Constrained MC for QCD evolution with rapidity ordering and minimum kT[⋆]

S. Jadach[ac], **W. Płaczek**[b], **M. Skrzypek**[ac],
P. Stephens[a] and **Z. Wąs**[ac]

[a]*Institute of Nuclear Physics, Polish Academy of Sciences,
ul. Radzikowskiego 152, 31-342 Cracow, Poland.*
[b]*Marian Smoluchowski Institute of Physics, Jagiellonian University,
ul. Reymonta 4, 30-059 Cracow, Poland.*
[c]*CERN, PH Department, TH Division, CH-1211 Geneva 23, Switzerland.*

Abstract

With the imminent start of LHC experiments, development of phenomenological tools, and in particular the Monte Carlo programs and algorithms, becomes urgent. A new algorithm for the generation of a parton shower initiated by the single initial hadron beam is presented. The new algorithm is of the class of the so called "constrained MC" type algorithm (an alternative to the backward evolution MC algorithm), in which the energy and the type of the parton at the end of the parton shower are constrained (predefined). The complete kinematics configurations with explicitly constructed four momenta are generated and tested. Evolution time is identical with rapidity and minimum transverse momentum is used as an infrared cut-off. All terms of the leading-logarithmic approximation in the DGLAP evolution are properly accounted for. In addition, the essential improvements towards the so-called CCFM/BFKL models are also properly implemented. The resulting parton distributions are cross-checked up to the 10^{-3} precision level with the help of a multitude of comparisons with other MC and non-MC programs. We regard these tests as an important asset to be exploited at the time when the presented MC will enter as a building block in a larger MC program for W/Z production process at LHC.

To be submitted to Computer Physics Communications

1 Introduction

In the past, as at present, the central goal of high energy physics is to explore new ranges of energies of interactions. These new ranges of energies either facilitate the discovery of new particles and interactions or validate the Standard Model of elementary interactions, as understood today, by extending them to an even broader range of energies (distances) than currently available. The new generation of experiments at the nearly completed Large Hadron Collider (LHC) in Geneva will be ready soon to take data.

For the proper interpretation of expected new data, an effort in understanding known physics is needed. In particular, it might be that the signatures of the new physics will have to be deciphered from the background of dominant processes expected from the Standard Model.

In the case of hadron colliders, description of the Standard Model processes is rather complicated; the colliding hadron beams are not the elementary fields of the Standard Model but bounded states of quarks and gluons. Even worse, in the low energy limit quantum Chromodynamics (QCD) looses predictive power and does not control the relations between the hadron wave function and elementary fields representing quarks and gluons. It is necessary, albeit highly nontrivial, to combine a phenomenological description of low energy strong interaction phenomena with the rigorous perturbative QCD predictions at high energies.

A multitude of techniques have been developed to merge low energy aspects of strong interaction with the high energy calculations from perturbative QCD. In this work we concentrate on the methodology based on the so-called parton distribution functions (PDF) and parton shower Monte Carlo (PSMC). Special attention will be payed to technical aspects, in particular to precision testing of the numerical tools. We believe that this is very important for the future efforts in minimizing overall systematic errors of the QCD prediction in which PSMCs are used.

In this work we shall concentrate on the question of the evolution equation of the PDF in the Monte Carlo (MC) form. Such an evolution equation for the initial state hadron describes how the PDF responds to an increase of the dimensional, large energy scale $Q = \mu$, set by the hard process probing the PDF. The formula for the integrated cross section, which combines the hard process with the matrix element, has been proved within perturbative QCD in a form of the so called *factorization theorems*, see [1, 2]. These theorems are proved directly from the Feynman diagrams, integrated over the phase space and convoluted with the nonperturbative parton wave function in a hadron. The evolution equation of the PDF can also be formulated using renormalization group and operator product expansion, [3].

However, for the real life practice of the present and future hardon-hadron and hadron-electron collider experiments, one needs a more refined (exclusive) picture of the multi-parton production, the simplest one being the so called parton shower, governed by perturbative QCD. In principle it should reproduce the evolution of PDFs, after integrating it over the Lorentz invariant phase space. If the above is conveniently implemented in the form of the parton shower Monte Carlo event generator, see for example [4, 5], the

inclusion of the process of parton hadronization is possible and very useful for collider experiments. In the construction of such a PSMC the evolution equation of the PDF is used as a guide for defining distributions of the partons emitted from single energetic hadron (the parton initiating shower in such a hadron)[1].

Having in mind the above context, this work has several aims. The principal aim is to use once again the evolution equation of the PDF in order to model multiparton parton shower initiated by the single initial parton located within the single hadron of the collider beam. We shall insist that, as in the factorization theorems, this modelling has to be *universal*, that is independent from showering of the other beam, spectator partons, and the type of the *hard process* of the parton-parton scattering at large energy scale Q.

Another aim is to define an maintain a clear prescription relating variables in the evolution of the PDF with the four-momenta in the PS MC. We keep in mind that such a PS MC will be a *building block* to be used for two beams[2], hence the upper limit of the multiparton phase space (related to Q) should allow for smooth coverage of the entire phase space, without any gaps and overlaps.

There is also an important technical problem to be addressed: the off-shell parton entering the hard process has to have predefined energy and flavour matching preferences of the hard process; hence, a Markovian MC (which is a natural MC implementation of the PSMC) cannot be used to model the initial state PSMC. However, instead of using the so called *backward evolution* [7], our choice will be to employ the technique of the *constrained* MC (referred to as CMC technique); that is to generate the multiparton distribution with the restriction on the value of the parton energy and type of parton. Two distinct versions of this relatively new technique were proposed and tested in refs. [8, 9, 10] for the DGLAP [11] evolution in the leading-logarithmic (LL) approximation.

The aim of this work is to extend the most promising variant of the above CMC technique to a wider class of evolution kernels, beyond DGLAP, towards evolution models of the CCFM class [12]; maintaining at the same time the explicit mapping of the evolution variables into four-momenta.

The other important longer term goal is to facilitate the inclusion of the complete NLO corrections by means of the clearer/cleaned modelling of the PS MC, as compared with the existing combined NLO and PS MC calculations [13, 14]. This will be achieved, for example, by means of better coverage of the phase space in the basic parton shower MC.

The outline of the paper is the following: In Section 2 we shall formulate the general formalism of the evolution equations and their solutions in a form suitable for the CMC technique. In Section 3 we discuss two particular types of the kernels and related Sudakov formfactors. In Section 4 we outline the CMC algorithm for the pure gluonstrahlung segments. In Section 5 we shall introduce in the CMC quark-gluon transitions. In Section 6 results of precision numerical tests of our CMC implementation will be reported. For

---

[1]This is clearly a kind of "backward engineering" – it would be better to get distributions of partons forming PDF at large Q directly from the Feyman diagrams. Unfortunately it is too difficult.

[2]This will be done in the forthcomming paper on the new parton shower MC for W/Z production in hadron collider of the forthcomming paper [6].

testing CMC implementations we shall use auxiliary Markovian MC programs which are described and tested in separate papers [15]. Finally we summarize the main results; some technical details will be included in the appendices.

2 Evolution equations and solutions

In the following we shall formulate the mathematical framework for the evolution equation and its solutions in a form suitable for the construction of the CMC algorithm in the latter part of the paper.

The generic evolution equation covering several types of evolution reads

$$\partial_t D_f(t, x) = \sum_{f'} \int_x^1 du \; \mathcal{K}_{ff'}(t, x, u) D_{f'}(t, u). \tag{1}$$

It describes the evolution of the parton distribution function $D_j(t, u)$, where x is fraction of the hadron momentum carried by the parton and j is the type (flavour) of the parton. Variable $t = \ln Q$ is traditionally called an *evolution time* and it represents the (large) energy scale $Q = \mu$ at which the PDF is probed using a hard scattering process. The LL DGLAP case [11] is covered by eq. (1) with the following identification

$$\mathcal{K}_{ff'}(t, x, u) = \frac{1}{u} \mathcal{P}_{ff'}\left(t, \frac{x}{u}\right) = \frac{\alpha_S(t)}{2\pi} \frac{2}{u} P_{ff'}^{(0)}\left(t, \frac{x}{u}\right), \tag{2}$$

where $P_{ff'}^{(0)}(z)$ is the standard LL DGLAP kernel and the factor 2 is related to our definition of the evolution variable t.

In the compact operator (matrix) notation eq. (1) reads

$$\partial_t \mathbf{D}(t) = \mathbf{K}(t) \, \mathbf{D}(t). \tag{3}$$

Given a known $\mathbf{D}(t_0)$, the formal solution at any later "time" $t \geq t_0$ is provided by the time ordered exponential

$$\mathbf{D}(t) = \exp\left(\int_{t_0}^t \mathbf{K}(t')dt'\right)_T \mathbf{D}(t_0) = \mathbf{G_K}(t, t_0)\mathbf{D}(t_0). \tag{4}$$

The *time-ordered exponential* evolution operator reads[3]

$$\mathbf{G_K}(t, t_0) = \mathbf{G}(\mathbf{K}; t, t_0) = \exp\left(\int_{t_0}^t \mathbf{K}(t')dt'\right)_T = \mathbf{I} + \sum_{n=1}^{\infty} \prod_{i=1}^n \int_{t_0}^t dt_i \theta_{t_i > t_{i-1}} \mathbf{K}(t_i), \tag{5}$$

For the sake of completeness, let us write the explicit definition[4] of the multiplication operation as used and defined in eqs. (3)-(5):

$$\left(\mathbf{K}(t_2)\mathbf{K}(t_1)\right)_{f_2, f_1}(x_2, x_1) = \sum_{f'} \int_{x_2}^{x_1} dx' \; \mathcal{K}_{f_2 f'}(t_2, x_2, x') \mathcal{K}_{f' f_1}(t_1, x', x_1). \tag{6}$$

---

[3]Here and in the following we define $\prod_{i=1}^n A_i \equiv A_n A_{n-1} \ldots A_2 A_1$.

[4]In the case of QCD evolution $\mathbf{K}(t_i)$ transforms $u \in (0, 1)$ into x obeying $x < u$. This is due to 4-momentum conservation.

We shall be dealing often with the case of the kernel split into parts, for example:

$$\mathbf{K}(t) = \mathbf{K}^A(t) + \mathbf{K}^B(t). \tag{7}$$

In such a case the solution of eq. (4) can be reorganized as follows[5]

$$\mathbf{D}(t) = \mathbf{G}_{\mathbf{K}^B}(t, t_0)\,\mathbf{D}(t_0) + \sum_{n=1}^{\infty}\left[\prod_{i=1}^{n}\int_{t_0}^{t} dt_i\,\theta_{t_i > t_{i-1}}\mathbf{G}_{\mathbf{K}^B}(t_{i+1}, t_i)\mathbf{K}^A(t_i)\right]\mathbf{G}_{\mathbf{K}^B}(t_1, t_0)\,\mathbf{D}(t_0)$$

$$= \mathbf{G}_{\mathbf{K}^B}(t, t_0)\,\mathbf{D}(t_0) + \sum_{n=1}^{\infty}\left[\prod_{i=1}^{n}\int_{t_{i-1}}^{t} dt_i\right]\mathbf{G}_{\mathbf{K}^B}(t, t_n)\left[\prod_{i=1}^{n}\mathbf{K}^A(t_i)\mathbf{G}_{\mathbf{K}^B}(t_i, t_{i-1})\right]\mathbf{D}(t_0), \tag{8}$$

where $t_{n+1} \equiv t$ and $\mathbf{G}_{\mathbf{K}^B}$ is the evolution operator of eq. (5) of the evolution with the kernel $\mathbf{K}^B$. Formal proof of eq. (8) is given in ref. [16].
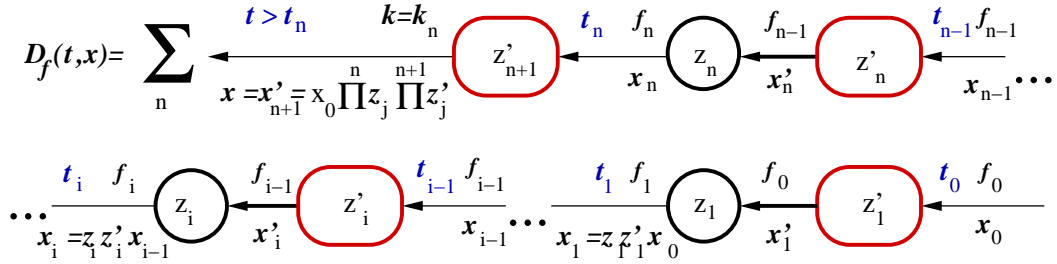

Figure 1: Scheme of integration variables and summation indices in eq. (10), circles correspond to $\mathbf{K}^A(t_i)$ and ovals to $\mathbf{G}_{\mathbf{K}^B}$, $z_i' = x_i'/x_{i-1}$ and $z_i = x_i/x_i'$.

In the following, we are going to nest eq. (8) twice. First, in order to isolate gluonstrahlung and flavour-changing parts of the evolution, with the following split of the kernel

$$\mathcal{K}(t)_{ff'} = \mathcal{K}^A(t)_{ff'} + \mathcal{K}^B(t)_{ff'} = (1 - \delta_{ff'})\mathcal{K}(t)_{ff'} + \delta_{ff'}\mathcal{K}(t)_{ff}. \tag{9}$$

In this case $\mathbf{G}_{\mathbf{K}^B}$ represents pure gluonstrahlung and is diagonal in the flavour index.

---

[5] Here and in the following, we understand that scope of the indices ceases at the closing bracket. However, validity scope of indiced variables, like t_i extends until the formula's end. The use of eq. (6) is understood to be adjusted accordingly.

Eq. (8), in the standard integro-tensorial notation looks as follows:

$$D_f(t, x) = \int_x^1 dx_0 \, G_{ff}(K^B; t, t_0; x, x_0) \, D_f(t_0, x_0) + \sum_{n=1}^{\infty} \sum_{f_{n-1}, \dots, f_1, f_0}$$

$$\times \int_x^1 dx_0 \left[\prod_{i=1}^n \int_{t_{i-1}}^t dt_i \int_0^1 dx_i \int_0^1 dx_i' \, \theta_{x_i < x_i' < x_{i-1}} \right] G_{ff}(K^B; t, t_n, x, x_n) \quad (10)$$

$$\times \left[\prod_{i=1}^n \mathcal{K}_{f_i f_{i-1}}^A(t_i, x_i, x_i') \, G_{f_{i-1} f_{i-1}}(K^B; t_i, t_{i-1}, x_i', x_{i-1}) \right] D_{f_0}(t_0, x_0),$$

where $f_n \equiv f$. In the above and the following equations we adopt the following notation:

$$\delta_{x=y} = \delta(x - y)$$

and

$$\theta_{y<x} = 1 \quad \text{for} \quad y < x \quad \text{and} \quad \theta_{y<x} = 0 \quad \text{for} \quad y \geq x.$$

Similarly, $\theta_{z<y<x} = \theta_{z<y} \theta_{y<x}$. The chain of integration variables and flavour indices is depicted schematically in Fig. 1.

