the word 'tide' as used by sailors at sea means horizontal motion of water; but when used by landsmen or sailors in port, it means vertical motion of the water.*"

Tides are caused and influenced by the superposition of a number of effects that are, in fact, harmonic oscillations with different amplitudes, frequencies, and phases.[66] The most important of these effects are the earth's rotation around its own axis, the rotation of the earth around the sun, the rotation of the moon around the earth, the precession of the moon's perigee, the precession of the plane of the moon's orbit, etc.

In 1872 Kelvin developed the first *harmonic synthesizer*, a specialized analog computer for generating harmonics and adding these together to generate a

63 For these machines the phrase "*substitute brass for brain*" was coined (see [Zachary 1999, p. 49] and [Everyday Science and Mechanics 1932]).

64 Following the death of his first wife, Margaret Thomson nee Crum on June 17, 1870, his interest for seafaring increased and he bought a 126 ton schooner, the Lalla Rookh, which in turn sparked his interest in tide prediction.

65 See [Thomson 1882, Part I].

66 These effects are called *partial tides*.

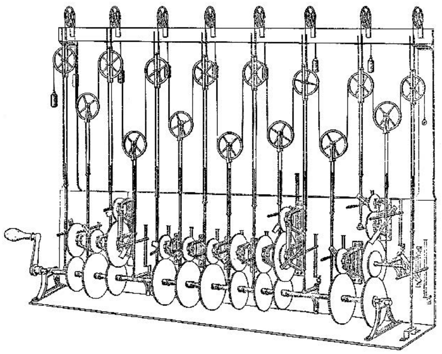


Fig. 2.15. Principle of operation of KELVIN's tide predictor (see [THOMSON 1911])

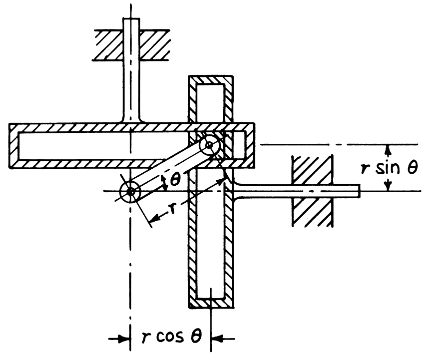


Fig. 2.16. A scotch yoke mechanism (cf. [KARPLUS et al. 1958, p. 242])

tide prediction.[67] This machine took ten partial tides into account. Its basic construction is shown in figure 2.15.[68] Using this machine it took only four hours to compute the 1 400 tides that occur during a year for a given harbour. Performing the same task manually took several months.[69]

Generally speaking, a harmonic synthesizer like a tide predictor generates a function $f(x)$ based on a given set of harmonics:[70]

$$f(x) = a_0 + a_1 \sin(x + b_1) + a_2 \sin(2x + b_2) + \cdots + a_n \sin(nx + b_n)$$

The basic harmonic functions are traditionally generated using a *scotch yoke mechanism* as shown in figure 2.16. It consists of a boom that is mounted on a shaft so that it can rotate around this mounting point. The angular position of this shaft represents the input variable θ. The free end of the boom guides two carriages that are restricted to perform horizontal respectively vertical movements only. The movements of these carriages then represent the values $r\sin(\theta)$ and $r\cos(\theta)$ where r denotes the radius of the circle described by the free end of the rotating boom.

Since the values $r\cos(\theta)$ and $r\sin(\theta)$ are represented by linear displacements, harmonics generated this way cannot be added together using a differential gear. Instead, a steel band of constant length is used, as can be seen in figure 2.15.

67 Lord KELVIN's machine is on display at the Science Museum, London.

68 Later machines generated even more partial tides. The *United States Coast and Geodetic tide-predicting machine No. 2* that was completed in 1910 generated 37 harmonic terms. [AUDE et al. 1936] describes a machine that took 62 harmonics into account (this machine was in operation in Hamburg, Germany, until 1968 and is now on display at the *Deutsches Museum* in Munich).

69 Cf. [SAUER].

70 Cf. [BERKELEY et al. 1956, pp. 135 ff.].