pandora:

an object-oriented event generator for Linear Collider physics

Michael E. Peskin
SLAC, Stanford U.

the ideal LC physics generator should:

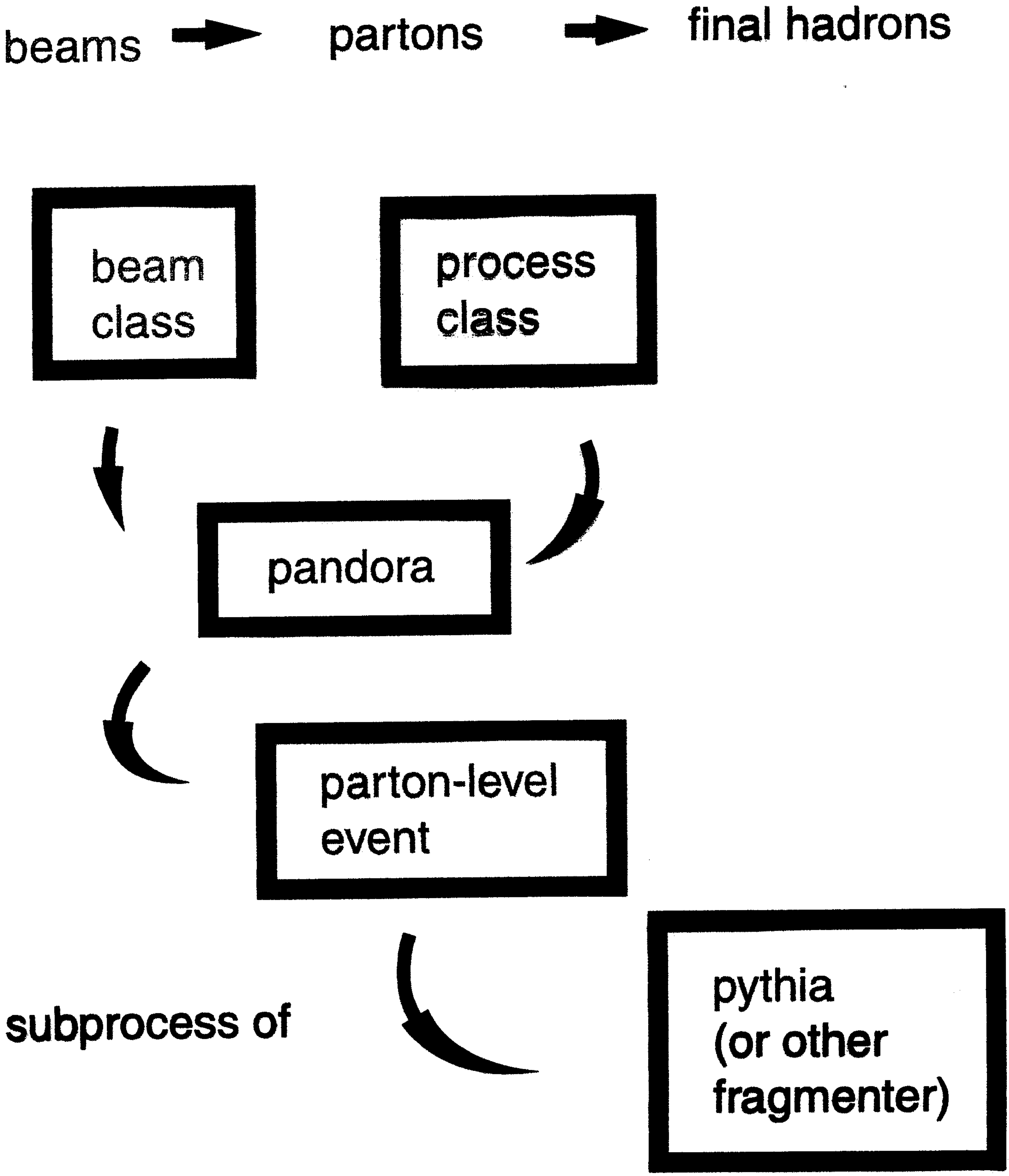
include beamstrahlung
and initial state radiation
include polarized beams
propagate polarization effects to
final-state particle distributions
allow input of any parton-level process