Next, we use eq. (8) in order to resum the virtual IR-divergent part $\mathcal{K}^V$ of the gluon-strahlung kernel $\mathcal{K}^B$

$$\mathcal{K}_{ff'}^B(t, x, u) = \delta_{ff'} \mathcal{K}_{ff}(t, x, u) = \delta_{ff'} \left(\mathcal{K}_{ff}^V(t, x, u) + \mathcal{K}_{ff}^R(t, x, u) \right),$$
$$\mathcal{K}_{ff}^V(t, x, u) = -\delta_{x=u} \mathcal{K}_{ff}^v(t, x), \quad (11)$$
$$\mathcal{K}_{ff}^R(t, x, u) = \theta_{x<u-\Delta(x,u)} \mathcal{K}_{ff}(t, x, u),$$

where $\Delta(x, u)$ is finite IR cut-off, not necessarily infinitesimal. In order to resum (exponentiate) the virtual part $\mathcal{K}^V$ of the kernel we employ the following version of eq. (8)

$$\mathbf{G}_{\mathbf{K}^V + \mathbf{K}^R}(t, t_0) = \mathbf{G}_{\mathbf{K}^V}(t, t_0) +$$
$$+ \sum_{n=1}^{\infty} \left[\prod_{i=1}^n \int_{t_0}^t dt_i \, \theta_{t_i > t_{i-1}} \mathbf{G}_{\mathbf{K}^V}(t_{i+1}, t_i) \mathbf{K}^R(t_i) \right] \mathbf{G}_{\mathbf{K}^V}(t_1, t_0). \quad (12)$$

Since $\mathbf{K}^V(t_i)$ is diagonal in x, u and in the flavour index, and also because of

$$\{G_{\mathbf{K}^V}(t_{i+1}, t_i)\}_{ff}(x, u) = \delta_{x=u} \, e^{-\Phi_f(t_{i+1}, t_i | x)}, \qquad \Phi_f(t_{i+1}, t_i | x) = \int_{t_i}^{t_{i+1}} dt \, \mathcal{K}_{ff}^v(t, x), \quad (13)$$

we obtain immediately

$$G_{ff}(K^B; t_b, t_a, x, u) \equiv \{\mathbf{G}_{\mathbf{K}^B}(t_b, t_a)\}_{ff}(x, u) =$$
$$= e^{-\Phi_f(t_b, t_a | x)} \delta_{x=u} + \sum_{n=1}^{\infty} \left[\prod_{i=1}^n \int_{t_a}^{t_b} dt_i \, \theta_{t_i > t_{i-1}} \int_x^u dx_i \theta_{x_i < x_{i-1}} \right]$$
$$\times e^{-\Phi_f(t_b, t_n | x)} \left[\prod_{i=1}^n \mathcal{K}_{ff}^R(t_i, x_i, x_{i-1}) e^{-\Phi_f(t_i, t_{i-1} | x_{i-1})} \right] \delta_{x=x_n}, \quad (14)$$

5

where $t_{n+1} \equiv t_b$, $t_0 \equiv t_a$ and $x_0 \equiv u$. The above result can be also obtained by iterating the evolution equation for $\mathbf{G_{K^B}}(t, t_0)$ with the boundary condition $\mathbf{G_{K^B}}(t_0, t_0) = \mathbf{I}$, see for instance ref. [17].

The algebra resulting from (8) is quite general and does not rely on any particular form of the kernel. For example, in the above calculations we did not have to invoke the energy sum rules, or any other specific restrictions on the overall normalization embodied usually in the virtual corrections. Also, we did not define yet the relation between the evolution variables t_i and x_i and parton four-momenta. This will be done in the following section.

3 Evolution kernel and variables

Before we specify details of the evolution kernels used in this work, let us discuss the relation between the evolution variables x_j, t_j and the emitted parton four-momenta k_j^μ in the corresponding parton shower MC.

In the proofs of the factorization theorems [1, 18, 2] and its practical realizations like in ref. [19], typically, the within $\overline{MS}$ scheme, one projects the four-momenta of the (off-shell) partons into the 1-dimensional variable of the evolution, typically a dimensionless lightcone variable x. The so-called factorization scale $\mu_F = Q$ measuring the size of the available parton emission phase space is usually set by the kinematics of the hard process. The underlying QCD differential distribution (matrix element times the phase space) is reduced to a chain of parton splittings with the x variable of the PDF being the fraction of the initial hadron energy E_h carried by the parton entering the hard process. Variable $e^t = Q$ defines the boundary (maximum value) for many ordered variables[6] t_i. The variable t may be related directly to one of the phase space parameters (virtuality, transverse momentum, angle) or with the abstract dimensional scale variable μ_F resulting from the formal procedure of cancelling IR singularities in the dimensional regularization method. In the classical construction of the parton shower MC, one must invert the above mapping of the phase space variables into evolution variables; that is to construct parton four-momenta out of t_i and x_i and to reconstruct fully differential parton distributions in terms of these four-momenta. Obviously this procedure is not unique and requires guidance from the detailed knowledge of the structure of the IR singularities[7] of the original QCD matrix element.

In this work we shall construct the CMC algorithm for two new types of generalized kernels, in addition to LL DGLAP of eq. (2) for which a CMC was constructed already in refs. [8] and [9]. The new kernels are based on the following generic form

$$
\begin{aligned}
\mathcal{K}_{ff'}(t, x, u) &= \frac{\alpha_S(t, x, u)}{\pi} \frac{1}{u} P_{ff'}^{(0)}\left(t, \frac{x}{u}\right) = -\mathcal{K}_{ff}^v(t, u)\delta_{ff'}\delta_{x=u} + \mathcal{K}_{ff'}^\theta(t, x, u), \\
\mathcal{K}_{ff'}^\theta(t, x, u) &= \mathcal{K}_{ff'}(t, x, u)\theta_{u \geq x + \Delta(t, u)},
\end{aligned}
\tag{15}
$$

---

[6] This simplified picture is valid at least in the leading-logarithmic approximation.

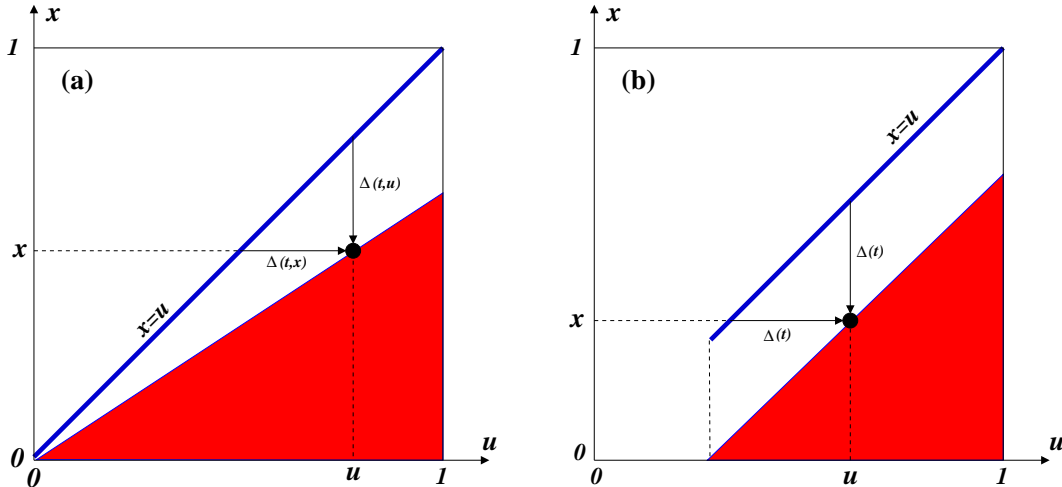[7] We understand IR singularities as both collinear and soft ones.

Figure 2: Two types of the infrared (IR) boundaries on the $u - x$ plane (a) $u - x > u\epsilon$ and (d) $u - x > \epsilon$ for the triangle (red) which depicts the area where real emission part $\mathcal{K}^\theta$ is nonzero. The diagonal line (blue) represents places where virtual $\mathcal{K}^v$ is nonzero.

where $P_{ff'}^{(0)}(t, z)$ is the standard LL kernel (DGLAP) and t-dependence enters into it only through IR regulator $\Delta(t, u)$. The new kernel types correspond to different choices of the argument of the strong coupling constant and to different forms of the $\Delta(t, u)$ regulator. The two new types of $\Delta(t, u)$ regulators used in this work are depicted on the $x - u$ plane in Fig. 2 and will be defined in the following Section.

The other important departure from DGLAP is the strong coupling constant $\alpha_S(t, x, u)$ which now may depend on all evolution variables. We shall consider strong coupling constant α_S depending on $z = x/u$ or on the transverse momentum k^T defined below. Before we define the evolution kernel in a detail, we have to elaborate first the mapping of the evolution variables into four-momenta.

3.1   Relating evolution variables to four-momenta

The essential decision in the construction of the parton shower MC concerns the choice of a kinematics variable in the solutions of the evolution equations (10) and (14); this choice is in one-to-one correspondence with the evolution time variable t_i and the limiting value t. We chose to associate t with the rapidity (angle) of the emitted parton, following the well known arguments on the colour coherence exposed in many papers, see for instance refs. [20, 21, 22] and further references therein.

We define the lightcone variables $q^\pm = q^0 \pm q^3$ and normalize the parton momenta with respect to the energy E_h of the initial massless hadron, see Fig. 3,

$$q_h^+ = 2E_h, \quad \text{and} \quad q_i^+ = x_i 2E_h, \quad \text{where} \quad 0 \leq x_i \leq 1. \tag{16}$$

These relations hold in the rest frame of the hard process system or the center of mass system, where the initial state hadron h moves in the direction of the z-axis.
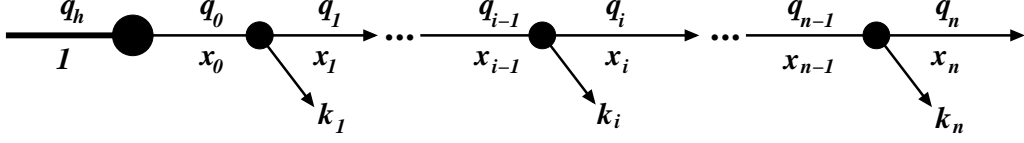
Figure 3: Kinematics in the evolution history (tree).

In particular, for the parton initiating a parton cascade, we have

$$q_0^+ = x_0 2E_h. \tag{17}$$

This parton has negligible transverse momentum[8], $k_0^T = 0$. In the RHS each on-shell i-th *emitted particle* will take away a part of the lightcone variable

$$k_i^+ = q_{i-1}^+ - q_i^+ = (x_{i-1} - x_i)2E_h. \tag{18}$$

The above is not enough to define k^μ. For this we need to define at least $k^T = |\vec{k}^T|$ or rapidity $\eta_i = \frac{1}{2}\ln(k_i^+/k_i^-)$. Once the azimuthal angle φ_i is added, we can complete the mapping from the evolution variables to the 4-momentum of the i-th emitted parton $k_i^\mu(t_i, x_i, x_{i-1}, \varphi_i)$.

If we associate the evolution time t_i with rapidity η_i then the following relation

$$k_i^T = \sqrt{k_i^+ k_i^-} = k_i^+ \sqrt{\frac{k_i^-}{k_i^+}} = k_i^+ e^{-\eta_i} = (x_{i-1} - x_i)2E_h e^{-\eta_i}, \tag{19}$$

valid for $k_i^2 = 0$, provides the transverse momentum and thus k_i^μ. We can also eliminate k_i^T with the help of the following conventional relation which defines the evolution time t

$$k_i^T \equiv e^{t_i}(x_{i-1} - x_i). \tag{20}$$

Note that this relation translates into $k^T = e^t(u - x)$ in the general evolution equation of eq. (1). We observe that evolution time t_i and rapidity η_i are related by a linear transformation of the following explicit form:

$$e^{t_i} = e^{\ln(2E_h) - \eta_i} \quad \Rightarrow \quad \eta_i = \ln(2E_h) - t_i. \tag{21}$$

The above relation is the main result of this section.

Summarizing, the mapping of t_i and x_i into 4-momenta $k_i^\mu, i = 1, 2, \ldots n$ can be written now in an explicit manner:

$$\begin{aligned} k_i^+ &= (x_{i-1} - x_i)2E_h, & k_i^T &= (x_{i-1} - x_i)e^{t_i}, & k_i^- &= (k_i^T)^2/k_i^+, \\ k_i^0 &= (k_i^+ + k_i^-)/2, & k_i^3 &= (k_i^+ - k_i^-)/2. \end{aligned} \tag{22}$$

_____

[8]In the realistic MC k_0^T will be distributed according to a Gaussian profile of the width of order $\sim \lambda$.

Last but not least, we have to define also the phase space limits, $t_i \in (t_{\min}, t_{\max})$ and $\eta_i \in (\eta_{\min}, \eta_{\max})$. One has to be very careful at this step. The maximum evolution time $t_{\max}$ (minimum rapidity $\eta_{\min}$) is set by the requirement that all emitted partons are confined to the forward hemisphere[9], $90° \geq \theta_i$, or equivalently $\eta_i \geq \eta_{\min} = 0$. This implies

$$t_{\max} = \ln(2E_h), \quad E_h = \frac{1}{2}e^{t_{\max}}. \tag{23}$$

Minimum evolution time is determined by the phase space opening point for the first emission, due to minimum transverse momentum $k^T_{\min} = \lambda$:

$$e^{t_1}x_0 \geq \lambda. \tag{24}$$

This leads to

$$t_1 > t_\lambda - \ln x_0, \quad t_\lambda \equiv \ln \lambda$$

and therefore to

$$t > t_n > t_{n-1} > \ldots > t_2 > t_1 > t_\lambda - \ln x_0 \equiv t_{\min}.$$

This automatically determines the maximum rapidity

$$\eta_{\max} = \ln(2E_h) - \ln \lambda + \ln x_0.$$

Altogether

$$\begin{aligned}
t_i &\in (t_{\min}, t_{\max}) = (\ln \lambda - \ln x_0, \ln(2E_h)), \\
\eta_i &\in (\eta_{\min}, \eta_{\max}) = (0, \ln(2E_h) + \ln x_0 - \ln \lambda)), \\
\eta_{\max} - \eta_{\min} &= t_{\max} - t_{\min} = \ln(2E_h/\lambda).
\end{aligned} \tag{25}$$

Let us remark that the naive assignment $t_{\max} = \ln(E_h)$, without the factor of 2, would lead, because of eq. (21), to a partial coverage of the forward hemisphere only, $\eta_i \geq \ln(2)$. On the other hand, this factor of 2 may look justified; the absolute kinematic range of the transverse momentum is $k^T_i \in (\lambda, k^T_{\max})$, where $k^T_{\max} = e^{t_{\max}} = 2E_h$ results from the relation $k^T_i \equiv e^{t_i}(x_{i-1} - x_i)$ and energy conservation $x_i \leq 1$. The reader may notice that this limit is a factor of 2 higher than in the familiar inequality $k^T_i \leq E_h$ resulting from the 4-momentum conservation operating in both hemispheres simultaneously, i.e. on k^+_i and k^-_i. However our limit, including the factor of 2, is valid for a single hemisphere separated from activity on the other side!

Finally, we illustrate the real emission phase space in Fig. 4(a) using the rapidity variable η and the log of transverse momentum k^T. Directions of the lightcone variables $k^\pm$ are also indicated. In this figure we indicate, as black numbered points, momenta of three example emitted partons. Available phase space is limited from below by $k^T_{\min} = \lambda$, while

---

[9]Sharp minimum/maximum rapidity is essential for avoiding mismatch between the parton distributions from two independent constrained MCs "operating" in the backward and forward hemispheres for the initial stated radiation.
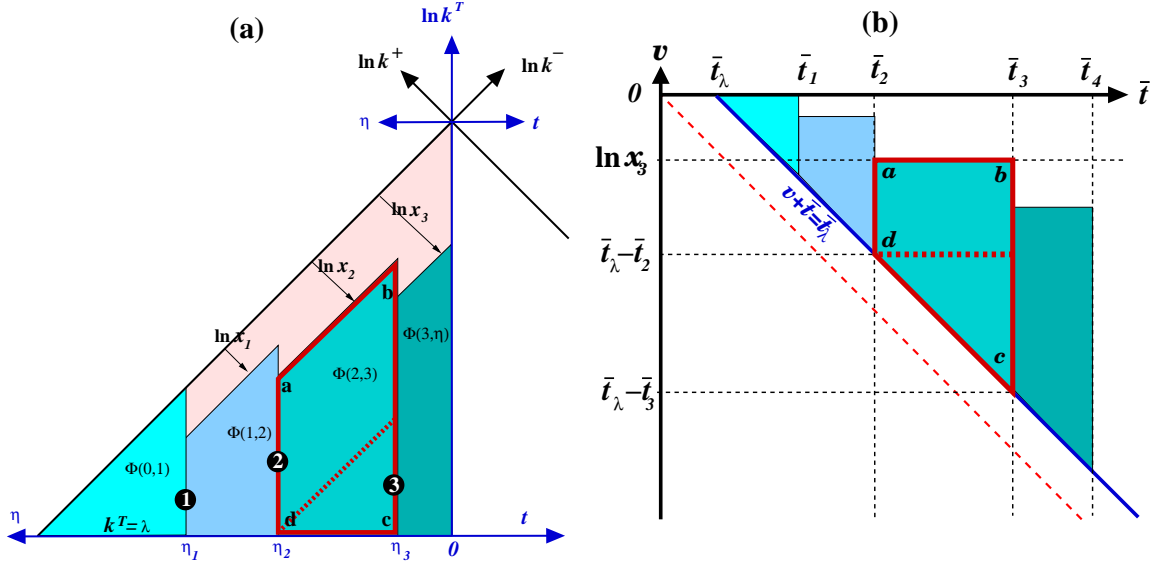
9

Figure 4: Sudakov plane parametrized using two different sets of variables (t, v) and $(\eta, \ln k^T)$.

in the direction k^+ the boundary is controlled by the total available q^+, which is diminished by a factor x_{i-1} at i-the step. (For simplicity we have set $x_0 = 1$.) The minimum rapidity $\eta = 0$ limits the emission phase space from the right hand side. The triangle and trapezoids show integration domains of the consecutive formfactors $\Phi_f(t_i, t_{i-1}|x_{i-1})$ in eq. (14), see also Section 3.3 and Appendix A.4.

