Summary of $\frac{1}{3}$P3 “Simulations and Computations in Theoretical Physics and Phenomenology”:

— Automated Calculation and Simulation Systems —

Thorsten Ohl
— Universität Würzburg —
⟨ohl@physik.uni-wuerzburg.de⟩

ACAT 2002, Moscow State University, June 28, 2002

Future (Linear) Colliders as New Frontier in Energy and Precision:

- final states with many tagged weakly interacting particles accessible
- *(in the absence of low energy SUSY:)* physics beyond the standard model may only be accessible in precision tests of standard model processes
- e. g.:
  - W/Z couplings in $e^+e^- \to f_1\bar{f}_2f_3\bar{f}_4$ (already at LEP2)
  - WW scattering in $e^+e^- \to \nu_e\bar{\nu}_e f_1\bar{f}_2f_3\bar{f}_4$
  - $t\bar{t}$ production in $e^+e^- \to f_1\bar{f}_2f_3\bar{f}_4f_5\bar{f}_6$
  - $WW/t\bar{t}$ scattering in $e^+e^- \to \nu_e\bar{\nu}_e f_1\bar{f}_2f_3\bar{f}_4f_5\bar{f}_6$

$\therefore$ we will need reliable predictions and simulation tools to unleash the full potential of the Future Colliders

- studying EWSB requires complete (gauge invariant!) calculations
- polarization must be included

☹ qualitatively more complicated than, say, LEP1

- the number of Feynman diagrams explodes combinatorially
- the algebraic expressions grow much more complicated with the growing number of building blocks (independent momenta and polarizations)
- the gauge cancellations become extremely hazardous
- the phase space also becomes much more intricate

∵ even if we had enough graduate students and postdocs, we should not waste them on repetitive "assembly line" calculations

∴ formalize the calculations so that the repetitive part can be delegated to patient computers.

☹ loops for many particles will need a lot more work

∵ one-loop calculations for $2 \to 4$ remain the limit of our capabilities

Fully automated calculation and simulation systems in particle physics aim to produce

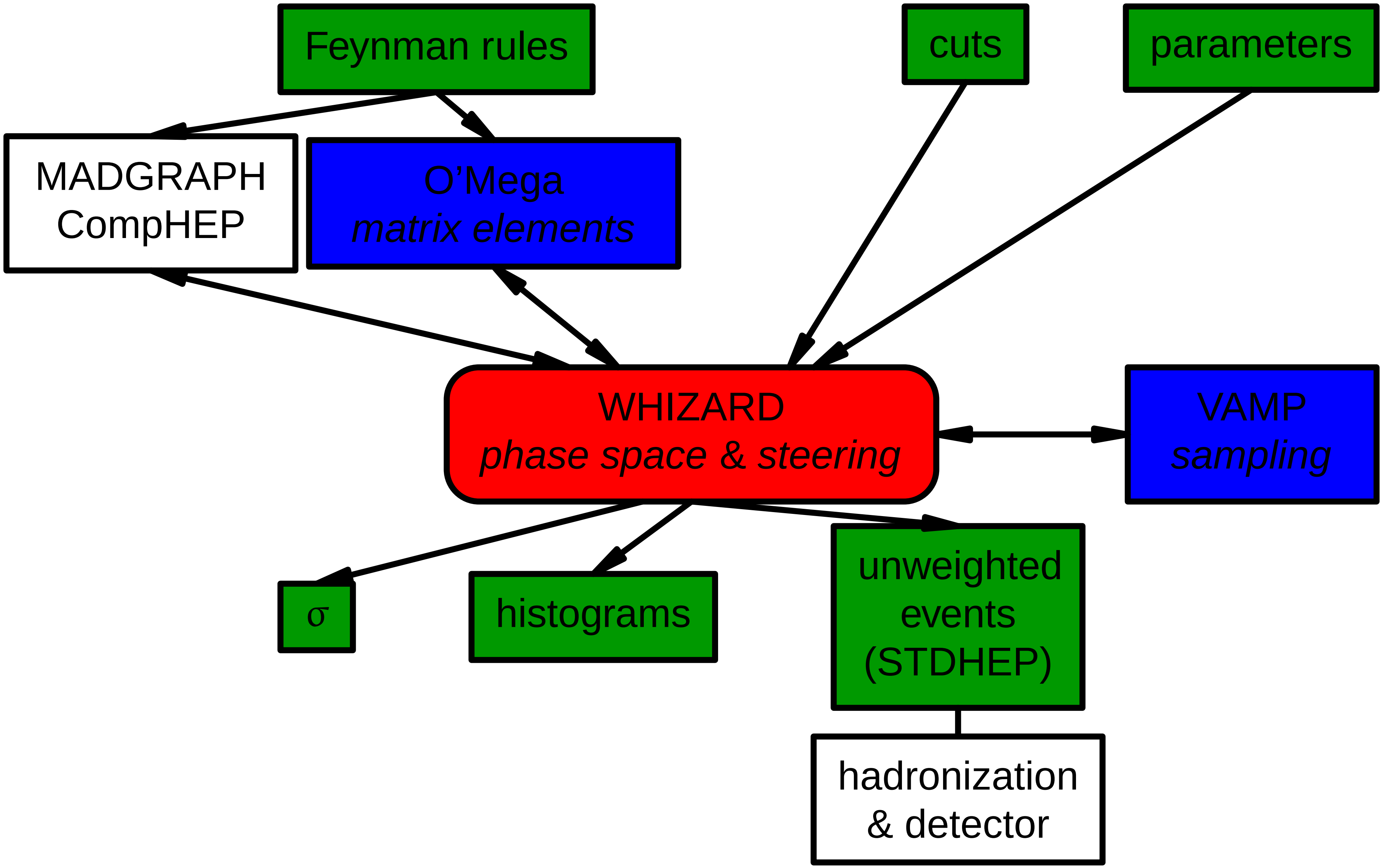
$$\left.\begin{array}{ccc} \text{model} & & \mathcal{L} \\ \text{parameters} & & m, g, \ldots \\ \text{process} & \text{e. g.} & e^+e^- \to \nu_e\bar\nu_e\mu^+\nu_\mu d\bar u \\ \text{cuts} & & p_{T,\text{min}}, E_{\text{min}}, \ldots \end{array}\right\} \Longrightarrow \left\{\begin{array}{c} \sigma \\ \text{event samples} \end{array}\right.$$

