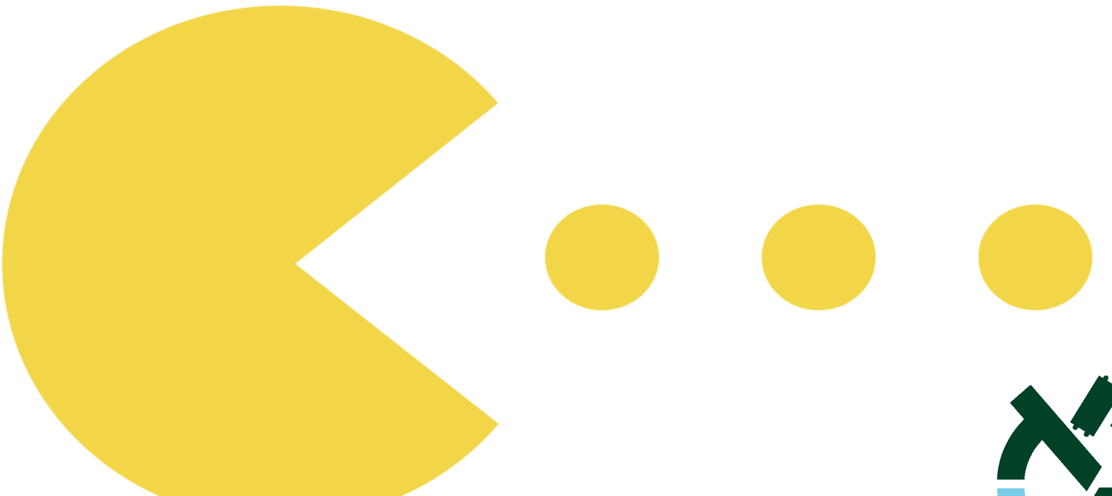
SOPRANO : numerics

Damien Bégué
Bar Ilan University

28/02/2023

With Sargis Gasparyan and Narek Sahakyan



The model : a blob

- No geometry : a blob of (constant) radius R moving with (constant) Lorentz factor Γ.
- A (constant) magnetic field threads the blob.
- Injection of particles.
- Track the particles with their secondaries for 1 dynamical time scale.
- It is also possible to include escape and track the emitting zone for longer.

The processes

- **Leptonic** (electrons/positrons):
  - Synchrotron (no self-absorption for now)
  - Inverse Compton
  - Pair production (but no annihilation for now)

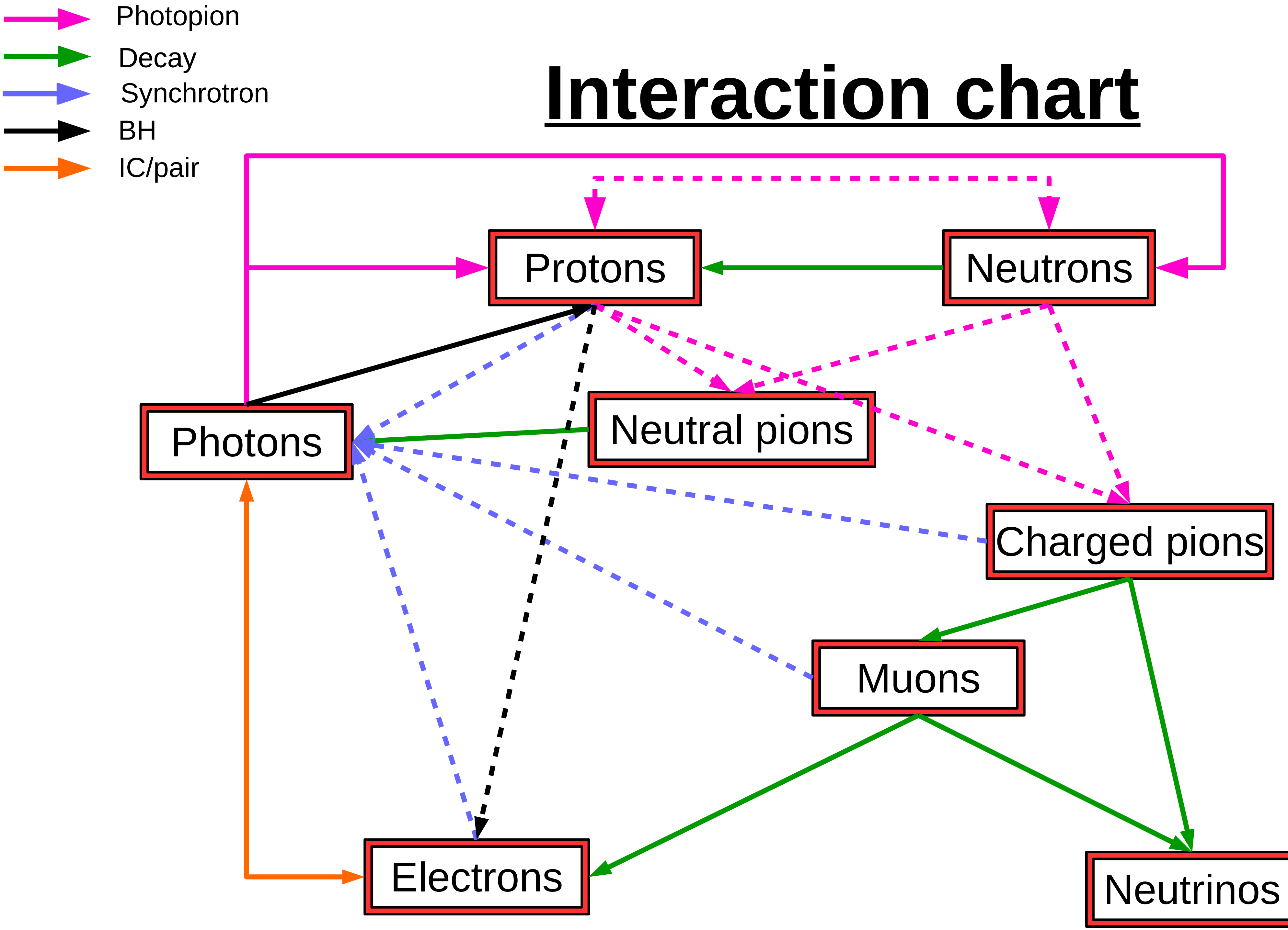
The processes

- **Leptonic** (electrons/positrons):
  - Synchrotron (no self-absorption for now)
  - Inverse Compton
  - Pair production (but no annihilation for now)
- **Hadronic** (protons/neutrons/pions/muons/neutrino):
  - Synchrotron
  - Photomeson production
  - Photonpair production (BH)
  - Decay

The processes

- **Leptonic** (electrons/positrons):
  - Synchrotron (no self-absorption for now)
  - Inverse Compton
  - Pair production (but no annihilation for now)
- **Hadronic** (protons/neutrons/pions/muons/neutrino):
  - Synchrotron
  - Photomeson production
  - Photonpair production (BH)
  - Decay
- **Photons:**
  - Produced by synchrotron of all charged particles
  - Scattered by IC (no cooling of photons)
  - Annihilation
  - Produced by neutral pion decay