The above mapping of the evolution variables into four-momenta, with the minimum k^T and built-in angular ordering, is identical to the one used in refs. [23, 24, 25], up to the following redefinition of the evolution time: $\hat{t}_i = \ln E_h - \eta_i + \ln x_{i-1} = t_i + \ln x_{i-1}$. For the redefined $\hat{t}$ we have

$$\hat{t} + \ln x > \hat{t}_n + \ln x_n > \hat{t}_{n-1} + \ln x_{n-1} > \ldots > \hat{t}_2 + \ln x_2 > \hat{t}_1 + \ln x_1 > t_\lambda,$$

and consequently $\hat{q}_i > \hat{q}_{i-1} z_{i-1}$, with $z_i = x_i / x_{i-1}$.

3.2 Evolution kernels

Once the phase-space parametrization is explained, we may define our choices for the evolution kernels, introduced so far only in the generic form in eq. (15). The kernels to be used in the CMC in this work will be of three kinds. The main difference between them is in the choice of the variable used as an argument of the coupling constant α_S. The appearance of the Landau pole in α_S will limit the choice of the IR cut-off in the multigluon phase space. Let us define first the gluonstrahlung kernel $f = f'$, where f is flavour type, $f = G, q, \bar{q}$, in all three cases.

Case (A): Standard DGLAP LL of ref. [8], which is used here as a reference case:

$$\mathcal{K}_{ff}^{(A)\theta}(t, x, u) = \frac{\alpha_S(e^t)}{\pi} \frac{1}{u} P_{ff}^{(0)}(z) \, \theta_{1-z \geq \epsilon} = \frac{\alpha_S(e^t)}{\pi} \frac{1}{u} P_{ff}^{(0)}(x/u) \, \theta_{u-x \geq u\epsilon}, \tag{26}$$

10

where ϵ is infinitesimally small and $z = x/u$.

Case (B): The argument in α_S is $(1-z)e^t = k^T/u$; such a choice was already advocated in the early work of ref. [26]. For the IR cut-off we use $\Delta(t, u) = \lambda u e^{-t}$, where $\lambda > \Lambda_0$. It cannot be infinitesimally small, however, it becomes vary small at large t. Using $z = x/u$ we define

$$
\begin{aligned}
\mathcal{K}_{ff}^{(B)\theta}(t, x, u) &= \frac{\alpha_S^{(0)}((1-z)e^t)}{\pi} \frac{1}{u} P_{ff}^{(0)}(z)\, \theta_{1-z \geq \lambda e^{-t}} \\
&= \frac{\alpha_S^{(0)}((1-x/u)e^t)}{\pi} \frac{1}{u} P_{ff}^{(0)}(x/u)\, \theta_{u-x \geq u\lambda e^{-t}}\ .
\end{aligned}
\tag{27}
$$

Case (C): The coupling constant α_S depends on the transverse momentum $k^T = (u-x)e^t$, while for an IR cut-off we choose $\Delta(t, u) = \Delta(t) = \lambda e^{-t}$. The kernel reads:

$$
\mathcal{K}_{ff}^{(C)\theta}(t, x, u) = \frac{\alpha_S^{(0)}((u-x)e^t)}{\pi} \frac{1}{u} P_{ff}^{(0)}(x/u)\theta_{u-x \geq \lambda e^{-t}},
\tag{28}
$$

For the (B) and (C) cases the choice of λ for the IR cut-off must be such that we avoid the Landau pole in the coupling constant

$$
\alpha_S^{(0)}(q) = \frac{2\pi}{\beta_0} \frac{1}{\ln q - \ln \Lambda_0},
\tag{29}
$$

taken in the LL approximation; however, the insertion of the $1 - z$ factor to the argument of the coupling constant resums higher order effects in the bremsstrahlung parts of the evolution [26]. In the non-diagonal, flavour-changing, elements of the kernel, there is no need for this kind of resummation. We can, therefore, use $\alpha_S(e^t)$. On the other hand, although quark-gluon transition kernel elements have neither IR divergence nor a Landau pole, it makes sense to keep the restriction $u - x < \Delta(t, u)$ because the cut out part of the phase space is (will be) already populated by the k^T distribution of the primordial parton[10].

3.3 Virtual part of the kernel and formfactors

For the evolution with any type of kernel the momentum sum rule

$$
\begin{aligned}
0 = \partial_t \sum_f \int dx\, x D_f(t, x) &= \sum_f \int_0^1 du \left\{ \sum_{f'} \int_0^u dx\, x \mathcal{K}_{ff'}(t, x, u) \right\} D_{f'}(t, u) \\
&= \sum_f \int_0^1 du \left\{ -u \mathcal{K}_{ff}^v(t, u) + \sum_{f'} \int_0^u dx\, x \mathcal{K}_{ff'}^\theta(t, x, u) \right\} D_{f'}(t, u)
\end{aligned}
\tag{30}
$$

is imposed, the same as in the reference DGLAP case.

---

[10]Without this restriction in the early evolution time quark-gluon transitions would completely take over the bremsstrahlung.

This sum rule determines unambiguously the virtual part of the kernel for all (A-C) cases

$$\mathcal{K}^v_{ff}(t, u) = \sum_{f'} \int_0^u \frac{dx}{u} x \, \mathcal{K}^\theta_{f'f}(t, x, u). \tag{31}$$

It should be stressed, that in case (C) the $\mathcal{K}^v(t, u)$ includes implicitly $\theta_{u>\Delta(t)}$, as visualized in Fig. 2(B). The following Sudakov formfactor results immediately:

$$\Phi_f(t_1, t_0|u) = \int_{t_0}^{t_1} dt \, \mathcal{K}^v_{ff}(t, u) = \sum_{f'} \int_{t_0}^{t_1} dt \int_0^u \frac{dx}{u} x \, \mathcal{K}^\theta_{f'f}(t, x, u)$$

$$= \sum_{f'} \int_{t_0}^{t_1} dt \int_0^u \frac{dy}{u}(u - y) \, \mathcal{K}^\theta_{f'f}(t, u - y, u) = \sum_{f'} \int_{t_0}^{t_1} dt \int_0^1 dz \, uz\mathcal{K}^\theta_{f'f}(t, uz, u), \tag{32}$$

where $z \equiv x/u$ and $y \equiv u - x = (1 - z)u$. The $\mathcal{K}^\theta_{f'f}$ is constructed using IR-singular, non-singular and flavour-changing parts of the DGLAP (LL) kernel accordingly to the following decomposition

$$zP^\theta_{f'f}(z) = \left[\delta_{f'f} \left(\frac{A_{ff}}{1 - z} + F_{ff}(z) \right) + (1 - \delta_{f'f})F_{f'f}(z) \right] \theta_{u-x>\Delta(u)}, \tag{33}$$

For the list of the coefficients A_{ff} and functions $F_{f'f}(z)$ see Appendix of ref. [17].

We also need to define the generalized kernels beyond the case of bremsstrahlung, for the quark-gluon transitions. One of the possible extensions, valid for all three cases $X = A, B, C$, reads

$$x\mathcal{K}^{\theta,(X)}_{f'f}(t, x, u) = \delta_{f'f} \, x\mathcal{K}^{\theta(A)}_{f'f}(t, x, u) + (1 - \delta_{f'f})\frac{\alpha_S(e^t)}{\pi}F_{f'f}(z)\theta_{u-x>\Delta^{(X)}(u)}, \tag{34}$$

where α_S in the flavour changing elements has no z- or k^T-dependence and the IR cut-off $\Delta^{(X)}$ is the same as in the bremsstrahlung case. The Sudakov formfactors resulting from the above kernels are split into three corresponding parts:

$$\Phi_f(t_1, t_0|u) = \boldsymbol{\Phi}_f(t_1, t_0|u) + \Phi^b_f(t_1, t_0|u) + \Phi^c_f(t_1, t_0|u). \tag{35}$$

In case (C) we get three genuinely u-dependent components

$$\boldsymbol{\Phi}_f(t_1, t_0|u) = \int_{t_0}^{t_1} dt \int_0^1 dz \, \frac{\alpha_S((1 - z)ue^t)}{\pi} \frac{A_{ff}}{1 - z} \theta_{(1-z)u>\lambda e^{-t}}$$

$$= A_{ff}\frac{2}{\beta_0}\varrho_2(\bar{t}_0 + \ln u, \bar{t}_1 + \ln u; \bar{t}_\lambda),$$

$$\Phi^b_f(t_1, t_0|u) = \int_{t_0}^{t_1} dt \int_0^1 dz \, \frac{\alpha_S((1 - z)ue^t)}{\pi}F_{ff}(z) \, \theta_{(1-z)u>\lambda e^{-t}} \, , \tag{36}$$

$$\Phi^c_f(t_1, t_0|u) = \int_{t_0}^{t_1} dt \, \frac{\alpha_S(e^t)}{\pi} \sum_{f'\neq f} \int_0^1 dz \, F_{f'f}(z) \, \theta_{(1-z)u>\lambda e^{-t}},$$

12

where $\bar{t}_i \equiv t_i - \ln\Lambda_0$, $\bar{t}_\lambda \equiv t_\lambda - \ln\Lambda_0$ and function ϱ_2 is defined in Appendix A.2 in terms of log functions. The integration domains for the consecutive formfactors of the above type are also shown in a pictorially way in Fig. 4, using the well known logarithmic Sudakov plane.

In case (B) the u dependence disappears due to the fact that both $\alpha_S((1-z)e^t)$ and $\theta_{1-z>\Delta(t)}$ depend on u exclusively through $z = x/u$:

$$\boldsymbol{\Phi}_f(t_1, t_0) = \boldsymbol{\Phi}_f(t_1, t_0|1), \qquad \Phi_f^b(t_1, t_0) = \Phi_f^b(t_1, t_0|1), \qquad \Phi_f^c(t_1, t_0) = \Phi_f^c(t_1, t_0|1). \quad (37)$$

In other words, setting $u = 1$ brings us from case (C) to the case (B).

The reason behind the seemingly at hoc three-fold split of $\boldsymbol{\Phi}_f(t_1, t_0)$ is practical. In the MC, the formfactor has to be calculated event per event. One dimensional integration for each MC event is acceptable – it does not slow down MC generation noticeably. Here, the most singular part of $\boldsymbol{\Phi}_f$ is calculable analytically. In Φ_f^b we are able to integrate analytically over t and the integration over z is done numerically while in Φ_f^c we can integrate analytically over z and the integration over t is has to be done numerically (see also Appendix A.3). Altogether, we are thus able to avoid 2-dimensional numerical integration for each MC event (or use of look-up tables and interpolation for fast evaluation of the Sudakov formfactors for each MC event).

Finally, formfactors for the simplest DGLAP LL case read as follows:

$$\boldsymbol{\Phi}_f(t_1, t_0) = \frac{2}{\beta_0} \left(\tau(t_1) - \tau(t_0)\right) A_{ff} \ln\frac{1}{\epsilon},$$

$$\Phi_f^b(t_1, t_0) = \frac{2}{\beta_0} \left(\tau(t_1) - \tau(t_0)\right) \int_0^1 dz \, F_{ff}(z) \, \theta_{1-z>\epsilon} , \qquad (38)$$

$$\Phi_f^c(t_1, t_0) = \frac{2}{\beta_0} \left(\tau(t_1) - \tau(t_0)\right) \sum_{f' \neq f} \int_0^1 dz \, F_{f'f}(z) \, \theta_{1-z>\epsilon} ,$$

where $\tau(t) = \ln(t - \ln\Lambda_0)$.

At present, in the MC implementation, we use for cases (B) and (C), a slightly different form of the quark-gluon changing kernels elements:

$$x\mathcal{K}_{f'f}^{\theta(B')}(t, x, u) = \delta_{f'f} \, x\mathcal{K}_{f'f}^{\theta(B)}(t, x, u) + (1 - \delta_{f'f}) \frac{\alpha_S((1-z)e^t)}{\pi} F_{f'f}(z)\theta_{1-z>\lambda e^{-t}} ,$$

$$x\mathcal{K}_{f'f}^{\theta(C')}(t, x, u) = \delta_{f'f} \, x\mathcal{K}_{f'f}^{\theta(C)}(t, x, u) + (1 - \delta_{f'f}) \frac{\alpha_S(u(1-z)e^t)}{\pi} F_{f'f}(z)\theta_{y>\lambda e^{-t}} , \qquad (39)$$

that is, we use the same arguments of α_S as for gluonstrahlung. We will refer to them as cases (B') and (C'). This is done mainly to facilitate numerical comparisons with our Markovian MCs. One can easily go back from cases (B') and (C') to (B) and (C) with an extra (well behaving) MC weight, if needed. The corresponding formfactor $\Phi_f^c(t_1, t_0)$ gets properly redefined in cases (B') and (C'), of course.

4 Constrained Monte Carlo for pure bremsstrahlung

Let us discuss the case of pure gluonstrahlung first. We will focus our attention on the following integral, being part of eq. (14)

$$G_{ff}^n(K^B; t_b, t_a, x, u) = \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t_b} dt_i \int_x^u dx_i \, \mathcal{K}_{ff}^R(t_i, x_i, x_{i-1}) \right] e^{-\sum_{n=1}^{n+1} \Phi_{f_{i-1}}(t_i, t_{i-1}|x_{i-1})} \delta_{x=x_n}.$$

(40)

It describes the emission of n gluons. The following "aliasing" of variables is used: $t_{n+1} \equiv t_b$, $t_0 \equiv t_a$ and $x_0 \equiv u$. The integrand is well approximated by the product of the IR singularities in terms of variables $y_i = x_i - x_{i-1}$

$$\prod_{i=1}^{n} \mathcal{K}_{ff}^R(x_i, x_{i-1}) \simeq \prod_{i=1}^{n} \frac{1}{x_i - x_{i-1}} = \prod_{i=1}^{n} \frac{1}{y_i} \simeq \prod_{i=1}^{n} \frac{1}{1 - z_i}.$$

(41)

Hence, switching from variables x_i to y_i or z_i is almost mandatory and the multigluon distribution with the δ-function constraining the total energy of emitted gluons takes the following symmetric form:

$$\int \delta_{x=x_n} \prod_{i=1}^{n} \frac{dx_i}{x_i - x_{i-1}} = \int \delta \left(u - x - \sum_{j=1}^{n} y_j \right) \prod_{i=1}^{n} \frac{dy_i}{y_i} \simeq \int \delta \left(x - u \prod_{j=1}^{n} z_j \right) \prod_{i=1}^{n} \frac{dz_i}{1 - z_i}.$$

(42)

Constructing the MC program/algorithm for the multidimensional distribution featuring such a δ-function is a hard technical problem and it is the problem of constructing a Constrained Monte Carlo (CMC).

It should be kept in mind, that it is possible, as shown in ref. [9], to generate the above distribution in x-space without such a δ-function, provided G_{ff} is convoluted with the power-like function x_0^ω. Such a solution was labeled as CMC class II, while the CMC of this paper was already referred to in ref. [9] as CMC class I.

In ref. [8] the first CMC algorithm of the class I was found and tested for DGLAP kernel, that is for our case (A). This algorithm is based on the observation that for the product of steeply rising functions, proportional to $1/y_i$, the δ-function constraint is effectively resolved by a single (let's say) y_k, while all *other* y_i, can be considered as unconstrained. We shall extend the CMC class I solution to more complicated kernels, that is of our type (B) and (C). The main complication with respect to case (A) is due to a more complicated singular y-dependence (or z-dependence) entering through the coupling constant α_S, and even more important, through formfactors. In addition, our new CMCs will not only generalize the solutions of ref. [8], but will be described in such a way that any future extension to other types of evolution kernels will be rather easy. In the following sub-chapters we shall present the details of our generalized solutions in detail.

4.1 Generic CMC class I

As already indicated, we intend to introduce the formulation of the CMC algorithm which covers three types of kernels (A)-(C), such that further extensions are possible and easy. At first, let us consider the following generic expression including the sum of constrained multidimensional integrals

$$D(v) = \Psi'(v)e^{-\int_{v_0}^{v_x} K(v')dv'}\left\{\delta_{\Psi(v)=\Psi(v_0)} + \sum_{n=1}^{\infty}\frac{1}{n!}\left[\int_{v_0}^{v}\prod_{i=1}^{n}K(v_i)dv_i\right]\delta_{\Psi(v)=\sum_{j=1}^{n}\Psi(v_j)}\right\}. \quad (43)$$

The following properties will be assumed: (a) The function $\Psi(v)$ must be monotonous with v and its derivative must remain non-negative, $\Psi'(v) \geq 0$, for all $v > v_0$. (b) The choice of the term[11] $\delta(v - v_0)$ to fix $v = v_0$ is a convention. In practical realization it represents a no-emission event[12]. It can be placed at any point outside the (v_0, v_x) range. (c) variable v_x controlling overall normalization will be defined later on, i.e. adjusted to get convenient normalization, (d) v_0 will be associated with the IR boundary (e) the true upper integration limit of v_i is below v and is in fact determined by the δ-function.

