



# CONCEPTION ET MONTAGE D'UN RECEPTEUR SOLAIRE POUR CENTRALE A TOUR

MEMOIRE POUR L'OBTENTION DU  
MASTER D'INGENIERIE EN SCIENCES DE L'EAU ET DE L'ENVIRONNEMENT.  
OPTION : Génie Energétique

-----

Présenté et soutenu publiquement le 21 Juin 2013 par

**Yao Manu SESHIE**

**Travaux dirigés par :** **Ing. Emmanuel CLERC**  
Ingénieur de Recherche au LESEE  
**Ing. Madieumbe GAYE**  
Ingénieur de Recherche au LESEE  
**Dr. Yao K. AZOUMAH**  
Enseignant-Chercheur au LESEE  
Centre Commun de Recherche Energie et Habitat Durables

Jury d'évaluation du stage :

Président : Dr. Daniel YAMEGUEU

Membres et correcteurs : Drt. David TSUANYO  
Ing. Madieumbe GAYE  
Ing. Emmanuel CLERC

**Promotion [2012/2013]**



**Institut International d'Ingénierie de l'Eau et de l'Environnement**

Fondation 2iE - Rue de la Science - 01 BP 594 - Ouagadougou 01 - BURKINA FASO – IFU 00007748B  
Tél. : (+226) 50. 49. 28. 00 - Fax : (+226) 50. 49. 28. 01 - Mail : [2ie@2ie-edu.org](mailto:2ie@2ie-edu.org) - [www.2ie-edu.org](http://www.2ie-edu.org)





## Dédicaces et Remerciements

A mon père pour les efforts consentis et les conseils prodigués jusque lors,  
A ma mère pour l'amour et le soutien inconditionnels et sans relâche,  
A ma grande sœur pour les joyeux et tristes moments des jours passés,  
A mon beau-frère Hervé AYITE, à Yvon KPATIVOR, AMOUZOU Boniface et à Claude AKOUSSAN pour avoir été des modèles à différents moments de ma vie,  
A mes cousins SESHIE Landry et Jean, AGBOBLI-ATAYI Ayi et Cécile, et à Beny JOHNSON pour les merveilleux moments passés ensemble,  
Et aux proches qui ont marqué mon existence à tout jamais.

Mes remerciements vont :

Au Dr. Yao K. AZOUMAH pour la confiance accordée pour mes stages du cycle ingénieur,  
A David Blaise TSUANYO pour l'amitié, la confiance et les conseils,  
A Emmanuel CLERC pour l'encadrement, la confiance et la présence,  
A Henri KOTTIN pour avoir été un modèle de chercheur,  
A Madieumbe GAYE et au personnel du LESEE pour leurs conseils et encouragements.

## Résumé

L'objectif de ce stage était de concevoir, dimensionner et fabriquer un récepteur solaire pour la centrale solaire CSP4AFRICA (centrale à tour d'une puissance de 100 kWth au niveau du champ solaire). Après une étude bibliographique détaillée sur les récepteurs de type centrale à tour, nous avons opté pour un récepteur solaire de type cavité à absorbeur tubulaire pour ses pertes thermiques faibles et une absorptivité apparente excellente. Une première modélisation basée sur le 'Lancer des rayons' a été implémentée sous environnement Python® pour comparer les performances des cavités parallélépipédiques et cubiques du point de vue des émissions infrarouges. Les pertes radiatives ont été évaluées à 2% de la puissance reçue avec un léger avantage pour la géométrie cubique. Dans le même temps, la géométrie cylindrique a été adoptée pour sa simplicité de mise en œuvre au niveau local et les pertes de charges faibles du corps de chauffe. Une deuxième modélisation prenant en compte les échanges radiatifs et les transferts convectifs avec le fluide caloporteur, basée sur les travaux de thèse de Stefania TESCARI dirigés par Pierre NEVEU nous a permis de concevoir finalement notre récepteur. Le dimensionnement a abouti à une hauteur de 1m, un diamètre d'ouverture équivalent au diamètre du cylindre de 0,7m. La fabrication n'étant pas terminée à temps, nous avons proposé un protocole expérimental comportant deux expériences pour la caractérisation du récepteur : l'un pour la vérification de l'isolation et l'autre, pour l'évaluation du profil de température sur la paroi qui sera une base de données précieuse pour l'amélioration des modélisations.

Mots clés : Récepteur solaire, récepteur à cavité, absorbeur tubulaire, facteur de forme, pertes thermiques

## Abstract:

The objective of this internship is to design, size and manufacture a solar receiver for the concentrating solar plant CSP4AFRICA (power tower with 100 kWth solar array). After a detailed state of art on central solar receivers, we opted for cavity type solar receiver with tubular absorber for his lower thermal losses and his excellent absorptivity. A first model based on 'Ray tracing' was implemented under Python® programming environment to compare the performances of parallelepiped and cylindrical cavities in term of infrared emissions. Radiative losses were estimated at 2% of power received with a little advantage for the parallelepiped geometry. At the same time, cylindrical geometry was choose for the simplicity of his manufacture and the pressure



losses of fluid system. A second model taking into account radiative exchanges and convective transfers with the heat fluid, based on thesis of TESCARI Stefania directed by Pierre NEVEU finally help us design our receiver. The sizing gives a height of 1m, an aperture diameter equivalent to cylinder diameter equal to 0,7m. We suggested a protocol comprising two experiences to characterize the receiver: one to make sure the receiver is well insulated and the other one, to estimate the temperature profile on the wall in order to generate a database for the improvements of the models.

**Keywords:** Solar receiver, cavity receiver, tubular absorber, view factor, thermal losses



### Liste des abréviations

CSP: Concentrating Solar Plant

RS: Récepteur solaire

## Sommaire

I.	Introduction .....	9
1.	Contexte.....	9
2.	Présentation du lieu du stage .....	11
II.	Synthèse bibliographique .....	12
1.	Centrales solaires à concentration .....	12
a.	Centrales solaires à capteurs cylindro-paraboliques.....	13
b.	Centrales solaires à capteurs linéaires de Fresnel .....	14
c.	Les centrales solaires à tour .....	15
d.	Les centrales à capteurs paraboliques .....	15
2.	Récepteurs solaires pour Centrales à tour .....	16
a.	Les récepteurs externes plans.....	17
b.	Les récepteurs à cavité.....	18
III.	Hypothèses de travail .....	23
IV.	Conception.....	23
1.	Modélisation 2D .....	23
2.	Modélisation 3D .....	29
a.	Etude d'une cavité parallélépipède.....	29
b.	Etude d'une cavité cylindrique .....	35
3.	Modèle simplifié .....	39
V.	Fabrication .....	43
VI.	Protocole expérimental .....	46
1.	Etude de l'isolation.....	47
2.	Caractérisation du récepteur .....	49
VII.	Conclusion .....	53
VIII.	Annexes .....	55



