Leszek KASPRZYK
Poznań University of Technology

Optimization of Lighting Systems with the use of the Parallelized Genetic Algorithm on Multi-Core Processors using the .NET Technology

***Abstract**. The paper presents possible parallelization of the optimization process of complex lighting systems with the use of the genetic algorithm. The features of modern personal computers and the tools enabling distribution of the computation process among multi-core processors are depicted. The duration of a test task computed on the machines provided with the Intel processors of P4, i5 and i7 types with the use of the tools available in the .NET environment has been investigated.*

***Streszczenie.** W artykule przedstawiono przykładowe metody skrócenia czasu realizacji zagadnienia optymalizacyjnego wykorzystującego algorytm genetyczny, za pomocą rozproszenia obliczeń na procesorach wielordzeniowych, stosowanych w komputerach klasy PC. Podczas przygotowywania aplikacji obliczeniowej wykorzystano wbudowane mechanizmy udostępnione w środowisku .NET. (**Optymalizacja systemów oświetleniowych za pomocą zrównoleglonego algorytmu genetycznego na procesorach wielordzeniowych z wykorzystaniem technologii .NET**).*

Keywords: Optimization, Genetic Algorithm, Parallel Computation, Multi-Core Processors
Słowa kluczowe: opotymalizacja, algorytm genetyczny, obliczenia równoległe, procesory wielordzeniowe

Introduction

Due to growing normative restrictions, and economical and ecological reasons, the modern electric devices and equipment are expected to be characterized with higher effectiveness and durability and, at the same time, lower manufacturing and operation cost. This may be possible in effect, among others, of improved accuracy of the designing calculation. The accuracy of the calculation may be better in result of consideration of the phenomena occurring in the systems subject to modeling, e.g. by increasing of the number of discrete elements used for modeling the considered objects. Unfortunately, such an approach considerably enlarges duration of the computation [2,3].

The duration may be additionally increased in case of searching the best solution with the use of various optimization methods. The problem becomes particularly important in case of large tasks related to electromagnetism, that require solving large systems of equations [5,6]. Complexity of the phenomena arising in the considered objects imposes the need of using a discretization mesh of large number of elements. Otherwise, the results may considerably deviate from real values. As an example of such a problem the analysis of electromagnetic field in visual range, i.e. analysis of the lighting field, may be mentioned [2]. Such an analysis is highly time-consuming and the optimization calculation for a complex lighting object may last hours. In case of small number of the discrete elements the results may deviate from the true values by hundreds percent and, in consequence, may give wrong optimization output [4].

Nevertheless, it should be noticed that from several years the personal computers are provided with the processors composed of several cores that enable multi-thread operation (the Hyper-Threading technology has been implemented for the first time in 2002 in the Xeon Foster MP processors). Even so most of the applications, even the ones designed for highly complex calculation, do not make use of all the processor cores. The PC computers that are recently so popular are provided with 4 cores based on the HT technology. This causes that the operating system sees 8 computation units, i.e. so-called virtual processors (modern processors, e.g. Intel Core i7-980X, are provided with 6 HT cores, i.e. 12 virtual processors). Moreover, the analysts forecast that modern personal computer processors will be soon provided with 64 cores. The existing operating systems are able to balance operation of such computation units, but only in case of many processes running simultaneously on the computer. In case of a single application the operating systems cannot distribute the computation among several computer threads. This is due to the fact that the applications are formulated in a single-thread way and, in consequence, only a small fraction of the computation power of modern PC computers is used.

Hyper Threading Programming in the .NET Environment

The single threaded applications are developed mainly because until recent time the multi threaded programs required very good knowledge of the programming language and distributed programming. Designers of modern environments serving for developing the computer applications make the task easier and provide the libraries that enable distributing the computation among many cores. One of the example environments serving for writing multi-threaded applications is Microsoft Visual C# 2010 .NET. It includes the Task Parallel Library with the functions for distributing the calculation on the multi-core machines. It makes use of the solutions that enable using the multi-core architecture of the processor for paralleling the computation procedures. They allow for effective use of the power of modern processors. The programming interference into the process of distribution of the computation tasks among particular cores and their threads enables more effective management of the power of the processing units than in case of automatic distribution of the tasks managed by the operating system.

The most popular technique of the use of the multi-core processors consists in managing many threads (the Thread class), tasks (the Tasks class), and control of their course with the use of the so-called semaphores (the Semaphor class). Temporary locking of selected object availability (the lock mechanisms) enables operation of the algorithms with the thread synchronization. The 100 percent distribution efficiency is not achievable. Similarly as in massively parallel computers the reduced effectiveness is due to time delay necessary for creating these structures in the operating memory. In case of the tasks the thread sequence of which is not important, the Parallel class provides the parallelized versions of the *For()* and *ForEach()* loops that uniformly charge all the processor cores, precluding the control of the sequence of the iterations.