In the following examples, the variable v_i will be defined as $v_i = \ln(y_i) = \ln(x_i - x_{i-1})$ or $v_i = \ln(1 - z_i) = \ln(1 - x_i/x_{i-1})$, while the function $\Psi(v)$ will be typically rather simple; $\Psi(y_i) = y_i$ or $\Psi(z_i) = \ln(z_i)$.

Assuming $K(v) > 0$, we can define a mapping (and its inverse) which removes $K(v)$ from the integrand:

$$r_i = R(v_i) = \int_{v_0}^{v_i} K(v')dv', \qquad v_i = V(r_i) = R^{-1}(r_i). \quad (44)$$

Our master formula transforms then as follows:

$$D(v) = \Psi'(v)e^{-R(v_x)}\left\{\delta_{\Psi(v)=\Psi(v_0)} + \sum_{n=1}^{\infty}\frac{1}{n!}\left[\int_{0}^{R(v)}\prod_{i=1}^{n}dr_i\right]\delta_{\Psi(v)=\sum_{j=1}^{n}\Psi(R^{-1}(r_j))}\right\}. \quad (45)$$

Function $f(r) = \Psi(R^{-1}(r))$ is usually a very steeply growing function of r, hence the constraint is effectively resolved by a single $r_j \simeq R(v)$, the biggest one. Let us exploit this fact in order to replace the complicated constraint with the simpler one[13] $\delta(R(v) - \max_j r_j)$. This is done in three steps. *Step one:* introduce a new auxiliary integration variable X countered by a δ-function:

$$D(v) = e^{-R(v_x)}\delta_{v=v_0} + \Psi'(v)e^{-R(v_x)}$$
$$\times \sum_{n=1}^{\infty}\frac{1}{n!}\int dX\left[\int_{0}^{R(v)}\prod_{i=1}^{n}dr_i\right]\delta_{\Psi(v)=\sum_{j=1}^{n}\Psi(R^{-1}(r_j))}\delta_{R(v)=X+\max_j r_j}. \quad (46)$$

---

[11]This term is, of course, equivalent to $\delta_{\Psi(v)=\Psi(v_0)}$.

[12]"No-emission" event will have precise interpretation in a given environment.

[13]Function $\max_j r_j$ is equal the biggest among r_j, $j = 1, 2, ..., n$.

Step two: Change variables $r_i = r'_i - X$:

$$
D(v) = e^{-R(v_x)}\delta_{v=v_0} + \Psi'(v)e^{-R(v_x)}
$$
$$
\times \sum_{n=1}^{\infty} \frac{1}{n!} \int dX \left[\int_0^{R(v)} \prod_{i=1}^{n} dr'_i \, \theta_{r_i>0} \right] \delta_{\Psi(v)=\sum_{j=1}^{n} \Psi(R^{-1}(r'_j-X))} \delta_{R(v)=\max_j r'_j}. \tag{47}
$$

From now on we have to watch out for the condition $r_i = r'_i - X > 0$ explicitly[14]. *Step three:* Eliminate old constraint by integrating over X

$$
D(v) = e^{-R(v_x)}\delta_{v=v_0} + e^{-R(v_x)} \sum_{n=1}^{\infty} \frac{1}{n!} \left[\int_0^{R(v)} \prod_{i=1}^{n} dr'_i \, \theta_{r_i>0} \right] \delta_{R(v)=\max_j r'_j} \, \mathfrak{J}, \tag{48}
$$

where $X_0(r'_1, r'_2, ..., r'_n)$ is found by means of solving numerically (iterative method) the original constraint for X. The Jacobian factor $\mathfrak{J}$ is defined as follows

$$
\mathfrak{J} = \frac{\Psi'(v)}{\partial_X \sum_{j=1}^{n} \Psi(R^{-1}(r'_j - X_0))} = \frac{\Psi'(v)}{\sum_{j=1}^{n} \Psi'(v_j)(R'(v_j))^{-1}}. \tag{49}
$$

In the above factor the j-th term satisfying $R(v) = \max_j r'_j$ dominates the sum in the numerator; consequently $\mathfrak{J} \simeq R'(v) = K(v)$. This form, valid in the limit only, we keep explicitly in the integrand, while the remaining part of $\mathfrak{J}$ is incorporated in a complicated but mild Monte Carlo weight:

$$
D(v) = e^{-R(v_x)}\delta_{v=v_0} + R'(v)e^{-R(v_x)} \sum_{n=1}^{\infty} \frac{1}{n!} \left[\int_0^{R(v)} \prod_{i=1}^{n} dr'_i \right] \delta_{\max_j r'_j = R(v)} \, w^{\#}, \tag{50}
$$

where the MC weight is

$$
w^{\#} = \frac{\Psi'(v)}{R'(v) \sum_{j=1}^{n} \Psi'(v_j)(R'(v_j))^{-1}} \prod_{i=1}^{n} \theta_{r_i>0}. \tag{51}
$$

Last, but not least, we isolate cleanly a part of the integrand normalized properly to one with the help of the integration variable $\xi_i = r_i/R(v)$ and Poisson distribution $P(n|\lambda) = \exp(-\lambda)\lambda^n/n!$:

$$
D(v) = e^{-R(v_x)}\delta_{v=v_0} + R'(v)e^{R(v)-R(v_x)} \sum_{n=1}^{\infty} P(n-1|R(v)) \left[\prod_{i=1}^{n} \int_0^1 d\xi_i \right] \frac{\delta_{\max_j \xi_j = 1}}{n} \, w^{\#}(\xi),
$$
$$
\tag{52}
$$

The nice thing is that $\bar{D}(v)$, obtained by neglecting $w^{\#}$, is known analytically

$$
\bar{D}(v) = D(v)|_{w^{\#}=1} = e^{-R(v_x)}\delta_{v=v_0} + \theta_{v>v_0} R'(v)e^{R(v)-R(v_x)} \tag{53}
$$

---

[14]On the other hand no problem with $r'_i \leq v$, because we shall get $X \geq 0$.

and is normalized to one

$$\int_{v_0}^{v_x} \bar{D}(v)dv = e^{-R(v_x)} + \int_{\exp(-R(v_x))}^{1} d\left(e^{R(v)-R(v_x)}\right) = e^{-R(v_x)} + \left(1 - e^{-R(v_x)}\right) = 1. \quad (54)$$

The above convenient unitary normalization is achieved by means of identifying v_x with the upper limit of v, see also below for particular realizations. In the implementation of quark-gluon transitions using FOAM [27] (see Section 5) one introduces integration variable $U = U(v) = e^{R(v)-R(v_x)} \in (0,1)$ and the above equation transforms into

$$\int_{0}^{\exp(-R(v_x))} dU + \int_{\exp(-R(v_x))}^{1} dU = \int_{0}^{1} dU. \quad (55)$$

The MC procedure of generating variables v, n and $(v_1, v_2, ..., v_n)$ obeying the constraint is the following:

- Generate v according to $\bar{D}(v)$ times whatever the other function of v in the Monte Carlo problem, for that purpose use variable $U = e^{R(v)-R(v_x)} \in (0,1)$.

- If $U \le e^{-R(v_x)}$ then set $v = v_0$ and $n = 0$ (no-emission event).

- Otherwise U is translated into v ($v > v_0$) and $n > 0$ is generated according to $P(n-1|R(v))$.

- Generate variables $\xi_i \in (0,1)$ $i = 1, 2, ..., n$, except one of them $\xi_j = 1$, where $j = 1, 2, ...n$ is chosen randomly with uniform probability.

- Variables ξ_i are now translated into r'_i and the transcendental equation defining X_0 shift is solved numerically.

- Once X_0 is known, then all $v_i(r_i(r'(i(\xi_i)))$ are calculated.

- MC weight $w^{\#}$ is evaluated. The check on $r_i > 0$ can be done earlier.

The above algorithm generates weighted MC events $(n; v_1, v_2, ..., v_n)$ exactly according to resumed series of the integrands defining $D(v)$.

4.2 Treatment of t-ordering, generic case

In case (A) of the DGLAP kernel, discussed in ref. [8], the t-ordering in the integrals of eq. (40) can be easily traded for a $1/n!$ factor, while $K(v)$ in the previous section includes un-ordered integrals $\int dt_i$ over the entire available range for each t_i. However, even in this simple case one has to be careful if one aims not only at the numerical evaluation of the PDF, but also at the proper simulation of the entire MC events $\{n, (t_1, x_1), (t_2, x_2), ..., (t_n, x_n)\}$. The CMC class I algorithm of ref. [8] (DGLAP) is formulated in terms of $z_i = x_i/x_{i-1}$; the point is that at the end of the MC algorithm one has to calculate x_i using a well defined ordering of z_i, which is exactly the same as in

the sequence of the (originally) un-ordered t_i. In practice, in the end of the MC generation, one has to order t_i and z_i *simultaneously* before calculating $x_i = x_0 z_1 z_2 ... z_i$. This is because the original integrand is symmetric with respect to interchange of the pairs of variables $(t_i, z_i) \leftrightarrow (t_k, z_k)$ and not with respect to interchanging $t_i \leftrightarrow t_k$ or $z_i \leftrightarrow z_k$ done independently. This property we will call *pairwise* permutation symmetry.

In our most general case (C), the integrand is not *pairwise* symmetric, mainly due to nontrivial x_i-dependency in the formfactor, see eq. (40). In this case, the strategy is such that we introduce a simplified integrand, with simplified formfactor and simplified kernel $\bar{\mathbb{P}}$ (the IR-singular part of the kernel), such that the simplified integrand features *pairwise* permutation symmetry. The above simplifications are immediately compensated by means of the MC weight $w^{\mathbb{P}}$, which absorbs all possible *pairwise* non-symmetry.

Let us translate what was said above into rigorous algebra. We start from a more sophisticated variant of our generic multi-integral (43), covering all cases (A-C) and possibly other similar cases, but this time including explicitly ordered t-integrations:

$$
D(t, t_0 | v) = \Psi'(v) e^{-\int_{t_0}^{t} dt' \int_{v_0(t')}^{vx} \bar{\mathbb{P}}(v', t') dv'} \Bigg\{ \delta_{\Psi(v) = \Psi(v_0)} +
$$
$$
+ \sum_{n=1}^{\infty} \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t} dt_i \int_{v_0(t_i)}^{v} dv_i \, \bar{\mathbb{P}}(t_i, v_i) \right] w^{\mathbb{P}}(\mathbf{t}, \mathbf{v}) \, \delta_{\Psi(v) = \sum_{j=1}^{n} \Psi(v_j)} \Bigg\}, \tag{56}
$$

where $t_n \equiv t$. The bold-face variable $\mathbf{t}$ denotes the entire vector $(t_1, t_2, ..., t_n)$ and similar convention is used for the definition of vector $\mathbf{v}$.

In order to get rid of the t-ordering in the basic MC algorithm we proceed carefully step by step as follows:

1. Introduce formally *pairwise* symmetrization, i.e. we sum up over all permutations $\mathfrak{P}$ of the pairs of the variables (v_i, t_i), compensating by means of a $1/n!$ factor:

$$
\frac{1}{n!} \sum_{\mathfrak{P}} \left[\prod_{i=1}^{n} \int_{t_0}^{t} dt_i^{\mathfrak{P}} \int_{v_0(t_i^{\mathfrak{P}})}^{v} dv_i^{\mathfrak{P}} \, \bar{\mathbb{P}}(t_i^{\mathfrak{P}}, v_i^{\mathfrak{P}}) \right] \theta_{t_n^{\mathfrak{P}} > t_{n-1}^{\mathfrak{P}} > ... > t_1^{\mathfrak{P}}} \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P}}, \mathbf{v}^{\mathfrak{P}}).
$$

2. The part of the integrand in the brackets is now perfectly *pairwise* symmetric and the permutation $\mathfrak{P}$ can be undone in this part:

$$
\frac{1}{n!} \left[\prod_{i=1}^{n} \int_{t_0}^{t} dt_i \int_{v_0(t_i)}^{v} dv_i \, \bar{\mathbb{P}}(t_i, v_i) \right] \sum_{\mathfrak{P}} \theta_{t_n^{\mathfrak{P}} > t_{n-1}^{\mathfrak{P}} > ... > t_1^{\mathfrak{P}}} \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P}}, \mathbf{v}^{\mathfrak{P}}).
$$

3. The sum over permutations $\sum_{\mathfrak{P}}$ can be dropped out because only one permutation contributes at a given point $\mathbf{t} = (t_i, t_2, ..., t_n)$. This particular permutation we denote $\mathfrak{P}_{\mathbf{t}}$ obtaining:

$$
D(t, t_0 | v) = \Psi'(v) e^{-\int_{t_0}^{t} dt' \int_{v_0(t')}^{vx} \bar{\mathbb{P}}(v', t') dv'} \Bigg\{ \delta_{\Psi(v) = \Psi(v_0)} +
$$
$$
+ \sum_{n=1}^{\infty} \frac{1}{n!} \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t} dt_i \int_{v_0(t_i)}^{v} dv_i \, \bar{\mathbb{P}}(t_i, v_i) \right] \delta_{\Psi(v) = \sum_{j=1}^{n} \Psi(v_j)} \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P}_{\mathbf{t}}}, \mathbf{v}^{\mathfrak{P}_{\mathbf{t}}}) \Bigg\}. \tag{57}
$$

18

4. In the basic MC the weight $w^{\mathbb{P}}$ is temporarily neglected and t_i are generated unordered. The permutation $\mathfrak{P}_{\mathbf{t}}$ is then read from the ordering in $\mathbf{t} = (t_1, t_2, ..., t_n)$. It is then used to construct the sequence of x_i out of v_i and to calculate $w^{\mathbb{P}}$.

The final step in bringing eq. (57) into the standardized form of eq. (43) is the interchange of the integration order and integration over t_i

$$\int_{t_{i-1}}^{t} dt_i \int_{v_0(t_i)}^{v} dv_i \, \bar{\mathbb{P}}(t_i, v_i) = \int_{v_0}^{v} dv_i \int_{t_{i-1}(v_i)}^{t(v_i)} dt_i \, \bar{\mathbb{P}}(t_i, v_i) = \int_{v_0}^{v} dv_i \, K(v_i) \int_0^1 d\sigma_i,$$

The additional standard mapping

$$\sigma(t, v) = \frac{\int_{t_{\min}(v)}^{t} dt' \, \bar{\mathbb{P}}(t', v)}{\int_{t_{\min}(v)}^{t_{\max}(v)} dt' \, \bar{\mathbb{P}}(t', v)} = \frac{\int_{t_{\min}(v)}^{t} dt' \, \bar{\mathbb{P}}(t', v)}{K(v)} \tag{58}$$

allows to map uniformly generated σ_i into t_i, for every value of v_i, for a particular realization see Appendix A.3. Luckily, $\sigma_i(t_i)$ can be inverted analytically, i.e. $t_i(\sigma_i, v_i)$ is available as an analytical formula, in all cases (A-C).

Summarizing, we obtain the following standardized generic formula ready for the MC implementation:

$$D(t, t_0|v) = \Psi'(v) e^{- \int_{v_0}^{v_x} K(v')dv'} \left\{ \delta_{\Psi(v)=\Psi(v_0)} + \right.$$

$$\left. + \sum_{n=1}^{\infty} \frac{1}{n!} \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t} dt_i \, K(v_i) \int_0^1 d\sigma_i \right] \delta_{\Psi(v)=\sum_{j=1}^{n} \Psi(v_j)} \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P}_{\mathbf{t}}}, \mathbf{v}^{\mathfrak{P}_{\mathbf{t}}}) \right\}$$

$$= e^{-R(v_x)} \delta_{v=v_0} + \theta_{v>v_0} R'(v) e^{R(v)-R(v_x)}$$

$$\times \sum_{n=1}^{\infty} P(n-1|R(v)) \left[\prod_{i=1}^{n} \int_0^1 d\xi_i \int_0^1 d\sigma_i \right] \frac{\delta_{\max_j \xi_j = 1}}{n} \, w^{\#}(\xi) \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P}_{\mathbf{t}}}, \mathbf{v}^{\mathfrak{P}_{\mathbf{t}}}). \tag{59}$$

In the following subsection we shall describe three realizations of the above CMC class I schemes for three types of the kernels, (A-C).

