Formalization of Ray Optics

(Highlights)

Umair Siddique

Hardware Verification Group (HVG)
Department of Electrical and Computer Engineering,
Concordia University, Montreal, Quebec, Canada

Formalization of Physics Vs Mathematics

- Computational Physic is richer than computational mathematics*

Mathematics

Numbers

Numbers

Numbers

Physics

Physical Quantities

↓

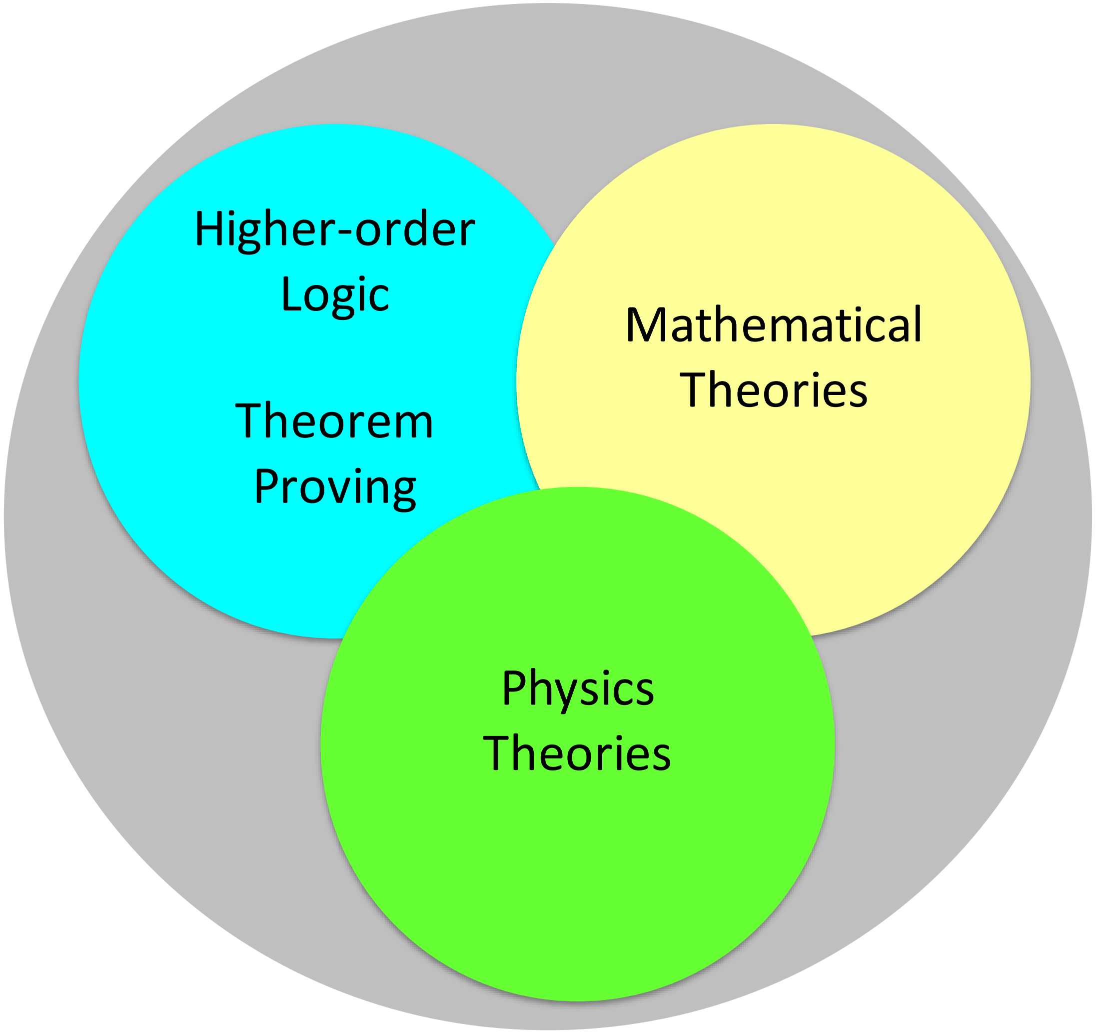
(numbers)

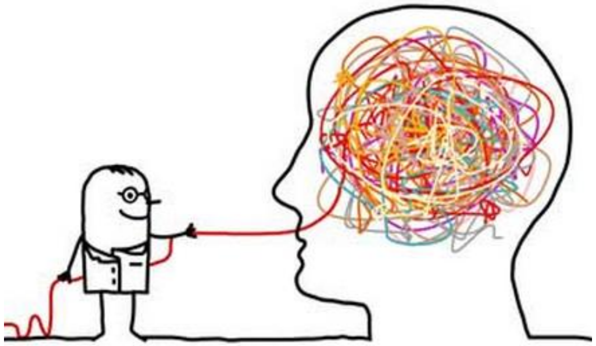
↑

Information

* Enzo Tonti, Why Starting with Differential Equations for Computational Physics?, JCP, 2014