## Liste des tableaux

Tableau 1 : Classement de quelques récepteurs en fonction de différents fluides caloporteurs [11] .....	20
Tableau 2 : Classement de quelques récepteurs en fonction de leurs types [13] .....	22



## Liste des figures

Figure 1 : Principe général de la « Voie solaire thermodynamique » [2] .....	12
Figure 2 : Prototype de centrale solaire à capteurs cylindro-paraboliques [7] .....	14
Figure 3 : Prototype d'une centrale solaire à capteurs de Fresnel [7] .....	14
Figure 4 : Prototype d'une centrale solaire à tour [7] .....	15
Figure 5 : Prototype d'une centrale solaire à capteurs paraboliques [7] .....	16
Figure 6 : Prototype du récepteur plan de la centrale Solar Two [12] .....	18
Figure 7 : Prototype d'un récepteur à cavité à quatre ouvertures [12] .....	19
Figure 8 : De gauche à droite : tube échangeur, surface active et absorbeur [11] .....	19
Figure 9 : Vue d'ensemble de la cavité en 2D avec dimensions .....	24
Figure 10 : Génération de rayons à partir d'un point de la paroi verticale et de la paroi horizontale avec un écart d'angle en radians de $\pi/180$ .....	25
Figure 11 : Représentation graphique de l'efficacité en chaque point des parois horizontale (Gauche) et verticale (Droite) .....	26
Figure 12 : Élément infinitésimal de récepteur .....	27
Figure 13 : Profil de température considéré sur la paroi verticale de la cavité réceptrice .....	28
Figure 14 : Aperçu graphique de la cavité à étudier .....	29
Figure 15 : Aperçu graphique de la génération de rayons à partir d'un point de la façade du fond de la cavité .....	30
Figure 16 : Observation d'un rayon généré sur la façade interne suivant les plans yz (*), zx (**) et xy (***) ..	31
Figure 17 : Aperçu graphique de l'évolution de l'efficacité sur la façade de fond d'une cavité parallélépipédique .....	32
Figure 18 : Aperçu graphique de la génération de rayons à partir d'un point de la façade latérale de la cavité .....	33
Figure 19 : Observation d'un rayon généré sur la façade latérale suivant les plans (zx), (yz) et (xy) .....	33
Figure 20 : Aperçu graphique de l'évolution de l'efficacité sur une façade latérale de la cavité parallélépipédique .....	35
Figure 21 : Aperçu de la cavité cylindrique avec les différents repères d'études .....	36
Figure 22 : Observation de la cavité cylindrique suivant le plan (yz) .....	37
Figure 23 : Aperçu graphique de l'évolution de l'efficacité sur la paroi verticale .....	37
Figure 24 : Observation de la base supérieure de la cavité cylindrique suivant le plan (xy) .....	38
Figure 25 : Vue globale d'un cylindre avec les différentes grandeurs entrant dans le calcul du Facteur de forme .....	39
Figure 26 : Représentation graphique des échanges radiatifs et convectifs sur un élément de paroi du récepteur .....	40
Figure 27 : Courbes d'évolution des températures de fluide et de paroi (Gauche) et Absorption du flux en fonction de la hauteur du récepteur .....	42
Figure 28 : Aperçu de l'absorbeur (gauche) et du récepteur (droite) .....	42
Figure 29 : Aplatissage de la ferraille (Gauche) et constitution du tambour (Droite) .....	43
Figure 30 : Soudure des différents tubes (Gauche) pour constituer le corps de chauffe (Droite) .....	43
Figure 31 : Présentation de la cavité en tôle et du thermoplongeur .....	44
Figure 32 : Schéma de principe pour la création d'un environnement radiatif à l'intérieur du récepteur .....	47
Figure 33 : Aperçu de la disposition des thermocouples sur l'extérieur du récepteur .....	49
Figure 34 : Localisation des différents éléments sur la vue en 2D du récepteur .....	52

## I. Introduction

### 1. Contexte

Les populations rurales d'Afrique Sub-Saharienne souffrent du manque d'accès aux services énergétiques (Taux d'électrification  $< 30\%$ ). La distance qui sépare les zones rurales des centres de distribution ainsi que le coût prohibitif des unités de production sont à l'origine de ce problème.

Si de nos jours, l'essor économique et la prospérité industrielle d'un état s'évaluent à sa consommation énergétique, il est dès lors impensable d'espérer un développement de la zone Subsaharienne sans assurer au préalable la couverture électrique permanente des différents milieux géographiques qui la constituent.

Les pays d'Afrique Subsaharienne ne manquent pourtant pas de ressources renouvelables, encore inexploitées. En particulier, la plupart bénéficie d'un gisement solaire exceptionnel (4 à 7 kWh/m<sup>2</sup>/jour), ce qui fait des systèmes solaires autonomes, une solution quasi incontournable à la production décentralisée d'électricité. Ces systèmes se déclinent sous deux formes : les centrales solaires photovoltaïques et les centrales solaires thermodynamiques.

Si les premières sont fiables et rapides à installer, elles peinent à compenser l'intermittence de l'irradiation solaire : le stockage est problématique et coûteux. Les centrales thermodynamiques présentent l'avantage d'un stockage thermique. Cependant, un facteur technico-économique essentiel ressort des études sur les centrales solaires à concentration : ces dernières ne sont rentables que pour des puissances élevées ce qui fait appel à des coûts d'investissements énormes pour nos états.

Pour répondre au problème d'approvisionnement électrique des populations défavorisées, Le LESEE entreprend le développement d'une centrale solaire à tour « mini-réseaux ». C'est dans ce cadre qu'a été initié le projet CSP4AFRICA dans lequel les innovations prévues au niveau du champ solaire et de la turbine devraient conduire à une réduction importante des coûts d'investissements.

Outre ces innovations, le projet comporte de nombreux défis notamment la conception d'un récepteur solaire adapté à ce type de centrale : c'est le sujet de ce mémoire.

La première partie de ce travail consistera en une étude bibliographique générale sur les centrales solaires puis détaillée sur les récepteurs pour centrales à tour. Cette bibliographie nous permettra de réaliser les premiers choix généraux du récepteur que nous allons concevoir. Nous effectuerons ensuite plusieurs modélisations afin d'étudier celui-ci de manière approfondie. Ces modélisations



vont conduire à l'estimation du comportement du récepteur et par suite à son dimensionnement. L'ensemble de ces étapes mettra un terme à la phase de conception du projet. Après la fabrication du récepteur par un entrepreneur local, les premiers tests au sol seront réalisés dans le cadre de ce stage. L'objectif est double : d'une part, s'assurer de l'efficacité de l'isolation, et d'autre part, caractériser le récepteur.