4.3 CMC case (A), DGLAP

Let us start with the particular realization of the above CMC for the easiest case of the DGLAP kernel, type (A). Such a CMC was exposed in detail already in ref. [8]. Here it serves as a reference case and a warm-up example. Let us start from pure bremsstrahlung evolution operator of eq. (14):

$$\frac{x}{u} G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a)} \delta_{x=u} + \sum_{n=1}^{\infty} \left[\prod_{i=1}^{n} \int_{t_a}^{t_b} dt_i \, \bar{\theta}_{t_i>t_{i-1}} \int_x^u dx_i \right]$$

$$\times e^{-\Phi_f(t_b, t_n)} \left[\prod_{i=1}^{n} \frac{x_i}{x_{i-1}} \mathcal{K}_{ff}^R(t_i, x_i, x_{i-1}) e^{-\Phi_f(t_i, t_{i-1})} \right] \delta_{x=x_n}, \tag{60}$$

19

where we have dropped the non-existing dependence on x_i in the formfactor Φ. Hence, one may exploit the relation $\sum_{i=0}^{n} \Phi(t_i, t_{i-1}) = \Phi(t_b, t_a)$. After identifying

$$x_i \mathcal{K}_{ff}^R(t_i, x_i, x_{i-1}) = \alpha_S(e^t) z_i P(z_i), \qquad z_i = x_i / x_{i-1}$$

a simplified expression is obtained:

$$\frac{x}{u} G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a)} \delta_{x=u} +$$
$$+ e^{-\Phi_f(t_b, t_a)} \sum_{n=1}^{\infty} \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t_b} dt_i \int_{x/u}^{1} dz_i \, \alpha_S(e^{t_i}) z_i P_{ff}(z_i) \theta_{1-z_i < \epsilon} \right] \delta_{x=u \prod_{j=1}^{n} z_j}. \tag{61}$$

In the next step, the simplified kernel is introduced

$$\prod_{i=1}^{n} \int^{1-\epsilon} dz_i \, \alpha_S(e^{t_i}) z_i P_{ff}(z_i) \rightarrow \prod_{i=1}^{n} \int^{1-\epsilon} \frac{dz_i}{1 - z_i} \alpha_S(e^{t_i}) A_{ff} =$$
$$= \prod_{i=1}^{n} \int_{\ln \epsilon} dv_i \, \alpha_S(e^{t_i}) A_{ff} = \prod_{i=1}^{n} \int_{v_0} dv_i \, \bar{\mathbb{P}}(t_i, v_i), \tag{62}$$
$$\bar{\mathbb{P}}(t_i, v_i) = \alpha_S(e^{t_i}) A_{ff}, \quad v_i = \ln(1 - z_i), \quad v_0 = \ln \epsilon.$$

Once v is defined, we are ready to deduce our kernel $K(v)$ and constraint function $\Psi(v)$ from

$$\frac{x}{u} G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a)} \frac{1}{x} \delta_{\ln(x/u)=1}$$
$$+ e^{-\Phi_f(t_b, t_a)} \sum_{n=1}^{\infty} \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t_b} dt_i \int_{v_0}^{\ln(1-x/u)} dv_i \, \bar{\mathbb{P}}(t_i, v_i) \right] \frac{1}{x} \delta_{\ln(x/u)=\sum_{j=1}^{n} \ln(1-\exp(v_j))} \, w_1^{\bar{\mathbb{P}}}, \tag{63}$$

where

$$w_1^{\bar{\mathbb{P}}} = \prod_{i=1}^{n} \frac{\alpha_S(e^{t_i}) z_i (1 - z_i) P_{ff}(z_i)}{\bar{\mathbb{P}}(t_i, v_i)}. \tag{64}$$

While comparing the above expressions with eq. (56) one can easily identify the following components:

$$v = \ln(1 - x/u), \quad \Psi(v) = \ln(1 - e^v), \quad \Psi'(v) = \frac{e^v}{1 - e^v} = \frac{u - x}{x},$$
$$K(v) = \int_{t_a}^{t_b} dt \, \bar{\mathbb{P}}(t_i, v_i) = A_{ff} \frac{2}{b_0} \big(\tau(t_b) - \tau(t_a) \big). \tag{65}$$

The expression for the basic formfactor follows:

$$R(v) = \int_{v_0}^{v} dv' \, K(v') = \int_{v_0}^{v} dv' \int_{t_a}^{t_b} dt' \, \bar{\mathbb{P}}(t', v') = A_{ff} \frac{2}{b_0} \big(\tau(t_b) - \tau(t_a) \big) (v - v_0). \tag{66}$$

20

Variable v_x represents the upper boundary of $v = \ln(1 - x/u)$. For fixed x and maximal $u = 1$ we have $v_x = \ln(1 - x)$. With all the above elements at hand, we are able to complete the following standardized formula

$$\frac{x}{u} G_{ff}(K^B; t_b, t_a, x, u) = \frac{1}{x} \frac{1}{\Psi'(v)} \left\{ e^{-R(v_x)} \delta_{v=v_0} + \theta_{v>v_0} R'(v) e^{R(v)-R(v_x)} \right.$$
$$\left. \times \sum_{n=1}^{\infty} P(n-1|R(v)) \left[\prod_{i=1}^{n} \int_0^1 d\xi_i \int_0^1 d\sigma_i \right] \frac{\delta_{\max_j \xi_j = 1}}{n} w^{\#}(\xi) \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P_t}}, \mathbf{v}^{\mathfrak{P_t}}) \right\}. \tag{67}$$

where

$$w^{\bar{\mathbb{P}}} = e^{R(v_x)-\Phi_f(t_b, t_a)} \prod_{i=1}^{n} \frac{\alpha_S(e^{t_i}) z_i (1-z_i) P_{ff}(z_i)}{\bar{\mathbb{P}}(t_i, v_i)} \tag{68}$$

and the second weight $w^{\#}$ is fully determined from eq. (51), given that $\Psi(v)$ and $K(v)$ are already defined.

In the CMC, we usually integrate over u for fixed x, hence the following formula is useful

$$\int_x^1 du \, G_{ff}(K^B; t_b, t_a, x, u) = \int_0^1 dU \left(\frac{u}{x}\right)^2 \left\{ \theta_{U \leq \exp(-R(v_x))} |_{v=v_0} + \theta_{U > \exp(-R(v_x))} \right.$$
$$\left. \times \sum_{n=1}^{\infty} P(n-1|R(v)) \left[\prod_{i=1}^{n} \int_0^1 d\xi_i \int_0^1 d\sigma_i \right] \frac{\delta_{\max_j \xi_j = 1}}{n} w^{\#}(\xi) \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P_t}}, \mathbf{v}^{\mathfrak{P_t}}) \right\}. \tag{69}$$

where[15] $U = U(x, u) = e^{R(v)-R(v_x)} = e^{R(\ln(1-x/u))-R(\ln(1-x))}$. Because $R(v_0) = 0$, the point $v = v_0$ (IR boundary) translates into $U_0 = e^{-R(v_x)} = e^{-R(\ln(1-x))}$. With all elements needed in eq. (59) in hand, we are ready to reconstruct from our generic formulation the complete CMC class I algorithm of ref. [8], at least for pure bremsstrahlung.

4.4 CMC case (B), $\alpha_S(e^t(1 - z))$

This case is to some extent similar to the previous DGLAP case. We shall, therefore, concentrate on the differences. The starting point is now

$$\frac{x}{u} G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a|1)} \delta_{x=u} + \sum_{n=1}^{\infty} \left[\prod_{i=1}^{n} \int_{t_a}^{t_b} dt_i \, \theta_{t_i > t_{i-1}} \int_x^u dx_i \right]$$
$$\times e^{-\Phi_f(t_b, t_n|1)} \left[\prod_{i=1}^{n} \frac{x_i}{x_{i-1}} \mathcal{K}_{ff}^R(t_i, x_i, x_{i-1}) e^{-\Phi_f(t_i, t_{i-1}|1)} \right] \delta_{x=x_n}, \tag{70}$$

where one may again combine formfactors into a single one: $\sum_{i=0}^{n} \Phi(t_i, t_{i-1}|1) = \Phi(t_b, t_a|1)$. Now, in terms of

$$x_i \mathcal{K}_{ff}^R(t_i, x_i, x_{i-1}) = \alpha_S(e^t(1-z_i)) z_i P(z_i), \qquad z_i = x_i/x_{i-1},$$

---

[15] Another ingredient was the identity $\frac{R'(v)}{\Psi'(v)} = \left(\frac{d\Psi}{du}\right)^{-1} \frac{dR}{du} = \frac{1}{x} \frac{dR}{du}$.

the simplified expression reads

$$\frac{x}{u}G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a|1)}\Big\{ \delta_{x=u} +$$

$$+ \sum_{n=1}^{\infty} \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t_b} dt_i \int_{x/u}^{1} dz_i \, \alpha_S((1-z_i)e^{t_i})z_i P_{ff}(z_i)\theta_{1-z_i < \lambda e^{-t_i}} \right] \delta_{x=u\prod_{j=1}^{n} z_j} \Big\}. \tag{71}$$

Simplification of the kernel goes as follows:

$$\prod_{i=1}^{n} \int^{1-\lambda e^{-t_i}} dz_i \, \alpha_S((1-z_i)e^{t_i})z_i P_{ff}(z_i) \to \prod_{i=1}^{n} \int^{1-\lambda e^{-t_i}} \frac{dz_i}{1-z_i} \, \alpha_S((1-z_i)e^{t_i})A_{ff} =$$

$$= \prod_{i=1}^{n} \int_{\ln\lambda - t_i} dv_i \, \alpha_S(e^{t_i+v_i})A_{ff} = \prod_{i=1}^{n} \int_{v_0(t_i)} dv_i \, \bar{\mathbb{P}}(t_i, v_i), \tag{72}$$

where

$$\bar{\mathbb{P}}(t_i, v_i) = \alpha_S(e^{t_i+v_i})A_{ff}, \quad v_i = \ln(1-z_i), \quad v_0(t) = \ln\lambda - t. \tag{73}$$

So far we have followed closely the DGLAP case, except of the more complicated IR cut-off and extra factor in the argument of α_S. The choice of variable v is the same, hence also the function $\Psi(v)$ remains unchanged:

$$\frac{x}{u}G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a)}\frac{1}{x}\Big\{ \delta_{\ln(1-e^v)=1} +$$

$$+ \sum_{n=1}^{\infty} \left[\prod_{i=1}^{n} \int_{t_{i-1}}^{t_b} dt_i \int_{v_0(t_i)}^{v} dv_i \, \bar{\mathbb{P}}(t_i, v_i) \right] \delta_{\ln(1-e^v)=\sum_{j=1}^{n}\ln(1-e^{v_j})} \, w_1^{\bar{\mathbb{P}}} \Big\}, \tag{74}$$

where

$$w_1^{\bar{\mathbb{P}}} = \prod_{i=1}^{n} \frac{\alpha_S(e^{t_i+v_i})z_i(1-z_i)P_{ff}(z_i)}{\bar{\mathbb{P}}(t_i, v_i)}. \tag{75}$$

While comparing the above expressions with eq. (56) we immediately identify the following components:

$$v = \ln(1-x/u), \quad \Psi(v) = \ln(1-e^v), \quad \Psi'(v) = \frac{e^v}{1-e^v} = \frac{u-x}{x}. \tag{76}$$

In principle, the evaluation of $K(v)$ requires a change of the integration order used in the formfactor integral, see also Appendix A.3,

$$R(v; t_b, t_a) = \int_{t_a}^{t_b} dt' \int_{\ln\lambda - t'}^{v} dv' \, \bar{\mathbb{P}}(t', v') = \int_{v_0}^{v} dv' \int_{\max(t_a, \ln\lambda - v')}^{t_b} dt' \, \bar{\mathbb{P}}(t', v') = \int_{v_0}^{v} dv' \, K(v'). \tag{77}$$

22

In practice, it is slightly easier to calculate $R(v; t_b, t_a)$ with the t-integration external, and then obtain $K(v)$ by differentiation:

$$R(v; t_b, t_a) = A_{ff}\frac{2}{b_0}\int_{t_a}^{t_b} dt' \int_{\ln\lambda - t'}^{v} dv' \frac{1}{t' + v' - \ln\Lambda_0} = A_{ff}\frac{2}{b_0}\varrho_2(\bar{t}_b + v, \bar{t}_a + v; \bar{t}_\lambda), \quad (78)$$

and

$$K(v) = \partial_v R(v; t_b, t_a) = A_{ff}\frac{2}{b_0}\partial_v \varrho_2(\bar{t}_b + v, \bar{t}_a + v; \bar{t}_\lambda). \quad (79)$$

The functions ϱ_2 and $\partial_v \varrho_2$ are given in the Appendices A.2 and A.3. The rest of the algebra is almost the same as in DGLAP case:

$$\frac{x}{u}G_{ff}(K^B; t_b, t_a, x, u) = \frac{1}{x}\frac{1}{\Psi'(v)}\left\{e^{-R(v_x)}\delta_{v=v_0} + \theta_{v>v_0}R'(v)e^{R(v)-R(v_x)}\right.$$
$$\left. \times \sum_{n=1}^{\infty} P(n-1|R(v))\left[\prod_{i=1}^{n}\int_0^1 d\xi_i \int_0^1 d\sigma_i\right]\frac{\delta_{\max_j \xi_j=1}}{n} w^{\#}(\xi)\, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P}_t}, \mathbf{v}^{\mathfrak{P}_t})\right\}. \quad (80)$$

The variable $v_x = \ln(1 - x)$ and weights $w^{\#}$ and $w^{\mathbb{P}}$ are defined in the same way as discussed already in the case (A) of DGLAP. One has to remember only that v_i enters into $\alpha_S(e^{t_i+v_i})$. The final integral form coincides with that for DGLAP, see eq. (69). The important differences are in the definitions of the components; in particular the function $R(v)$ is here more complicated.

