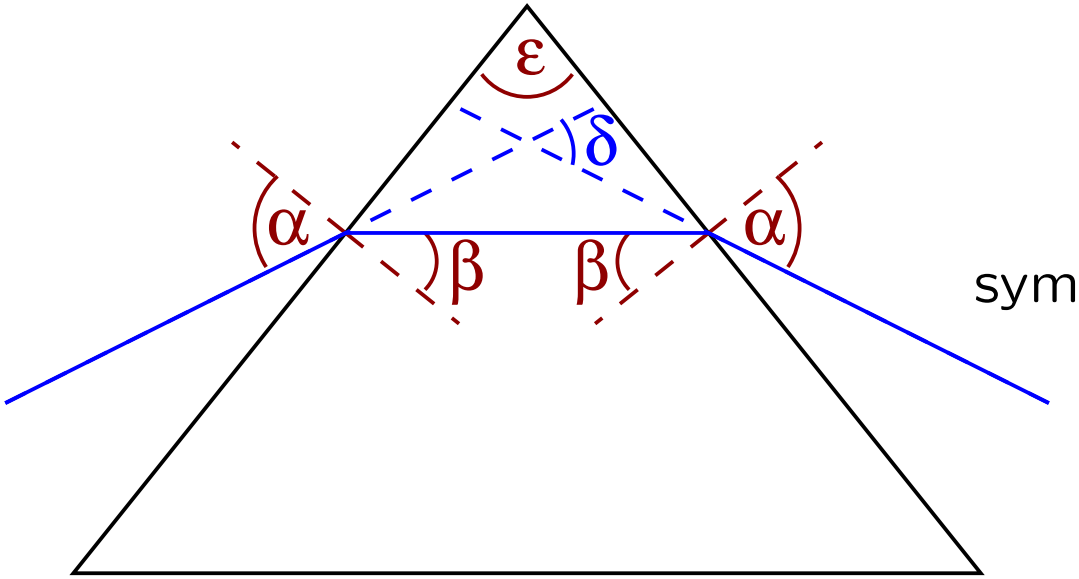
Rainbow Sort

– Sorting at the Speed of Light –

Dominik Schultes

31. May 2004

Basics: Prism



symmetrical optical path

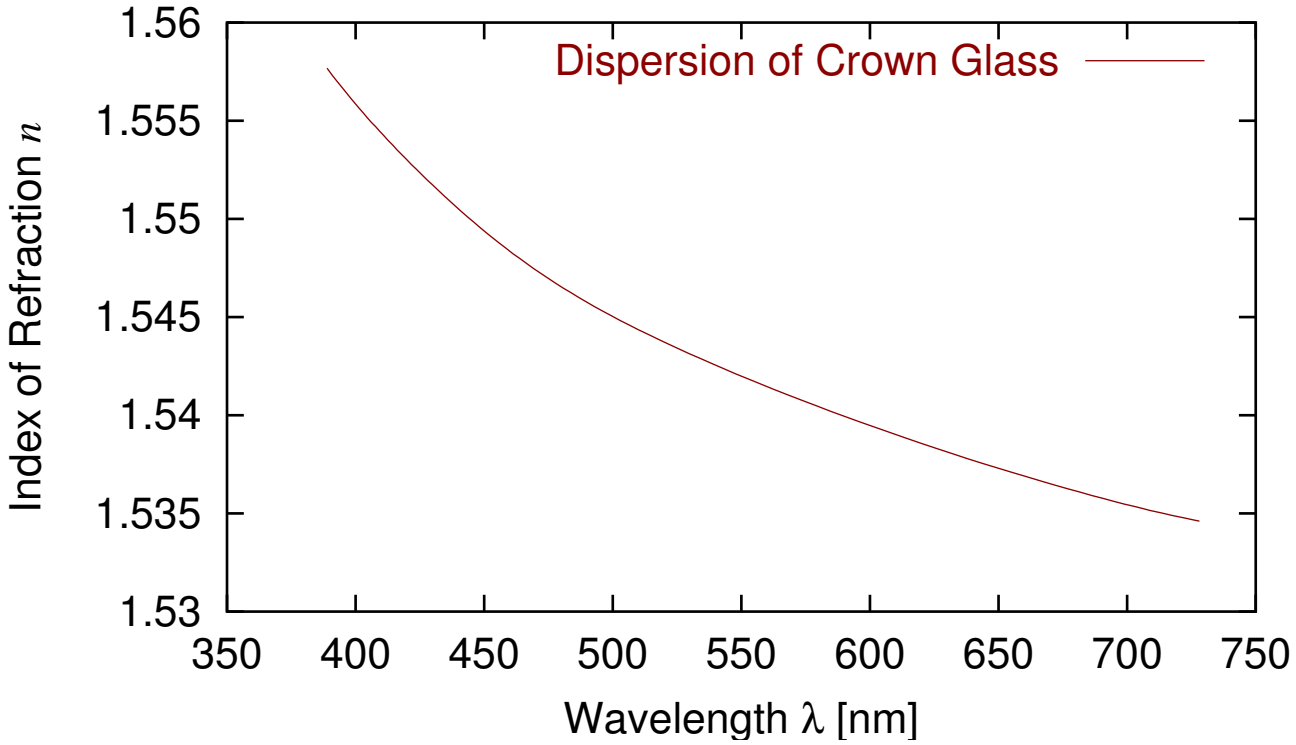
$$\text{angular deviation : } \delta = 2\arcsin\left(n\sin\frac{\varepsilon}{2}\right) - \varepsilon$$