hadronization and detector modelling are slow,
so event selection need not be rapid

so, use methods that are general, if brute-force

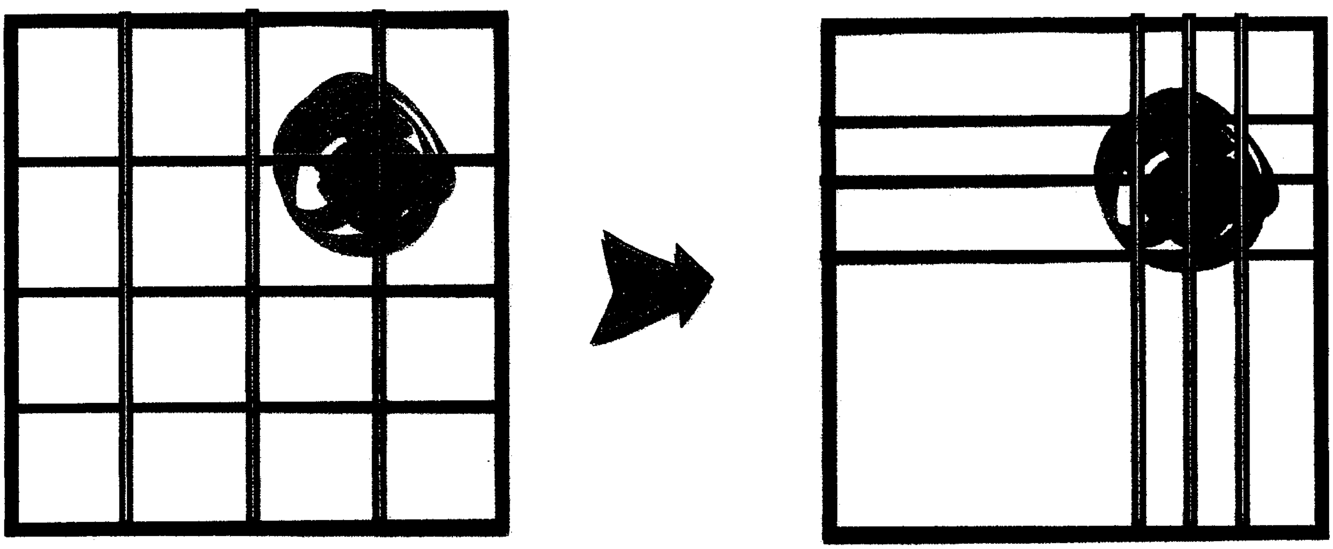
package the complexity of the problem
into modules using classes

working of an event generator:



method of event selection:

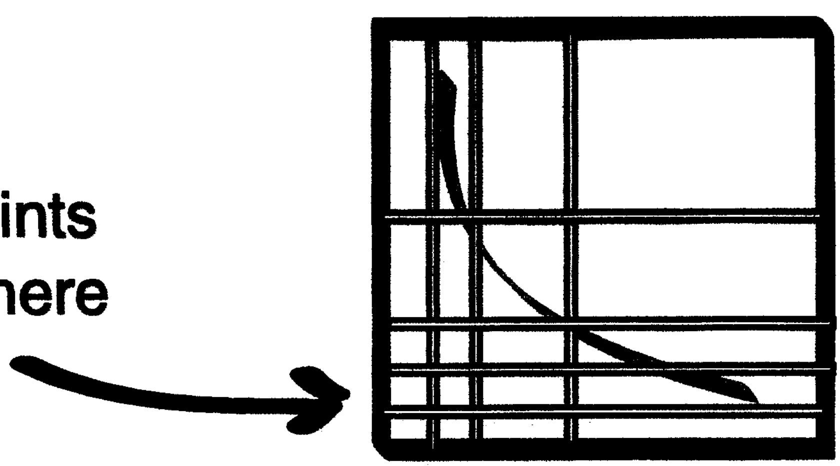
VEGAS (as in BASES, SPRING, by Kawabata)



adaptation takes about 10^5 events

a well-known problem with this approach (see Ohl's VAMP)

too many points are chosen here



two ways to use pandora:

'How do I ...?'