## 2. Présentation du lieu du stage

Le présent stage s'est déroulé dans le Laboratoire Energie Solaire et Economie d'Energie (LESEE). Le LESEE est l'un des laboratoires du Centre Commun de Recherche Energie et Habitat Durables (CCREHD) de 2iE.

Le LESEE a pour objectif de contribuer à l'innovation industrielle dans le domaine du solaire pour une production et une utilisation rationnelle et durable d'énergies électrique, mécanique et thermique en Afrique.

Le LESEE est dirigé par Dr. Yao AZOUMAH. Son équipe actuelle se compose de 03 enseignants-chercheurs permanents, 01 post Doctorant, 04 Doctorants, 11 Ingénieurs de recherche, 01 Assistant d'enseignement et de recherche, 01 Technicienne de labo, 01 assistante administrative et 03 Professeurs associés.

Le LESEE exerce ses activités à travers deux axes de recherche :

### >Axe 1 : *Conversion de l'énergie solaire sous environnements chaud et rude*

Cet axe se décompose en trois sous axes de recherche :

- Systèmes solaires photovoltaïques en milieu hostile
- Centrales solaires à concentration (CSP) pour mini réseaux
- Habitats solaires

### >Axe 2 : *Gestion optimale d'énergie durable : production, transport, distribution*

Cet axe se décompose en deux sous-axes de recherche :

- L'hybridation des systèmes énergétiques
- L'interaction sources – réseaux intelligents - utilisateurs

## II. Synthèse bibliographique

### 1. Centrales solaires à concentration

Les Centrales Solaires à Concentration ou Concentrating Solar Plant (CSP) désignent les centrales de production d'électricité qui concentrent le rayonnement solaire et le convertissent en chaleur à haute température qu'elles utilisent dans un cycle thermodynamique. Autrement dit, les CSP utilisent des miroirs pour réfléchir et concentrer les rayons solaires sur des « récepteurs » qui convertissent l'énergie solaire concentrée en chaleur. Cette énergie thermique peut ensuite être utilisée pour produire de l'électricité via une turbine entraînant un générateur [1].

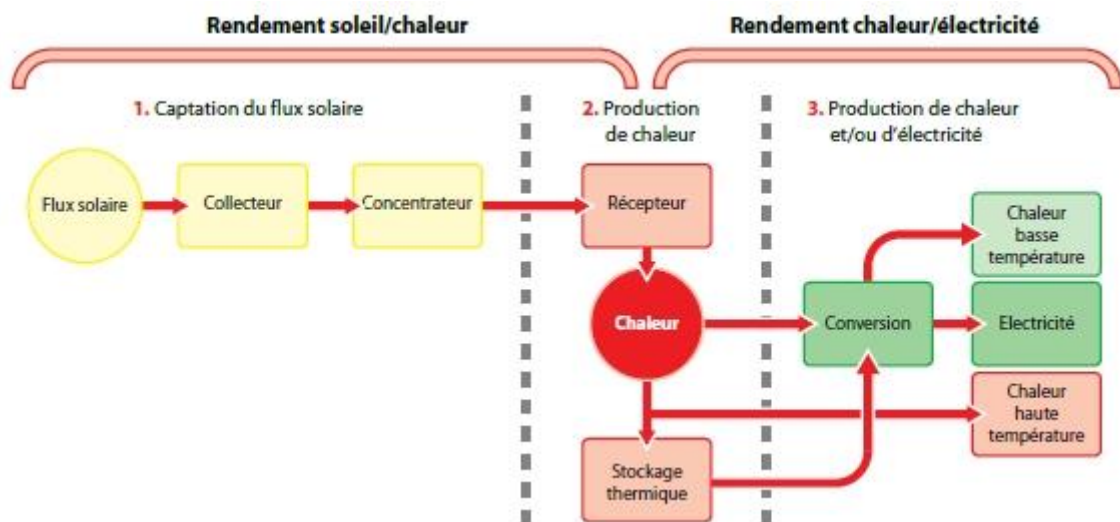


Figure 1 : Principe général de la « Voie solaire thermodynamique » [2]

Ce processus de production permet de distinguer les différentes parties d'une CSP ; on peut ainsi citer comme composantes de base [3], [4]:

- **Le champ solaire** qui représente l'équivalent de l'alimentation en combustible pour les centrales thermiques. Il est essentiellement composé de concentrateurs (miroirs réfléchissants) qui captent le rayonnement solaire direct et le concentrent sur le récepteur qui transmet la puissance captée à un fluide caloporteur sous forme de chaleur.
- **Le réseau de conduites** qui assure le transfert du fluide caloporteur chauffé du champ solaire vers l'unité de production d'électricité et son retour après refroidissement vers le champ solaire.

- **L'Unité de puissance** ou encore Unité de production d'électricité contient les pompes, les échangeurs de chaleur, le chauffage de la turbine, la turbine, le générateur, la tour de refroidissement et l'unité de déminéralisation, requise pour éviter la formation de cristaux de sels très abrasifs dans la turbine.

Le rayonnement solaire étant la ressource fondamentale pour la mise en place d'une CSP, ces dernières s'implantent dans des zones recevant un ensoleillement direct minimal de 1800 kWh/m<sup>2</sup>/an pour favoriser la rentabilité de la centrale [3].

#### Les différentes technologies de centrales solaires à concentration

Il existe plusieurs types de centrales solaires à concentration qui permettent de caractériser les filières de CSP. Cette différence apparaît essentiellement au niveau du champ solaire de par la géométrie de ses miroirs réflecteurs; on distingue ainsi les miroirs cylindro-paraboliques, les miroirs linéaires de Fresnel, les miroirs plans (héliostats) et les miroirs paraboliques donnant respectivement leurs noms aux différentes filières de CSP à savoir [5] [6]:

- Les centrales solaires à capteurs cylindro-paraboliques
- Les centrales solaires à miroirs de Fresnel
- Les centrales solaires à tour
- Les centrales solaires à capteurs paraboliques avec moteur Dish / Stirling

##### a. Centrales solaires à capteurs cylindro-paraboliques

Le champ solaire de cette centrale se compose de rangées parallèles de longs miroirs cylindro-paraboliques reposant sur des supports mécaniques à guidage automatique (mécanisme de poursuite) dont le rôle est d'adapter l'inclinaison du capteur de manière à ce que l'irradiation solaire incidente soit toujours perpendiculaire à l'ouverture de la parabole : ainsi, elle est réfléchi au foyer de la parabole et concentrée sur le récepteur solaire dans lequel circule le fluide caloporteur [6].