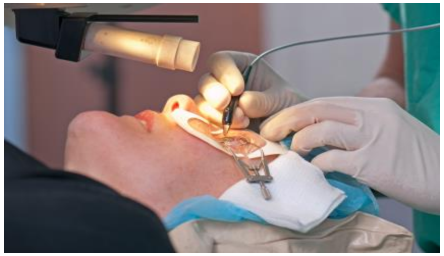
Formalization of Physics

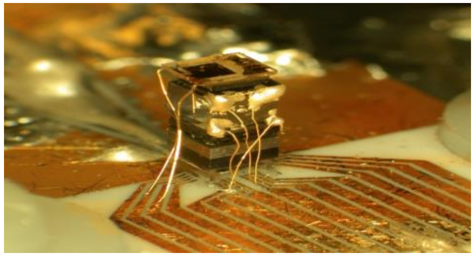




Why Optics?

- In particular:
  - widely used theory
  - Safety & mission critical application

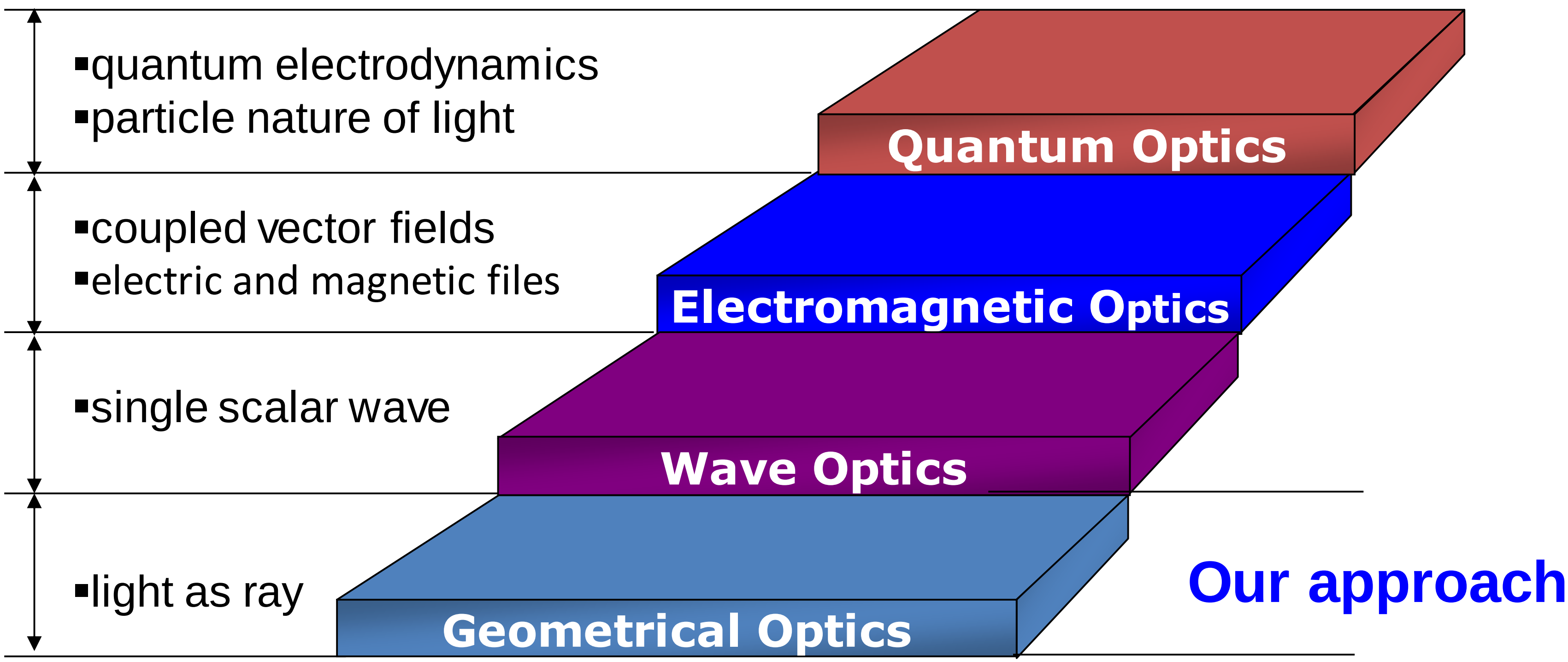




- More complex than usual electronic hardware
  - difficult to verify
- In general:
  - Same motivation as of formalized mathematics

