Acta Technica Jaurinensis

Győr, Transactions on Engineering

Vol. 3, No. 1

The Historical Development of Thermodynamics

D. Bozsaky

"Széchenyi István" University Department of Architecture and Building Construction, H-9026 Győr, Egyetem tér 1.
Phone: +36(96)-503-454, fax: +36(96)-613-595
e-mail: bozsaky@gmail.com

Abstract: Thermodynamics as a wide branch of physics had a long historical development from the ancient times to the 20th century. The invention of the thermometer was the first important step that made possible to formulate the first precise speculations on heat.

There were no exact theories about the nature of heat for a long time and even the majority of the scientific world in the 18th and the early 19th century viewed heat as a substance and the representatives of the Kinetic Theory were rejected and stayed in the background. The Caloric Theory successfully explained plenty of natural phenomena like gas laws and heat transfer and it was impossible to refute it until the 1850s when the Principle of Conservation of Energy was introduced (Mayer, Joule, Helmholtz).

The Second Law of Thermodynamics was discovered soon after that explanation of the tendency of thermodynamic processes and the heat loss of useful heat. The Kinetic Theory of Gases motivated the scientists to introduce the concept of entropy that was a basis to formulate the laws of thermodynamics in a perfect mathematical form and founded a new branch of physics called statistical thermodynamics.

The Third Law of Thermodynamics was discovered in the beginning of the 20th century after introducing the concept of thermodynamic potentials and the absolute temperature scale. At the same period of time the scientific issue of thermal radiation was also solved.

Keywords: history, physics, thermodynamics, heat transfer, thermal radiation

1. The invention of the thermometer [4][5][9][11][12]

The first important step to discover the principle of thermodynamics was the invention of the thermometer because precise and reliable survey results were needed. In the Ancient Times scientists wanted to measure the attributes of substances including their temperature.

Philo of Byzantium (280 BC – 220 BC) reported in his manuscript about a heat-sensing instrument. He constructed tube with a hollow sphere that was extended over a jug of

water. When the sphere was placed in the sun the water began to bubble as the air expanded out of the sphere. If he put it in the shade the water rose in the tube as the air contracted in the sphere. *Hero of Alexandria* (10 AD – 70 AD) also inspected that the water level in a container rises and sinks due to the change in temperature.

In the Middle Ages scientists and physicians raised the necessity of measuring temperature. They knew that the flame has higher intensity of heat than a hot piece of iron while the quantity of heat is much lower in it but they could not clearly define the difference between temperature and quantity of heat

The Persian polymath *Avicenna* (980-1037) also recorded that he knew a mechanism to show the hotness and coldness of the air and developed an instrument in which the water level was controlled by the contraction and expansion of air but the really improvement came in Europe in the 16th century.

The Italian *Galileo Galilei* (1564-1642) created the first thermometer in 1597 which was really a thermoscope because it did not have numerical scale so Galilei could find out only the relative differences between air temperature. Scientists in the 17th century constructed lots of thermometers (Sagredo, Santorio, Fludd, Drebbel) but they all suffered from the disadvantage that they were also barometers. In 1654 *Ferdinando II de' Medici* (1610-1670), Grand Duke of Tuscany made a thermometer of sealed tube filled with alcohol that was only sensible to temperature and it was independent of air pressure.

The Englishman, *Robert Boyle* (1627-1691) was the first who realized the necessity of standard scales in 1662 during his experiments with that he discovered his law (Boyle's Law) that describes the relationship between the absolute pressure and volume of gas if the temperature is kept constant within a closed system.

In 1665 *Christiaan Huygens* (1629-1695) suggested to use the melting and boiling point of water as a standard scale and in 1694 *Carlo Renaldini* (1615-1698) proposed to use them as fix points with twelve equal parts between them but it was not accepted immediately because scholars were unsure that the freezing and boiling points of water are constant.

In 1724 the German physicist and glassblower *Daniel Gabriel Fahrenheit* (1686-1736) proposed a thermometer with reliable universal scale using mercury instead alcohol as the fluid within and it had three fix points. Zero was the coldest day of the winter in Danzig, the freezing point of the water was 32 degrees and the healthy human body temperature was 96 degrees that resulted 212 degrees for the boiling point of the water.

Fahrenheit's thermometer was the first standardised instrument that was suitable for scientific measurements. It was cleared that all substances have defined freezing and boiling points. Starting from this fact in 1742 the Swedish astronomer *Anders Celsius* (1701-1744) produced a thermometer with a standard scale using the melting point of water as zero and boiling point of water as 100 degrees. This scale bears his name and it is under use with Fahrenheit scale world-wide nowadays. Because of its simplicity the Celsius scale is more popular.

2. The first scientific issue: Is heat a substance or a motion? [4][5][9][11][12]

Ancient people related heat with flame and fire. The ancient Egyptians viewed it as a formation with mysterious origins. The Chinese Taoists believed that fire is one of the five principle elements like air, wood, metal and water.

Ancient Greeks generally viewed fire and heat as a substance and often connected it with life and motion. *Heraclitus* (535 BC – 475 BC) was the first who framed a theory on heat. He argued that there are three principle elements in nature – fire, water, earth – from which the fire is the central element controlling and modifying the other two. Heraclitus claimed that heat is connected with the motion because he observed that living creatures are warm and died bodies are cold. The later ancient scholars (*Empedocles, Aristotle*) believed in four principle elements (water, earth, air, fire) and they also connected the heat with life and coldness with death.

In the Middle Ages some Islamic scientists examined heat and fire and all of them connected it clearly with motion. *Abū Rayhān Bīrūnī* (973-1038) stated that the causes of heat are movement and friction. *Avicenna* and *Abd Allah Baydawi* (?-1286) also made similar discoveries that heat is generated from motion of external things and it may occur through motion-change.

Even all the scientists of the 17th century believed in the essential connection between heat and motion. The English philosopher, *Francis Bacon* (1561-1626) in his work called *Novum Organum* demonstrated that heat is a kind of motion. *Robert Boyle* and his colleague *Robert Hooke* (1635-1703) had comparable opinion that heat is nothing else but vehement motion of the elementary particles.

3. Roundabout ways: The Phlogiston and the Caloric Theory [5][9][11][12]

Now we would think that it led directly to the Kinetic Theory, but the level of the mathematical knowledge was not enough high to create satisfying answers to a lot of questions. This is why the theories on the material nature of heat became conspicuous because they were much more suitable for explaining the phenomena like melting heat, boiling heat, thermal radiation, heat transfer etc. In 1669 *Joachim Johann Becher* (1635-1682) established the Phlogiston Theory that was later developed by *Georg Ernst Stahl* (1659-1735). In his work entitled *Experimenta chymicae et physicae* (1731) proposed that heat was associated with an undetectable substance called phlogiston that was driven out of the material when it was burnt. The theory was finally refuted in 1783 by *Antoine-Laurent de Lavoisier* (1743-1794) proving the participation of oxygen in burning. He framed instead the Caloric Theory that saw heat as a weightless and invisible fluid that moves to hot bodies from the cold ones.

Herman Boerhaave (1668-1738) was the first who went to the very limits of the Caloric Theory. He pronounced that we can not make equal sign between heat, fire and light because they can manifest separately. Boerhaave supposed connection between heat and motion because rubbing together two parts of flint-stones fire came into being no matter how hot or cold they were. He tried to determine the weight of Caloricum and examined the phenomena of thermal expansion.

The concept of fire and heat became clear only in the middle of the 18th century as the Scottish physicist, *Joseph Black* (1728-1799), started his experiments at the Glasgow

University in the 1750s. He defined the difference between temperature and the quantity of heat and founded the concept of specific heat that is the measure of heat (or thermal energy) required to increase the temperature of a unit quantity of a substance by one unit.

Black's most important discovery was the observation that melting ice absorbs heat without changing temperature. From this recognition he came to a conclusion that ice needs latent heat for this modification of physical condition. It was the main substantial proof of the material nature of heat for him and in 1779 one of his students, *William Cleghorn* (1754-1783), formulated the precise definition of the Caloricum.

4. The most important results of the Caloric Theory: gas laws and heat transfer [1] [4] [5][9][11]

4.1. Gas Laws [1][9][10][11]

The reason for the long survival of the Caloric Theory was that it opened the door to obtain the gas laws and to explain the heat flow.

Based on Boyle's work *Guillaume Amontons* (1663-1705) made an accurate thermometer in 1695 and investigated the pressure and temperature of gases. He found that the pressure of gas increases by one third between the temperature of cold and boiling water. From this Amontons concluded that the reduction of temperature leads to the disappearance of pressure and with this statement he founded the theory of absolute zero of temperature.

Knowing the Caloric Theory *Jacques Alexander César Charles* (1746-1823) discovered in 1787 that at constant pressure the volume of gas increases or decreases by the same factor as its temperature. This theory was further developed by *Joseph Louis Gay-Lussac* (1778-1750) and in 1802 he published his law that the pressure of a gas of fixed volume is directly proportional to its temperature.

Then only one step was needed when in 1834 *Benoît Paul Émile Clapeyron* (1799-1864) formulated the Combined Gas Law and stated that the ratio between the pressure-volume product and the temperature of a gas remains constant.

Clapeyron could not calculate the value of this constant without the knowledge of Avogadro's Law and the absolute zero of temperature, but when these things were discovered and also accepted a decade later, the French chemist *Henri Victor Regnault* (1810-1878) created the Ideal Gas Law

$$pV = \frac{m}{M} R_0 T$$

where p is the absolute pressure of gas, V is volume, m is the mass, M is the molar mass, R_0 is the ideal gas constant and T is the absolute temperature.

4.2. The theory of heat transfer [3][5][9][10][11]

Even in 1686 *Edmund Halley* (1656-1742) identified the fact that warm air rises and realized that solar heating is the cause of atmospheric motions.

The first publication about heat transmission was written by *Isaac Newton* (1643-1727) in 1701 and stated that the rate of heat loss of a body is proportional to the difference in temperatures between the body and its surroundings. This law was not enough precise and it was further developed after the foundation of the laws of fluid mechanics. The first attempt to prove this law was made by *Pierre-Louis Dulong* (1785-1838) and *Alexis Thérèse Petit* (1791-1820) in 1817 and pointed that Newton's Law is correct only by low differences of temperature.

In the early 18th century it was not easy to see that all materials had determined conductivity of heat but when the new science of electricity appeared it became apparent that some materials were good conductors and others were effective insulators. In 1785 *Jan Ingen-Housz* (1730-1799) raised an idea that based on their electrical properties some materials might be good thermal conductors or thermal insulator too.

In 1777 *Carl Wilhelm Scheele* (1742-1786) distinguished the three forms of heat transfer from each other – the thermal radiation, thermal conduction and thermal convection – and lots of experiments began in the late 18th century about them. In 1804 *John Leslie* (1766-1832) observed that the cooling effect of stream is increasing with its speed. In the same year he carried out his famous experiments with the Leslie cube (see later). He was the first who artificially froze water into ice in 1810.

The most important result of the Caloric Theory is associated with the name of the French mathematician, *Jean Baptiste Joseph Fourier* (1768-1830). In 1807 he formulated his empirical law of heat conduction based on his observations. It states that the rate of heat flow through two surfaces at right angles of a homogenous solid in a unit of time is directly proportional to thermal conductivity (heat transfer coefficient, λ) and to the temperature difference along the path of the heat flow and inversely proportional to the distance between the ends of the crossed surfaces:

$$q = \lambda \frac{\Delta T}{\Delta x}$$

In this formula q is the heat flux, λ is the heat transfer coefficient, ΔT is the temperature difference between the ends and Δx is the difference between the ends.

Fourier's Law was not accepted for 15 years and it was finally published in 1822 in his monograph entitled *Théorie analytique de la chaleur* (The Analytic Theory of Heat). In this work Fourier summarized his most important discoveries and formulated own theory in a correct mathematical form by working out the differential form of thermal conduction with the help of Fourier series.

The decisive step in the application of Fourier's Law and the concept of heat transfer coefficient was taken by *Ernst Karl Wilhelm Nusselt* (1882-1957) when his paper called *Das Grundgesetz des Wärmeübergangs* (The Basic Law of Heat Transfer) was published in 1915.

5. The first attempts of the Kinetic Theory [4][9][11]

In spite of the rapid successes and propagation of the Caloric Theory there were a few scientists who took a stand for the Kinetic Theory of heat. In 1716 *Jakob Hermann* (1678-1733) pointed out that the atmospheric pressure is proportional to the air density and to the square of the average velocity of moving particles in atmosphere. *Leonhard Paul Euler* (1707-1783) even computed the value of this average velocity as 477 m/s.

In 1738 *Daniel Bernoulli* (1700-1782) published his most important work called *Hydrodynamique* (Hydrodynamics). Based on the relation of Boyle's Law showed that as temperature changes the pressure will change proportionally to the square of the particle velocities.

In 1745 the Russian chemist *Mikhail Vasilyevich Lomonosov* (1711-1765) also wrote a relevant work against Caloric Theory under the title of *Размышления о причине теплоты и холода* (Reflections on the Reason of Heat and Cold). He reported that heat is generated by motion because when we rub our hands together or strike the iron intensively they become warmer. He explained that heat is nothing else but the high-speed velocity of motion of invisible material particles. In his later works he tried to put into words the Principle of Conservation of Energy and diagnosed that however much matter is added to any body, as much is taken away from another.

In the 18th century works of these scientists about the Kinetic Theory created little stir throughout the world because of the huge popularity of the Caloric Theory. In addition there were lingual difficulties too, because Lomonosov published his works in Russian and they were not attractive in Western Europe.

Caloric Theory had only two weak points – the friction heat and the weight of the Caloricum – and a few practical researchers tried to take advantage of this situation. The cannon manufacturer *Benjamin Thomson, Count Rumford* (1753-1814) realized that the most suitable moments to take the weight of Caloricum when the ice is melting because at this moment ice absorbs a lot of heat without changing temperature. He took absolutely accurate and precise measurements with his apparatus and finally declared that even if Caloricum had weight it is immensely small.

In 1798 Rumford made a study about the frictional heat that was generated through boring the cannons. He immersed a cannon barrel in water and showed that the water could be boiled by the frictional heat generated by the boring tool. Rumford demonstrated through the use of friction that it was possible to convert work to heat and this heat seemed to be inexhaustible. As a result of these experiments Rumford suggested that heat is a form of motion.

The connection between heat and friction was also analysed in 1799 by *Humphrey Davy* (1778-1829). In his experiment he rubbed two pieces of insulated ice together and showed that melting heat could be originated only from mechanical work.

Rumford and Davy were very close to refute the Caloric Theory but the advocates of it could easily explain the results of their experiments supposing the weightlessness of Caloricum.

6. The devolution of the Caloric Theory and the recruitment of the Kinetic Theory [2][4][9][11]

The overthrow of the Caloric Theory became possible after the birth and verification of the Principle of Conservation of Energy. Although the foundations of this theory are findable in the work of Thales of Miletus and lots of others the first mathematical formula was created by *Gottfried Wilhelm Leibniz* (1646-1716) who noticed that in many mechanical systems a determinate quantity of vis viva (living force) is conserved.

Instead the name of vis viva *Thomas Young* (1773-1829) suggested to use the expression of energy in 1802 but he still used Leibniz's formula (mv^2) to calculate the quantity of it. Only in 1829 *Gaspard-Gustave de Coriolis* (1792-1843) recalibrated it to an appropriate formula of $½mv^2$ and named it as kinetic energy.

It was easy to understand the connection between mechanical work and kinetic energy but the verification of the mechanical equivalent of heat was much more difficult. The encouragement was brought by the steam engine that was already invented in the 17th century but it was improved by *James Watt* (1736-1819) only in 1769. The main problem with these machines was that they were slow and converted less than 2% of the invested fuel into useful work. It was immediate to enlarge the useful effect of steam engines that was urged by Watt.

About the first experiments and measurement on the enlargement of the useful effect was published in 1776 by the Scottish engineer *John Smeaton* (1724-1792) in which he supported the vis viva theory. Decades later *William Hyde Wollaston* (1766-1828) and *Peter Ewart* (1767-1842) also confirmed Smeaton's publication but they were attacked on the plea that they are in conflict with Newton's law on impulse.

An important step was presented in 1824 by the French engineer, *Nicolas Leonard Sadi Carnot* (1796-1832) who published his work under the title of *Réflexions sur la puissance motrice du feu et sur les machines propres à développer cette puissance* (Reflextions on the Motive Power of Fire). On the analogy of the hydropower engine he designed a hypothetical engine (Carnot heat engine) that transfers energy from a warm region to a cool region of space and, in the process, converting some of that energy to mechanical work. This engine operates on a thermodynamic cycle called Carnot cycle that consists of four steps: 1. Reversible isothermal expansion of the gas at a hot temperature T_H (isothermal heat addition). 2. Isentropic (reversible adiabatic) expansion of the gas (isentropic work output). 3. Reversible isothermal compression of the gas at a cold temperature T_C (isothermal heat rejection). 4. Isentropic compression of the gas (isentropic work input). After the fourth step the gas returns to the initial state.

Based on the Caloric Theory Carnot viewed heat as a substance and computed the efficiency of the Carnot heat engine with the following relationship:

$$\eta = \frac{Q(T_1 - T_2)}{Q} = \frac{T_1 - T_2}{T_1}$$

where η is the efficiency, Q is the heat put into the system, T_1 is the absolute temperature of the hot reservoir and T_2 is the absolute temperature of the cold reservoir.

Although he got seemly correct value for the efficiency the analogy that he used was perfectly incorrect. There is not as much heat in the warm region as in the cold region because the heat changes into mechanical work. On the other hand the recognition that the useful effect of the engine depends only on the temperature difference and it is independent of the working substance was perfect.

In his later memorandums before his early death there are plenty of indications to the Principle of Conservation of Energy and to the vitality of the Kinetic Theory.

7. The First Law of Thermodynamics – The Principle of Conservation of Energy [1][2][9][11][12]

The German surgeon *Julius Robert von Mayer* (1814-1878) started a study on the physical side of the symptoms of life during his journey in Dutch East India in 1840 and noticed that the venous blood of the sailors in the tropics is much darker than in cold climates. He concluded that the chemical processes of the body get their sources of energy for oxidation from the nature.

Arriving home he wrote a scientific paper in 1841 under the name of *Über die quantitative und qualitative Bestimmung der Kräfte* (On the Quantitative and Qualitative Determination of Forces). It was ignored by the physicists because of its strange argumentation that were based on the principle of causa aequat effectum so he could publish it next year in a chemical journal under the title of *Bemerkungen über die Kräfte der unbelebten Natur* (Remarks on the Forces of Inorganic Nature). This fundamental paper contained the first adequate formulation about the Law of Conservation of Energy that although work and heat are different forms of energy, they can be transformed into one another. He also specified theoretically the numerical value of the mechanical equivalent of heat as 365mkp (3580J) which is a little bit far from the real value but the order of size and the deduction was correct. Mayer also gave suggestions how to transform experimentally kinetic energy into heat.

Contemporaneously *James Prescott Joule* (1818-1889) made experiments and measurements to estimate the mechanical equivalent of heat and in 1843 he announced his results in a scientific meeting in Cork but there was only meagre attendance. In 1845 Joule wrote a paper *On the Existence of an Equivalment Relation Between Heat and the Ordinary Forms of Mechanical Power* and sent it to the British Association meeting in Cambridge. He reported about his best-known experiment using a falling weight to spin a paddle-wheel in an insulated barrel of water that increased the water temperature. Firstly he estimated the mechanical equivalent of heat as 424mkp (4158J) that was later refined by him as 427mkp (4187J).

The Law of Conservation of Energy was outlined in the works of Mayer and Joule but the modern form of it was formulated by the German physician *Ludwig Ferdinand von Helmholtz* (1821-1894). Studying the muscle metabolism he observed that no energy is lost in the muscle movement. In 1847 he based his book *Über die Erhaltung der Kraft* (On the Conservation of Energy) on a rule that all form of energy (mechanic, heat, light, magnetism) are equivalent. His theorem was hardly disputed and the Law of Conservation of Energy could be gone out of mind if did not raise up the interest of *William Thomson, Lord Kelvin* (1824-1907) who recognized the significance of Helmholtz's paper. He experimented in order to bloster Joule's results and in 1848 he

published his article *On the Absolute Thermometric Scale*. He suggested the introduction of an absolute temperature scale about which Amontons had speculated in 1695. Based on the Celsius scale Kelvin determined the absolute zero temperature in -273°C under which the kinetic energy of material particles is as low as possible.

8. The Second Law of Thermodynamics and the Kinetic Theory of Gases[1][2][4][5][9][11][12]

In the middle of the 19th century it was trivial that the Law of Conservation of Energy in not enough to explain the natural phenomenon because – as Carnot stated formerly – there is a determined tendency of the thermodynamic processes and the heat can spontaneously flow only from hot to cold materials. This is why the Second Law of Thermodynamics was needed and this necessity was recognized by *Rudolf Julius Emanuel Clausius* (1822-1888).

In 1850 he wrote his famous paper *Über die bewegende Kraft der Wärme* (On the Moving Force of Heat and the Laws of Heat) in which he stated the basic idea of the second law that heat generally cannot flow spontaneously from cold to hot bodies. If it could happen it would be possible to transform the 100% of heat into mechanical energy.

Another formulation of the second law was written down in 1851 by Lord Kelvin in his work entitled *On the Dynamical Theory of Heat* that it is impossible to convert heat completely into work in a cyclic process.

These negative sentences as the law of thermodynamics sounded very strange for the physicists so a new idea was needed to formulate a more adequate definition. It is going to be the idea of entropy a decade later.

At the same time the work of Bernoulli was rediscovered by *John Herapath* (1790-1868) in 1816 and submitted a paper to the Royal Society but it was rejected because its conclusions were seemed to be erroneous.

After the studying of Bernoulli's and Herapath's work *John James Waterston* (1811-1883) wrote a publication in 1843 under the title of *Thoughts on the Mental Functions*. He correctly derived the consequence that the gas pressure is generated by the high-speed motion of the material particles and countable with multiplying the number of molecules per unit volume, the molecular mass, and the molecular mean-squared velocity.

However it contained the elementary form of the Kinetic Theory of Gases this paper was rejected by the Royal Society because of its modern intonation and he could publish a short abstract of it. In 1848 Joule made calculations in order to compute the speed of the hydrogen molecule but his article in 1851 did not arouse the interest so together with Waterston's work it had only a little influence on the next generation.

The real breakthrough came after the article of *August Karl Krönig* (1822-1879) in 1856. It was based on Waterston's work and its simple gas-kinetic model gave plenty of motivations and ideas for the other researchers. In 1857 Clausius wrote a paper under the title of *Über die Art der Bewegung, welche wir Wärme nennen* (On the Kind of Motion which we call Heat) in which he stated that the internal energy of gases equals

with the kinetic energy of the atoms or molecules of gases He developed a much more complex but sophisticated theory than Kröning that included not only the translational but also the rotational and vibrational molecular motions.

This article motivated the Scottish physicist, *James Clerk Maxwell* (1831-1879) to give up the theorem that in a given amount of gas the molecules have the same speed and formulated the Maxwell Distribution of Molecular Velocities with which he founded a new branch of physics called statistical thermodynamics. He published his formula in 1860 in his work called *Illustrations of the Dynamical Theory of Gases* that described the particle speeds of gases at a determinate temperature and showed the statistical distribution of it. Maxwell worked out the equipartition theorem which means that in thermal equilibrium the total kinetic energy of a system is shared equally (in average) among all of its various forms, so the average kinetic energy in the translational motion of a molecule should equal the average kinetic energy in its rotational motion. After universalizing this law he also stated that the internal energy is equally shared between the degrees of freedom and it depends only on the temperature of the system.

9. Entropy, Statistical Thermodynamics and the Third Law of Thermodynamics [1][2][7][8][9]

Joule and Kelvin also speculated that there was an inevitable loss of useful heat in all thermodynamic processes and observed that natural processes are tended from an organized to a disorganized state. In addition in the 1850s it was necessary to find a correct mathematical description for the Second Law of Thermodynamics because the former definitions were not as accurate as needed.

This is why coined Clausius the concept of entropy in 1865 which means how organized or disorganized a system is. With the help of entropy we can explain the tendency of processes because the most likely event happens in the nature. It was also possible to formulate mathematically why flows spontaneously heat from hot into cold bodies. Because decreasing of temperature results the increasing of entropy.

The young Austrian physicist *Ludwig Eduard Boltzmann* (1844-1906) started to deal with the Kinetic Theory of gases in 1866. His work was promoted by Maxwell's book called *Theory of Heat* in 1871 and confirmed that the thermodynamic systems is tended towards the thermal equilibrium because this is the most likely state.

Developing Maxwell's equipartition theory and the distribution of molecular velocities he calculated the value of kinetic energy to each degree of freedom with the formula of:

$$\frac{1}{2}kT$$

where T is the absolute temperature and k is the Boltzmann's constant and equals to $1{,}38065\times10^{-23}$ J/K. With the help of entropy Boltzmann redefined the Second Law of Thermodynamics in 1877. He introduced the concept of thermodynamics probability as the number of microstates corresponding to the current macrostate and formulated the connection between entropy and molecular motion showing that the logarithm of thermodynamic probability (W) is directly proportional with the entropy (S).

$$S = k * \ln W$$

Before the Third Law of Thermodynamics the last important step was taken by the American physicist and chemist *Josiah Willard Gibbs* (1839-1903) by introducing the concept of the thermodynamic potentials and free energy in 1876 in his monograph called *On the Equilibrium of Heterogeneous Substances*. Thermodynamic potentials could be formulated with the help of the state parameters like volume (V), pressure (p), temperature (T) and internal energy (U) and they make easier to calculate some characteristics of the system (heat capacity, reaction heat). These potentials are free energy (F), enthalpy (H) and free enthalpy (Gibbs energy, G) and they measure the useful work of a closed thermodynamic system at constant temperature and volume (free energy), or at constant pressure (enthalpy) or at constant pressure and temperature (Gibbs energy).

Studying the high-temperature reaction of gases *Walther Hermann Nernst* (1864-1941) analyzed these kinds of thermodynamic potentials in 1889. He was deeply influenced by the thermodynamic researches of *Max Karl Ernst Ludwig Planck* (1858-1947) and the birth of quantum mechanics in 1900 and started to examine the change in specific heat of different materials. In 1906 he published his theorem with which he established the Third Law of Thermodynamics. This law describes the behaviour of a thermodynamic system as the temperature decreases to the absolute zero. Nernst stated that the entropy of a system at a temperature of absolute zero becomes zero in the case of perfect crystalline substances. He also laid down that it is impossible to reduce the temperature of any system to the absolute zero in the finite number of steps.

$$\lim_{T \to 0} \Delta S = 0$$

In this formula T is the absolute temperature and S is the entropy of the system.

10. Thermal radiation [1][3][4][5][6][9][11][12]

It was an important scientific issue from the beginnings of the thermodynamics to solve the problem of thermal radiation. Scientists in the Middle Ages observed that a heated piece of iron radiates heat and light at the same time but the forms of heat transfer were distinguished only in 1777 by Scheele as convection, conduction and radiation.

The Swiss physicist *Pierre Prévost* (1751-1839) showed it first in 1791 that all bodies radiate heat no matter how hot or cold they are and discovered in 1809 that the radiated heat depends only on the temperature of the radiating body and it is independent from the temperature of the surroundings.

In 1804 *John Leslie* (1766-1832) experimented with his famous apparatus called Leslie cube in order to monitor the intensity of radiant heat. He filled a cubical vessel with boiling water and composed one side with highly polished metal and two sides with dull metal. One side of the cube was painted black. During his experiments he detected the greatest radiation from the black side and irrelevant from the polished side.

Using an optical bench that was set up with theropiles, shields and heat and light sources the Italian physicist *Macedonio Melloni* (1798-1854) examined carefully the black body radiation and in 1831 he showed that radiant heat could be reflected, refracted and polarized as light.

The Prussian physicist, *Gustav Robert Kirchhoff* (1824-1887) was interested in black-body radiation too and in 1859 he noticed a simple but important connection between the emission and absorption of radiating bodies. Kirchhoff's Law of the Thermal Radiation states that in a unit of time the emission of a radiating body or a surface at given temperature and frequency equals its absorption, so the ratio of the emission and absorption is independent from the material parameters of the radiating body.

The further development was promoted by Maxwell's conclusion in 1862 that there is a clear connection between light, electromagnetic and thermal radiation. *John Tyndall* (1820-1893) also made experiments about thermal radiation in the 1860s and loaded with errors measured that the emission of black-body at 1473K is 11,7 times higher than at a temperature of 798K. His measurements were analyzed by the Slovenian physicist, *Jožef Štefan* (1835-1893) in 1879 and realized a connection between Tyndall's results. He constructed a law that the total energy (E) radiated per unit surface area of a black body in unit of time is directly proportional to the fourth power of the black body's absolute temperature (T):

$$E = \sigma T^4$$

Using the laws of thermodynamics Boltzmann also recognized the same connection in 1884 therefore this law was named Štefan-Boltzmann Law. The σ constant in the formula (Štefan-Boltzmann constant) was determined as $5{,}672 \times 10^{-8} W/m^2K^4$.

The solution of the radiant heat problem got near when in 1893 *Wilhelm Karl Werner Wien* (1864-1928) noticed an empirical formula between the temperature (T) of the body and the peak wavelength (λ_{max}) emitted by it:

$$\lambda_{\max} T = 2{,}8978 * 10^{-3} mK$$

He also ascertained that hotter bodies emit most of their radiation at shorter and colder bodies at longer wavelengths. Based on Maxwell's Law of Speed Distribution he created a formula to describe the intensity of black body radiation in 1896.

At the same time *Lord Rayleigh* (1842-1919) and *James Hopwood Jeans* (1877-1946) tried to introduce another kind of formula to describe spectral radiance of electromagnetic radiation that was later known as Rayleigh-Jeans Law.

Plenty of scientific researchers (Lummer, Pringsheim, Rubens, Kurlbaum) wanted to measure the intensity of thermal radiation in a huge scale of wavelengths and found out that Wien's Law is applicable only at short and Rayleigh-Jeans Law only at long wavelengths.

Finally the German physicist *Max Karl Ernst Ludwig Planck* (1858-1947) solved the problem and created a perfect formula that describes the black body radiation at all wavelengths as a function of temperature and wavelength

$$E = \frac{c^2}{h\lambda^5} \frac{1}{e^{\frac{ch}{k\lambda T}} - 1}$$

where c is the speed of light, h is Planck's constant, λ is the wavelength, k is Boltzmann's constant and T is the temperature of the black body.

To construct this relationship Planck had to postulate that energy could be emitted only in quantized form. This was presented by him on 14th December 1900 in Berlin and this date is declared as the birth of Quantum Physics.

Planck also gave a very simple formula to describe the energy quantum (energy of the photon) with the product of the frequency of its associated electromagnetic wave (ν) and the Planck constant (h=6,626×10^{-34} Js):

$$E = h\nu$$

On the basis of Planck's quantum theory *Albert Einstein* (1879-1955) could come forward in 1905 with the idea of the quantization of light. In his article entitled *Über einen die Erzeugung und Verwandlung des Lichtes betreffenden heuristischen Gesichtspunkt* (On a Heuristic Point of View Concerning the Production and Transformation of Light) Einstein stated that light consists of localized particles (quanta). This theory was first rejected and it became fully accepted only in 1919. In 1906 Einstein also solved with the help of the quantum theory the dilemma why exists a huge difference between the theoretically and measured specific heat of solids.

References

[1] Bakányi Márton, Fodor Erika, Marx György, Sarkadi Ildikó, Tóth Eszter, Ujj János: "*Fizika – Gimnázium I. osztály*" Nemzeti Tankönyvkiadó, Budapest, 1997

[2] Richard Culham: *"History of Thermodynamics"* electronic handbook, University of Waterloo, Department of Mechanical and Mechatronics Engineering, Microelectronics Heat Transfer Laboratory, 2007 http://www.mhtlab.uwaterloo.ca/courses/me354/history.html

[3] Dr. Horváth András, Berta Miklós: *"Fejezetek a fizikából"* Novadat Bt., Győr (H), 1996

[4] HyperJeff Network: *"History of Statistical Mechanics and Thermodynamycs"* HyperJeff Network/Histories/Statistical Mechanics and Thermodynamics http://history.hyperjeff.net/statmech

[5] Dr. Inzelt György: *"Az energia, az energiaváltozás és az energiaátalakítás fogalmának fejlődéstörténete (a hőtantól a termodinamikáig)"* előadás, Eötvös Loránd Tudományegyetem TTK Fizikai Kémiai Tanszék, Budapest, 2009 http://www.chem.elte.hu/w/modszertani/index_elemei/kem_tortenete_elemei/INZELT%20GY%20TERMODINAMIKA.ppt

[6] Dr. Író Béla: *"Hő és áramlástan"* Universitas Kiadó, Győr (H), 2007

[7] Dr. Kellermayer Miklós: *"Termodinamika"* előadás, Semmelweis Egyetem Biofizikai és Sugárbiológiai Intézet, Orvosi Biofizika, 2010-03-24 http://biofiz.sote.hu/run/dl_t.php?id=238&tid=2

[8] Dr. Keszei Ernő: *"Bevezetés a kémiai termodinamikába"* Eötvös Loránd Tudományegyetem, Budapest, 2006

[9] Simonyi Károly: *"A fizika kultúrtörténete a kezdetektől 1990-ig"* Akadémiai Kiadó, Budapest (H), 1998

[10] Dr. Szalay Béla: *"Fizika"* Műszaki Könyvkiadó, Budapest, 1966

[11] Articles from Wikipedia, the Free Encyclopedia http://www.wikipedia.org

[12] Stephen Wolfram: *"A New Kind of Science"* Wolfram Media, 2002 http://www.wolframscience.com

Road safety Performance Indicators in Hungary

P. Holló

KTI Institute for Transport Sciences Non-profit Ltd.
H-1518 Budapest, PO Box 107, Hungary
Phone : +36(1)3715823, Fax : +36(1)2055932
e-mail: hollo.peter@kti.hu
"Széchenyi István" University H-9026 Győr, Egyetem tér 1.

Abstract: The paper shows some convincing Hungarian examples for the necessity of road safety performance indicators. In lack of these data sometimes it is almost impossible to explain the changes in the road safety situations. What is more, it is also impossible to discover the deteriorating factors behind general improvement, which is important in order to make the road safety policy more target-oriented and effective. In Hungary, the monitoring system has been working for more than a decade and the time series show very interesting and important changes. (In the field of safety belt wearing rate and usage of DRL we have such data collection methodology which has been considered as best practice in the framework of the SafetyNet project.) Based on the monitoring of such data, some important countermeasures could be introduced in Hungary. International comparison of safety performance indicators could have significant impact on the national road safety policy. The paper shows some examples for such impacts as well.

Keywords: *road safety, performance indicators, safety belt, daytime running lights, child safety*

1. The system of data collection

Collection and evaluation of some kinds of road safety performance indicators began in 1992 in Hungary. All work items were carried out by the TÜV NORD-KTI Kft. [1]. Since the usage of Daytime Running Lights (DRL) became obligatory from 1993 on, it seemed to be obvious to collect the safety belt wearing and DRL usage rates together. The yearly sample size is approximately 10 000 vehicles, including cars, microbuses and small vans (categories M1 and N1). The cars equipped with foreign and taxi licence plates haven't been taken into account. Sample sizes by different road types (country roads, motorways, roads inside built-up areas) were above 3000.

Since collection of the DRL usage and the safety belt wearing rates have been combined, the observations were carried out always in good weather and visibility conditions in order to avoid the influence of these factors on DRL usage rates. Data collection was carried out always in the same period of the year (May, June) for the sake of comparability.

Performance indicators mentioned are so-called behavioural ones [2], since they reflect the rate of following the rules by drivers.

As the next examples show, there are already relatively long time series of SPIs in Hungary. They reflect different trends, which are useful in the evaluation and elaboration of road safety policies.

The methodology of Hungarian data collection in the field of safety belt wearing rate and the usage rate of DRL has been considered as best practice in the framework of the SafetyNet project.

After the introduction of uniform EU legislation regarding DRL (automatic DRL for new cars), this kind of performance indicators will lose its importance step by step.

2. Safety belts

2.1. Changes in safety belt wearing rates

In Figure 1 the development of the safety belt wearing rate can be seen by seat positions and in general. The first survey was carried out in 1992 and for the year 2006 – in lack of contract – we do not have data.

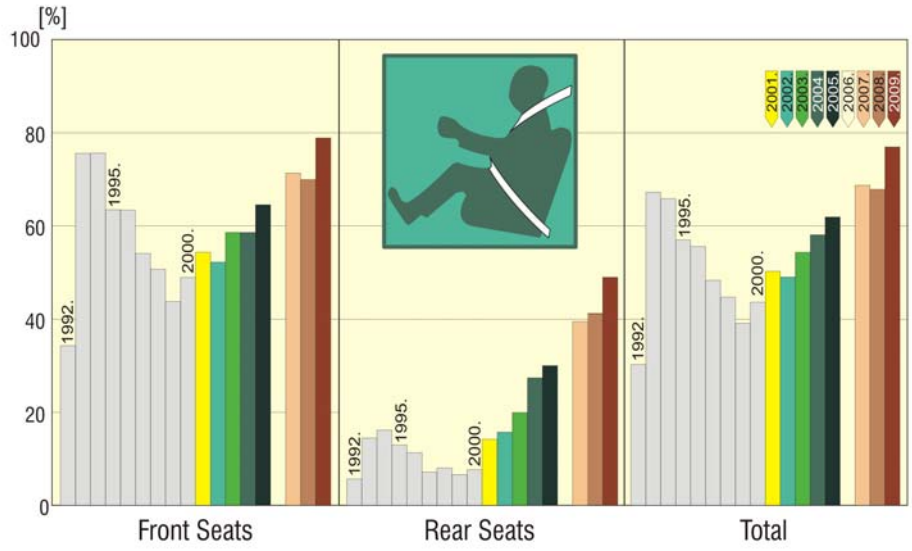


Figure 1: Safety belt wearing rates in Hungary

After a sudden increase in the wearing rate in the front seats, following the modification of the Highway Code in 1993, a declining trend was characteristic from this year until 1999. It is very contradictory and shows clearly the necessity of the road safety performance indicators, that this declining trend in safety belt wearing rate could be observed in a period, which was a so-called "success story" in the history of the Hungarian road safety. (Between 1990 and 2000, the number of the people killed in road traffic accidents decreased by more than 50%).

After the nadir in 1999 there was an increasing trend in safety belt wearing rate not only in the front, but in the back seats as well. This trend is the same even today; the values in 2009 are higher than the earlier ones. Although safety belt wearing rate in the back seats of passenger cars is below 50% yet, the relative change (from 6.6 % in 1999 to 49.3 % in 2009) was higher (42.7 %) than in the front seats (from 43.8 % in 1999 to 79.2 % in 2009 = 35.4 %).

The Figure 2 shows the changes in safety belt wearing rate outside built-up areas by road categories and seat positions.

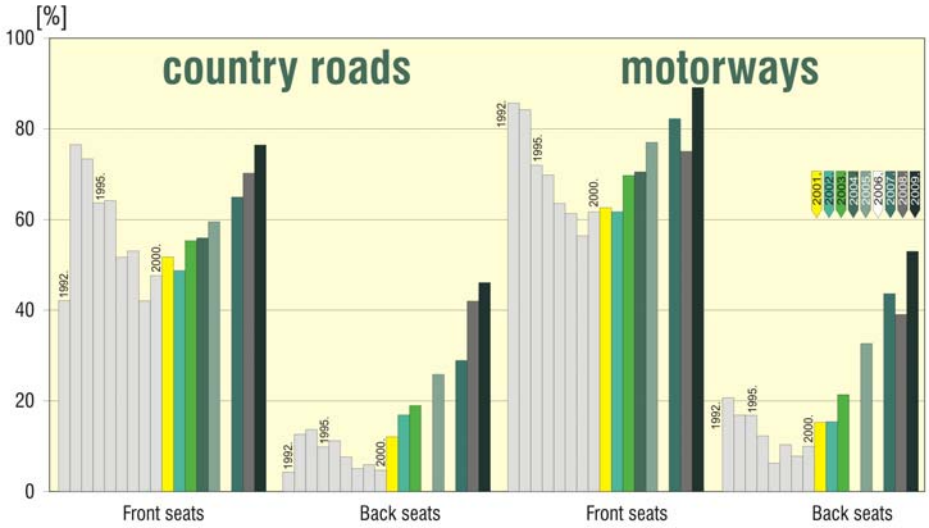


Figure 2: Safety belt wearing rates outside built-up areas on rural roads and on motorways

There are similar changes on rural roads (in the IRTAD database: country roads) and motorways as well: after a declining trend, until this year there was an increasing rate in the safety belt wearing. It can be observed that the rates are higher on motorways than on rural roads. It seems that car occupants consider the accident risk on motorways (travelling at higher speed) higher than on rural roads. In 2009, the safety belt wearing rate was 75.5 % in the front seats of passenger cars on rural roads and almost 89.0 % on motorways.

