SEVENTH FRAMEWORK PROGRAMME

"Ideas" Specific programme

European Research Council

Grant agreement for Advanced Grant

Annex I - "Description of Work

Project acronym: HEPGAME
Project full title: Solving High Energy Physics Equations using Monte Carlo Gaming Techniques
Grant agreement no.: 320651
Duration: 60 months

Date of preparation of Annex I (latest version): Januari 10,2013

Principle Investigator: J.A.M. (Jos) Vermaseren
Host Institution: Nikhef Amsterdam (Stichting FOM)

Solving High Energy Physics Equations using Monte Carlo Gaming Techniques

HEPGAME

- Name of the Principal Investigator (PI): J.A.M.Vermaseren
- Name of the PI's host institution for the project: Nikhef, Amsterdam
- Proposal full title: Solving High Energy Physics Equations using Monte Carlo Gaming Techniques
- Proposal short name: HEPGAME
- Proposal duration in months: 60

The main objective of this proposal is to perform (hitherto unsolved) calculations in Quantum Field Theory (QFT) most of which are urgently needed to make optimal use of upcoming experimental data from the Large Hadron Collider. These specific calculations have been intractable thus far due to their enormous demand of man and computer power. We will make use of the brand new technique of Monte Carlo Tree Search (MCTS) from the fields of Artificial Intelligence (AI) and gaming to resolve this issue and automatize the derivation of formulas and the construction of computer programs. To do so, we will first develop MCTS into a viable QFT tool. Calculations and derivation of the formulas will be done by the (open source) computer algebra system FORM developed by the PI. A spinoff of the proposal will be the adaptation and extension of FORM to allow the physics and the AI to work well together. We will make the new technology available for other researchers, enabling a wide range of calculations at a new level of precision.

Section 1a: Curriculum Vitae

Name Dr. Jos (Jozef Antoon Maria) Vermaseren

Current Position

Staff member at the National Institute for Subatomic Physics (Nikhef), Amsterdam, The Netherlands

Contact information

Nikhef, Science Park 105, 1098 XG Amsterdam.
E-mail: t68@nikhef.nl, URL: www.nikhef.nl/∼t68/

Date of Birth 02-07-1949 (Roermond, The Netherlands)

Education

1977: PhD Theoretical Physics (SUNY Stony Brook, New York, USA)
1973: MSc Theoretical Physics (University of Nijmegen, The Netherlands)

Employment

1981-now: Staff member, Nikhef, Amsterdam
1979-1981: Researcher at the European Organization for Nuclear Research (CERN), Geneva, Switzerland
1977-1979: Researcher at Purdue University, USA

Awards

2007 Humboldt Research Award of the Alexander von Humboldt Foundation, in recognition of *his outstanding long term contributions to precision calculations in Quantum Chromodynamics (QCD), notably on the scaling violations of the nucleon structure functions.*

Publication record

55 publications (according to Web of Science), nearly 4000 citations, H-index 29.
79 publications in the Spires database with more than 7000 citations, H-index 40. In addition the talk about FORM (in arXiv.org math-ph/0010025) has more than 760 citations).

Research achievements

- **Particle Physics:** My research focusses on the phenomenology of particle physics. I am specialized in doing calculations that are much needed but very difficult to do so that few people will even attempt them. These involve for instance the evaluation of the four loop beta function in QCD (1996-1997) and the complete three loop anomalous dimensions and coefficient functions for QCD in unpolarized deep inelastic scattering (1995-2004) which were needed for an accurate determination of the quark and gluon distributions inside the proton.

- **Symbolic Manipulation System FORM:** To be able to do extreme calculations I have constructed a unique computer algebra program: FORM[1] that has been widely recognized as the fastest available and that can handle by far the largest formulas. People have used it with intermediate formulas of more than one Terabyte. It is freely available and used in nearly all big calculations in Quantum Field Theory (QFT). Developing this program further and keeping it suitable for the latest techniques and technologies goes hand in hand with doing calculations in the above category. Version 4.0 of FORM will have many new and original

[1] See http://www.nikhef.nl/∼form/

features as a preparation for future calculations. It will be released as an open-source program early 2012.

- **Mathematics:** I have additional expertise in the area of mathematics, because the answers of my calculations may involve functions that either have not been properly defined yet, or whose properties are insufficiently known. Examples are papers on group theory factors (1999), harmonic sums (1999), harmonic polylogarithms (2000) and multiple zeta values (2007-2010) which are also used by mathematicians.

- **QCD three-loop anomalous dimensions**: My research on the physics aspects of the calculation of the unpolarized three loop QCD anomalous dimensions in deep inelastic scattering has attracted much attention. My two 2004 papers on this topic were selected by the journal Nuclear Physics as their top papers of that year (see fig. 1). The 2007 Humboldt Research Award also honoured my achievements in this area.

- **Gaming Techniques**: This proposal joins high-energy physics with techniques from the gaming world. In this respect it is relevant to mention my achievements in the Chinese-Japanese board game *Go*. I have achieved the so-called "4 dan" rank, designating a high level of expertise, and reached fourth place in the 1983 Dutch championships and twelfth in the 1985 European championships.

Other professional activities

Member of the Board of the Foundation CAN (Computer Algebra Nederland) since its start in 1989. In the period of 1995 to 2000 chairman with the task to reorganize CAN. It has been split into a successful bussiness (candiensten), an expertise center that was taken over by the university of Amsterdam, and a research center that was taken over by the university of Eindhoven. The stichting (foundation) still exists, but has a reduced level of activities.

International activities

I have been visiting researcher at various institutes. Highlights (at least 3 months) include:
- CERN (Geneva): 3 mo in 1990, 3 mo in 1992,
- KEK (Tsukuba, Japan): 7 mo in 1991/1992, 3 mo in 2004/2005, 3 mo in 2005,
- UAM (Madrid, Spain): regular visits (1994-2012),
- DESY/Zeuthen: 3 mo in 2007, 3 mo in 2009,
- Universität Karlsruhe (Germany): 3 mo in 2008.

Funding ID

The PI is currently involved in two funded research projects. One is a program grant by the Dutch Foundation for Fundamental Research on Matter (FOM). This is a proposal by 10 senior physicists in the Netherlands with the subject "Theoretical Particle Physics in the Era of the LHC" and runs from 2008 to 2013. The total amount of money involved is 2.4 MEuro. The other funding is from the EU Initial Training Network. The project is called LHCPhenoNet and runs from 2011 till 2015. For the Dutch node this involves 36 months ESR, 12 months ER and 4 months visitor money. The research topic is "Advanced Particle Phenomenology" at the LHC in Geneva.

The PI is also involved in the application for funding from the EU Initial Training Network under the name HiggsTools. Here the application for the Dutch node is for 36 months ESR to work on Higgs and top quark related physics at the LHC in collaboration with experimentalists.

There is and there will be no funding overlap with the ERC grant requested and any other source of funding for the same activities and costs that are foreseen in this project.

Section 1b: 10-Year Track-Record

i: Top 10 publications in the past 10 years as senior author.

1. *The Four loop beta function in quantum chromodynamics.* By T. van Ritbergen, J.A.M. Vermaseren, S.A. Larin. [hep-ph/9701390]. Phys.Lett. B400 (1997) 379-384. Cited 555(11) times in the Spires database.
2. *New features of FORM. By J.A.M. Vermaseren.* [math-ph/0010025]. Cited 768(26) times in the Spires database.
3. *The Three loop splitting functions in QCD: The Nonsinglet case.* By S. Moch, J.A.M. Vermaseren, A. Vogt. [hep-ph/0403192]. Nucl.Phys. B688 (2004) 101-134. Cited 396(23) times in the Spires database. Selected by Nuclear Physics as one of their two top papers in 2004 (2004 advertising flyer of NP).
4. *The Three-loop splitting functions in QCD: The Singlet case.* By A. Vogt, S. Moch, J.A.M. Vermaseren. [hep-ph/0404111]. Nucl.Phys. B691 (2004) 129-181. Cited 487(24) times in the Spires database. Selected by Nuclear Physics as one of their two top papers in 2004 (2004 advertising flyer of NP).
5. *The Longitudinal structure function at the third order.* By S. Moch, J.A.M. Vermaseren, A. Vogt. [hep-ph/0411112]. Phys.Lett. B606 (2005) 123-129. Cited 88(12) times in the Spires database.
6. *The Third-order QCD corrections to deep-inelastic scattering by photon exchange.* By J.A.M. Vermaseren, A. Vogt, S. Moch. [hep-ph/0504242]. Nucl.Phys. B724 (2005) 3-182. Cited 170(18) times in the Spires database.
7. *Higher-order corrections in threshold resummation.* By S. Moch, J.A.M. Vermaseren, A. Vogt. [hep-ph/0506288]. Nucl.Phys. B726 (2005) 317-335. Cited 73(11) times in the Spires database.
8. *The Quark form-factor at higher orders.* By S. Moch, J.A.M. Vermaseren, A. Vogt. [hep-ph/0507039]. JHEP 0508 (2005) 049. Cited 78(4) times in the Spires database.
9. *The Multithreaded version of FORM.* By M. Tentyukov, J.A.M. Vermaseren. [hep-ph/0702279]. Comput.Phys.Commun. 181 (2010) 1419-1427. Cited 40(7) times in the Spires database.
10. *The Multiple Zeta Value Data Mine.* By J. Blumlein, D.J. Broadhurst, J.A.M. Vermaseren. [arXiv:0907.2557 [math-ph]]. Comput.Phys.Commun. 181 (2010) 582-625. Cited 28(4) times in the Spires database.