'Give me more!'

partons only,
with full
polarization
effects

interface to pythia
(T. Barklow
and M. Iwasaki)

underlying classes

DVector — vector of doubles

LVector — Lorentz vector w. boost and rotation operations

LVlist — list of LVectors

LLVlist — list of lightlike LVectors (knows about spinor products)

LEvent — list of

LVector	ID	Partner	Parent	Final?

Beam class

integration variables: x_1, x_2

↳ momentum fraction: x_{out}

sample constructor:

```
ebeam(250.0, electron, photon, 0.9, 1, 1, NLC500)
```

ECM
from
to
polarization
ISR on
beamstr. on
design

treatment of beamstrahlung:

'consistent Yokoya-Chen'

M. E. Peskin, LCC Note 0010 (1999)

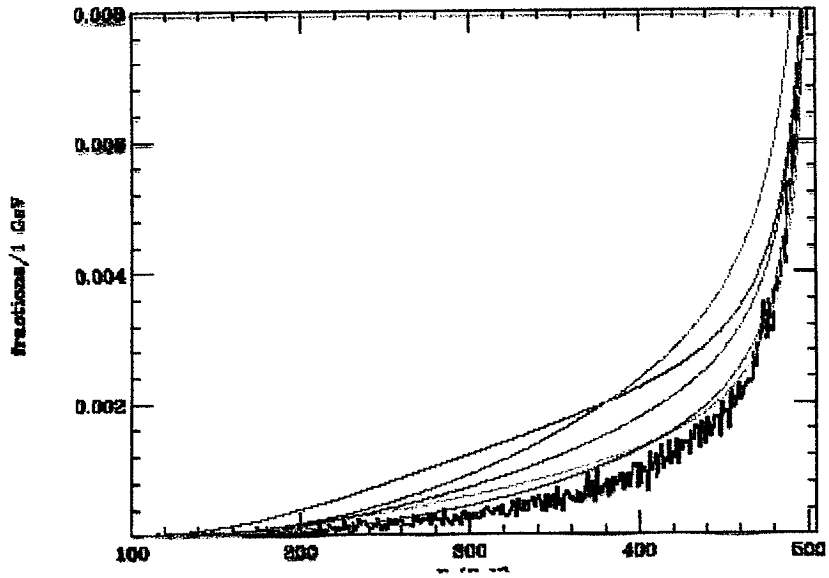
$\kappa = 2/3\Upsilon \qquad N = 1/2\ N\gamma$

$$f_e(x) = e^{-N}\left[\ \delta(x-1) + \frac{e^{-\kappa(1-x)/x}}{x(1-x)}\, h(y)\ \right]$$