Interaction chart



Kinetic equations

$$\frac{\partial N_p}{\partial t} = C_{p\gamma\to p\pi} + C_{p\gamma\to e^+e^-} + C_{\rm synch} - S_{\gamma p\to n\pi} + Q_{\gamma n\to p\pi}$$

$$\frac{\partial N_n}{\partial t} = -S_{n\gamma\to p\pi} + Q_{p\gamma\to n\pi} + C_{n\gamma\to n\pi}$$

$$\frac{\partial N_{\pi_\pm}}{\partial t} = Q_{p\gamma\to\pi} + Q_{n\gamma\to\pi} - S_\pi + C_{\rm synch}$$

$$\frac{\partial N_\mu}{\partial t} = Q_{\pi_\pm} - S_\mu + C_{\rm synch}$$

$$\frac{\partial N_{\nu,\zeta}}{\partial t} = Q_{\pi_\pm} + Q_\mu$$

C : diffusion-type equation
Q : integro-differential on a different species
S : sink term “exponential-like”
R_IC : integro-differential

$$\frac{\partial n_{\rm ph}}{\partial t} = -S_{\gamma\gamma\to e^+e^-} + Q_{\pi_0} + R_{\rm IC} + \sum_i Q^i_{\rm synch}$$

$$\frac{\partial N_{\rm e\pm}}{\partial t} = Q_\mu + Q_{p\gamma\to e^+e^-} + Q_{\gamma\gamma\to e^+e^-} C_{\rm IC} + C_{\rm synch}$$

Numerical discretization

- DG 1st order, finite volume : natural conservation of particle number (when required).
- Second order would be possible but large changes to the code would be required.

Numerical discretization

- DG 1st order, finite volume : natural conservation of particle number (when required).
- Second order would be possible but large changes to the code would be required.
- Widely different time scales require implicit method.

Numerical discretization

- DG 1st order, finite volume : natural conservation of particle number (when required).
- Second order would be possible but large changes to the code would be required.
- Widely different time scales require implicit method.
- Positivity of distribution function requires CC-like scheme (+ 1st order backward Euler).

Numerical discretization

- DG 1st order, finite volume : natural conservation of particle number (when required).
- Second order would be possible but large changes to the code would be required.
- Widely different time scales require implicit method.
- Positivity of distribution function requires CC-like scheme (+ 1st order backward Euler).
- Non-linear system :
  - treated as semi-implicit for photopair and photopion.
  - fully implicit for leptonic processes.

Numerical discretization

- DG 1st order, finite volume : natural conservation of particle number (when required).
- Second order would be possible but large changes to the code would be required.
- Widely different time scales require implicit method.
- Positivity of distribution function requires CC-like scheme (+ 1st order backward Euler).
- Non-linear system :
  - treated as semi-implicit for photopair and photopion.
  - fully implicit for leptonic processes.
- Energy is conserved by balancing cooling/heating and particle creation.

Numerical discretization

Energy grid

Numerical discretization

Energy grid

$$F^{\mathcal{I}} = \sum_i a_i L_i^{\mathcal{I}}$$

Example of discretization : pɣ interation

$$\frac{\partial N_i}{\partial t}(E_i) = S_i(E_i).$$

$$S_i(E_i) = \int_{E_p' \geq E_i} dE_p' \frac{cN(E_p')}{2{\gamma_p'}^2 \beta_p'} \int_{\epsilon} d\epsilon \frac{n}{\epsilon^2} \int d\epsilon_r \epsilon_r \int_{\psi} d\psi M_i \frac{\partial \sigma_{p\gamma}(\epsilon_r, \psi)}{\partial \psi} \delta(E_i - \xi_i(\psi, E_p', \epsilon) E_p')$$

Example of discretization : pɣ interation

$$\frac{\partial N_i}{\partial t}(E_i) = S_i(E_i).$$

$$S_i(E_i) = \int_{E_p^{\prime} \geq E_i} dE_p^{'} \frac{cN(E_p^{'})}{2\gamma_p^{'\,2}\beta_p^{'}} \int_{\epsilon} d\epsilon \frac{n}{\epsilon^2} \int d\epsilon_r \epsilon_r \int_{\psi} d\psi M_i \frac{\partial \sigma_{p\gamma}(\epsilon_r, \psi)}{\partial \psi} \delta(E_i - \xi_i(\psi, E_p^{'}, \epsilon) E_p^{'})$$

$$\int_{K_\pi} L_{0,K_\pi}(E_\pi) \frac{\partial N_\pi}{\partial t}(E_\pi) dE_\pi = \frac{\partial N_\pi^{K_\pi}}{\partial t} \times \int_{K_\pi} L_{0,K_\pi}^2(E_\pi) dE_\pi = \frac{\partial N_\pi^{K_\pi}}{\partial t}$$

Example of discretization : pɣ interation

$$\frac{\partial N_i}{\partial t}(E_i) = S_i(E_i).$$

$$S_i(E_i) = \int_{E_p' \geq E_i} dE_p' \frac{cN(E_p')}{2\gamma_p'^2 \beta_p'} \int_\epsilon d\epsilon \frac{n}{\epsilon^2} \int d\epsilon_r \epsilon_r \int_\psi d\psi M_i \frac{\partial \sigma_{p\gamma}(\epsilon_r, \psi)}{\partial \psi} \delta(E_i - \xi_i(\psi, E_p', \epsilon) E_p')$$

↓

$$\left\{ \begin{aligned} &\int_{K_\pi} L_{0,K_\pi}(E_\pi) \frac{\partial N_\pi}{\partial t}(E_\pi) dE_\pi = \frac{\partial N_\pi^{K_\pi}}{\partial t} \times \int_{K_\pi} L_{0,K_\pi}^2(E_\pi) dE_\pi = \frac{\partial N_\pi^{K_\pi}}{\partial t} \\ &= \int_{K_\pi} dE_\pi L_{0,K_\pi}(E_\pi) \int_{E_p' \geq E_\pi} dE_p' \frac{cN(E_p')}{2\gamma_p'^2 \beta_p'} \int_\epsilon d\epsilon \frac{n}{\epsilon^2} \int d\epsilon_r \epsilon_r \int_\psi d\psi M_i \frac{\partial \sigma_{p\gamma}(\epsilon_r, \psi)}{\partial \psi} \delta(E_\pi - \xi_i(E_p, E_p', \epsilon)) \end{aligned} \right.$$