4.5 CMC case (C), $\alpha_S(k^T)$, CCFM

For this case we are dealing with the most general implementation of the generic solution defined in eq. (14):

$$\frac{x}{u}G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a|x)}\delta_{x=u}$$
$$+ \sum_{n=1}^{\infty}\left[\prod_{i=1}^{n}\int_{t_a}^{t_b} dt_i\,\theta_{t_i>t_{i-1}}\int_x^u dx_i\frac{x_i}{x_{i-1}}\mathcal{K}_{ff}^R(t_i, x_i, x_{i-1})\right]e^{-\sum_{j=0}^{n}\Phi_f(t_j, t_{j-1}|x_{j-1})}\delta_{x=x_n}, \quad (81)$$

where we are aliasing the following variables: $x = x_n$, $u = x_0$, $t_0 = t_a$ and $t_n = t_b$. The kernel is again more complicated than the one used in the previous Sections

$$x_i\mathcal{K}_{ff}^R(t_i, x_i, x_{i-1}) = \alpha_S(e^t(1 - z_i)/x_{i-1})\,z_i P(z_i), \qquad z_i = x_i/x_{i-1}.$$

With the help of the following transformation of the integration variables

$$\int dx_i\frac{x_i}{x_{i-1}}\frac{\theta_{x_{i-1}-x_i>\lambda e^{-t}}}{1 - z_i} = \int dx_i\frac{\theta_{x_{i-1}-x_i>\lambda e^{-t}}}{x_{i-1} - x_i}x_i = \int_{\lambda e^{-t}}\frac{dy_i}{y_i}x_i, \qquad y_i \equiv x_{i-1} - x_i,$$

23

we obtain

$$\frac{x}{u}G_{ff}(K^B; t_b, t_a, x, u) = e^{-\Phi_f(t_b, t_a|u)}\delta_{x=u}+$$

$$+\sum_{n=1}^{\infty}\left[\prod_{i=1}^{n}\int_{t_{i-1}}^{t_b}dt_i\int_{\lambda e^{-t_i}}^{1}\frac{dy_i}{y_i}\,\alpha_S(y_ie^{t_i})z_i(1-z_i)P_{ff}(z_i)\right]e^{-\sum_{j=0}^{n}\Phi_f(t_j, t_{j-1}|x_{j-1})}\delta_{x-u=\sum_{j=1}^{n}y_j}.$$

$$(82)$$

The simplification for the kernels and formfactor, to be compensated later on by the MC weight, reads as follows:

$$\prod_{i=1}^{n}\int_{\lambda e^{-t_i}}\frac{dy_i}{y_i}\,\alpha_S(y_ie^{t_i})z_i(1-z_i)P_{ff}(z_i)e^{-\sum_{j=0}^{n}\Phi_f(t_j, t_{j-1}|x_{j-1})}\longrightarrow$$

$$\longrightarrow e^{-\boldsymbol{\Phi}_f(t_b, t_a|1-x)}\prod_{i=1}^{n}\int_{\lambda e^{-t_i}}\frac{dy_i}{y_i}\,\alpha_S(y_ie^{t_i})A_{ff} =$$

$$= e^{-\boldsymbol{\Phi}_f(t_b, t_a|1-x)}\prod_{i=1}^{n}\int_{\ln\lambda - t_i}dv_i\,\alpha_S(e^{t_i+v_i})A_{ff} = e^{-\boldsymbol{\Phi}_f(t_b, t_a|1-x)}\prod_{i=1}^{n}\int_{v_0(t_i)}dv_i\,\bar{\mathbb{P}}(t_i, v_i),$$

$$(83)$$

$$\bar{\mathbb{P}}(t_i, v_i) = \alpha_S(e^{t_i+v_i})A_{ff}, \quad v_i = \ln(y_i), \quad v_0(t) = \ln\lambda - t.$$

where $\boldsymbol{\Phi}_f$ is that of eq. (36). With the above definitions we obtain

$$\frac{x}{u}G_{ff}(K^B; t_b, t_a, x, u) = e^{-\boldsymbol{\Phi}_f(t_b, t_a|1-x)}h(x, u)\left\{\delta_{\exp(v)=1}+\right.$$

$$\left.+\sum_{n=1}^{\infty}\left[\prod_{i=1}^{n}\int_{t_{i-1}}^{t_b}dt_i\int_{v_0(t_i)}^{v}dv_i\,\bar{\mathbb{P}}(t_i, v_i)\right]\delta_{\exp(v)=\sum_{j=1}^{n}\exp(v_j)}\,w^{\bar{\mathbb{P}}}\right\},$$

$$(84)$$

where

$$v = \ln(u-x), \quad \Psi(v) = e^v, \quad \Psi'(v) = e^v = u-x, \qquad (85)$$

and

$$w^{\bar{\mathbb{P}}} = e^{\boldsymbol{\Phi}_f(t_b, t_a|u)-\sum_{j=0}^{n}\Phi_f(t_j, t_{j-1}|x_{j-1})}\prod_{i=1}^{n}\frac{\alpha_S(e^{t_i+v_i})z_i(1-z_i)P_{ff}(z_i)}{\bar{\mathbb{P}}(t_i, v_i)},$$

$$h(x, u) = e^{\boldsymbol{\Phi}_f(t_b, t_a|1-x)-\boldsymbol{\Phi}_f(t_b, t_a|u)}.$$

$$(86)$$

The factor[16] $h(x, u)$ is compensating for the deliberate use of $\boldsymbol{\Phi}_f(t_b, t_a|u)$ in the weight $w^{\bar{\mathbb{P}}}$, with the aim of ensuring $w^{\bar{\mathbb{P}}} \leq 1$. Since the simplified kernel $\bar{\mathbb{P}}$ is the same as in the previous case (B), the standardized kernel and formfactor are also the same:

$$R(v; t_b, t_a) = A_{ff}\frac{2}{b_0}\varrho_2(\bar{t}_b + v, \bar{t}_a + v; \bar{t}_\lambda), \quad K(v) = \partial_v R(v; t_b, t_a), \qquad (87)$$

---

[16]It will be efficiently dealt with by the general purpose MC tool FOAM [27], see the next Section.

Moreover, the upper phase space boundary is also the same $v_x = \ln(1 - x)$, hence

$$R(v_x; t_b, t_a) = \mathbf{\Phi}_f(t_b, t_a | v_x).$$

The standardized formula for the MC reads as follows

$$
\begin{aligned}
\frac{x}{u} G_{ff}(K^B; t_b, t_a, x, u) &= \frac{h(x, u)}{\Psi'(v)} \bigg\{ e^{-R(v_x)} \delta_{v=v_0(t_b)} + \theta_{v>v_0(t_b)} R'(v) e^{R(v)-R(v_x)} \\
&\times \sum_{n=1}^{\infty} P(n - 1 | R(v)) \bigg[\prod_{i=1}^{n} \int_0^1 d\xi_i \int_0^1 d\sigma_i \bigg] \frac{\delta_{\max_j \xi_j = 1}}{n} \, w^{\#}(\xi) \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P_t}}, \mathbf{v}^{\mathfrak{P_t}}) \bigg\}.
\end{aligned}
\tag{88}
$$

The final integral, defining the distribution to be implemented in the CMC, takes the following form:

$$
\begin{aligned}
\int_x^1 du\, G_{ff}(K^B; t_b, t_a, x, u) &= \int_0^1 dU\, \frac{u}{x} h(x, u) \bigg\{ \theta_{U<\exp(-R(v_x))} |_{v=v_0(t_b)} + \theta_{U>\exp(-R(v_x))} \\
&\times \sum_{n=1}^{\infty} P(n - 1 | R(v)) \bigg[\prod_{i=1}^{n} \int_0^1 d\xi_i \int_0^1 d\sigma_i \bigg] \frac{\delta_{\max_j \xi_j = 1}}{n} \, w^{\#}(\xi) \, w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P_t}}, \mathbf{v}^{\mathfrak{P_t}}) \bigg\},
\end{aligned}
\tag{89}
$$

where[17] $U = U(x, u) = e^{R(v)-R(v_x)} = e^{R(\ln(u-x))-R(\ln(1-x))}$. The weight $w^{\#}$ is evaluated according to eq. (51). All important differences with the previous cases (A) and (B) are hidden in the definitions/constructions of the components of the above formula. The only explicit difference is in the presence of the factor $h(x, u)$.

4.6 Summary on CMC for bremsstrahlung

X	v	v_i	$\Psi(v)$	v_0	$\mathbb{P}(t_i, v_i)$	$w^{\mathbb{P}}$	$K(v), R(v)$
A	$1 - \frac{x}{u}$	$1 - \frac{x_i}{x_{i-1}}$	$\ln\left(1 - e^v\right)$	$\ln \epsilon$	$\alpha_S(e^{t_i}) A_{ff}$	Eq.(64)	Eqs.(65,66)
B	$1 - \frac{x}{u}$	$1 - \frac{x_i}{x_{i-1}}$	$\ln\left(1 - e^v\right)$	$\ln \lambda - t$	$\alpha_S(e^{t_i+v_i}) A_{ff}$	Eq.(75)	Eqs.(79,78)
C	$u - x$	$x_{i-1} - x_i$	e^v	$\ln \lambda - t$	$\alpha_S(e^{t_i+v_i}) A_{ff}$	Eq.(86)	Eq.(87)

Table 1: For three types of the evolution kernel $\mathcal{K}^{(X)}$, $X = A, B, C$ list of components in the generic eq. (59) for CMC. Variable $v_x = \ln(1 - x)$ is always the same.

Before we extend our CMC to the complete evolution with quark-gluon transitions let us summarize the case of pure bremsstrahlung. As we have seen, all three cases of kernels (A-C) are compatible with the generic formula of eq. (59), provided we identify (define) properly all components there. These components are collected and compared in Table 1 for all three cases (A-C). The appearance of the extra factor $h(x, u)$ should be kept in mind.

---

[17]In this case $R'(v)/\Psi'(v) = dR/du$.

5 Complete CMC with quark-gluon transitions

Monte Carlo simulation/integration of variables related to quark-gluon transitions is managed by the general purpose MC tool FOAM [27]. It has to be provided with the user-defined integrand, the so-called *density function*. Arguments of this function have to be inside a unit hyperrectangle (or simplex). Starting from eq. (10) we are going to reorganize explicitly its integration variables, see also the scheme in Fig. 1, paying attention to the integration order:

$$
D_f(t,x) = \int_x^1 dx_0 \, G_{ff}(K^B; t, t_0; x, x_0) \, D_f(t_0, x_0) +
$$

$$
+ \sum_{n=1}^{\infty} \sum_{f_{n-1}, \ldots, f_1, f_0} \left[\prod_{k=1}^{n} \int_{t_{k-1}}^{t} dt_k \right] \int_x^1 dx_n \, G_{ff}(K^B; t, t_n, x, x_n)
$$

$$
\times \left[\prod_{k=1}^{n} \int_{x_k}^{1} dx'_k \, \mathcal{P}^A_{f_k f_{k-1}}(t_k, x_k, x'_k) \int_{x'_k}^{1} dx_{k-1} \, G_{f_{k-1} f_{k-1}}(K^B; t_k, t_{k-1}, x'_k, x_{k-1}) \right] D_{f_0}(t_0, x_0),
$$

$$\tag{90}$$

where $f_n \equiv f$. Here, and in the following, we understand that the following integration order for all multiple integrations[18] is used.

$$
\prod_{i=1} \int d\alpha_i = \int d\alpha_n \int d\alpha_{n-1} \ldots \int d\alpha_2 \int d\alpha_1.
$$

Let us now re-parametrize the above integral into a form better suited for integration by FOAM, that is in terms of $3n+1$ variables: $U_0 \in (0,1)$ and $U_k, \alpha_k, \beta_k \in (0,1), k = 1, 2, \ldots n$:

$$
D_f(t,x) = \int_0^1 dU_0(x, x_0) \, H^{(X)}(x_0, x) \sum\!\!\!\!\!\!\int dG^0 \, W_0^G D_f(t_0, x_0) +
$$

$$
+ \sum_{n=1}^{N} \sum_{f_{n-1}, \ldots, f_1, f_0} \left[\prod_{k=1}^{n} \int_0^1 (t - t_{k-1}) d\alpha_k \right] \int_0^1 dU_n(x, x_n) \, H^{(X)}(x_n, x) \sum\!\!\!\!\!\!\int dG^{(n)} \, W_n^G
$$

$$
\times \left[\prod_{k=1}^{n} \int_0^1 d\beta_k \, (1 - x_k) \mathcal{K}^A_{f_k f_{k-1}}(t_k, x_k, x'_k(\beta_k)) \right.
$$

$$
\left. \times \int_0^1 dU_{k-1}(x'_k, x_{k-1}) \, H^{(X)}(x_{k-1}, x'_k) \sum\!\!\!\!\!\!\int dG^{(k-1)} \, W_{k-1}^G \right] D_{f_0}(t_0, x_0),
$$

$$\tag{91}$$

---

[18] This is the same order as for the operator products and in the time-ordered exponentials.

where for all three cases $X = A, B, C$, multi-differentials dG are defined[19], in the following way

$$\int_x^1 du\ G_{ff}(K^B; t_b, t_a, x, u) = \int_0^1 dU\ H^{(X)}(x, u) \oint dG\ W^G \qquad (92)$$

and

$$\oint dG \equiv \theta_{U < \exp(-R(v_x))}|_{v=v_0(t_b)} +$$

$$+ \theta_{U > \exp(-R(v_x))} \sum_{n=1}^{\infty} P(n-1|R(v)) \left[\prod_{i=1}^n \int_0^1 d\xi_i \int_0^1 d\sigma_i \right] \frac{\delta_{\max_j \xi_j = 1}}{n}, \qquad (93)$$

$$\int_0^1 dU \oint dG \equiv 1, \qquad W^G = w^\#(\xi)\ w^{\mathbb{P}}(\mathbf{t}^{\mathfrak{P}_\mathbf{t}}, \mathbf{v}^{\mathfrak{P}_\mathbf{t}}).$$

For our three example kernels we have

$$H^{(A)}(x, u) = H^{(B)}(x, u) = \left(\frac{u}{x}\right)^2,$$

$$H^{(C)}(x, u) = \frac{u}{x}\ h(x, u). \qquad (94)$$

The summation over the number of quark-gluon transitions n is also mapped into one of the FOAM variables, after truncation to a finite N. In fact, the precision level of $\sim 10^{-4}$ is achieved for $N = 4$, see numerical results below. Following the prescription of ref. [8] the sum $\sum_{f_{n-1}, \dots, f_1, f_0}$ can be reduced just to a single term. Also, as in ref. [8], the additional mapping to $\tau_i = \tau(t_i)$ variables is done in order to compensate partly for the t-dependence of $\alpha_S(t_i)$ in $\mathcal{P}$ kernels. The purpose of this mapping is to boost slightly the efficiency of FOAM. Let us stress that FOAM generates all $3N + 2$ variables, neglecting the MC weight

$$W = \prod_{i=0}^n W_i^G. \qquad (95)$$

The calculation of this weight can be completed only after gluons are generated for all $n + 1$ bremsstrahlung segments first. Also, as in ref. [8], FOAM treats distributions as continuous in U_i-variables, ignoring δ-like structure in $x'_k - x_{k-1}$, corresponding to the no-emission case. This is very important and powerful trick. The δ-like no-emission terms are replaced by the integrals over finite intervals in U_i, exactly as in eq. (55).

With all the above formalism at hand, we can formulate our CMC algorithm similarly as in the general LL DGLAP case:

- Generate superlevel variables n, f_i, t_i x'_i and x_i with the help of general-purpose MC tool FOAM according to eq. (91), and neglecting $W = \prod W_i^G$.

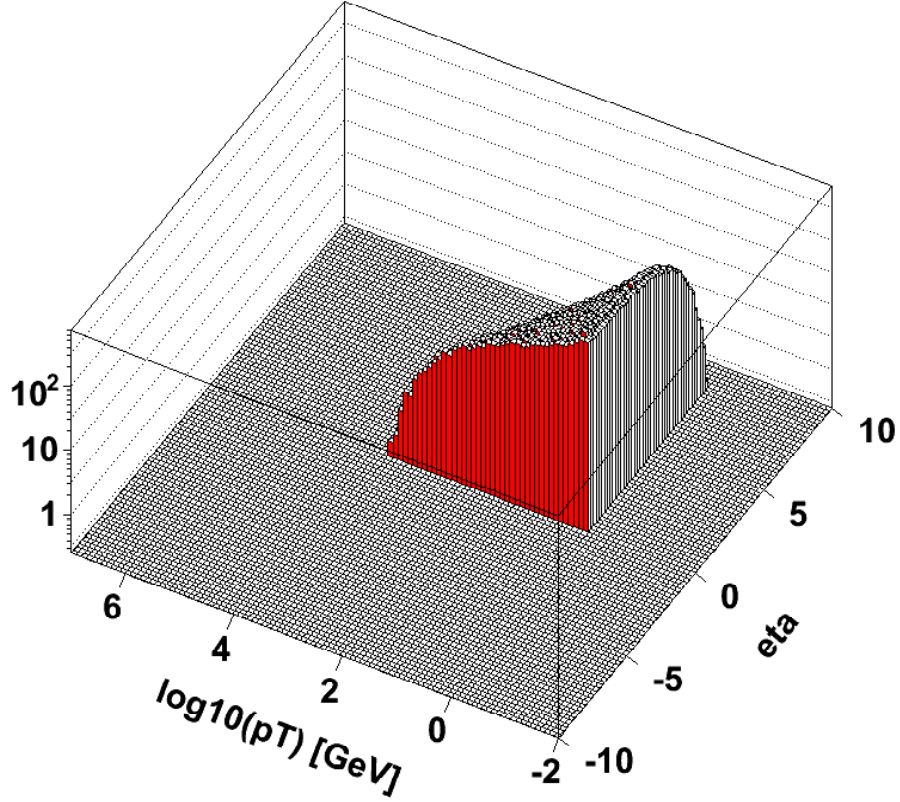
[19]See eqs. (69) and (89).

Figure 5: Distribution of rapidity and log of k^T from CMC for $2E_h = 1000$ GeV and $\lambda = 1$ GeV, for kernel type (C), pure bremsstrahlung.

- In the above, the number of flavour transitions ($G \to Q$ and $Q \to G$) is limited to $n = 0, 1, 2, 3, 4$, aiming at the precision of $\sim 0.2\%$.

- For each i-th pure gluon bremsstrahlung segment emissions variables are generated using the dedicated bremsstrahlung CMC of Section 4, according to eq. (93). Weights W_i^G are calculated.

- Generated MC events are weighted with $W = \prod W_i^G$. They are optionally turned into weight=1 events using the conventional rejection method.

- Four-momenta k_i^μ and q_i^μ are reconstructed out of evolution variables and azimuthal angles.