The number of citations is the total number in the Spires database. Between parentheses is the number of auto-citations.

ii. Invited presentations to peer-reviewed internationally established conferences.

Conferences during the past few years

- "The FORM project" in "Loops and Legs" 2008 Sondershausen, Germany.
- "Some obstacles in the calculation of structure functions" in "DIS" 2009 Madrid.
- "Large-scale computer algebra calculations of Feynman diagrams", in "QCD: The Modern View of Strong Interactions" 2009 Berlin.

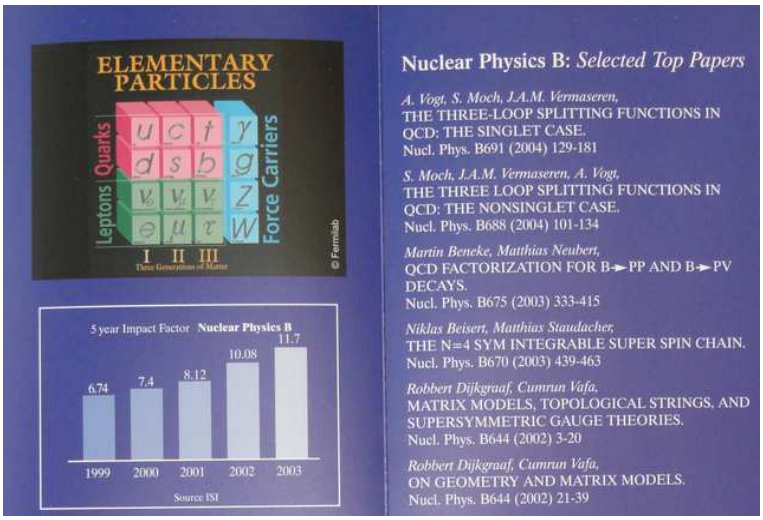


Figure 1: From a 2004 advertising flyer of Nuclear Physics

- "Euler Sums and Multiple Zeta Values" in "Integrability in Gauge and String Theory" 2009 Potsdam
- "FORM facts" in "Loops and Legs" 2010 Wörlitz, Germany.

iii. Organization of International conferences.

- Chair, XI International Workshop on Advanced Computing and Analysis Techniques in Physics Research (ACAT2007)
- Member of the International Advisory Board of the ACAT workshop (held on average once every 18 months since 1989)
- Member of the International Advisory Board of the Loops and Legs workshops (workshop held every 2 years).

iv. International Prizes/Awards.

- 2007 Humboldt Research Award of the Alexander von Humboldt Foundation, in recognition of *outstanding long term contributions to precision calculations in Quantum Chromodynamics (QCD), notably on the scaling violations of the nucleon structure functions.*

v. Major contributions to early careers of excellent researchers.

- Dr. S. Moch has been a postdoctoral fellow at Nikhef and on the basis of the work we did together he has obtained a position at DESY, located in Zeuthen.
- Prof. A. Vogt has been a postdoctoral fellow at Nikhef and on the basis of the work we did together he has obtained a chair at the university of Liverpool.

Section 1c: Extended Synopsis of the scientific proposal

Ambition and objectives

The proposition of HEPGAME is to enable calculations that are currently orders of magnitude too complicated to be solved. The calculations are encountered for example in particle physics and quantum field theory. We will achieve this new fundamental ability by introducing algorithms that have revolutionized the field of game programs. The proposal has two main objectives, each with impact on various fields of science.

O1 To be able to perform a number of new symbolic calculations needed in high-energy particle physics that are one to two orders of magnitude more complicated than the current state of the art.

O2 To develop a method to do these calculations in as automatic a way as possible. This will be achieved by applying the new method of Monte Carlo Tree Search (MCTS). MCTS has already led to much progress in the development of computer programs for the Oriental game of Go. We will apply MCTS for the first time beyond the field of gaming to the derivation of formulas.

The two objectives go hand in hand: O1 will be rather inefficient or even impossible without achieving O2, while O2 can never be efficient without the testing grounds of O1. Reaching these objectives will impact three areas.

- The development of new techniques will require many tests. We will start with simpler reactions in high-energy particle physics than the reactions that are our eventual target. This implies that we will achieve a wide range of physics results using the new methodology.
- The project will result in new concepts in the field of formula manipulation by computer. We will implement these in the open source computer algebra system FORM in a way that everybody can benefit from it.
- Applying MCTS to physics equations will introduce new techniques in MCTS itself as well. This will benefit other projects that depend on MCTS.

The need for more accurate QFT calculations

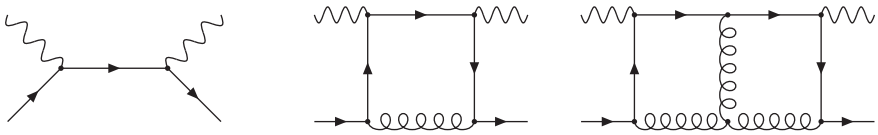
In particle physics and quantum field theory (QFT) calculations are needed for a detailed comparison between data and theory. It is only natural to demand that the theoretical results are at least as accurate as the data, because usually the data are much more expensive to obtain. One prominent example here is the measurement and the QED calculation of the electric dipole moment of the electron (also called $g-2$) which gives agreement between theory and data to almost 10 digits.[2] In QCD, due to its larger coupling constant and a bigger number of Feynman diagrams, currently the best accuracies are in the few percent range. With the advent of the new Large Hadron Collider (LHC) at CERN, which is expected to measure many reactions to the one percent range, *it is essential that the accuracy in QFT calculations is drastically improved beyond the current state of the art.*

Current approach to perform QFT calculations

Although there are nowadays successful nondiagrammatic methods for certain categories of reactions, the standard method for calculations in QFT is still by means of Feynman diagrams. These describe reactions pictorially, based on the terms that are present in the formulas describing the theory. A perturbative expansion in terms of the strength of the interactions corresponds to the number of

[2] T.Aoyama, M. Hayakawa, T. Kinoshita and M. Nio,e-Print: arXiv:1110.2826 [hep-ph]

closed loops in the diagrams. Hence a more accurate calculation will involve diagrams with more loops. Each loop introduces a degree of freedom that is the four-momentum of the loop. We have to integrate over this four-momentum.[3] For example: a three-loop diagram has three integrals over four-momenta. The number of possible diagrams increases more than exponentially with the number of loops and also the degree of difficulty to calculate them depends very strongly on the number of loops.



Examples of Feynman diagrams with zero, one and two loops.
The different types of lines represent different particles.

There are various ways to calculate Feynman diagrams, all of which have a limited range of application. For the type of computations we envisage the best method is currently the integration-by-parts method (IBP). The integration here is the integration over the four-momenta related to the loops. There exists a way[4] to construct complete classes of IBP identities that relate diagrams with different properties. These properties are the powers of the propagators which are represented by the lines in the diagrams and possibly powers of some four-vector dot products. In many cases the equations make it possible to reduce one of those powers to zero. If a line has no propagator in it, it means the line is not present and we have a diagram with one line less. In principle this is a simpler diagram. If we can remove enough lines, eventually we will hit on diagrams that can be done relatively easily. However this is not always possible and often a few diagrams remain that cannot be further reduced by this technique. Those are called master integrals. They will have to be worked out by different techniques.[5] But in many cases it is sufficient that millions of different integrals with different topologies and properties can be systematically reduced to just a few master integrals and once these have been worked out by another technique, the calculation can be done. Some of the most difficult master integrals may even cancel when all diagrams are added. For all calculations we refer to, the master integrals are known to sufficient precision.

The challenge: towards more complicated calculations

The principle described above has been used to solve at least some of the less complicated reactions. As an example, we consider a reaction that we calculated in past years (1995-2004).[6] This calculation was needed for a more accurate determination of the distribution of quarks and gluons inside the proton which in turn is needed by all precision calculations of reactions at the LHC in Geneva. The calculation involved $\mathcal{O}(10000)$ Feynman diagrams with three loops, distributed over some 200 topologies. Some topologies were easy to resolve, but a few dozen hard ones remained. The more difficult topologies involved 16 parameters.[7] Integration had to make 14 of those disappear and 2,

[3] We often make an analytic continuation to $D = 4 - 2\epsilon$ dimensions to avoid problems with divergencies. If done right we can let ϵ go to zero at the very last stage of the calculation.

[4] First introduced in K.G. Chetyrkin and F.V. Tkachov, Nucl.Phys. B192 (1981) 159-204, this method is explained well in: S.G.Gorishni, S.A.Larin,L.R. Surguladze, F.V. Tkachov, Comput.Phys.Commun. 55 (1989) 381-408

[5] There are various techniques here. Each is usually much work by hand but it is a fixed set that we have to prepare before we start the calculation. Fortunately they are the same ones for many different calculations.