Example of discretization : pɣ interation

$$\frac{\partial N_i}{\partial t}(E_i) = S_i(E_i).$$

$$S_i(E_i) = \int_{E_p' \geq E_i} dE_p' \frac{cN(E_p')}{2\gamma_p'^2 \beta_p'} \int_\epsilon d\epsilon \frac{n}{\epsilon^2} \int d\epsilon_r \epsilon_r \int_\psi d\psi M_i \frac{\partial \sigma_{p\gamma}(\epsilon_r, \psi)}{\partial \psi} \delta(E_i - \xi_i(\psi, E_p', \epsilon) E_p')$$

$$\int_{K_\pi} L_{0,K_\pi}(E_\pi) \frac{\partial N_\pi}{\partial t}(E_\pi) dE_\pi = \frac{\partial N_\pi^{K_\pi}}{\partial t} \times \int_{K_\pi} L_{0,K_\pi}^2(E_\pi) dE_\pi = \frac{\partial N_\pi^{K_\pi}}{\partial t}$$

$$= \int_{K_\pi} dE_\pi L_{0,K_\pi}(E_\pi) \int_{E_p' \geq E_\pi} dE_p' \frac{cN(E_p')}{2\gamma_p'^2 \beta_p'} \int_\epsilon d\epsilon \frac{n}{\epsilon^2} \int d\epsilon_r \epsilon_r \int_\psi d\psi M_i \frac{\partial \sigma_{p\gamma}(\epsilon_r, \psi)}{\partial \psi} \delta(E_\pi - \xi_i(E_p, E_p', \epsilon))$$

$$= \frac{c}{2} \sum_{\substack{I_p \\ E_p' > E_\pi}} \sum_{J_\gamma} \frac{N_p^{I_p} n_\gamma^{J_\gamma}}{\sqrt{||J_\gamma||.||I_p||.||K_\pi||}} \times \int_{K_\pi} \int_{I_p} \int_{J_\gamma} \frac{dE_\pi dE_p' d\epsilon}{\gamma_p'^2 \beta_p' \epsilon^2} \sigma_s(E_p, \epsilon, E_\pi)$$

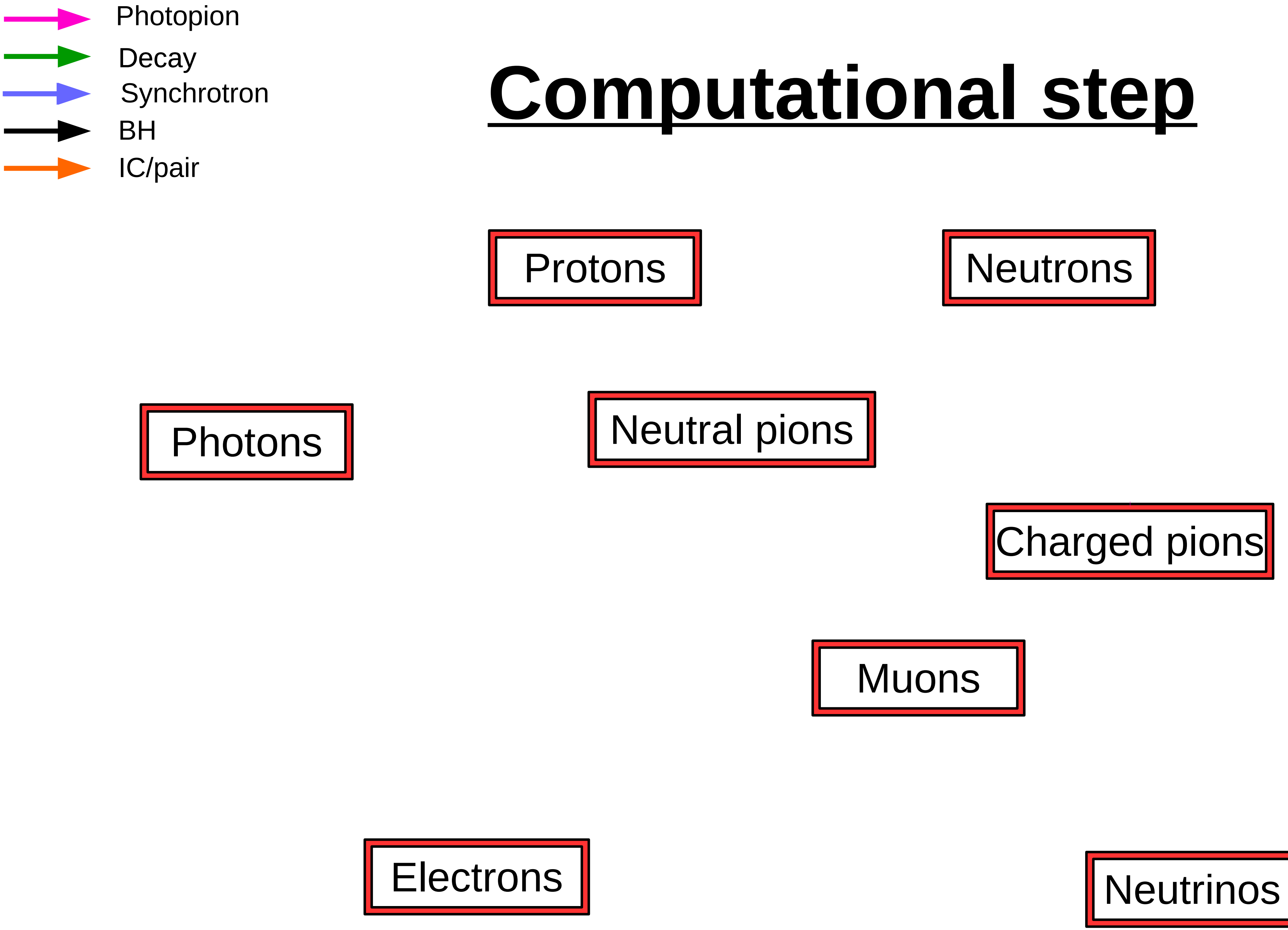
Example of discretization : Compton scattering