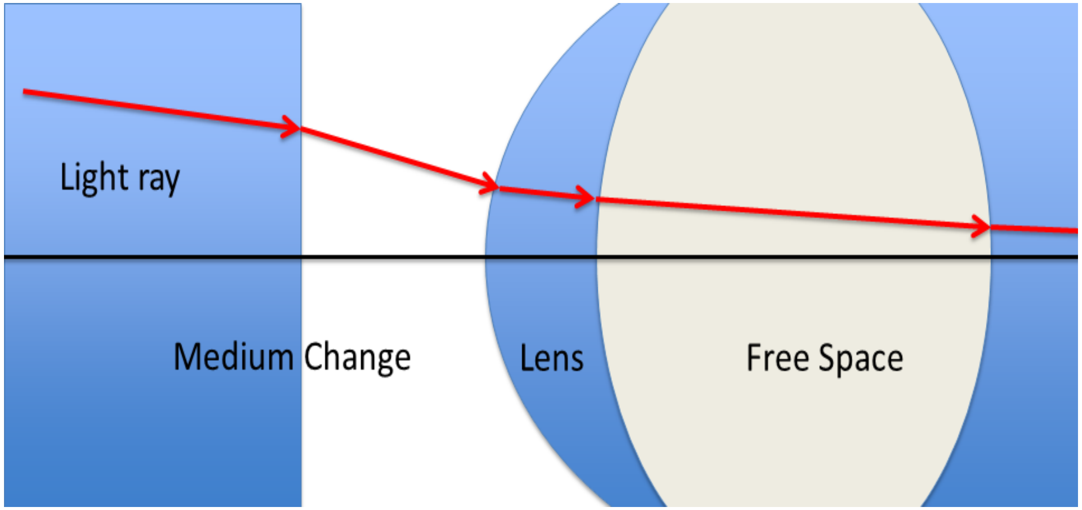
Optics Approaches

4 levels of abstraction



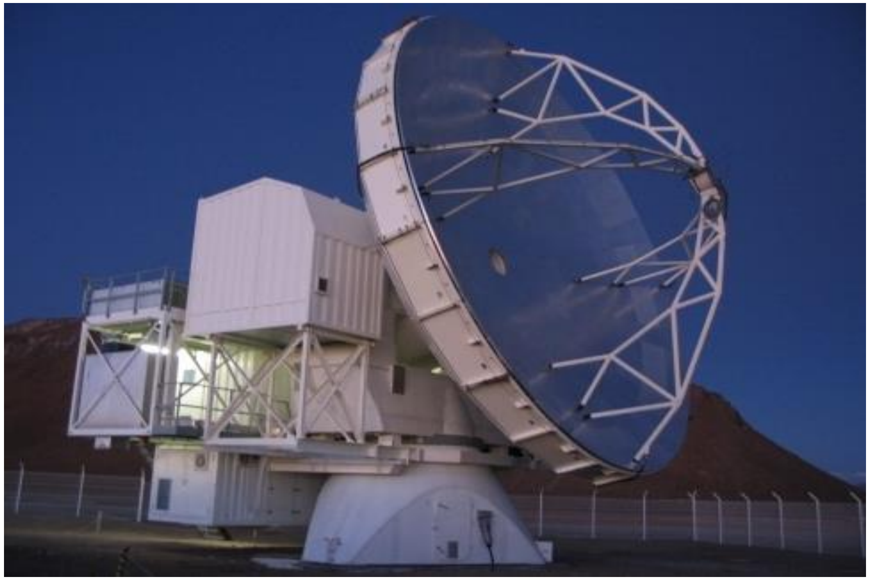
Geometrical Optics

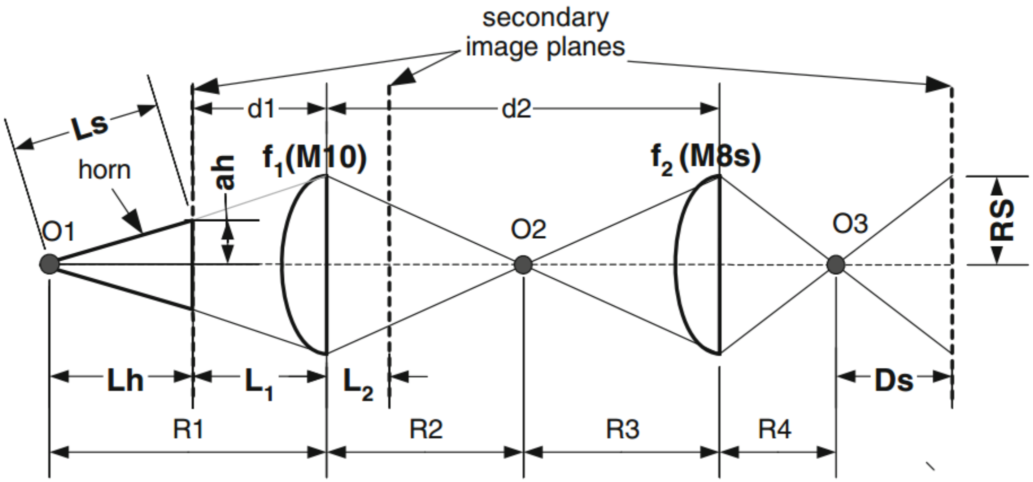
- Light is a ray / beam
- Optical medium (refractive index)
- Fermat's principle of least time



Why Geometrical Optics ?