[6] S.A. Moch, J.A.M. Vermaseren, A. Vogt: Nucl.Phys. B688 (2004) 101-134, B691 (2004) 129-181, B724 (2005) 3-182.

[7] For a complete explanation at the simpler two loop level, see S. Moch and J.A.M. Vermaseren. hep-ph/9912355, Nucl. Phys. B573, (2000) 853-907.

N and ϵ needed to remain in the answer. The answer had to be a power expansion in the parameter ϵ with "harmonic sums"[8] in terms of the Mellin parameter N.[9] For each topology we determined 20-30 equations relating diagrams with different powers of the propagators to each other with the parameters N and ϵ in them. The next step was to combine and apply these equations in such an order that the 14 parameters systematically are reduced to either zero or one. However, there are extra degrees of freedom as we may have to apply equations with N replaced by $N+1$ or similar shifts with the other 14 parameters. Hence, effectively the number of possible equations is much larger. Some parameters need two steps to be eliminated: once in a simple way to lower a power $m+1$ to m (with coefficient $1/m$) and once in a more complicated way for the transition from one to zero in some of the terms. Combining equations so that the new equations have fewer parameters, can result in rather lengthy equations (more than 1000 lines) and thus this work was done by computer, using the FORM program. Our approach was as follows.

1. The equations are generated by a FORM program.
2. The equations are inspected.
3. We select one of the equations to use and add a few lines to the FORM program to substitute it in the other equations.
4. We run the FORM program. This may take anywhere from a few seconds to up to many minutes when we are working on the very last equations.
5. We inspect the output. If satisfactory, we continue with step 2, unless we are done already. If we do not like the result we have to undo the code of the last phase and try another equation. Maybe we have to backtrack by much more than one phase. With bad luck we may have to start over again.

In many cases this approach led to a satisfactory solution, even though the work for a single topology might take anywhere from hours to weeks. Extending the same calculation to one additional loop would increase the number of topologies to more than 1000, the number of parameters would increase by 6 and the number of equations would go to 30-40. *Without extra levels of automatization such calculations are currently impossible.*

In principle, for the three loop calculation, we start out with $\mathcal{O}(100)$ or more possibilities and we may need to go up to 25 steps deep. At the end there are rather few possibilities. In all there might be $\mathcal{O}(10^{40})$ different paths which would increase to $\mathcal{O}(10^{70})$ for a four loop calculation. Once a path has been found, the equations that define the path can be written to file as a FORM procedure, ready to be applied to the Feynman diagrams of the given topology. *It is the purpose of this project to investigate how to automatize the search through the possibilities and thereby create the capability to do more complicated calculations.*

Introducing MCTS to enable more complicated calculations

Clearly, what is needed is to automatize the search through the equations to determine the order in which to apply them most effectively and how. Surprisingly, this looks very much like what is common in strategic games like Chess and Go. Hence, let us have a closer look at this.

For clarity of our understanding let us look first at the more traditional minimax method in which an ideal Go program tries a number of moves, for each move it tries a number of answers by the opponent up to a certain prefixed depth and then evaluates the resulting positions. Of all moves tried it takes the maximal value of the evaluation function while the opponent part of the program tries to minimize the score.

[8] See J.A.M. Vermaseren Int.J.Mod.Phys A14 (1999) 2037-2076.

[9] The N-th Mellin moment of the function $f(x)$ is defined by $M(f,N) = \int_0^1 dx\; x^{N-1} f(x)$. Knowing all even or all odd Mellin moments allows one to reconstruct the function.

How does this way of thinking translate to our formulas? To solve our problem we have a goal: finding a complete path of equations that breaks down all integrals of a given topology in a desired way. Some paths may be more efficient than others, just like in Go one may win/lose by a smaller or larger margin. The application of an equation corresponds to a move by ourselves plus the move by the opponent. We may assume that the opponent (nature) plays perfectly. Thusfar evaluation is defined as the "inspection" phase of the procedure we outlined. Now we need a good evaluation routine, which, just like in a Chess or Go program, might not be perfect. An adequate evaluation and a well-balanced strategy unit as in a Chess or Go program should help us finding a solution in our maze of equations. However, one intrinsic problem is that for equation solving no appropriate evaluation function is known, not even one that roughly approximates the situation rather well. Interestingly, precisely the lack of a heuristic evaluation has held back progress in Go for decades.

This situation was changed markedly by the introduction of Monte Carlo techniques.[10] Monte Carlo techniques can be parallelized easily and thus can exploit much more computer power in a straightforward way without too much synchronization overhead.[11] The development of Monte Carlo game playing methods culminated in 2006 in MCTS. In a sense the need for a heuristic evaluation function was replaced by a probabilistically derived evaluation function.[12,13,14]

MCTS creates its "automatic evaluation function" as follows. Instead of performing a heuristic evaluation, a single randomized line of play is simulated to the end, where the score (win/loss/draw) is readily available. A large number of these simulations are repeated, and the results are averaged. Thus, in MCTS the evaluation function is replaced by the average outcome of randomized playouts. After a decades-long focus on deterministic maximization/minimization, the turn to stochastic averaging in Go may come as a surprise.[15]

The method has been proven to be equally succesful in other application areas, from games (Hex, Amazons, Havannah), mathematics (Voronoi diagrams), to combinatorial optimization problems relevant to the petrochemical industry. If the performance with Go is an indication an MCTS procedure may be expected to be better than humans at finding a path through the equations. Still, transferring this method to a new high performance application domain will be challenging. From the physics side the difficult part is to present solving the set of equations as a "proper game" with which MCTS can work. From the Artificial Intelligence side the challenge is to have MCTS work with this "game", its rules, and to devise heuristic MCTS enhancements. The reason is that, although MCTS is probabilistic in nature, its performance does benefit significantly from the use of domain-dependent knowledge, enhancements that exploit the structure of the search space, and its corresponding tuning methods. The new method for solving equations has to be created from scratch.

[10] B. Bouzy and B. Helmstetter, "Monte-Carlo Go Developments", ACG (2003) 159-174. R. Coulom, "Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search" in "Computers and Games" (2006) 72-83. http://dx.doi.org/10.1007/978-3-540-75538-8_7

[11] G. Chaslot, M.H.M. Winands and H. J. van den Herik, "Parallel Monte-Carlo Tree Search" in "Computers and Games" (2008) 60-71. http://dx.doi.org/10.1007/978-3-540-87608-3_6

[12] PhD thesis of G. Chaslot, 2010. http://www.unimaas.nl/games/files/phd/Chaslot_thesis.pdf

[13] Some people claim that the complexity of Chess is comparable to 11x11 Go. Maybe 13x13 Go is more realistic. The complexity of our problem should be somewhere in the range of 9x9 to 12x12 Go.

[14] Whereas in Chess explicit domain knowledge is available (mainly number and value of the pieces) and can be coded relatively easily, this can be done neither in Go nor in equation solving. Here MCTS offers an implicit evaluation function: by using probe-searches it builds up knowledge and constructs a statistical, non-heuristic, evaluation function.

[15] Of course, there are many improvements over the basic idea. It is an active field with much research, not in the least because there is a world championship computer Go.

Methodology and team

To achieve the goals of this project, progress in three different fields of science is needed. The fte estimates are in fte-years.

Physics [9.5 fte-years]: The past 22 years the PI has led very successful collaborations in this field. The work here involves (i) the automatic generation of all equations that belong to a given topology, (ii) creating a program that will do this for any topology, (iii) studying how these techniques can be applied to more reactions than just the ones described above. These should be the QCD polarized anomalous dimensions in deep inelastic scattering at the three loop level, the three loop Drell-Yan reaction (quark/gluon + quark/gluon $\rightarrow$ lepton pair), three loop Higgs production at the LHC and possibly the unpolarized anomalous dimensions in deep inelastic scattering at the four loop level but this is more speculative. It will require extensive research to make our methods work for all these reactions. We will need here 8.5 fte-years at the postdoc level and 1 fte-year of the PI.

Computer Algebra [3.7 fte-years]: The FORM system and its parallel versions TFORM and/or ParFORM will be the main computational tools. They will require new capabilities when the project advances. As creator of FORM, the PI is the most qualified person in the world to guide this. We estimate that 1.5 fte-years with a physics orientation and 1.2 fte-years with an AI orientation and 1 fte-year of the PI should suffice.