The use of the presented mechanisms enables such distribution of the tasks that uniformly charge all the processor cores during the calculations. The effectiveness of the processor usage attained this way may be many times better than for a single-threaded application [2]. Effectiveness of the calculation distribution depends on the algorithm and the type of the mechanisms used in the program that are made available in the .NET. environment.

Optimization of Complex Lighting Systems

Optimization of the lighting systems is aimed at reducing the manufacturing costs and the cost of electric power consumption used for the lighting. At the same time, the normative requirements should be met. The optimization process includes as well optimal selection of the lighting equipment, optimal selection and location of the light sources so as to maintain uniform lighting of required intensity. The lighting systems in large objects are of complex character and include K separate subareas. Every part of the system is characterized by a set of the lighting parameters that describe the lighting system structure. These parameters make a set of the decision variables. They include the type of the lighting fitting and reflector, the type of the light source, the ignition system, the parameters related to location and number of the fittings. In case of K separate subareas the absolute value of the vector x of independent variables amounts to 14K and in real systems reaches dozens. The proposed form of the objective function is given by the expression [4]:

(1)

$$J(\mathbf{x}) = r \cdot k_{ass}\left[\sum_{i=1}^{K}\sum_{m=1}^{M_i}(C_{fit_{m_i}} + w_{mi} \cdot C_{src_{m_i}} + C_{fix_{m_i}})\right] + $$
$$+\sum_{i=1}^{K}\sum_{m=1}^{M_i}\left[P_{fit_{m_i}} \cdot T_w \cdot C^*_{pow} \cdot Y + C^*_{main_{mi}} \cdot Y + w_{mi} \cdot (C^*_{utilsrc_{mi}} + C^*_{srcexch_{mi}}) \cdot \frac{T_w \cdot Y}{T_{src_{mi}}}\right]$$

where: k_{ass} – a factor of the assembling cost, M – the number of fittings, $_{mi}$ – index of the m-th fitting belonging to the i-th lighting subsystem ($i=1,2,...,K;$ $m=1,2,M_i$), P_{fitmi} – power of the mi-th fitting [kW], T_w – operating time within one calendar year [h], T_{srcmi} – life of the source located in the mi-th fitting [h], w_{mi} – number of the sources of the mi-th fitting, C_{pow} – the price of 1kWh of electric power [€],C_{mainmi} – yearly maintenance cost of the mi-th fitting [€], C_{fitmi} – price of the mi-th fitting [€], C_{srcmi} – price of the source in the mi-th fitting [€], C_{fixmi} – price of the fixtures of the mi-th fitting [€], $C_{srcexchmi}$ - price of exchange of the sources of the mi-th fitting, Y – life-time of the lighting system [years], r – the rate of extended reproduction; * - taking into account the inflation factor. Most of the decisive variables occur inexplicitly in the objective function – they affect values of other parameters.

Once the objective function is determined, the optimization conditions (mainly of normative character) should be checked. The luminous flux distribution in the considered object should be determined for this purpose. It may be computed with the use of the Maxwell equations, with some simplifications. It is assumed that the considered interior is a part of the Ω space surrounded by a concave surface S that restrains the light field. The S surface characterized with the reflection coefficient ρ reflects a part Φ' of the luminous flux incident on it directly from the source. In result of many reflections of the luminous flux from particular surfaces the total luminous flux Φ incident on the surface equals the sum of the direct Φ' and indirect Φ'' components:

(2)

$$\Phi_i = \Phi'_i + \int_S f_{ij}\rho_i\Phi_j dS$$

where: Φ_i - the total luminous flux on the i-th element Φ'_{ij} - the luminous flux incident on the i-th element and coming from the j-th element; Φ_j - the luminous flux incident on the j-th element; ρ_i reflection coefficient of the i-th surface element; f_{ij} – coupling coefficient of the j-th and i-th elements.