The safety belt wearing rates inside built-up areas have been always the lowest in Hungary in comparison with roads outside built-up areas (country roads and motorways).

This is the case in 2009, too, in spite of the fact that increasing trend is characteristic on roads inside built-up areas as well. The amount of increase is outstandingly high in back seats of passenger cars. This can be seen in the Figure 3 very well.

Although the increasing trend in safety belt wearing seems to be general in Hungary, the international comparison shows that there is a further potential of improvement in this field.

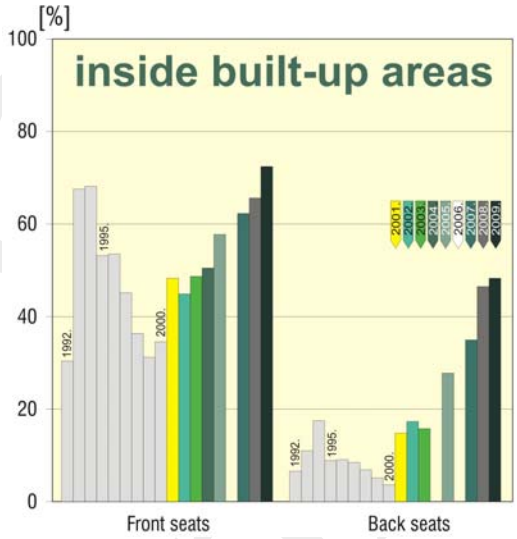


Figure 3: Safety belt wearing rates inside built-up areas (in Budapest)

2.2. International comparison of safety belt wearing rates

In the IRTAD database there are data for safety belt wearing rates observed in the front seats of passenger cars. These data are appropriate for international comparison. Unfortunately only some countries have data for 2008, most of them have only those for 2007.

In Figure 4 the safety belt wearing rates observed inside built-up areas in 2007 and 2008 can be seen.

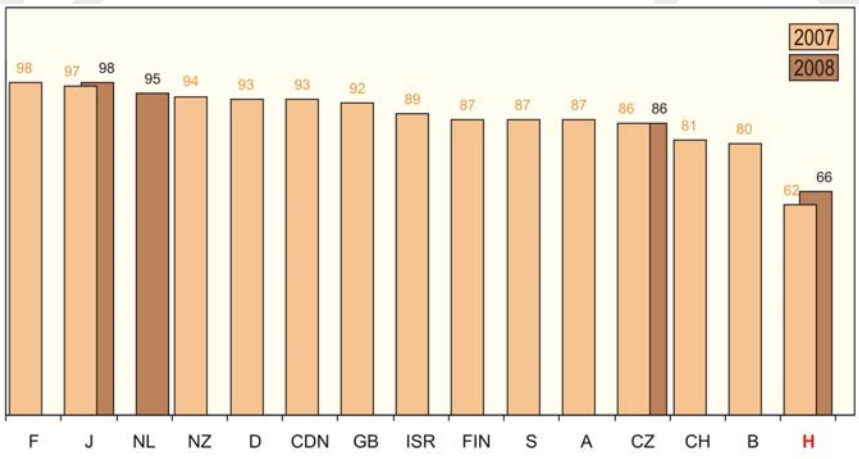


Figure 4: International comparison of safety belt wearing rates in front seats of passenger cars inside built-up areas (Source: IRTAD)

In spite of the improvement of recent years, Hungary is the last one out of the investigated countries. Even the rate of 2009 (72 %) would be the lowest among the countries shown in Figure 4. It is surprising that in France, Japan and the Netherlands 95-98 % of the drivers are wearing the safety belt inside built-up areas.

Figure 5 shows the same comparison for country roads. Here, the Hungarian data for 2009 (75.5 %) were found to be equal to the Belgian ones for 2007.

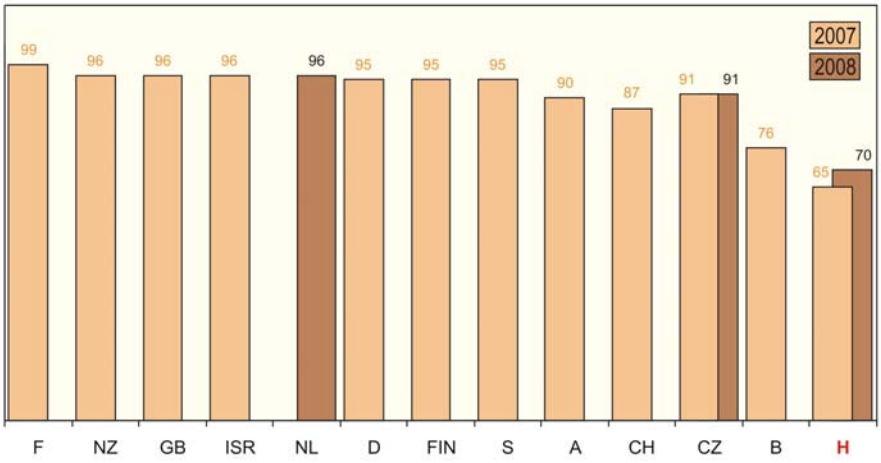


Figure 5:International comparison of safety belt wearing rates in front seats of passenger cars on rural roads (Source: IRTAD)

Despite the clear improvement in Hungary, we are still the last considering the usage of safety belts on country roads, out of the countries displayed in Figure 5. The French rate is almost 100 % and the rates in New-Zealand, Great Britain, Israel, and the Netherlands are 96 %.

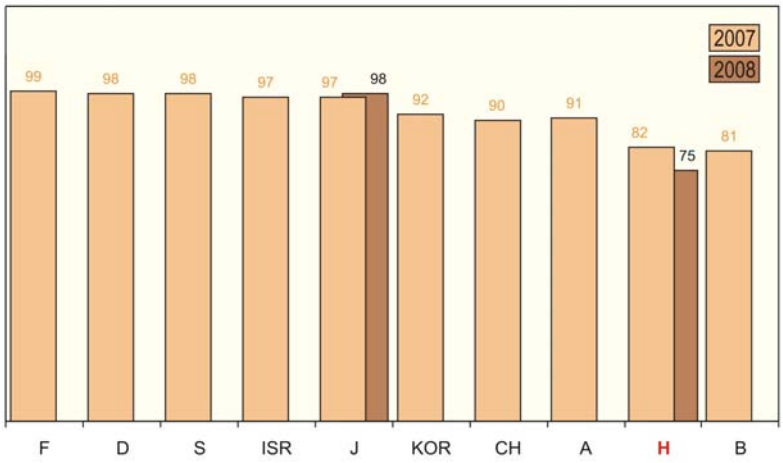


Figure 6: International comparison of safety belt wearing rates in front seats of passenger cars on motorways (Source: IRTAD)

The safety belt wearing rates observed in front seats of passenger cars on motorways can be seen in Figure 6. Here, the number of countries compared is lower than in the earlier two Figures. If we take into account the Hungarian rate for 2009 (89 %, cannot be seen in the Figure), we can say that perhaps Hungary could "overtake" Belgium, what is more, this data would be very close to the Swiss and Austrian figures registered in 2007.

2.3. Child Safety

The safety of children is of high priority in Hungary.

In Figure 7 the usage rate of child safety devices can be observed.

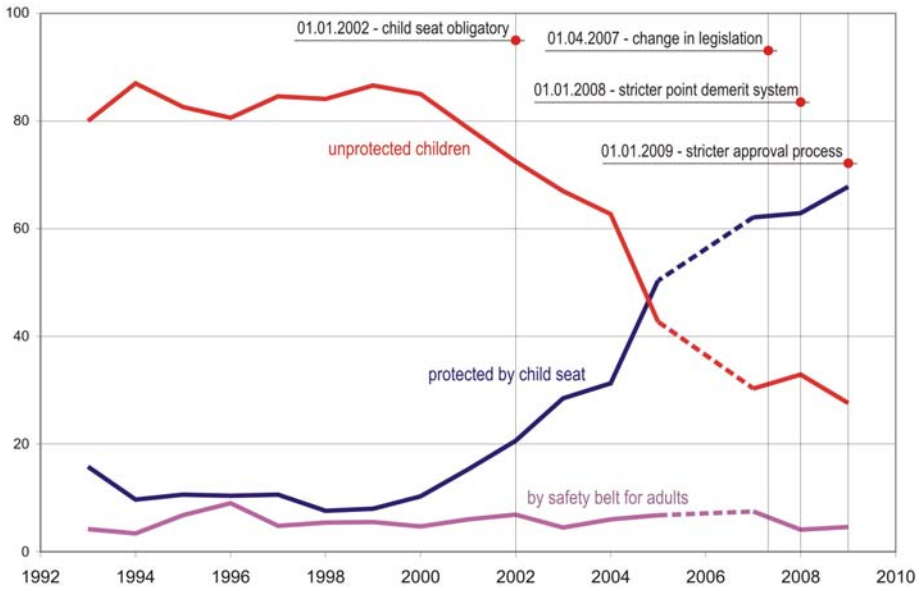


Figure 7: Usage rate of child safety devices in Hungary

Until the year 2000 the usage rate of child seats and the rate of protected children were very low, about, or below 10%. It means that the rate of unprotected children was very high, between 80 and 90 %. The first step towards the higher safety of children was the introduction of the mandatory use of child seats as of 1 January 2002. From that time children younger than 12 years and below 150 cm height are obliged to travel in child safety seats. From 1 April 2007 the legislation changed. It means that instead of age, the body height is decisive. Under 150 cm height, children have to travel in child restraint systems which are in accordance with their weight.

From 1 January 2008 the point demerit system became stricter. The number of demerit points for non-using the proper safety device has been doubled: from this date on the number of demerit points for this offence are two.

As of 1 January 2009 only the child safety seats having an approval number starting with 03 or 04 can be sold and used. This means a stricter approval procedure.

Due to all of these measures and road safety campaigns, the usage rate of child restraint systems was almost 70% in 2009. The rate of children restrained by adult safety belts remained below 10%. Promising result is that the rate of unprotected children decreased from 65% (1994) to 28% (2009). Of course, in the future there is a lot to do for the higher safety of children, but there is a clear positive progress.

The number of killed and injured child car-occupants in the last three years shows clearly the positive development (Table 1).

Table 1: Number of killed and injured child (0-14 years) car-occupants

Year	Killed	Seriously injured	Slightly injured	All casualties
2000	13	128	641	782
2001	15	114	837	966
2002	17	124	843	984
2003	15	156	907	1078
2004	22	142	988	1152
2005	19	135	1044	1198
2006	23	134	1033	1190
2007	18	118	1063	1199
2008	13	104	915	1032

2.4. The remaining safety potential of safety belt wearing in Hungary

The earlier estimations regarding the number of avoidable deaths and injuries by further increase in the safety belt wearing rate have been based mostly on Bohlin's [3] research results. In the meantime a lot of research projects have been carried out and now their meta-analysis is available [4]. These results can be considered as the most reliable ones, therefore the data from Elvik and Vaa are considered to be the basis of our estimation, too. It is important that the authors [4] make a distinction from the point of view of efficiency between the front and the back seats safety belts, what is more, between drivers and front seat passengers, too. According to the results of the meta-analysis the fatality risk of the car driver can decrease by 50% as a result of safety belt wearing. The respective values for the decrease of the risks of serious and slight injuries are 45% and 25%. The same values for the front seat passengers are the following:

Fatality risk: –45%
Risk of serious injuries: –45%
Risk of slight injuries: –20%

The values for the back seats are:

Fatality risk: –25%

Risk of serious injuries: –25%
Risk of slight injuries: –20%

It seems to be surprising at the first sight that the results of the meta-analysis estimate much less decrease in the number of slight injuries than Bohlin's results (80%) do. But is is very probable, if we consider that the safety belt can save life by decreasing the severity of injuries. This means that the persons who sustained fatal injuries without safety belt will only be serious injured if wearing a belt, and the persons who sustained serious injuries without safety belt will only be slight injured with safety belt. Taking this into account the results from Elvik and Vaa are more understandable.

First we carry out the estimation for 100% safety belt wearing rate.

In Hungary 146 car drivers sustained fatal, 511 serious, and 1096 slight injuries without safety belt in 2008. It means that in their group

146×0.5 = 73 fatal
511×0.45=230 serious
1096×0.25=274 slight

Injuries could have been avoided in case of 100% safety belt wearing rate. (In 2008 the average safety belt wearing rate was 77% in Hungary).

In the same year 59 front seat car passengers sustained fatal, 203 serious, 526 slight injuries without safety belt. Having applied the data of the meta-analysis:

59×0.45 = 27 fatal
203×0.45 = 91 serious
526×0.20 =105 slight

Injuries could have been prevented in case of 100% safety belt wearing rate.

Finally, in the back seats of cars 55 passenger died, 266 sustained serious and 622 slight injuries without safety belt in 2008. Having used these data and the results of the meta-analysis:

55×0.24 = 14 fatal
266×0.25 = 67 serious
622×0.20 = 124 slight

Injuries could have been prevented in case of 100% safety belt wearing rate in the group of car occupants.

Taking into account that 100% safety belt wearing rate – especially in Hungary – is unrealistic, the real target could only be 95%. The example of highly motorized countries has shown that this rate could be kept permanently above 90% with appropriate awareness campaigns and police enforcement.

In case of 95% safety belt wearing rate:

114×0.95 = 108 fatal
388×0.95 = 369 serious
503×0.95 = 478 slight

Injuries could have been prevented.

Nowadays – in the era of high-tech safety belt systems (belt retention system, pyrotechnic device) – the safety potential of the belts is higher, not to mention the fact that they "work together" in most cases with airbag systems.

3. Daytime Running Lights (DRL)

In Hungary, the obligatory use of DRL was introduced in two steps. First, as of 1 March 1993, this involved only the main roads outside built-up areas and the so-called motor roads (semi-motorways). Later, from 1 June 2004 the use of DRL became obligatory on all roads outside built-up areas. It means that on roads inside built-up areas the usage of DRL is not obligatory. The legislation is valid outside built-up areas throughout the whole year.

In Figure 8 the changes in DRL usage rate can be seen by road categories. (The data for 2006 are lacking here also). The trend of the DRL usage rates is entirely different from those of safety belt wearing rates. Here – outside built-up areas – an almost continuous increasing trend can be observed. After the obligatory introduction of DRL, this rate was below 60 % and in 2009 it was almost 95 % on rural roads and on motorways. The rates in Budapest are very low, which is obvious, since – as mentioned earlier – inside built-up areas the usage of DRL is not mandatory. In spite of this, DRL usage rate increased in the last two years in Budapest as well.

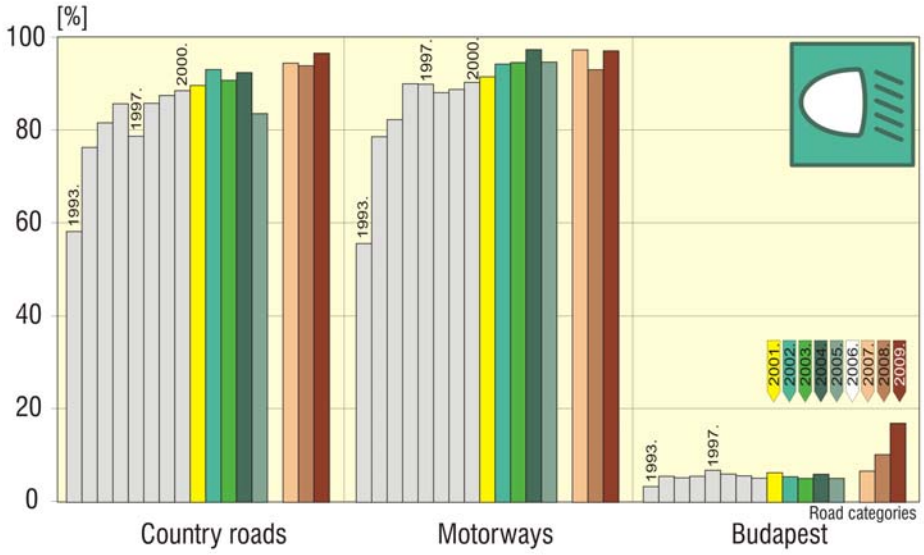


Figure 8: DRL usage rates in Hungary according to road categories

4. Conclusions

Hungary has reliable performance indicators on the rate of safety belt and DRL users. Time series of these indicators are available from 1992 or 1993, respectively. The trend in safety belt wearing shows almost the same changes on all road types and seat positions: declining rate from 1992 to 1999 and increasing rate from 2000 on until now.

This positive development confirms that the road safety policy is on the right track regarding safety belt wearing. The increasing rates are the results of the further development of the point demerit system, the co-ordinated awareness campaigns, the more intensive police enforcement and the more serious consequences of non-wearing. In spite of the positive development of recent years, there is a relatively great potential in the further increasing of the safety belt wearing rates. According to estimations 108 fatalities, 369 serious injuries and 478 slight injuries could have been prevented in case of 95% safety belt wearing rates in Hungary.

The usage of child restraint systems shows also a great development, the rate of unprotected children decreased from 65% (1994) to 28 % (2009), though on the other hand it means, that almost one third of children travel still unprotected.

The rate of DRL users shows a continuously increasing trend.

The introduction and widespread usage of other performance indicators detecting the behavioural characteristics in the field of legislation regarding speed, drinking and driving, etc. would be very important in the future [5].

References:

[1] Analysis of usage of passive safety devices, suggestions for the further development of the legislation (In Hungarian: A passzív biztonsági védőeszközök használatának elemzése, javaslat kidolgozása, szabályozás korszerűsítése.
Research Report No. I-8885/09, Budapest, TÜV NORD- KTI Kft., Supervisor: Dr. Tamás Véssey

[2] Elvik, R.; Vaa, T.: The handbook of road safety measures, Elsevier, 2004.

[3] Fred Wegman (working committee chair): Transport Safety Performance Indicators. Brussels European Transport Safety Council. 2001. 56 p.

[4] Bohlin, N.: A Statistical Analysis of 28 000 Accident Cases with Emphasis on Occupant Restraint Value, Göteborg, AB Volvo, 1967.

[5] Holló, P.: Data and monitoring on road safety performance indicators in Hungary. IRTAD International Traffic Safety Data and Analysis Group , 4th IRTAD Conference, Road Safety data: collection and monitoring performances and progress. Seoul, Korea, 16-17 September 2009, Proceedings, pp. 301-306.

Smooth Maximum Based Algorithms for Classification, Regression, and Collaborative Filtering

G. Takács

Széchenyi István University, Győr, Hungary

Abstract: Unbalanced classification problems are quite common in the practice of machine learning. Unbalancedness means that the distribution of the class labels is far from uniform. Convex polyhedron classifiers are special binary classifiers that fit well to unbalanced problems. In this paper I propose novel and computationally efficient algorithms for training convex polyhedron classifiers. The proposed algorithms are based on the smooth approximation of the maximum function. I also give the analogous variant of the approach for regression and collaborative filtering. Finally, I demonstrate the usefulness of the approach via experiments on artificial and real datasets.

Keywords: classification, regression, collaborative filtering, convex polyhedron, smooth maximum function

1. Introduction

In this paper *machine learning* is considered as discovering the relationship between the features of a phenomenon, based on a dataset that was collected by observing the phenomenon. Classification, regression, and collaborative filtering are three important special cases of machine learning.

In the problem of *classification* the phenomenon is modeled by a random pair $(\mathbf{X},Y)$, where

- $\mathbf{X}$ taking values from $\mathbb{R}^d$ is called *input*, and
- Y taking values from $\mathcal{C} = \{c_1,\ldots,c_M\}, M \geq 2$ is called *label*. If $M = 2$, then the problem is termed *binary classification*, otherwise it is termed *multiclass classification*.

The goal is to predict Y from $\mathbf{X}$ with a function[1] $g : \mathbb{R}^d \mapsto \mathcal{C}$ called *classifier* such that the probability of error

$$L(g) = \mathbf{P}\{g(\mathbf{X}) \neq Y\}$$

[1] Functions are always assumed to be measurable in this paper. Otherwise the function of a random variable would not necessarily be a random variable.

is minimal.

Typically, the distribution of $(\mathbf{X},Y)$ is unknown, therefore the minimal error probability and the optimal classifier are unknown too. We only have a finite sequence of corresponding input–label pairs from the past

$$\mathbf{T} = ((\mathbf{X}_1,Y_1),\ldots,(\mathbf{X}_n,Y_n)),$$

called training set. It is assumed that these pairs were drawn independently from the unknown distribution of $(\mathbf{X},Y)$, and also that $(\mathbf{X},Y)$ and $\mathbf{T}$ are independent. In practice we usually observe only one realization of $\mathbf{T}$ denoted by $\mathbf{t} = ((\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_n,y_n))$. This is our data at hand that we have to live with. The task is to *estimate* the optimal classifier on the basis of $\mathbf{T}$.

In the problem of *regression* the phenomenon is a random pair $(\mathbf{X},Y)$, where

- $\mathbf{X}$ taking values from $\mathbb{R}^d$ is called *input*, and
- Y taking values from $\mathbb{R}$ is called *target*.

The goal is to predict Y from $\mathbf{X}$ with a function $g : \mathbb{R}^d \mapsto \mathbb{R}$ called *predictor* such that the mean squared error

$$L(g) = \mathbf{E}\{(g(\mathbf{X}) - Y)^2\}$$

is minimal. It is true again that typically we have no information about the distribution of $(\mathbf{X},Y)$. The task is to estimate the optimal predictor based on an independent and identically distributed sample.

In *collaborative filtering*, the phenomenon is a random triplet (U,I,R), where

- U taking values from $\{1,\ldots,N_U\}$ is called the *user identifier*,
- I taking values from $\{1,\ldots,N_I\}$ is called the *item identifier*, and
- R taking values from $\{v_1,\ldots,v_M\} \subset \mathbb{R}$ is called the *rating value*.

A realization of (U,I,R) denoted by (u,i,r) means that the u-th user rated the i-th item with value r. The goal is to predict R from $(U,\ I)$ with a function $g : \{1,\ldots,N_U\} \times \{1,\ldots,N_I\} \mapsto \{v_1,\ldots,v_M\}$ such that mean squared error

$$L(g) = \mathbf{E}\{(g(U,I) - R)^2\}$$

is minimal.

Collaborative filtering can be viewed as a special case of regression. However, classical regression techniques are not suitable for solving collaborative filtering problems, because of the unique characteristics of the input variables.

Denote the random training set by $\mathbf{T} = ((U_1,I_1,R_1),\ldots,(U_n,I_n,R_n))$, and its realization by $\mathbf{t} = ((u_1,i_1,r_1),\ldots,(u_n,i_n,r_n))$. Denote the set of user–item pairs appearing in the training set by $\mathcal{T} = \{u,i : \exists k : u_k = u,i_k = i\}$.

In real life, if a user has rated an item, then it is unlikely that he/she will rate the same item again. Therefore it is unrealistic to assume that the elements of the training set are independent. A more reasonable assumption is

$$\mathbf{P}\{U_k = u_k,I_k = i_k,R_k = r_k\} = \\ \mathbf{P}\{U = u_k,I = i_k,R = r_k | \cap_{l=1}^{k-1} (U \neq u_l,I \neq i_l)\},$$

which means that the training set is generated by a "sampling without replacement" procedure.

If this assumption holds, then the training data can be represented as a partially specified matrix $\mathbf{R} \in \mathbb{R}^{N_U \times N_I}$ called rating matrix, where the matrix elements are known in positions $(u,i) \in \mathcal{T}$, and unknown in positions $(u,i) \notin \mathcal{T}$. The value of the matrix $\mathbf{R}$ at position $(u,i) \in \mathcal{T}$, denoted by r_{ui}, stores the rating of user u for item i.

1.1. Convex polyhedron classification

Many interesting classification problems arising in practice are *unbalanced*, which means that the distribution of labels is far from uniform. For example, in the case of breast cancer screening most patients are (fortunately) healthy. This results that in the corresponding binary classification problem most training examples belong to the "healthy" class. Convex polyhedron classifiers are special nonlinear classifiers that fit well to unbalanced problems.

Let us consider an unbalanced binary classification problem with labels c_1 and c_2. Let us call c_1 the positive and c_2 the negative class, and assume that the class with higher probability is the negative class. A convex K-polyhedron (polyhedron) is the intersection of K half-spaces (any number of half-spaces).

A *convex polyhedron (K-polyhedron) classifier* is a function $g : \mathbb{R}^d \mapsto \{c_1,c_2\}$ such that $\{\mathbf{x} \in \mathbb{R}^d : g(\mathbf{x}) = c_1\}$ is a convex polyhedron (K-polyhedron). An equivalent definition is the following: A function $g : \mathbb{R}^d \mapsto \{c_1,c_2\}$ is called a convex K-polyhedron classifier, if it can be written as

$$\begin{aligned} g(\mathbf{x}) =& \mathrm{th}(\min\{\mathbf{w}_1^T\mathbf{x} + b_1,\ldots,\mathbf{w}_K^T\mathbf{x} + b_K\}) \\ =& \mathrm{th}(-\max\{-\mathbf{w}_1^T\mathbf{x} - b_1,\ldots,-\mathbf{w}_K^T\mathbf{x} - b_K\}), \end{aligned} \tag{1}$$

where $\mathbf{w}_1,\ldots,\mathbf{w}_K$ are called weight vectors, $b_1,\ldots,b_K$ are termed biases, and

$$\mathrm{th}(z) = \begin{cases} c_1 & \text{if } z \geq 0 \\ c_2 & \text{if } z < 0 \end{cases}$$

is the threshold function.

When classifying an input $\mathbf{x}$, we iterate over the weight vectors. If $\mathbf{w}_k^T\mathbf{x} + b_k < 0$ for any $k \in \{1,\ldots,K\}$, then the input can be classified as negative immediately. As a consequence, convex polyhedron classifiers tend to classify negative examples quickly. This property makes the approach particularly suitable for unbalanced problems.

Despite this appealing property, currently convex polyhedron classifiers are not frequently used in practice. The main reason for that is the lack of efficient and practical training algorithms. In the next section we will overview the small literature of the area. Then, I will propose novel algorithms that attempt to make the convex polyhedron classifier a practical tool.

2. Known methods

Probably the best known work that applied convex polyhedron classifiers for solving a practical problem is [10]. In this paper the authors propose the *maximal rejection* (MR) approach that can be applied for training convex polyhedron classifiers. The key idea of MR is defining the criterion function

$$\mathcal{M}(\mathbf{w}) = \frac{(\mathbf{w}^T\mathbf{m}_1 - \mathbf{w}^T\mathbf{m}_0)^2 + \mathbf{w}^T\mathbf{R}_1\mathbf{w} + \mathbf{w}^T\mathbf{R}_0\mathbf{w}}{\mathbf{w}^T\mathbf{R}_1\mathbf{w} + \lambda\mathbf{w}^T\mathbf{w}}, \tag{2}$$

where $\mathbf{m}_1$, $\mathbf{m}_0$, $\mathbf{R}_1$ and $\mathbf{R}_0$ are the empirical means and covariances of the classes in the training set, and λ is the regularization coefficient. The detailed derivation of this criterion function (without the regularization term $\lambda\mathbf{w}^T\mathbf{w}$) can be found in [10]. The main idea is to modify the criterion function of Fisher discriminant analysis [12] such that we allow more variance within class 0, if the variance within class 1 is smaller. If we introduce the notation $\mathbf{Q} = (\mathbf{m}_1 - \mathbf{m_0})(\mathbf{m}_1 - \mathbf{m_0})^T + \mathbf{R}_1 + \mathbf{R}_0$, then $\mathcal{M}$ can be written as

$$\mathcal{M}(\mathbf{w}) = \frac{\mathbf{w}^T\mathbf{Q}\mathbf{w}}{\mathbf{w}^T(\mathbf{R}_1 + \lambda\mathbf{I})\mathbf{w}},$$

where $\mathbf{I}$ is the $d \times d$ identity matrix.

It can be shown that the $\mathbf{w}$ that maximizes $\mathcal{M}$ is an eigenvector of $(\mathbf{R}_1 + \lambda\mathbf{I})^{-1}\mathbf{Q}$ corresponding to the largest eigenvalue. Note that the maximum is not unique, since $\mathcal{M}(\mathbf{w}) = \mathcal{M}(\alpha\mathbf{w})$ for every $\alpha \neq 0$.

The outline of MR training is the following[2]:

- For $k = 1,\ldots,K$:

[2]This variant is a bit more flexible than the original one.

– Set $\mathbf{w}_k$ to $\arg\max_{\mathbf{w}\in\mathbb{R}} \mathcal{M}(\mathbf{w})$.
– If $\sum_{i:y_i=1} \mathbf{w}_k^T \mathbf{x}_i / \sum_i y_i < \sum_{i:y_i=0} \mathbf{w}_k^T \mathbf{x}_i / \sum_i (1-y_i)$, then flip the sign of $\mathbf{w}_k$.
– Define $g_k(\mathbf{x})$ as $\mathrm{th}(\min\{\mathbf{w}_1^T\mathbf{x}+b_1,\ldots,\mathbf{w}_k^T\mathbf{x}+b_k\})$.
– Set b_k by minimizing $\sum_{i:y_i=1} I\{g_k(\mathbf{x}_i)\neq y_i\} + \beta\sum_{i:y_i=0} I\{g_k(\mathbf{x}_i)\neq y_i\}$.
– Exclude examples from the training set for which $g_k(\mathbf{x}_i)=0$.

The output of training is a convex K-polyhedron classifier. The parameter $\beta > 0$ expresses our willingness to tolerate false negative classifications (larger β results more false negatives and less false positives).

There also exist other known methods for training convex polyhedron classifiers, but they are less practical than MR. Some of the alternatives are the following:

- [17] tries to separate each negative example from the positive class individually with an adaptation of the multiclass support vector machine method [8]. The algorithm can be used only for small problems due to its large computational complexity.
- The training algorithm if the ID3 decision tree [18] can also be used for training convex polyhedron classifiers, if we introduce the following restrictions: all features have to be continuous or binary, and one of the partitions has to be labeled as negative after each split. Unfortunately, the modeling power of this approach is quite limited.
- In the literature of *probably approximately correct learning* (PAC learning) [22] one can find theoretical works related to convex polyhedron classification, for example [11, 14, 15, 23]. PAC is a formalism for determining how much data is needed for a given classification algorithm to achieve a given accuracy on a given fraction of test examples. Unfortunately, the convex polyhedron classification algorithms published in the PAC papers are not practical methods. They are instead tools for proving theorems about PAC-learnability.

3. Smooth maximum functions

One of the factors that make the training of convex K-polyhedron classifiers hard is the non-differentiable maximum function appearing in the definition formula. One possible way of handling the difficulty is approximating maximum taking with a smooth[3] function.

Let us start the discussion with a simple observation. Assume that we have K different real numbers $u_1,\ldots,u_K$, and a function $f:\mathbb{R}\mapsto\mathbb{R}$ with the following property:

$$\forall u\in\mathbb{R}: \lim_{\Delta\to\infty}\frac{f(u+\Delta)}{f(u)} = \infty.$$

[3]Infinitely many times differentiable.

Denote the largest number by $u_{\max} = \max\{u_1,\ldots,u_K\}$, and the smallest number by $u_{\min} = \min\{u_1,\ldots,u_K\}$. Let us apply f on the numbers and investigate the values $f(u_1),\ldots,f(u_K)$. If the difference between $u_{\max}$ and the other numbers is large enough, then the following approximation is admissible:

$$\frac{f(u_j)}{f(u_{\max})} = \frac{f(u_j)}{f(u_j + (u_{\max} - u_j))} \approx \begin{cases} 0 & \text{if } u_j \neq u_{\max}, \\ 1 & \text{if } u_j = u_{\max}. \end{cases} \tag{3}$$

It follows from (3) that

$$\frac{f(u_j)}{\sum_{k=1}^{K} f(u_k)} = \frac{f(u_j)/f(u_{\max})}{\sum_{k=1}^{K} f(u_k)/f(u_{\max})} \approx \begin{cases} 0 & \text{if } u_j \neq u_{\max}, \\ 1 & \text{if } u_j = u_{\max}. \end{cases} \tag{4}$$

If f is monotonically increasing and smooth, then based on (4) it is possible to define smooth approximations for the maximum function:

$$\begin{aligned} &\text{A)}\ \max\{u_1,\ldots,u_K\} \approx f^{-1}\left(\sum_{k=1}^{K} f(u_k)\right), \\ &\text{B)}\ \max\{u_1,\ldots,u_K\} \approx f^{-1}\left(\frac{1}{K}\sum_{k=1}^{K} f(u_k)\right), \\ &\text{C)}\ \max\{u_1,\ldots,u_K\} \approx \sum_{j=1}^{K} \frac{f(u_j)}{\sum_{k=1}^{K} f(u_k)} u_j. \end{aligned} \tag{5}$$

Schemes A and B are similar: the only difference between them is the $\frac{1}{K}$ factor appearing in B. An advantage of A over B is that it approximates the max function better, if the difference between $u_{\max}$ and the other numbers is large. An advantage of B over A is that its result is always between $u_{\min}$ and $u_{\max}$. Scheme C is an interesting one: it calculates the answer by assigning a weight to each variable, and it does not need the inverse of f. It is also true for C that the output is always between $u_{\min}$ and $u_{\max}$.

The most natural choice for f is the exponential function $f(u) = \exp(\alpha u)$,$\alpha > 0$. The power function $f(u) = u^{\alpha}$,$\alpha > 1$ is also suitable in the nonnegative domain. With the given approximation schemes and f functions we can define 6 different smooth maximum

functions:

$$\begin{aligned}
\mathrm{smax}_{A1}(\mathbf{u}) &= \frac{1}{\alpha}\ln\left(\sum_{k=1}^{K}\exp(\alpha u_k)\right)\\
\mathrm{smax}_{A2}(\mathbf{u}) &= \left(\sum_{k=1}^{K}u_k^{\alpha}\right)^{1/\alpha}\\
\mathrm{smax}_{B1}(\mathbf{u}) &= \frac{1}{\alpha}\ln\left(\frac{1}{K}\sum_{k=1}^{K}\exp(\alpha u_k)\right)\\
\mathrm{smax}_{B2}(\mathbf{u}) &= \left(\frac{1}{K}\sum_{k=1}^{K}u_k^{\alpha}\right)^{1/\alpha}\\
\mathrm{smax}_{C1}(\mathbf{u}) &= \sum_{j=1}^{K}\frac{\exp(\alpha u_j)}{\sum_{k=1}^{K}\exp(\alpha u_k)}u_j\\
\mathrm{smax}_{C2}(\mathbf{u}) &= \sum_{j=1}^{K}\frac{u_j^{\alpha}}{\sum_{k=1}^{K}u_k^{\alpha}}u_j
\end{aligned} \tag{6}$$

where $\mathbf{u} = [u_1,\ldots,u_K]$ denotes the vector containing all numbers. Parameter α can be used to control the "degree of smoothness" (larger α results better approximation, but less smooth functions). Note that smax_{A1} and smax_{B1} differ only in a constant, and smax_{A2}, smax_{B2}, smax_{C2} are admissible only if $u_1,\ldots,u_K$ are all non-negative [4]. The surface plot of the maximum function in 2 dimensions can be seen in Figure (1). The presented smooth maximum functions are depicted in Figure (2) and their difference from $\max$ in Figure (3).

Two simple properties of the maximum function are interchangeability with constant addition and non-negative constant multiplication:

$$\begin{aligned}
\max\{u_1 + C,\ldots,u_K + C\} &= \max\{u_1,\ldots,u_K\} + C,\\
\max\{Cu_K,\ldots,Cu_K\} &= C\max\{u_1,\ldots,u_K\},
\end{aligned}$$

where C is an arbitrary constant in the first case and a non-negative constant in the second case. Interestingly, for 5 of the given smooth maximum functions exactly *one* of these properties is true (smax_{A1}, smax_{B1} and smax_{C1} have the first, smax_{B2} and smax_{C2} have the second property).

[4] $\mathrm{smax}_{C2}(\mathbf{0})$ can be defined as zero.

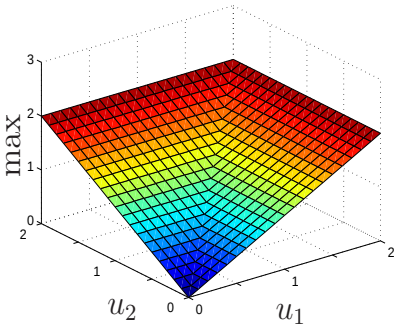


Figure 1: The maximum function in 2 dimensions.

Let us introduce the following abbreviation ($j = 1, \ldots, K$):

$$p_j = \frac{f(u_j)}{\sum_{k=1}^{K} f(u_k)}.$$

The quantity p_j can be interpreted as a "measure of dominance" of the j-th number over the others. If $f(u) = \exp(\alpha u)$, then $p_j = \frac{\exp(\alpha u_j)}{\sum_{k=1}^{K} \exp(\alpha u_k)}$. If $f(u) = u^\alpha$, then $p_j = \frac{u_j^\alpha}{\sum_{k=1}^{K} u_k^\alpha}$.
The partial derivatives of the proposed smooth maximum functions are ($j = 1, \ldots, K$):

$$\begin{aligned}
\mathrm{smax}'_{j,A1}(\mathbf{u}) &= \frac{\partial \mathrm{smax}_{A1}}{\partial u_j}(\mathbf{u}) = p_j, \\
\mathrm{smax}'_{j,A2}(\mathbf{u}) &= \frac{\partial \mathrm{smax}_{A2}}{\partial u_j}(\mathbf{u}) = p_j \frac{s}{u_j}, \\
\mathrm{smax}'_{j,B1}(\mathbf{u}) &= \frac{\partial \mathrm{smax}_{B1}}{\partial u_j}(\mathbf{u}) = p_j, \\
\mathrm{smax}'_{j,B2}(\mathbf{u}) &= \frac{\partial \mathrm{smax}_{B2}}{\partial u_j}(\mathbf{u}) = p_j \frac{s}{K u_j}, \\
\mathrm{smax}'_{j,C1}(\mathbf{u}) &= \frac{\partial \mathrm{smax}_{C1}}{\partial u_j}(\mathbf{u}) = p_j \left(1 + \alpha(u_j - s)\right), \\
\mathrm{smax}'_{j,C2}(\mathbf{u}) &= \frac{\partial \mathrm{smax}_{C2}}{\partial u_j}(\mathbf{u}) = p_j \left(1 + \alpha\left(1 - \frac{s}{u_j}\right)\right),
\end{aligned} \tag{7}$$

where s is the value of smax at $\mathbf{u}$ (always the same smooth max type is used as on the corresponding left hand side).

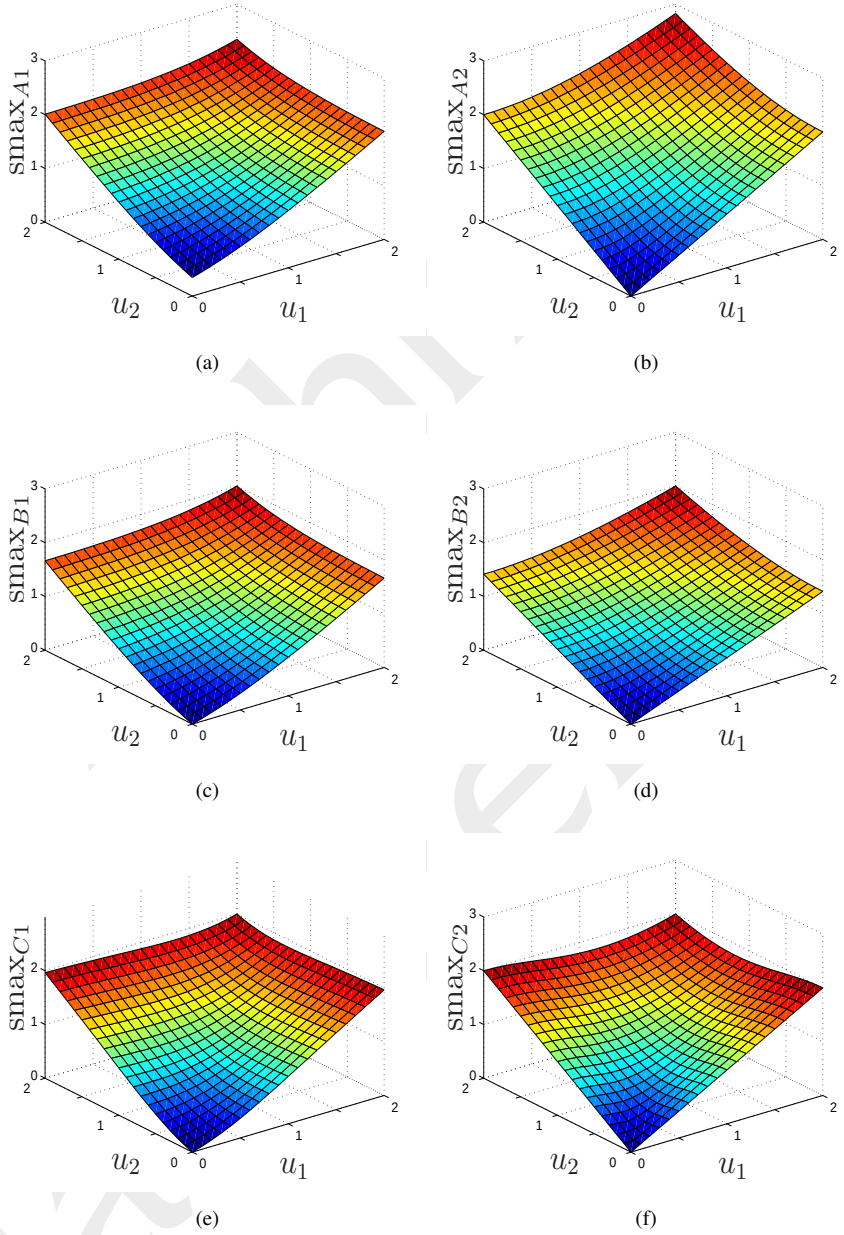


Figure 2: Smooth maximum functions in 2 dimensions ($\alpha = 2$).