Artificial Intelligence and Gaming [9.8 fte-years]: The more advanced aspects of this project will be worked out in collaboration with the Articifial Intelligence (AI) group of Profs. H. J. van den Herik and A. Plaat at the Tilburg University. Van den Herik has published over 120 papers on AI. He is editor-in-chief of the Journal of the International Computer Games Association, the leading journal of the field. He has also supervised more than 60 PhD theses, many of which on computer games and related techniques such as machine learning, vision, and natural language. He recently worked closely on MCTS with Guillaume Chaslot during his PhD studies on the subject (see footnote 12). Van den Herik supervised the PhD research by Levente Kocsis, who went on to co-author the paper introducing MCTS.[16] Prof. A. Plaat is an expert on search algorithms for games and parallel search algorithms with experience at the University of Alberta (the world's largest games group) and the supercomputing technologies group at MIT. The AI part of the work involves 9.8 fte-years as we need to devise new machine learning techniques to optimize heuristics for MCTS. We will also create a new parallelization technique based on TDS.[17] The intertwining of MCTS and FORM at the user level (external FORM programs) will also be part of the AI work (at least 1.2 fte-years charged to the FORM part of the project). If properly optimized we can run the equation solver for each diagram separately rather than for each topology. For reasons explained in section B2, this would give much more efficient programs. The 9.8 fte-years will be divided into 6.8 fte-years for PhD students, 0.5 fte-year of the PI and 2.5 fte-years for the Profs. van der Herik and Plaat together.

Integration: The PI has the background and track record to bring the separate fields together. Since it will be nearly impossible to find people with expertise in all fields simultaneously, we will have to appoint postdocs/Phd students from the various disciplines and have them collaborate effectively. Once the physicists have constructed their equations, the problem can become quite transparent for the AI people, and similarly, once the AI people can explain the nature of MCTS and its intricacies, the physicists can keep that in mind when constructing the equations.

[16] Levente Kocsis, Csaba Szepesvári (2006). "Bandit based Monte-Carlo Planning" ECML-06, LNCS/LNAI 4212, pp. 282-293. http://dx.doi.org/10.1007/11871842_29

[17] J.W. Romein, H.E. Bal, J. Schaefer and A. Plaat, "A Performance Analysis of Transposition-Table-Driven Work Scheduling in Distributed Search" IEEE Transactions on Parallel and Distributed Systems 13 (2002) 447-459.

Equipment: Minor funds (45 kEuro) are requested to cover multicore computer resources.

a: State of the art and objectives

Need for precision

The LHC will make much data available at an expected precision of about one percent. This is to be used to improve our theoretical understanding of particle physics and field theory. It puts a heavy pressure on the theoreticians to compute relevant reactions to a matching precision. In practice this means for QCD the two loop level or in some instances even the three loop level. Working out a reaction at such an accuracy is in many cases beyond our capabilities, either due to a lack of mathematical techniques[18] or a shortage of computer resources.[19] Addressing this issue is the main topic of HEPGAME.

> The LHC is designed to give answers to a number of very fundamental questions. The first is whether the standard model of weak, electromagnetic and strong interactions is indeed complete. This hinges on finding the Higgs particle. In the existing theory, the Higgs particle is responsible for the masses of all other particles, but the value of its own mass would also give us indications about whether there will be more forces of nature. The second question concerns the precise values of the couplings of the Higgs particle to the other particles, which can be an indicator of new physics as well.
> Of course, it is possible that the LHC will find more particles. In that case we will want to measure their parameters accurately to allow us to determine the theory that describes them best. In all cases the results of the LHC will have influence on our understanding of cosmology and the forming of the universe.

Fundamental quantities

Before any explicit calculations of reactions can be performed, some fundamental quantities have to be evaluated. This concerns the beta function, which describes the variation of the strength of the coupling with energy, and the proper anomalous dimensions which describe the energy dependence of other variables. Because their lowest order is already at the one loop level, the anomalous dimensions have to be worked out at one more loop than the calculation of the actual reactions. Hence, for all reactions at the two loop level one needs the QCD beta function and the QCD anomalous dimensions at three loops. The anomalous dimensions are then used to obtain the proper parton[20] distribution functions that can be used for the calculations. To obtain the anomalous dimensions there is a problem. To determine the gluon distribution inside the proton we use the HERA data. These data involve a virtual photon interacting with the proton. However the gluon does not couple directly to the photon and we would need an extra "intermediate loop" to have access to the gluons. Hence, to compute the third order anomalous dimensions directly would correspond to a four loop calculation. Providentially, for the anomalous dimensions we can use a similarity between the divergencies of the unpolarized photon reaction and the reaction of a scalar. By introducing a special scalar particle that couples directly to the gluon we can keep the calculation at the three loop level. Because we use this particle only externally, we do not have to worry whether this particle and its couplings are renormalizable. For polarized distributions a scalar particle will not work and a ficticious spin

[18] We still do not even know the complete class of functions that would describe all Feynman diagrams. People like D. Broadhurst, D. Kreimer, F. Brown and others investigate this class of functions but it is not even clear what numbers will occur (e.g., Multiple Zeta Values).

[19] Some groups work on obtaining the integrals numerically, but the eventual computer programs are very demanding, even for a single integral.

[20] The partons we refer to are quarks and gluons.

2 particle has to be introduced. This makes it possible to remain at the three loop level, but the integrals are much more complicated than for the scalar particle.[21] Although it may not be the most urgent calculation the computation of the full anomalous dimensions for polarized scattering is a proper starting point for the project under consideration. It needs the new techniques that are proposed here and allows testing by means of the fixed Mellin moments which are obtained by a simpler program.

Other calculations

It will be important to convert the calculations of the deep inelastic electron proton scattering to the production of lepton pairs at the LHC (called the Drell-Yan reaction). In the easier scalar channel it will be applicable to Higgs production at the LHC. Having knowledge of this process at the three loop level will eventually be desirable, because while finding the Higgs may be spectacular, measuring its properties and parameters as accurately as possible is of similar importance. The Drell-Yan reaction, which can be measured much easier and more precise, would serve as a luminosity check that can be used for the normalization of the Higgs production and other reactions. The conversion will still require much study of the underlying physics and therefore it is part of the work proposed here.

For a complete computation of the above quantities one would need to know the parton distributions on the basis of the QCD anomalous dimensions at the four loop level. Currenly these are not known. The Karlsruhe group has managed to obtain one fixed moment. However, to create either more fixed moments or the complete set will be a demanding undertaking. The complete set would definitely need the methods proposed here. It may require two orders of magnitude more resources than the three loop calculation. We consider the idea at this moment speculative, but if the resources will continue to grow, the execution of the idea is very high on the list of projects. Even if we cannot obtain all moments, already a few fixed moments would lead to reasonable estimates of the parton distributions.[22] The outcome could be hybrid: most diagrams will be known for all moments, whereas for some of the most difficult diagrams only a limited number of fixed moments can be obtained. It may also be possible that everything can be reduced to a few master integrals for which we have to run as many fixed moments as possible. However following that way we may obtain more fixed moments than in a complete fixed moment calculation,[23] and thus more accurate results.

Current technological level

The calculation of three loop diagrams is currently limited to certain subclasses. The easiest subclass is that of the so-called vacuum bubbles. These are diagrams without external legs. The next subclass consists of propagator diagrams. They are diagrams with only two external legs which, due to momentum conservation must have the same momentum, one incoming and one outgoing. If all internal lines in these diagrams are massless, the diagrams can be computed.[24] There exists a rather efficient FORM program for this computation, called Mincer.[25]

[21] Recently we (S. Moch, A. Vogt and the PI) have been able to work this out for a number of fixed Mellin moments. A publication is forthcoming.

[22] See recent works by S. Moch and A. Vogt.

[23] We calculate either all odd or all even moments. Each next moment takes roughly 4-5 times the resources of the previous moment. This means that eventually the calculation of all moments as a function of the moment number N must be economical.

[24] First introduced in K.G. Chetyrkin and F.V. Tkachov, Nucl.Phys. B192 (1981) 159-204, this method is explained well in: S.G.Gorishni, S.A.Larin, L.R. Surguladze, F.V. Tkachov, Comput. Phys. Commun. 55 (1989) 381-408.

[25] This program is freely available in the FORM distribution. It has been written by the PI and has been used in a large number of publications. In order to optimize the performance of the Mincer program many new and original

In the case that we calculate the total interaction between a probe (virtual photon, vector boson, ficticious scalar or graviton) with momentum q and a parton in the proton with momentum P the computation can be converted into computing the Mellin moments N of the reaction in terms of the variable $x = -q\cdot q/P\cdot q$ which is the fraction of the momentum of the proton carried by the quark or gluon.[26]

> We need to compute the total cross section between the probe with momentum q and the parton with momentum P. This is equivalent to computing the imaginary part of the reaction $q + P \to q + P$. It can be proven that the N-th Mellin moment of this imaginary part can be obtained by taking N derivatives with respect to P of the amplitude of the reaction $q + P \to q + P$ because the full integrated answer will be a power series expansion in $q\cdot P/q\cdot q$, assuming that $P\cdot P = 0$. If one takes N derivatives and then puts P equal to zero, there will be N open Lorentz indices which are removed by a so-called harmonic projection. The N-th harmonic projector in q is a tensor with N indices that is symmetric and traceless in those indices. These tensors can have many terms, but FORM has some special combinatoric functions that allow the construction of these tensors on the spot. Because symmetries are properly taken into account, the effective number of terms is usually rather moderate. This is the method used for the fixed moment calculations. For the more complicated calculation of all moments we derive recursion relations in N that give us directly the powers of $P\cdot q$.