Les rayons solaires se réfléchissent sur les miroirs cylindro-paraboliques pour se concentrer sur le collecteur axé sur la focale. L'énergie thermique reçue au collecteur est absorbée par un tuyau métallique à l'intérieur d'un tube en verre sous vide. Le fluide caloporteur qui circule à l'intérieur du tuyau est chauffé à une température d'environ 400°C ; il est ensuite pompé à travers des échangeurs conventionnels afin de produire de la vapeur surchauffée qui fait fonctionner une turbine / générateur électrique [6].

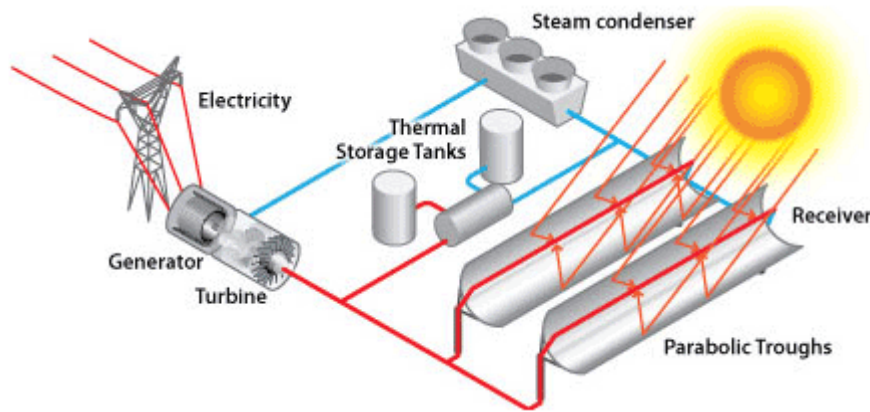


Figure 2 : Prototype de centrale solaire à capteurs cylindro-paraboliques [7]

#### b. Centrales solaires à capteurs linéaires de Fresnel

Dans le but de diminuer le coût qu'entraîne la mise en forme parabolique du verre, l'idée est donc d'approximer ladite forme parabolique (des miroirs cylindro-paraboliques) par une succession de miroirs plans [6], c'est la raison pour laquelle ces CSP sont considérés comme des variantes des CSP à capteurs cylindro-paraboliques.

Chacun de ces miroirs peut pivoter en se plaçant à une position angulaire intermédiaire entre le soleil et le récepteur pour rediriger et concentrer en permanence les rayons solaires vers un tube absorbeur [8] fixe ou un ensemble de tubes récepteurs fixes [9]. Les miroirs de Fresnel constituent ainsi un premier étage de réflecteurs. Le second étage de réflecteurs (placé au-dessus du récepteur) permet quant à lui de rediriger le rayonnement vers le tube récepteur pour lequel il joue aussi le rôle d'isolant. Les miroirs pivotent et réfléchissent les rayons solaires sur le(s) tube(s) récepteur(s). Le fluide caloporteur y circulant peut-être vaporisé puis surchauffé jusqu'à 500°C. La vapeur alors produite actionne une turbine qui produit de l'électricité. Le cycle thermodynamique est généralement direct, ce qui permet d'éviter les échangeurs de chaleur [9].

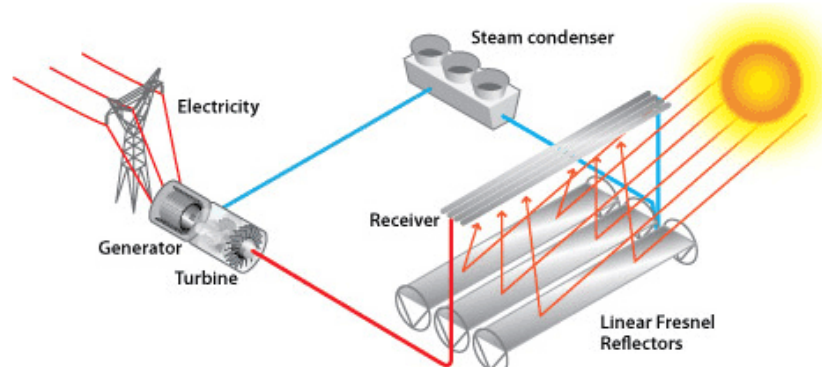


Figure 3 : Prototype d'une centrale solaire à capteurs de Fresnel [7]

### c. Les centrales solaires à tour

Leurs champs solaires sont constitués de plusieurs miroirs plans appelés « héliostats ». Chacun de ces héliostats est orientable puisque monté sur un système de guidage automatique qui lui permet en permanence de réfléchir les rayons solaires sur un récepteur situé au sommet d'une tour. On obtient ainsi une concentration « ponctuelle » au niveau dudit récepteur.

Les rayons réfléchis sur le « récepteur solaire » au sommet de la tour permettent d'atteindre un niveau de concentration élevé et une température de l'ordre de 600°C à 1000°C [9].

Dans ce type de centrale, la chaleur peut être ramenée au niveau du sol via l'utilisation d'un fluide caloporteur qui passe dans le récepteur et s'échauffe. Il servira à produire de la vapeur qui à son tour sera utilisée pour produire de l'électricité.

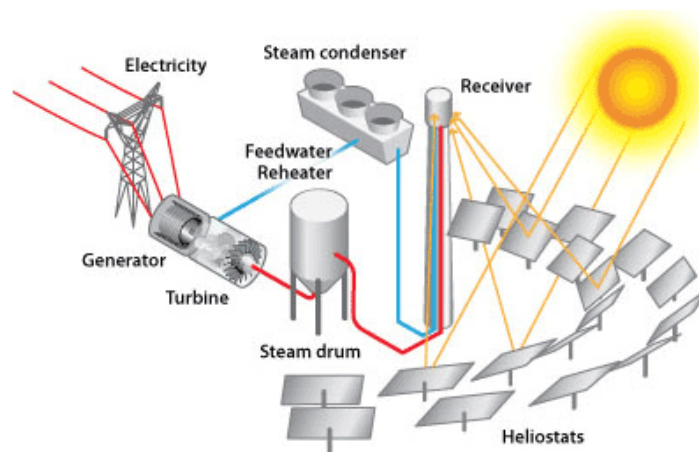


Figure 4 : Prototype d'une centrale solaire à tour [7]

### d. Les centrales à capteurs paraboliques

Les centrales solaires à capteurs paraboliques fonctionnent d'une manière autonome. Reposant aussi sur un système de guidage, ils s'orientent automatiquement afin de réfléchir et concentrer les rayons solaires sur un point appelé foyer : ce foyer est le lieu de positionnement du récepteur de cette CSP. Le récepteur est le plus souvent une enceinte fermée contenant du gaz qui monte en température sous l'effet de la concentration. Cela entraîne un moteur Stirling qui convertit l'énergie solaire thermique en énergie mécanique puis en électricité. Ce type de CSP peut être installé dans des endroits isolés non raccordés aux réseaux électriques. Il n'admet cependant pas de stockage [9].