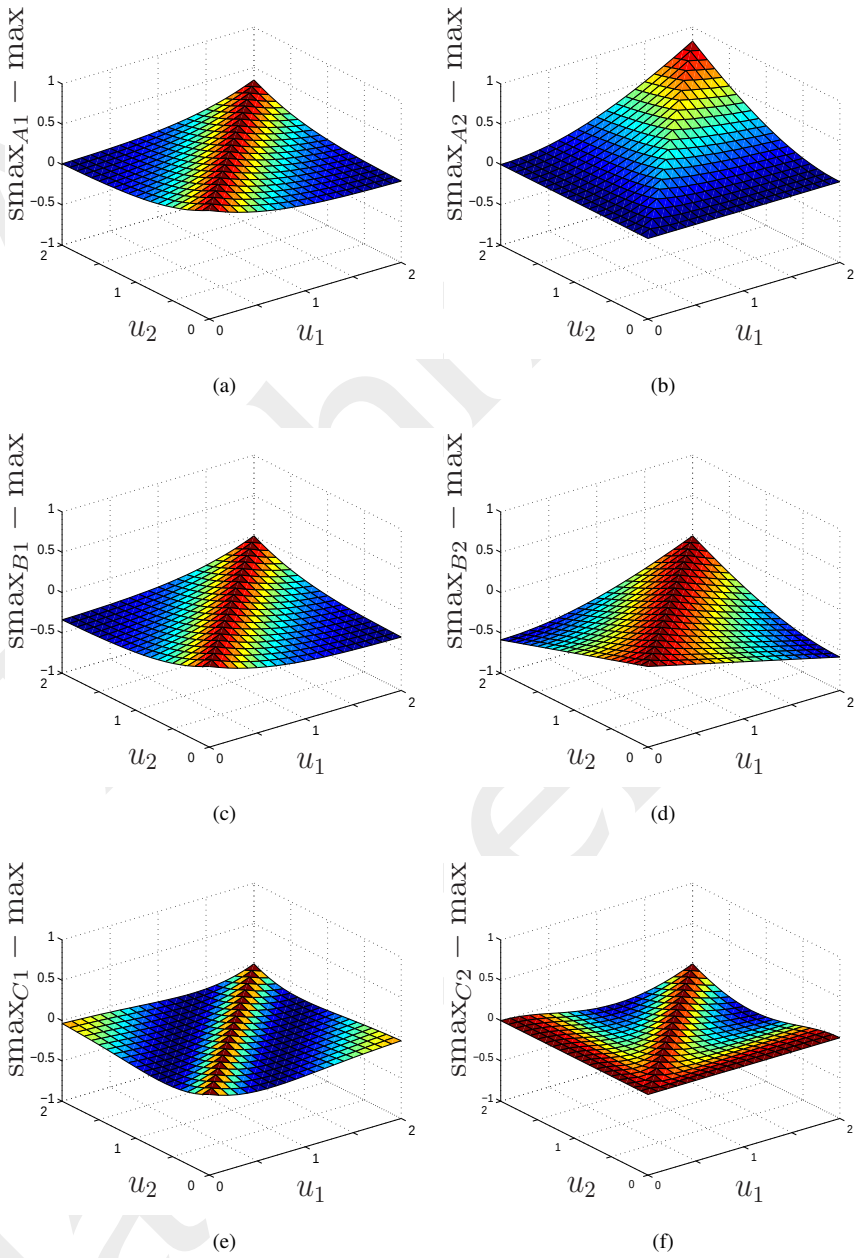


Figure 3: The error of smooth maximum functions in 2 dimensions ($\alpha = 2$).

Interestingly, each derivative contains the factor p_j. In the case of power function based approximations (smax_{A2}, smax_{B2} and smax_{C2}), the derivative also depends on the ratio of the approximated maximum and the j-th number. In the case of smax_{C1}, the derivative also depends on the difference of the approximated maximum and the j-th number.

The second partial derivatives are the following (j,$k = 1,\ldots,K$):

$$
\begin{aligned}
\mathrm{smax}''_{jk,A1}(\mathbf{u}) &= \frac{\partial^2 \mathrm{smax}_{A1}}{\partial u_j \partial u_k}(\mathbf{u}) = (-p_j p_k + \delta_{jk} p_j)\alpha, \\
\mathrm{smax}''_{jk,A2}(\mathbf{u}) &= \frac{\partial^2 \mathrm{smax}_{A2}}{\partial u_j \partial u_k}(\mathbf{u}) = (-p_j p_k + \delta_{jk} p_j)\frac{(\alpha-1)s}{u_j u_k}, \\
\mathrm{smax}''_{jk,B1}(\mathbf{u}) &= \frac{\partial^2 \mathrm{smax}_{B1}}{\partial u_j \partial u_k}(\mathbf{u}) = (-p_j p_k + \delta_{jk} p_j)\alpha, \\
\mathrm{smax}''_{jk,B2}(\mathbf{u}) &= \frac{\partial^2 \mathrm{smax}_{B2}}{\partial u_j \partial u_k}(\mathbf{u}) = (-p_j p_k + \delta_{jk} p_j)\frac{(\alpha-1)s}{K^2 u_j u_k}, \\
\mathrm{smax}''_{jk,C1}(\mathbf{u}) &= \frac{\partial^2 \mathrm{smax}_{C1}}{\partial u_j \partial u_k}(\mathbf{u}) = \left(-p_j s'_k - p_k s'_j + \delta_{jk}(s'_j + p_j)\right)\alpha, \\
\mathrm{smax}''_{jk,C2}(\mathbf{u}) &= \frac{\partial^2 \mathrm{smax}_{C2}}{\partial u_j \partial u_k}(\mathbf{u}) = \left(-\frac{p_j s'_k}{u_j} - \frac{p_k s'_j}{u_k} + \delta_{jk}\left(\frac{s'_j}{u_j} + \frac{p_j s}{u_j^2}\right)\right)\alpha,
\end{aligned}
\tag{8}
$$

where $\delta_{jk} = I\{j = k\}$ is the Kronecker delta symbol and s'_j is the value of $\frac{\partial \mathrm{smax}}{\partial u_j}$ at $\mathbf{u}$ (always the same smooth max type is used as on the corresponding left hand side).

4. Smooth maximum based training

A large family of training algorithms can be introduced for convex polyhedron classifiers with the help of smooth maximum functions. One branching point is what smooth maximum type to use. Another is how to approximate the convex polyhedron classifier itself.

Let us introduce the notation $\mathbf{z} = [z_1,\ldots,z_K] = [\mathbf{w}_1^T\mathbf{x} + b_1,\ldots,\mathbf{w}_K^T\mathbf{x} + b_K]$. Three equivalent forms of the convex polyhedron classifier are:

$$
\begin{aligned}
g(\mathbf{x}) &= \mathrm{th}(\min\{z_1,\ldots,z_K\}) \\
&= \min\{\mathrm{th}(z_1),\ldots,\mathrm{th}(z_K)\} \\
&= \min\{\mathrm{th}(z_1) - 1,\ldots,\mathrm{th}(z_K) - 1\} + 1
\end{aligned}
$$

Using the maximum function the previous formulae can be written as

$$\begin{aligned} g(\mathbf{x}) &= \mathrm{th}(-\max\{-z_1,\ldots,-z_K\}) \\ &= -\max\{-\mathrm{th}(z_1),\ldots,-\mathrm{th}(z_K)\} \\ &= -\max\{1-\mathrm{th}(z_1),\ldots,1-\mathrm{th}(z_K)\}+1. \end{aligned}$$

Note that in the third case we always take the maximum of positive numbers.

Now we are ready to introduce smooth versions of g: $\max$ can be replaced with a smooth max and $\mathrm{th}(\gamma)$ with $\mathrm{sgm}(\gamma) = 1/(1+\exp(-\gamma))$ or $\gamma + 0.5$, where sgm is called the logistic sigmoid function. After filtering out some irrelevant combinations we get the following smooth versions of g:

$$\begin{aligned} h_A(\mathbf{x}) &= \mathrm{sgm}(-\mathrm{smax}(-z_1,\ldots,-z_K)), \\ h_B(\mathbf{x}) &= -\mathrm{smax}(-z_1,\ldots,-z_K)+0.5, \\ h_C(\mathbf{x}) &= -\mathrm{smax}(1-\mathrm{sgm}(z_1),\ldots,1-\mathrm{sgm}(z_K))+1. \end{aligned}$$

In the first two cases, smax takes value from $\{\mathrm{smax}_{A1},\mathrm{smax}_{B1},\mathrm{smax}_{C1}\}$. In the third case, smax takes value from $\{\mathrm{smax}_{A1},\mathrm{smax}_{A2},\mathrm{smax}_{B1},\mathrm{smax}_{B2},\mathrm{smax}_{C1},\mathrm{smax}_{C2}\}$.

It will be useful to unify the three branches by decomposing h functions into three parts:

$$h(\mathbf{x}) = h_2(\mathrm{smax}(h_1(\mathbf{z_1}),\ldots,h_1(\mathbf{z_K}))),$$

where h_1 and h_2 are $\mathbb{R} \mapsto \mathbb{R}$ mappings. The h_1 and h_2 parts of the given h functions are the following:

$$\begin{aligned} h_{A1}(z) &= -z, & h_{A2}(s) &= \mathrm{sgm}(-s), \\ h_{B1}(z) &= -z, & h_{B2}(s) &= -s+0.5, \\ h_{C1}(z) &= 1-\mathrm{sgm}(z), & h_{C2}(s) &= -s+1. \end{aligned} \tag{9}$$

The first and the second derivatives of the above functions are:

$$\begin{aligned} h'_{A1}(z) &= -1, & h'_{A2}(s) &= -h_{A2}(s)(1-h_{A2}(s)), \\ h'_{B1}(z) &= -1, & h'_{B2}(s) &= -1, \\ h'_{C1}(z) &= -h_{C1}(z)(1-h_{C1}(z)), & h'_{C2}(s) &= -1, \end{aligned} \tag{10}$$

$$\begin{aligned} h''_{A1}(z) &= 0, & h''_{A2}(s) &= -h'_{A2}(s)(1-2h_{A2}(s)), \\ h''_{B1}(z) &= 0, & h''_{B2}(s) &= 0, \\ h''_{C1}(z) &= h'_{C1}(z)(1-2h_{C1}(z)), & h''_{C2}(s) &= 0. \end{aligned} \tag{11}$$

Let us denote the output of h for input $\mathbf{x}$ by $a = h(\mathbf{x})$. The error of the classifier on example $(\mathbf{x},y)$ can be measured with differentiable loss functions. Two possible choices are the squared loss and the logistic loss:

$$\begin{aligned} \mathrm{loss}_S(a,y) &= \tfrac{1}{2}(a-y)^2, \\ \mathrm{loss}_L(a,y) &= -\ln\left(a^y(1-a)^{1-y}\right). \end{aligned} \tag{12}$$

In the first case, h takes value from $\{h_A,h_B,h_C\}$. In the second case, a has to fall into $[0,1]$, therefore h takes value from $\{h_A,h_C\}$, but if $h = h_C$, then the smooth maximum function cannot be smax_{A1} or smax_{A2}.

The first and the second derivatives of the proposed loss functions with respect to a are:

$$\begin{aligned} \mathrm{loss}'_S(a,y) &= \frac{\partial \mathrm{loss}_S}{\partial a}(a,y) = a-y, \\ \mathrm{loss}'_L(a,y) &= \frac{\partial \mathrm{loss}_L}{\partial a}(a,y) = \frac{1-y}{1-a} - \frac{y}{a}, \end{aligned} \tag{13}$$

$$\begin{aligned} \mathrm{loss}''_S(a,y) &= \frac{\partial^2 \mathrm{loss}_S}{\partial^2 a^2}(a,y) = 1, \\ \mathrm{loss}''_S(a,y) &= \frac{\partial^2 \mathrm{loss}_L}{\partial^2 a^2}(a,y) = \frac{1-y}{(1-a)^2} - \frac{y}{a^2}. \end{aligned} \tag{14}$$

Based on the per example loss, the regularized total loss can be defined as

$$\mathcal{L}(b_1,\mathbf{w}_1,\ldots,b_K,\mathbf{w}_K) = \left(\sum_{i=1}^{n} \mathrm{loss}(h(\mathbf{x}_i),y_i)\right) + \lambda\left(\frac{1}{2}\sum_{j=1}^{K}\mathbf{w}_j^T\mathbf{w}_j\right), \tag{15}$$

where $\mathrm{loss} \in \{\mathrm{loss}_S,\mathrm{loss}_L\}$, and λ is called regularization coefficient. The number of allowed choices for $(\mathrm{smax},h,\mathrm{loss})$ is 19. In every case, a local minimum of $\mathcal{L}$ can be found by derivative based algorithms. This proposed approach of training convex polyhedron classifiers will be referred as SMAX from now.

It is important to note that smooth approximations are used only during the training. In the classification phase, the original formula of the convex polyhedron classifier is applied. Obviously, using different prediction formulae at training and classification may deteriorate the accuracy. A possible way to to handle this problem is to gradually decrease the smoothness of the approximation during the training by increasing the value of α.

The first proposed training method uses stochastic gradient descent for the approximate minimization of $\mathcal{L}$. The pseudo-code of the algorithm can be seen in Figure (4).

Input: $(\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_n,y_n)$ // the training set
Input: smax, α, h, loss, K, R, E, B, η, μ, λ, A_0, A_1 // meta-parameters
Output: $(\mathbf{w}_1,b_1),\ldots,(\mathbf{w}_K,b_K)$ // the trained model

$(\mathbf{w}_1,b_1),\ldots,(\mathbf{w}_K,b_K) \leftarrow$ uniform random numbers from $[-R,R]$
// initialization
$(\mathbf{w}_1^{\text{old}},b_1^{\text{old}}),\ldots,(\mathbf{w}_K^{\text{old}},b_K^{\text{old}}) \leftarrow (\mathbf{w}_1,b_1),\ldots,(\mathbf{w}_K,b_K)$
$(\mathbf{w}'_1,b'_1),\ldots,(\mathbf{w}'_K,b'_K) \leftarrow$ zeros

macro AccumlateGradient(i) **begin**
    for $j \leftarrow 1$ **to** K **do** $z_j \leftarrow \mathbf{w}_j^T\mathbf{x}_i + b_j$ // calculate branch activations
    $\mathbf{u} \leftarrow [h_1(z_1),\ldots,h_1(z_K)]^T$
    $s \leftarrow \text{smax}(\mathbf{u})$
    $a \leftarrow h_2(s)$ // calculate answer
    for $j \leftarrow 1$ **to** K **do** // update gradient
        $c'_j \leftarrow \text{loss}'(a,y_i) \cdot h'_2(s) \cdot \text{smax}'_j(\mathbf{u}) \cdot h'_1(z_j)$
        $\mathbf{w}'_j \leftarrow \mathbf{w}'_j + c'_j\mathbf{x}_i + \lambda\mathbf{w}_j/n$
        $b'_j \leftarrow b'_j + c'_j$
    end
end

for $e \leftarrow 1$ **to** E **do** // for all epochs
    $\alpha \leftarrow A_1\alpha + A_0$ // update smoothness
    for $i \leftarrow 1$ **to** n **do** // for all examples
        AccumlateGradient(i)
        if $i \equiv 0 \pmod{B}$ **then** // update model
            for $j \leftarrow 1$ **to** K **do**
                $\Delta \leftarrow \mathbf{w}_j - \mathbf{w}_j^{\text{old}}, \quad \mathbf{w}_j^{\text{old}} \leftarrow \mathbf{w}_j$
                $\mathbf{w}_j \leftarrow \mathbf{w}_j - \eta\mathbf{w}'_j + \mu\Delta$
                $\Delta \leftarrow b_j - b_j^{\text{old}}, \quad b_j^{\text{old}} \leftarrow b_j$
                $b_j \leftarrow b_j - \eta b'_j + \mu\Delta$
            end
            $(\mathbf{w}'_1,b'_1),\ldots,(\mathbf{w}'_K,b'_K) \leftarrow$ zeros // reset gradient
        end
    end
end

Figure 4: Stochastic gradient descent with momentum for training the convex polyhedron classifier.

The meanings of the algorithm's meta-parameters are as follows:

- $\mathrm{smax} \in \{\mathrm{smax}_{A1}, \mathrm{smax}_{A2}, \mathrm{smax}_{B1}, \mathrm{smax}_{B2}, \mathrm{smax}_{C1}, \mathrm{smax}_{C2}\}$: smooth max function,
- $\alpha \in \mathbb{R}$: initial value of the smoothness parameter,
- $h \in \{h_A, h_B, h_C\}$: smooth replacement of g,
- $\mathrm{loss} \in \{\mathrm{loss}_S, \mathrm{loss}_L\}$: per example loss function,
- $K \in \mathbb{N}$: number of hyperplanes in the convex polyhedron classifier,
- $R \in \mathbb{R}$: range of random number generation at model initialization,
- $E \in \mathbb{N}$: number of epochs (iterations over the training set),
- $B \in \mathbb{N}$: batch size — the model is updated after each B example,
- $\eta \in \mathbb{R}$: learning rate — step size at model update,
- $\mu \in \mathbb{R}$: momentum factor — the weight of the previous update in the current one,
- $\lambda \in \mathbb{R}$: regularization coefficient — how aggressively the weights are pushed towards 0,
- $A_0, A_1 \in \mathbb{R}$: coefficients for controlling the change of α.

The time requirement of one iteration is $O(ndK)$, and the time requirement of the algorithm is $O(EndK)$, therefore the algorithm can be run on very large problems. In practice it is not always necessary to find a local minimum. It is often enough to reach a sufficiently small objective function value. Of course, there is no guarantee that the trained model will be acceptable after a modest number of iterations, but at least we are able to test it.

The second proposed training algorithm uses Newton's method for the approximate minimization of $\mathcal{L}$. The pseudo-code of the algorithm can be seen in Figure (5). The meta-parameters of the algorithm are the same as before except that there is no batch size B, learning rate η, and momentum factor μ, and there is a new parameter S, the number of step sizes tried before model update. The role of parameter S is to make the algorithm more stable.

The time requirement of one iteration is $O(nd^2K^2 + d^3K^3)$ and the time requirement of the algorithm is $O(End^2K^2 + Ed^3K^3)$. An advantage of Newton's method over stochastic gradient descent is better accuracy. A disadvantage is the substantially increased time complexity of iterations. It may happen that we are unable to run even one iteration.

It is also true that Newton's method is typically less robust than gradient method. It is more sensible to stuck in minor local minima, and also it is more prone to diverge. A possible way to overcome these difficulties is to introduce a hybrid approach that starts the minimization with gradient method, and then switches to Newton's method.

Input: $(\mathbf{x}_1,y_1),\ldots,(\mathbf{x}_n,y_n)$ // the training set
Input: smax, α, h, loss, K, R, E, λ, A_0, A_1, S // meta-parameters
Output: $(\mathbf{w}_1,b_1),\ldots,(\mathbf{w}_K,b_K)$ // the trained model

$(\mathbf{w}_1,b_1),\ldots,(\mathbf{w}_K,b_K) \leftarrow$ uniform random numbers from $[-R,R]$ // initialization
$(\mathbf{w}'_1,b'_1),\ldots,(\mathbf{w}'_K,b'_K)$, $\mathbf{H}$, $\mathbf{g} \leftarrow$ zeros
$\mathcal{L}_{\min} \leftarrow \infty$

macro AccumlateHessian(i) **begin**
    $\mathbf{x}_{i0} \leftarrow 1$ // consider the 0-th coordinate as 1
    for j, k **in** $\{1,\ldots K\} \times \{1,\ldots K\}$ **do**
        $c''_{jk} \leftarrow \mathrm{loss}''(a,y_i) \cdot h'_2(s)^2 \cdot \mathrm{smax}'_j(\mathbf{u})\mathrm{smax}'_k(\mathbf{u}) \cdot h'_1(z_j)h'_1(z_k) +$
            $\mathrm{loss}'(a,y_i) \cdot h''_2(s) \cdot \mathrm{smax}'_j(\mathbf{u})\mathrm{smax}'_k(\mathbf{u}) \cdot h'_1(z_j)h'_1(z_k) +$
            $\mathrm{loss}'(a,y_i) \cdot h'_2(s) \cdot \mathrm{smax}''_{jk}(\mathbf{u}) \cdot h'_1(z_j)h'_1(z_k) +$
            $\mathrm{loss}'(a,y_i) \cdot h'_2(s) \cdot \mathrm{smax}'_j(\mathbf{u})\delta_{jk} \cdot h''_1(z_j)\delta_{jk}$

        for l, m **in** $\{0,\ldots,d\} \times \{0,\ldots,d\}$ **do** // update Hessian
            $\widehat{j} \leftarrow (j-1)(d+1)+l+1$
            $\widehat{k} \leftarrow (k-1)(d+1)+m+1$
            $h_{\widehat{j}\widehat{k}} \leftarrow h_{\widehat{j}\widehat{k}} + c''_{jk}x_{il}x_{im} + \lambda\delta_{l0}\delta_{m0}$
        end
    end
end

for $e \leftarrow 1$ **to** E **do** // for all epochs
    $\alpha \leftarrow A_1\alpha + A_0$ // update smoothness

    for $i \leftarrow 1$ **to** n **do** // for all examples
        AccumlateGradient(i)
        AccumlateHessian(i)
    end

    $\mathbf{v} \leftarrow [(b_1w_{11}\cdots w_{1d})\cdots(b_Kw_{K1}\cdots w_{Kd})]^T$
    $\mathbf{g} \leftarrow [(b'_1w'_{11}\cdots w'_{1d})\cdots(b'_Kw'_{K1}\cdots w'_{Kd})]^T$
    for σ **in** $\{1,2^{-1},\ldots,2^{-S+2},0\}$ **do** // try S step sizes
        $\mathbf{v}_{\mathrm{new}} \leftarrow \mathbf{v} - \sigma\mathbf{H}^{-1}\mathbf{g}$
        $\mathcal{L}_{\mathrm{new}} \leftarrow \mathcal{L}(\mathbf{v}_{\mathrm{new}})$ // use (15)
        if $\mathcal{L}_{\mathrm{new}} < \mathcal{L}_{\min}$ **then** $\mathcal{L}_{\min} \leftarrow \mathcal{L}_{\mathrm{new}}$, $\mathbf{v}_{\mathrm{best}} \leftarrow \mathbf{v}_{\mathrm{new}}$
    end
    $[(b_1w_{11}\cdots w_{1d})\cdots(b_Kw_{K1}\cdots w_{Kd})] \leftarrow \mathbf{v}_{\mathrm{best}}^T$ // update model
    $(\mathbf{w}'_1,b'_1),\ldots,(\mathbf{w}'_K,b'_k)$, $\mathbf{H}$, $\mathbf{g} \leftarrow$ zeros // reset gradient and Hessian
end

Figure 5: Newton's method for training the convex polyhedron classifier.

4.1. Algorithms for regression

Convex polyhedron regression can be introduced analogously with convex polyhedron classification. A *convex K-polyhedron predictor* is a function $g : \mathbb{R}^d \mapsto \mathbb{R}$ that can be written in the following form:

$$\begin{aligned} g(\mathbf{x}) &= \min\{\mathbf{w}_1^T\mathbf{x} + b_1, \ldots, \mathbf{w}_K^T\mathbf{x} + b_K\} \\ &= -\max\{-\mathbf{w}_1^T\mathbf{x} - b_1, \ldots, -\mathbf{w}_K^T\mathbf{x} - b_K\}. \end{aligned} \tag{16}$$

The only difference from convex the K-polyhedron classifier is that the threshold function is missing. The reason behind using the term "convex polyhedron" here is that the set $\{(\mathbf{x},y) \in \mathbb{R}^{d+1} : y \leq g(\mathbf{x})\}$ is a convex polyhedron in $\mathbb{R}^{d+1}$.

Like in the case of classification, it is possible define smooth maximum based algorithms for training. The only difference is that now the only reasonable choice for h and loss is h_B and loss_S, because the target takes value from $\mathbb{R}$. Apart from this restriction, the training algorithms remain the same.

Note that in the case of regression we always have to evaluate all scalar products at prediction. Therefore, unlike the case of classification, there is no extra speedup in the prediction phase, however, the prediction is still not slow. Applying a convex polyhedron predictor can be a reasonable choice, if we know a priori that the optimal predictor g^* is convex.

4.2. Algorithms for collaborative filtering

In the case of collaborative filtering we can obtain a convex polyhedron approach via the generalization of matrix factorization. The answer of the convex polyhedron predictor for user u and item i is

$$g(u,i) = b_u + c_i - \max\left\{-\left(\sum_{l=1}^{L} p_{ul}^{(1)} q_{li}\right), \ldots, -\left(\sum_{l=1}^{L} p_{ul}^{(K)} q_{li}\right)\right\},$$

where $\mathbf{P}^{(k)} \in \mathbb{R}^{N_U \times L}$, $[\mathbf{P}^{(k)}]_{ul} = p_{ul}^{(k)}$, $k = 1, \ldots, K$ called user factor matrices, $\mathbf{Q} \in \mathbb{R}^{L \times N_I}$, $[\mathbf{Q}]_{li} = q_{li}$ called item factor matrix, $\mathbf{b} \in \mathbb{R}^{N_U}$ called user bias vector, and $\mathbf{c} \in \mathbb{R}^{N_I}$ called item bias vector are the parameters of the model.

An analogous variant can be obtained, if we have one user factor matrix $\mathbf{P}$ and K item factor matrices $\mathbf{Q}^{(1)}, \ldots, \mathbf{Q}^{(K)}$:

$$g(u,i) = b_u + c_i - \max\left\{-\left(\sum_{l=1}^{L} p_{ul} q_{li}^{(1)}\right), \ldots, -\left(\sum_{l=1}^{L} p_{ul} q_{li}^{(K)}\right)\right\}.$$

Let us assume the first variant and introduce the notation $\mathbf{z} = [z_1,\ldots,z_K]$, $z_k = \sum_{l=1}^{L} p_{ul}^{(k)} q_{li}$ $(k = 1,\ldots,K)$. The smooth version of g can be obtained as

$$h(u,i) = b_u + c_i + \mathrm{smax}(\mathbf{z}),$$

where $\mathrm{smax} \in \{\mathrm{smax}_{A1},\mathrm{smax}_{B1},\mathrm{smax}_{C1}\}$.

Now it is possible measure the error at example (u,i) with a differentiable loss function:

$$\mathcal{L}_{ui}(\mathbf{w}) = \frac{1}{2}\left(h(u,i) - r_{ui}\right)^2 + \lambda_U \frac{1}{2}\sum_{k=1}^{K}\sum_{l=1}^{L}\left(p_{ul}^{(k)}\right)^2 + \lambda_I \frac{1}{2}\sum_{l=1}^{L}\left(q_{li}\right)^2,$$

where $\mathbf{w}$ denotes the vector containing all parameters of the model ($\mathbf{P}^{(1)},\ldots,\mathbf{P}^{(K)},\mathbf{Q},\mathbf{b},\mathbf{c}$), and λ_U, λ_I are the regularization coefficients. The total loss on the training set is the sum of the per example losses:

$$\mathcal{L}(\mathbf{w}) = \sum_{(u,i)\in\mathcal{T}} \mathcal{L}_{ui}(\mathbf{w}).$$

Similarly to classification and regression, the approximate minimization of $\mathcal{L}$ can be done with stochastic gradient descent. This approach of training the convex polyhedron predictor will be referred as $\mathrm{SMAX}_{\mathrm{CF}}$. Note that in the case of collaborative filtering the typical problem size is large (say $N_U,N_I > 1000$, $L > 10$), therefore Newton's method is computationally too expensive.

The partial derivatives of $\mathcal{L}_{ui}$ can be written as

$$\begin{aligned}
\frac{\partial \mathcal{L}_{ui}}{\partial p_{ul}^{(k)}}(\mathbf{w}) &= \left(h(u,i) - r_{ui}\right)\left(\mathrm{smax}_k'(\mathbf{z}) q_{ui}\right) + \lambda_U p_{ul}^{(k)}, \\
\frac{\partial \mathcal{L}_{ui}}{\partial q_{li}}(\mathbf{w}) &= \left(h(u,i) - r_{ui}\right)\left(\sum_{k=1}^{K} \mathrm{smax}_k'(\mathbf{z}) p_{ul}^{(k)}\right) + \lambda_I q_{li}, \\
\frac{\partial \mathcal{L}_{ui}}{\partial b_u}(\mathbf{w}) &= h(u,i) - r_{ui}, \\
\frac{\partial \mathcal{L}_{ui}}{\partial c_i}(\mathbf{w}) &= h(u,i) - r_{ui},
\end{aligned} \tag{17}$$

Note that the second equation builds upon the assumption $\mathrm{smax} \in \{\mathrm{smax}_{A1},\mathrm{smax}_{B1},\mathrm{smax}_{C1}\}$, and it would *not* be true, if smax was an arbitrary differentiable function.

The pseudo-code of stochastic gradient descent based training can be seen in Figure (6). The meanings of the meta-parameters are the same as before, except that now we have different learning rate and regularization coefficient for users and items. The role of parameter D is to control whether ordering by date within user ratings should be used.

Input: $r_{ui} : (u,i) \in \mathcal{T}, |\mathcal{T}| = n$ // the training set
Input: smax, α, K, R, E, η_U, η_I, λ_U, λ_I, D, A_0, A_1 // meta-parameters
Output: $\mathbf{P}^{(1)}, \ldots, \mathbf{P}^{(K)}, \mathbf{Q}$ // the trained model

$\mathbf{P}^{(1)}, \ldots, \mathbf{P}^{(K)}$, $\mathbf{Q}$, $\mathbf{b}$, $\mathbf{c} \leftarrow$ uniform random numbers from $[-R,R]$
// initialization

for $e \leftarrow 1$ **to** E **do** // for all epochs
  $\alpha \leftarrow A_1\alpha + A_0$ // update smoothness

  for $u \leftarrow 1$ **to** N_U **do** // for all users
    $\mathcal{T}_u \leftarrow \{i : \exists u : (u,i) \in \mathcal{T}\}$
    $\mathcal{I} \leftarrow$ a random permutation of the elements of $\mathcal{T}_u$
    if $D = 1$ **and** *dates are available for ratings* **then**
      $\mathcal{I} \leftarrow$ the elements of $\mathcal{T}_u$ sorted by rating date (in ascending order)
    end

    for i **in** $\mathcal{I}$ **do** // for user's ratings
      for $k \leftarrow 1$ **to** K **do** $z_k \leftarrow \sum_{l=1}^{L} p_{ul}^{(k)} q_{li}$
      for $k \leftarrow 1$ **to** K **do** $s'_k \leftarrow \mathrm{smax}'_k(-\mathbf{z})$
      $a \leftarrow b_u + c_i - \mathrm{smax}(-\mathbf{z})$ // calculate answer
      $\varepsilon \leftarrow a - y_i$ // calculate error

      $b_u \leftarrow b_u - \eta_U \varepsilon$ // update biases
      $c_i \leftarrow c_i - \eta_I \varepsilon$

      for $l \leftarrow 1$ **to** L **do** // update factors
        $p \leftarrow \sum_{k=1}^{K} s'_k p_{ul}^{k}$
        for $k \leftarrow 1$ **to** K **do** $p_{ul}^{(k)} \leftarrow p_{ul}^{(k)} - \eta_U(\varepsilon s'_k q_{li} + \lambda_U p_{ul}^{(k)})$
        $q_{li} \leftarrow q_{li} - \eta_I(\varepsilon p + \lambda_I q_{li})$
      end
    end
  end
end

Figure 6: Stochastic gradient descent for training the convex polyhedron predictor.

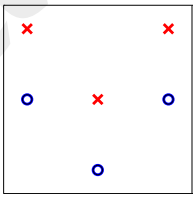


Figure 7: The TOY dataset.

5. Experiments

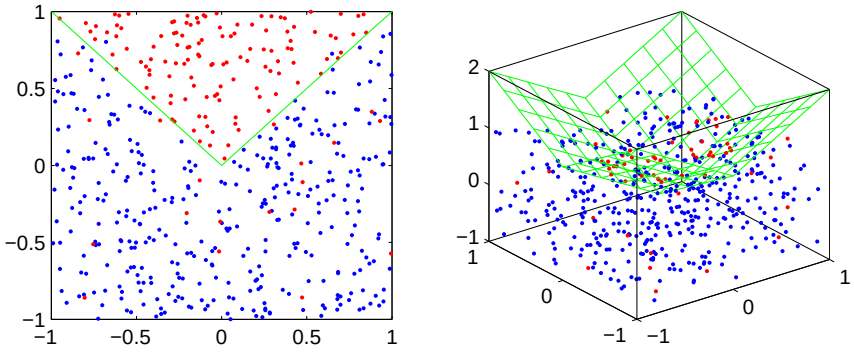
5.1. Classification

In this section, we will compare convex polyhedron classification algorithms with other methods both on artificial and real-life datasets. The artificial datasets involved in the experiments are the following:

- **TOY**: This dataset contains 6 examples: $\mathbf{x}_1 = [-1,2], y_1 = 1$, $\mathbf{x}_2 = [0,1], y_2 = 1$, $\mathbf{x}_3 = [1,2], y_3 = 1$, $\mathbf{x}_4 = [-1,1], y_4 = 0$, $\mathbf{x}_5 = [0,0], y_5 = 0$, $\mathbf{x}_6 = [1,2], y_6 = 0$. The classes can be separated from each other with 2 lines (see Figure 7). One may find it interesting to analyze the differences between the many proposed training algorithms on such an extremely simple dataset.
- **V2**: Let us define the V distribution as the following: The components of the input vector $\mathbf{X}$ are drawn independently, according to uniform distribution over $[-1, +1]$. If $X_d \geq \sum_{j=1}^{d-1} |X_j|$, then the class label Y is set to 1, otherwise it is set to 0. Finally, the value of Y is flipped with probability α. Note, that the Bayes classifier for the V distribution is a convex 2^{d-1}-polyhedron classifier. The V2 dataset contains $n = 10^5$ examples generated according to the V distribution with settings $d = 2$ and $\alpha = 0.05$ (see Figure 8).
- **V3**: The 3-dimensional ($d = 3$, $n = 10^5$) version of the previous dataset (see Figure 8).

The real-life datasets involved in the experiments were extracted from the UCI machine learning repository [2]. Convex polyhedron classification assumes two classes, therefore all problems were transformed to binary ones by merging classes. The specific datasets were the following:

- **ABALONE**: Here the task is to predict from various physical characteristics (e.g. length, diameter, height) whether the number of rings of an abalone is greater than 12. The number of input features in the dataset after variable encoding is $d = 10$, and the number of of examples is $n = 4177$ (16.6 % of the examples belong to the positive



(a) (b)

Figure 8: The V distribution with settings $d = 2$, $\alpha = 0.05$ (a) and $d = 3$, $\alpha = 0.05$ (b). The optimal decision boundary is indicated with green.

class).

- **BLOOD**: This dataset originates from the donor database of Blood Transfusion Service Center in Hsin-Chu City, Taiwan. The goal is to predict whether a donor donated blood at a given date in 2007. The dataset contains $d = 4$ input features (months since last donation, total number of donations, total blood donated in c.c., months since first donation), and $n = 748$ examples (23.8 % positives).
- **CHESS**: This dataset came from the domain of chess endgames. The $d = 6$ input features are integers, representing the location of the white king, the white rook and the black king. The task is to decide whether black can escape from being mated in 14 (or less) moves. The number of examples is $n = 28056$ (9.1 % positives).
- **SEGMENT**: The instances were drawn randomly from a database of 7 outdoor images. The images were hand-segmented to create a classification for every pixel. The task is to predict whether a pixel is part of a window object in the image. The number of features is $d = 19$, and the number of examples is $n = 2310$ (14.3 % positives).

The classification algorithms included in the comparison were the following:

- **FDA**: Regularized Fisher discriminant analysis [12]. Regularization was done by adding the term $\lambda \mathbf{w}^T \mathbf{w}$ to the denominator of the objective function, where $\mathbf{w} \in \mathbb{R}^d$ denotes the weight vector. The sole parameter of the algorithm is the regularization coefficient λ (default value: 10^{-6}).
- **LOGR**: Logistic regression [26] with L2 regularization applied on the weight vector. The minimization of the objective function was done by Newton's method. The

starting point was the all-zero vector. The algorithm has 2 parameters: the regularization coefficient λ (default value: 10^{-6}) and the number of iterations E.

- **SPER**: The smooth variant of Rosenblatt's perceptron [19] with L2 regularization. The activation function was the logistic sigmoid function. The minimization of the objective function was done by Newton's method, started from the all-zero vector. The algorithm has 2 parameters: the regularization coefficient λ (default value: 10^{-6}) and the number of iterations E.
- **ALN**: Adaptive linear neuron [25] with L2 regularization applied on the weight vector. The only parameter of the algorithm is the regularization coefficient λ (default value: 10^{-6}).
- **LSVM**: Linear support vector machine [6]. The algorithm has one parameter: the tradeoff coefficient C.
- **KNN**: K nearest neighbors [13]. The only parameter of the approach is the number of relevant neighbors K.
- **ID3**: ID3 decision tree [18]. All features were treated as continuous ones. The algorithm has 3 parameters: the number of splitting values tried K (default value: 10), the Laplace smoothing term β, and the information gain threshold $\mathcal{G}_{\min}$.
- **MLP**: Multilayer perceptron [24] with L2 regularization applied on the non-bias weights. The objective function was minimized by batch gradient descent with momentum. The parameters of the algorithm are the regularization coefficient λ (default value: 10^{-6}), the number of hidden units K (default value: 5), the range of random initialization R, the number of epochs E, the learning rate η, and the momentum factor μ.
- **SVM**: Support vector machine [6] with Gaussian kernel. The only parameter of the algorithm is the tradeoff coefficient C.
- **MR**: Convex polyhedron classifier with maximal rejection based training (see page 30). The parameters of the algorithm are the number of hyperplanes K, and the tolerance β.
- **SMAX**: The proposed smooth maximum function based approach for convex polyhedron classification (see page 37). The parameters of the algorithm are the training method (G: gradient method, see page 40, N: Newton's method, see page 42, G+N: start with gradient method, and then continue with Newton's method — default: G+N), the smooth max function (smax_{A1}, smax_{A2}, smax_{B1}, smax_{B2}, smax_{C1}, or smax_{C2} — default: smax_{A1}), the smoothness parameter α (default value: 2), the smoothness change coefficients A_1 and A_0 (default values: $A_1 = 1$, $A_0 = 0$) the h function (h_A, h_B, or h_C — default: h_A), the per example loss function (loss_S or loss_L — default: loss_S), the number of hyperplanes K, the range of random initialization R (default value: 1), the number of epochs in the G phase E, the number of epochs in the N phase E_2, the batch size B (default value: n, which means batch mode), the regularization coefficient λ (default value: 10^{-6}), the learning rate η, the momentum factor μ (default value: 0.95),

and the number of step sizes tried before Newton updates S (default value: 10).