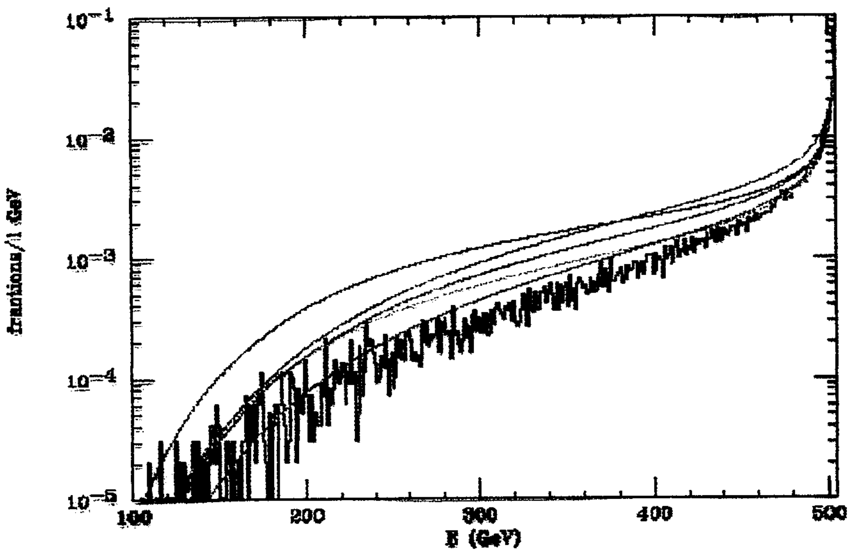
$$y = N\ (\kappa(1-x)/x)^{1/3}$$

$f_\gamma(x)$ is similarly simple

consistent-YC compared to Guinea Pig data

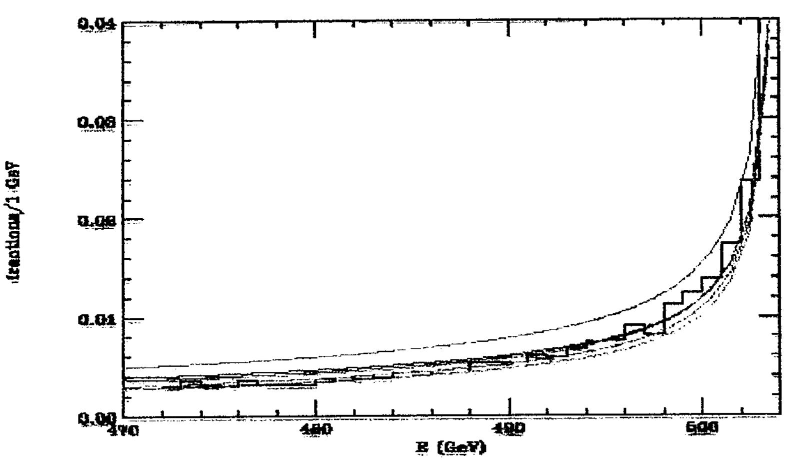


NLC 1000B



c-YC

— Chen92



thanks to
K. Thompson

Process class

integration over n variables $X_i \, \varepsilon \, [0,1]$

a process class implements an interface:
the following functions must be defined

int validEvent(X,s,beta)

returns 1 if we are able to procede

DVector crosssection(X,s)

returns $C[m] = \frac{d^n\sigma}{d^nX}$ for (++,+-,-+,--) incoming helicities

LEvent buildEvent(X,s)

returns a parton-level LEvent in the CM frame

implementation of e^+e^- -> W^+W^-

crosssection = prefactor()
* norm(helicityamplitude())
/W.divideby()*W.divideby()

helicityamplitude() =

sum of terms:

$amp(h_W)\ e^{ih_W \psi_W}\ decayamp(h_W)$

W, t decay amplitudes are methods of Wplusdecay, Wminusdecay, tdecay, etc. classes

conventions for helicity amplitudes:

assume final partons are produced
in the 3,1 plane
(works for 2- and 3-body final states)

for partons || 3 axis:

fermions (use 2-component notation)