The above algorithm is already implemented in the form of a program in C++ and tested using upgraded version of the Markovian MC of refs. [28] and [17]. The numerical results are documented in the following section.
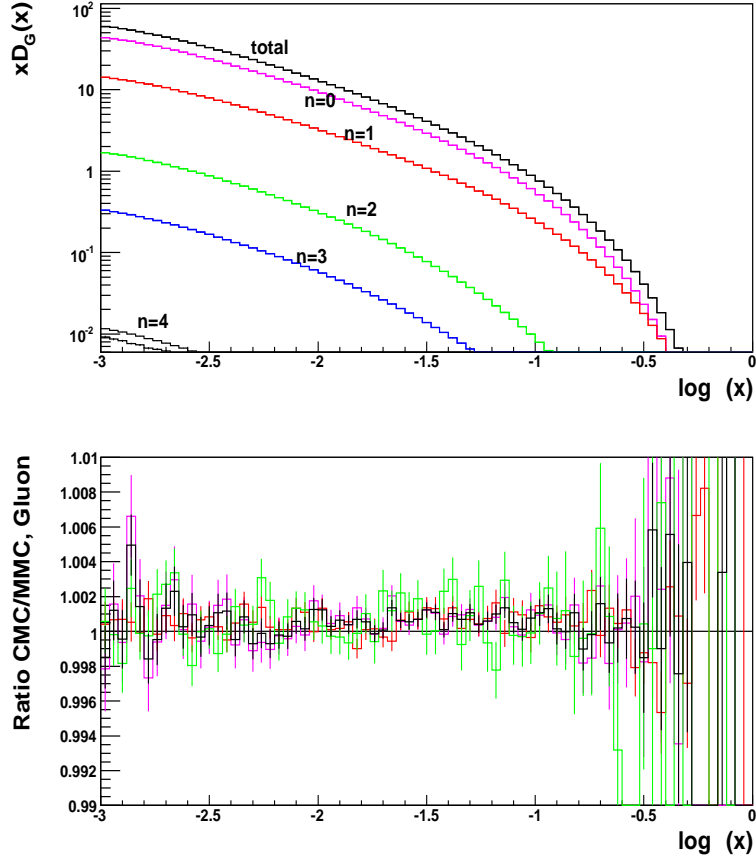
Figure 6: Gluon distribution from CMC and MMC for evolution with kernel (C'), $\lambda = 1$ GeV and $e^{t_{max}} = 2E_h = 1$ TeV. Contributions from fixed number of the quark-gluon transitions $n = 0, 1, 2, 3, 4$ are also shown. The ratios CMC/MMC in the lower plot are separately for the total result and for the number of quark-gluon transition $n = 0, 1, 2$. MC statistics is $4.5 \cdot 10^9$ weighted events for CMC and 10^{10} for MMC.

6   Numerical tests

Most of the numerical tests proving that the new CMCs of this paper work correctly were done by means of comparison with the updated version of the Markovian MC program MMC [15], which is a descendant of that described in refs. [17, 28]. The version of MMC used below implements exactly the same type evolution with exactly the same kernels and boundary conditions. We can therefore expect numerical agreement of CMC and MMC to within statistical MC error, which will be of order 10^{-3}, for about 10^{10} MC events (for about 10^2 different values of x in a single MC run).

We start, however, with the very simple tests in which we verify the correctness of the mapping evolution variables into four-momenta. In Fig. 5 the distribution of rapidity and k_i^T of the emitted gluons is shown. Sharp cut-offs corresponding to the minimum rapidity
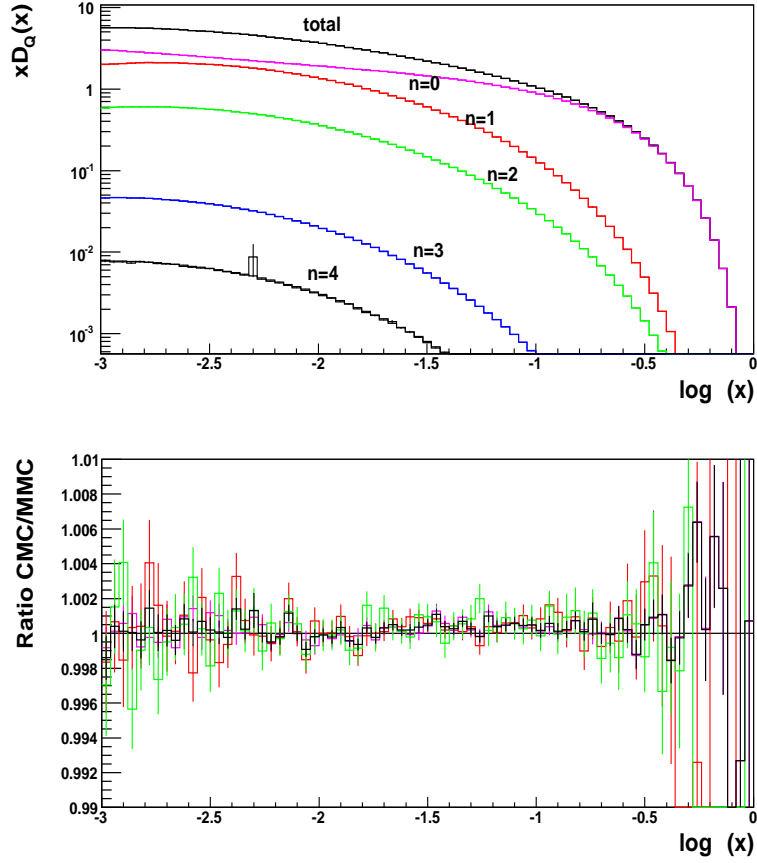
29

Figure 7: Quark distribution from CMC and MMC for evolution with kernel (C'), $\lambda = 1$ GeV and $e^{t_{\max}} = 2E_h = 1$ TeV. Contributions from fixed number of the quark-gluon transitions $n = 0, 1, 2, 3, 4$ are also shown. The ratios CMC/MMC in the lower plot are separately for the total result and for the number of quark-gluon transition $n = 0, 1, 2$. MC statistics is 10^{10} weighted events for both CMC and MMC.

$\eta \geqslant 0$ (maximum t) and minimum $k^T \geqslant 1$ GeV are clearly visible in the plot. NB. This plot shows the same triangular area in the logarithmic Sudakov plane in Fig. 4(a), but now populated with the MC events.

6.1 Testing CMC versus MMC

We examine results of the evolution from the initial energy scale $\lambda = 1$ GeV up to final energy scale of $E_h = 1$ TeV, using exactly the same initial quark and gluon distributions in a proton at the $Q_0 = e^{t_0} = \lambda = 1$ GeV scale as in ref. [8] (they were also used in ref. [28]).

In Fig. 6 the distributions of gluons at $E_h = 1$ TeV are shown, while in Fig. 7 the corresponding results for quarks, $Q = q + \bar{q}$, are exposed, also at the same high energy
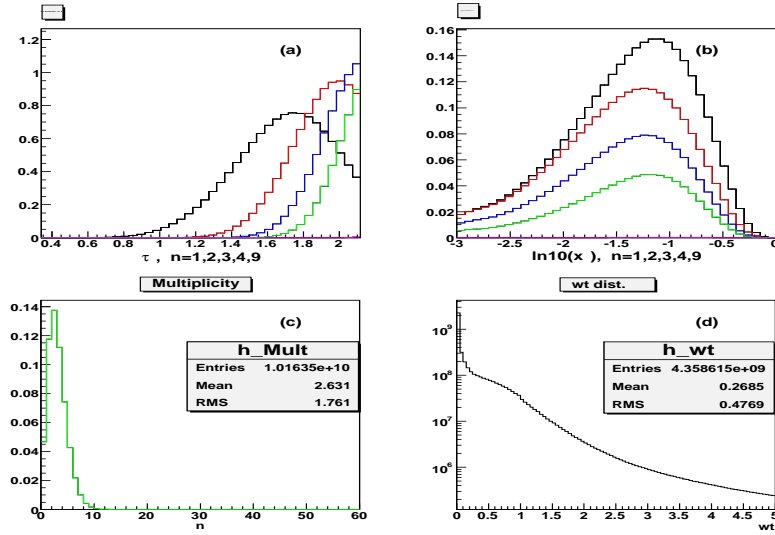
Figure 8: Distributions of τ_i and x_i and emission multiplicity from CMC and MMC programs. Results from MC run for outgoing gluon.
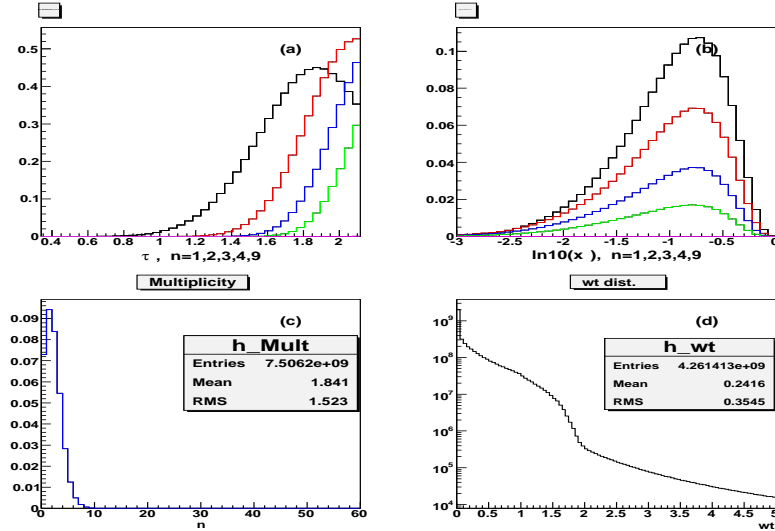

Figure 9: Distributions of τ_i and x_i and emission multiplicity from CMC and MMC programs. Results from MC run for outgoing quark.

scale $E_h = 1$ TeV. Numerical results are for the evolution type (C'), see Section 3.2, that is for $\alpha_S(k^T)$ and the evolution time being identical with the rapidity of the emitted particle. In these two figures we compare gluon and quark distributions obtained from the current CMC program and from the updated MMC program [15]. The principal numerical results from both programs, marked as "total" in the upper plot of both figures, are indistinguishable. We therefore plot their ratio in the lower plot of both figures. Parton distributions from CMC and MMC programs *agree perfectly*, within the statistical error

31

of $\sim 10^{-3}$ in the entire range of x. In the same plots we include individual contributions from a fixed number of the quark-gluon transitions $n = 0, 1, 2, 3, 4$, and their ratios. Again very good agreement between CMC and MMC results is seen[20].

It should be stressed that the MMC program has been tested numerically at the same $\sim 10^{-3}$ precision level for all types of kernels (A-C) by comparison with the semi-analytical (non-MC) program APCHEB [29]. In addition, for the DGLAP case, MMC was also successfully compared with another non-MC program QCDnum16 [30]. These dedicated test of MMC using non-MC programs will be published separately in the forthcoming work [15].

Additionally, in Figs. 8 and 9, we show comparisons of selected (semi-)exclusive distributions from CMC and MMC, i.e. we have chosen for the test the distributions of the consecutive evolution time variables $\tau_i(t_i)$, energy fractions x_i and total parton emission multiplicity from both CMC and MMC. The above distributions resulting from separate CMC and MMC runs are interposed – no visible difference is seen within the resolution of the plots. The above plots are for a final parton being a gluon in Fig. 8 and a quark in Fig. 9. In these plots one is testing a nontrivial aspect of the CMC algorithm related to removing and restoring t-ordering described in Section 4.2. In fact, any departure from the *pairwise* ordering procedure described in Section 4.2 would ruin the agreement of CMC and MMC seen in Figs. 8 and 9! The weight distribution from CMC is also shown in the lower part of these figures.

All the above tests were done for the most interesting case of CMC with kernel (C). What about numerical tests for cases (A) and (B)? In ref.[8] numerical agreement within $\sim 10^{-3}$ statistical error between CMC and MMC for the LL DGLAP case (A) was already documented. While preparing this work, we have compared CMC and MMC for case (B), $\alpha_S(e^{-t}\lambda)$, obtaining equally good numerical agreement. The corresponding plots[21] look quite similar to these in Fig. 6 and Fig. 7.

7   Discussions and summary

We have generalized the constrained Monte Carlo algorithm of ref. [8] from DGLAP to two other more complicated types of the evolution equations, one of them fully compatible[22] with the CCFM evolution equation [12]. The above extentions of the CMC algorithm are implemented in the computer program CMC, tested by comparing it with another Markovian MC, which solves exactly the same evolution equations.

Let us point out that the most complicated version (C) of the evolution equation elaborated in this work follows closely the CCFM model as formulated in [25] (except temporarily omitted non-Sudakov formfactors). Its Markovian MC implementation SMALLX

---

[20]Certain disagreements for $n = 4$, which affect total result at the 10^{-4} level can be traced back to well understood inefficiencies of FOAM at higher dimensions.

[21]They were shown in the contribution by S. Jadach to the Ustron Conference, September 2005, see `http://home.cern.ch/jadach/public/ustron05.pdf`.

[22]We did not include into the discussion the so called non-Sudakov form-factor for CCFM, case (C). However, it is already included in the code.

was worked out in refs. [23, 24], and later on exploited in the construction of the CAS-CADE MC of ref. [31], which is based on the backward evolution algorithm [7]. Our CMC program was tested against our own Markovian MC, see previous section. However, it would be interesting to compare it also with SMALLX and CASCADE programs. We hope this will be done in the future.

Since the CMC program of this work will be used as a building block in the MC event generator for Drell-Yan type processes and deep inelastic lepton-hadron scattering, the explicit mapping of the evolution variables to four-momenta of the emitted particles is also defined, implemented and tested. Evolution time is mapped into rapidity variable (angular ordering). A sharp cut-off is imposed on the k^T and rapidity of the emitted particle. The sharp cut-off in rapidity will be useful when combining two MMCs for two colliding initial state hadrons, with neither gaps nor overlaps in the emission phase space.

The CMC algorithm was worked out in detail and tested for three types of evolution kernels and phase space limits. It was purposely defined/described in such a way that it can be easily generalized to other types of the evolution kernels. In the following works it will be used as a component in the MC modelling of the initial state parton shower for showering of a single hadron in a more complete MC project for LHC.

Acknowledgments

Appendix

A   Auxiliary functions in formfactors

A.1   Triangle function ϱ

The simplest component function in the Sudakov formactor in cases (B) and (C) of z-dependent α_S reads as follows

$$\varrho(\bar{t}_b; \bar{t}_\lambda) = \int^{\bar{t}_b} d\bar{t} \int^0 dv \frac{\theta_{\bar{t}+v>t_\lambda}}{\bar{t}+v} = \theta_{\bar{t}_b>\bar{t}_\lambda}\{\bar{t}_b \left[\ln \bar{t}_b - \ln \bar{t}_\lambda - 1\right] + \bar{t}_\lambda\}, \qquad (96)$$

where λ defines IR cut-off. Here, and in the following, we follow certain notation rules allowing us to write the above and similar functions in a compact way:

1. For all variables like t, t_b and t_λ bar over them means $\bar{t} = t - \ln \Lambda_0$, $\bar{t}_b = t_b - \ln \Lambda_0$, $\bar{t}_\lambda = t_\lambda - \ln \Lambda_0$ etc., where Λ_0 is that in eq. (29). i.e. position of the Landau pole.

2. Occasionally we shall omit the explicit dependence on $t_\lambda = \ln \lambda$, that is we always understand $\varrho(\bar{t}_b) \equiv \varrho(\bar{t}_b; \bar{t}_\lambda)$.
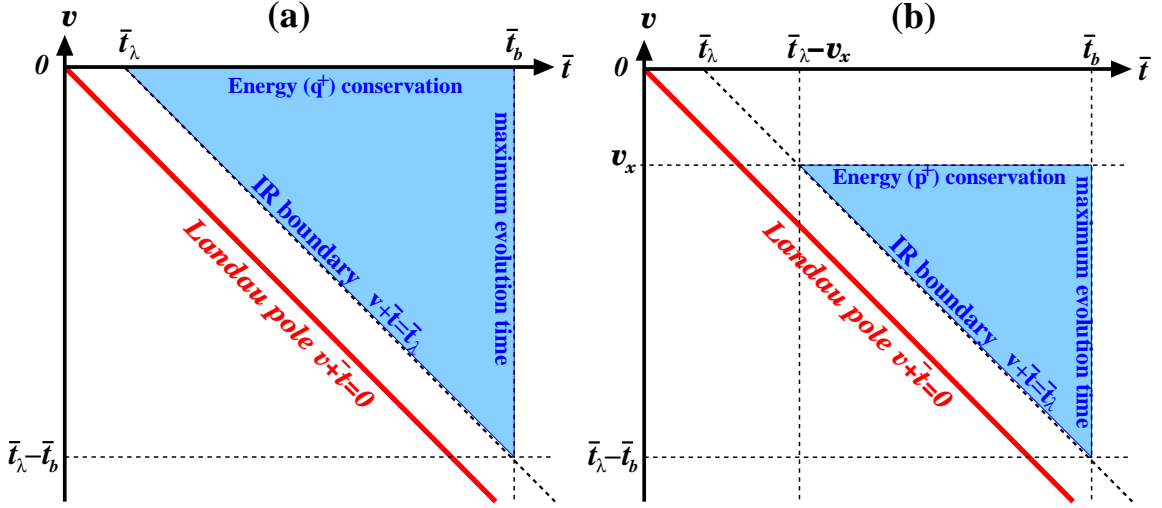
Figure 10: Integration area in the definition of the functions: (a) $\varrho(\bar{t}_b; \bar{t}_\lambda)$ and (b) $\varrho(\bar{t}_b + v_x; \bar{t}_\lambda)$.