For LSVM and SVM the libsvm [7] implementation was used, via the built-in Python interface. All other algorithms were implemented from scratch in Python [20], using the NumPy [1] module. The hardware environment was a notebook PC with Intel Pentium M 2 GHz CPU and 1 Gb memory. If the value of a parameter is not specified, then the default value is used.

5.1.1. Comparing the variants of SMAX

In these experiments I ran the variants of SMAX training on the TOY dataset. Every valid combination of optimization method, loss, h and smax were tested with the 3 different optimization methods (G, N, G+N). The parameters E (number of epochs in the G phase), η (learning rate), E_2 (number of epochs in the N phase), and R (range of random initialization) were set heuristically via "trial and error" for each setting. The other parameters were kept fixed at their default values. The results can be seen in Table 1 and Table 2. The meaning of the last 3 columns are:

- $\|\mathbf{W}\|$: The Frobenius norm of weight matrix part of the solution ($\sqrt{\sum_{k=1}^{K}\sum_{j=1}^{d} w_{kj}^2}$).
- $\mathcal{L}_{01}$: The number of training examples misclassified by the trained model.
- $\mathcal{L}$: The value of the regularized total loss at the solution.

It can be seen, that the G and the G+N methods were always able to build a classifier that does not err on the training set. In contrast, the N method sometimes converged to local minima with relatively high $\mathcal{L}$ value. This is because gradient descent with momentum is less prone to stuck in local minima than Newton's method (however, it needs more iterations to converge).

If we consider the categories defined by the second and third columns, then we can observe the following:

- **SA** ($\mathrm{loss} = \mathrm{loss}_S$, $h = h_A$): In this category the G+N method required a relatively short G phase. The lowest $\mathcal{L}$ value was achieved by $\mathrm{smax} = \mathrm{smax}_{C1}$.
- **SB** ($\mathrm{loss} = \mathrm{loss}_S$, $h = h_B$): This category required 2 magnitudes smaller learning rates than the other ones. (This is because in the case of h_B the changes of the weights are not "dampened" by the sigmoid function.) The $\|\mathbf{W}\|$ values were relatively small. The G+N method required a relatively long G phase. The lowest $\mathcal{L}$ value was achieved by $\mathrm{smax} = \mathrm{smax}_{C2}$.
- **SC** ($\mathrm{loss} = \mathrm{loss}_S$, $h = h_C$): The $\mathcal{L}$ value was typically higher than in SA and SB. The lowest $\mathcal{L}$ value was achieved by $\mathrm{smax} = \mathrm{smax}_{C2}$.
- **LA** ($\mathrm{loss} = \mathrm{loss}_L$, $h = h_A$): In this category the pure N method was more stable than in the other categories: it was always able to achieve zero misclassifications. The

variant	**loss**	**h**	**smax**	**method**	**parameters**	$\|\mathbf{W}\|$	$\mathcal{L}_{01}$	$\mathcal{L}$
#1	S	A	A1	G	$E = 1000, \eta = 0.1$	9.6	0	0.0478
#2	S	A	A1	N	$E_2 = 10, R = 0.5$	4.8	1	0.3493
#3	S	A	A1	G+N	$E = 50, \eta = 0.1, E_2 = 10$	9.6	0	0.0474
#4	S	A	B1	G	$E = 1000, \eta = 0.1$	9.6	0	0.0487
#5	S	A	B1	N	$E_2 = 10$	9.6	0	0.0483
#6	S	A	B1	G+N	$E = 50, \eta = 0.1, E_2 = 10$	9.6	0	0.0483
#7	S	A	C1	G	$E = 1000, \eta = 0.1$	9.3	0	0.0478
#8	S	A	C1	N	$E_2 = 10, R = 0.5$	9.3	0	0.0449
#9	S	A	C1	G+N	$E = 50, \eta = 0.1, E_2 = 10$	9.3	0	0.0476
#10	S	B	A1	G	$E = 1000, \eta = 0.001$	2.4	0	0.0013
#11	S	B	A1	N	$E_2 = 10$	2.4	0	0.0014
#12	S	B	A1	G+N	$E = 200, \eta = 0.001, E_2 = 10$	2.4	0	0.0013
#13	S	B	B1	G	$E = 1000, \eta = 0.001$	2.4	0	0.0015
#14	S	B	B1	N	$E_2 = 10$	2.4	0	0.0015
#15	S	B	B1	G+N	$E = 100, \eta = 0.001, E_2 = 10$	2.4	0	0.0014
#16	S	B	C1	G	$E = 1000, \eta = 0.001$	2.0	0	0.0011
#17	S	B	C1	N	$E_2 = 10$	0.7	1	0.3533
#18	S	B	C1	G+N	$E = 200, \eta = 0.001, E_2 = 10$	2.0	0	0.0011
#19	S	C	A1	G	$E = 1000, \eta = 0.1$	10.2	0	0.2362
#20	S	C	A1	N	$E_2 = 20$	10.2	0	0.2362
#21	S	C	A1	G+N	$E = 200, \eta = 0.1, E_2 = 10$	10.2	0	0.2362
#22	S	C	A2	G	$E = 1000, \eta = 0.1$	9.5	0	0.0488
#23	S	C	A2	N	$E_2 = 20$	9.5	0	0.0488
#24	S	C	A2	G+N	$E = 50, \eta = 0.1, E_2 = 10$	9.5	0	0.0488
#25	S	C	B1	G	$E = 1000, \eta = 0.1$	10.5	0	0.1442
#26	S	C	B1	N	$E_2 = 10, R = 0.5$	10.4	0	0.1440
#27	S	C	B1	G+N	$E = 100, \eta = 0.1, E_2 = 10$	10.5	0	0.1442
#28	S	C	B2	G	$E = 1000, \eta = 0.1$	8.7	0	0.1628
#29	S	C	B2	N	$E_2 = 20, R = 0.5$	9.8	0	0.1576
#30	S	C	B2	G+N	$E = 200, \eta = 0.1, E_2 = 10$	8.8	0	0.1623
#31	S	C	C1	G	$E = 1000, \eta = 0.1$	10.0	0	0.0714
#32	S	C	C1	N	$E_2 = 10$	4.0	1	0.3665
#33	S	C	C1	G+N	$E = 50, \eta = 0.1, E_2 = 10$	10.0	0	0.0714
#34	S	C	C2	G	$E = 1000, \eta = 0.1$	9.3	0	0.0464
#35	S	C	C2	N	$E_2 = 10, R = 0.5$	4.0	1	0.3117
#36	S	C	C2	G+N	$E = 50, \eta = 0.1, E_2 = 10$	9.3	0	0.0464

Table 1: Results of SMAX training on the TOY dataset (with squared loss).

variant	loss	h	smax	method	parameters	$\|\mathbf{W}\|$	$\mathcal{L}_{01}$	$\mathcal{L}$
#37	L	A	A1	G	$E = 1000, \eta = 0.05$	17.5	0	0.1652
#38	L	A	A1	N	$E_2 = 10$	17.5	0	0.1620
#39	L	A	A1	G+N	$E = 200, \eta = 0.05, E_2 = 10$	17.5	0	0.1620
#40	L	A	B1	G	$E = 1000, \eta = 0.05$	17.5	0	0.1662
#41	L	A	B1	N	$E_2 = 10$	17.5	0	0.1592
#42	L	A	B1	G+N	$E = 50, \eta = 0.05, E_2 = 10$	17.5	0	0.1639
#43	L	A	C1	G	$E = 1000, \eta = 0.05$	17.2	0	0.1643
#44	L	A	C1	N	$E_2 = 10$	17.2	0	0.1508
#45	L	A	C1	G+N	$E = 50, \eta = 0.05, E_2 = 10$	17.2	0	0.1638
#46	L	C	B1	G	$E = 1000, \eta = 0.05$	17.1	0	0.8212
#47	L	C	B1	N	$E_2 = 10, R = 0.5$	11.3	0	1.3688
#48	L	C	B1	G+N	$E = 50, \eta = 0.05, E_2 = 10$	17.1	0	0.8210
#49	L	C	B2	G	$E = 1000, \eta = 0.05$	14.8	0	0.8795
#50	L	C	B2	N	$E_2 = 10, R = 0.5$	7.5	2	3.2423
#51	L	C	B2	G+N	$E = 100, \eta = 0.05, E_2 = 10$	14.8	0	0.8783
#52	L	C	C1	G	$E = 1000, \eta = 0.05$	17.1	0	0.4095
#53	L	C	C1	N	$E_2 = 10, R = 0.5$	8.4	1	1.9603
#54	L	C	C1	G+N	$E = 50, \eta = 0.05, E_2 = 10$	17.1	0	0.4094
#55	L	C	C2	G	$E = 1000, \eta = 0.05$	17.2	0	0.1583
#56	L	C	C2	N	$E_2 = 10, R = 0.5$	8.4	1	1.9607
#57	L	C	C2	G+N	$E = 50, \eta = 0.05, E_2 = 10$	17.3	0	0.1582

Table 2: Results of SMAX training on the TOY dataset (with logistic loss).

lowest $\mathcal{L}$ value was achieved by $\mathrm{smax} = \mathrm{smax}_{C1}$.

- **LC** ($\mathrm{loss} = \mathrm{loss}_L$, $h = h_C$): In this category the pure N method performed poorly in terms of misclassifications. The $\mathcal{L}$ value was typically higher than in LA. The lowest $\mathcal{L}$ value was achieved by $\mathrm{smax} = \mathrm{smax}_{C2}$.

5.1.2. Comparing SMAX with other methods

In the next experiments we will compare the proposed SMAX approach with other classification algorithms. The algorithms were tested on the given problems using 10-fold cross validation. This means that each dataset was partitioned randomly into 10 parts. Then, each algorithm was run on each problem 10 times, so that in the i-th run the i-th part was used as the test set, and the other parts as the training set.

The performance measures used in the experiments were the following:

- **AUC**: The area under the receiver operating characteristics [9]. Given a predictor g and a test dataset $(\mathbf{x}_1,\mathbf{y}_1),\ldots,(\mathbf{x}_m,y_m)$ the AUC value can be obtained as follows: At first, the values $g(x_1),\ldots,g(x_m)$ are calculated, and sorted in descending order. If we denote the indices of the sorted sequence by $(1),\ldots,(m)$, and introduce the notations

$\mathrm{TP}_0 = 0, \mathrm{FP}_0 = 0, \mathrm{TP}_k = \sum_{i=1}^{k} y_{(i)} / \sum_{i=1}^{m} y_i$, and $\mathrm{FP}_k = \sum_{i=k+1}^{m} y_{(i)} / \sum_{i=1}^{m} (1 - y_i)$ $(k = 1 \ldots, m)$, then the AUC value can be written as:

$$\mathrm{AUC} = \sum_{k=1}^{m} (\mathrm{FP}_k - \mathrm{FP}_{k-1})(\mathrm{TP}_k + \mathrm{TP}_{k-1})/2.$$

In each experiment, we measure the empirical mean and standard deviation of the AUC values obtained from the 10 runs.

- **ΔAUC**: The difference of the AUC value from the value from the AUC of FDA, which is treated as the baseline result. Again, we measure the empirical mean and the standard deviation of the ΔAUC values obtained from the 10 runs. Note that the standard deviation of AUC and ΔAUC provide different information about the uncertainty of the measurement. The second value is typically smaller than the first, and it is more useful for comparing algorithms.
- **TTIME**: Training time in seconds, summed over the 10 runs.
- **VTIME**: Validation time in seconds, summed over the 10 runs.

The results of classification algorithms on the V2 dataset can be seen in Table 3. Not surprisingly, nonlinear methods outperformed linear ones on this problem in terms of AUC. According to the (mean) AUC value, SMAX was the third best algorithm. According to $\Delta\mathrm{AUC}/\mathrm{VTIME}$, it was the best one.

Results on the V3 dataset can be seen in Table 4. The accuracy of the methods is generally lower than in the case of V2 in terms of AUC. This is because now the optimal decision surface is more complex, and the input space is less densely filled with training examples as in previous case. Again, according to the AUC value, SMAX was the third best algorithm, and according to $\Delta\mathrm{AUC}/\mathrm{VTIME}$, it was the best one.

Results for the ABALONE dataset can be seen in Table 5. Although the highest AUC values were achieved by nonlinear methods, the accuracy of linear methods was relatively good. Nevertheless, SMAX was still the best algorithm in terms of $\Delta\mathrm{AUC}/\mathrm{VTIME}$, slightly outperforming SPER. According to the AUC value, SMAX was the third best method. The other convex polyhedron method, MR performed weak on this problem.

Results for the BLOOD dataset can be seen in Table 6. The best accuracies achieved by linear and nonlinear methods were close to each other. Some nonlinear methods (including MR) performed weak. According to the AUC value, SMAX was the third best algorithm. According to $\Delta\mathrm{AUC}/\mathrm{VTIME}$, it was the second best one (beaten by MLP, tied with LOGR).

Results for the CHESS dataset can be seen in Table 7. We can observe that ID3 and KNN show outstanding accuracy. This interesting phenomenon can be explained by the characteristics of the chess endgames domain. Recall that the inputs are 6 integers,

Method	Parameters	AUC	ΔAUC	TTIME	VTIME
FDA		0.877 (±0.017)	+0.000 (±0.000)	0.08	0.006
LOGR	$E = 1$	0.877 (±0.017)	+0.000 (±0.000)	0.45	0.006
SPER	$E = 1$	0.877 (±0.017)	+0.000 (±0.000)	0.47	0.006
ALN		0.877 (±0.017)	+0.000 (±0.000)	0.43	0.006
LSVM	$C = 10$	0.877 (±0.017)	+0.000 (±0.000)	79.0	0.006
KNN	$K = 35$	0.930 (±0.015)	+0.053 (±0.010)	0.00	27.6
ID3	$\beta = 1, \mathcal{G}_{\min} = 0.001$	0.929 (±0.014)	+0.052 (±0.009)	31.6	0.18
MLP	$R = 0.2, E = 200,$ $\eta = 0.0005, \mu = 0.9$	0.923 (±0.014)	+0.046 (±0.005)	120	0.07
SVM	$C = 10$	0.927 (±0.013)	+0.050 (±0.010)	61.5	3.28
MR	$K = 6, \beta = 0.2$	0.920 (±0.014)	+0.043 (±0.006)	0.31	0.010
SMAX	$K = 2, R = 0.1,$ $E = 500, E_2 = 0,$ $\eta = 0.005$	0.925 (±0.013)	+0.048 (±0.007)	54.6	0.009

Table 3: Results of classification algorithms on the V2 dataset.

Method	Parameters	AUC	ΔAUC	TTIME	VTIME
FDA		0.846 (±0.017)	+0.000 (±0.000)	0.08	0.006
LOGR	$E = 1$	0.846 (±0.017)	+0.000 (±0.000)	0.45	0.006
SPER	$E = 1$	0.846 (±0.017)	+0.000 (±0.000)	0.47	0.006
ALN		0.846 (±0.017)	+0.000 (±0.000)	0.43	0.006
LSVM	$C = 10$	0.846 (±0.018)	+0.000 (±0.000)	60.7	0.006
KNN	$K = 35$	0.887 (±0.020)	+0.041 (±0.010)	0.00	27.5
ID3	$\beta = 1, \mathcal{G}_{\min} = 0.001$	0.877 (±0.019)	+0.031 (±0.009)	22.1	0.19
MLP	$R = 0.2, E = 200$ $\eta = 0.0005, \mu = 0.9$	0.877 (±0.016)	+0.031 (±0.005)	120	0.07
SVM	$C = 10$	0.888 (±0.015)	+0.041 (±0.007)	125	3.17
MR	$K = 12, \beta = 0.2$	0.867 (±0.017)	+0.021 (±0.006)	0.47	0.012
SMAX	$K = 6, R = 0.1,$ $E = 500, E_2 = 0,$ $\eta = 0.005$	0.884 (±0.017)	+0.038 (±0.007)	105	0.011

Table 4: Results of classification algorithms on the V3 dataset.

Method	Parameters	AUC	ΔAUC	TTIME	VTIME
FDA		0.844 (±0.018)	+0.000 (±0.000)	0.067	0.004
LOGR	$E = 1$	0.849 (±0.018)	+0.006 (±0.002)	0.206	0.004
SPER	$E = 2$	0.852 (±0.018)	+0.009 (±0.005)	0.256	0.004
ALN		0.849 (±0.018)	+0.006 (±0.002)	0.195	0.004
LSVM	$C = 10$	0.850 (±0.021)	+0.007 (±0.005)	16.47	0.004
KNN	$K = 25$	0.847 (±0.017)	+0.003 (±0.007)	0.000	7.037
ID3	$\beta = 2, \mathcal{G}_{\min} = 0.001$	0.821 (±0.024)	−0.023 (±0.011)	164.2	0.153
MLP	$R = 0.2, E = 500$ $\eta = 0.002, \mu = 0.95$	0.866 (±0.017)	+0.022 (±0.006)	138.1	0.031
SVM	$C = 1000$	0.863 (±0.015)	+0.019 (±0.007)	30.23	1.517
MR	$K = 6, \beta = 0.2$	0.748 (±0.030)	−0.095 (±0.020)	0.481	0.007
SMAX	$K = 5, R = 0.2,$ $E = 5000, E_2 = 5,$ $\eta = 0.01$	0.855 (±0.019)	+0.012 (±0.008)	430.1	0.007

Table 5: Results of classification algorithms on the ABALONE dataset.

Method	Parameters	AUC	ΔAUC	TTIME	VTIME
FDA		0.754 (±0.043)	+0.000 (±0.000)	0.009	0.002
LOGR	$E = 2$	0.755 (±0.044)	+0.001 (±0.007)	0.039	0.002
SPER	$E = 4$	0.754 (±0.043)	+0.000 (±0.008)	0.055	0.002
ALN		0.753 (±0.041)	−0.002 (±0.010)	0.034	0.002
LSVM	$C = 0.1$	0.745 (±0.048)	−0.009 (±0.020)	1.738	0.002
KNN	$K = 35$	0.759 (±0.053)	+0.004 (±0.047)	0.000	0.157
ID3	$\beta = 5, \mathcal{G}_{\min} = 0.005$	0.732 (±0.056)	−0.022 (±0.038)	1.538	0.010
MLP	$R = 0.5, E = 2000$ $\eta = 0.01, \mu = 0.95$	0.768 (±0.053)	+0.013 (±0.024)	93.35	0.008
SVM	$C = 2000$	0.744 (±0.053)	−0.010 (±0.035)	17.74	0.067
MR	$K = 4, \beta = 0.1$	0.711 (±0.067)	−0.043 (±0.052)	0.054	0.003
SMAX	$K = 6, R = 0.2,$ $E = 500, E_2 = 5,$ $\eta = 0.0001$	0.757 (±0.040)	+0.002 (±0.009)	55.64	0.004

Table 6: Results of classification algorithms on the BLOOD dataset.

Method	Parameters	AUC	ΔAUC	TTIME	VTIME
FDA		0.853 (±0.005)	+0.000 (±0.000)	0.353	0.013
LOGR	$E = 5$	0.854 (±0.005)	+0.001 (±0.002)	1.949	0.013
SPER	$E = 6$	0.854 (±0.005)	+0.001 (±0.001)	2.711	0.013
ALN		0.832 (±0.006)	−0.020 (±0.004)	1.282	0.013
LSVM	$C = 0.001$	0.651 (±0.123)	−0.201 (±0.120)	106.9	0.013
KNN	$K = 15$	0.982 (±0.002)	+0.129 (±0.005)	0.000	279.5
ID3	$\beta = 0.1$, $\mathcal{G}_{\min} = 0.001$	0.993 (±0.003)	+0.140 (±0.006)	192.9	0.700
MLP	$R = 0.01$, $E = 500$ $\eta = 0.0002$, $\mu = 0$	0.836 (±0.006)	−0.017 (±0.004)	916.1	0.178
SVM	$C = 100$	0.955 (±0.006)	+0.102 (±0.007)	277.5	18.76
MR	$K = 6$, $\beta = 0.2$	0.916 (±0.008)	+0.063 (±0.008)	0.783	0.026
SMAX	$K = 6$, $R = 0.2$, $E = 1000$, $E_2 = 5$, $\eta = 0.0002$	0.937 (±0.006)	+0.085 (±0.009)	2231	0.026

Table 7: Results of classification algorithms on the CHESS dataset.

representing the coordinates of the pieces, and the task is to decide if black can avoid being mated in 14 moves. All of the given methods except ID3 and KNN base their model upon the linear combination(s) of the features. In chess, this information is not very useful, because the position value is a highly nonlinear function of the coordinates of the pieces (e.g. the relation is not monotonic). If we analyze the performance of SMAX, then we can see that it was the fourth best method in terms of AUC, and it was the best method in terms of $\Delta\mathrm{AUC}/\mathrm{VTIME}$.

Results for the SEGMENT dataset are shown in Table 8. We can see that linear methods were strongly outperformed by nonlinear ones in terms of accuracy on this problem. The only nonlinear method that performed poorly was MLP. According to the AUC value, SMAX was the best algorithm, tied with SVM. According to $\Delta\mathrm{AUC}/\mathrm{VTIME}$, it was the sole best.

Summarizing the results of the experiments, we can say that SMAX proved to be a useful classification algorithm. Typically, it was less accurate than sophisticated nonlinear methods but more accurate than linear methods. Compared to MR, the other convex polyhedron algorithm, SMAX was more accurate in all of the 6 test problems. If take both accuracy and classification speed into account, then SMAX performed particularly well.

A disadvantage of SMAX on the given problems was relatively long training time (however

Method	Parameters	AUC	ΔAUC	TTIME	VTIME
FDA		0.930 (±0.019)	+0.000 (±0.000)	0.077	0.003
LOGR	$E = 10$	0.945 (±0.017)	+0.015 (±0.009)	0.420	0.003
SPER	$E = 10$	0.942 (±0.020)	+0.012 (±0.011)	0.448	0.003
ALN		0.931 (±0.024)	+0.001 (±0.011)	0.127	0.003
LSVM	$C = 10$	0.939 (±0.024)	+0.009 (±0.014)	5.519	0.003
KNN	$K = 15$	0.988 (±0.009)	+0.058 (±0.019)	0.000	2.748
ID3	$\beta = 0.01, \mathcal{G}_{\min} = 0.001$	0.987 (±0.005)	+0.057 (±0.018)	39.09	0.045
MLP	$R = 0.01, E = 500$ $\eta = 5 \cdot 10^{-6}, \mu = 0.95$	0.858 (±0.026)	−0.072 (±0.025)	76.75	0.018
SVM	$C = 5 \cdot 10^5$	0.989 (±0.010)	+0.058 (±0.019)	84.82	0.260
MR	$K = 8, \beta = 0.2$	0.973 (±0.013)	+0.042 (±0.017)	0.342	0.006
SMAX	$K = 6, R = 0.05,$ $E = 500, E_2 = 5,$ $\eta = 0.008$	0.989 (±0.010)	+0.059 (±0.016)	195.6	0.006

Table 8: Results of classification algorithms on the SEGMENT dataset.

it was still acceptable). I emphasize that the complexity of gradient method based SMAX training is $O(EndK)$, therefore the approach is able to deal with very large problems (as it will be demonstrated in the collaborative filtering experiments).

5.1.3. Notes on running times

Because the implementation environment was Python + NumPy, the measured running times not always reflect the true time requirements of the algorithms. The reason why such phenomena can occur is that Python is a relatively slow, interpreted language, while NumPy is a highly optimized library of numerical routines.

In most cases (FDA, LOGR, SPER, ALN, KNN, MLP, MR, SMAX with gradient training), it was possible to translate every important step of the algorithm to linear algebra operations supported by NumPy, and therefore the overhead of using an interpreted language was small.

In other cases (ID3, SMAX with Newton training), there were critical parts written in pure Python, which resulted a significantly increased running time. These algorithms could be speeded up greatly (up to a constant factor only of course), if we implemented them in C/C++.

In the case of the support vector machines (LSVM, SVM), Python was used only as a wrapper. Most of the computation was done by the highly optimized libsvm library, therefore, the measured running times can be considered as "state of the art".

5.2. Collaborative filtering

5.2.1. The NETFLIX dataset

This collaborative filtering dataset is currrently one of the largest publicly available machine learning datasets. It contains about 100 million rating from over 480 thousand users on nearly 18 thousand items (movies). The dataset was provided generously by Netflix, the popular movie rental service, for the Netflix Prize (NP) competition [5].

The examples are (u,i,r,d) quadruplets, representing that user u rated item i as r on date d. The ratings values are integers from 1 to 5, where 1 is the worst, and 5 is the best. The data were collected between October, 1998 and December, 2005 and reflect the distribution of all ratings received by Netflix during this period.

The collected data was released in a train–test setting in the following manner: Netflix selected a random subset of users from their entire customer base with at least 20 ratings in the given period. A Hold-out set was created from the 9 most recent ratings of the users, consisting of about 4.2 million ratings. The remaining data formed the Training set. The ratings of the Hold-out set were split randomly with equal probability into three subsets of equal size: Quiz, Test and Probe. The Probe set was added to the Training set and was released with ratings. The ratings of the Quiz and Test sets were withheld as a Qualifying set to evaluate competitors. The Quiz/Test split of the qualifying set is unknown to the public. I remark that the date based partition of the entire NP dataset into train–test sets reflects the original aim of recommender systems, which is the prediction of future interest of users from their past ratings/activities.

As the aim of the competition is to improve the prediction accuracy of user ratings, Netflix adopted RMSE (root mean squared error) as evaluation measure. The goal of the competition is to reduce the RMSE on the Test set by at least 10 percent, relative to the RMSE achieved by Netflix's own system Cinematch.[5] The contestants have to submit predictions for the Qualifying set. The organizers return the RMSE of the submissions on the Quiz set, which is also reported on a public leaderboard.[6] Note that the RMSE on the Test set is withheld by Netflix.

[5]The first team achieving the 10 percent improvement is promised to be awarded by a Grand Prize of $1 million by Netflix. Not surprisingly, this prospective award drawn much interest towards the competition. So far, more than 3 000 teams submitted entries for the competition.

[6]`http://www.netflixprize.com/leaderboard`

There are some interesting characteristics of the data and the set-up of the competition that pose a difficult challenge for prediction:

- The distribution over the time of the ratings of the Hold-out set is quite different from the Training set. As a consequence of the selection method, the Hold-out set does not reflect the skewness of the movie-per-user, observed in the much larger Training set. Therefore the Qualifying set contains approximately equal number of queries for often and rarely rating users.
- The designated aim of the release of the Probe set is to facilitate unbiased estimation of RMSE for the Quiz/Test sets despite of the different distributions of the Training and the Hold-out sets. In addition, it permits off-line comparison of predictors before submission.
- We already mentioned that users' activity at rating is skewed. To put this into numbers, ten percent of users rated 16 or fewer movies and one quarter rated 36 or fewer. The median is 93. Some very active users rated more than 10,000 movies. A similar biased property can be observed for movies: The most-rated movie, *Miss Congeniality* was rated by almost every second user, but a quarter of titles were rated fewer than 190 times, and a handful were rated fewer than 10 times [3].
- The variance of movie ratings is also very different. Some movies are rated approximately equally by the user base (typically well), and some partition the users. The latter ones may be more informative in predicting the taste of individual users.

5.2.2. Comparing $\mathrm{SMAX}_{\mathrm{CF}}$ with other methods

The algorithms involved in the experiments were the following:

- **DC**: Double centering [4]. The only parameter of the algorithm is the number of epochs E (default value: 2).
- **BRISMF**: Biased regularized incremental simultaneous matrix factorization [21]. The parameters of the algorithm are the number of epochs E, the number of factors L, the user learning rate η_U (default value: 0.016), the item learning rate η_I (default value: 0.005), the user regularization coefficient λ_U (default value: 0.015), and the item regularization coefficient λ_I (default value: 0.015).
- **NSVD1**: Item neighbor based approach with factorized similarity (also known as Paterek's NSVD1) [16]s. The parameters of the algorithm are the number of epochs E, the number of factors L, the user learning rate η_U (default value: 0.005), the item learning rate η_I (default value: 0.005), the user regularization coefficient λ_U (default value: 0.015), and the item regularization coefficient λ_I (default value: 0.015).
- $\mathbf{SMAX}_{\mathrm{CF}}$: The proposed smooth maximum based convex polyhedron approach (see page 43). The parameters of the algorithm are the smooth max function (default value: smax_{A1}), the smoothness parameter α (default value: 2), the smoothness change

parameters A_1 and A_0 (default value: $A_1 = 1$, $A_0 = 0.25$), the number of epochs E, the number of factors L, the user learning rate η_U (default value: 0.016), the item learning rate η_I (default value: 0.005), the user regularization coefficient λ_U (default value: 0.015), and the item regularization coefficient λ_I (default value: 0.015).

All algorithms were implemented in C++ from scratch. The hardware environment was a server PC with Intel Pentium Q9300 2.5 GHz CPU and 3 Gb memory.

Let us denote the NETFLIX Training set by $\mathcal{T} = \{(u_1,i_1,r_1,d_1),\ldots,(u_n,i_n,r_n,d_n)\}$, and the Probe set by $\mathcal{P} = \{(u_1,i_1,r_1,d_1),\ldots,(u_m,i_m,r_m,d_m)\}$. The exact sizes of the sets are $n = 100{,}480{,}507$ and $m = 1{,}408{,}395$. All algorithms were trained using $\mathcal{T} \setminus \mathcal{P}$, and then the Probe RMSE of the trained predictor g was calculated as

$$\text{Probe RMSE} = \sqrt{\frac{1}{|\mathcal{P}|}\sum_{(u,i,r,d)\in\mathcal{P}} (g(u,i) - r)^2}.$$

The results of individual algorithms are shown in Table 9. Recall that SMAX_{CF} can be considered as a generalization of BRISMF. We can see, that the SMAX_{CF} approach was able to boost the accuracy of BRISMF, however if we used more factors, then the benefit was smaller. The NSVD1 approach was less accurate than than BRISMF and SMAX_{CF}, and not surprisingly, DC was the worst in terms of RMSE. It is true for all of BRISMF, NSVD1, and SMAX_{CF} that the accuracy was increasing with introducing more factors.

Each experiment consists of three main phases: data loading, training, and validation. The last column of the table shows the total running time of the experiments in seconds. If we take into account that more than 99 million examples were used for training, then we can conclude that all of the presented algorithms are efficient in terms of time requirement.

In the last experiments the predictions of the previous methods for the Probe set were blended with L2 regularized linear regression. The value of the regularization coefficient was $\lambda = 1.4$. The results can be seen in Table 10.

The last column shows the 10-fold cross validation Probe RMSE of the optimal linear combination of the inputs. The reason why the single-input blends (#11, #12, and #13) have lower RMSE than the inputs themselves is that the linear blender introduces a bias term too. We can see that the SMAX_{CF} approach was able to improve the result of the combination of BRISMF and NSVD1 models. This indicates that SMAX_{CF} was able to capture new aspects of the data that was not captured by BRISMF and NSVD1.

No.	Method	Parameters	Probe RMSE	Running time (seconds)
#1	DC		0.9868	11
#2	BRISMF	$L = 10$, $E = 13$	0.9190	161
#3	BRISMF	$L = 20$, $E = 12$	0.9125	263
#4	BRISMF	$L = 50$, $E = 12$	0.9081	598
#5	NSVD1	$L = 10$, $E = 26$	0.9492	568
#6	NSVD1	$L = 20$, $E = 24$	0.9459	1057
#7	NSVD1	$L = 50$, $E = 22$	0.9435	1900
#8	SMAX_{CF}	$L = 10$, $E = 18$	0.9169	861
#9	SMAX_{CF}	$L = 20$, $E = 18$	0.9114	1234
#10	SMAX_{CF}	$L = 50$, $E = 18$	0.9079	2692

Table 9: Results of collaborative filtering algorithms on the NETFLIX dataset.

No.	Inputs	Probe RMSE
#11	#4	0.9069
#12	#7	0.9430
#13	#10	0.9069
#14	#2+#3+#4	0.9065
#15	#5+#6+#7	0.9429
#16	#8+#9+#10	0.9068
#17	#14+#15	0.9035
#18	#14+#16	0.9050
#19	#15+#16	0.9033
#20	#14+#15+#16	0.9021

Table 10: Results of linear blending on the NETFLIX dataset.

6. Conclusion

Convex polyhedron classifiers are special binary classifiers that fit well to unbalanced problems, because they tend to classify negative examples quickly. Despite this appealing property, the approach is not frequently used in practice. The main reason for that is the lack of good training algorithms.

In this paper I proposed novel and computationally efficient algorithms for training convex polyhedron classifiers. The proposed algorithms are based on the smooth approximation of the maximum function. I also introduced the analogous variant of the smooth maximum approach for regression and collaborative filtering.

The usefulness of the proposed methods was demonstrated via experiments on artificial and real datasets. It turned out turned out that smooth maximum based classifiers are able to provide a good tradeoff between accuracy and classification time on unbalanced classification problems. In the case of collaborative filtering, smooth maximum based methods are able to complement other methods well.

References

[1] D. Ascher, P. F. Dubois, K. Hinsen, J. Hugunin, and T. Oliphant. Numerical Python, 2001. URL: http://www.numpy.org/.

[2] A. Asuncion and D. J. Newman. UCI Machine Learning Repository, 2007. URL: http://www.ics.uci.edu/∼mlearn/MLRepository.html.

[3] R. Bell, Y. Koren, and C. Volinsky. Chasing $1,000,000: How we won the Netflix Progress Prize. *ASA Statistical and Computing Graphics Newsletter*, 18(2):4–12, 2007.

[4] R. M. Bell and Y. Koren. Improved neighborhood-based collaborative filtering. In *Proc. of the KDD Cup and Workshop 2007*, pages 7–14, 2007.

[5] J. Bennett and S. Lanning. The Netflix Prize. In *Proc. of the KDD Cup and Workshop 2007*, pages 3–6, 2007.

[6] B. Boser, I. Guyon, and V. Vapnik. A training algorithm for optimal margin classifiers. In *Proc. of the Fifth Annual Workshop on Computational Learning Theory*, pages 144–152, 1992.

[7] C. Chang and C. Lin. *LIBSVM: a library for support vector machines*. National Taiwan University, 2001. URL: http://www.csie.ntu.edu.tw/∼cjlin/libsvm.

[8] K. Crammer and Y. Singer. On the algorithmic implementation of multiclass kernel-based vector machines. *Journal of Machine Learning Research*, 2:265–292, 2001.

[9] J. P. Egan. *Signal detection theory and ROC analysis*. Academic Press, New York, 1975.

[10] M. Elad, Y. Hel-Or, and R. Keshet. Pattern detection using a maximal rejection classifier. In *Proc. of the 4th International Workshop on Visual Form*, pages 514–524, 2001.

[11] P. Fischer. More or less efficient agnostic learning of convex polygons. In *Proc. of the 8th Annual Conference on Computational Learning Theory*, pages 228–236. ACM Press, New York, 1995.

[12] R. A. Fisher. The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7:179–188, 1936.

[13] E. Fix and J. L. Hodges. Discriminatory analysis: Non-parametric discrimination: Consistency properties. Technical Report 4, US Air Force School of Aviation Medicine, 1951.

[14] A. R. Klivans, R. O'Donnell, and R. A. Servedio. Learning intersections and thresholds of halfspaces. *Journal of Computer and System Sciences*, 68(4):804–840, 2004.

[15] S. Kwek and L. Pitt. PAC learning intersections of halfspaces with membership quieries. *Algorithmica*, 22:53–75, 1998.

[16] A. Paterek. Improving regularized singular value decomposition for collaborative filtering. In *Proc. of the KDD Cup and Workshop 2007*, pages 39–42, 2007.

[17] I. Pilászy and T. Dobrowiecki. Constructing large margin polytope classifiers with a multiclass classification algorithm. In *Proc. of the 4th IEEE Workshop on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS'2007)*, pages 261–264, 2007.

[18] J. R. Quinlan. Induction of decision trees. *Machine Learning*, 1(1), 1986.

[19] F. Rosenblatt. *Principles of neurodynamics: Perceptrons and the theory of brain mechanisms*. Spartan Books, Washington D.C., 1962.

[20] G. van Rossum. An introduction to Python, 2006.
URL: http://www.network-theory.co.uk/docs/pytut/.

[21] G. Takács, I. Pilászy, B. Németh, and D. Tikk. Scalable collaborative filtering approaches for large recommender systems. *Journal of Machine Learning Research (Special topic on Mining and Learning with Graphs and Relations)*, 10:623–656, 2009.

[22] L. G. Valiant. A theory of the learnable. *Communications of the ACM*, 27(11): 1134–1142, 1984.

[23] S. Vempala. A random sampling based algorithm for learning the intersection of half-spaces. In *Proc. of the 38th Annual Symposium on Foundations of Computer Science*, pages 508–513, 1997.

[24] P. J. Werbos. *Beyond regression: New tools for prediction and analysis in the behaviour sciences*. PhD thesis, Harvard University, Cambridge, MA, 1974.

[25] B. Widrow. An adaptive "adaline" neuron using chemical "memistors". Technical Report 1553-2, Stanford Electronics Laboratories, 1960.

[26] E. B. Wilson and J. Worcester. The determination of L.D.50 and its sampling error in bio-assay. *Proceedings of the National Academy of Sciences*, 29:79–85, 1943.

A Simultaneous Solution for General Linear Equations on a Ring or Hierarchical Cluster

G. Molnárka, N. Varjasi

"Széchenyi István" University Győr, Hungary, H-9026 Egyetem tér 1.
Phone: +3696503400, fax:
e-mail: varjasin@sze.hu

Abstract: There are several iterative models for solving general, large linear equations. In this paper a parallel algorithm with slow convergence speed for studying the speed-up effect of the parallel algorithms has been presented. The difference between the ring and hierarchical topology considering running speed and efficiency has also been explored. Detailed numerical test results of the algorithm including the speedup of parallel execution are shown.

Keywords: parallel programming, linear equation, cluster computing

1. The minimal residual algorithm

The main goal of this article is to study the speed up effect on a parallel computer using a parallel algorithm for solving a full rank, general, symmetric, positive definite not sparse (but dense) linear equation with high condition number ($n = 1000$, $cond = 2 \cdot 10^{10}$; $n = 5000$, $cond = 4 \cdot 10^{13}$; $n = 10000$, $cond = 7 \cdot 10^{12}$; $n = 15000$, $cond = 5 \cdot 10^{16}$), where $cond(A) = \|A\| \cdot \|A^{-1}\|$ and Euclidean norm was used.

The condition number shows the difficulty of the linear equation. Higher condition number means the complexity of the problem, in other words the equation gets more and more difficult with numerical methods.

The solving algorithms of the general $Ax = b$ equation are well known [1] [2] [3]. In the case of large systems direct algorithms are inefficient. Only iterative methods can be used that can produce results with the desired precision [4] [10], otherwise the floating point arithmetic causes several rounding errors.

The base of the presented numerical algorithm for the solution of linear systems of equations is a generalisation of the classical one-step iterative algorithm (such as the gradient method). Generalisation will not improve the convergence speed of the algorithm but it highly improves parallel execution. In a sequential case the base algorithm [5] [6] has slow convergence speed. The minimal residual algorithm is a widely known algorithm, but the suggested versions of *Algorithm1* and *Algorithm2* have been created by the authors. The results of the parallel realisation of the algorithms and the measured data are the results of the present research.

The suggested algorithms give a good opportunity to study the effects of parallel processing. If a cluster or a multiprocessor computer is used, one can expect considerable speed-up effect.

2. Methods for parallelisation on homogeneous and heterogeneous systems

The aim of parallel processing is to break a large problem down to several smaller components or calculations that can be solved parallelly with different processors at the same time. The most efficient tools for scientific computations would be massively parallel computers, with a large shared memory, but this hardware is expensive and unattainable for the research team. Other solutions can be distributed systems and cluster computing. A small cluster of 16 PCs with normal network connection and an interconnected cluster machine with 88 processors (HP BladeSystem C3000) were used. On the cluster computing model a message passing software (MPI) was used to solve the tasks.

2.1. Ring and hierarchical topologies

In parallel solutions there are two bottlenecks for optimisation: the communication and the calculations. These aspects have been studied with two topologies. For linear equations based on numerical models the ring model is often used [8]. In this case every node has a connection with the two neighbouring nodes, or other nodes. This solution works efficiently on homogeneous systems. On this model the heartbeat algorithm is useful (see *Figure 1 a*). First it starts an initialization procedure, then a loop starts. In the first phase of the loop a data sending and receiving mechanism process is accomplished (synchronization). This is the data exchange period between each computation node. After that, every node runs the computation algorithm. This is the "cpu" period of the work. The loop runs until the stopping criteria. In this model every node has an equivalent role. This model needs a homogeneous network and the same type of processor because each synchronization step made by the slowest node. Fast nodes need to wait for them.