The computation of the higher moments involves much algebra, but the above method eliminates the momentum P from the problem. As a result the integrals that are left are all of the propagator type. Hence they can be computed by the Mincer program, although for the high derivatives they may take substantial computer resources.

The real problem arises when we want to do the computation for all values of N. In other words, we would like an outcome that is a function of the Mellin parameter N. This allows us to make an inverse Mellin transform and obtain the function in terms of the variable x that is (sometimes indirectly) measured by the experiments. If we do so by "brute force", making formulas for N derivatives and following this with all the recursions that are needed to simplify the diagrams, we may end up with many nested sums which can only be solved in exceptional cases. The method that does work is a "delayed expansion". We take the diagrams with their P dependence, leave them in their unexpanded form but remember that in the end we will have terms with N powers of P. In the case that P flows through one line in the diagram only, we can expand because the expansion will yield a single term with exactly N powers of P. Diagrams of this type are called Basic Building Blocks (BBBs). Diagrams in which the P momentum flows through more than one line are called Composite Building Blocks (CBBs) (see Figure 1). They lead, when expanded, to the sums aimed at. The way to proceed now is to write down all identities that can be derived for a given integral. Most come from integration by parts, some come from Lorenz invariance. In addition, there are a few for which we do not have a name. There are typically some 20-30 of them, but because we have to apply them sometimes with, for instance, N replaced by $N + 1$ or something similar for the other parameters the effective number is more than 100 at the start of the problem. These identities relate to integrals with different values of the powers of the propagators or some extra dot products. If we can apply these equations in the proper way we can reduce the integrals to less complicated integrals or to one basic integral that is a function of N, but for which we obtain a (higher order) recursion relation in terms of N.

features have been added to FORM.

[26] See for instance F.J. Yndurain, "The Theory of Quark and Gluon Interactions", Springer Verlag.

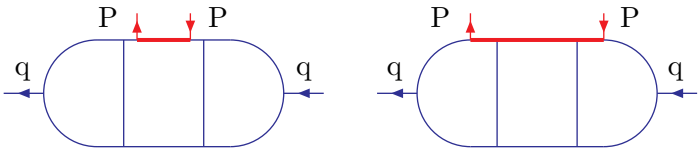
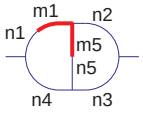
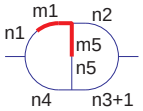


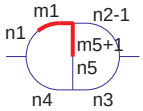
Figure 2: Examples of a BBB diagram and a CBB diagram.

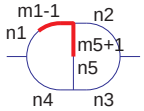
An example of an equation that can be used to either reduce n_3 to one or eliminate n_2.

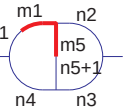


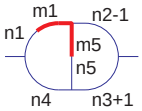
$$= \int d^Dp_1\, d^Dp_2 \frac{1}{(p_1^2)^{n_1}(p_2^2)^{n_2}(p_3^2)^{n_3}(p_4^2)^{n_4}(p_5^2)^{n_5}((p_1+P)^2)^{m_1}((p_5+P)^2)^{m_5}}$$

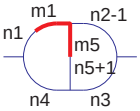
n_3Q^2 

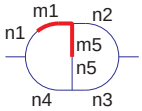
$= m_5$ 

$- m_5$ 

$- n_5$ 

$+ n_3$ 

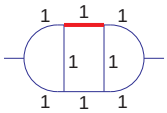
$+n_5$ 

$+ (n_5 + n_3 + 2n_2 + m_5 - 4 + 2\epsilon)$ 

Most equations are much longer and have a structure that is far less transparent.

The problem is in the "if we can". For the BBBs a complete set of solutions has been constructed, although it may not be optimal.

An example of a BBB integral. It is an exceptionally short expression (fewer than 160 bytes). On average the BBB and CBB integrals take more than 20 kilobytes.



$$= 20\zeta_5 + 4S_{3,1,1}(N+1) - 4S_{2,1,2}(N+1) - 8\zeta_3 S_2(N+1)$$

$$+ \frac{1}{N+1}(-4S_{2,1,1}(N+1) + 4S_{1,1,2}(N+1) + 8\zeta_3 S_1(N+1))$$

The function S is a nested harmonic sum as in:

$$S_{m_1,m_2,m_3}(N) = \sum_{i_1}^{N} \frac{1}{i_1^{m_1}} \sum_{i_2}^{i_1} \frac{1}{i_2^{m_2}} \sum_{i_3}^{i_2} \frac{1}{i_3^{m_3}}$$

Based on this complete set of solutions, there are FORM programs that will solve any integral of the three loop BBB type, although some are demanding in terms of computer resources. This may be influenced by the order in which the variables are eliminated. If there is a single nontrivial variable that is eliminated first, it can make one or more other variables nontrivial and this will propagate through the system all the way to the last variable, making the expressions sometimes of astronomical size. If, in contrast, it would be the last variable to be eliminated, the first equations do not do anything –those variables are fine already– and we never get impractically large intermediate expressions.

However, it was already such an effort to create one solution. It would be far too much hand work to create solutions tailored to each integral. It is one of the objectives of the proposal to replace the hand work by a new method.

To show the importance of having the action at the last equation we look at a toy example of a solution in which we have three variables

$$\begin{aligned}
n_1 I(n_1,n_2,n_3) &= I(n_1-1,n_2+1,n_3+1)+I(n_1-1,n_2+1,n_3)\\
&\quad +I(n_1-1,n_2,n_3+1)-I(n_1-1,n_2,n_3)\\
&\quad +I(n_1,n_2-1,n_3)+I(n_1,n_2,n_3-1)\\
n_2 I(1,n_2,n_3) &= I(1,n_2-1,n_3+2)+I(1,n_2-1,n_3+1)\\
&\quad +I(1,n_2-1,n_3)+I(1,n_2,n_3-1)+I(0,n_2,n_3)\\
n_3 I(1,1,n_3) &= I(1,1,n_3-1)+I(1,1,n_3-2)+I(0,1,n_3).
\end{aligned}$$

To reduce $I(3,1,1)$ to objects in which the arguments are at most one takes 2354 intermediate terms and an answer with 24 terms, while reducing $I(1,1,3)$ generates only 5 terms and gives and answer with 4 terms. In the end we have only terms with either all arguments one, or one of them 0 or negative.

The situation with the CBBs is even worse. The finding of a solution is considerably more difficult, because the diagrams are more complicated. When one or more lines are missing, we have a less complicated topology and we have to use the procedures for that topology to break it down into even more fundamental topologies. Eventually we encounter topologies for which the integrals are known. The procedures define a hierarchy of topologies and determine the difficulty of the integrals. It is rather common that when one parameter is lowered one or more others get raised (as in the example above). It can lead to intermediate expressions with a large number of terms. If everything works well we can define a hierarchy here and eventually we obtain an expression that contains only integrals with simpler topologies (for example BBBs) and integrals computed before and stored in tables. It turns out that for some topologies it is intrinsically difficult to find a solution[27] and it may take weeks or even more time to find the solution. In some cases no general solution has been found or the solutions that were found were only for special cases. Sufficiently many special cases could be solved to allow a number of physics calculations. In the case of the full QCD anomalous dimensions of deep inelastic scattering, the project involved about 20 fte-years and several years of CPU time. The new calculations that need to be done will require even more difficult special cases. Without a new technology this means efforts that are next to impossible to organize. A four loop calculation using existing technology would need a multiple of these resources and is assessed impractical. Hence, it is important that a new method be devised to find a general solution by systematic and automatic means.

In the unlikely case that full success cannot be achieved and there are topologies left for which we cannot find a complete and general solution, there are still two possible solutions. Both will need heavy use of the MCTS techniques (proposed below) because "hand guiding" becomes too complicated. Each solution can be achieved by a new method.

[27] In all these derivations we have to avoid substitutions of the type $\epsilon I(\cdots) = \cdots$ because it means that integrals in the r.h.s. need to be known to one extra power in ϵ. Only in three exceptional cases this is allowed. In the BBBs we managed to avoid them all. In the CBBs this is far more complicated and the real reason that we have not (yet?) found general solutions for all topologies. We call these factors ϵ spurious poles. This problem is related to numerical stability problems.

The first method allows a reduction in which we take all integrals with their powers of the propagators. It means that we consider the integrals individually as in $I(1,\cdots)$, $I(2,\cdots)$, $I(3,\cdots)$, etc. and not generically as in $I(n_1,\cdots)$. Because of the number of parameters this yields a large number of integrals, and therefore an impractical number of relevant relations. However, the method gives more flexibility in the reduction scheme and we do not have to apply all reductions in, for instance, n_1 straight after each other. The obvious drawback is that the search for a solution is more complicated and can only be done by computer.

The second method is needed when we run into spurious poles (see footnote 27) in ϵ that the first method cannot resolve. In some cases this is not problematic, because one or more of the three allowed cases may not occur and hence we have one or more spare powers of ϵ. But in the case that there are more spurious poles than reserve powers of ϵ we need extra powers in ϵ for the integrals in the reduction identity. This may require rerunning all those integrals to the desired precision. What the desired precision will be is not known in advance. Each extra power in ϵ needs computer time that is exponential in the power.[28] At this point it is important that the MCTS part of the program tries a solution that is close to the optimal solution. It is defined as the solution with the smallest number of spurious poles. Obviously this method requires substantial computational resources.

