



LUND
UNIVERSITY

Final States in DIPSY

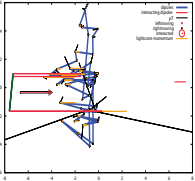
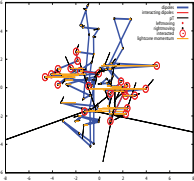
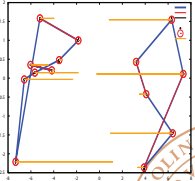
Christoffer Flensburg

Department of Theoretical Physics
Lund University

MCnet, CERN 13-14/01 2010

Work done with Gösta Gustafson and Leif Lönnblad.

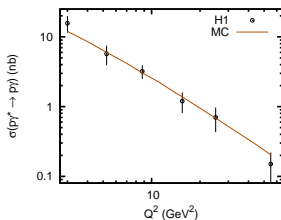
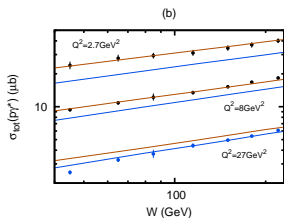
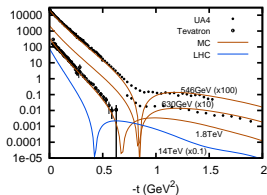
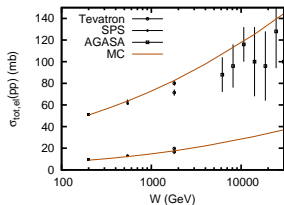
Low x colour dipoles evolved in transverse space and rapidity.

Cascade	Interaction	Final State
$\frac{dP}{dy} = \frac{\bar{\alpha}}{2\pi} d^2 r_2 \frac{r_{01}^2}{r_{02}^2 r_{12}^2}$ <p>+NLO corrections Evolve to y_0.</p>	$f_{ij} = \frac{\alpha_s^2}{2} \ln^2 \left(\frac{r_{13} r_{24}}{r_{14} r_{23}} \right)$ $P = 1 - e^{-\sum_{ij} f_{ij}}$ $\sigma = \int db \langle P(b) \rangle$	<p>Decide interactions. Remove virtuals. FSR, hadronisation.</p>
		



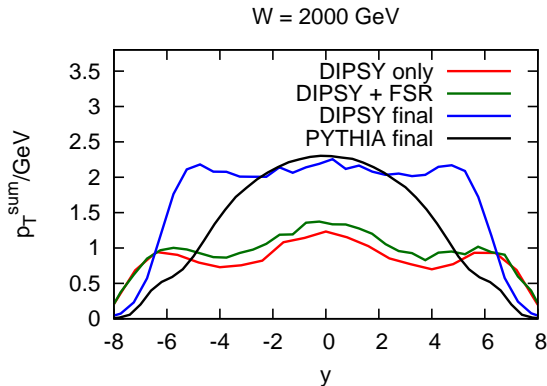
Some sample results

pp and γ^*p : total, elastic and diffractive cross section.



Final states

Preliminary results! Average p_{\perp} activity agrees with pythia, but other observables are off.



Problems with final states

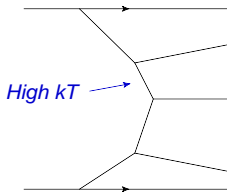
Needed to decide which particles are virtual, and which come on shell.

A parton can come on shell depending on if the other partons come on shell or not.

Where to start? Chicken or the egg?



Example: non-DGLAP chains in BFKL



Local maxima in p_{\perp} are allowed in the BFKL cascade, but suppressed with a factor $\frac{k_{\perp, \text{low}}^2}{k_{\perp, \text{high}}^2}$.

This can be interpreted as the emissions being reabsorbed after collision.

With the initial weight $\frac{d^2 k_{\perp}}{k_{\perp}^2}$ this gives that the highest k_{\perp} in the event always goes as $\frac{d^2 k_{\perp}}{k_{\perp}^4}$.



non-DGLAP chains in Muellers dipoles



$\frac{dP}{dy} = \frac{\bar{\alpha}}{2\pi} d^2 r_2 \frac{r_{01}^2}{r_{02}^2 r_{12}^2}$ gives weights of $\frac{d^2 r}{r^2}$ to the dipole sizes. This is the Fourier transform of the BFKL $\frac{d^2 k_{\perp}}{k_{\perp}^2}$ weight.

The extra suppression is associated with an absorption after collision, and Muellers dipole model only calculates total cross sections. This effect is not included.



non-DGLAP chains in DIPSY

Based on Mueller dipoles, DIPSY generates dipoles with a weight $\frac{d^2 r}{r^2}$ to first order.

$$k_{\perp} \sim \frac{1}{r} \Rightarrow \text{weights } \frac{d^2 k_{\perp}}{k_{\perp}^2}.$$

Small dipoles need suppression $\frac{r_{\text{small}}^2}{r_{\text{large}}^2}$. This happens if the small dipole emits another real gluon, or if it interacts with the other state, otherwise not.

To decide if the dipole should be absorbed or not, we need to know the rest of the final state.





Getting the weights right

Keep the small dipoles during evolution.

If absorbed after collision, then recheck the cascade for inconsistencies, such as unordered emissions or new small dipoles.



Repeated until fully consistent.

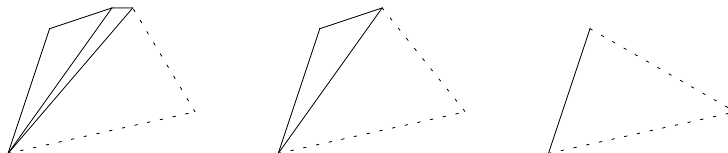
In this way the weights always end up as they should, while maintaining the ordering of the partons.



So is the problem solved?

Always gives ordered consistent states, but may miss phase space.

The allowed phase space for further emissions is reduced significantly by the emission of a small dipole.



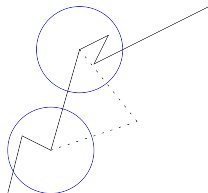
The small dipoles in the evolution block phase space that will be allowed if a small dipole is absorbed.



Fixing the fix: effective partons

Again we try to cover all cases: better to emit too much in the cascade and then remove them.

Here we will always cover all possible phase space by doing further emission as if the small dipoles never existed.



The emitted gluon will use the energy provided by all partons inside the circle.

In this way we manage to always get the right suppression for high k_{\perp} while maintaining consistently p_{\pm} ordered states and fully covered phase space.



Conclusions

Further progress with the model, closer to experimental data.
Still many observables that are off though.

Final states are complicated, but doable.

Currently focusing on correlation lengths, multiple interactions
and diffractive events.

Hope to have good news by next meeting! :)

Thanks for listening!