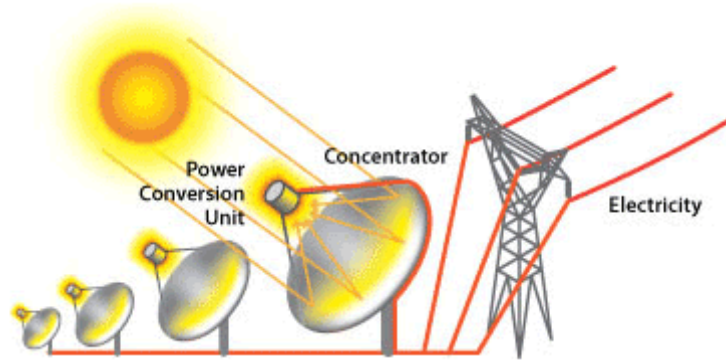


Figure 5 : Prototype d'une centrale solaire à capteurs paraboliques [7]

La centrale CSP4AFRICA étant une CSP à tour, première en Afrique Sub-saharienne où les conditions climatiques sont rudes, l'étude technique de ses différentes composantes s'avère nécessaire. Ainsi, comme mentionné au début de ce mémoire, nous nous intéressons principalement au récepteur solaire de ces types de CSP dont l'étude bibliographique est détaillée dans les paragraphes suivants.

## 2. Récepteurs solaires pour Centrales à tour

Elément de la boucle thermique, le récepteur est situé au-dessus de la tour elle-même appartenant au bloc de captage [3]. Siège de la conversion du rayonnement solaire concentré en chaleur à haute température, le récepteur est le composant qui supporte les températures les plus élevées dans une centrale solaire : il est donc soumis à des régimes transitoires violents liés aux variations d'ensoleillement [11]. Ce composant joue en quelque sorte un rôle d'échangeur de chaleur puisqu'il transmet la majeure partie de l'énergie calorifique qu'il reçoit au fluide caloporteur.

Il est essentiellement constitué d'un absorbeur, d'un isolant thermique, d'un système de distribution et de collecte du fluide et optionnellement d'un concentrateur secondaire, d'une cavité réceptrice et d'une vitre de protection / isolation / pressurisation [5].

Les récepteurs solaires sont conçus pour avoir une longue durée de vie ainsi qu'une fiabilité élevée ; un rendement de conversion élevé, une bonne tenue à la fatigue thermique et une bonne résistance aux hautes températures sont les objectifs techniques visés.

Toutefois, bon nombre de problèmes sont rencontrés lors de la phase de conception ; ces derniers sont d'ordre [11] :

- Scientifiques

- Comportement et propriétés des matériaux
- Etude des écoulements fluides turbulents
- Intensification des transferts thermiques
- Technologiques
  - Sélection des matériaux
  - Mise en forme desdits matériaux
  - Assemblage des matériaux
- Economiques
  - Maîtrise des coûts de fabrication
  - Maîtrise des coûts de maintenance

Il existe plusieurs types de récepteurs solaires pour centrales à tour ; ils peuvent être classés suivant la nature de l'absorbeur, le type de fluide utilisé, le type d'élément d'absorbeur, la configuration du récepteur, la configuration de l'absorbeur et la pression du fluide de travail [5]. Dans la suite de ce document, la classification portant sur la configuration du récepteur sera adoptée ; on distinguera ainsi deux types de récepteurs : les récepteurs « externes plans » et à cavité [5], [11].

a. Les récepteurs externes plans

Ils sont constitués de panneaux contenant plusieurs petits tubes verticaux soudés les uns aux autres pour approximer la forme cylindrique. Un système de distribution fournit le fluide caloporteur au récepteur cylindre par sa base inférieure et recueille le fluide chauffé à la base supérieure.

Les récepteurs externes ont un rapport hauteur sur diamètre variant de 1 à 2. La surface du récepteur est maintenue au minimum possible pour réduire les pertes thermiques. **La limite inférieure est déterminée par la température maximale de fonctionnement des tubes et par conséquent la capacité du fluide caloporteur à extraire de la chaleur.**

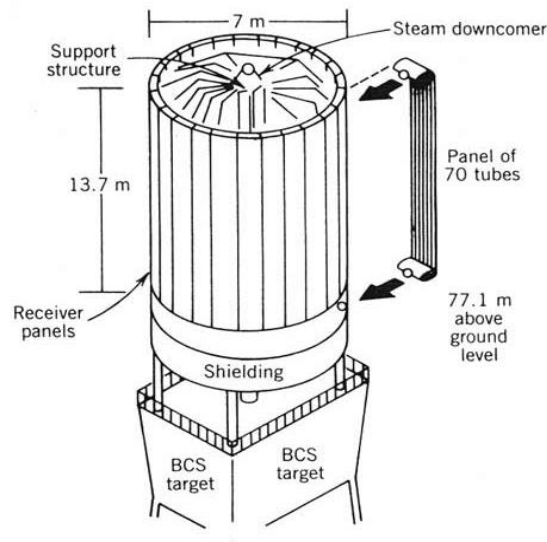


Figure 6 : Prototype du récepteur plan de la centrale Solar Two [12]

Par exemple, la conception d'un récepteur utilisant le sodium liquide comme fluide caloporteur pour une puissance de sortie de 380 MW nécessite un récepteur de 15m de haut et 13m de diamètre. Si le fluide caloporteur était du sel de nitrate fondu plutôt que du sodium liquide, une surface d'environ le double du précédent serait requise pour obtenir la même puissance de sortie ainsi que la même température ceci à cause des basses propriétés de transfert de ce fluide [12].

#### b. Les récepteurs à cavité

Dans le souci de réduire les pertes thermiques au niveau du récepteur, certaines configurations recommandent de placer la surface absorbant le flux solaire à l'intérieur d'une cavité isolée, ainsi les pertes convectives venant de l'absorbeur sont réduites. Le flux provenant du champ d'héliostats est réfléchi à travers une ouverture sur les surfaces absorbantes représentant les parois de la cavité. Ces types de récepteurs ont une surface d'ouverture allant du tiers à la moitié de la surface absorbante interne. La taille de l'ouverture est réduite pour diminuer les pertes convectives et radiatives sans pour autant trop empêcher le flux d'atteindre le récepteur. Les dimensions de l'ouverture du récepteur sont approximativement identiques à celles de l'image du soleil réfléchi par l'héliostat le plus éloigné avec une perte de 1 à 4% [12].