Once a path through the equations has been found, the program will automatically construct FORM procedures that will solve the given integral(s). The integrals that have been computed are put in tables that can be used by the tablebase facility of FORM.

It is the objective of this proposal to develop techniques that will allow the automatic construction of solutions for the sets of equations. This will then result in a number of computations that are needed for the LHC and for which at the moment no other practical methods are known.

State of the art with MCTS

To construct these solutions automatically a method is needed to guide the search process in an intelligent way: which equation should be selected for processing. As noted above, this choice can have dramatic consequences for the efficiency of the search. In computer game playing researchers conventionally construct an evaluation function based on heuristics that (1) are a good indicator of the quality of a position and (2) are efficient to compute. For example, in Chess factors as the material balance and mobility are reasonably good indicators of one's chances to win, and can be computed efficiently. The heuristic evaluation functions are a way to capture relevant domain knowledge to guide the search algorithm. For many games, and for other application domains as well, suitable heuristics have been found. However, for the Oriental game of Go no efficient heuristic evaluation function has been found. (Researchers have experimented with the concept of influence, but it is expensive to compute, and it lacks predictive quality.)

[28] The number of possible functions in the answer increases with powers of 3 and for many "fundamental" integrals we need to solve complicated recursion relations by solving a linear set of equations containing the coefficients of all those functions.

Go is considered the oldest and most strategic game of all board games. It is played by two players who alternatingly place a stone on any empty grid position of a 19x19 board. The object is to gain more influence on the board than the opponent, subject to a rule by which it is possible to remove enemy stones from the board, if they are fully and tightly enclosed. The game is most popular in countries like China, Korea, Japan and Taiwan, but also gains in popularity in the West. For beginners, the 9x9 board is often used. One of the things that makes Go so attractive is that there is a good handicap system that does not alter the character of the game much. This allows players of different strength to play interesting games; it also enables the computer to play against a top level player.

Researchers tried many alternatives, such as pure knowledge based search, neural networks, and probabilistic methods. In 2006 the combination of a purely probabilistic evaluation function with a new tree search algorithm[29] created a breakthrough.[30]

MCTS has been developed in the context of game playing programs. Search procedures in game playing programs consist of a basic search strategy, which is enhanced and tuned to achieve maximum performance for the game at hand. In high performance Go programs, the basic MCTS algorithm is enhanced with specialized playout strategies, domain knowledge encoded in large pattern databases, transposition tables, a parallelization approach, and other techniques, to take advantage of domain-specific properties.

MCTS consists of four main steps: selection, expansion, simulation (playout), and backpropagation (see Figure 2). The main steps are repeated as long as there is time left. Per step the activities are as follows. (1) In the selection step the tree is traversed from the root node until we reach a node, where we select a child that is not part of the tree yet. (2) Next, in the expansion step the child is added to the tree. (3) Subsequently, during the simulation step moves are played in self-play until the end of the game is reached. The result R of this—simulated—game is $+1$ in case of a win for Black (the first player in Go), 0 in case of a draw, and -1 in case of a win for White. (4) In the back-propagation step, R is propagated backwards, through the previously traversed nodes. Finally, the move played by the program is the child of the root with the best win/visit count,[31] depending on UCT[32] probability calculations (for details we refer to the references). An outline of the four step of MCTS is depicted in Figure 2.[33]

To adapt MCTS to high performance solving of QFT equations first the main steps must be converted (node selection, expansion, simulation, backpropagation). The performance of MCTS depends to a large extent on further enhancements to the expansion strategy, simulation phase, and the parallelization techniques. So, the second step is to devise enhancements and refinements to these areas. This will be of crucial importance to achieving high efficiency for the equation solver.

The performance breakthrough in Go caused by MCTS has spawned active research in further enhancements, driven by the hope of achieving grandmaster level performance on the full 19x19 board (and not just on the 9x9 board). Some techniques are Go-specific, but a number of them are

[29] Levente Kocsis, Csaba Szepesvári (2006). "Bandit based Monte-Carlo Planning" ECML-06, LNCS/LNAI 4212, pp. 282-293. http://dx.doi.org/10.1007/11871842_29

[30] One of the strongest programs of that moment, MoGo-TITAN, has beaten, for the first time, a professional Go player in a 9-stone handicap game. This event took place in August 2008 in Portland, OR. MoGo-TITAN ran on 800 cores of the Huygens supercomputer in Amsterdam.

[31] Different programs may have a different definition of what is best.

[32] Upper Confidence bounds applied to Trees

[33] G. Chaslot, M.H.M. Winands and H. J. van den Herik, "Parallel Monte-Carlo Tree Search" in "Computers and Games" (2008) 60-71. http://dx.doi.org/10.1007/978-3-540-87608-3_6

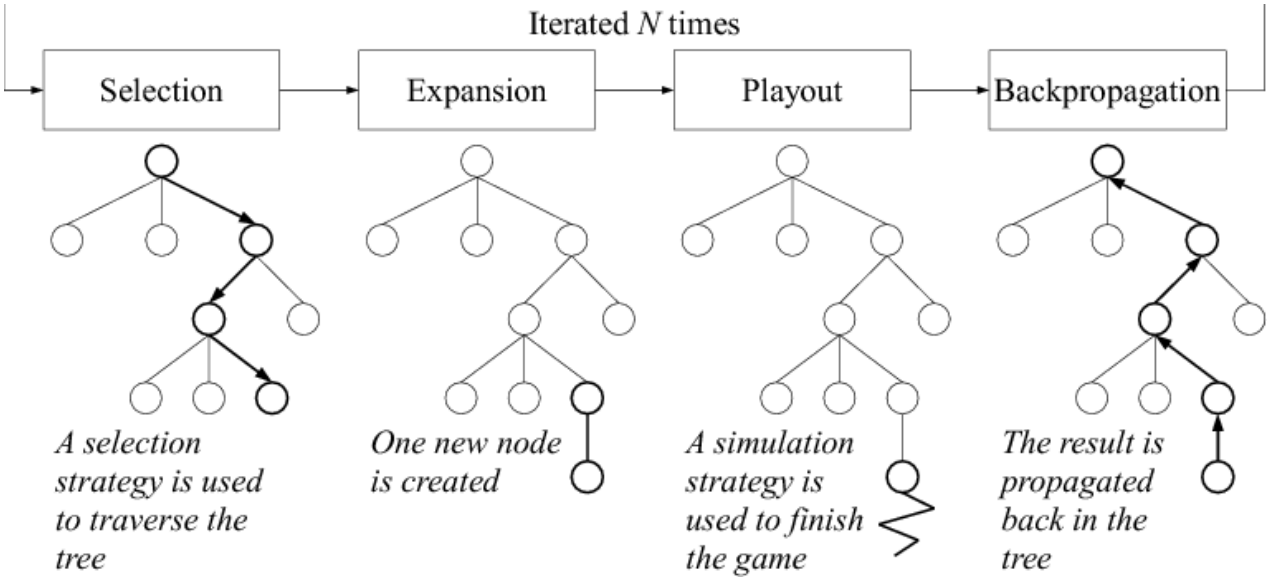


Figure 2: Monte Carlo Tree Search scheme

domain independent and they should be transferable to FORM.

Basically, MCTS playouts are uniformly random. A substantial improvement in performance is achieved when features of the search space are used to guide the playouts. Many of these features rely on domain dependent knowledge, such as, in Go, local patterns or shapes of stones. In equation solving, such features are, for example, the different types of equations and the kinds of factors and terms present in those equations.

In Go handcrafting an effective simulation policy is problematic, requiring hours of tedious and error-prone experimentation. In FORM handcrafting the strategy to solve equations is hitting a similar limit (as has been argued at length elsewhere in this document). Many of the top Go programs from 2006-2011 utilised a small number of simple patterns and rules, based largely on the default policy used in MoGo, the first program to win against a human professional Go player.[34,35] Adding further Go knowledge without breaking MoGo's "magic formula" has proven to be surprisingly difficult.

The most straighforward approach to policy optimization is trial and error. Some knowledge is implemented in playouts, and its effect on playing strength is estimated by measuring the winning rate against other programs. This approach is often slow and costly, because measuring the winning rate by playing games takes a large amount of time, and many trials fail. It is difficult to guess what change in playout policy will make the program stronger, because making playouts play better often causes the Monte Carlo program to become weaker.[36] In order to avoid the difficulties of crafting a playout policy manually, some authors tried to establish principles for automatic optimization. Three promising approaches are described below.

[34]Sylvain Gelly, Yizao Wang, Remi Munos and Olivier Teytaud. Modification of UCT with Patterns in Monte-Carlo Go. Technical Report INRIA No. 6062. November 2006b.

[35]Louis Chatriot, Sylvain Gelly, Jean-Baptiste Hoock, Julien Perez, Arpad Rimmel, Olivier Teytaud. Combining expert, offline, transient and online knowledge in Monte-Carlo exploration. European Workshop on Reinforcement Learning, 2008.