$u+ = \sqrt{E-p}\ (1,0)$
$u- = \sqrt{E+p}\ (0,1)$

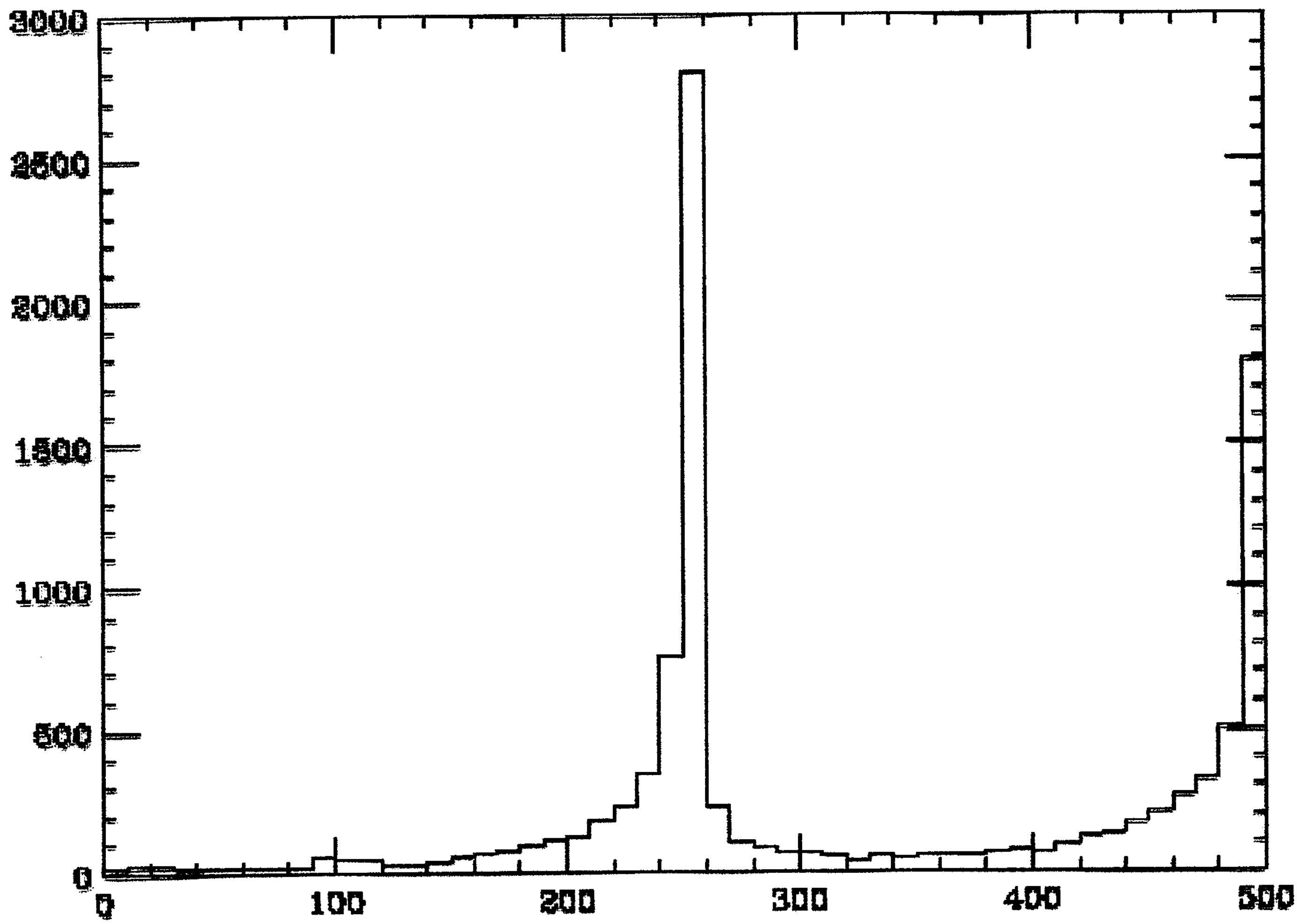
$v+ = \sqrt{E+p}\ (0,1)$
$v- = \sqrt{E-p}\ (-1,0)$

vectors

$\varepsilon+ = 1/\sqrt{2}\ (0, 1, i, 0)$
$\varepsilon- = 1/\sqrt{2}\ (0, 1, -i, 0)$
$\varepsilon0 = (k/m, 0, 0, E/m)$