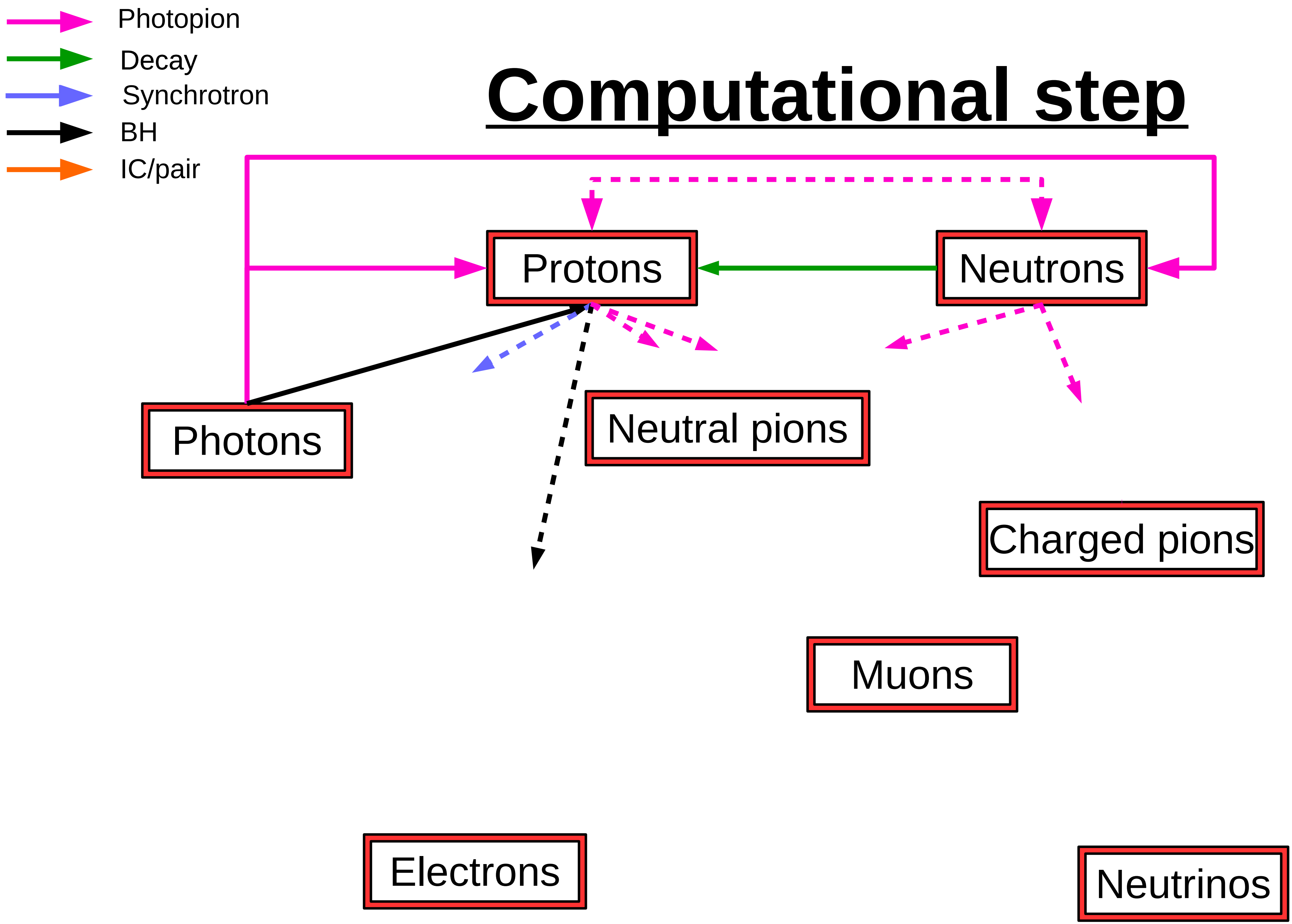
$$\frac{\partial n_{\rm ph}}{\partial t}(x_2) = \int_\gamma \int_{x_1} d\gamma dx_1 R(\gamma, x_1 \to x_2) N_{\rm e^\pm}(\gamma) n_{\rm ph}(x_1)$$
$$- n_{\rm ph}(x_2) \int_\gamma \int_{x_1} d\gamma dx_1 R(\gamma, x_2 \to x_1) N_{\rm e^\pm}(\gamma)$$

Example of discretization : Compton scattering

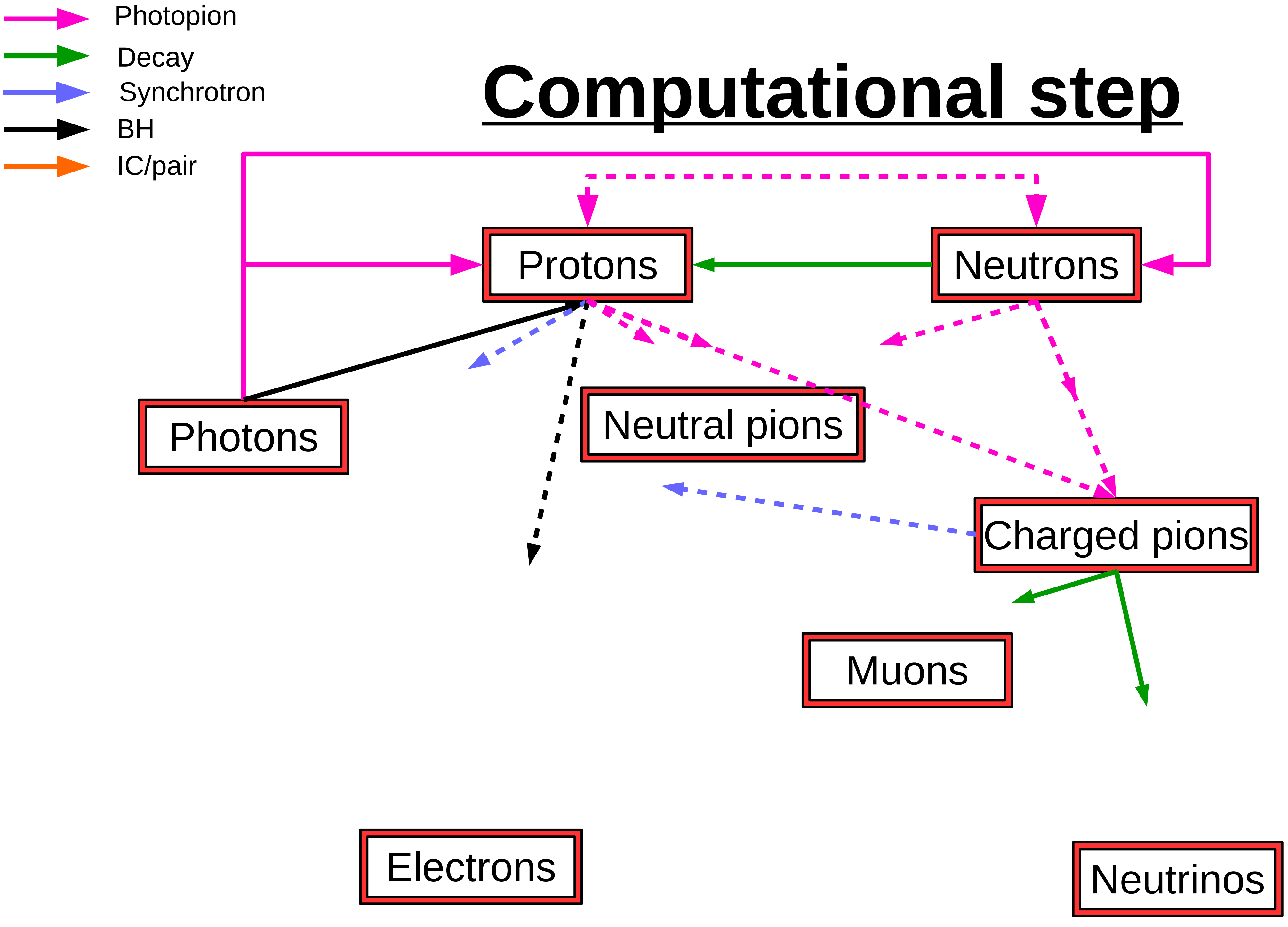
$$\frac{\partial n_{\rm ph}}{\partial t}(x_2) = \int_\gamma \int_{x_1} d\gamma dx_1 R(\gamma, x_1 \to x_2) N_{\rm e^\pm}(\gamma) n_{\rm ph}(x_1)$$
$$- n_{\rm ph}(x_2) \int_\gamma \int_{x_1} d\gamma dx_1 R(\gamma, x_2 \to x_1) N_{\rm e^\pm}(\gamma)$$