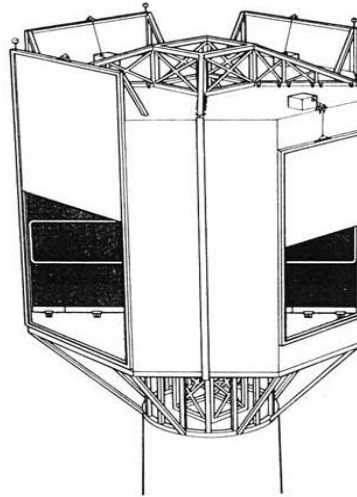


Figure 7 : Prototype d'un récepteur à cavité à quatre ouvertures [12]

Les récepteurs à cavité se distinguent fondamentalement par leurs absorbeurs ; ces derniers sont considérés comme des échangeurs de chaleur puisque leur rôle principal est d'absorber le flux solaire concentré afin de chauffer le fluide caloporteur. Ils se classent en deux catégories : tubulaires et volumétriques. Ces dénominations font référence aux formes géométriques des lieux où circulent les fluides caloporteurs.

#### Absorbeurs tubulaires (surfacciques)

Un absorbeur tubulaire est une association de tubes échangeurs de chaleur dans lesquels circule le fluide caloporteur. Cette combinaison de tubes permet d'obtenir une surface « active » d'échange thermique encastrée d'une part dans un distributeur (qui lui fournit le fluide) et de l'autre dans un collecteur qui reçoit le fluide chauffé ou la vapeur qui en est issue.

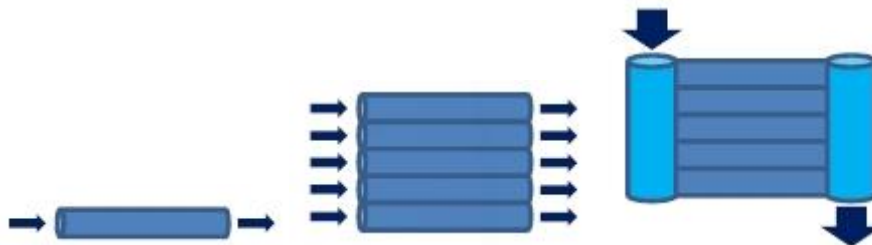
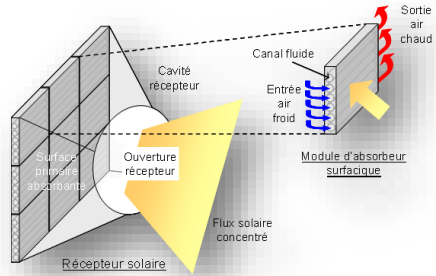
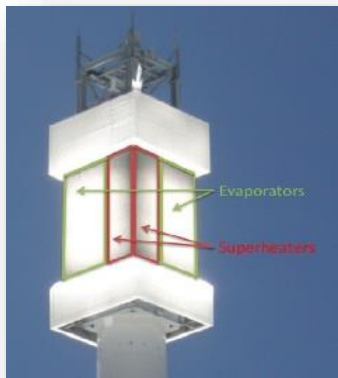



Figure 8 : De gauche à droite : tube échangeur, surface active et absorbeur [11]

Il existe plusieurs types d'absorbeurs pour récepteurs solaires ; ils peuvent se différencier suivant plusieurs paramètres toutefois la classification ci-dessous se fera suivant le fluide caloporteur utilisé.

Tableau 1 : Classement de quelques récepteurs en fonction de différents fluides caloporteurs [11]

Fluide caloporteur	Nom de la centrale associée	Description	Représentation
Air pressurisé	Pégase	Concept original de récepteur métallique à cavité et à paroi absorbante plane modulaire et étagée.	
Eau / Vapeur	eSolar	Récepteur à vapeur surchauffée avec une pression de 55 bars et une température de fonctionnement de 420°C. Adapté à la configuration d'un champ circulaire.	
Sels fondus	Thémis	Récepteur à tubes en Acier inox Pyromark. Pertes thermiques faibles. Température allant à 450°C. Adapté à la configuration d'un champ semi-circulaire.	

## Absorbeurs volumiques

Comme l'indique leur nom, ces absorbeurs représentent en fait des volumes dans lesquels se déroulent les échanges thermiques. L'intérêt pour ces types de récepteurs vient d'une limitation en température des récepteurs tubulaires. Ces récepteurs sont soit métalliques (fils, canaux), céramiques (canaux, mousse, tiges, fibres) ou à particules (entraînées, lits) [13].

Les récepteurs volumiques existent en plusieurs types ; on peut distinguer :

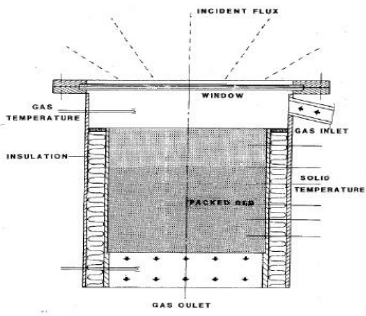
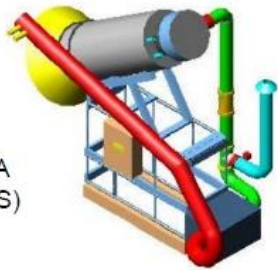
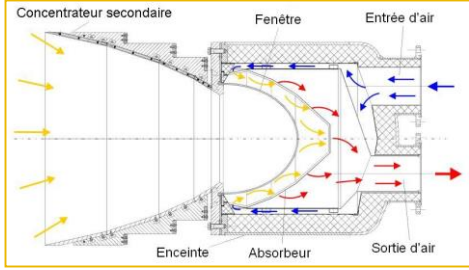
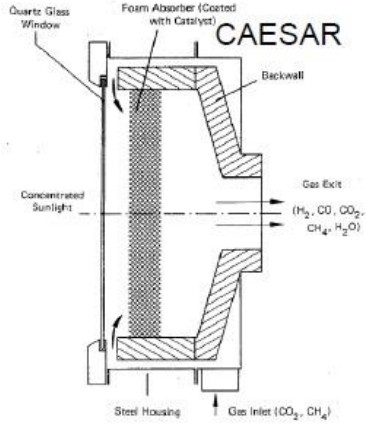
- Récepteurs sélectifs
- Récepteurs à air atmosphérique
- Récepteurs à air pressurisé
- Récepteurs à particules
- Récepteurs-réacteurs

Ces récepteurs présentent des avantages de taille mais aussi des inconvénients majeurs. On peut lister :

- Avantages
  - Température élevée
  - Modulaire (Pas de partie mobile)
  - Peu de contraintes sur les absorbeurs
  - Grandes surface spécifiques pour les réactions (Récepteurs-réacteurs)
- Inconvénients
  - Présence de hublot
  - Déformation de la structure porteuse
  - Dégradation du matériau (Récepteurs-réacteurs)
  - Mauvaise distribution du débit d'air (point chaud, flux incident variable...)
  - Coût d'investissement élevé
  - Réalisation du matériau (Dépôt, dopage...) (Récepteurs-réacteurs)