[36]Shih-Chieh Huang, Remi Coulom, and Shun-Shii Lin, Monte-Carlo Simulation Balancing in Practice, Proceedings of the Computer Games Conference, October 2010, Kanazawa, Japan, LNCS 6515, Springer Verlag 2011.

First, it is possible directly to optimize numerical parameters with generic stochastic optimization algorithms such as the cross-entropy method. Such a method may work for a few parameters, but it still suffers from the rather high cost of measuring strength by playing games against some opponents. This cost may be overcome by methods such as reinforcement learning, or by supervised learning from good moves collected from game records. Supervised learning from game records has been quite successful, and is used in some top-level Go programs such as Zen or Crazy Stone.

A second approach is the introduction of a reinforcement learning method called simulation balancing.[37] The key to simulation balancing is to optimize the balance of a simulation policy, so that an accurate spread of simulation outcomes is maintained, rather than optimizing the direct strength of the simulation policy. Minimizing the deviation of outcomes yields better approximations of the true value than tuning for better playouts. As is stressed by the authors[37]: "For clarity of presentation we have focussed on deterministic two-player games with terminal outcomes. However, [the reinforcement learning technique simulation balancing] generalises directly to stochastic environments and intermediate rewards." Finding efficient equation solving strategies in FORM using Monte Carlo search techniques is such an environment.

A third approach to improve the convergence of MCTS has recently been reported.[38] This method is called Accelerated UCT. It is a domain independent technique. It addresses the problem of slow and inconsistent winning ratios in the MCTS tree. It reads as follows. MCTS builds up its knowledge of the search space incrementally. The UCT values at the nodes start uninitialized, and the winning ratios are inaccurate; however, they improve as the number of playouts increases. Given an infinite number of playouts, selecting the move with the highest winning ratio is proved to be optimal. We admit that this theoretical property assumes the condition of infinite time given to UCT. But UCT must determine the next move to play in limited time, i.e., the number of playouts is usually too small to converge to the optimal value. UCT therefore suffers from a deceptive structure in its partially built tree, which may be a cause of selecting a wrong move that may lead to losing a game (in Go) or to selecting an inefficient equation for substitution precluding the solving process to exhaust resource limits (in FORM). Because UCT's playouts often mistakenly evaluate winning positions as losses or vice versa, one example of deceptive structures is caused by an inaccurate winning ratio computed by such faulty playout results. Indeed, the transitory period in which UCT chooses the suboptimal move(s) can last for a very long time due to the over-optimistic behaviour of UCT. A possible way to overcome the above situation is to discount wins and losses for the playouts performed in an earlier stage (DUCT).[39] Because the current playout is performed at a more valuable leaf than previously performed ones, the current playout result is considered to be more reliable. Thus, MCTS can search a more important part of the search tree by forgetting the side effects of previous playouts.[40] In contrast, Accelerated UCT introduces a way of focussing on more recent winning ratios than Discounted UCB. The Accelerated UCT algorithm aims at accelerating the search in a direction of avoiding deceptive structures in the partially built tree. Similarly to Discounted UCB, Accelerated UCT considers recent playouts to be more valuable. However, unlike

[37] David Silver and Gerald Tesauro. Monte-Carlo Simulation Balancing. 26th International Conference on Machine Learning, Montreal, Canada, 2009.

[38] Junichi Hashimoto, Akihiro Kishimoto, Kazuki Yoshizoe, and Kokolo Ikeda, "Accelerated UCT and its application to two-player games, Advances in Computer Games 13, Tilburg The Netherlands, November 2011, LNCS 7168, Springer Verlag 2012 (best paper award) (forthcoming).

[39] It is interesting to note that such an effect also occurs with the self learning Monte Carlo integration program Vegas (G.P. Lepage: "Vegas: An Adaptive Multidimensional Integration Program" (1980)) when the first iteration sampling misses the most important part of a function. This will give a small result with a small error and consequently this iteration is given too much importance in later stages when the results and hence the errors are much bigger.

[40] Kocsis and Szepesvári's Discounted UCB algorithm is an example of such an approach, but no successful results for adapting it to tree search have been reported.

the decay of Discounted UCB, Accelerated UCT non-uniformly decreases the reliability of subtrees that contain the past playout results.[41]

Supervised and unsupervised machine learning methods of feature vectors have been shown to work in practice in Go. These methods, and methods such as Accelerated UCT, are, however, domain independent methods that will equally work in FORM equation solving.

MCTS objectives

In addition to the benefit of solving previously unsolvable Feynman Diagrams, disseminating the MCTS algorithm to a new problem domain will yield new heuristic techniques, new insights into pattern matching, and new parallelization techniques. **As such, the study of an advanced problem domain will be beneficial for artifical intelligence as well. Furthermore, the methods will be beneficial to combinatorial optimization**.

Although at first blush it may seem that equation solvers and game playing programs require different, or even incompatible, search algorithms, the structure of the problem space is quite similar. Game playing programs search a problem space defined by min/max functions, and equation solvers search an AND/OR space. The similarity has been known for some time. For satisfiability, an NP-complete problem with a problem structure related to equation solving, probabilistic algorithms have been used sucessfully (for example WalkSAT[42].).

Therefore, despite the admittedly challenging nature of this proposal, we firmly believe that Monte Carlo Tree Search methods will bring the necessary performance improvement to solving QCD equations.

FORM

For all above work the major tool is the symbolic program FORM. It counts by far as the fastest system for this type of work and it can handle by far the biggest expressions. The last mentioned performance is due to fact that its internal data representation is extremely compact and in addition it can let the formulas overflow to disk with hardly any degradation in performance. As case in point we mention that a recent calculation of relations between Multiple Zeta Values (MZVs) could prove crucial identities to weight 27 and weight 28 whereas the best mathematical programs[43] (programmed in C and dedicated to the problem) could go only to weight 20 and determine only the size of the basis, not expressing all elements in terms of such a basis. The problem has an exponential size. Each increment in weight makes the mathematical program at least a factor 4 slower.

Special projects such as the one described in this proposal always have special demands for software to be used. So far this holds for all big calculations that the PI undertook during his career. New software typically resulted in the addition of new features in programs such as FORM. Many of these features are not only updates, but are original and unique. Some of them are later implemented in other systems.

It is a secondary objective of this proposal to extend FORM with commands and features that enable it to work efficiently and in harmony with the MCTS code. The ideal outcome would be that MCTS can be applied to a large number of problems in physics and

[41] Hashimoto et al. (footnote 38 contains further details).

[42] "Local Search Strategies for Satisfiability Testing. " Bart Selman, Henry Kautz, and Bram Cohen. Final version appeared in Cliques, Coloring, and Satisfiability: Second DIMACS Implementation Challenge, October 11-13, 1993. David S. Johnson and Michael A. Trick, ed. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 26, AMS, 1996

[43] M. Kaneko, M. Noro and K. Tsurumaki, On a conjecture for the dimension of the space of the multiple zeta values, Software for Algebraic Geometry, IMA 148 (2008), 478.

mathematics; at the moment it is hard to predict in detail how this will happen. All we can say is that thus far such extensions have materialized.

b: Methodology

For the methodology to be used in this project we have to consider two sides. On the one hand the proposal constitutes a physics problem and on the other hand a problem of gaming techniques. These two areas of expertise need to be integrated, with FORM as the common denominator. The PI has a long track record in leading such integrative projects. He will lead two sub-teams, one within a physics environment (at Nikhef in Amsterdam) and one embedded in the AI group in Tilburg. The subgroups will benefit from the local expertise and networks. The physics subgroup will consist of the PI (2 fte-year) and three postdocs, one for five years, one for three years and one for two years (10 fte-years). It will be located at Nikhef. The second subgroup will be at the Tilburg University in the AI group of the Profs. H. J. van den Herik and A. Plaat who will support the effort with their expertise and will consist of two PhD students, the PI (0.5 fte-year) and the Profs. van den Herik and Plaat (together 2.5 fte-years). Both groups will need to obtain a thorough understanding of FORM which will be reached through training by its inventor, the PI of the project. The interaction between the two subgroups will be frequent (weekly meetings, online exchange of results and progress reports), which is easily achieved given the acceptable distance between Amsterdam and Tilburg (less than 90 minutes by car).

The estimates in units of fte that follow below should be taken as a rough assessment and may be subject to small changes, depending on the progress of the project and the specific expertise of the individual postdocs and PhD students. They refer to fte-years for the whole period of the project.

The physics side

On the physics side we need people that are experienced with FORM and the computation of multi loop diagrams. In addition they have to know about hadron physics. This means that we prefer postdocs, but even then some time may have to be spent on training, because it is nearly impossible to find people with the perfect profile. The PI has already much experience with this type of training. If this can be done with two or three postdocs simultaneously it will be much more efficient. The training will lead towards the construction of programs that can automatically generate the topologies needed for a given reaction and the derivation of all integration by parts equations for this topology. This phase will need much attention of the PI and should take about one year (based on previous experience).