Atacama Pathfinder Experiment
APEX Telescope





Our Approach

Formalize underlying theories of optics

- rays, beams, imaging properties
- unified framework, unambiguous proofs
- time consuming

Formalize necessary mathematical theories

- complex matrices, eigenvalues, etc.
- can be used in other fields

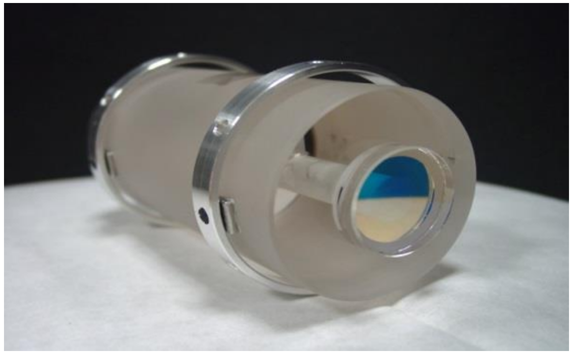
Verify practical optical systems

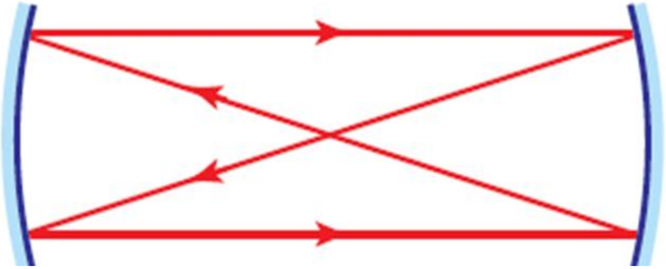
- lenses, ray tracing algorithms
- accurate
- can be complimented with existing methods

A Real-World Example (1/4)

- Optical resonator

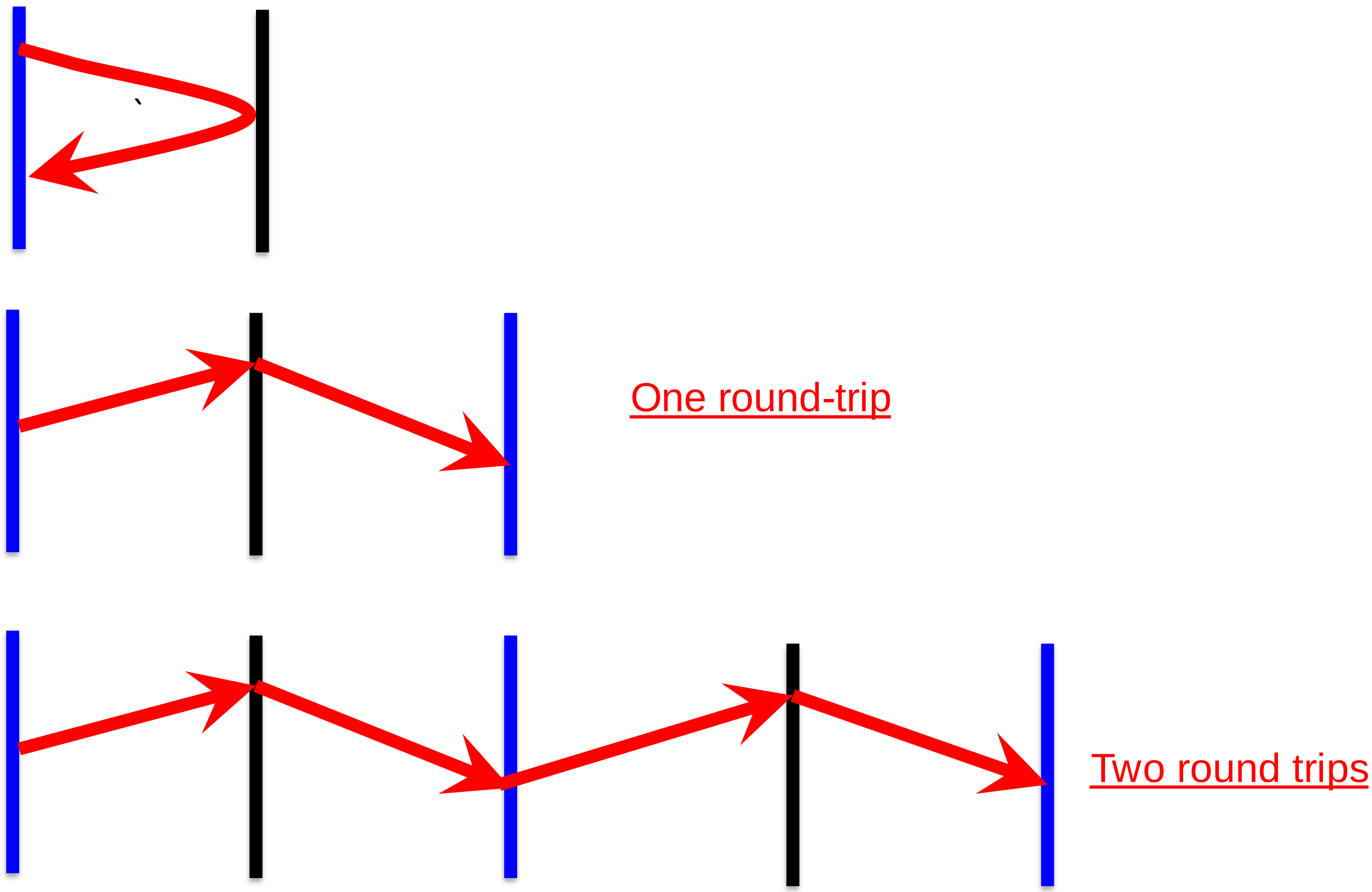
  - ensures the confinement of light within optical cavity