Consideration of the coupling coefficient [5,6], the effect of all the surface elements [5,6], and possible concave surfaces allows to formulate the system of linear equations with total light fluxes on planar elements dSi as unknown variables [5,6]:

(3)

$$\begin{bmatrix} 1+\rho_1\frac{S_{10}}{S_1}f_{1N} & -\frac{S_{20}}{S_2}(\rho_2 f_{21}-\rho_2 f_{2N}) & .. & -\rho_N\frac{S_{N0}}{S_N}f_{N1}-1 \\ -\frac{S_{10}}{S_1}(\rho_1 f_{12}-\rho_1 f_{1N}) & 1+\rho_2\frac{S_{20}}{S_2}f_{2N} & .. & -\rho_N\frac{S_{N0}}{S_N}f_{N2}-1 \\ ... & ... & .. & ... \\ 1-\rho_1\frac{S_{10}}{S_1} & 1-\rho_2\frac{S_{20}}{S_2} & .. & 1-\rho_N\frac{S_{N0}}{S_N} \end{bmatrix}\begin{bmatrix}\Phi_1 \\ \Phi_2 \\ ... \\ \Phi_N\end{bmatrix} = \begin{bmatrix}\Phi'_1-\Phi'_N \\ \Phi'_2-\Phi'_N \\ ... \\ \Phi_{src}\end{bmatrix}$$

where: i, j – indexes of surface elements; ρ_i – reflection coefficient of the i-th surface element; f_{ji} – the usage factor of the i-th and j-th element dS_i, Φ_{scr} – the total luminous flux of the light sources of the analyzed object, S_{i0} - the surface closing the i-th concave element, S_i – total surface area of the i-th concave element.

Example of the Computation

Example optimization calculation has been carried out for an industrial object including two subareas of different lighting requirements. Total length 70m (40m and 30m), width 25m and the height varying between 5m and 6m. There are no partitions that could douse the light flux between the subareas. The normative requirements are determined in accordance with the European Standard EN 12464-1:2002. In order to find an optimal solution a set of the lighting fittings and light sources has been selected that meets the conditions related to the fitting types, light source types, the power of a single lighting point, etc. It was assumed for purposes of the optimization that the lifetime of the lighting system amounts to 10 years.

Taking into account the type of the problem and the form of the proposed objective function the random method of genetic algorithm has been chosen for optimization of the complex lighting systems. The set of deterministic operations belonging to the genetic algorithm leads to good results, even in case of the tasks that are difficult to solve with deterministic methods [1]. The size of the generation used in this case amounted to 60 individuals, the crossing probability 0.7, and the mutation probability 0.001. The optimal solution has been found after 50 generations.

The optimization calculation has been distributed making use of the characteristic feature of the genetic algorithms, that consist in many times repeated determination of the adaptation function. In order to check correctness of the conditions the light flux must be determined. This operation is the most time-consuming during the optimization process of the lighting systems of high complexity. The calculation for each of the individuals of the generation is carried out independently that may result in remarkable shortening of the computation time.

The so developed algorithm has been implemented in Microsoft Visual C# 2010 environment, with the use of the mechanisms available in the Task Parallel Library (the .NET

4.0 Platform). The test computation has been carried out with three PC computers. The first one of them was provided with Intel Pentium 4 processor, supporting the HT Technology (2 threads), of the timing rate 2.8 GHz, 1024 kB CACHE and 4 GB RAM of DDR1 type. The second one had the Intel Core i5-2300 processor composed of 4 cores (4 threads), with processor timing rate of 2.8 GHz, 6 MB CACHE and 8 GB RAM of DDR3-1066/1333 type. The third one was provided with Intel Core i7-2600 processor composed of 4 cores according to HT technology (8 threads), processor timing rate 3.4 GHz, 8 MB CACHE and 8 GB RAM of DDR3-1066/1333 type.

Results of the Computation

The test included analysis of the duration of the applications designed for optimization of a complex lighting system that was run on both the above mentioned personal computers. The computation was four times repeated – with sequential and parallel algorithms, with the use of the *Parallel.For()* loops, threads (the Thread Class) and with tasks (the Tasks Class). The results are presented in Fig. 1.

Moreover, effectiveness of the parallelization has been assessed with respect to the size of the calculation task. In this case equal tasks of various sizes (with N size factors) have been solved. Durations of these tasks enabled to define the speed-up effect of the parallel computation, determined by comparison to the duration of the same operation performed with the help of the sequential algorithm. The calculation has been made on the i7-2600 processor, with the use of the tasks distribution mechanism (the Tasks Class) (Fig. 2).