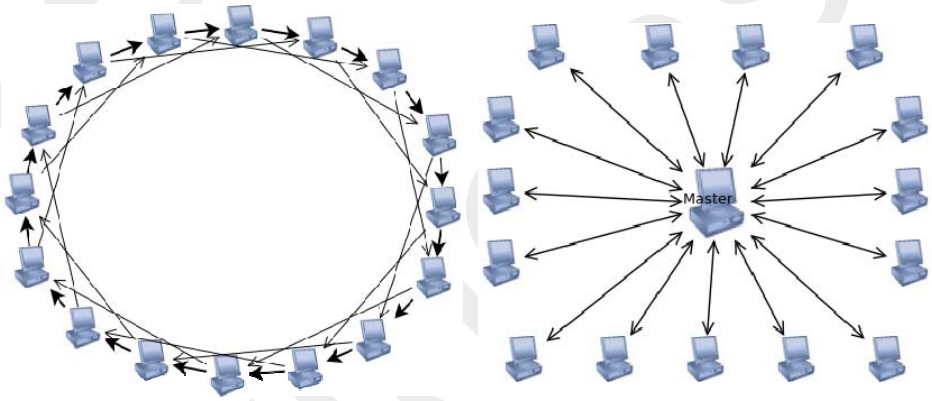


Figure 1. Ring (a) and hierarchical (b) topology

In other cases hierarchical topologies are useful [9]. The master-worker model is based on a distributed and large, heterogeneous cluster (see *Fig 1b.*). The master node controls

the running processes, assigns problems to workers and manages the partial solutions. The role of the worker nodes is to solve smaller parts of the problem. This model works well with asynchronous methods, too because worker nodes have not connected to each other. Every worker node can reach the maximum performance of the processors.

2.2. Algorithm 1

For a ring topology the following algorithm has been used:

I. Let P be the number of nodes, let *eps* be the tolerated error value.

II. Let A be the matrix to be solved and let b be the solution vector.

III. For every node $p \in P$ do in parallel: generate x_1 random vector.

IV. For every node $p \in P$ do in parallel:

Operation() while a result arrives or converges.

V. The result of solution is x_1 on master node.

The algorithm uses the *Operation()* function on every node:

1. do
2. let x_2 be a new random vector
3. let $r_1 = Ax_1 - b$ and $r_2 = Ax_2 - b$, where $r_1 - r_2 \neq 0$
4. let $c_{12} := \dfrac{(r_2 - r_1, r_2)}{\|r_1 - r_2\|^2}$
5. let $x_{12} := c_{12}x_1 + (1 - c_{12})x_2$
6. let $r_{12} := c_{12}r_1 + (1 - c_{12})r_2$
7. let $x_1 := x_{12}$, and $r_1 := r_{12}$
8. if $\|r_1\|^2 < min$ or $iterationnum > n$

then send x_1 to the next node and wait for a new x_2 vector

9. while $\|r_1\|^2 < eps$
10. return x_1, the solution with desired precision.

Remarks:

From the vector exchange it is expected that the given result is better, or when the algorithm reaches a local minimum value this x_1 solution is sent to another node, which continues the computation with a new random number coming from another node (see *Figure 2*).

In the implementation of the algorithm the Mersenne-twister pseudorandom number generator has been applied [7] and the independence of iteration sequences is based on the independent clock of the computing nodes.

Two error measure methods have been used in the algorithm. The first was the general residual error: $err_r = \|r_1\|^2 = \|Ax - b\|^2$. When the exact solution of the test case is known (x'), there is a chance to compute the absolute error value: $err_x = \|x - x'\|^2$, where x is the approximate solution vector.

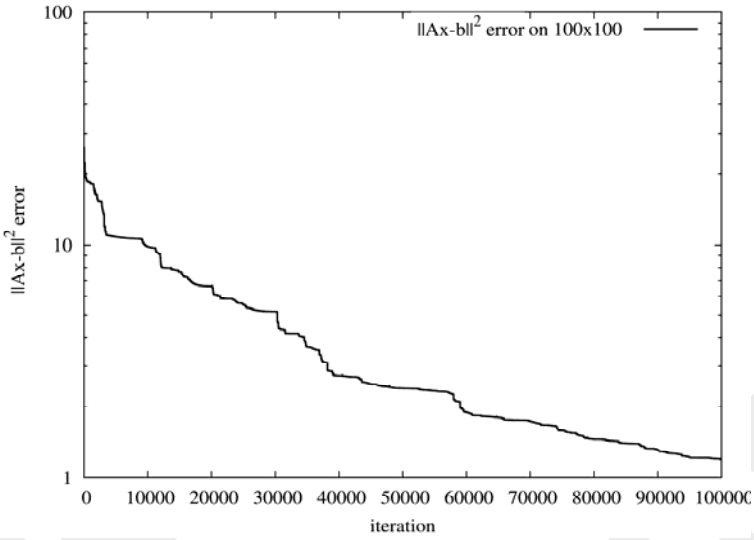


Figure 2. The convergence of Algorithm1 (n = 100)

In the case of large-sized, badly conditioned linear equations these error values are relatively high numbers (with $cond = 10^{16}$, $err_r = 1000$ means a close solution as shown in *Figure 4*, in detail that means $\|x - x'\|^2$ is $\sum_{i=1}^{n} |x_i - x_i'|^2$ and the error for every member of the solution vector is approximately $|x_i - x_i'|^2 \approx 10^{-8}$).

If a problem was solved where the correct solution had already been known, and the residual and absolute error were compared it has to be noted that the absolute error of the solution is always better than the residual.

The efficiency of the algorithm depends on load balancing: the operation can be repeated several times with slow convergence speed or the result vector can be exchanged between nodes to give extra speedup. In this case and referring to Amdahl's law the ring model has a theoretical maximum number of nodes. If more nodes are used and the best solution is sent to the next node, the larger ring will increase running time as the solution waves slowly on to the other nodes.

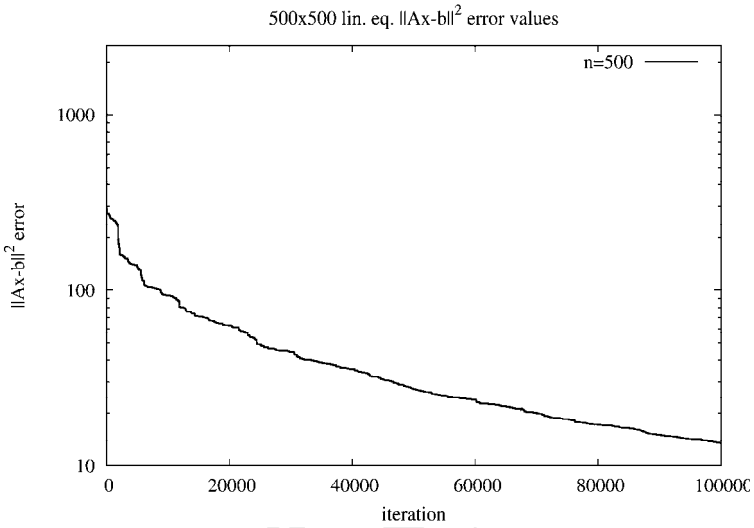


Figure 3. The convergence of Algorithm1 (n = 500)

This *Algorithm1* works well on a homogeneous cluster (see *Figure3*). But if there is a slower or a loaded computer on the ring the send-receive method will be slow and several traffic jams are expected and running times also grow.

Algorithm1 has been revised and a new, hierarchical model has been composed.

2.3. Algorithm 2

I. Let *P* be the number of nodes, let *eps* be the tolerated error value.

II. Let *A* be the matrix to be solved and let *b* be the solution vector.

III. The master node generates a random vector x_2 and sends for every worker node.

IV. For every node $p \in P$ do in parallel: generate a random vector x_1

V. On master node do *Control()* while $\|x_1\|^2 < eps$

VI. For every worker node $p \in P$ do in parallel:

Operation(x1) while a vector arrives

VII. The result of solution is x_1 on master node.

On the worker nodes the *Operation()* function uses a residual approach like *Algorithm1*. The difference is that the operation function sends and receives data from the master node only.

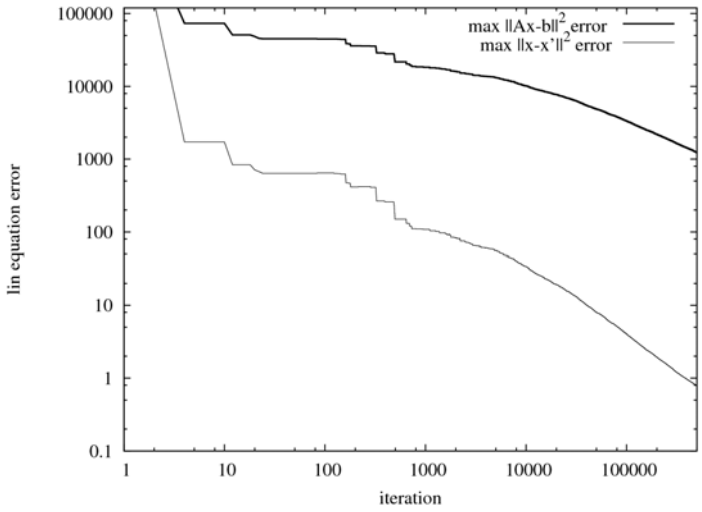


Figure 4. Residual and absolute error of the solution vector (n = 15000, P = 88 cpus, log-log scale)

The *Control* function of the master node distributes and collects data from every worker node. On the master node the problem is not solved, but the result is presented here. The master node controls the data exchanges, and presents the best approximate result vector for every node (see *Figure 4*). This model is flexible because the number of worker nodes number can grow dynamically [8].

This model can be used on heterogeneous clusters, too, because the worker nodes are independent and communicate only with the master node. Every node works on the master's best solution.

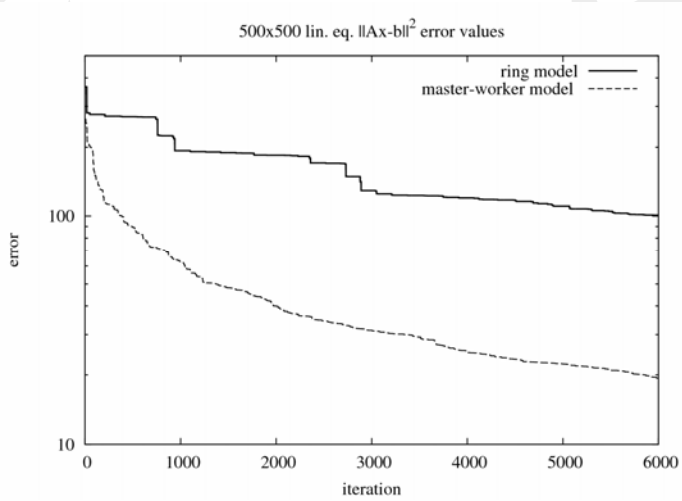


Figure 5. Convergence speed between topologies (n = 500, P = 16 cpus);

The difference between the ring and hierarchical algorithm is that the hierarchical algorithm results in smaller computational time and better convergence. As it can be seen in *Figure 5* the ring solution (solid line) has a minor gradient whereas the hierarchical solution (dotted line) has a steeper gradient. This is because at the ring model the corrective effect of a new solution reaches the previous node in *P–1* steps, while in the hierarchical model the corrective effect is achieved in one step. Moreover, on the ring model we can only take the result of one or two neighbours into consideration.

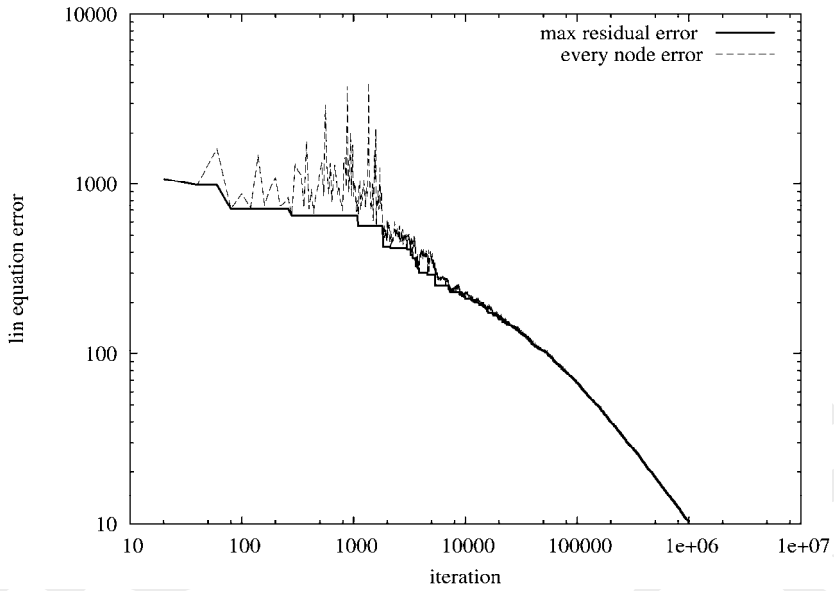


Figure 6. The results of working nodes (dotted) and the minimal residual error (solid line) (n = 10000, P = 88, log-log scale)

The hierarchical model has better convergence features. At every iteration loop the master node sends the best solution vector for the workers. This adds some genetic features to the algorithm [5]. If we examine the details on *Figure 3* and *Figure 4* steps in the curve can be seen. It has to be noted that in a P-processor master-worker model only P-1 processors solve the linear equation.

Let us focus now on this hierarchical solution. If the convergence curves are observed it can be noticed that all of the solutions are similar, and the final solution has always the same order of error in every case. The differences between the exact values of the errors are unfortunately caused by the pseudo-random number generator. In this algorithm only an approximate solution is achieved, not the exact vector.

If the problem is examined from another point of view and the execution time of the algorithm is recorded, the results shown in *Figure 7* are achieved. If more computing nodes are used the running time is expected to decrease.

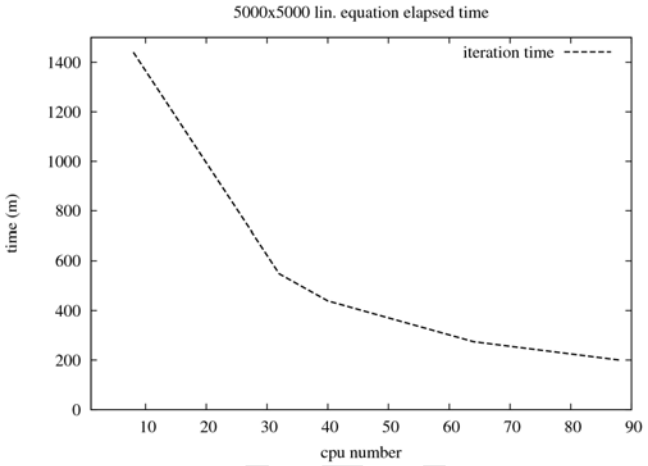


Figure 7. Time of execution and processor numbers (n = 5000);

In these cases only parallel execution provides results in an acceptable time. At some points larger than linear relative speed-up can be achieved, as it is shown in *Fig 8.* But when the number of processors grows, the effective speedup and efficiency decreases as the worker nodes report their own solutions and the load of the master node grows. More precise load balancing can be used, but the size of the problem and the communication delays prevent further advances. For better results another method has to be used.

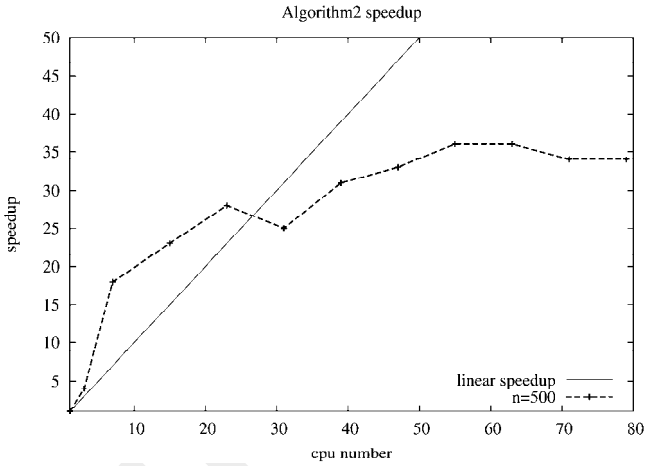


Figure 8. Algortihm2 relative speedup (n = 500)

3. Results

Above a new type of algorithm for the solution of a linear equation system on heterogeneous clusters has been presented. The algorithm is based on a residual minimisation technique with master-worker solutions. The algorithm has some genetic features because the new, better vectors are made from a group of good vectors as seen in *Figure 8* for extra speed-up.

Computer tests have proved the theoretical results; parallel implementation is much better than the sequential one. We get a considerable speed-up effect using a parallel computer.

The goal of creating these algorithms has been basic research but the solution of bad condition linear equations is a useful method for several practical and realistic problems. Optimization, finite element methods, control or simulation problems are often based on large dense linear equations.

We have to note that only the simplest algorithm has been tested. The test with more effective algorithms will be the subject of a forthcoming work.

4. References

[1] Louis A. Hageman, Davis M. Joung: *Applied Iterative Methods*, Computer Science and Applied Mathematics, Academic Press, (1981).

[2] P. G. Ciarlet: *Introduction à l'analyse numérique matricielle et à l'optimisation*, MASSON, Paris, (1982).

[3] G. Golub, A. Greenbaum, M. Luskin, eds., *Recent Advances is Iterative Methods,* The IMA Volumes in Math. and its Applications Vol.60., Springer Verlag, (1994).

[4] G. Molnárka, N. Varjasi: *Parallel algorithm for solution of general linear systems of equations*, Informatika a felsőoktatásban 2005, Debrecen ISBN 963 472 909 6, pp.176.

[5] G. Molnárka: *A scalable parallel algorithm for solving general linear system equations*, 77th GAMM annual meeting 2006, Berlin (2006) pp.441.

[6] N. Varjasi, *Parallel Algorithm for linear equations with different network topologies,* Proceedings of International e-Conference on Computer Science (IeCCS) 2006 in Lecture Series on Computer and Computational Sciences, (2007) pp. 502-505, Brill Academic Publishers, ISBN 978-90-04-15592-3

[7] M. Matsumoto and T. Nishimura, Mersenne Twister: *A 623-dimensionally equidistributed uniform pseudorandom number generator,* ACM Trans. on Modeling and Computer Simulation Vol. 8, No. 1, January (1998) pp. 3-30

[8] A. Basermann, B. Reichel, C. Schelthoff, *Preconditioned CG methods for sparse matrices on massively parallel machines*, in Parallel Computing 23. (1997) pp. 381-398

[9] E. J. H. Yero, M. A. A Henriques, *Speedup and scalability analysis of Master-Slave applications on large heterogeneous clusters,* in Journal of Parallel and Distributed Computing 67. (2007) pp. 1155-1167

[10] I. S. Duff, H. A. van der Vorst, *Developments and trends in the parallel solution of linear systems,* in Parallel Computing 25. (1999) pp. 1931-1970

Calculation of the Numerical Solution of Two-dimensional Helmholtz Equation

G. Hegedűs, M. Kuczmann

Laboratory of Electromagnetic Fields, Department of Telecommunications
"Széchenyi István" University, Egyetem tér 1, H-9026 Győr, Hungary
Pannon University, Deák F. u. 16, H-8360, Keszthely, Hungary
Phone: +3683545372, fax: +3683545373
e-mail: hegedus@georgikon.hu

Abstract: Many physical phenomena in acoustics, optics and electromagnetic wave theory are governed by the scalar wave equation. In the frequency-domain, the wave equation is the so called Helmholtz equation. In many cases, a theoretical numerical solution can be obtained for this equation by using finite differences with Sommerfeld boundary condition, resulting in a system of linear equations to be solved. The Sommerfeld boundary condition is used to solve uniquely the Helmholtz equation. However, in practice great difficulties are caused by the above method's great demand on operative storing capacity and calculation time. In the following contribution, a method for directly solving a linear equation system with a five off-diagonal matrix is presented. We show, that for this method, the number of computational steps and the memory requirement can be significantly reduced, and the possibilities for parallelization are also analyzed.

Keywords: Helmholtz equation, Sommerfeld boundary condition, finite difference method, sparse matrix

1. Introduction

Let $u = u(x, y)$ be the complex valued wave function on the region Ω, satisfying the Helmholtz equation [13][12]

$$\Delta u + k^2 u = 0\,, \tag{1}$$

where

$$\Delta u = \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2}\,, \tag{2}$$

and k is the wave number, $k = \frac{2\pi}{\lambda}$, with λ being the wavelength. Let the shape of Ω be a rectangle, and the wave propagate in the Ω plane. A Sommerfeld boundary condition [1] is applied, i.e.,

$$\frac{\partial u}{\partial n} - iku = 0 \tag{3}$$

on the subset $\Gamma = \partial\Omega \setminus \Gamma'$, where $\partial\Omega$ is the boundary of domain, on the set Γ' the values of u are known. Here $\boldsymbol{n}$ denotes the unit normal vector of $\partial\Omega$, and i means the imaginary unit. Let the examined domain Ω be covered with an equidistant grid of spacing d, centered in a certain grid point with coordinates (x, y). Applying this choice, the discretized wave function is given only in the grid points as $u(x, y) = u(pd, qd) = u_{pq}$ [14], with $0 \le p \le a \quad 0 \le q \le b$, $p, q \in \mathrm{N}$. The discretization scheme is illustrated in Figure 1. The aim of the work is to determine the values of u in these points according to the prescribed boundary conditions.

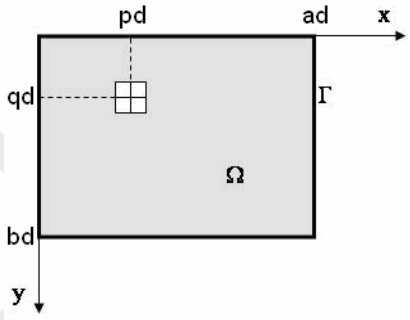


Figure 1. The studied domain Ω with the boundary $\Gamma = \partial\Omega$, and the applied grid centered in the gray point.

The discretization, together with the finite difference approximation, results in a system of linear equations. The system matrix is a large but sparse matrix with complex values. In order to obtain a sufficiently accurate numerical solution, the number of grid points per wavelength should be sufficiently large. As a result, the linear system becomes extremely large.

In this work, a method is presented to generate a solution of the problem. Efforts were made to place as much valuable (non-zero) data into the memory as possible and to apply the fastest possible operations.

The linear equation system describing the studied wave-range is composed of matrices with five non-zero off-diagonals, which can be transformed into matrices containing five valuable lines. This is, however, still too large to be kept in the memory simultaneously. We can achieve further memory size decrease by applying a sliding working-window in which the data transfer is minimized for the optimized operation. Within the work-window a direct procedure was used based on the Gaussian elimination. Further decrement in the necessary storing capacity can be achieved by dividing the domain.

Considering everything that depends on the capacity of the operating memory, we can achieve a good calculation capacity if the wave-range is optimally selected within the memory and the applied variables are ideally organized.

The effectiveness of the presented method is investigated by a numerical example of the beam propagation in a homogeneous medium.

2. The difference equations and the boundary equations

Inside the domain, the studied point is elements of an equidistant grid [11] which can be seen in Figure 2.

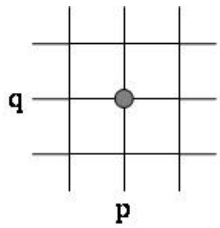


Figure 2. The equidistant grid for finite differences method.

The equation (1) can be approximated by the 5-point difference scheme [3][11]:

$$u_{p-1q} + u_{pq-1} + u_{p+1q} + u_{pq+1} - (4 - k^2h^2)u_{pq} = 0\,. \tag{4}$$

Two types of boundary points can be defined, where the values u are unknown. Figure 3. A) shows the $x = ad$ side points, except for the edges. Calculating with grid points on the side [7][8]:

$$(4 - 2ikh - k^2h^2)u_{pq} - 2u_{p-1q} - u_{pq-1} - u_{pq+1} = 0\,. \tag{5}$$

The same procedure can be followed on the other corners.

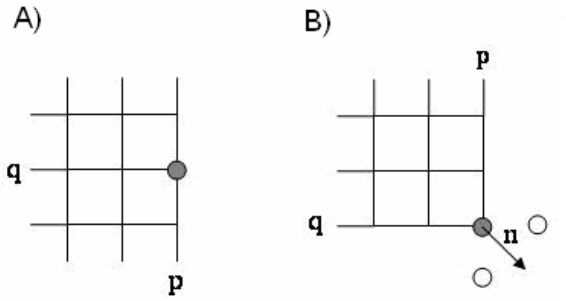


Figure 3. The boundary points of type A) side, B) corner.

Applying the $n=\left(\frac{\sqrt{2}}{2},\frac{\sqrt{2}}{2}\right)$ condition and second-order accuracy Taylor series expansion, on the $x=ad \quad y=bd$ corner of the domain: [4]

$$\frac{\partial u_{pq}}{\partial n}=\left(\frac{\partial u_{pq}}{\partial x},\frac{\partial u_{pq}}{\partial y}\right)n=\frac{\sqrt{2}}{2h}\left(2u_{pq}-u_{p-1q}-u_{pq-1}+\frac{1}{2}\frac{\partial^2 u_{pq}}{\partial x^2}h^2+\frac{1}{2}\frac{\partial^2 u_{pq}}{\partial y^2}h^2\right). \quad (6)$$

Based on (4) it yields on the $x=ad \quad y=bd$ corner :

$$0=\left(2-\sqrt{2}ikh-\frac{1}{2}k^2h^2\right)u_{pq}-u_{p-1q}-u_{pq-1}, \quad (7)$$

which can be applied for the other corners as well.

3. Numerical solution

3.1. Optimal buffering in the operating memory

Applying the conditions (4), (5) and (7), a 5 diagonal homogenous linear set of equations is received containing the equation of $(a+1)(b+1)$ [2]. Figure 4.A) represents the extended matrix of simultaneous equations in the case of $a=b=5$. The black places indicate zero values, while the white ones mean some complex values different from zero.

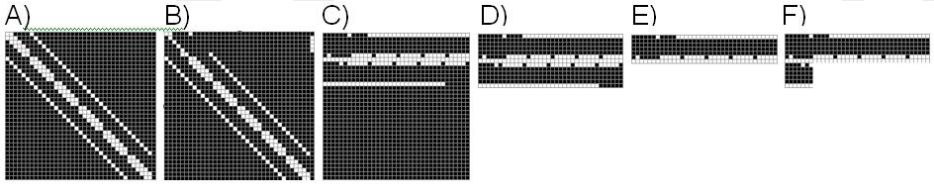


Figure 4. Storage layouts

Considering Dirichlet boundary points, inhomogeneous system of equation is resulted in that is illustrated the Figure 4.B). The Helmholtz equation with special preliminary conditions can be represented by a large system of equations with a sparse matrix. The size of the matrix is too large compared to the stored information [6]. The necessary storing capacity can be significantly reduced in the following way. The last column can be detached, and stored in a separate vector. The valuable diagonal dots of the system with coordinates (x,y) can be transformed into a row formation according to

$$(x,y)\mapsto\left(x-\left[\left(y/d-a-1\right) \mod \left((a+1)(b+1)\right)\right]d,y\right). \quad (8)$$

Figure 4.C) demonstrates the state after the row transformation.

Only the first $2a+3$ rows of the matrix are kept, it is unnecessary to reserve space for the others (Figure 4.D). After this reduction, extra care needs to be taken in order not to step out of the reduced $(2a+3)\times(a+1)(b+1)$ matrix during the elimination of the coefficients. During the Gaussian elimination modified considering (8), the coefficients

under the row $(a+2)$ can be zeroed in increasing column-index (Procedure I), then the coefficients above row $(a+2)$ in decreasing column-index can also be eliminated in the same way.

A further decrease in storing capacity can be obtained if the line number $(a+3)$ and the last row are stored in two separate vectors, and afterwards, rows after line number $(a+2)$ are left out. The size of the resulting matrix is $(a+2)\times(a+1)(b+1)$, which can be seen in Figure 4.E).

For Procedure I, a sliding matrix of size $(a+1)\times(a+2)$ can be used, which is filled by values from separate vectors for an iteration step. This sliding matrix also stores the transitional values of the elimination process under the „transformed main diagonal", as it can be seen in Figure 4.F).

When moving the sliding matrix one step to the right, the new incoming column can be written to the place of the outgoing column, thus there is no need to rewrite the whole matrix. The columns can be referred to with the modulo-index $(a+2)$.

With the first procedure, the range above the row $(a+2)$ gets saturated with transitional values, which is to be eliminated with Procedure II.

3.2. The required storing capacity

It becomes obvious that, basically the distance of the two side-diagonals determines the size of the storage demand according to the above method. On the other hand, this distance depends on the values of the border dimensions *a* and *b* in the studied range.

Further decrement in the necessary storing capacity can be achieved by dividing the domain Ω. The question is, what shape and size is practical for the resulting sub-domains. According to Figure 4.E), in case of 16 byte storage of the complex values, the necessary storage capacity in byte units is

$$S(a,b) = 16\big((a+2)(a+1)(b+1) + (a+2)(a+1) + 2(a+1)(b+1)\big). \tag{9}$$

The area of the domain Ω was

$$A(a,b) = \big((a+1)(b+1)d\big)^2. \tag{10}$$

For a given area *A* the necessary storage capacity can be reduced by decreasing parameter *a* as it can be derived from the following expression

$$S(a,A) = 16\left(\frac{\sqrt{A}}{d}(a+4) + (a+2)(a+1)\right). \tag{11}$$

Unfortunately parameter *a* can not be decreased arbitrarily because of the distortion of the result. According to the above considerations, it is effective to divide the system parallel to axis *y*, thus generating *n* congruent sub-domains [3]. Compared with the case of (9), the simultaneous storage of these data needs less capacity by a factor

$$\mu(a,n)=\frac{n\left(\left(\frac{a}{n}+4\right)\left(\frac{a}{n}+1\right)(b+1)+\left(\frac{a}{n}+2\right)\left(\frac{a}{n}+1\right)\right)}{(a+4)(a+1)(b+1)+(a+2)(a+1)}<\frac{n\left(\frac{a}{n}+4\right)\left(\frac{a}{n}+1\right)}{(a+2)(a+1)}<<1\ . \qquad (12)$$

Although even according to (12) a considerable memory load decrement can be achieved, it is not necessary to store the data of the sub-domains simultaneously, the procession of their data is possible separately.

The memory need of a sub-domain in case of $a=b$ is

$$S(a,n)=16\left(\frac{a}{n}+1\right)\left(\left(\frac{a}{n}\right)^2+6\frac{a}{n}+6\right). \qquad (13)$$

According to (13), the value of *S* increases strongly with the augmentation of *a*, but choosing the right number of sub-domains *n*, it can be divided into computationally manageable sub-problems. This experience can be derived from Figure 5.

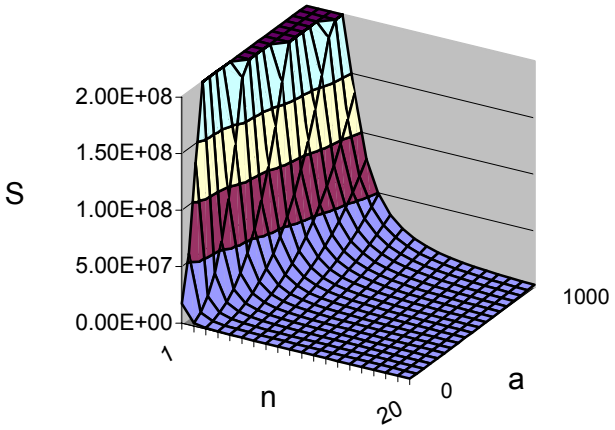


Figure 5. The necessary storing capacity as a function of the linear domain size a and the number of sub-domains n.

Applying the Huygens-Fresnel principle to the elementary domains, these sub-domains shall be treated as the starting objects of waves and the wave propagation between the sub-domains have to be ensured. In order to guarantee the proper wave propagation, domain Ω should not be divided into disjoint parts, but into overlapping regions. All of the two adjacent overlapping sets of points along the boundary are common. The sub-domains, containing known values of *u* at their boundary, can be calculated, applying Sommerfeld boundary condition in boundary points which are contained unknown values of *u* . Then the neighbouring sub-domains can receive the wave propagation data from the overlapping regions, i.e., through the u_{pq} values received from their neighbouring domains and through the relation (4) applicable as the continuation of the two common sets of points. In practice two grid lines of overlap in the division of Ω can ensure sufficient wave propagation.

The number of the possible starting threads in the parallel calculations is the number of the domains, where boundary values u is known. After the solution of one sub-domain two neighbouring sub-domains receive boundary values. Starting two new threads with each of these new boundary stripes the calculation can be carried out, the direction of the propagation remains the same. The gain from the parallel computing algorithm depends on the initial conditions, i.e., on the number of possible parallel treads and the position of their beginning sub-domain within the system. Generally only the following can be stated. For n sub-domains, even if the number of available computing units is sufficiently large, the necessary solution time of the parallel computation is at least n^{-2} times the sequential computing time. In the next section the duration of the calculation of a rather simple system is analyzed.

3.3. Simulation results

In Figure 6 the solution of the same problem with two Dirichlet boundary points can be seen for different subdivisions of the whole domain.

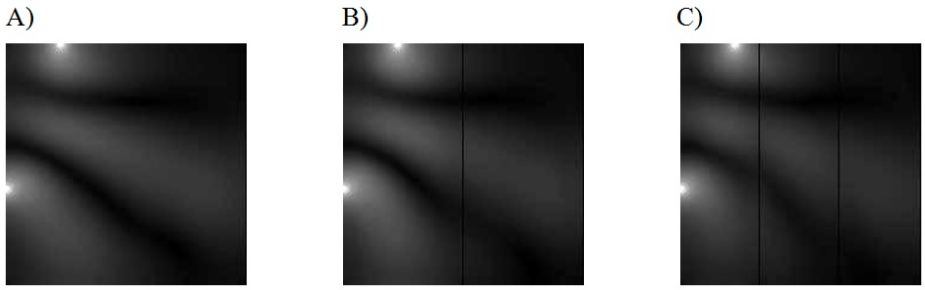


Figure 6. Subplots A), B), C) give the wave-space intensities corresponding to n=1, n=2, n=3 respectively, showing the overlapping regions. The applied data are the following. Grid size a=b=200, wavelength λ=0.0003m, d=0.000003m,

$$\Gamma' = \{(p = 45, q = 0, value = 1), (p = 0, q = 120, value = 1)\}$$

Calculation of error of solution with partition the domain compared to calculation of undivided domain is shown the following result. The relative error of the solution in Figure 6. B) according to norm $\|\ \|_1$ is 0.14, whereas according to norm $\|\ \|_\infty$ the relative error is 0.13 compared to the values of the non-divided solution in Figure 6. A). The relative error of the calculated solution in Figure 6. C) is 0.29 according to norm $\|\ \|_1$, while for norm $\|\ \|_\infty$ 0.22.

By compact storage of the matrix for determining the solution of the wave-space linear system of equations, not only the operative storing capacity load is decreased, but the necessary calculation time shortened, as well.

The operation steps demand corresponding to the case given in Figure 4.F) with the constraint $a = b$ is

$$M(a) \approx ((a+1)^2 - 1)a^2 + ((a+1)^2 - 1)(a+1) + (a+1)^2 = (a+1)^2(a^2 + a + 1) + a\,. \quad (14)$$

After dividing the domain Ω to n sub-domains, the necessary operation steps for one part can be reduced to

$$M_0(a,n) \approx \left(\frac{a}{n}+1\right)^2\left(\left(\frac{a}{n}\right)^2+\frac{a}{n}+1\right)+\frac{a}{n}. \tag{15}$$

If m parallel computational threads can be started, the wave propagation has to be followed in all the remaining sub-domains, thus the total number of operation steps is

$$M_m(a,n) \approx nm\left(\left(\frac{a}{n}+1\right)^2\left(\left(\frac{a}{n}\right)^2+\frac{a}{n}+1\right)+\frac{a}{n}\right) \tag{16}$$

Comparing M and M_m of equations (14) and (16) yields to an operation-saving factor, which relates the complete calculation within the domain to the calculation performed in the undivided Ω domain as

$$\vartheta_m(a,n) \approx \frac{M_m}{M} = \frac{nm\left(\left(\frac{a}{n}+1\right)^2\left(\left(\frac{a}{n}\right)^2+\frac{a}{n}+1\right)+\frac{a}{n}\right)}{(a+1)^2\left(a^2+a+1\right)+a} < 1 \qquad (1 < n). \tag{17}$$

By dividing the problem into sub-domain problems, the necessary real calculation time compared to that of the undivided problem depends on the value of ϑ_m, but of course, it is also affected by the programming technique.

As an example, let $a = b$ hold, and $t_2(a,n)$ denote the necessary computation time of the space with two Dirichlet boundary points in the case of n subdomains with the data visualized in Figure 6. Thus the time-saving factor $D1$ according to this calculation is

$$D1(n) = \frac{t_2(a,n)}{t_2(a,1)}. \tag{18}$$

On the basis of measurements, according to relation (18), the values $D1$ are given in Tab. 1. At a fixed grid size a let us introduce another factor $D2$ in order to facilitate the exploration of the relation between $D1$ and ϑ_2 as

$$D2(n) = c(n)\vartheta_2(a,n). \tag{19}$$

Expression $c(n)$ can be determined the following way. Condition

$$D1(n) \approx D2(n) \tag{20}$$

has to be satisfied, in order to make $D1$ and ϑ_2 comparable. Based on the experiments, condition (2) is ensured by relation

$$c(n) = n. \tag{21}$$

The value $D2$ calculated according to condition (21) can be seen in Table 1., while the realization of (20) is illustrated in Figure 7.

Table 1. The values D1 and D2 for various numbers of sub-domains n in case of c(n)=n.

n	2	3	4	5	6	7	8	9	10
D1	0,501	0,230	0,132	0,092	0,064	0,050	0,040	0,032	0,028
D2	0,508	0,229	0,131	0,085	0,06	0,045	0,035	0,028	0,023

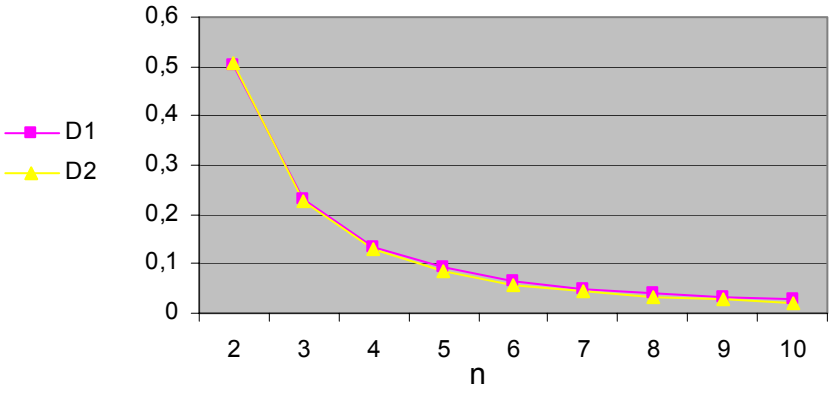


Figure 7. The values D1 and D2 for various numbers of sub-domains n in case of $c(n) = n$.

4. Conclusion

With the development of hardware and software technology, together with increasing calculation capacities and memory-optimization, the direct method can also be used successfully in the case of small-sized electromagnetic wave-ranges of relatively long wavelength. For reducing the required storing capacity, a special matrix reduction method was introduced in this paper, using sliding matrices. For aiding parallel computing, a wave space dividing method was also introduced and tested with small overlaps ensuring the wave front transmission between the space parts with reasonable results.

Beyond the studied discretization method, there are many other possibilities to solve the above problems. These solutions can be characterized by the number of computational steps and the necessary memory capacity for the sufficiently accurate results, which, at the end, determine the necessary computing time. The following estimations are based on the case of square grids with a=b. The method considered in Section 3, is based on stripped Gaussian elimination, its necessary computational capacity according to (14) is $O(a^4)$. The filling does not impact the whole matrix, but only approximately a^3 elements. Applying LU decomposition, a quicker solution can be achieved with significantly larger storage need. The conjugate gradient method is much more favourable both in computational (less than $O(a^4)$) and in storage needs, but it demands a symmetric positive definite matrix, which has to be pre-conditioned for an efficient convergence. The special shape of the studied domain makes it possible to apply

FFT (Fast Fourier Transform), which presents a quick solution with a computational requirement of $O(a^2 \log a)$ steps, and its storage need is $O(a^2 \log a)$. Wavelet based differential equation solving methods of order a exist [5], but their application range is limited mostly to elliptic differential equations and Schrödinger type eigenvalue equations [9], and their straightforward representation of the kinetic energy can lead to systematic errors [10], which results in slower convergence. The boundary element method's storage capacity demand is approximately the same as that of the finite differences method, but its algorithmical complexity is $O(a^3)$. The best solution seems to be the multigrid method, since its computational need is $O(a^2)$, and memory demand is about the same as in the conjugate gradient method. More accurate and effective solution can be achieved by unevenly meshed multigrid method.