The second step will involve the definition of the creation of a solution as a game with specific rules. This will require intensive interaction with the gaming group, because a common language has to be developed. At this point some straightforward examples will have to be used to facilitate the experiments.

Simultaneously one or more of the postdocs will investigate which reactions can benefit from these methods. In particular the reactions quark/gluon + quark/gluon goes to either a lepton pair (Drell-Yan) or a Higgs particle are important. As soon as the physics side of the reactions can be resolved, the required calculations can be performed by the software produced in this project.

Once the gaming concept is working, it is time to do some actual calculation(s). This will undoubtedly show many shortcomings and thus there will be a new period of experimenting. It should eventually lead to promising physics results, even though the programs by which they are obtained are not yet optimal.

One of the postdocs (1.5 fte-years) should also study the inner workings of FORM. This will allow additions in the field of equation solving by FORM that would greatly enhance the efficiency

of the whole process. In total we will need 9.5 fte-years for the physics side, 8.5 fte-years from postdocs and 1 fte-year from the PI.

The AI side

On the AI-side, we propose to assign two PhD students. The first PhD student will work on devising enhancements to the simulation phase. This student will work on domain-specific heuristics to enhance the speed of playouts. It will involve using off-line machine learning to learn which patterns of equations are good candidates to improve the performance of playouts, using approaches that have been successful in Go (described in brief in State of the art of MCTS on page 19). Because playouts will involve FORM programs, the PhD student will also familiarize him/herself with FORM to the point that additions concerning MCTS can be made to it if deemed necessary (1.2 fte-years). The graduate student will also be responsible for the conversion of the four main parts of MCTS to equation solving.

The second PhD student will work on parallelization. MCTS is relatively well suited to parallelization. Parallelization of MCTS for the game of Go is an active research area. We propose to assign a second PhD student to the task of devising a parallelization strategy. This work will be based on adapting existing MCTS parallelizations to equation solving, and on a novel hash-based parallelization scheme called NTDS (New Transposition Driven Scheduling). To reduce search overhead we propose to reformulate MCTS so as to allow asynchronous communication. For search algorithms this approach has been pioneered successfully in Transposition Driven Scheduling (TDS).[44] TDS is a fine-grain table-driven asynchronous scheduling approach with demonstrated linear speedup for single agent search. Its successful application to MCTS has so far remained an open question. We believe that NTDS is especially well suited for MCTS in the context of equation solving because MCTS consists of many small computations, with an intricate relationship between data. Here we will extend the work by Kishimoto.[45]

The Profs van den Herik and Plaat will together contribute 2.5 fte-years to this part of the project and the PI will devote 0.5 fte-year to it.

The AI side amounts to 9.8 fte-years for the whole project (1.2 fte-years of the first PhD student is charged to "the FORM side").

The FORM side

As mentioned above, both a member of the physics subgroup (1.5 fte-years) and a member of the AI subgroup (1.2 fte-years) should develop sufficient expertise to be able to make, under the guidance of the PI (1 fte-year), additions to the FORM system. The additions should either greatly enhance the efficiency of the whole process, or improve the applicability of MCTS for mathematical purposes. The approach will guarantee that a wide audience can benefit from the results of the project. Over the whole project this amounts to 3.7 fte-years.

The final configuration

In the final configuration, near the end of the project we would have two subgroups that interact well with each other and can improve the methods in a rather efficient way. Changes in the MCTS

[44] John W. Romein, Henri E. Bal, Jonathan Schaeffer and Aske Plaat. A Performance Analysis of Transposition-Table-Driven Work Scheduling in Distributed Search. IEEE Transactions on Parallel and Distributed Systems. Vol. 13, No. 5. pp. 447-459, May 2002.

[45] Akihiro Kishimoto and Jonathan Schaeffer, "Transposition Table Driven Work Scheduling in Distributed Game-Tree Search", Canadian Society for Computational Studies of Intelligence, 2002.

may need changes in what we call the "the rules of the game" and the inverse may also occur. Cooperation should lead to many experiments and should tune both sides. It will undoubtedly result in a considerable increase of knowledge, both about how to deal with the equations and how to search through such systems. Since the knowledge increase depends on active research and is expected to happen a few years from now, it is impossible to be very specific on the results.

c: Resources (incl. project costs)

Personnel

The planned personnel distribution in months is (assuming a starting date of 1 July 2013):

	2013	2014	2015	2106	2017	2018	Remarks
Postdoc 1	6	12	12	12	12	6	
Postdoc 2	6	12	12	6			
Postdoc 3				6	12	6	
PhD student 1	6	12	12	12	6		
PhD student 2		6	12	12	12	6	
PI	6	12	12	12	12	6	At 50%.

In section B2b about methodology is explained who will be doing what. We anticipate that not all people can be employed immediately at the beginning of the project. We hope that we can find two postdocs and one PhD student at the beginning, the second PhD student a year later and the third postdoc after the second postdoc has left.

Equipment

The project will put a heavy demand on computer resources. Much of this can be taken care of by the infrastructure at Nikhef. It holds in particular for those production runs that are made when little or no human interference is required. During the experimentation stage, however, it is rather important to have computers with many cores that all work together on the same program. It reduces human waiting time considerably. Alternatively, it will yield much better results in a fixed amount of time. Hence, it is important to have a number of servers with as many cores as possible. Since the field of computer hardware is changing rapidly the exact specifications will be determined at the moment we wish to buy these servers which is at the beginning of the project (July 2013). Experience of the PI at Nikhef shows that such state of the art computers cost typically 15,000 Euro each. In the past this has been the case for several multicore computers that were used for the development of TFORM, the multicore version of FORM. Of course, for more money one can buy better computers, but beyond the above amount the cost/benefit ratio starts deteriorating. For MCTS, it holds that in principle bigger computers would be beneficial, but since each core has to run a FORM program at a given point the programs may start interfering with each other and so the cost/benefit ratio will reach an optimum. Here important parameters are the size of the memory and the speed and size of the hard disk. We would like to have two such servers for the AI subgroup and one for the Physics subgroup. This comes to a budget of 45,000 Euro for equipment to be spent at the beginning of the project.

Travel

Travel costs of the team involve costs for the weekly meetings, visits to conferences and a budget of 40,000 Euro for visiting external experts.

Subcontracting

During the project there will be several (three) audits of the finances by an accountant. The total costs are estimated to be 8540 Euros.

Total costs table

	Cost category	**Month 1-18**	**Month 19-36**	**Month 37-54**	**Month 55-60**	**Total M1-60**
Direct costs	*Personnel*					
	PI	69,000	69,000	69,000	23,000	230,000
	Postdocs	195,000	195,000	195,000	65,000	650,000
	PhD students	102,000	153,000	127,500	25,500	408,000
	Total personnel	*366,000*	*417,000*	*391,500*	*113,500*	*1,288,000*
	Other direct costs					
	Equipment	45,000				45,000
	Travel team	17,825	20,925	19,375	5,425	63,550
	Travel experts	13,650	13,650	13,650	4,550	45,500
	Total other direct costs	*76,475*	*34,575*	*33,025*	*9,975*	*154,050*
Indirect costs		*88,495*	*90,315*	*84,905*	*24,695*	*288,410*
Subcontracting		*2,840*	*2,850*		*2,850*	*8,540*
Total costs of project		**530,970**	**541,890**	**509,430**	**151,020**	**1,739,000**
Requested grant		**530,970**	**541,890**	**509,430**	**151,020**	**1,739,000**

The PI will devote at least 50% of his time to the project. In the costs table he is counted for 40%.

The following "milestones" table can only be a very rough indication, because the project involves a number of parallel developments, that do interact with each other, but each as such will only reach their conclusion near the end of the project. Hence we can indicate only a few expectations.

"key intermediate goal" as defined in section 2.	Estimated % of total requested grant	Expected to be completed on month:	Comment
Additions to FORM for manipulating tree structures.	10%	30	The time spent here is a combination of research to find out what needs to be done, and implementing it. This is in parallel with the other objectives.
A first effective MCTS system for solving coupled recursion equations.	20%	36	This should be the first, still far from perfect, system that can work its way through the equations of a number of simple systems and the system for the polarized anomalous moments. It should incorporate new techniques to search the trees involved. It should also lead to several publications.
Polarized anomalous moments at three loops for DIS.	20%	36	The systems of equations here have turned out to be rather resistent to human manipulation. Computer solutions should solve them to a sufficient degree. This will lead to several publications.
More additions to FORM for coupled recursion relations.	5%	60	Almost certainly some new mathematics may be needed. This will need new FORM procedures and new FORM capabilities.
A complete AI system for solving coupled recursion equations.	20%	60	The eventual results from the game theory side will involve at least an complete program for solving the targetted physics reactions. Most likely there will be much more. In addition efficient parallelization will make the calculations much faster than before. Clearly a number of publications will come from this.
The three loop Drell-Yan calculation and possibly more calculations.	25%	60	This is one of the targets and it does have spinoffs like three loop Higgs production if that has not been calculated by the with the use of other techniques. This will give a number of publications. This step will also include an attempt at solving the equations for a 4-loop system.
Total	100%		