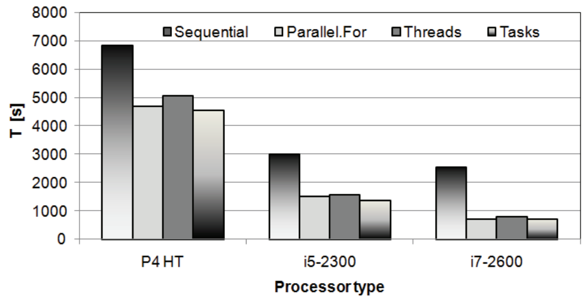

Fig. 1. Duration of the optimization process with respect to the processor type and parallelization method

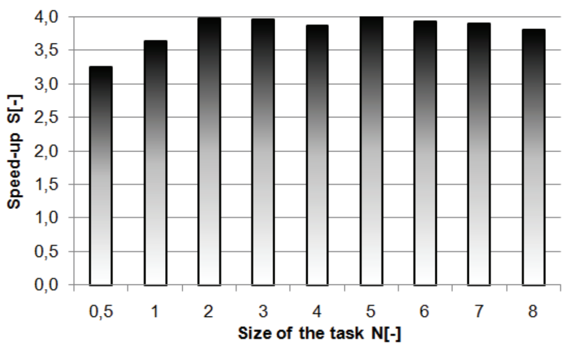

Fig. 2. Speed up of the process with respect to the size of the calculation task (with the use of Tasks, realized on the i7-2600 processor)

Summary

The preliminary tests show that distribution of the calculation among particular processor cores and threads may considerably reduce duration of the task solution to be run on personal computer. The presented above optimization example of complex lighting systems gave 1.5-, 2.2-, and 3.6-fold calculation speed-up in case of the process carried out on P4 HT, i5-2300, and i7-2600 processors, respectively. It was shown, that the size of the task results in about 20 percent effect on the speed-up. This gives evidence of correct application of the algorithm and good scaling ability of the cores used for this purpose.

The above allows to state that application of the mechanisms available in the TPL Library of the .NET Platform enables effective use of all the processor cores and provides effective method of reduction of the numerical calculation time. In result, such an approach allows to improve accuracy of the numerical calculation and, in consequence, better consideration of electromagnetic effect exerted on electric devices and equipment. This, in turn, enables more precise consideration of their interaction and allows to develop optimal systems, taking into account the technological and economical reasons.

It should be noticed that a some of the functions available in the .NET environment and serving for computation distribution require no detailed knowledge of distributed programming. This opens possibility for many users (programmers) of desktop computers, for whom the massively parallel computers remain unavailable, to parallelize their calculations and considerably reduce the numerical analysis duration. Moreover, the existing single-threaded applications may be easily modified to multi-threaded ones.

Various speed-up results obtained with the help of particular distribution methods are due to application of different parallelization methods. The best results have been obtained for the presented example while distributing the computation with the use of the Tasks class. It was due to modern mechanisms (added in Framework 4.0) that enable more effective sequencing of the tasks, without context triggering. This reduces memory demand. Computation distribution with the help of threads appeared the worst. It is caused by time delay necessary for preparing dynamical structures in the operating memory each time when the thread is created.

REFERENCES

[1] M. Corcione, L. Fontana, Optimal design of outdoor lighting systems by genetic algorithms, *Lighting Research and Technology*, Vol. 35, No 3, pp. 261-280, 2003

[2] K. Bednarek, L. Kasprzyk, A. Tomczewski, Efficiency and economic aspects in electromagnetic and optimization calculations of electrical systems, *Electrical Review*, ISSN 0033-2097, Vol. 86, No 12, pp. 57-60, 2010

[3] K. Bednarek, L. Kasprzyk, Speeding up of electromagnetic and optimization calculations by the use of the parallel algorithms, *Electrical Review* ISSN 0033-2097, Vol. 85, No 12, pp. 57-60, 2009

[4] L. Kasprzyk, R. Nawrowski, A. Tomczewski: Optimization of Complex Lighting Systems in Interiors with use of Genetic Algorithm and Elements of Paralleling of the Computation Process, Intelligent Computer Techniques in Applied Electromagnetics, Vol. 119, Wiak Slawomir, Krawczyk Andrzej, Dolezel Ivo (Eds.), *Springer-Verlang* Berlin Heidelberg New York, pp. 13-18, 2008

[5] L. Kasprzyk, R. Nawrowski, A. Tomczewski, Analysis of a Light Field with the use of Parallel Computers, *Compumag'2007*, Aachen, pp. 919-920, 2007

[6] Kasprzyk L., Nawrowski R., Tomczewski A.: Application of a Parallel Virtual Machine for the Analysis of a Luminous Field, LNCS, Vol. 2474, *Springer-Verlang* Berlin Heilderberg New York, pp. 122-129, 2002

***Author**: dr inż. Leszek Kasprzyk, Politechnika Poznańska, Instytut Elektrotechniki i Elektroniki Przemysłowej, ul. Piotrowo 3a, 60-965 Poznań, e-mail: Leszek.Kasprzyk@put.poznan.pl*