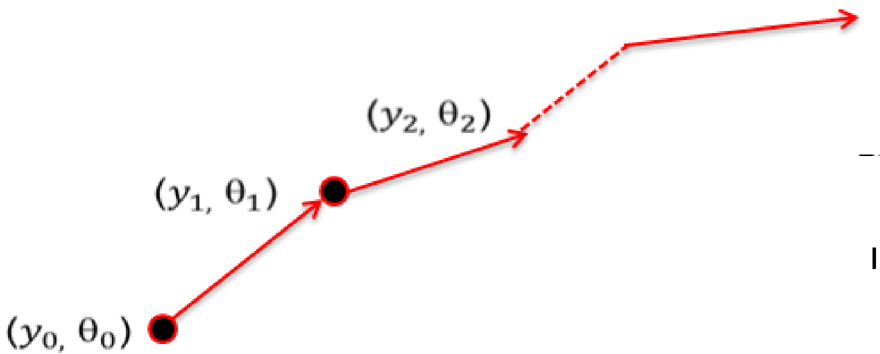


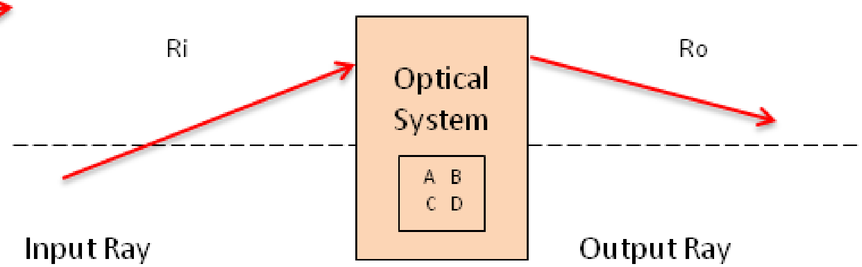
- Some practical uses
  - Lasers
  - Biological sensing
  - Optical transmission

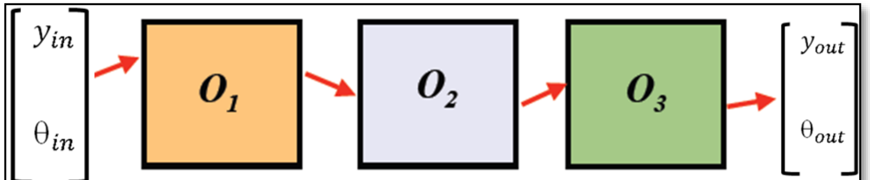
A Real-World Example (2/4)



A Real-World Example (3/4)







$$\begin{bmatrix} y_{out} \\ \theta_{out} \end{bmatrix} = O_3 \left\{ O_2 \left(O_1 \begin{bmatrix} y_{in} \\ \theta_{in} \end{bmatrix} \right) \right\} = O_3\, O_2\, O_1 \begin{bmatrix} y_{in} \\ \theta_{in} \end{bmatrix}$$

$$\begin{bmatrix} y_n \\ \theta_n \end{bmatrix} = \prod_{i=n}^{1} M_i \begin{bmatrix} y_0 \\ \theta_0 \end{bmatrix}$$

A Real-World Example (4/4)