without (or with as little as possible) expert human intervention.

NB: these systems do typically produce partonic final states and leave fragmentation and hadronization to `Pythia`, `HERWIG` et al.

The job of automated calculation and simulation systems can be divided in two steps

1. calculate matrixelement T (i. e. generate Feynman diagrams, derive arithmetical expression and generate executable code)
2. integrate $|T|^2 d\mu$ or generate events according to $|T|^2 d\mu$, usually with a little help from the structure of T.


Feynman rules
cuts
parameters
MADGRAPH
CompHEP
O'Mega
matrix elements
WHIZARD
phase space & steering
VAMP
sampling
σ
histograms
unweighted
events
(STDHEP)
hadronization
& detector

Some systems are complete, some provide components of complete systems:

- CompHEP: complete system (2 talks, 2 posters)
- CalcHEP: CompHEP clone (1 talk)
- GRACE: complete system (1 talk and additional talks on sampling!)
- O'Mega: matrix elements
- WHIZARD: phase space
- HELAC/PHEGAS: standard model matrix elements and phase space
- Madgraph: standard model matrix elements
- Alpha: standard model matrix elements
- FeynArts/FeynCalc: loop diagrams
- CalcPHEP: standard model $2 \to 2$ processes @ 1-loop (2 talks, 2 posters)

- tree-level standard model for $2 \to 4$ and $e^+e^- \to 6$particles is well under control for some time now and can readily be used by non-experts

- support for general $2 \to 6$ and $2 \to 8$ processes is not completely usable for production yet
  - systems with already complete physics support do not scale optimally
  - optimally scaling systems not completely implemented yet

  but is coming along fast

Challenges

- 'Beyond The Standard Model' (MSSM in particular)
  - progress reported at ACAT 2002 (see below)
- loops
  - growing evidence, concern and consensus that the classic analytical approach "does not scale"
  - projects have started working on (semi-)numerical approaches

Currently, all automated systems use a set of Feynman rules (propagators and vertices) as input

- ☺ mathematically equivalent to the lagrangian describing the model
- ☹ derivation from lagrangian extremely tedious and error-prone, e. g.: many thousand different vertices in the MSSM

Talks

- T. Kaneko: A Package for Generating Feynman Rules in the GRACE System
- A. Semenov: The CompHEP/SUSY Package

Posters

- A. Kryukov, V. Bunichev, A. Vologdin: Using `form` fir Symbolic Evaluation of Feynman Diagrams in the `CompHEP` Package
- A. Belyaev: Study of Viable SUSY GUTs with Non-Universal Gaugino Mediation: `CompHEP` and `ISAJET` Application

Tools for the derivation of Feynman rules for complicated models (MSSM, in particular) need to

- expand lagrangians in component fields and momenta
- handle mixing from non-diagonal mass matrices

using a very natural notation in order to avoid errors.