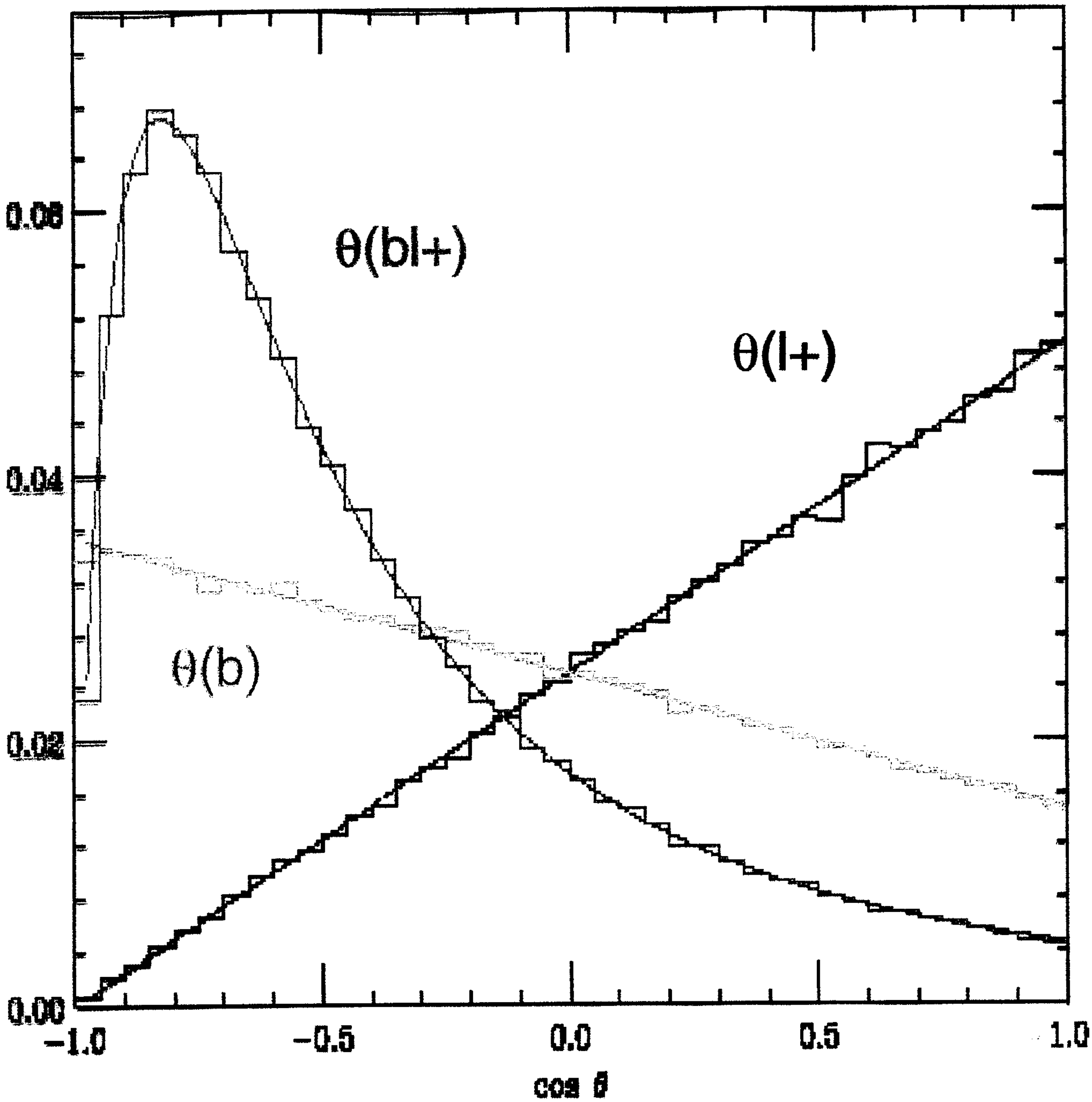
for other angles in the plane, rotate
counterclockwise about the 2 axis.

distribution of E(l+l-) in e+e- - l+l-



$\sqrt{s}$ = 500 GeV

t_R decay distributions



Current distribution of pandora:

http://www.slac.stanford.edu/
~mpeskin/LC/pandora.html

includes e+,e- beams only,

e+e- -> fermion pairs, boson pairs

$\gamma\gamma$ -> fermion pairs

W, Z, t decay functions

γe -> Wν, e+e- -> Z h, and other processes should soon be included

pandora_phythia: (T. Barklow)

shell script which uses pandora's hooks as input to pythia; events are output in stdHEP format

http://www.slac.stanford.edu/
~timb/generators.html

M. Iwasaki's hooks to TAOLA to provide polarized tau decays will soon be available at this site.