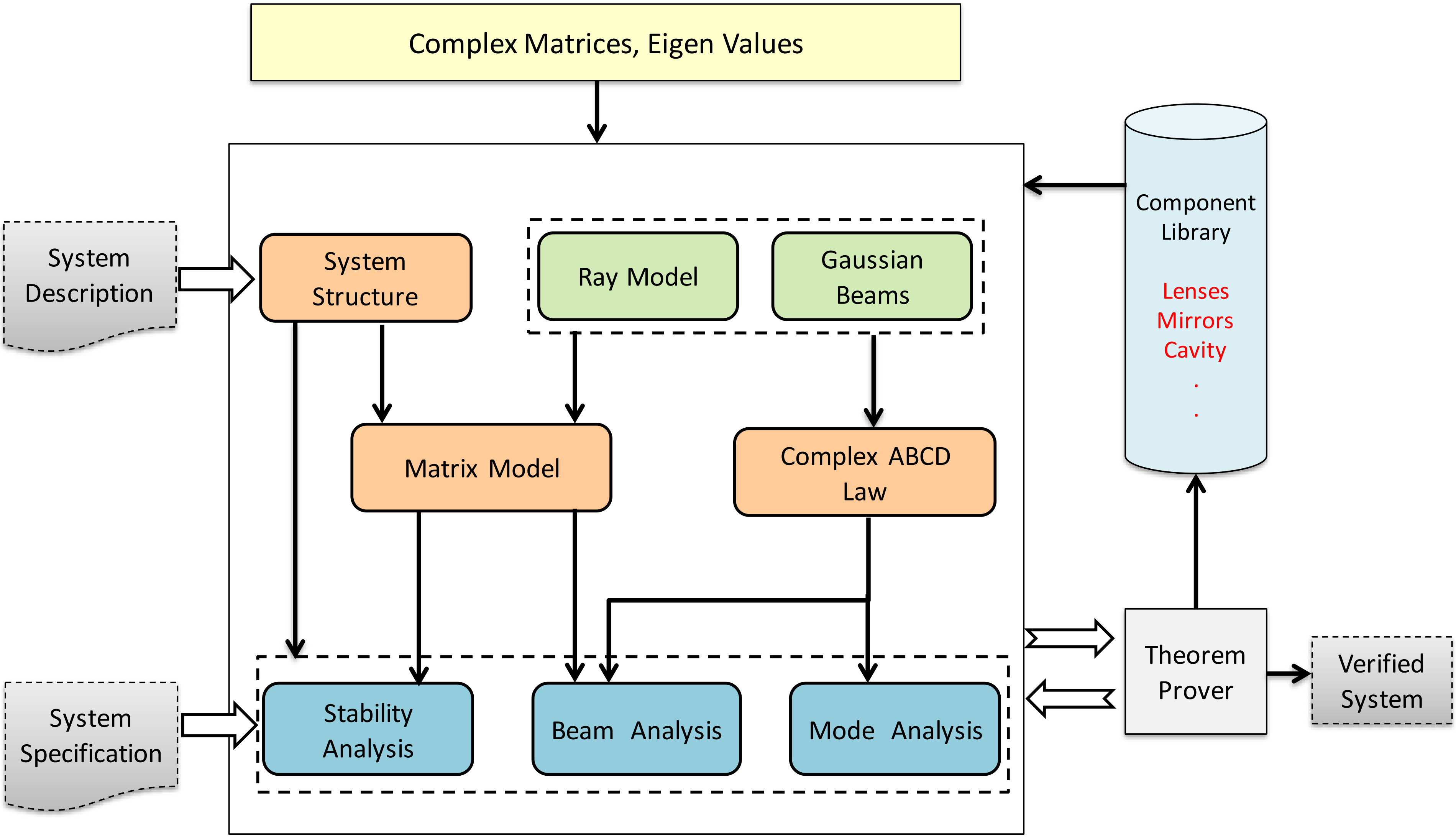
Formal Definition of Stability

$$\vdash \forall \texttt{res}.\ \texttt{is\_stable\_resonator res} \Leftrightarrow (\forall \texttt{r}.\ \exists \texttt{y}\ \theta.\ \forall \texttt{N}.$$
$$\texttt{is\_valid\_ray\_in\_system r (unfold\_resonator res N)} \Longrightarrow$$
$$(\texttt{let } y_n, \theta_n = \texttt{last\_single\_ray r in abs}(y_n) \leq y \wedge \texttt{abs}(\theta_n) < \theta))$$

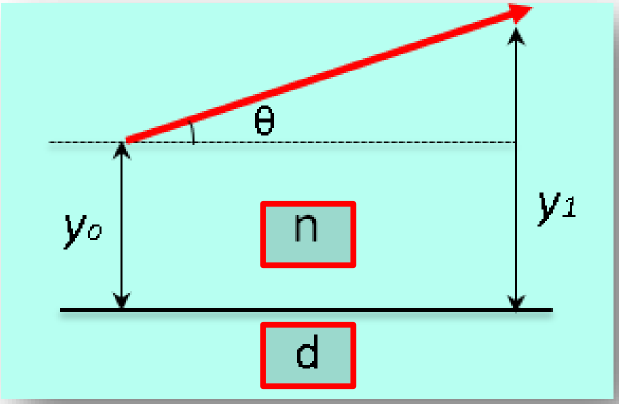
General Stability Theorem

$$\vdash \forall \texttt{res}.\ \texttt{is\_valid\_resonator res} \wedge$$
$$(\forall \texttt{N}.\ \texttt{let M = system\_composition (unfold\_resonator res 1) in}$$
$$\texttt{det M} = 1 \wedge -1 < \frac{M_{1,1}+M_{2,2}}{2} \wedge \frac{M_{1,1}+M_{2,2}}{2} < 1) \Longrightarrow$$
$$\texttt{is\_stable\_resonator res}$$