The main advantage of the method presented in this article is the simple algorithm which can be easily applied even if no complex, efficient program is available, and the development time has to be minimal. It can also play a role in the design of the uneven grid multigrid method by giving a rough scale solution of the problem as a starting point.

References

[1] Arnold, S.: *Mathematische Theorie der Diffraction*, Math. Ann., Vol. 47, pp. 317–374, (1896).

[2] Buzbee, B. L., Dorr, F. W., George, J. A., and Golub, G. H.: *The direct solution of the discrete Poisson equation on irregular regions*, SIAM J. Numer. Anal., Vol. 8, pp. 722-730, (1971).

[3] Choi, C.T.M., Webb, J.P.: *The wave-envelope method and absorbing boundary conditions*, IEEE Transactions on Magnetics, Vol. 33, Issue 2, pp.1420 – 1423 (1997).

[4] Claudio M.: A *Reference Discretiazion Strategy for the Numerical Solution of Physical Field Problems,* Advances In Imaging and Electron Physics, Vol. 2. (2002).

[5] Dahmen, W.: *Wavelet methods for PDEs – some recent developments*, J. Comput. App. Math., Vol.128, pp. 133-185, (2001)

[6] Duff, I. S., Erisman, A. M., and Reid, J. K.: *Direct Methods for Sparse Matrices*, Clarendon*,* Oxford (1986).

[7] Iványi, A.: *Folytonos és diszkrét szimulációk az elektrodinamikában (Continuous and discrete simulations in electrodynamics)*, Akadémiai Kiadó*,* Budapest, pp. 180-240, (2003).

[8] Lois C. M., Romeo F. S., David E. K., Hafiz M. A.: *Additive Schwarz Methods with Nonreflecting Boundary Condition of Helmholtz Problems*, Contemporary Mathematics*,* Vol. 218, pp. 349-353, (1998).

[9] Nagy, Sz., Pipek, J.: *A wavelet-based adaptive method for determining eigenstates of electronic systems*, Theor. Chem. Acc., Vol. 125, pp. 471-479, (2010).

[10] Pipek, J., Nagy, Sz.: *Artifacts of grid-based electron structure calculations*, Chem. Phys. Lett., Vol. 464, pp. 103-106, (2008).

[11] Shashkov, M.: *Conservative Finite-Difference Methods on General Grids*, CRC Press, Boca Raton, FL (1996).

[12] Shashkov, M., Steinberg, S.: *Support-operator finite-difference algorithms for general elliptic problems*. J. Comput. Phys., Vol. 118, pp. 13 1-15 1, (1995).

[13] Simonyi, K., Fodor, Gy.: *Electrodynamics*, Tankönyvkiadó, Budapest Vol. 2, pp. 290–364, (1967).
[14] Strand, B.: *Summation by parts for finite difference approximation for dldx.* J. Comput. Phys. Vol. 110, pp.47-67, (1994).

Advanced Sensitivity Measurement of Low Frequency RFID Transponder Coils

P. Csurgai, M. Kuczmann

"Széchenyi István" University, Department of Telecommunication
H-9026 Győr, Egyetem tér 1. peter.csurgai@yahoo.com

Abstract: *Problem:* More and more applications use Radio Frequency Identification (RFID) technology for wireless identification or data transfer in consumer electronics, automation and automotive market nowadays. The most important component from the communication point of view is the transponder coil. There are numerous transponder coil manufacturers in the world, and even more RFID application manufacturer. For both of them it is important to recognize the differences of various transponder constructions. This paper focuses only on the measurement of the Low Frequency RFID transponder coils, which operates using the inductive coupling.
Solution: The parameter which gives the most information about the performance of the transponder coil is the sensitivity. This paper explains what the sensitivity means, how it can be calculated, and gives alternatives for the measurements. Reveals the disadvantage of the widely used standard sensitivity measurement, and provides a proposal for a new kind of measurement, by changing the excitation signal and tuning the transponder coil with a capacitor. The new measurement method results two independent parameters instead of one that give more information about the transponder coil and its performance. The paper also presents the exact phenomenon that takes place in the resonant circuit, in other words the time function of the voltage and current of the circuit, during the measurement and the standard operation.
Verification: There can be found in the end of this paper an example for the measurement, where different transponder constructions are compared. The evaluation of the measurement is also introduced in two different ways. Based on the measurement and on the evaluation of the different transponder constructions the impacts of the changes in the construction can be identified and proved.

Keywords: LF RFID, LF transponder, Sensitivity, Sensitivity measurement

1. Introduction

More and more applications use RFID technology for wireless identification or data transfer in consumer electronics, automation and automotive market nowadays. These

RFID systems can be classified in certain groups based on their properties, requirements and operating principles. Most commonly when high privacy and thus low reading distance, but reliable connection between the transmitter and the receiver is needed together with passive operating mode low frequency RFID system is chosen. Due to the operating principle these systems have the advantage that the transmission is not disturbed very much or blocked by obstacles or bad weather conditions. The operating principle of these systems is the inductive coupling. The main components of the system from the communication point of view, which ensure the inductive coupling are the primary and secondary resonant circuits. The resonant circuits consist of at least two components, a tuning capacitor and a reader or transponder coil, depending on which side of the coupled inductors is observed. Inductive coupling means a connection between the two coils through the electromagnetic field. The mutual inductance between the transponder and Reader coil helps to forward the energy and the transmitted signals between the two coils. These two coils can be seen as a weakly coupled transformer, where the primary coil is the Reader and the secondary coil is the transponder coil. The properties of these two coils determine mostly the limits of the communication. Therefore the coils in the two side of the transformer play a major role in the energy, the signal transfer and the operating of the complete RFID system. From this point of view it is very important to be aware of the behaviour, the properties and the limitations of the reader and transponder coils. For instance the maximum reading distance of the RFID application is determined by the coils. [1]

2. Transponder coil parameters and standard sensitivity measurement

The key parameters of the transponder coils – just like other inductors – are the inductance value, the DC resistance and the quality factor. As the component has to withstand only signal level load and not power level therefore values like saturation current or rated current are not required and so thus the manufacturers don't define or state these parameters. However due to the special use of the inductor a special parameter is introduced for the transponder coils, which helps comparing different constructions and the products of different manufacturers. This parameter is the sensitivity value, which gives information about how sensitive the component to the changing external electromagnetic field is. The sensitivity is defined as the quotient of the induced voltage and the strength of the magnetic field, which induces the voltage across the inductor. [2]

$$S = \frac{V_i}{H} \text{ or } S = \frac{V_i}{B}, \tag{1}$$

where V_i is the induced voltage across the inductor due to changing electromagnetic field, and H or B is the strength of the electromagnetic field, or the flux density.

The unit of the sensitivity is

$$[S] = \left[\frac{mV}{Am^{-1}}\right] \text{ or } \left[\frac{mV}{\mu T}\right], \tag{2}$$

depending what type of excitation is considered during the measurement.

The measurement of the sensitivity can be performed with the help of a device, which is capable of creating uniform (nearly constant in magnitude and in orientation as well) electromagnetic field. These devices are the Helmholtz coil and the Maxwell coil. A typical sensitivity test setup consists of a signal generator, a Helmholtz coil and an oscilloscope, Figure 1. [2]

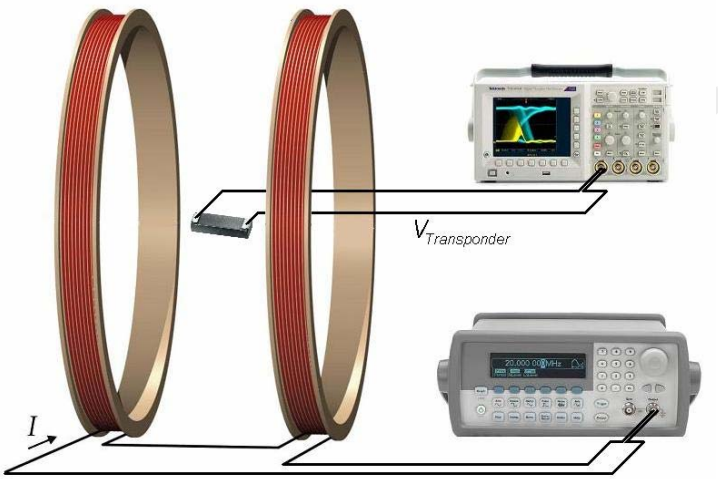


Figure 1. Typical Sensitivity test setup

This test setup is the most common and most conservative version in the variations of the sensitivity measurements. The inductor is measured as a single part and so there is not any attached component like tuning capacitor or damping resistor. The excitation signal is a sine wave, which is generated by the signal generator, and feeds the Helmholtz coil, which creates the changing electromagnetic field. The induced voltage on the transponder coil is measured by an oscilloscope, and the sensitivity can be calculated using the formula (1). This kind of measurement results one scalar parameter, which allows the comparison of different transponder coils, but does not give any further information about the reason or the background of the difference. The sensitivity measurement as single component measurement arises some questions and problems. The first problem is – which has an influence on the measured sensitivity value – the first Self Resonance Frequency (SRF) of the inductor. There is not any passive component, which behaves in the frequency domain purely resistive, capacitive, or inductive. Inductors should be considered not only as an inductive component, but as a more complex circuit where the source of the losses can be identified by capacitive and resistive components, Figure 2. [3, 4]

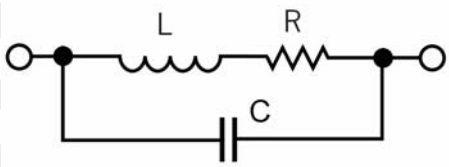


Figure 2. Equivalent circuit of a lossy inductor

The transponder coil during the measurement behaves not only as a purely inductive component, but as a damped resonant circuit. So the measured sensitivity value is influenced by the resonance phenomenon, which might result a higher value than the real one. The different parasitic capacitances and thus the different self resonance frequencies are determined by the different winding methods and the used components and materials. The higher the parasitic capacitance the lower the self resonance frequency is. For example an inductor, which is wound by layer winding appears to have higher sensitivity than an inductor which has the same number of turns and ferrite core, but the winding method is random winding. In resonant circuits in case of sinusoidal excitation the output voltage of an element can be calculated using the impedances instead of the differential equation of the circuit. According to this the transfer function of the inductor during the sensitivity measurement can be calculated as the quotient of the output and the input voltage, which is equal to the inductor impedance divided by the total impedance of the *RLC* circuit:

$$|H(\omega)| = \frac{\omega L}{\sqrt{R^2 + \left(\omega L - \frac{1}{\omega C}\right)^2}} \tag{3}$$

Drawing the transfer function curves of transponder coils with different parasitic (winding) capacitances the huge difference is evident, Figure 3.

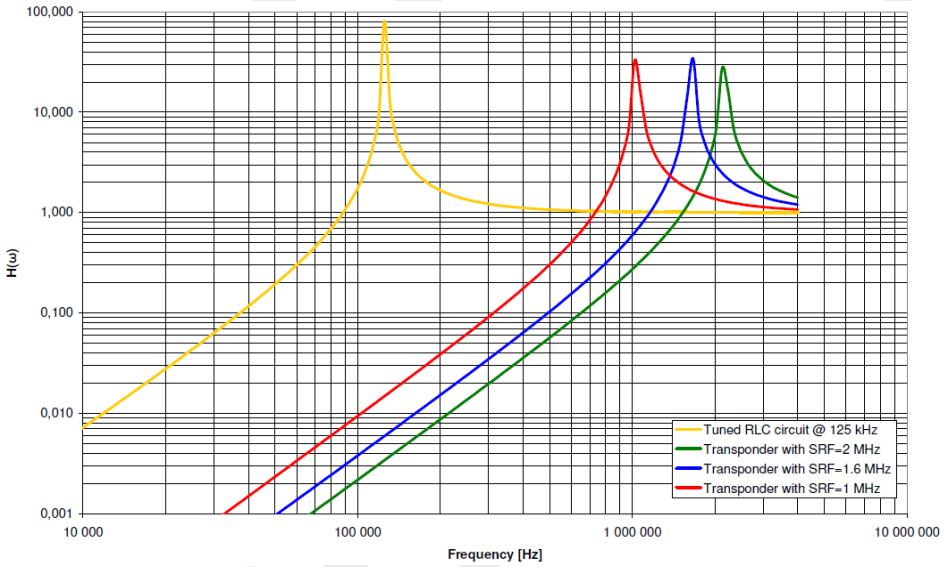


Figure 3. Transfer function curves of transponder coils with different capacitances

The difference in the transfer function values between the various transponders represents the difference in the measured sensitivity value. The false measurement may lead to false conclusion and decision, because a transponder coil with lower real

performance might have a higher sensitivity value in this kind of measurement. Another problem of this measurement is that the test does not simulate the real operating of the inductor. In the real RFID application the transponder is always tuned to a specific frequency, and thus the induced voltage across the inductor is many times higher due the resonance phenomenon, as can be seen in the transfer function, Figure 3.

3. Sensitivity measurement in tuned resonant circuit

The above explained measurement error can be eliminated by the resonant circuit measurement, where the circumstances and the mode of the operation is almost identical to the real use of the transponder coil. In these kinds of measurements the transponder coil is tuned to the desired frequency by a tuning capacitor. The frequency of the RFID system is usually between 125 kHz and 134.2 kHz. Although this measurement approximates better the real operation, but it still has the problem that the output is only one scalar parameter, because the used evaluation method of the measurement is identical to the formula (1).

It is important to note when a resonant circuit is measured, the capacitor is qualified together with the transponder coil as well. So the capacitor influences the whole circuit and the measured values. But choosing a proper capacitor the losses of the capacitor can be neglected compared to the losses of the transponder coil. A rotary capacitor, which has air gap between its plates, has a sufficient high quality factor. The frequency of the measurement allows the using of these kinds of capacitors, because in this frequency range the dielectric loss of the capacitor can be neglected. From this point of view using tuning capacitor will not change noticeably the measured sensitivity value.

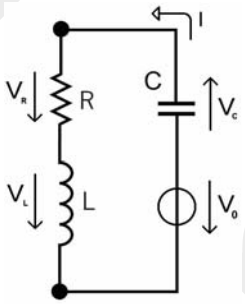


Figure 4. Ttransponder coil in tuned resonant circuit

4. Advanced Sensitivity measurement

Changing the excitation signal and the evaluation of the measurement two independent parameters can be identified. Taking a deeper look in the resonant circuit (Figure 4), which is built up from the transponder coil (*L*), the tuning capacitor (*C*) and the damping resistor (*R*) the complete resonance phenomenon can be described as follows:

- Using the Kirchoff's loop rule on the circuit (with the reference in Figure 4) the following equation can be identified:

$$V_L + V_R + V_C = V_0 . \tag{4}$$

- Due to the same current flows through all the components it is practical to calculate the voltages in the function of the current:

$$V_R = R \cdot i \,, \tag{5}$$

$$V_L = L \cdot \frac{di}{dt} \,, \tag{6}$$

$$V_C = \frac{1}{C} \cdot \int i dt \,. \tag{7}$$

- Substituting the voltages described above (5-7) to the Kirchhoff's loop (4) the following second order differential equations will be created:

$$L\frac{di}{dt} + Ri + \frac{1}{C}\int_0^t i dt = V_0 \,, \tag{8}$$

$$\frac{d^2 i}{dt^2} + \frac{R}{L}\frac{di}{dt} + \frac{1}{LC} i = \frac{dV_0}{dt} \,. \tag{9}$$

- Introducing the $\omega_0 = \frac{1}{\sqrt{LC}}$ parameter the differential equation can be written in the following simpler form:

$$\frac{d^2 i}{dt^2} + \frac{R}{L}\frac{di}{dt} + \omega_0{}^2 i = \frac{dV_0}{dt} \,. \tag{10}$$

- The characteristic equation of the (10) differential equation is:

$$\lambda_{1,2} = -\frac{R}{2L} \pm \sqrt{\frac{R^2}{4L^2} - \omega_0{}^2} \,. \tag{11}$$

- Assuming that the damping of the resonant circuit is below the critical damping, and introducing the $\delta = \frac{R}{2L}$ the root of the characteristic equation will be:

$$\lambda_{1,2} = -\delta \pm j \cdot \sqrt{\omega_0{}^2 - \delta^2} \,. \tag{12}$$

- Introducing the $\omega = \sqrt{\omega_0{}^2 - \delta^2}$ the root of the equation can be further simplified in the following form:

$$\lambda_{1,2} = -\delta \pm j\omega \,. \tag{13}$$

- The solution of the differential equation (with considering the $i_{st} = 0$ condition) is:

$$i = A_1 e^{\lambda_1 t} + A_2 e^{\lambda_2 t} \,. \tag{14}$$

- The initial condition of the current in the resonant circuit is:

$$i(0) = A_1 + A_2 = 0 \,. \tag{15}$$

- Substituting the (6) equation with the (15):

$$L\left(\frac{di}{dt}\right)_{t=0} = L\left[A_1\lambda_1 + A_2\lambda_2\right] = V_0 \,. \tag{16}$$

- The following results will occur for the A_1 and A_2 parameters from the last two equations:

$$A_1 = \frac{V_0}{(\lambda_1 - \lambda_2)L} \,, \tag{17}$$

$$A_2 = -\frac{V_0}{(\lambda_1 - \lambda_2)L} \,. \tag{18}$$

- So thus the current of the circuit in complex form will be the following:

$$i = \frac{V_0}{(\lambda_1 - \lambda_2)L} \cdot \left(e^{\lambda_1 t} - e^{\lambda_2 t}\right) \,. \tag{19}$$

- Using the Euler's formula the solution for the current is:

$$i = \frac{V_0}{j \cdot 2\omega L} \cdot e^{-\delta t} \cdot \left(e^{j\omega t} - e^{-j\omega t}\right) \,, \tag{20}$$

$$i = \frac{V_0}{\omega L} \cdot e^{-\delta t} \cdot \left(\frac{e^{j\omega t} - e^{-j\omega t}}{2j}\right) \,, \tag{21}$$

$$i = \frac{V_0}{\omega L} \cdot e^{-\delta t} \cdot \sin(\omega t) \,. \tag{22}$$

- The voltage of the coil, which shall be monitored during the measurement according to the (6) equation is:

$$V_L(t) = L \cdot \frac{di}{dt} = \frac{V_0 L}{\omega L} \cdot \left(-\delta \cdot e^{-\delta t} \cdot \sin(\omega t) + \omega \cdot e^{-\delta t} \cdot \cos(\omega t)\right) \approx$$

$$\approx V_0 \cdot e^{-\delta t} \cdot \cos(\omega t) \,. \tag{23}$$

The calculated voltage and current of a resonant circuit can be seen in the Figure 5. The calculated example represents a typical LF RFID circuit, where the components have the following values: L=2.36mH, C=680pF, R=20Ω.

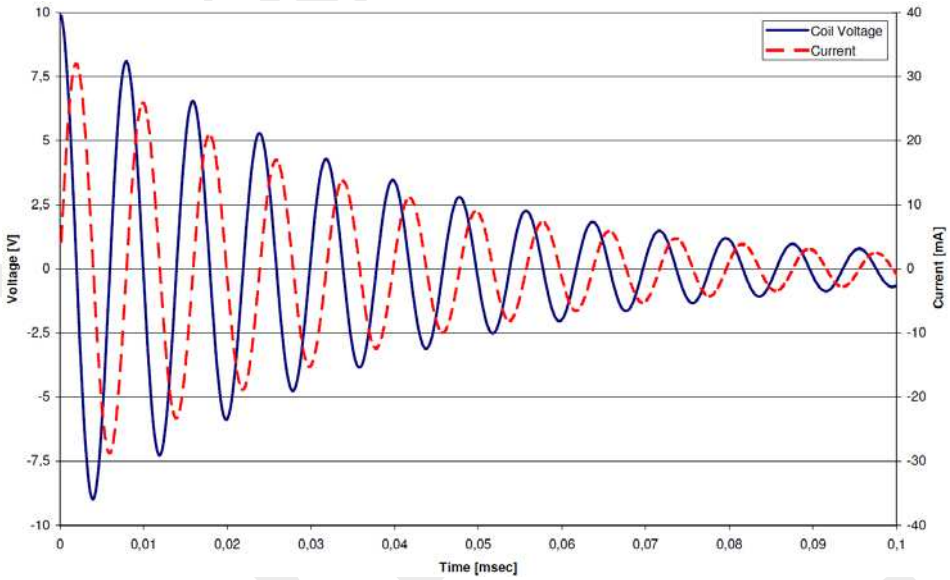


Figure 5. Simulated time function of the voltage and current of a RLC circuit

As can be seen from the time function of the voltage on the coil, the shape of the curve is determined by three parameters: the initial amplitude (V_0), the damping constant (δ) and the resonant frequency of the circuit (ω). One of these three parameters is intentionally fixed in the measurement of different transponder coils. The fixed parameter is the resonance frequency. The frequency is defined by the application, which shall be the same during the benchmark of different transponder constructions to provide the same external circumstances for the coils. This resonance frequency can be adjusted by a tuning capacitor. The two other parameters are the amplitude and damping factor. These two parameters give more information than only the scalar sensitivity value, because both parameters are independent from each other, and have a physical meaning. The amplitude of the induced voltage gives information how sensitive really the transponder coil to the electromagnetic field is. Without any exception a more sensitive coil shall have higher induced voltage (amplitude of the first peak) regardless what the self resonance frequency of the coil is. The damping factor gives information about the quality factor of the whole resonant circuit. As the losses of the resonant circuit mostly determined by the losses of the transponder coil, the damping factor gives information only about the transponder coil. In this frequency range the losses of the inductor are the eddy current losses, the iron losses of the ferrite core and the resistive losses of the winding wire. Usually in the low frequency range skin and proximity effect doesn't play major role. It is obvious that an inductor with higher losses has a higher damping factor too and so the swinging energy between the tuning capacitor and the transponder coil will collapse much faster. It is important to emphasize that the measurement setup is almost identical in this case than in the original measurement, only the tuning capacitor and the excitation signal is different.

The excitation of the resonant circuit shall be a Dirac delta pulse. The resonant circuit is fed through the transponder coil, therefore the Dirac delta will occur on the coil due to the electromagnetic field created by the Helmholtz coil. The induced voltage which feeds the resonant circuit can be identified by the Faraday's law of induction:

$$V_L = -N\frac{d\Phi}{dt} = \delta(t) \tag{24}$$

The required signal on the Helmholtz coil for getting the Dirac delta pulse on the transponder coil can be calculated according to the (24) equation. If the required signal on the coil is a Dirac delta pulse, the signal on the Helmholtz coil shall be the integral of the Dirac delta pulse, as it can be seen in the follows.

$$\Phi_{Helmholtz} = \int \delta(t)dt = 1(t) \tag{25}$$

So the flux of the Helmholtz coil should be a step function. As the flux and the field strength of the coil is proportional to the current, therefore the excitation current of the Helmholtz coil should be also a step function. The current of a coil can not be changed in infinite short time, therefore the only way to get the desired induced voltage on the transponder coil to apply an increasing current on the Helmholtz coil with sufficient high slope.

The evaluation of the measurement can be performed by analysing the time function of the voltage or the current of the transponder coil. The current or the voltage of the coil can be easily recorded with the help of an oscilloscope and a voltage or current probe. Based on the curves the two parameters (amplitude, damping factor) can be calculated, which reveal the background of the difference of the transponders.

5. Verification of the new advanced measurement

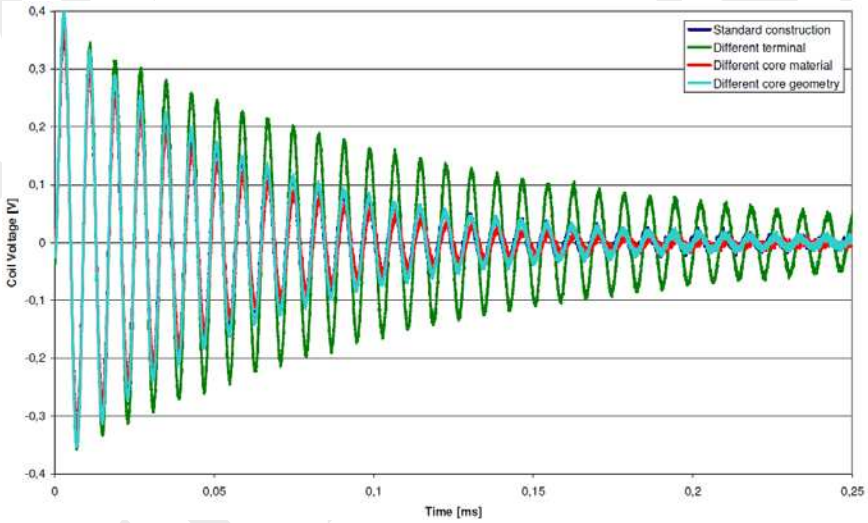


Figure 6. V(t) function of different transponder constructions

A comparison of different transponders with the above mentioned new measurement can be seen in the follows.

For the verification of the measurement four different transponder constructions were tested. The first sample, which is called standard construction is available on the market from an existing manufacturer, and represents a standard transponder coil. The three other parts differs from the standard one in the ferrite core material, the ferrite core geometry and the terminal length. As it can be seen the measurement method gives information about the reason of the difference in the sensitivity value as well. So thus based on the curves of this new measurement the impacts of each change in the construction can be evaluated.

The differences can be more clearly seen, if the exponential envelopes of the different constructions are plotted only, without the disturbing sine wave, Figure 7.

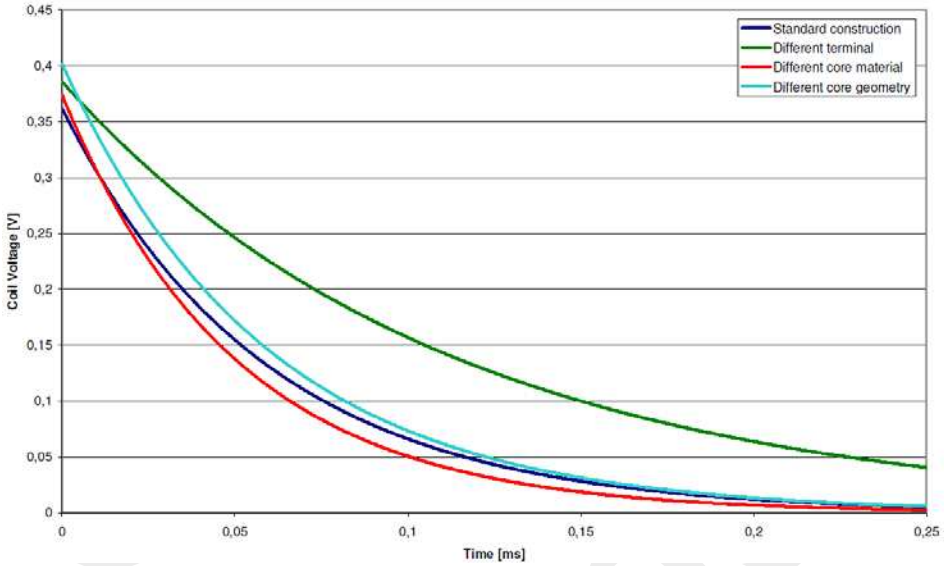


Figure 7. Exponential envelopes of different transponder constructions

Comparing each sample to the standard one, conclusions about the construction changes can be made.

- Different terminal: By choosing a better terminal geometry not only the amplitude of the signal, but the damping factor of the circuit is also better than using the standard one. The reason is that there are much less eddy current losses generated in the terminal, and so the resonant circuit dissipates less energy.
- Different core material: Changing the core material from NiZn to MnZn ferrite, on one hand causes higher sensitivity amplitude, but on the other hand the damping factor is higher, therefore this change made the transponder coil worse in total. The reason of the higher sensitivity amplitude is the higher permeability, and the higher damping factor occurred due to the high conductivity of the MnZn ferrite, which leads to higher eddy current losses.

- Different core geometry: The highest sensitivity amplitude value is given in this case. The reason is that the sensitivity is mostly determined by the length and the shape of the ferrite core.

Combining and using the conclusions of the tests above, the performance of the transponder coil can be increased during a development process. These conclusions would have been more difficult to find out if this new measurement had not been available.

The final result of the measurement also can be achieved by evaluating the measured data with Fourier Transform. Using the Fast Fourier Transform procedure on the measured data the spectral density of the signal (current or voltage of the resonant circuit) can be calculated. It is not possible directly to calculate the amplitude and the damping factor from the spectral density, but the peak amplitude on the resonant frequency and the bandwidth,

$$B = f_H - f_L, \tag{26}$$

provides the same kind of information as the sensitivity amplitude and the damping factor in time domain.

The quality factor of the resonant circuit using the bandwidth of the signal can be calculated by the following equation:

$$Q = \frac{f_0}{B} = \frac{f_0}{f_H - f_L} \tag{27}$$

It is a bit harder to evaluate the FFT data, as can be seen on the Figure 8, but the comparison of different constructions also can be made.

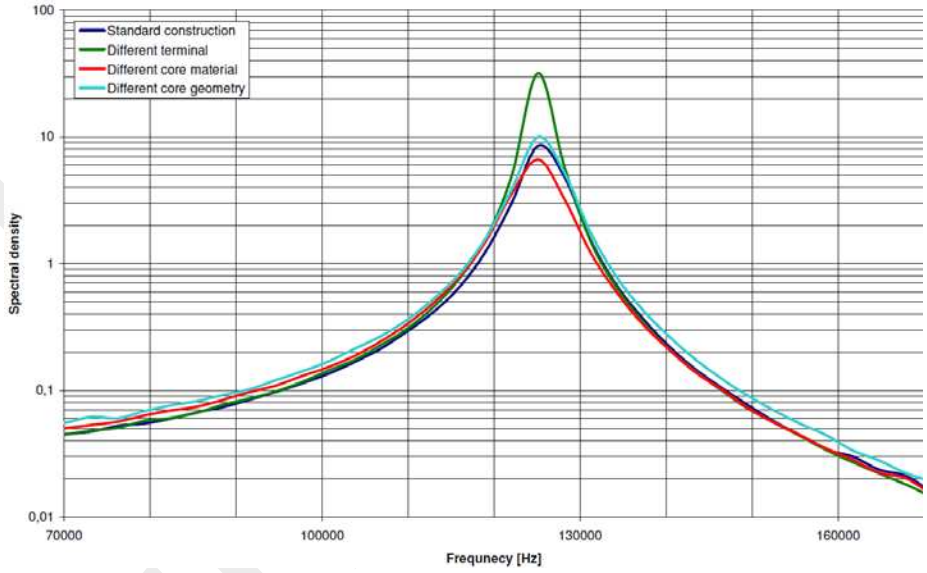


Figure 8. Spectral density of the signals of the sensitivity measurement

6. Summary and field of application

The design processes require accurate and reliable information about all the components and parts what the designed product contains either it is an RFID application or a transponder coil design or development. As all the engineers try to reach the highest possible efficiency and performance it is highly required to improve the available measurement methods to learn even more about the components. The presented measurement method helps to identify and understand the differences and backgrounds of different transponder constructions.

References

[1] Klaus Finkenzeller – RFID Handbook, Carl Hanser Verlag, Munich/FRG, 1999.
[2] EPCOS Inductors Handbook 2009, EPCOS AG, Germany 2008.
[3] Marian K. Kazimierczuk, Giuseppe Sancineto, Gabriele Grandi, Ugo Reggiani, Antonio Massarini – High-Frequency Small-Signal Model of Ferrite Core Inductors, IEEE Transactions on Magnetics. Vol 35, No. 5, September 1999.
[4] Leslie Green, Gould-Nicolet Technologies – RF-inductor modelling for the 21st century, www.ednmag.com September/2001.

RF Inductor Development by Using the FEM

Z. Pólik, M. Kuczmann

"Széchenyi István" University Department of Telecommunication
H-9026 Győr, Egyetem tér 1. polikzoltan@gmail.com

Abstract: The design of inductors is not an easy and cheap task considering the dimensions, the nominal value of inductance, the quality factor and the impedance of the component. Before the beginning of manufacturing a new type of inductors, a lot of trial components have to produce, have to measure and have to try out. Finite element modelling is a well-tried process to examine engineering products before manufacturing them. To reduce the cost and the time of the design process, in the paper a finite element model has been built up to simulate inductors. By the implemented model the component designers can examine the behaviour of an arbitrary inductor and the effects of the modification on its geometry or on its winding.

Keywords: RF inductor, Finite element method, Quality factor, Absorbing boundary condition

1. Introduction

In the paper the model is presented, which is able to simulate the important attributes of the component, for example the inductance, the impedance and the quality factor. The comparison of the experimental and the simulated attributes of the inductor will also be shown. By using the built up model the development possibilities of the inductor have been examined through the modification of the winding.

An electronic component manufacturer develops, manufactures and markets electronic components, modules and systems, focusing on fast-growing leading-edge technology markets: in information technology (IT) and telecommunications, but also in automotive, industrial and consumer electronics. To satisfy the private demand of some customers, the company needs to design new components and modify the actual parameters of several types of inductors. The company has a developing team to find the best geometry, material, and manufacturing process of inductors. Many researches are improving the attributes of their components, i.e. the inductance, the quality factor, the maximum current, the sensitivity and so on, through applying new materials and new geometries, which are also developing there [1].

The first aim of this work is to build up a finite element model, with which the attributes of an optional inductor can be simulated. Since inductors are working in wide range of the frequency, the model has to provide correct results at low and very high frequencies, as well.

The second aim is to examine the development possibilities of the quality factor by modifying the coil of the component. In physics and engineering, the quality factor, or Q-factor is a dimensionless parameter that compares the time constant for decay of an oscillating physical system's amplitude to its oscillation period. To increase the quality factor, the original winding of the inductor has been replaced by a newly designed one. In the new coil several wires and winding type – "closely-" and "widely spaced" coils – have been applied to found the best arrangement. The attributes of prepared components have been measured and by using the finite element model they have been simulated. The measured and the simulated results have been compared.

In the engineering point of view, the optimization of an attribute in the case of an optional component is a very huge problem, which cannot be solved by using the well-tried manufacture and measure method. Numerically, it can be solved by using a finite element model and an iterative modification of the parameters of the problem can be achieved. It is the main motivation of this research. Here, the model has been built up and checked by experimental results.

The inductor, which has been examined, is an SMT (Surface Mount Technology) one, which is usually working in the range of the radio frequency. The dimensions of the component are 1.24±0.04 mm × 1.22±0.04 mm × 2.03±0.04 mm [1]. The component has a cubic coil on ferrite or ceramic core, depending on the application field of it. The diameter of the winding wire is 50 µm, is welded to the thick film coating on its terminations, is made of silver, palladium and platinum or, in an other case it is made of wolfram, nickel and gold. It has a flat top made of epoxy for vacuum pickup. The major features of the inductor are the high resonant frequency, between 300 MHz and 9 GHz depending on the type of the component, and the close inductance tolerance. This type of inductor is used in resonant circuits, antenna amplifiers, mobile phones, Digital Enhanced Cordless Telecommunications (DECT) systems, car access systems, tire pressure monitoring systems (TPMS), wireless communication systems and global positioning systems (GPS) [1]. The microscopic photo of the component can be seen in Figure 1.

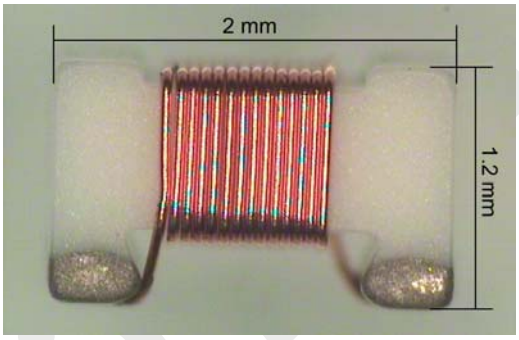


Figure 1. The microscopic photo of the component

It is important to note that different applications need different values of inductance, resistance, maximum current and quality factor. The most of the parameters can be changed easily by the modification of the winding wire or the material of the core, but the modification of one parameter causes variation in the other parameters, as well [2].

For example, if the inductance of the component is modified via the modification of the winding, i.e. the number of turns is increased or the distance between adjacent coils are decreased, the resistance of the inductor is increasing, the quality factor is decreasing and the SRF (self resonant frequency) is also decreasing both in the two cases. But the reason of the variation of the attributes is different in the mentioned two examples.

In the first case, the resistance is increasing in the effect of the more coils, because the longer wire means higher resistance, the quality factor is decreasing according to the expression of the quality factor, and the SRF is decreasing through the higher capacitance between the coils. In the second case, the resistance is increasing by the reason of the higher proximity effect between the coils, which get closer to each other, the Q-factor is decreasing because of the increasing of the resistance, and finally, the SRF is decreasing through the high capacitance between the closer coils. It seems that it is not an easy task to improve parameters without the deterioration of other ones [2].

Between 2.7 nH and 820 nH, inductors are manufactured with ceramic core and over 1 µH they are made with ferrite core. The reason of this is that the higher value of the inductance is only achievable with higher permeability of the core. However, ferrite core has disadvantages, i.e. the eddy current losses and the hysteresis losses, so the quality factor of a ceramic core inductor can be higher.

The quality factor is one of the most important attribute of inductors, the high value of the Q is necessary in several cases, for example in oscillators, and in tuned circuits. Because of the continuous development of electronic components manufacturers needs to produce inductors with higher and higher Q. Now, the reachable value of it is at least 60 between 85 MHz and 110 MHz. In the present, this value is about 30 as it can be seen in Figure 2. At first, only the effects of the modification of the winding wire have been examined. The core has standard dimensions and it is made of standard materials, so they should not be modified in the present study.

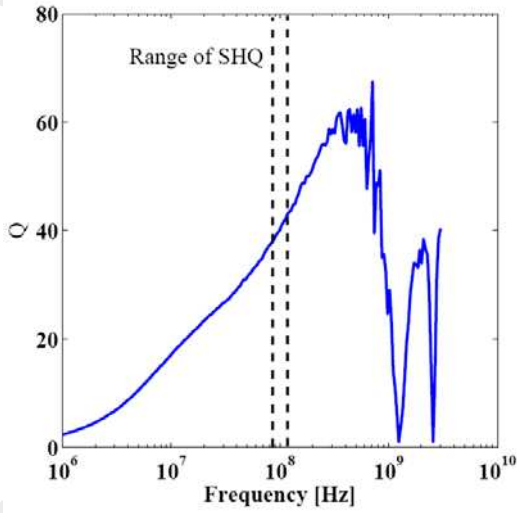


Figure 2. The actual quality factor as a function of the frequency

The finite element model of the problem has been built up by using the COMSOL Multiphysics software package and the trial components have been measured by an Agilent E4991A RF impedance and material analyzer, which can be seen in Figure 3 [3].

Figure 3. E4991A RF impedance and material analyzer

2. Governing equations

During the simulation of inductors, the base equations are the full form of the Maxwell's equations, because at high frequency the effect of the eddy currents and the displacement currents cannot be neglected [2], [4]. Since only sinusoidal excitation have been used in the problem, the operator $\partial/\partial t$ has been replaced by $j\omega$.