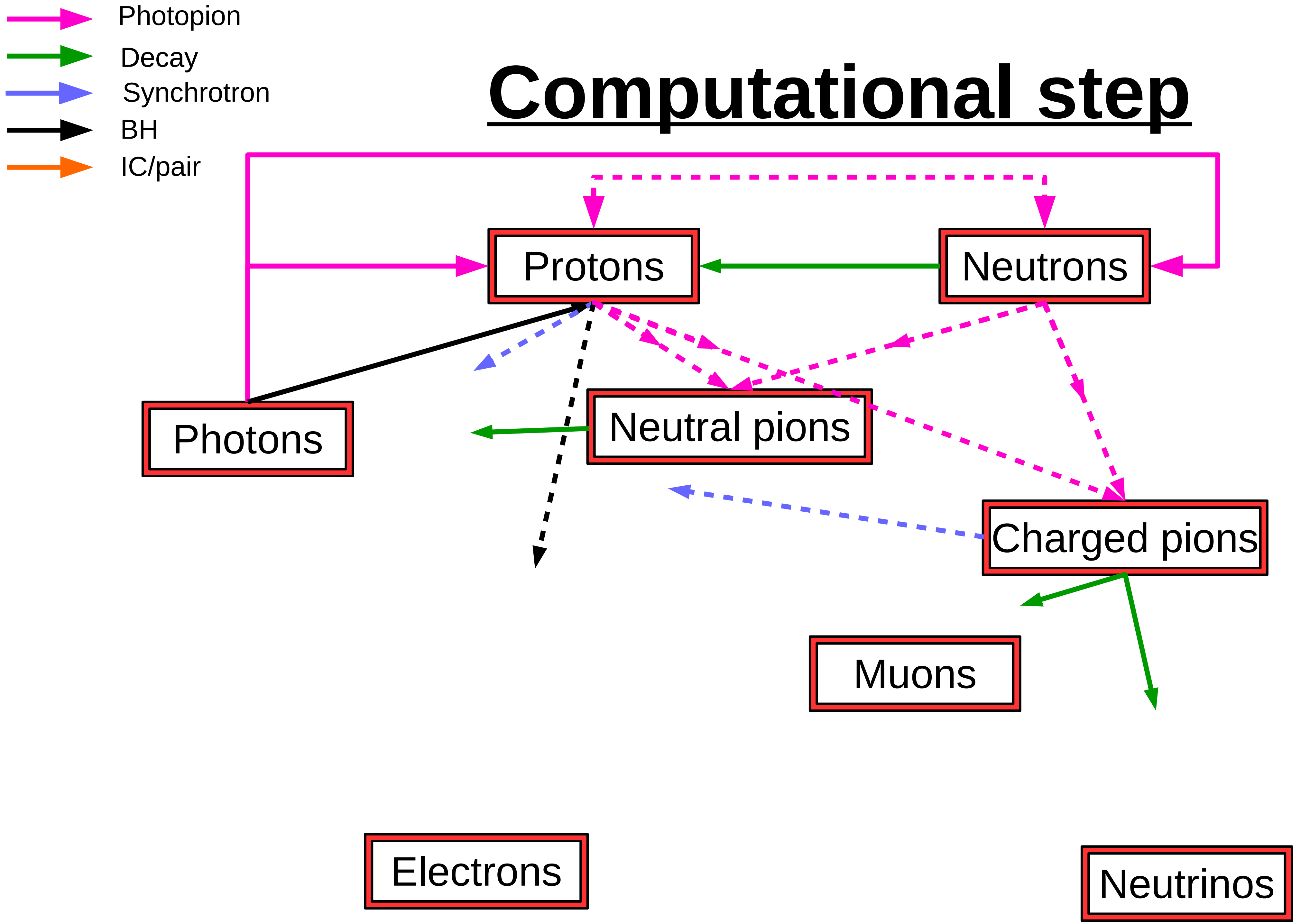
$$\frac{\partial n^J_{\rm ph}}{\partial t} = \frac{1}{\sqrt{||J||}} \sum_k \sum_{I<J} \frac{N^k_{\rm e}}{\sqrt{||K||}} \frac{n^I_{\rm ph}}{\sqrt{||I||}} \sigma_{IKJ}$$
$$- \frac{n^J_{\rm ph}}{||J||} \sum_k \frac{N^k_{\rm e}}{\sqrt{||K||}} \sum_{I>J} \sigma_{JKI}$$
$$\sigma_{IKJ} \equiv \int_I \int_J \int_K \sigma(\nu_I, \gamma_K \to \nu_J) d\nu_I d\nu_J d\gamma_K$$


Photopion
Decay
Synchrotron
BH
IC/pair
Computational step
Protons
Neutrons
Neutral pions
Photons
Charged pions
Muons
Electrons
Neutrinos