Ci-dessous, un tableau donnant quelques exemples de récepteurs volumiques :

Tableau 2 : Classement de quelques récepteurs en fonction de leurs types [13]

Type	Description, Généralités	Représentation
Récepteur sélectif	Récepteur sélectif bi-couche : nid d'abeille en Si et lit de particules SiC 72% d'efficacité à 800°C	
Récepteur à air atmosphérique (SIREC)	Fils tressés, 48% d'efficacité à 710°C Régulation et Contrôle du débit d'air rendus difficiles par de forts gradients dans l'absorbeur (600°C – 760°C)	 Sirec à PSA (SSPS-CRS)
Récepteur à air pressurisé (REFOS)	Maillage de fils (Inconel 600) Hublot elliptique Conception d'un second concentrateur performant et léger, mauvaise isolation du récepteur Dégradation du hublot 67% d'efficacité à 800°C	
Récepteurs- réacteurs (CAESAR)	Mousse $Al_2O_3$ avec un dépôt de Rhodium (Catalyseur) Reformage de $CH_4$ à base de $CO_2$ Rendement de 66% pour le $CH_4$ Température supérieure à 1200°C Dislocation de l'absorbeur Désactivation du catalyseur	

Au vu de cette revue bibliographique et compte tenu des éléments obtenus sur les technologies des récepteurs solaires des centrales à tour, nous avons fait le choix d'un récepteur solaire à cavité avec absorbeur tubulaire. Ils présentent l'avantage d'avoir une absorptivité excellente et de minimiser les pertes thermiques.

### III. Hypothèses de travail

Lors du précédent stage, les géométries cubiques, hémisphériques et sphériques [14] ont été comparées du point de vue des pertes convectives. Les études ont été menées pour trois différents ratios d'ouverture ( $R = 1$ ,  $R=0,5$  et  $R=0,25$ ) et cinq inclinaisons distinctes ( $0^\circ$ ,  $30^\circ$ ,  $45^\circ$ ,  $60^\circ$  et  $90^\circ$ ). Elles ont permis de faire ressortir un ordre de grandeur de 1,2% pour les pertes convectives du récepteur.

Ces pertes étant très faibles, il est essentiel d'évaluer les pertes radiatives qui représentent les pertes majoritaires. Cette évaluation sera réalisée en utilisant différentes géométries afin de comparer leur efficacité.

Nous allons élaborer un modèle pour simuler le comportement global du récepteur. Ce modèle intégrera les pertes de charges, les transferts convectifs avec le fluide caloporteur et les transferts radiatifs. Pour procéder par étape, nous étudierons dans un premier temps uniquement les pertes radiatives et nous comparerons les géométries cubique et cylindrique.

Comme nous ne disposons pas de facteur de forme pour la géométrie cubique, nous allons calculer son équivalent numériquement. Pour cela, nous utiliserons la technique du 'Lancer de rayons' [15] qui se prête très bien aux simulations optiques.

### IV. Conception

#### 1. Modélisation 2D



Le premier modèle élaboré est en 2 dimensions. Les mises en équation y sont simplifiées, ce qui nous permettra de vérifier la pertinence des premiers résultats avant de passer à une modélisation plus avancée (3 dimensions).

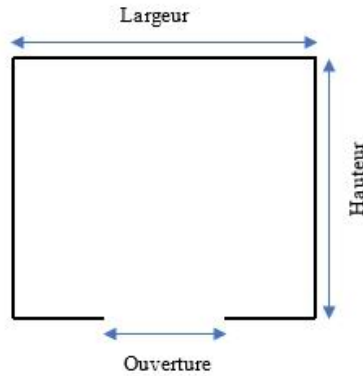


Figure 9 : Vue d'ensemble de la cavité en 2D avec dimensions

Nous considérons une cavité rectangulaire dont les différentes dimensions à savoir la largeur (l), la hauteur (h) et l'ouverture (o) sont connues (Figure 9). On suppose que le rayonnement émis par un corps est isotrope.

Calculer l'équivalent numérique du facteur de forme en un point déterminé peut-être réalisé par une modélisation simple. On génère en ce point un ensemble de rayons répartis de manière isotrope. Il suffit de comptabiliser le nombre de rayons sortants par l'ouverture et le nombre de rayons générés. L'absorptivité apparente est le rapport du nombre de rayons qui ne sont pas sortis au nombre de rayons générés au total soit :

$$A = \frac{(n_T - n_S)}{n_T} \text{ avec } \begin{cases} n_T = \text{nombre total de rayons générés} \\ n_S = \text{nombre de rayons sortis} \end{cases} \quad (1)$$

Calculer l'absorptivité apparente revient donc à paramétrer l'équation d'une droite avec un angle (que l'on nomme  $\alpha$ ), à faire varier cet angle par pas égaux et à comptabiliser les rayons sortants en fonction de l'intersection de cette droite avec le plan de l'ouverture. En plaçant un repère dont l'origine est au centre de l'ouverture et en appelant  $x_i, y_i$  les coordonnées de l'intersection, le critère qui permet de déterminer si le rayon (représenté par la droite) est sortant s'exprime simplement par :

$-\frac{o}{2} \leq x_i \leq \frac{o}{2}$  avec  $x_i$  la coordonnée de l'intersection entre la droite (représentant le rayon) et le plan de l'ouverture.

L'équation d'une droite qui prend son origine sur la paroi verticale et faisant un angle  $\alpha$  avec celle-ci est :  $y = (x - \frac{l}{2}) + y_p \times \tan \alpha$  (2)

Avec  $y_p$  la hauteur du point origine de la droite.

L'équation d'une droite qui prend son origine sur la paroi horizontale et faisant un angle  $\alpha$  avec celle-ci est :  $y = (x - x_p) \tan \alpha - h$  (3)

Avec  $x_p$  la coordonnée  $x$  du point origine de la droite.

La coordonnée  $x_i$  de l'intersection est dans chaque cas la solution de l'équation  $y = 0$  soit :

$x_i = \frac{l}{2} - y_p \tan \alpha$  : si la droite prend son origine sur la paroi verticale

$x_i = x_p + \frac{h}{\tan \alpha}$  : si la droite prend son origine sur la paroi horizontale

Grâce aux interfaces graphiques aisément manipulables sous python, nous pouvons procéder à une vérification du fonctionnement du modèle par représentation graphique.

Il suffit de tracer les droites générées de 2 couleurs différentes : le vert a été choisi pour les rayons sortants et le rouge pour les autres. On voit sur la Figure 10 que dans les 2 cas (origine sur la paroi horizontale ou origine sur la paroi verticale) **seuls** les rayons passants par l'ouverture sont verts, et que **tous** les rayons ne passant pas par l'ouverture sont rouges. La mise en équation associée peut donc être considérée comme valide.