arcsin is strictly monotonic increasing

$\rightsquigarrow$ the greater n, the greater the deviation

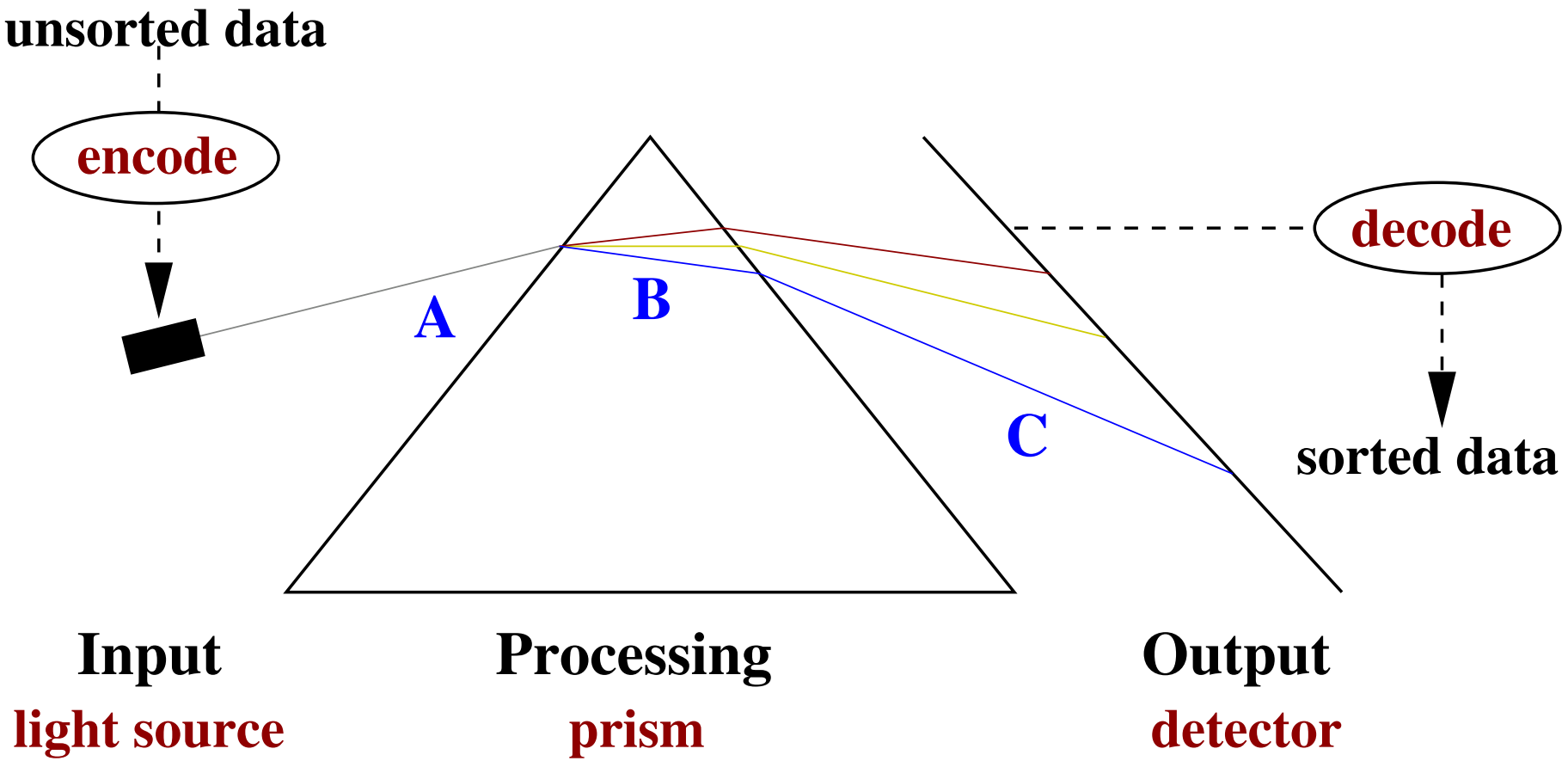
Basics: Dispersion

Refraction index depends on the wavelength of the ray



the less the wavelength λ, the greater n, the greater the deviation

Setup


unsorted data
encode
A
B
C
decode
sorted data
Input
light source
Processing
prism
Output
detector

Algorithm

Input

`Ray` $ray := \emptyset$

for each $x \in Input$ **do** $ray := ray \cup f(x)$

Processing

send ray through prism

Output

`Stack` $sorted := \emptyset$

`Wavelength` $cur\lambda := \infty$

whenever $\min \lambda(incoming\ rays) < cur\lambda$ **do**

$cur\lambda := \min \lambda(incoming\ rays)$

$sorted$.push($f^{-1}(cur\lambda)$)

if $sorted$.size $= n$ **then return** $sorted$

Conclusion

Input	$\Omega(n)$	$O(n+m)$
Processing	$\Theta(1)$	
Output	$\Omega(n)$	$O(n+m)$
Space	$\Omega(n)$	$O(n+m)$

- if measurement required: $\Theta(n+m)$
- otherwise: somewhere between $\Omega(n)$ and $O(n+m)$

in general: lower bound for sorting (time / space) ?