3. We always understand that $\int_a^b dx \equiv 0$ when $b \leq a$.

The area of the integration in eq. (96) is the triangle depicted if Fig. 10 (a).

The following similar integral, with the same integrand, but slightly different triangular integration area depicted in Fig. 10 (b), can be expressed using the same function

$$\int^{\bar{t}_b} d\bar{t} \int^{v_x} dv \frac{\theta_{\bar{t}+v>\bar{t}_\lambda}}{\bar{t} + v} = \varrho(\bar{t}_b + v_x; \bar{t}_\lambda). \tag{97}$$

A.2 Trapezoid function ϱ_2

The other basic IR-singular component function in the Sudakov formfactor reads

$$\varrho_2(\bar{t}_a, \bar{t}_b; \bar{t}_\lambda) = \int_{\bar{t}_a}^{\bar{t}_b} d\bar{t} \int^0 dv \frac{\theta_{\bar{t}+v>\bar{t}_\lambda}}{\bar{t} + v} = \theta_{\bar{t}_b>\bar{t}_a} \{\varrho(\bar{t}_b; \bar{t}_\lambda) - \varrho(\bar{t}_a; \bar{t}_\lambda)\} \tag{98}$$

The corresponding integration area is the trapezoid depicted in Fig. 11 (a). The above calculation result is obvious if one notices that the trapezoid is the difference of two overlapping triangles. Similarly, the following integral with the trapezoid integration area of Fig. 11 (a) is again expressed in terms of the functions already defined above

$$\varrho_2^{[1]}(v_x, \bar{t}_a, \bar{t}_b; \bar{t}_\lambda) = \int_{\bar{t}_a}^{\bar{t}_b} d\bar{t} \int^{v_x} dv \frac{\theta_{\bar{t}+v>\bar{t}_\lambda}}{\bar{t} + v}$$
$$= \theta_{\bar{t}_b>\bar{t}_a} \{\varrho(\bar{t}_a + v_x; \bar{t}_\lambda) - \varrho(\bar{t}_b + v_x; \bar{t}_\lambda)\} = \varrho_2(\bar{t}_a + v_x, \bar{t}_b + v_x; \bar{t}_\lambda). \tag{99}$$

Let us stress that keeping $\theta_{\bar{t}_b>\bar{t}_\lambda}$ in the basic function ϱ of eq. (96) is essential for validity of the evaluation of all the following related functions and integrals.
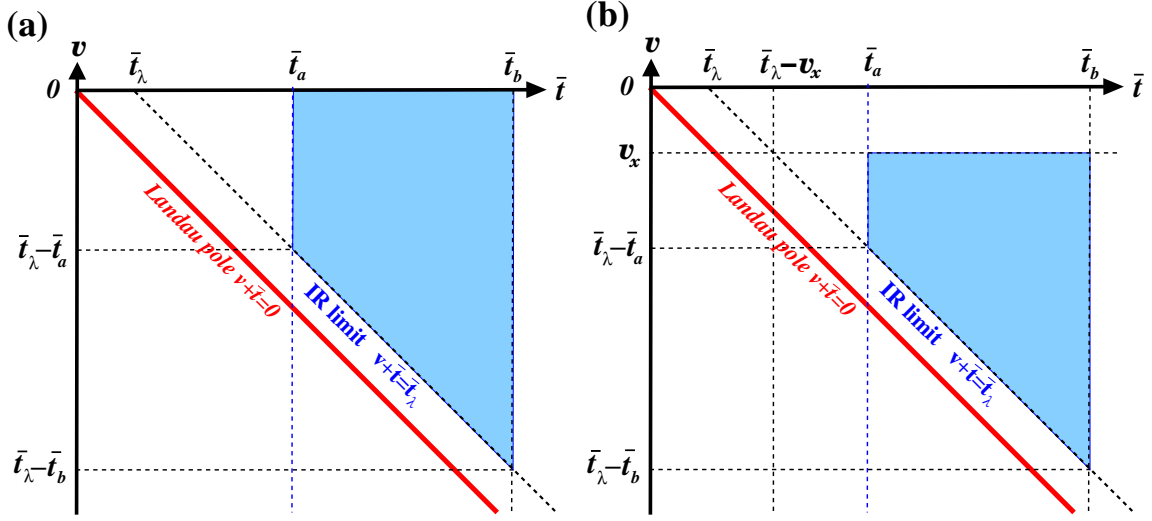
Figure 11: Integration area in the functions: (a) $\varrho_2(\bar{t}_a, \bar{t}_b; \bar{t}_\lambda)$ and (b) $\varrho_2(\bar{t}_a + v_x, \bar{t}_b + v_x; \bar{t}_\lambda)$.
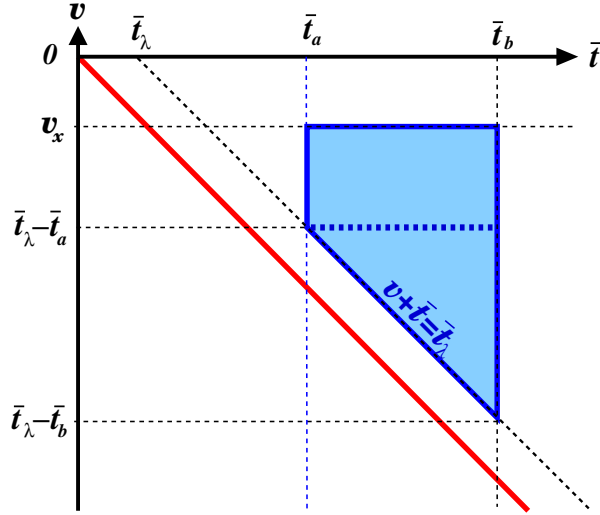

Figure 12: Integration area in the function $\varrho_2^{[F]}(\bar{t}_a, \bar{t}_b, v_x; \bar{t}_\lambda)$.

A.3 Non-singular functions and mapping

Let us evaluate now the following integral which is similar to ϱ_2, except that we insert into the integrand the additional function $F(v, t)$:

$$\varrho_2^{[F]}(v_x, \bar{t}_a, \bar{t}_b; \bar{t}_\lambda) = \int_{\bar{t}_a}^{\bar{t}_b} d\bar{t} \int^{v_x} dv \, \frac{\theta_{\bar{t}+v>\bar{t}_\lambda}}{\bar{t}+v} F(v, \bar{t}). \tag{100}$$

Of course, for $F = 1$ we recover ϱ_2. However, even in this case the following evaluation of $\varrho_2^{[F]}$ will be interesting, because we shall swap the integration order and introduce variable mapping, exactly the same which is used for finding out $K(v)$ and eliminating $K(v)$

through additional mapping $r(v)$. The above integral is also used for evaluation of the non-IR part of the bremsstrahlung integral and flavour changing contributions. In these cases we have $F(v,t) = F(v)$ and swapping integration order makes sense, because the inner integration over $\bar{t}$ can be done analytically (the outer one over v is done numerically). First, the integration order is swapped

$$\varrho_2^{[F]}(v_x, \bar{t}_a, \bar{t}_b; \bar{t}_\lambda) = \int\limits_{\bar{t}_\lambda - \bar{t}_b}^{v_x} dv \left\{ \theta_{v_x < \bar{t}_\lambda - \bar{t}_a} \int\limits_{\bar{t}_\lambda - v_x}^{\bar{t}_b} d\bar{t}\, \frac{1}{\bar{t} + v} + \theta_{v_x > \bar{t}_\lambda - \bar{t}_a} \int\limits_{\bar{t}_a}^{\bar{t}_b} d\bar{t}\, \frac{1}{\bar{t} + v} \right\} F(v, \bar{t}) \quad (101)$$

and the integration is split into two parts, first for the triangle and second for the rectangle integration area, see Fig. 12 for illustration. The above change of the integration order was also exploited by the authors of HERWIG MC [5, 32][23]. The internal integral can be transformed using identities

$$\varrho'(\bar{t}_b) = \partial_{\bar{t}_b} \varrho(\bar{t}_b; \bar{t}_\lambda) = \int\limits_{\bar{t}_\lambda}^{\bar{t}_b} d\bar{t}\, \frac{1}{\bar{t} + v} = \theta_{\bar{t}_b > \bar{t}_\lambda}(\ln \bar{t}_b - \ln \bar{t}_\lambda),$$
$$(102)$$
$$\varrho_2'^{[1]}(v_x, \bar{t}_a, \bar{t}_b) = \partial_{v_x} \varrho_2^{[1]}(v_x, \bar{t}_a, \bar{t}_b; \bar{t}_\lambda)$$
$$= \theta_{v_x > \bar{t}_\lambda - \bar{t}_a}[\varrho'(\bar{t}_b + v_x) - \varrho'(\bar{t}_a + v_x)] + \theta_{v_x > \bar{t}_\lambda - \bar{t}_a}\varrho'(\bar{t}_b + v_x)$$

to the following convenient form

$$\varrho_2^{[F]}(v_x, \bar{t}_a, \bar{t}_b; \bar{t}_\lambda) = \int\limits_{\bar{t}_\lambda - \bar{t}_b}^{v_x} dv\, \varrho_2'^{[1]}(v, \bar{t}_a, \bar{t}_b) \int\limits_0^1 d\left(\frac{\ln(\bar{t} + v)}{\varrho_2'^{[1]}(v, \bar{t}_a, \bar{t}_b)}\right) F(v, \bar{t})$$
$$(103)$$
$$= \varrho_2'^{[1]}(v_x, \bar{t}_a, \bar{t}_b) \int\limits_0^1 d\xi(v) \int\limits_0^1 d\sigma(v, \bar{t}), F(v, \bar{t})$$

where

$$\xi(v) = \frac{\varrho_2^{[1]}(v, \bar{t}_a, \bar{t}_b)}{\varrho_2^{[1]}(v_x, \bar{t}_a, \bar{t}_b)}, \quad \sigma(v, \bar{t}) = \frac{\ln(\bar{t} + v)}{\varrho_2'^{[1]}(v, \bar{t}_a, \bar{t}_b)} \quad (104)$$

The above mapping is used in the bremsstrahlung CMC in cases (B) and (C). In this context one also needs to perform the inverse mapping $v(\xi)$, which involves numerical inversion of $\varrho_2^{[1]}(v, \bar{t}_a, \bar{t}_b)$ as a function of v. Once v is known, the second inverse mapping $t(\sigma, \xi)$ is easily implemented, as it can be formulated in a fully analytical form. As already indicated, a very similar variant of the above integration order is used in the evaluation of the non-IR and flavour-changing parts of the Sudakov formfactor.

---

[23]We thank to Mike Seymour for private communication on that, see also Chapter 5 in *http://hepwww.rl.ac.uk/theory/seymour/thesis/*.

A.4 Integration area in the plane of $\ln k^T$ and rapidity η

Finally, let us relate the phase space v and t used in the calculation of the formfactor above with the Sudakov plane in $\ln k^T$ and rapidity η. This is done in the pictorial way in Fig. 4, where we depict a situation with three emission, underlining the trapezoid integration domain for the Sudakov function $\Phi(t_3, t_2|x_2)$. We show side by side the same the trapezoid marked by *abcd* in the plane of v and t on the right hand side and in the plane of $\ln k^T$ and rapidity η on the left hand side of the figure. One should remember that in this case (C) we define $v_i = \ln y_i = \ln(x_i - x_{i-1})$. Rapidities η_i and evolution times t_i are related by the simple linear transformation, see Sec. 3.1, eq. (21).

References

[1] J. C. Collins, D. E. Soper, and G. Sterman, *Nucl. Phys.* **B250** (1985) 199.

[2] G. T. Bodwin, *Phys. Rev.* **D31** (1985) 2616.

[3] R. Ellis, W. Stirling, and B. Webber, *QCD and Collider Physics.* Cambridge University Press, 1996.

[4] T. Sjostrand *et al.*, *Comput. Phys. Commun.* **135** (2001) 238–259, `hep-ph/0010017`.

[5] G. Corcella *et al.*, *JHEP* **01** (2001) 010, `hep-ph/0011363`.

[6] S. Jadach *et al.*, in preparation.

[7] T. Sjostrand, *Phys. Lett.* **B157** (1985) 321.

[8] S. Jadach and M. Skrzypek, *Comput. Phys. Commun.* **175** (2006) 511–527, `hep-ph/0504263`.

[9] S. Jadach and M. Skrzypek, *Acta Phys. Polon.* **B36** (2005) 2979–3022, `hep-ph/0504205`.

[10] S. Jadach and M. Skrzypek, Report CERN-PH-TH/2005-146, IFJPAN-V-05-09, *Contribution to the HERA–LHC Workshop, CERN–DESY, 2004–2005, http://www.desy.de/∼heralhc/*, `hep-ph/0509178`.

[11] L.N. Lipatov, *Sov. J. Nucl. Phys.* **20** (1975) 95;
V.N. Gribov and L.N. Lipatov, *Sov. J. Nucl. Phys.* **15** (1972) 438;
G. Altarelli and G. Parisi, *Nucl. Phys.* **126** (1977) 298;
Yu. L. Dokshitzer, *Sov. Phys. JETP* **46** (1977) 64.

[12] M. Ciafaloni, *Nucl. Phys.* **B296** (1988) 49;
S. Catani, F. Fiorani and G. Marchesini, *Phys. Lett.* **B234** 339, *Nucl. Phys.* **B336** (1990) 18;
G. Marchesini, *Nucl. Phys.* **B445** (1995) 49.

[13] S. Frixione and B. R. Webber, *JHEP* **06** (2002) 029, `hep-ph/0204244`.

[14] S. Frixione and B. R. Webber, `hep-ph/0601192`.

[15] S. Jadach *et al.*, in preparation.

[16] S. Jadach, M. Skrzypek, and Z. Was, `hep-ph/0701174`.

[17] K. Golec-Biernat, S. Jadach, W. Placzek, and M. Skrzypek, *Acta Phys. Polon.* **B37** (2006) 1785–1832, `hep-ph/0603031`.

[18] J. C. Collins and D. E. Soper, *Nucl. Phys.* **B193** (1981) 381.

[19] G. Curci, W. Furmanski, and R. Petronzio, *Nucl. Phys.* **B175** (1980) 27.

[20] A. H. Mueller, *Phys. Lett.* **B104** (1981) 161–164.

[21] S. Catani, B. R. Webber, and G. Marchesini, *Nucl. Phys.* **B349** (1991) 635–654.

[22] Y. Dokshitzer, V. Khoze, A. Mueller, and S. Troyan, *Basics of Perturbative QCD*. Editions Frontieres, 1991.

[23] G. Marchesini and B. R. Webber, *Nucl. Phys.* **B349** (1991) 617–634.

[24] G. Marchesini and B. R. Webber, *Nucl. Phys.* **B349** (1991) 617–634.

[25] G. Marchesini, *Nucl. Phys.* **B445** (1995) 49–80, `hep-ph/9412327`.

[26] D. Amati, A. Bassetto, M. Ciafaloni, G. Marchesini, and G. Veneziano, *Nucl. Phys.* **B173** (1980) 429.

[27] S. Jadach, *Comput. Phys. Commun.* **152** (2003) 55–100, `physics/0203033`.

[28] S. Jadach and M. Skrzypek, *Acta Phys. Polon.* **B35** (2004) 745–756, `hep-ph/0312355`.

[29] K. Golec-Biernat, the Fortran code to be obtained from the author, unpublished.

[30] M. Botje, ZEUS Note 97-066, http://www.nikhef.nl∼h24/qcdcode/.

[31] H. Jung and G. P. Salam, *Eur. Phys. J.* **C19** (2001) 351–360, `hep-ph/0012143`.

[32] G. Marchesini and B. R. Webber, *Nucl. Phys.* **B310** (1988) 461.