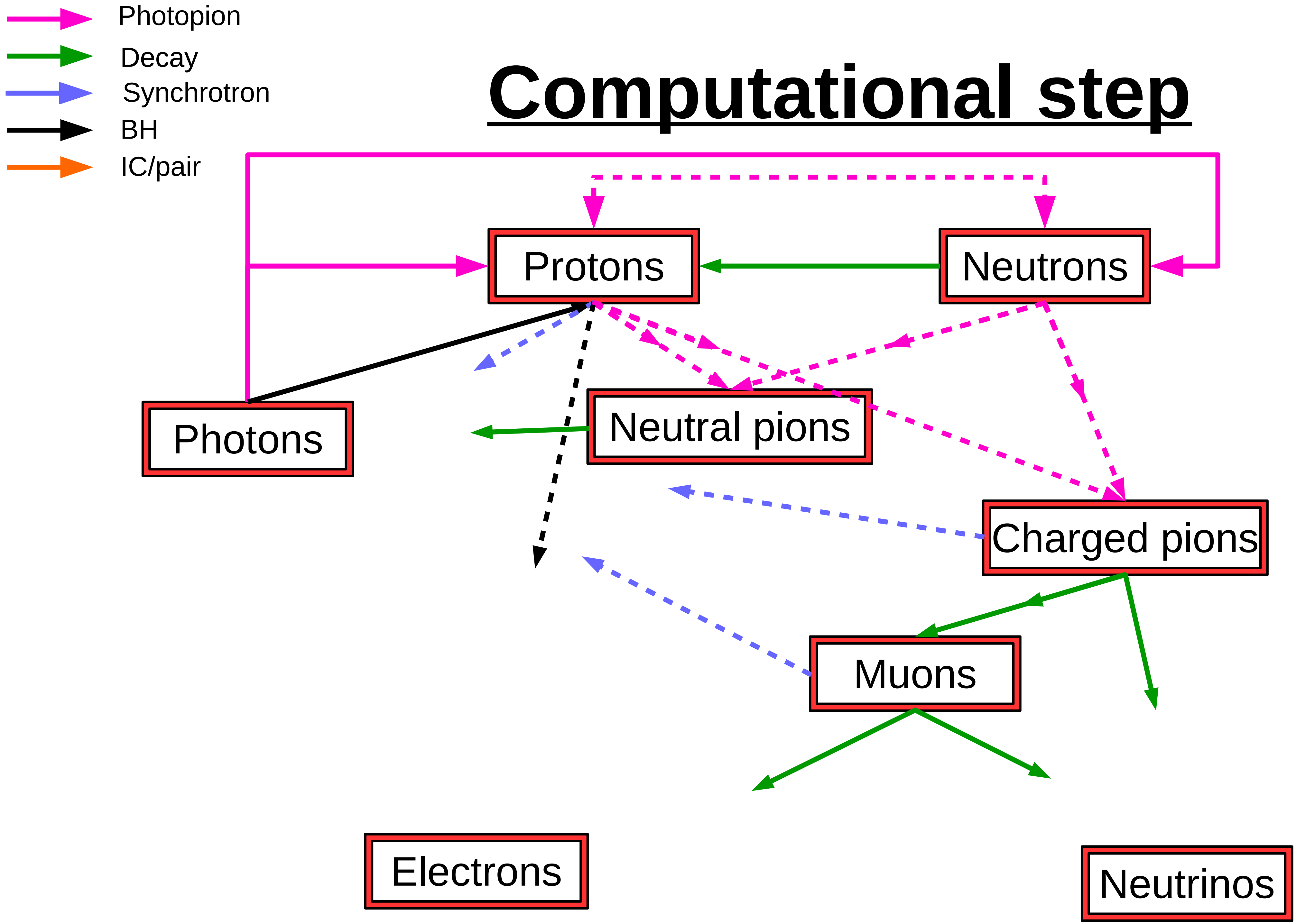

Photopion
Decay
Synchrotron
BH
IC/pair
Computational step
Protons
Neutrons
Photons
Neutral pions
Charged pions
Muons
Electrons
Neutrinos