The partial differential equations and the boundary conditions are the following in the investigated case [5]-[11]:

$$\nabla \times \left(\nu \nabla \times \boldsymbol{A}\right) - \omega^2 \varepsilon \, \boldsymbol{A} = \boldsymbol{0}, \quad \text{in} \quad \Omega_n , \tag{1}$$

$$\nabla \times (\nu \nabla \times \boldsymbol{A}) - \mathrm{j}\omega\sigma \boldsymbol{A} = \boldsymbol{J}_0, \quad \text{in} \quad \Omega_c , \tag{2}$$

$$\nabla \times (\nu \nabla \times \boldsymbol{A}) - \omega^2 \varepsilon \boldsymbol{A} = \boldsymbol{0}, \quad \text{in} \quad \Omega_D , \tag{3}$$

$$\nu \nabla \times \boldsymbol{A} = \boldsymbol{0}, \quad \text{on} \quad \Gamma_{H_n} , \tag{4}$$

$$\boldsymbol{n} \times \boldsymbol{A} = \boldsymbol{0}, \quad \text{on} \quad \Gamma_B , \tag{5}$$

$$\boldsymbol{n}_D \times \boldsymbol{A} + \boldsymbol{n}_n \times \boldsymbol{A} = \boldsymbol{0}, \quad \text{on} \quad \Gamma_{nD} , \tag{6}$$

$$(\nu \nabla \times \boldsymbol{A}) \times \boldsymbol{n}_D + (\nu \nabla \times \boldsymbol{A}) \times \boldsymbol{n}_n = \boldsymbol{0}, \quad \text{on} \quad \Gamma_{nD} , \tag{7}$$

$$\boldsymbol{n}_c \times \boldsymbol{A} + \boldsymbol{n}_D \times \boldsymbol{A} = \boldsymbol{0}, \quad \text{on} \quad \Gamma_{cD} , \tag{8}$$

$$(\nu \nabla \times \boldsymbol{A}) \times \boldsymbol{n}_c + (\nu \nabla \times \boldsymbol{A}) \times \boldsymbol{n}_D = \boldsymbol{0}, \quad \text{on} \; \Gamma_{cD} , \tag{9}$$

$$\boldsymbol{n} \times \boldsymbol{A} = \boldsymbol{0}, \quad \text{on} \quad \Gamma_E , \tag{10}$$

$$\nu \nabla \times \boldsymbol{A} = \boldsymbol{0}, \quad \text{on} \quad \Gamma_{H_D} , \tag{11}$$

$$(\nu \nabla \times \boldsymbol{A}) \times \boldsymbol{n}_D + (\nu \nabla \times \boldsymbol{A}) \times \boldsymbol{n}_n = \boldsymbol{0}, \quad \text{on} \quad \Gamma_{nD} , \tag{12}$$

where, $\boldsymbol{A}$ is the magnetic vector potential, ν is $1/\mu$, where μ is the permeability, ε is the permittivity, σ is the conductivity of the material, ω is the angular frequency of the excitation, $\boldsymbol{J}_0$ is the current density of the excitation and $\boldsymbol{n}$ is the normal unit vector of the boundary. In this example (6) and (8) are satisfied automatically [5], [11]. In *Figure 4*, the structure of a wave propagation field problem can be seen, where a dielectric material is bounded by Γ_E, Γ_{H_D} and Γ_{nD}; Γ_a is the artificial far boundary. The air is bounded by Γ_B and Γ_{H_n}. Γ_{cD} is the boundary between the conducting material and the dielectric material.

2.1. Absorbing boundary condition

In some cases, particularly at high frequencies it is important that the electromagnetic waves should not reflect from the artificial far boundary. Here, the so-called absorbing boundary condition can be used, which can be formulated as [11]

$$\frac{1}{\mu_{r2}} \boldsymbol{n} \times (\nabla \times \boldsymbol{E}) - \frac{\mathrm{j}k_0}{\eta} \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{E}) = \boldsymbol{0} , \tag{13}$$

or

$$\frac{1}{\varepsilon_{r2}} \boldsymbol{n} \times (\nabla \times \boldsymbol{H}) - \mathrm{j}k_0 \eta \, \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{H}) = \boldsymbol{0} , \tag{14}$$

where $\eta = \sqrt{\mu_{r1} / \varepsilon_{r1}}$ is the normalized intrinsic impedance of medium 1, which is equal to one in air, moreover $k_0 = \omega \sqrt{\varepsilon_0 \mu_0}$ is the wave number, $\varepsilon_{r2} = 1$, and $\mu_{r2} = 1$. Substituting η, k_0 and μ_{r2} into (13) results in

$$\boldsymbol{n} \times (\nabla \times \boldsymbol{E}) - j\omega \sqrt{\varepsilon_0 \mu_0} \cdot \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{E}) = \boldsymbol{0} , \tag{15}$$

where $\nabla \times \boldsymbol{E} = -j\omega\mu_0 \boldsymbol{H}$. After simplification, the absorbing boundary condition can be written as [2]

$$\sqrt{\frac{\mu_0}{\varepsilon_0}} \boldsymbol{n} \times \boldsymbol{H} + \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{E}) = \boldsymbol{0} . \tag{16}$$

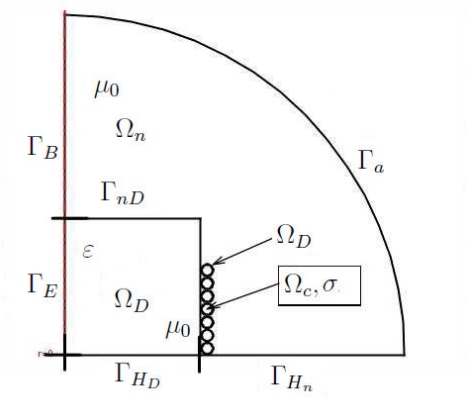


Figure 4. Structure of a wave propagation field problem

The absorbing boundary condition has been used on the artificial far boundary Γ_a. Substituting $\boldsymbol{H} = \nu_0 \nabla \times \boldsymbol{A}$ and $\boldsymbol{E} = -j\omega\boldsymbol{A}$ into (16) results in the used boundary condition,

$$-\nu_0 \boldsymbol{n} \times \nabla \times \boldsymbol{A} + \mathrm{j}\omega \sqrt{\frac{\varepsilon_0}{\mu_0}} \boldsymbol{n} \times (\boldsymbol{n} \times \boldsymbol{A}) = \boldsymbol{0}, \quad \text{on} \quad \Gamma_a . \tag{17}$$

It is important to note that the above partial differential equations and the absorbing boundary condition are valid only when the excitation is a sinusoidal current or a sinusoidal voltage.

The system of equation has been solved by using the weak form of the equations, which is the following:

$$\begin{aligned} &\int_{\Omega_n \cup \Omega_D} \left[\nu (\nabla \times \boldsymbol{W}_k) \cdot (\nabla \times \widetilde{\boldsymbol{A}}) - \omega^2 \varepsilon \widetilde{\boldsymbol{A}} \right] d\Omega + \\ &\int_{\Omega_c} \left[\nu (\nabla \times \boldsymbol{W}_k) \cdot (\nabla \times \widetilde{\boldsymbol{A}}) - j\omega\sigma \widetilde{\boldsymbol{A}} \boldsymbol{W}_k \right] d\Omega + \int_{\Gamma_a} j\omega \sqrt{\frac{\varepsilon_0}{\mu_0}} \boldsymbol{W}_k \widetilde{\boldsymbol{A}} \, d\Gamma = 0, \end{aligned} \tag{18}$$

where $k = 1, \cdots, J$, and $\widetilde{\boldsymbol{A}}$ is the function approximated the magnetic vector potential $\boldsymbol{A}$, moreover $\boldsymbol{W}$ is a weighting function.

3. Finite element model

While building up the model, the first problem was the complexity of the component. The largest problem was the cubical coil of the inductor, because in the COMSOL Multiphysics [12] the current flowing in the coil can be described by a mathematical formula, which can be determined from the equation of the circle in the case of a helical coil [5].

That is the reason why the shape of the core and the cubic coil were neglected and a two dimensional axial symmetry model has been created. The COMSOL Multiphysics software package can handle a three dimensional axial symmetry model, as a two dimensional axial symmetry model, so the built up model is equivalent to a three dimensional one. The procedure of the simplification can be seen in *Figure 5*.

3.1. Simplification of the model

The solution of the above problem provides the results of the unknown quantities via the computed potentials and the calculated integrals and expressions. This is the first chance to check the results and to execute modifications about the model. After the early simulations serious problems were discovered. There are too many finite elements, 59952 in the mesh, which cause 120085 unknowns in the simulation, that yields the simulation to be very slow. The solution time of the problem is 406 seconds with a computer having an Intel Pentium D 3.4 GHz processor with two cores and 4 GB RAM.

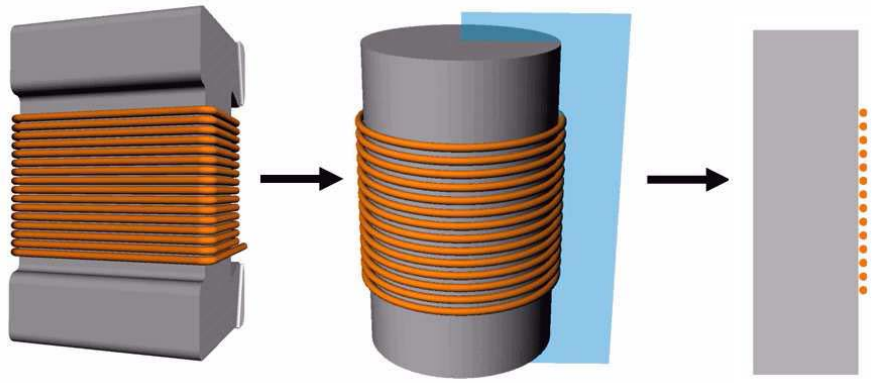


Figure 5. The procedure of the simplification

To decrease the processing time, the number of the mesh elements has been decreased through the removal of the enamel insulation of the winding wire. The mesh of the insulation effects high mesh elements, because it is not in the same order of magnitude with the whole model. The results show that the insulation of the wire can be neglected, because the results of the simulation are almost the same with and without the insulation. After the simplification the solution time decreased to 230 seconds.

Another problem was that, by using the two dimensional model, some attributes could not be simulated, i.e. the resistance of the terminations and the capacitance between the terminations. The winding wire is welded to the terminations where higher resistance has been appeared. Furthermore, the terminations have large surface, which causes some additional capacitance.

In the simulation the less resistance and the less capacitance cause higher self-resonant frequency and higher maximum value of the quality factor than the measured ones. To compensate these effects, an electric network model was created, wherein a capacitor and a resistor are in parallel with the simulated inductor to consider the higher

capacitance and the higher resistance. In *Figure 6* the applied electric network model can be seen [2].

The experiences show that the optimal value of the capacitance is 90 fF and the value of the resistance is 40 kΩ in the case of this type of inductor. The capacitance is marked by *C* and the resistance is symbolized by *R*. The network was built into the finite element model via the modification of the current passing through the component. The total current passing through the electric network can be determined by the following formula; henceforth it is used to calculate the impedance and other attributes [2],

$$I_m = I_{tot} + V_0 \mathrm{j}\omega C + \frac{V_0}{R_0}, \tag{19}$$

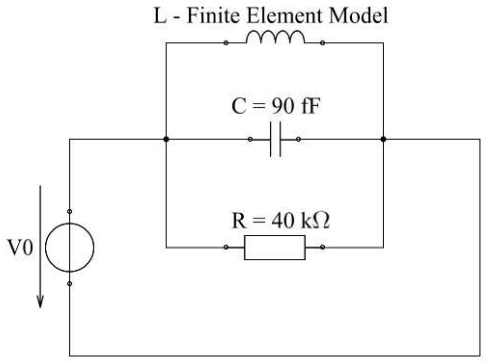


Figure 6. The applied network to consider the resistance and the capacitance on the terminations

where I_m is the modified current, I_{tot} is the total current of the finite element model and V_0 is the voltage of the network.

It is important to note that the components of the network model are only parameters.

4. Results of the simulations

After building up the finite element model, the computed DC resistance and DC inductance have been compared with analytic calculations and measured data to check the correctness of the model at low frequencies. The calculated resistance results in 0.463088 Ω by using Nagaoka's expression [2]. The DC resistance is 0.47 Ω, measured by the impedance analyzer. The computed DC resistance can be determined from the real part of the impedance, i.e. $R_{DC} = \mathrm{Re}\{\overline{Z}(\omega = 0)\}$, where $\overline{Z}(\omega = 0)$ is the value of the impedance in direct current case. The computed DC resistance results in 0.485 Ω.

The nominal value of the low frequency inductance is 180 nH of this type of inductor [1]. The L_a DC inductance of the component is 178.9 nH, by using the following analytical formula:

$$L_a = \frac{\mu_0 N^2 A}{l} K, \tag{20}$$

where N is the number of turns, A is the cross section of the wire, l is the length of the coil and K is a constant, which is changing by the function of the length and the cross section of the coil [2].

In the case of a specific inductor the measured inductance is 183 nH. The computed value of it in this case is 185 nH, which can be determined from the following equation [4]:

$$L = \frac{\mathrm{Im}\{\bar{Z}\}}{2\pi f}. \tag{21}$$

The obtained values are quite close to each other, so it is noticeable, that the created finite element model is working properly at low frequencies.

The comparison of the measured and the computed inductance between 10 MHz and 3 GHz can be seen in *Figure 7*, and the measured and the computed quality factor can be seen in *Figure 8*. It can be seen that the results are practically the same, so the finite element model is working properly at the whole range of the frequency. The difference between the measured and the computed quality factor, plotted in *Figure 8*, is probably caused by the simplification of the core.

At this point, the investigation of the modification of the winding to find the best geometry of the coil can be started. The aim is to find the maximal value of the quality factor. Several inductor models, with larger and smaller diameter of the wire, with closely- and widely-spaced coil, and with one and two layered coil were drawn to COMSOL Multiphysics [12]. In *Figure 9* finite element meshes of inductors can be seen with three different windings. During the examination, the finite element models were created and simulated and trial components were manufactured and measured with the same windings to compare the results.

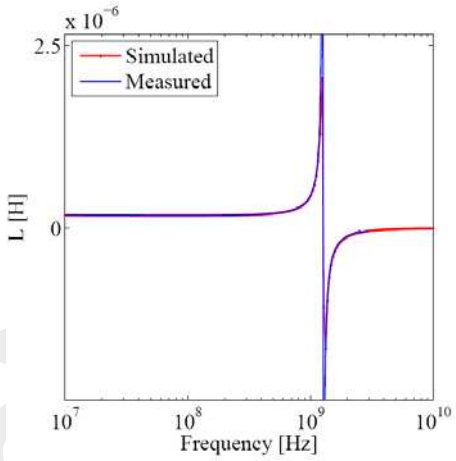


Figure 7. The measured and the computed inductance

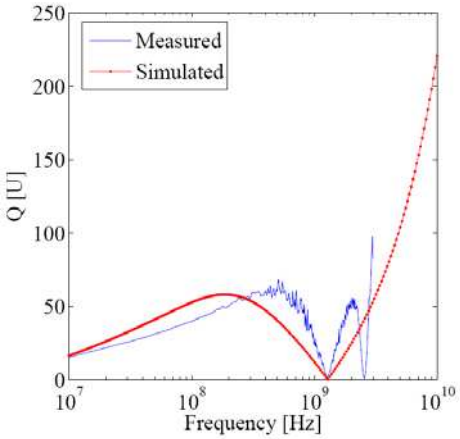


Figure 8. The measured and the computed quality factor

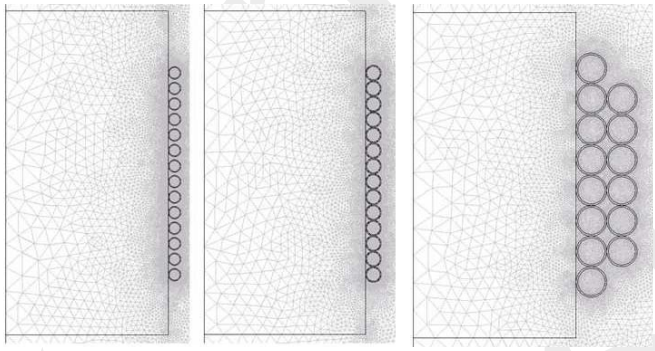


Figure 9. Finite element meshes of three different windings

The concrete experiments are the following about the increasing of the quality factor. It is trivial from the expression of the quality factor [4] that the value of it will increase if the imaginary part of the impedance increases or the real part of the impedance decreases. Because the nominal value of the inductance must be kept, the solution of the increasing of the *Q*-factor is the decreasing of the resistance.

The easiest way to decrease the resistance is using a wire with larger diameter in the coil. So a trial component was manufactured and a finite element model has been implemented with 60 μm diameter of the winding wire. The results in the simulation showed that the quality factor increase two or three percents in this case. Unfortunately, in the practice there are some problems. First of all, the measurements show that in the reality *Q* is increased slightly than the simulation shows, and only at lower frequencies in the studied range. At higher frequencies *Q* become smaller than in the case of the original wire, but it could not be a problem, because the quality factor has to increase at lower frequencies – between 85 MHz and 110 MHz. Furthermore, because of the manufacturing process the distance between the turns must be increased to eliminate the cross-windings, so using a wire with larger diameter and increasing the distance

between the turns caused that the value of the inductance is fallen to 170 nH. Because of the width of the winding cell, more turns to compensate the decreasing of the inductance cannot be used. Consequently, it is impossible to increase the quality factor by using thicker wire in the coil [2].

Then a wire with less diameter – it is 40 μm – has been tried out. By using thinner wire, the value of the inductance is increasing, so it can be enough to wind less turns to the core. Thus, there are more space to 'play game' with the wire. During the simulations and the experiments, it is cleared that the inductance is not increasing significantly to leave one or more turns. Therefore, 14 turns also must be used in this case. It is executable to spread the coil on the core, i.e. to increase the distance between the turns, to examine the effect of it. Our experiences show that the SRF is moved to higher frequency, through the less stray capacitance between the turns, and via it the maximum value of the *Q*-factor is also moved to higher frequency, moreover the maximum value of it is increased slightly. The rise of the quality factor is faster in this case, but unfortunately it starts in lower values than in the case of the original winding. Between 85 MHz and 110 MHz the *Q* of this trial component is lower than the *Q* of the present manufactured one. Consequently, the using of thinner wire in the coil is not the solution of the problem.

Another attempt was to manufacture the inductor with ferrite core. Because of the high relative permeability, the nominal value of the inductance can be achieved with less turns, effects the decreasing of the resistance of the coil. The examinations show that it is true, but the quality factor is not increased, moreover it is decreased significantly. The reason of this is the eddy currents and the hysteresis inside the ferrite core, which causes eddy current loss and hysteresis loss. These losses result the lower quality factor of a ferrite-cored component. Consequently, manufacturing inductors with ferrite core is not the solution of the problem of the *Q*-factor.

Finally, it can be said that thank to the experiences of the engineers, the presently manufactured component is nearly the best solution of the problem of the quality factor. So, the answer to the first question is that by the modification of the winding the quality factor cannot be increased significantly. But the question is hanging at poise: Is it possible to increase the quality factor, or not? The answer is yes, by the modification of the geometry and by using new materials.

5. Conclusions

The paper presents an actual problem of research engineers working with inductors and electronic components.

To solve several problems beyond the examination of the quality factor, a finite element model has been developed by using the COMSOL Multiphysics software package. The weak form of the potential formulations to solve the presented problems has been implemented from the Maxwell equations. The so-called absorbing boundary condition has been determined and has been applied to eliminate the effect of the reflected electromagnetic waves at the artificial far boundary.

The simulation of the simplified manufactured inductor has been done. To consider the capacitance and the resistance of the terminals, an electric network has been

implemented and the values of the parameters have been set to fit the computed results to the measurements. The built up finite element model has been tested. The measurements of the trial components have been executed and the results have been described and analyzed. The measured and the analyzed data have been compared with the results of the simulations. It is observable that the implemented model is working properly, the simulated attributes and the measurements are practically the same.

Consequently it can be said that by using the present materials and the present manufacturing technology, the quality factor cannot be increased significantly, as our experiences have shown. The increasing of the quality factor can only be realized by applying new materials and new geometries in the manufacturing.

The future aim of the research is to try out new materials in the manufacturing and to examine the effects of these materials to the quality factor. To execute this examinations a three dimensional, more accurate finite element model must be built up.

Acknowledgements

This paper was supported by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences (BO/00064/06), by Széchenyi István University (15-3210-02), by the EPCOS AG and the Hungarian Scientific Research Fund (OTKA PD 73242).

References

[1] www.epcos.com
[2] Pólik Z.: *Examination and development of RF inductors by using the finite element method*, Diploma Thesis, Széchenyi István University, Győr, 2008.
[3] www.agilent.com
[4] Pólik Z., Kuczmann M.: Examination and development of a radio frequency inductor, *Przeglad Elektrotechniczny,* Vol. 12, pp. 230–233, 2008.
[5] Kuczmann M., Iványi A.: *The finite element method in magnetics,* Akadémiai Kiadó, Budapest, 2008.
[6] Marcsa D., Kuczmann M.:, "Eddy Current Analysis With Non-Linearity", *Pollack Periodica* , pp. 97-109. Vol. 3, No. 2, 2008.
[7] Bíró O., Richter K. R.: CAD in electromagnetism, *in Series Advances in Electronics and Electron Physics,* Academic Press, New York, Vol. 82, 1991.
[8] Bíró O., Preis K., Richter K. R.: On the use of the magnetic vector potential in the nodal and edge finite element analysis of 3D magnetostatic problems, *IEEE Trans. on Magn*, Vol. 32, pp. 651–654, 1996.
[9] Fodor Gy.: *Electromagnetic fields* (in Hungarian), Műegyetemi Kiadó, 1996.
[10] Simonyi K., Zombory L.: *Theoretical electromagnetics*, (in Hungarian), Műszaki Könyvkiadó, Budapest, 2000.
[11] Jin J.-M.: *The finite element method in electromagnetics,* Wiley, New York, 2002.
[12] COMSOL:, *COMSOL Multiphysics User's Guide,* COMSOL AB, 2007.

Parallel Computations on the Blade Server at Széchenyi University – Classic Problems in Practice

L. Környei, G. Kallós, D. Fülep

"Széchenyi István" University, H-9026, Győr, Egyetem tér 1.
Phone: +36-96-503-400/3150
e-mail: kornyei@sze.hu, kallos@sze.hu, fulep@sze.hu

Abstract: Machines with huge computational power have become gradually affordable even for researchers of the universities in Hungary. In the summer of 2009 Széchenyi University also obtained a modern blade server. Using this machine we have solved several problems, these have given us a broad spectrum survey on the benefits and complications that come from performing concurrency in practice. After a short general introduction the machine of the university is presented in this paper (hardware, access, environment of execution), followed by the classic laws of multiprocessing systems. We present and examine several interesting problems afterwards. We show, that classic theorems are applicable in the case of problems with properly big computational- and properly low communication costs. Based on empiric argumentation we propose an amendment to Amdahl's laws to consider initialization time in multiprocessing systems. Additional effects of the multiprocessing environment are presented in the second part of the article.

Keywords: Supercomputers, MPI, PBS, Amdahl's law

1. Introduction, basic notions

1.1. Supercomputers

Some concepts in Informatics are interesting even for people with modest interest in this field, and some are even considered mystical. The "supercomputer" is definitely one, because of the high-tech applied or the extremely costly construction, for example.

The appearance of modern supercomputers is the result of a long development. Picking supercomputers from certain eras and comparing their computational powers, a clear picture can be drawn on this evolution. (Table 1., [1])[1]

[1] Here "Flops" is the abbreviation of floating point operation per second, which is a standard measure of computational power.

Table 1. Evolution of supercomputers

Computer's name	Delivery time	Max. computational power (Flops)
ENIAC	1946	500
IBM 704	1955	40 000
CDC 6600	1966	3 000 000
Cray 1	1976	250 000 000
Cray 2	1985	3 900 000 000
Intel ACSI Red	1997	1 300 000 000 000
IBM Roadrunner	2008-09	1 100 000 000 000 000

Behind this fascinating improvement there is indeed the most amazing development of the technology. The most substantial parts of it are the following:

- Miniaturization – there were more and more transistors integrated in each processor. For example, in case of PC-s the number of built-in transistors is doubled roughly every two years; from 30.000 (8088, 1979) it is increased up to approximately 300 million (Pentium D, 2005).
- Efficient solutions for multiprocessor architectures. The initial SIMD type machines were followed by MIMD type ones. Within MIMD, the first systems to handle relatively few processors were followed by advanced architectures (e.g. NUMA, hybrid systems), that had efficient memory access. Communication between units – e.g. among (a groups of) processors – were successfully accelerated to a great extent. Here we can discover a variety of ideas, from the theoretical investigation of network topologies to the development of infiniband connection[2].
- Changing over to multicore processors – several processor cores were integrated in each chip. Comparing with the traditional unicore solution, power and space consumptions were clearly reduced, making this solution more cost effective. Although the technology would allow the integration of even several hundred cores theoretically, the dominant manufacturers nowadays construct processors with 2, 3 or 4 cores. In the near future, gradual growing of the number of cores is expected.

The development trends briefly mentioned here are among the favorite topics in architecture curses in a more detailed fashion [2].

Generally, we can say that a supercomputer is a machine which is in the given era on the top of information technology. The top500 list is widely known, on which the best 500 supercomputers (according to computational power) of the world [3] are published. Following the rigorous interpretation (only) these are supercomputers. However, employing a less strict notion, nowadays machines of at least 1 Teraflops in computational power can be considered as supercomputers.

[2] We note here that a modern supercomputer is often a computer cluster, i.e. a group of tightly coupled computers which can be considered in many regards as one "big" computer.

1.2. Blade server at Széchenyi University

In the summer of 2009 Széchenyi University also obtained a modern blade server. Nowadays this is one of the most powerful engineering, scientific configurations in Hungary. Although we can not assert that this machine belongs to the most significant supercomputers on a word scale[3], on the basis of its computational power this machine can be considered as a real supercomputer.

This machine consists of 13 servers which are incorporated in a house of type HP c3000 [4]. One of the servers is prominent (the head node) its job is the control of the cluster, i.e. it runs the resource management software (PBS system, see below) which distributes the tasks appropriately and collects the results. The calculations are executed on the other servers (compute nodes). The compute nodes are positioned two by two in blade servers of type HP BL2×220c (6 pieces altogether) the blade containing the head node is of type BL260c, and the system includes also a container-blade of type SB40c which contains 6 fast 146 GB SAS disks.

The head node and the compute nodes equally contain 2-2 quad core Intel Xeon 5400 type processor with 3 GHz frequency; thus, the system altogether contains 26 processors and 104 cores. Because of the reasons mentioned above we can use only 96 cores for our computations (on compute nodes).

Figure 1. The blade server of Széchenyi University

The rapid connection among the servers (which is indispensable for effective calculations) is ensured beside the traditional Ethernet connection (1 Gbps) with quick infiniband switch (2×20 Gbps).

To compute nodes join 16 GB memory each, the total memory integrated into the system is 208 GB. The primary storages are the above mentioned SAS disks, besides

[3] According to the available former data it could have got into the top 500 list of the world's best supercomputers roughly 5-6 years ago.

this every compute node contains one 250 GB SATA drive (to store the temporary results of the computations), and the head node also includes even two of the same kind. Thus, the aggregate theoretical storage capacity is now 4.3 Terabytes, however, because of the safety technologies applied (mirroring, redundancy) we can use practically only about 3.9 Terabytes.

The cluster has theoretically 1,248 Teraflops computational power. Comparing this with the 4-70 Gigaflops power of a modern 2-4 core PC, we can conclude that the difference is roughly 17-300-times. Taking this fact into account, it can be really surprising for the external observer the relatively small dimension and power consumption of the machine.

1.3. Amdahl's law and its variants

G. M. Amdahl was the first, who published a general observation about the theoretical limits of the computations executed on multiprocessing architectures (1967, [5]) which remark was later named as Amdahl's law.

The assertion of Amdahl's law is the following: if an f part of a computation can be sped up parallel with a factor m, and the remaining part cannot be accelerated any more, then the non-acceleratable part will be quickly dominant in the total output, and any further improvement in the part f will have only little effect.

The same assertion using a formula:

$$\text{Speedup}_{\text{Amdahl}} = \frac{1}{1 - f + (f / m)} \tag{1}$$

where f is the parallelizable part and m is the number of processors.

If $m \to \infty$, then the we get for the speedup the following (ideal) upper bound:

$$\text{Speedup}_{\text{Amdahl}} \to \frac{1}{1 - f} \tag{2}$$

The data-amount does not appear directly in the assertion of Amdahl's law, because we have to execute a given problem i.e. a given number of calculation, sequentially and concurrently, respectively. Thus, the total amount of calculations does not change even in a multiprocessing system. We called this as a "fixed-size speedup" model.

If the fix workload to be performed is called by w, then the speedup factor can be written in the following form, too:

$$\text{Speedup}_{\text{FS}} = \frac{\text{sequential execution time for } w}{\text{parallel execution time for } w} \tag{3}$$

If the problem can be parallelized completely (or almost completely), then using m processors an m-times speedup is expectable.

With the appearance of machines with good computation power several deficiencies/limits of the Amdahl model came to light slowly. These limits derive from the fact that this model examines the execution of a fixed-size task. In many cases the

problem can be such type that it is not necessary or it is not possible to solve it completely. However, it is interesting even in these cases that in a given time with a given computational capacity how far we can get with the solution. Generally, not only the number of processors is important to estimate the parallel execution time, but we have to take into account even the possibilities of enlargement and effects of the workload (scalability).

Based on this realization, J. L. Gustafson published a new speedup formula in 1988 (fixed-time speedup model, [6]).

Let w be the workload which we can execute in a given time sequentially, i.e. with one processor. Let w' be the workload which we can execute in the same time using the concurrent architecture, with m processors. Then

$$\text{Speedup}_{\text{FT}} = \frac{\text{sequential execution time for } w'}{\text{sequential execution time for } w} = \frac{\text{sequential execution time for } w'}{\text{parallel execution time for } w'} \quad (4)$$

Expanding this formula we get the following result:

$$\text{Speedup}_{\text{FT}} = (1 - f) + mf. \quad (5)$$

Gustafson's law suggests that is it worthwhile constructing large parallel systems. The speedup grows proportionally with the size of the system, thus, for suitably large problems the system can be exploited well in fact.

With further development of architectures the deficiencies of this model became increasingly obtrusive. The theoretically possible expansion cannot be reached because of some physical constraints. Such can be e.g. the bound of memory-access which is described lately as the "memory-bounded speedup" model [7]. However, this model belongs not tightly to the scope of this paper; thus, here we omit the details.

There is a need of clarification here concerning arbitrary units of calculation. On the hardware level chips installed into the CPU-socket of the motherboard are commonly called processors. However, as we mentioned above, recent developments integrated several processing units into these chips, making each unit capable of executing code separately. These units are called processor-cores or simply cores. Later on we will use the notion processor and core as presented here. Also, by running on a certain number of cores we mean that we will start one process for each core.

1.4. What is our goal now?

The aim of this paper can be summarized in three points, as follows.

- The presentation of important notions and theoretical laws connected to the machines with large computational power (section 1, see above).
- The presentation of typical programming environment of these machines; and to overview the runtime system and methods for measuring the execution time, focusing on our blade server, specially (section 2.).
- The presentation of specific examples and measured results. In this part we analyze – taking up the blade server of the University to work – how the theoretical laws presented above are satisfied in the case of our examples (section 3 and second part).

The problems presented here were selected from the ones solved by us in such a way that those should be not only interesting, but also useful for the motivated reader, and should demonstrate the possibilities of real parallelization well. It was also important that the computational cost of the exercises be big enough to exploit the capacities of the cluster in all detail. Results were used here for verification foremost, besides this we dealt with effectiveness examinations too, but in this paper this respect was not the primary one.

It is important to underline yet that our programs are custom-built, so, this paper is not about the installation and execution experiences of various factory-made software.

Before we start to explain the exercises the runtime system and methods for measuring the execution time will be presented.

2. Programming environment

2.1. Message Passing Interface

Programming the multiprocessing systems needs a method greatly different from the traditional uniprocessor machines.

Surveying the history of parallel computing we can state – although the many benefits of this type of problem-solving were theoretically always clear comparing with the traditional, non-concurrent approach – that its real spreading and popularity was often significantly limited by the lack of adequate software support. This problem arose particularly sharply in the beginning of the 90s, by the time supercomputer prices had become more reasonable, and the smaller machines connected in networks – through their total computational power – become able to solve harder computational tasks, respectively.

At this time the demand appeared for such an efficient, portable programming recommendation with good expressive power which applies the popular message-passing model and fits well both types of parallel architectures mentioned above. The communication protocol called MPI (Message Passing Interface) was created which comprised luckily the good features of the existing systems and conformed to the goals set, so, it became very popular among the programmers of distributed systems soon, moreover, it turned into a kind of standard in fact.

The MPI implementations consist of collections of routines which support frequently environments in Fortran, C and C++ (recently even Java).

The two most successful "standards" are MPI 1.2 (often: MPI 1, 1994) and MPI 2.1 (often: MPI 2, 1996). MPI 1 supported basically a static programming environment contrary to MPI 2, in which new dynamic possibilities were built in (e.g. parallel I/O, dynamic process management), conforming to newer demand.

Major grouping of the MPI functions are as follows:

- Functions for point-to-point communication and data-exchange (between nodes);
- Synchronization functions;
- Functions which combine the results of partial computations.

A simple, "minimalist" MPI can be constructed even from a few functions (Table 2.).

Table 2. "Minimalist" MPI, based on language C, without detailing the parameters

MPI function	**Activity/Job**
MPI_Init	Initialization
MPI_Comm_size	Get number of ranks (tasks)
MPI_Comm_rank	Get own rank id
MPI_Send	Send a message
MPI_Recv	Receive a message
MPI_Bcast	Send a message to all at once
MPI_Reduce	Reduction operator
MPI_Finalize	End MPI

We will present a piece of code within the discussion of the first problem, which gives an example of using these functions.

For a further, more detailed introduction to MPI programming, see: [8] and [9].

2.2. Environment of execution on the blade server

PBS Professional (Portable Batch System) [10] is installed on the server, and is used to queue executions[4].

Due to reasons of security and efficiency, users can log in solely on the head node to issue execution on the cluster. It is possible to compile the code here, if needed, and to queue for execution via PBS. Our codes were compiled using MVAPICH2 version 1.2, which is an implementation of the MPI2 standard. Most MPI implementations use the command mpicc in place of the standard gcc. We compiled with the command:

```
mpicc -O2 -funroll-loops <source_code>.c -o <executable>
```

In order to run under PBS, it is advisable to use a script to include running parameters and the programs to execute. We include the script qn-pbs.sh for the N-queen-problem we will present next. The script is heavily simplified to help understanding, but will work, if used.

```
#!/bin/sh
#PBS -N nqueen                                  # job name
#PBS -m ae                                      # mail to user on abort and exit
#PBS -q workq                                   # working queue name
cd $PBS_O_WORKDIR                               # working directory
# $PBS_NODEFILE is the nodefile for this job
NNODES=`uniq $PBS_NODEFILE | wc -l`             # number of nodes
NPROCS=`wc -l < $PBS_NODEFILE`                  # number of processors
mpdboot -f $PBS_NODEFILE -n $NNODES             # creating MPI environment
./bench -error 0.01 mpiexec -np $NPROCS ./qn-mpi 20 | tee nq-$NPROCS.out ;
# "qn-mpi 20" will be executed on $NPROCS CPUs at least 3 times
# until the error of average execution time gets below 1%.
mpdallexit                                      # terminates MPI environment
```

[4] PBS is an efficient resource management software for cluster systems. It enables monitoring of resources and job queuing using preset rules in order to optimize utilization.

Please note, that `bench` is a custom developed code to measure wall-clock time of the total execution. The command `qn-mpi 20` will issue the actual calculation for board size 20.

To submit this job for execution allocating 4 resource units of 8 cores, 8 processes per unit (one per core) you need to invoke `qsub` as follows:

```
qsub –l select=4:ncpus=8:mpiprocs=8 ./qn-pbs.sh
```

Please refer to the PBS manual for a more extensive description. Resources can also be reserved within the submitted script; alas we used a wide combination of resources, which proved this method a bit clumsy. We used similar scripts to queue the programs of the additional problems we investigated, this makes presenting these superfluous.

2.3. Measurement method

To test the computational power of the blade cluster several custom-made programs were developed and wall-clock time cost of sequential and parallel solutions were measured. Parallel runs were submitted to typical configurations of 2 to 80 cores. Please note the denomination as follows: one node is the actual computer within the cluster.

As we mentioned previously, the blade cluster consists of a head node and 12 slave nodes. All nodes have two processor sockets on the motherboard, so we have a total of 26 processors. Every processor has four cores, which can work separately, but shares some resources, like cache, memory bandwidth and RAM. So this cluster is a 12+1 node, 24+2 processor, 96+8 core system. In this multiprocessing environment each core can handle several processes, though less efficiently.

So as we said we ran our program on 2 to 80 cores, this means we allocated a resource of some cores, and ran the same number of processes, one for each core. Alas, in this article the expression "running on *p* cores" or "running *p* processes" means the same.

In order to obtain reliable data, program execution was repeated several times, up to the point, where the relative error of the mean time cost was below 1%, but at least 3 times. The error of the mean, $\overline{x}$, is derived from the standard deviation of the measurements, σ, as follows:

$$\text{relative error of mean} = \frac{\sigma_{\overline{x}}}{\overline{x}} = \frac{\sigma}{\overline{x}\sqrt{n}} < 1\%,$$

where *n* is the number of executions.

The separate program bench was developed for this purpose. Wall-clock time is measured using `gettimeofday()`, and for even more precise data new processes are started with `vfork()`.

Benchmarks were made with all possible configurations from 2 to 8 cores within one node. Multinode measurements were utilizing all cores of the used nodes, so additional data on 16, 24, 32, 40, 48, 56, 64, 72 and 80 cores were obtained.

3. The N-queen-problem

The original problem comes from the regular chess board, and eight queen figures. The goal is to place all queens on the board, so none can strike the other. In general, you

must place N queens on a board of size $N \times N$. The original puzzle was proposed by chess player Max Bezzel in 1848, and was generalized by mathematicians later. The eight-queen problem has 40 different, but only 12 unique solutions. Using board symmetries on the 12, the remaining 28 can be reproduced.

Finding one solution for the problem can be achieved in linear time [11]. Counting and listing all possible configurations is still a hefty job, with exponential time cost. We set the later task as aim, whereby we put appropriate workload on the blade cluster, and grab the opportunity to present an example on parallel backtracking.

3.1. Sequential and parallel solutions

In 1972, Edsger Dijkstra used the very same problem to illustrate the power of what he called structured programming and the depth-first backtracking algorithm. A simple approach is to place the queens row by row. Having some placed, the next is set where it cannot strike any previous. If there are multiple possibilities, the first one is chosen. If there are none, the previous figure is picked up, and moved to the next valid position. If all the queens are placed, one configuration is found, and the next is sought with stepping back. The algorithm terminates, if there are no more valid positions left for the first queen to be placed. Some heuristics can haste this algorithm by some order of magnitude.

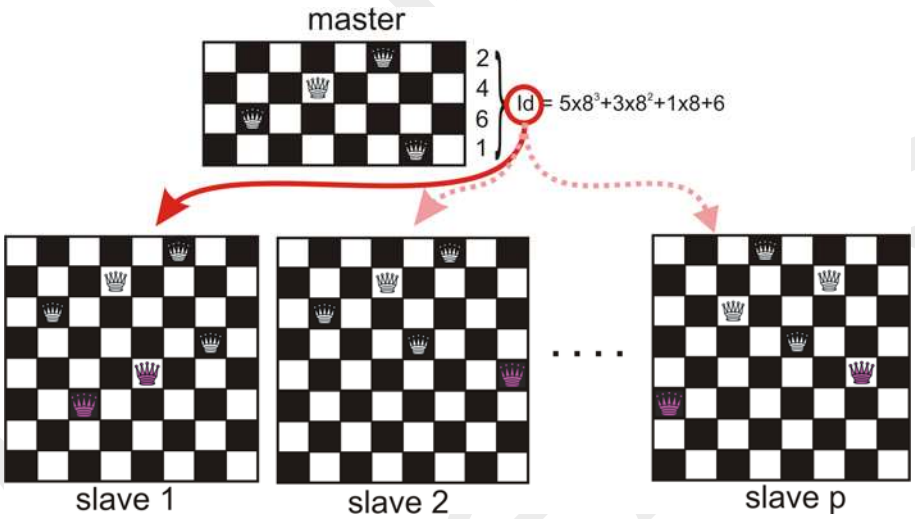


Figure 2. The master-slave setup. As a job is finished, the slave sends back the results to the master, which replies with the next viable configuration encoded in id*.*

Parallel implementation is the text-book example of the principle “divide and conquer”. After placing the first four queens, each viable branch will be walked by separate processes. Walk time for separate branches can differ greatly, so walking every p^{th} branch is not a viable option. A feasible solution is to have a master process, that will have the first four queens placed, have all viable solutions sent to slave processes, which will do the full walk (see Figure 2).