∴ such tools turn quickly into fully developed symbolic manipulation languages with special features, e. g.:

  - objects from mutually commutative, but non-commutative algebras (e. g. for non-simple gauge groups) are not described transparently by general purpose computer algebra systems
  - covariant derivatives play a dual role as operators and field components

☺ we will soon have several independently derived 'model files' for the MSSM

Moving ‘Beyond The Standard Model’ requires ‘first generation’ systems to relax some assumptions, e. g.:

- Dirac spinors $\longrightarrow$ Majorana spinors
- quartic vertices $\longrightarrow$ vertices of higher degree
- dimension-4 operators $\longrightarrow$ higher dimension operators (more complicated momentum depenence)
- non-commutative field theories (really strange vertices . . .)
- non Feynman diagram contributions, like K-matrix unitarization

One solution is to replace hard-coded subsystems making special assumptions by general purpose components, e. g.:

- `CompHEP` reported in a poster how the special purpose C-routines for symbolic squared diagram evaluation are being replaced by calls to the general purpose `form` system

Question:

- how can components communicate . . .
- . . . across abstraction layers? (subroutines, objects, modules, etc.)
- . . . across adress spaces? (processes, networks, etc.)
- . . . across time? (data storage, persistency)

Talks

- L. Lönnblad: Status of the `Pythia7` Project
- A. Cherstnev: Toolkit for Partonic Events Data Bases in the `CompHEP` Package

Pythia7 is not an automated system in the the strict sense (all physics knowledge is hardcoded), but all automated systems have to talk to Pythia for fragmentation and hadronization of hard events...

- A project to completely rewrite the Lund event generators in C++. Pythia7 can be used for simulation of any high-energy particle collision, but the main target is LHC physics. It will provide a general structure for implementing models for event generation, only the Lund Model (HERWIG++ has joined the effort).
- Pythia7 exists today as a proof-of-concept version (some basic $2 \to 2$ matrix elements, remnant handling and Lund string fragmentation.
- Plans
  - A new pre-release in 2002 with the first HERWIG stuff.
  - A usable generator in 2003
  - The Standard Generator at the start-up of LHC
- NB: Help is appreciated.

External representations of event samples useful for

- communicating from partonic event generators to soft QCD Monte Carlos to detector simulations
- saving intermediate and/or final results
  - reading stored events faster than regeneration for complicated partonic states
  - there will always be a gain for reweighting processed data
  - ☺ never underestimate the value of external representations for debugging

A. Cherstnev proposed a text based Data Description Language for event samples and implemented the necessary toolkit for manipulating it

- merging of data sets, etc.

Discussion:

- abstract syntax very useful, but the concrete syntax could be replaced with XML for even easier interfacing.

Talks

- D. Bardin: Project `CalcPHEP`, Calculus for Precision High Energy Physics
- P. Christova: QED Radiative Corrections within the `CalcPHEP` Project

Posters

- L. Kalinovskaya: About the Implementation of $e^+e^- \to f\bar{f}$ Processes in the Framework of the `CalcPHEP` Project
- G. Nanava: A Monte Carlo Simulation of Decays Within the `CalcPHEP` Project

Reimplementation of the complete standard model 1-loop radiative corrections (incl. soft bremsstrahlung) for $e^+e^- \to f\bar{f}$ from scratch

Motivation

- preserving the body of knowledge for future generations by providing a consistent and systematic option for redoing the calculations
- application to off-resonance physics at JLC/NLC/Tesla

Procedure

- decomposition into form factors
- calculation of form factors
  - collection of `form3` procedures for symbolic calculation, renormalization and creating Fortran code for numerics

Status

- t-channel NC and s-channel done
- s-channel/t-channel interferences and decays in progress

☺ very good agreement with existing codes

If we ever manage to produce a fully general and efficient automated calculation and simulation system, that is even easy to use for experimentalists . . .

☹ . . . we risk to make ourselves obsolete as phenomenological theorists.

Fortunately . . .

☺ . . . we will never get there

because

- smart theoretical theorists will always come up with new theories with new features that we have not anticipated
- smart experimentalists will always push the frontiers in energy and precision, calling for ever more precise calculations of ever more complicated processes

☹ uncharted territory:

- the only complete 1-loop calculations of $2 \to 4$ processes in the minimal standard model remains “almost completed”

$\therefore$ systematically charting model landscapes will remain practically impossible unless we develop new methods

Final word:

- variety is good, we need to be able to cross check our results
- communication is even better
  - among developers (ACAT 200x)
  - among programs using pluggable components