Computational step

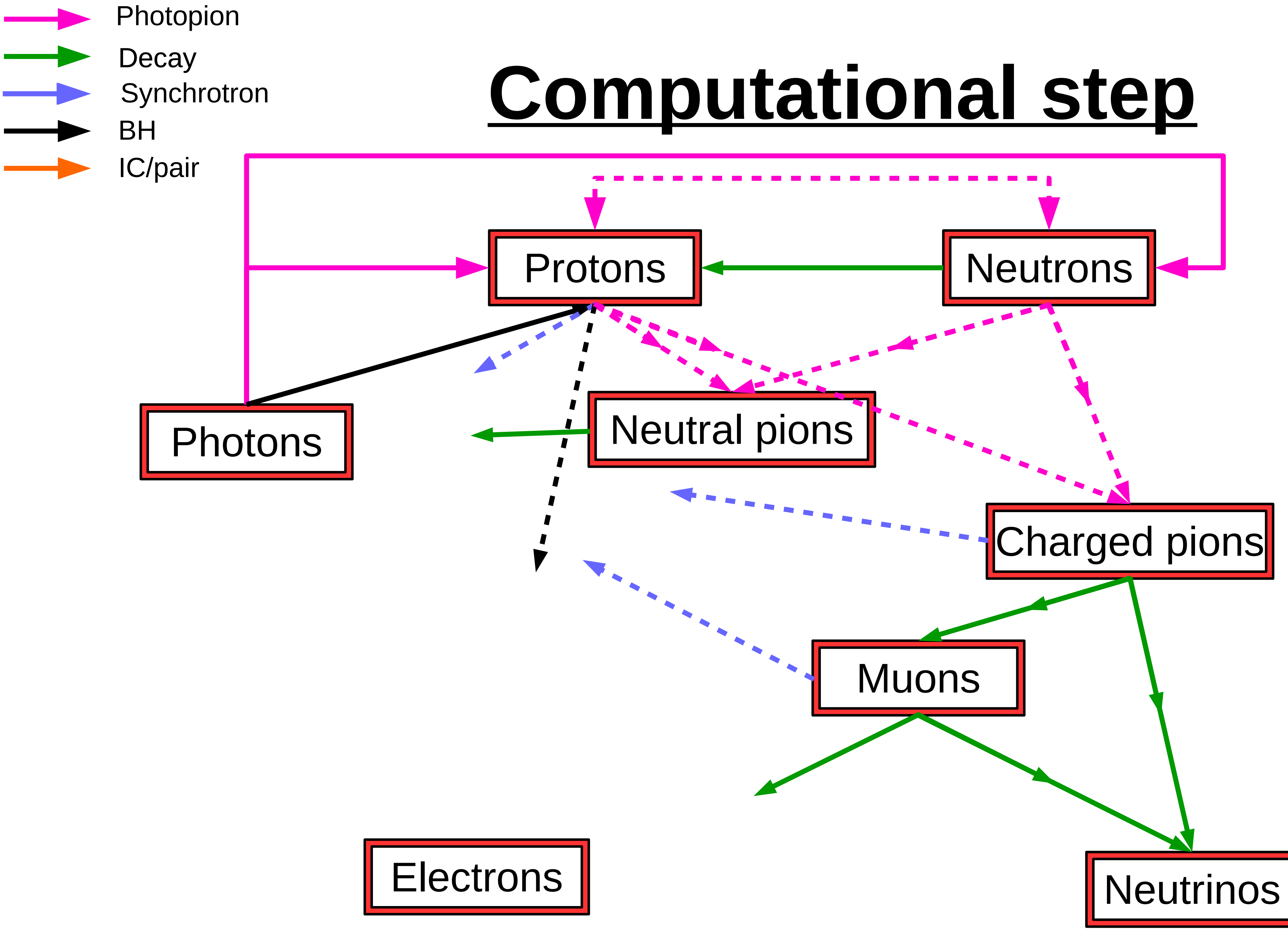


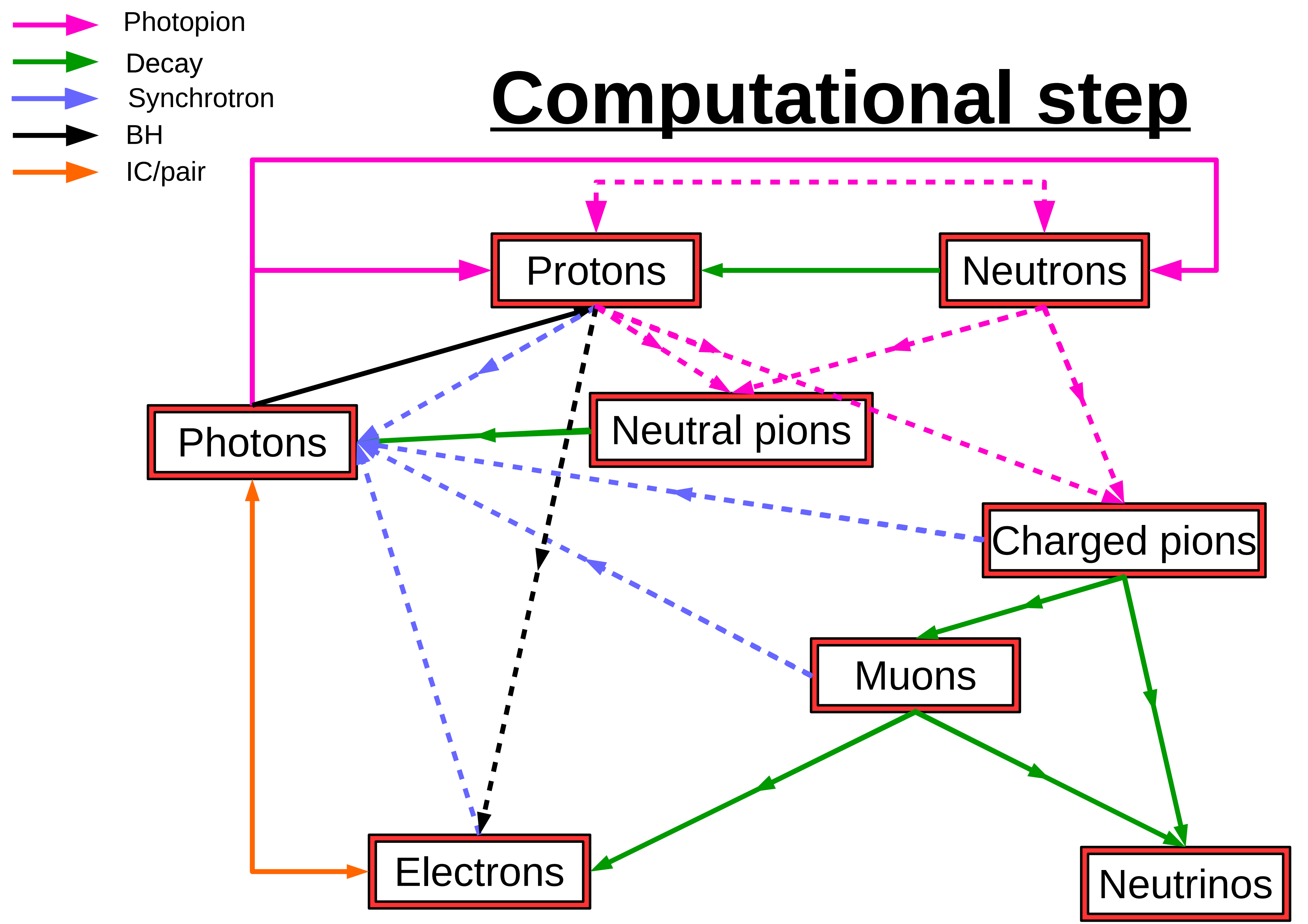


Photopion
Decay
Synchrotron
BH
IC/pair
Computational step
Protons
Neutrons
Photons
Neutral pions
Charged pions
Muons
Electrons
Neutrinos

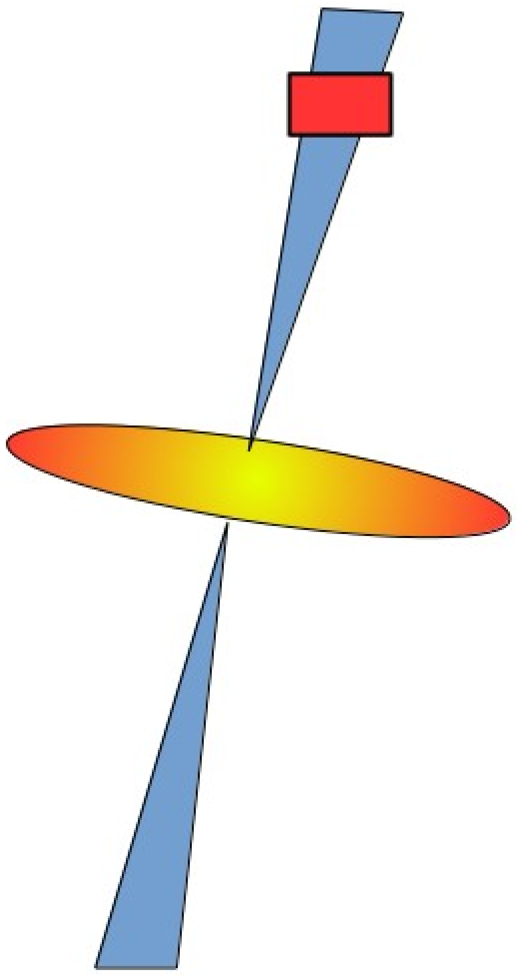

Photopion
Decay
Synchrotron
BH
IC/pair
Computational step
Protons
Neutrons
Photons
Neutral pions
Charged pions
Muons
Electrons
Neutrinos