Let us look at the part implementing this message passing in C. The master process will execute:

```
typedef struct { int jno; long long ret ; } packet; // type of comm package
packet send, recv;                                  // sent and recv'd
MPI_Status status;

send.jno = jobids[1];                       // the first viable conf to send
for(i=1; i<jobs + pnos; i++)
{
  MPI_Recv(&recv, sizeof(packet),           // recv, from anyone
           MPI_CHAR, MPI_ANY_SOURCE,
           TAG_REQ, MPI_COMM_WORLD,
           &status);                        // sender rank gets in status

  MPI_Send(&send, sizeof(packet),           // next job id is sent to sender
           MPI_CHAR, status.MPI_SOURCE,
           TAG_ACK, MPI_COMM_WORLD);

  if(recv.jno) answers += recv.ret;         // sum up results
  if(i > jobs) send.jno = 0;                // if no more jobs left, send 0
  else         send.jno = jobids[i];        // else send next job id
}
print_result(n, answers);
```

The main loop of the slave processes:

```
while(1)
{
    MPI_Sendrecv(&send, sizeof(packet),     // send current result,
                 MPI_CHAR, 0, TAG_REQ,      // 0 first
                 &recv, sizeof(packet),     // we get next job id
                 MPI_CHAR, 0, TAG_ACK,
                 MPI_COMM_WORLD, &status);

    if(recv.jno==0) return;                 // if got 0, no jobs left
    send.jno  = recv.jno;                   // job id copied to send package
    send.ret  = solver(n, recv.jno);        // solve current subproblem
}
```

We use an implementation by Kenji et al. [12], results for $N = 24$ are published herein. The impact on computation power due to master-slave setup is tangible in benchmark results.

3.2. Wall-clock time cost of the implementation

Regarding the N-queen problem, board sizes from 8×8 to 20×20 were used. Execution times for enumerating valid configurations are shown on Figure 3. We can observe that all parallel runs take at least a couple of seconds to complete, although the sequential code can be as fast as a few tenths of a millisecond. The straight line for bigger board sizes implies exponential complexity of the algorithm as the scale of the vertical axis is logarithmic. Execution times for the sequential and the parallel code become comparable only at a board size of 15-16. The advantage of the 64-core system gets tangible only beyond board sizes of 17. Clearly, the master-slave setup will make only one process work, if two are started, which makes it comparable to the sequential code. Note that for the smallest board sizes execution time grows with the number of processes. This could be caused by a hefty initialization time of the MPI.

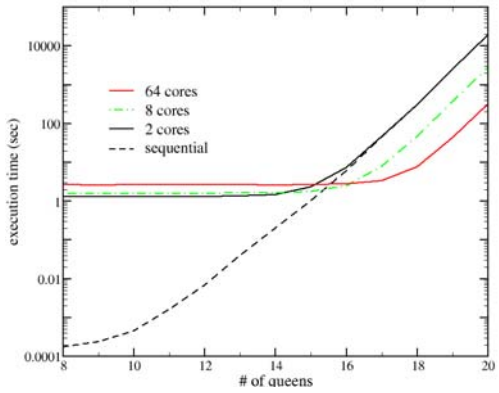


Figure 3. Execution time of the N-queen problem for board sizes N = 8…20 on a semi-log scale. The sequential and parallel runs for 2, 8 and 64 cores are shown. Dots are connected for better visuals.

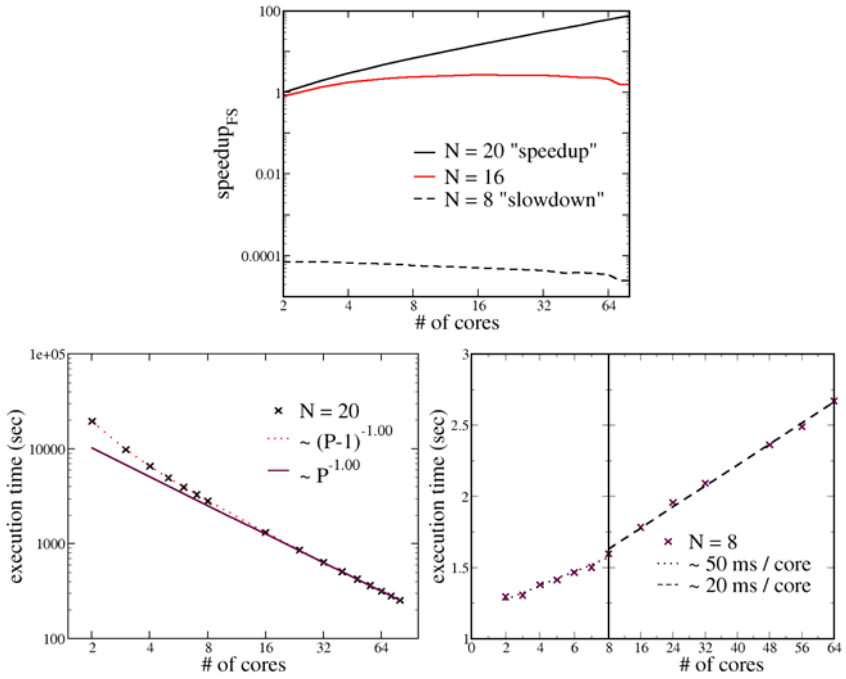


Figure 4. Fixed size speedup for board sizes 8, 16 and 20 are shown for various # of cores on log-log scale (above). The execution times for N = 20 including relevant fits on log-log scale (bottom, left). Execution times for N = 8 are shown including incremental costs for additional cores (below, right).

Figure 4 shows fixed size speedups that include slowdown effects, as well as execution times for the largest and smallest board size. The enormous gap between the "speedup"

and "slowdown" curves on the top plot comes from the fact, that while the sequential execution can be finished well below a millisecond, parallel execution needs at least a second. As table size grows, this initialization-like time becomes negligible, as observable on the plot for $N = 20$. This confirms an Amdahl-type argument: execution time is inversely proportional to the number of processes, p; actually as for the master-slave setup, to $p - 1$, which hints, that for a system big enough the problem can be fully parallelized – within the scope of the error.

As seen on Figure 3, execution time is independent of the board size below a critical value, and will increase with the number of processes started. This implies a kind of initialization time, which we measured at $N = 8$. Results and fits are shown on the lower right plot of Figure 4. We find that creating the MPI environment costs at least ~1.2 seconds, including ~50 ms/core for the first node, and ~20 ms/core for additional ones.

3.3. Time cost model

To formulate the fixed size speedup, we propose a model to describe time cost for the parallel algorithm depending on the number of processes, p, and board size, N. According to Amdahl's model sequential execution time can be divided into parts that can, and parts, that cannot be parallelized – let us denote these t_p and t_s accordingly. These can only depend on the board size, as the initialization time, t_i depends only on the number of processes. This yields the following form for the total execution time, t, for a master-slave type setup:

$$t(N,p) = t_s(N) + t_i(p) + \frac{t_p(N)}{p-1}. \tag{6}$$

Using the observation, that for N small enough, execution time becomes independent of board size, the form above will look like:

$$t(N,p) \approx t_i(p) \quad N << N_{\text{crit}}. \tag{7}$$

This makes t_i measureable indirectly, whereas t is measured directly. Using the following linearized form, t_p and t_s can be estimated, independently for all board sizes used:

$$(t(N,p) - t_i(p)) \cdot (p-1) = t_s(N) \cdot (p-1) + t_p(N) \tag{8}$$

Plotting the left side as a function of $(p - 1)$, also the validity of equation (6) can be verified, see Figure 5.

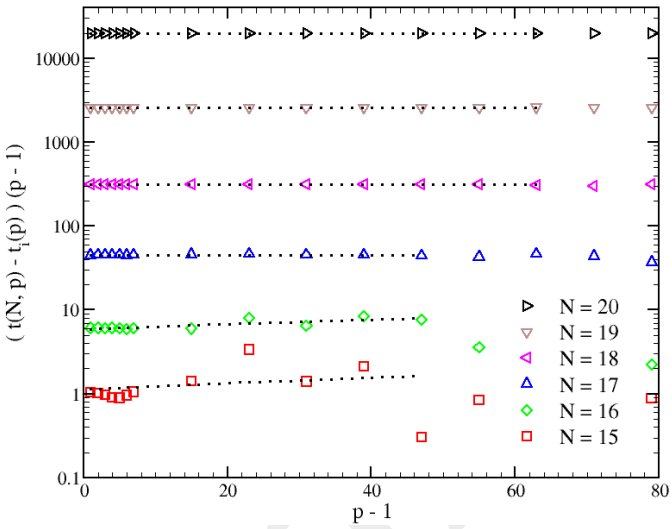


Figure 5. Linearization of eq. (6) respect to eq. (8), including linear fit. Multiplication with p – 1 amplifies errors, some data was excluded. Different symbols mean different board sizes from 15 to 20.

There are two important consequences of the figure. First, the cost function proposed in equation (6) holds valid. Also the linear slope obtained for t_s is not significantly different from 0. This is due to the high relative significance level for lower, and high absolute measurement error for higher board sizes.

Empirically there are two parts of the execution time. The first includes an initialization-type time cost that depends solely on the number of processes started, and increases linearly with them. The second is the classical parallel part that scales inversely with the number of slave processes. The first one dominates for smaller board sizes, and the second one for larger ones. Critical board size will depend on the number of processes as to be shown.

3.4. Fixed Size Speedup

Let us calculate the fixed size speedup according to the Amdahl's law (3). The numerator will hold the parallelizable part, $t_p(N)$, alone, the denominator will be $t(N, p)$ expanded with equation (6), and set $t_s = 0$:

$$\text{Speedup}_{\text{FS}} = \frac{t_p(N)}{t_i(p) + \dfrac{t_p(N)}{p-1}} = (p-1)\cdot\left(1 + (p-1)\frac{t_i(p)}{t_p(N)}\right)^{-1}. \qquad (9)$$

There are two dominant regions to simplify the formula. The speedup is linear, if the board size is big enough relative to the processes started, that is:

$$\text{Speedup}_{\text{FS}} = p-1, \qquad \text{if } t_p(N) >> (p-1)\cdot t_i(p). \qquad (10)$$

However, we cannot increase the number of slaves beyond any borders without severely crippling speedup, as:

$$\text{Speedup}_{\text{FS}} = \frac{t_p(N)}{t_i(p)}, \qquad \text{if } t_p(N) << (p-1)\cdot t_i(p)\cdot \tag{11}$$

Let us investigate the case, if the number of cores and processes is raised infinitely, as in the case of Amdahl's law (2). Here equation (9) will describe speedup validly. We can justly assume that $t_i(p) \to \infty$, as $p \to \infty$, that means

$$\text{Speedup}_{\text{FS}} \to 0, \qquad \text{as } p \to \infty. \tag{12}$$

This find can also be formulated as follows: If parallel execution time includes a member that increases limitlessly with the number of processes, fixed size speedup will converge to zero.

4. Discussion

In this article we studied the wall-clock execution time of the sequential and parallel implementation of the presented problem. Sequential time cost varied in a broad range from ~0.0002 up to ~20000 seconds, which are 8 orders of magnitude. The same range for a 64 core 8 node system is ~3 to ~300 seconds, these are just 2 orders. Two orders are won at the upper end, and four are lost at the lower one. Initialization time, held responsible, is investigated more thoroughly.

We make a proposition, that if the original problem is highly parallelizable in a master-slave setup, there will be two relevant parts of the execution time cost. One is responsible for the parallel environment setup, we called initialization time that will depend on only and increase with p, the number of processes started. The other for actual calculations will be inversely proportional to $p - 1$, the number of slave processes. Fixed size speedup will have a form of (9), also (10) and (11) will hold.

We also amend, that if the initialization time grows endlessly with the number of processes, the fixed size speedup will eventually decay to zero. Clearly this is the case, if any part of the time cost diverges with p.

There are many other limiting factors in a parallel computing environment; we will address some of these in the second part of our article.

The research is accomplished within the frame of TAMOP 4.2.2 – "Support for projects of basic and operational research of innovative research teams" program with support of the European Union and co-financing from the European Social Fund.

The Széchenyi István University as beneficiary won support of 291 939 640 HUF, which will be spent on the project entitled "Simulation and optimization basic research in numerical mathematics specifically for complex physical processes and production systems at the Széchenyi István University with a Newly Created International Research Team". The research is performed between 01.09.2009-30.08.2011.

References

[1] An up to date list of supercomputers: http://en.wikipedia.org/wiki/Supercomputers
[2] A. S. Tanenbaum, *Computer Networks* (4th ed.), Pearson Education Inc, New Jersey, 2003.
[3] List of the best supercomputers: http://www.top500.org
[4] Z. Horváth, B. Csábi, D. Fülep, Z. Kuti, *HPC Cluster Operation Guide*, Széchenyi István University, Győr, 2009.
[5] G. M. Amdahl, Validity of the Single-Processor Approach to Achieving Large Scale Computing Capabilities, Proc. of AFIPS Conference, 1967.
[6] J. L. Gustafson, *Reevaluating Amdahl's Law*, Communications of the ACM, 1988.
[7] X-H. Sun, Y. Chen, *Reevaluating Amdahl's Law in the Multicore Era*, J. Parallel Distrib. Comput., 2009.
[8] MPI tutorial: https://computing.llnl.gov/tutorials/mpi
[9] W. Gropp, E. Lusk, A. Skjellum, *Using MPI – Portable Parallel Programming with the Message Passing Interface*, Massachusetts Institute of Technology, 1999.
[10] PBS Professional: www.pbsworks.com
[11] R. Sosič, J. Gu, *3,000,000 Queens in Less Than One Minute*, SIGART Bulletin, Volume **2**, 2, 22-24 (1991)
[12] K. Kenji et al, *Solving the N-Queens Problem with a PG Cluster*, IEICE Transactions on Information and Systems, Volume **J87-D-1** 12 1145-1148 (2004)

Finite Element Modeling of Antenna Feeding

M. Kuczmann

"Széchenyi István" University, 9026 Győr, Egyetem tér 1.
Phone: 503 400, fax: 503 400
e-mail: kuczmann@sze.hu

Abstract: The most important measured parameters of an antenna are the input impedance and the radiation pattern. Other parameters, such as the reflection coefficient or the voltage standing-wave ratio can be calculated from the input impedance, the directivity as well as the gain can be obtained from the radiation pattern. The simulated input impedance is depending on the applied feed model that is why it is very important to know the advantages and the disadvantages of the feeding models. The most frequently used models are the current probe model, the voltage gap generator, the magnetic frill generator and the waveguide port. This paper presents the above mentioned approaches through a monopole antenna situated above a ground plane. The Finite Element Method (FEM) has been used in the numerical field analysis, which is a widely used technique to solve partial differential equations obtained from Maxwell's equations. Here, Helmholtz equation for the magnetic field intensity is studied in two dimensions supposing axial symmetry and in three dimensions modeling the complete geometry of the antenna. First, the problem and the corresponding equations are shown, and then the four feeding models are described. After the presentation of numerical results, a short discussion and summary close the paper.

Keywords: antenna feeding, antenna parameters, finite element method

1. Finite Element Method in Antenna Simulation

The Finite Element Method is a widely used numerical technique in computer aided design of electrical engineering problems. Only a brief introduction can be written here, the focus is only on the antennas and the corresponding equations. A detailed description can be found in [3-5].

The basis of the technique is the discretization of the problem region by simple finite elements. These finite elements are the triangle and the quadrangle in two dimensional problems, or the tetrahedral, hexahedral and prism elements in three dimensional problems. The system of equations to be solved for the potentials or for the field quantities can be assembled after obtaining the weak formulation of the partial differential equations and the boundary conditions of the problem. The latter equations can be derived from Maxwell's equations [3-5,7].

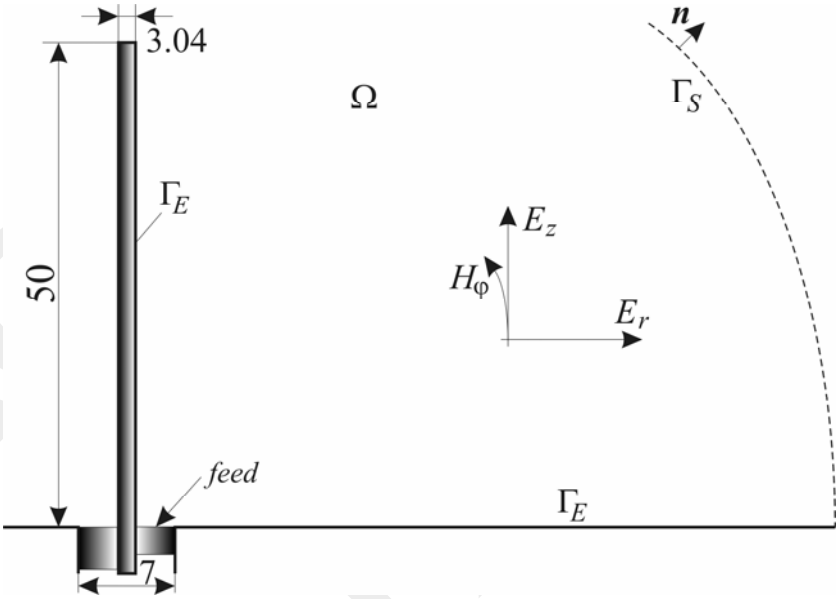


Figure 1. The geometry of the monopole antenna

The problem to be presented here is a monopole antenna situated above a ground plane [3]. The body of the antenna is the inner wire of a coaxial transmission line as it can be seen in Figure 1. The following Maxwell's equations must be solved in the domain Ω [7]:

$$\nabla \times \boldsymbol{H} = \mathrm{j}\omega\varepsilon\boldsymbol{E}\,, \tag{1}$$

$$\nabla \times \boldsymbol{E} = -\mathrm{j}\omega\mu\boldsymbol{H}\,, \tag{2}$$

$$\nabla \cdot \boldsymbol{H} = 0\,, \tag{3}$$

$$\nabla \cdot \boldsymbol{E} = 0\,, \tag{4}$$

where $\boldsymbol{H}$, $\boldsymbol{E}$, ω, ε, and μ are the magnetic field intensity and the electric field intensity, the angular frequency of excitation, the permittivity and the permeability, respectively. The phasor representation has been used, because of the time-harmonic excitation (the generator is supposed to be sinusoidal), i.e. $\mathrm{j}^2 = -1$ is the imaginary unit.

It is well known that the electromagnetic field of the monopole antenna is transverse magnetic (TM) [3,4,7], or in other words, the magnetic field has only one component in the φ-direction, and the electric field has two orthogonal components, as it is denoted in Figure 1.

The electric field intensity must be normal to the surface of the ground plane and the surface of the antenna, i.e. the boundary condition

$$\boldsymbol{E} \times \boldsymbol{n} = \boldsymbol{0} \tag{5}$$

can be supposed on Γ_E, and $\boldsymbol{n}$ is the outer normal unit vector.

On Γ_S, absorbing boundary condition must be prescribed to absorb the electromagnetic energy [3,4],

$$\lim_{r\to\infty} r\left[\nabla\times\boldsymbol{H}+\mathrm{j}k_0\,\boldsymbol{n}\times\boldsymbol{H}\right]=\boldsymbol{0}\,,\ \text{on}\ \ \Gamma_S\,, \tag{6}$$

which can be approximated by the first order absorbing boundary condition

$$\boldsymbol{n}\times\left[\nabla\times\boldsymbol{H}+\mathrm{j}k_0\,\boldsymbol{n}\times\boldsymbol{H}\right]=\boldsymbol{0}\,,\ \text{on}\ \ \Gamma_S\,, \tag{7}$$

where $k_0=\omega\sqrt{\mu\varepsilon}$ is the wavenumber in free space ($\mu=\mu_0$, $\varepsilon=\varepsilon_0$). This models the unbounded space. The calculation domain must be truncated somehow, because the discretization can not be performed at infinity, and the condition (7) on Γ_S can be used to decrease the domain volume. The efficiency of absorbing the electromagnetic energy along the boundary Γ_S can be increased by applying a perfectly matched layer (PML) which outer boundary has been assigned as the absorbing boundary [3].

Finally, $\boldsymbol{H}\times\boldsymbol{n}=\boldsymbol{0}$ must be satisfied along symmetry planes (along the line $r=0$ in axial symmetry situations).

It is evident that the application of the magnetic field intensity as the primary variable results in the most economic formulation. The partial differential equation to be solved for the magnetic field intensity here is the following [3,7]:

$$\nabla\times\nabla\times\boldsymbol{H}-k_0^2\boldsymbol{H}=\boldsymbol{0}\,, \tag{8}$$

and $\boldsymbol{E}=\left(\nabla\times\boldsymbol{H}\right)/\mathrm{j}\omega\varepsilon$ is the electric field intensity from (1) and (2). After some mathematical manipulations and using (3), the following partial differential equation can be obtained for H_φ:

$$\Delta H_\varphi+k_0^2H_\varphi=0\,, \tag{9}$$

which is a scalar Helmholtz equation of the magnetic field intensity.

2. Feeding Models in FEM

The feeding models of antennas are applied to take the input of the antenna into account. The most widely used feeding models are shown in Figure 2 [1-3,6].

2.1. The current probe model

The most widely used current probe model is a short current with a delta function, e.g.

$$\boldsymbol{J}(x,y,z)=\boldsymbol{e}_z\,I_0\,\delta(x-x_f,y-y_f)\,,\quad 0\le z\le d\,. \tag{10}$$

It models a wire with zero diameter, x_f and y_f are the coordinates of the current I_0 ($x_f=0$, and $y_f=0$ in Figure 2. a), and $\boldsymbol{J}$ has only one component in the z direction. This infinitesimal dipole can be generalized in any direction of the space. The electromagnetic field is singular in the vicinity of the probe [3]. This is the reason why

it is more convenient to prescribe the magnetic field intensity on the surface of the antenna wire as it is represented in Figure 2. a. The φ-component of the magnetic field intensity can be calculated by

$$H_{\varphi} = \frac{I_0}{2a\pi}, \quad 0 \le z \le d, \tag{11}$$

and $a = 1.52$ mm is the radius of the antenna. The length of the probe in the z direction should be as small as possible, but it can be concluded that $l << \lambda$ must be specified, and λ is the wavelength of the electromagnetic wave in vacuum, $\lambda = c/f$ (c is the speed of light and f is the frequency of excitation). Here $d = 1.6$ mm has been used.

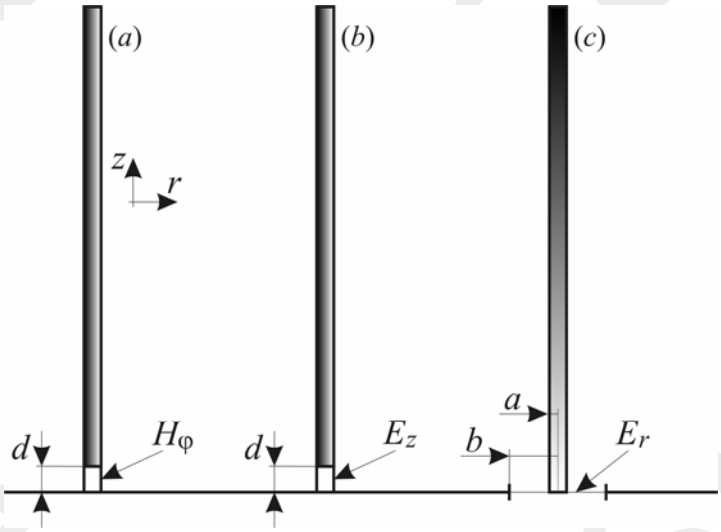


Figure 2. Feeding models

Once the electric field ***E*** is determined by the applied numerical method, the voltage across the probe can be computed as

$$U = -\int_0^d E_z(\mathrm{r} = \mathrm{a})\,\mathrm{d}z, \tag{12}$$

and the input impedance of the antenna is

$$Z = \frac{U}{I_0}. \tag{13}$$

The current distribution along the antenna can be calculated by the following form of Ampere's law:

$$I(z) = 2a\pi H_{\varphi}(z). \tag{14}$$

2.2. The voltage gap generator

This model is basically used in the Method of Moments (MoM) plane as it is presented in Figure 2. b. The z component of the electric field can be obtained by

$$E_z = -\frac{U_0}{d} \tag{15}$$

in the gap ($0 \le z \le d$), and d the length of the gap. The electric field intensity is prescribed along the line $r = a$, and $0 \le z \le d$ (see in Figure 2. b).

The current in the feeding point then can be calculated by (14) substituting $z = 0$, i.e.

$$I = 2a\pi H_\varphi(z = 0), \tag{16}$$

then the input impedance can be obtained by

$$Z = \frac{U_0}{I}. \tag{17}$$

The current distribution along the antenna can be simulated by (14).

2.3. The magnetic frill generator

The magnetic frill generator is a model of the antenna input fed by a coaxial line [1]. The following electric field intensity can be supposed in the radial direction by assuming purely TEM mode inside the coaxial transmission line (see in Figure 2. c.):

$$E_r(r) = \frac{U_0}{2r\ln(b/a)}, \quad a \le r \le b, \tag{18}$$

where a and b are the inner and outer radius of the coaxial line ($a = 1.52\,\text{mm}$ and $b = 3.5\text{mm}$) and U_0 is the potential difference between the inner wire and the outer shielding. The current distribution along the antenna can also be simulated by (14), and the input impedance can be calculated in the same way as presented in section 2.2.

2.4. The waveguide port

The waveguide port model is more accurate and more efficient approach in general case. This is based on the weighted sum of TEM, TE and TM waveguide modes, and the weighting coefficients are collected in tables [3]. This model has been implemented in Comsol Multiphysics [8].

The scattering parameter (reflection coefficient) S_{11} can be extracted from the simulated electric field, finally, the input impedance can be obtained as [3,8]

$$Z = Z_0 \frac{1 + S_{11}}{1 - S_{11}}, \tag{19}$$

where $Z_0 = 50\Omega$ is the characteristic impedance of the waveguide.

3. Simulation Results

The problem has been solved by the functions of the Radio Frequency module of Comsol Multiphysics [8]. This software is a very efficient FEM design environment. The above mentioned feeding models can be implemented and tried out in an easy way. The models can be downloaded from the Author's homepage [9].

First, the φ-component of the magnetic field intensity has been simulated by the TM Electromagnetic Waves application mode, and two dimensional axial symmetry geometry has been analyzed for simplicity, because the aim is the study of the different models. Second order Lagrange shape functions have been used to approximate the unknown field quantity. Second, the whole 3D problem has been analyzed aiming to compare the two dimensional and the three dimensional results. In 3D vector shape functions have been used [3-5].

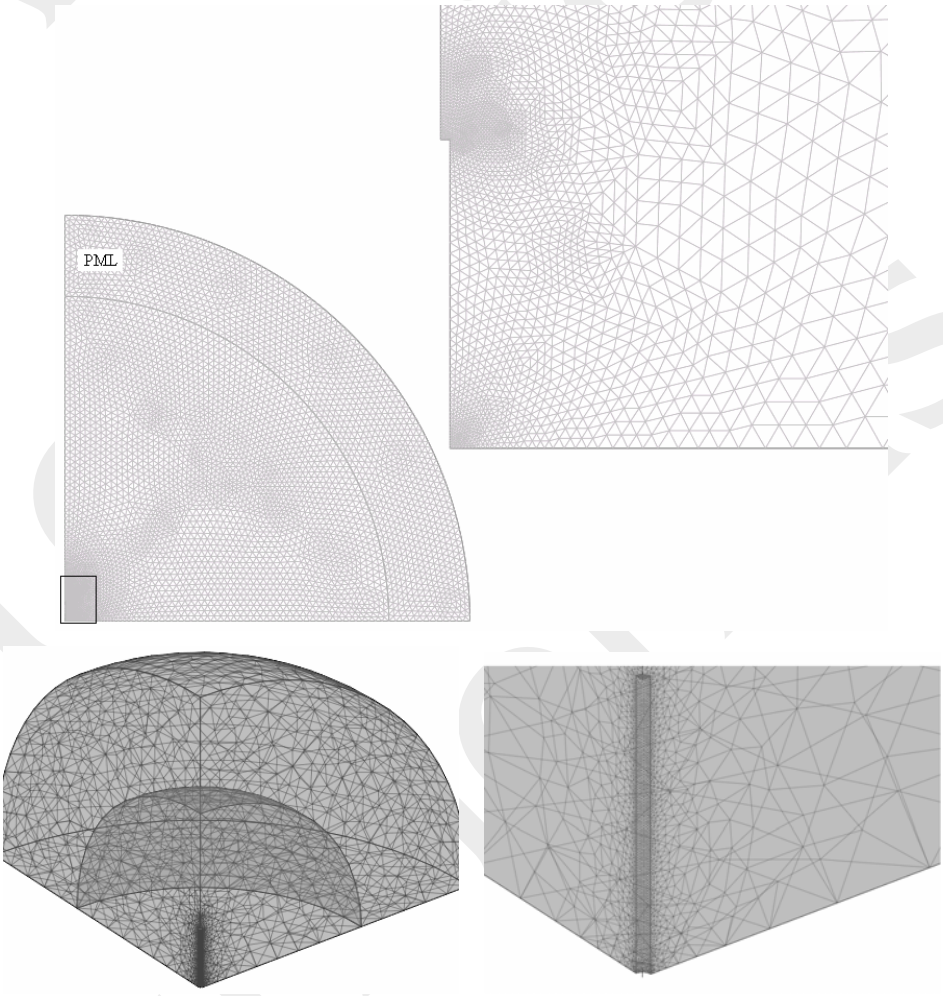


Figure 3. FEM mesh in 2D and in 3D, the vicinity of antenna is magnified

After some trials, 55296 triangles have been used to mesh the two dimensional geometry, and it results in 111329 unknowns. In 3D, 36682 tetrahedra have been generated, which results in 247814 unknowns. This is a very dense mesh and it can be seen in Figure 3.

The convergence of the simulated input impedance can be seen in Figure 4, where the measured impedance is also shown. Measured data are from [2]. The variation of input impedance is practically the same when applying finer and finer mesh. There is a permanent difference between measured and simulated data.

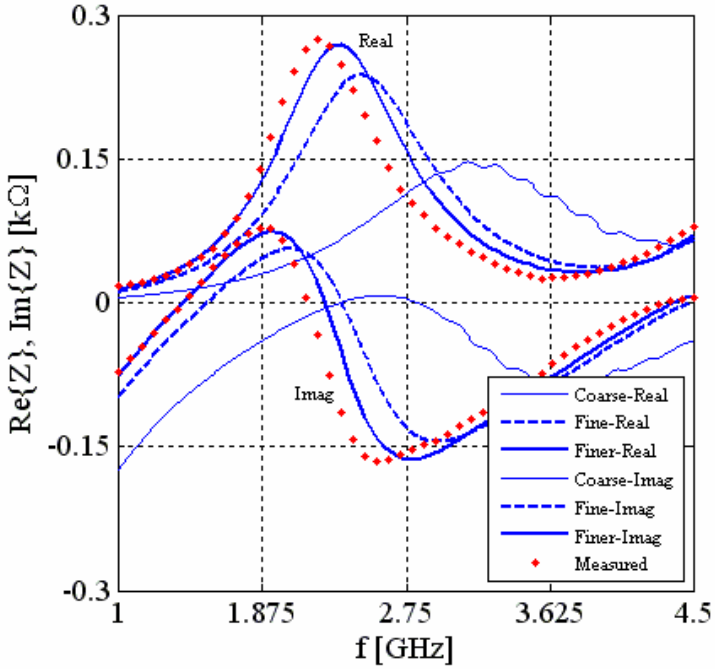


Figure 4. Convergence of the solution vs. number of triangles

The geometry of the antenna has been subtracted from the calculation domain, because it is supposed to be made of ideal conducting material, i.e. discretization is not necessary inside the wire. The same mesh has been used in all the frequency during the frequency sweep in the range of 1GHz and 4.5GHz. A PML layer [3,8] has been inserted to improve the absorption of electromagnetic field, and the radius of the computational domain is 1m in the two dimensional case and it is 0.25 m in the three dimensional case.

Figure 5 shows a comparison between measured input impedance and simulated ones. The application of current probe model results in the weakest approximation, the approximated value obtained from the other models are practically the same. The results from the 2D and 3D simulations are the same, i.e. the mentioned feeding models can be used in any three dimensional situations.

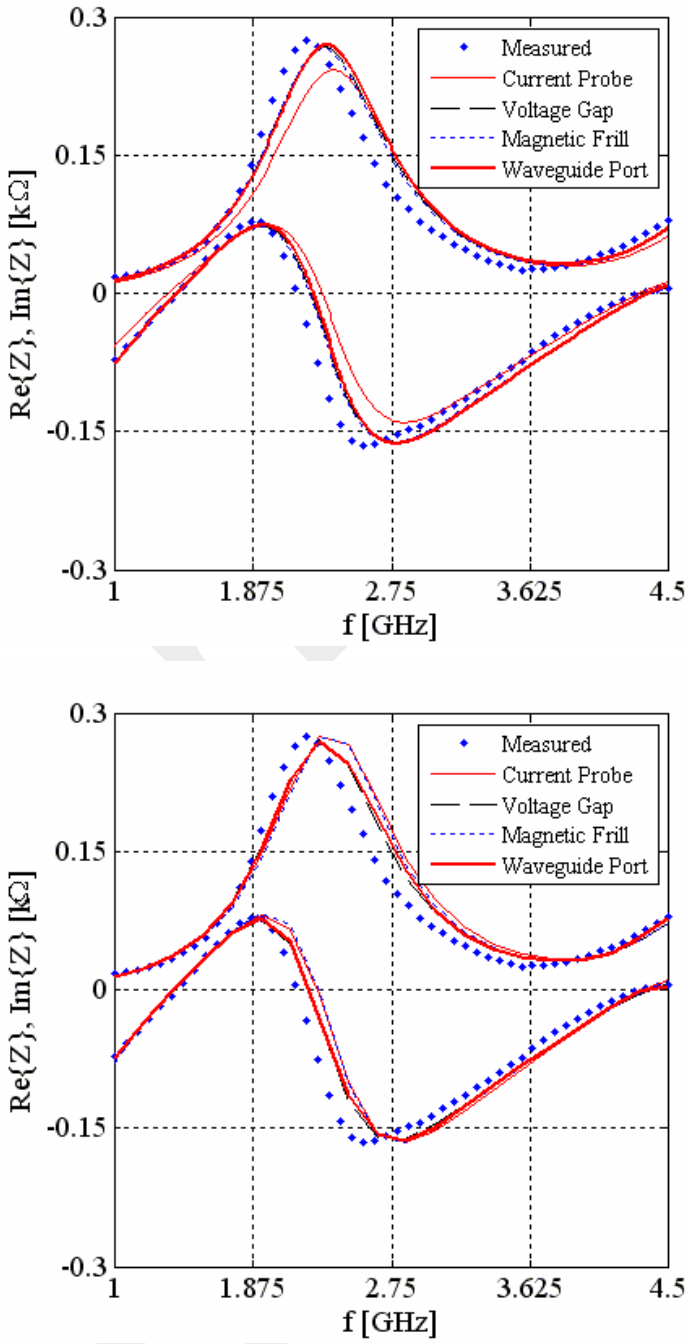


Figure 5. Comparison of the input impedance of the monopole antenna (up–2D simulations, down–3D simulations)

The current distribution along the antenna is a very important input data to calculate other quantities. A comparison between the obtained currents simulated by the above mentioned feeding models can be seen in Figure 6 at the frequencies $f = 1.5\text{GHz}$, $f = 3\text{GHz}$, and $f = 4.5\text{GHz}$. The results are practically the same, but a small difference can be seen in the vicinity of $z = 0$ (the feeding point), and it is the effect of the different feeding models.

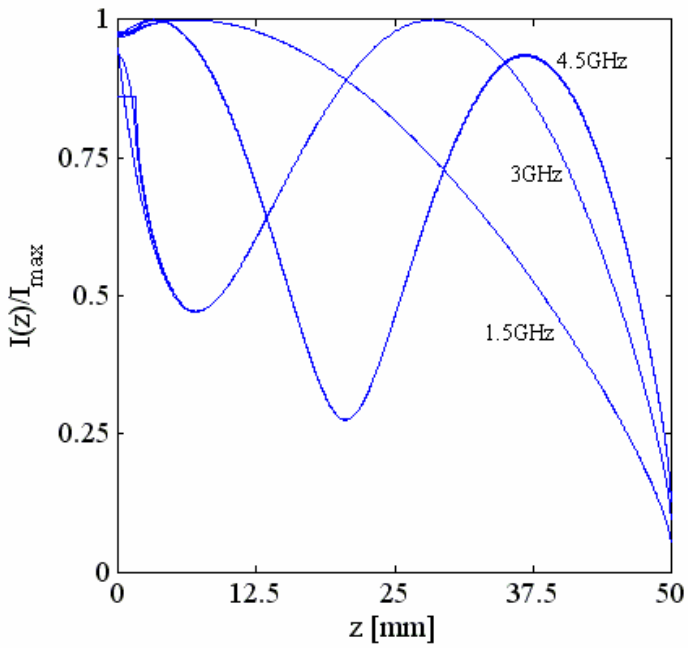


Figure 6. Normalized current distribution along the antenna at three different frequencies

Figure 7 shows a very spectacular three dimensional result. The variation of the electric field intensity is presented as normalized vectors on the plane $z = 0$. The magnetic field intensity is also shown in the figure as a slice on the plane $x = 0$.

4. Summary

Feeding models of antennas have been presented in the frame of FEM. The input impedance, the current distribution of a monopole antenna on a ground plane and the variation of electromagnetic field quantities around the antenna have been simulated and compared with measured data.

The next step of the research work is to apply the feeding models in the case of more complex antennas in three dimensional situations, and to compare the results with other numerical techniques, e.g. with MoM.

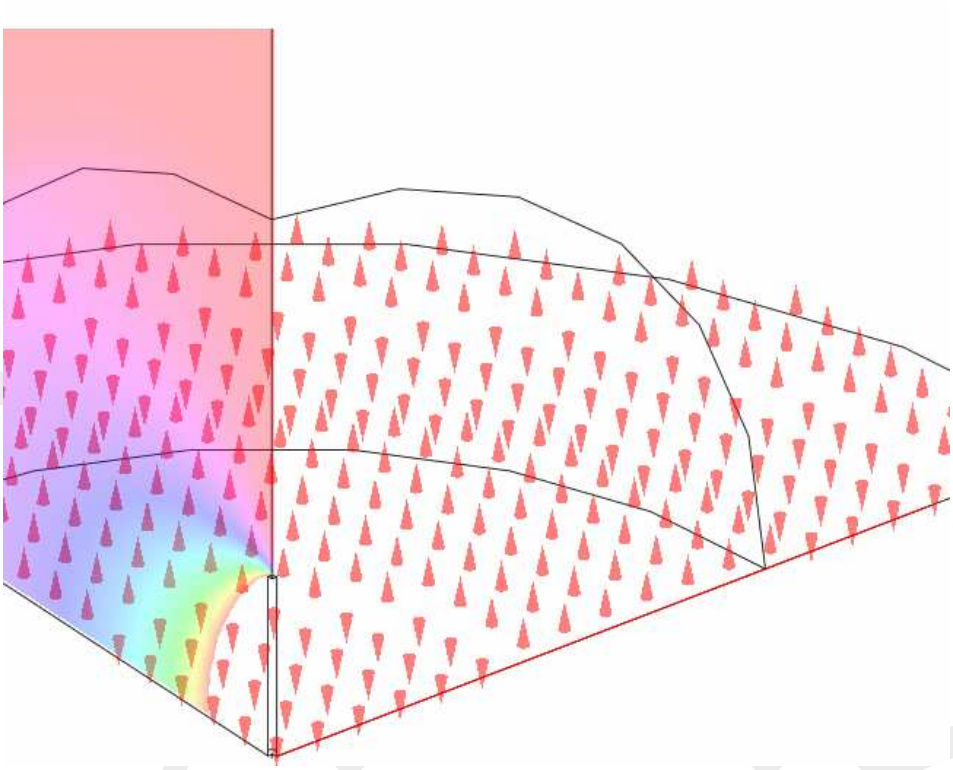


Figure 7. The magnetic field intensity and the electric field intensity around the antenna

Irodalomjegyzék

[1] Gibson, W.C.: *The Method of Moments in Electromagnetics*, Chapman & Hall/RCR (2008).

[2] Hertel, T., Smith, G.: *On the convergence of common FDTD feed models for antennas,* IEEE Trans. Antennas Propag., Vol. 51, No. 8, (2003), pp. 1771–1779.

[3] Jin, J-M., Riley, D.J.: *Finite Element Analysis of Antennas and Arrays*, Wiley, IEEE Press, (2009).

[4] Jin, J-M.: *The Finite Element Method in Electromagnetics*, Wiley, IEEE Press, (2002).

[5] Kuczmann, M., Iványi, A.: *The Finite Element Method in Magnetics*, Academic Press, Budapest (2008).

[6] Lou, Z., Jin, J-M.: *Modeling and Simulation of Broad-Band Antennas Using the Time-Domain Finite Element Method,* IEEE Trans. Antennas Propag., Vol. 53, No. 12, (2005), pp. 4099–4110.

[7] Simonyi, K., Zombory, L.: *Theoretical electromagnetics*, Műszaki Könyvkiadó, Budapest (2000).

[8] http://www.comsol.com

[9] http://maxwell.sze.hu/~kuczmann/Korszeru_antenna/Antenna_lap.htm