Proposed Framework



Formalization of Free Space

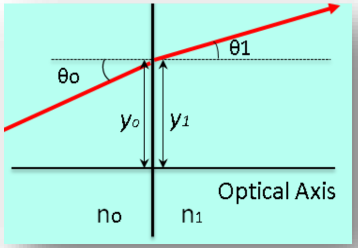


Free space: refractive index + distance

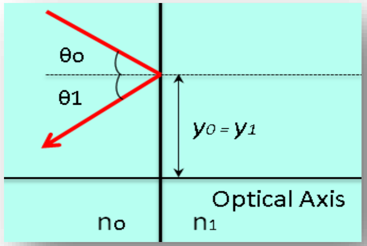
Definition: Free Space

$$\text{type } \textbf{free\_space} = \mathbb{R} \times \mathbb{R}$$

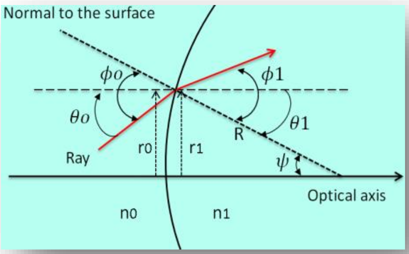
Optical Interface



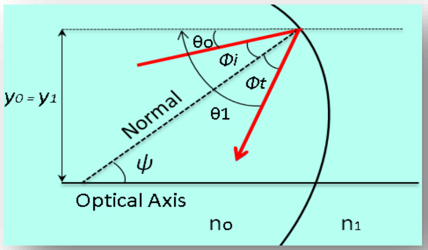
Plane
(transmitted)



Plane
(reflected)



Spherical
(transmitted)



Spherical
(reflected)

Definition: Optical Interface

$$\texttt{type}\ \textbf{interface} = \texttt{plane} \mid \texttt{spherical}(\mathbb{R})$$

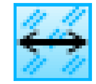
$$\texttt{type}\ \textbf{behaviour} = \texttt{reflected} \mid \texttt{transmitted}$$