Computational step

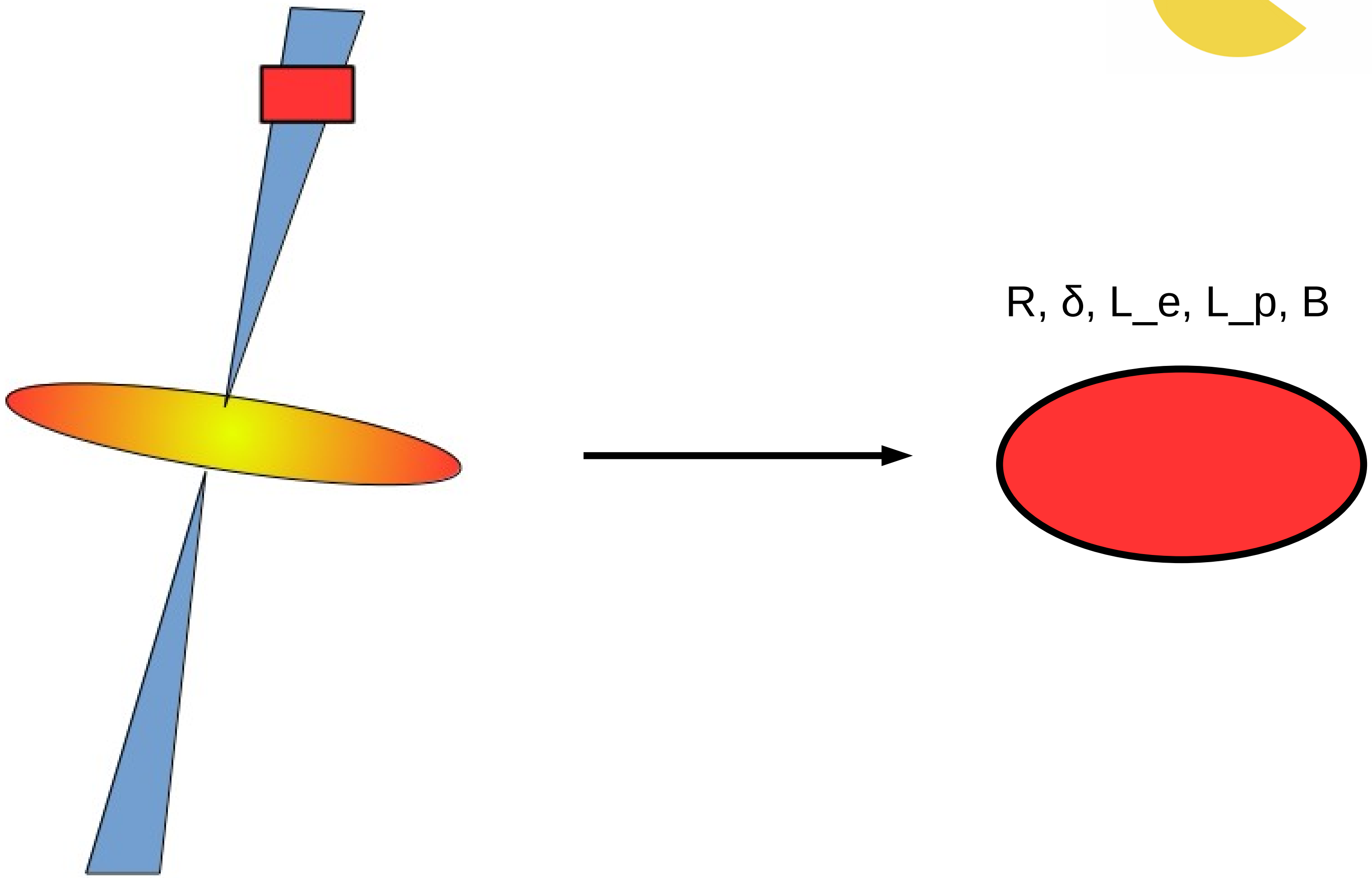



Computational step
Photopion
Decay
Synchrotron
BH
IC/pair
Protons
Neutrons
Photons
Neutral pions
Charged pions
Muons
Electrons
Neutrinos

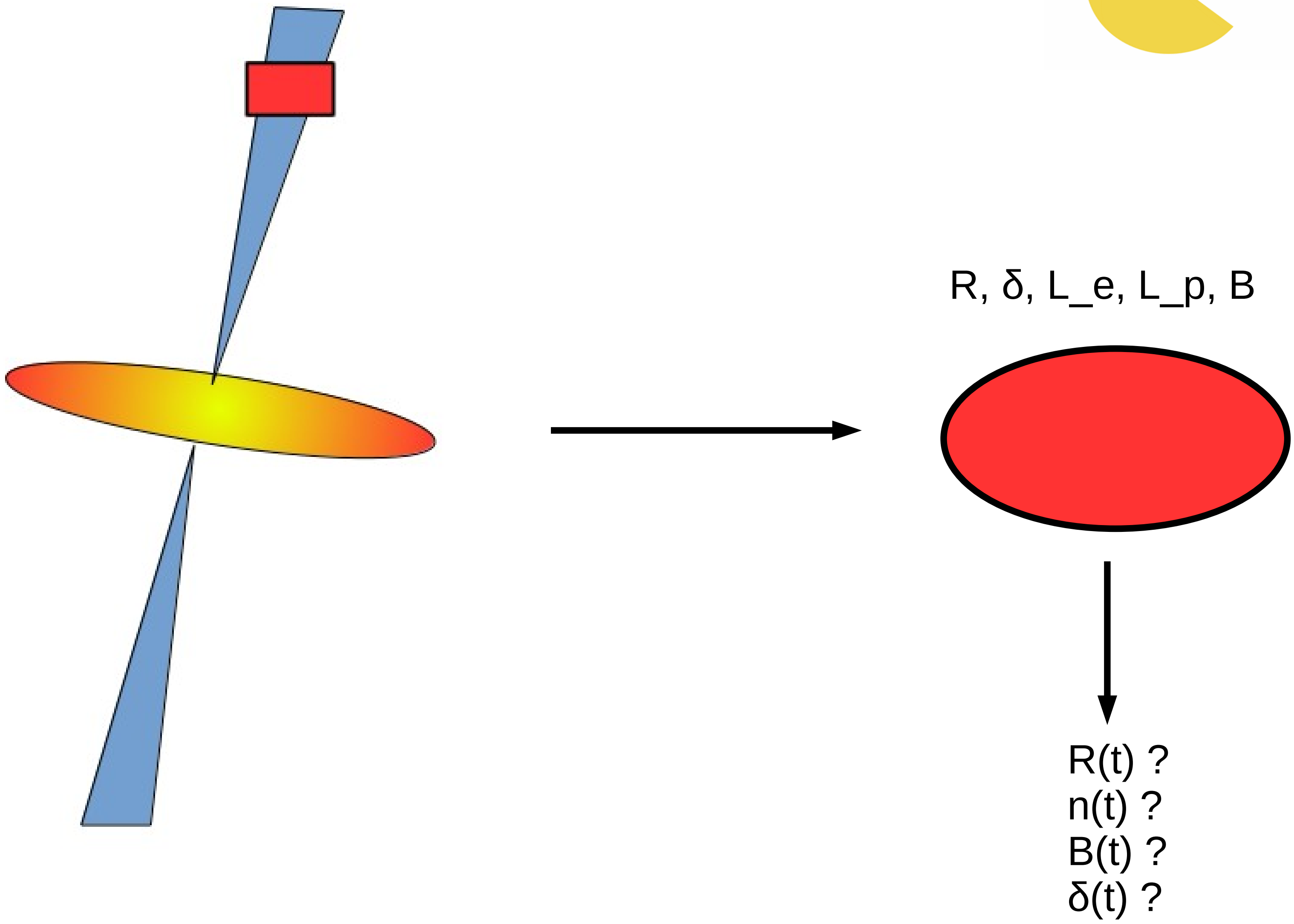
Physical / geometrical set up



Physical / geometrical set up



Physical / geometrical set up



Physical / geometrical set up

Stalled shock :

- Escape of particles,
- Escape of photons.

Physical / geometrical set up

Stalled shock :

- Escape of particles,
- Escape of photons.

Emission region changing with time :

- Adiabatic cooling,
- Particle escape,
- Equation of motion dictates how all important quantities are changing.

Physical / geometrical set up

Stalled shock :

- Escape of particles,
- Escape of photons.

Emission region changing with time :

- Adiabatic cooling,
- Particle escape,
- Equation of motion dictates how all important quantities are changing.

Dictated by the physical problem.

↓

Trivial changes to the code

Conclusion

- We wrote SOPRANO:
  - Efficient numerical computation (~20-30s per run)

Conclusion

- We wrote SOPRANO:
  - Efficient numerical computation (~20-30s per run)
  - Price to pay: heavy numerical computation for the grid (little flexibility).

Conclusion

- We wrote SOPRANO:
  - Efficient numerical computation (~20-30s per run)
  - Price to pay: heavy numerical computation for the grid (little flexibility).
- SOPRANO is currently used for studying blazars.