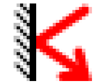
Component Library

-Formal definitions

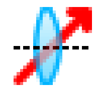
-Validity Constraints

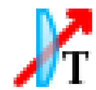
-Useful properties


Free Space

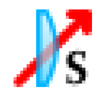

Plane Mirror


Spherical Mirror

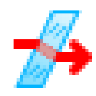




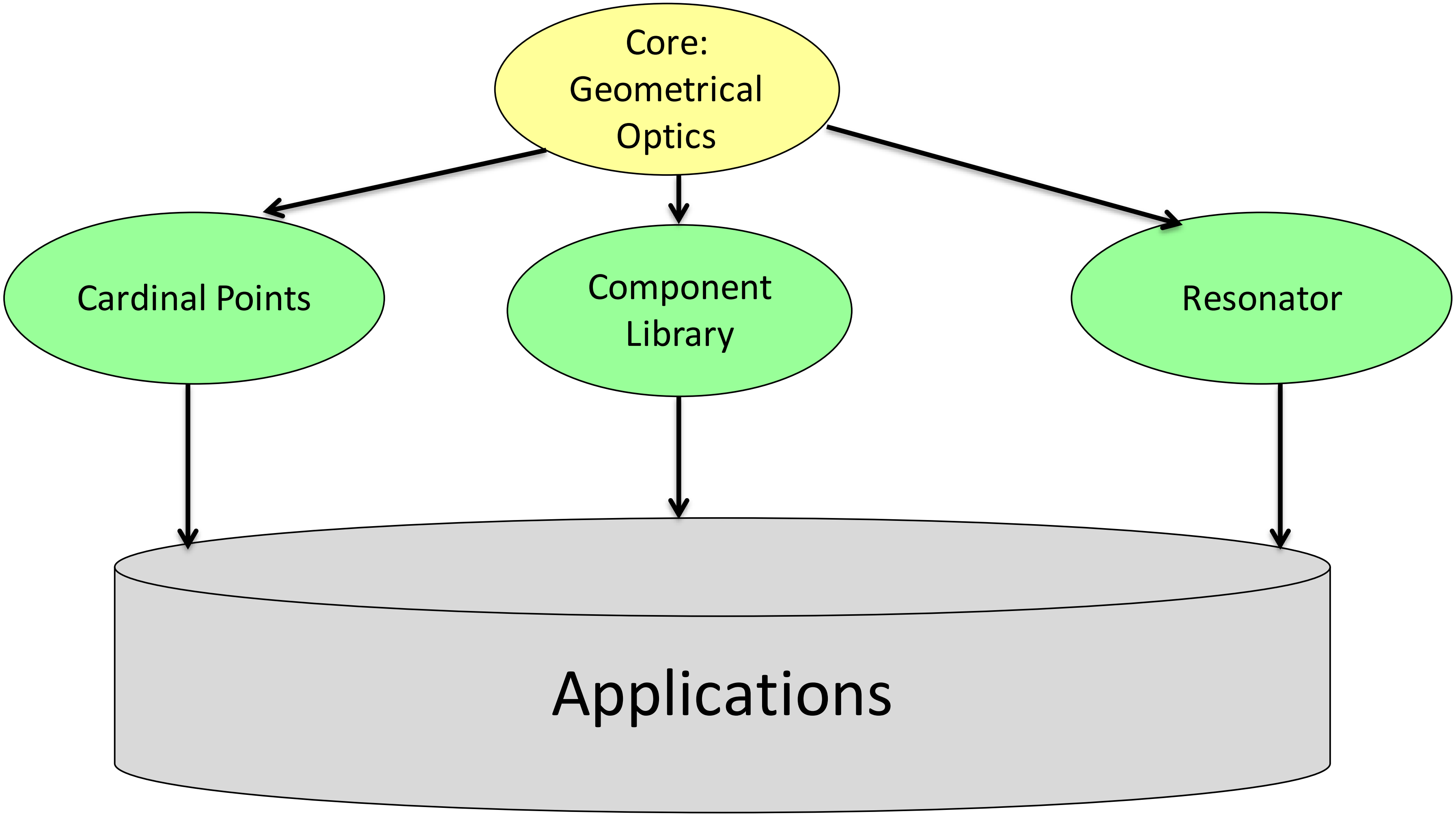
Tangential Cylindrical thin lens



Saggital Cylindrical thin lens


Tilted Parallel Plate

Hierarchy of HOL Light Developments



Conclusion

- Our focus: Geometrical Optics
- Theorem proving: best complementary approach
- Automation ?

- More readable proofs
- Graphical user interface

Thank You !!!

http://hvg.ece.concordia.ca/projects/optics/rayoptics.htm

HOL Light Code

```
let STABILITY_THEOREM_SYM = prove (
 `!res. is_valid_resonator res /\
     ((M:real^2^2) pow 2 = system_composition (unfold_resonator res 1)) /\
     (det (M:real^2^2) = &1) /\ -- &1 < (M$1$1 + M$2$2) / &2 /\ (M$1$1 + M$2$2) / &2 < &1
       ==> is_stable_resonator res`,
```

```
GEN_TAC THEN ONCE_REWRITE_TAC[EQ_SYM_EQ] THEN REPEAT STRIP_TAC THEN POP_ASSUM MP_TAC
THEN
POP_ASSUM MP_TAC THEN ONCE_REWRITE_TAC[EQ_SYM_EQ] THEN
REPEAT STRIP_TAC THEN REWRITE_TAC[is_stable_resonator] THEN
GEN_TAC THEN
SUBGOAL_THEN ` (?Y:real^2. !n. abs (((M pow 2) pow n ** vector [FST(fst_single_ray r);SND(fst_single_ray r) ])$1) <=
Y$1 /\
          abs ((((M:real^2^2) pow 2 ) pow n ** vector [FST(fst_single_ray r);SND(fst_single_ray r) ])$2) <= Y$2) `
ASSUME_TAC THENL[
MP_REWRITE_TAC STABILITY_LEMMA_GENERAL_SYM THEN ASM_SIMP_TAC[]; ALL_TAC] THEN
POP_ASSUM MP_TAC THEN STRIP_TAC THEN EXISTS_TAC(`((Y:real^2)$1):real`) THEN
EXISTS_TAC(`((Y:real^2)$2):real`) THEN REPEAT STRIP_TAC THEN
LET_TAC THEN SUBGOAL_THEN `(let (xi,thetai),(y1,theta1),rs = r in
      let y',theta' = last_single_ray r in
      vector [y'; theta'] =
      system_composition ((unfold_resonator res n):optical_system) **
      vector [xi; thetai])` ASSUME_TAC THENL[
MATCH_MP_TAC SYSTEM_MATRIX THEN ASM_SIMP_TAC[ VALID_UNFOLD_RESONATOR];ALL_TAC] THEN
POP_ASSUM MP_TAC THEN ONCE_REWRITE_TAC[MAT2X2_VECTOR_MUL_ALT] THEN
ONCE_REWRITE_TAC[RESONATOR_MATRIX] THEN ONCE_ASM_REWRITE_TAC[] THEN LET_TAC THEN
LET_TAC THEN DISCH_TAC THEN ONCE_ASM_REWRITE_TAC[] THEN REPEAT(POP_ASSUM MP_TAC) THEN
REWRITE_TAC[fst_single_ray;FST;SND] THEN SIMP_TAC[]);;
```