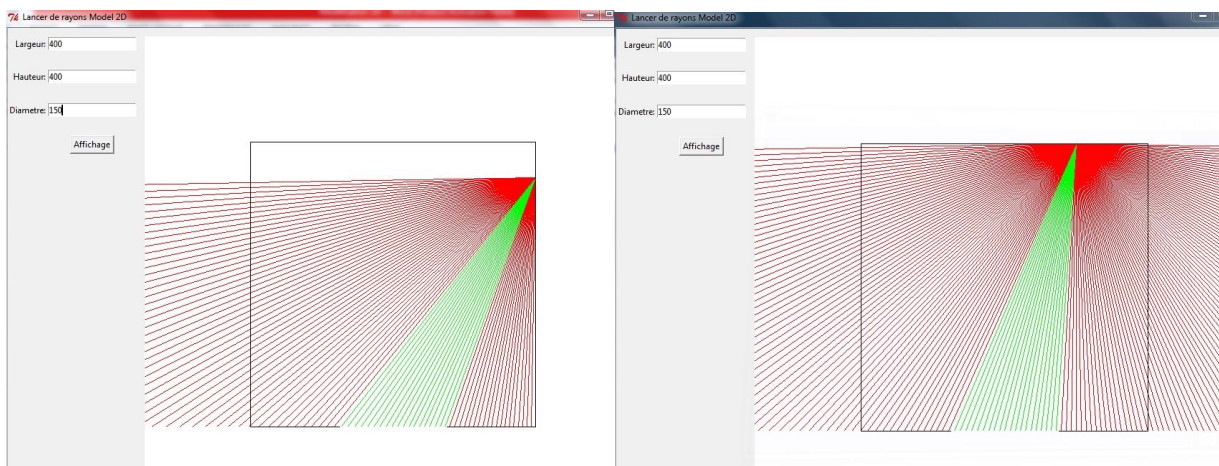


Figure 10 : Génération de rayons à partir d'un point de la paroi verticale et de la paroi horizontale avec un écart d'angle en radians de  $\pi/180$

### Nombre de rayons générés :

Le fait que le nombre de rayons générés soit fini engendre toujours une erreur dans l'estimation de l'absorptivité apparente. Plus le nombre de rayons augmente plus l'erreur diminue mais plus les

temps de calculs s'allongent. Nous avons donc augmenté le nombre de rayons générés jusqu'à obtenir une convergence des résultats. Par exemple on remarque que de 1000 à 10000 rayons générés par point, la valeur de l'absorptivité apparente varie de moins de 0,0001 : il est donc inutile de générer plus de 1000 rayons par point (dans le cas du présent modèle).

### Evaluation de l'efficacité locale

Dans cette partie, nous allons définir l'efficacité en chaque point de la paroi qui correspond à 100 fois l'absorptivité apparente :  $E = A \times 100$ . Nous affecterons à chaque point une couleur qui sera choisie en fonction de la valeur locale de l'efficacité (une légende à droite du récepteur permet d'en prendre connaissance). Il est important de remarquer que l'axe de symétrie du récepteur nous dispense des calculs sur la deuxième paroi verticale.

Les résultats obtenus sont présentés sur la figure 11.

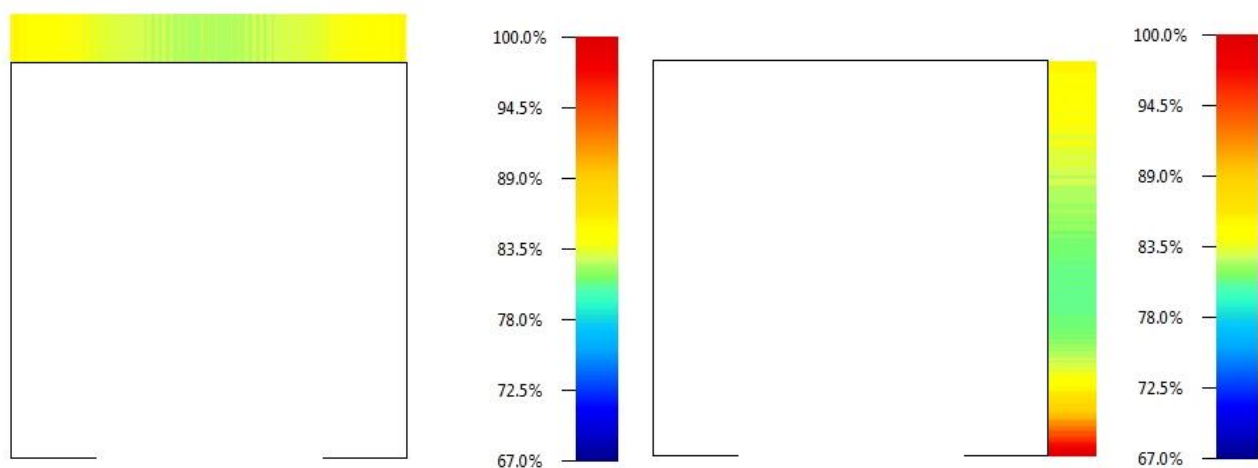


Figure 11 : Représentation graphique de l'efficacité en chaque point des parois horizontale (Gauche) et verticale (Droite)

Ces résultats vérifient un certain nombre d'assertions logiques.

#### Sur la paroi verticale :

- On constate la présence d'un point chaud au niveau du coin inférieur du récepteur : le rayonnement émis à partir de ce point ne peut effectivement pas s'échapper de la cavité puisqu'on y voit l'ouverture sous un angle nul.
- L'absorptivité apparente diminue ensuite graduellement à mesure que l'on s'éloigne du rebord inférieur

- Plus l'on se rapproche de la paroi supérieure et plus la distance à l'ouverture augmente. Il y a donc un point de basculement où l'angle de vue de l'ouverture commence à diminuer : à partir de ce point l'absorptivité apparente augmente indéfiniment

#### Sur la paroi horizontale :

- Le point situé sur l'axe de symétrie du récepteur est à la distance minimale de l'ouverture. En ce point l'absorptivité apparente est minimale.
- Plus l'on s'éloigne du centre et plus la distance à l'ouverture augmente, ce qui diminue l'angle de vue de celle-ci. L'absorptivité apparente augmente donc indéfiniment à mesure que l'on s'éloigne du centre

Pour caractériser correctement l'effet des pertes radiatives sur l'efficacité du récepteur il faut évaluer la puissance totale perdue par émissions radiatives. Nous désignerons cette puissance par  $P_r$  (puissance radiative perdue). Elle est la somme de toutes les puissances localement perdues le long de la paroi du récepteur ( $dP_r$ ). La figure 12 représente un élément infinitésimal du récepteur pour lequel on évalue la puissance radiative localement perdue.

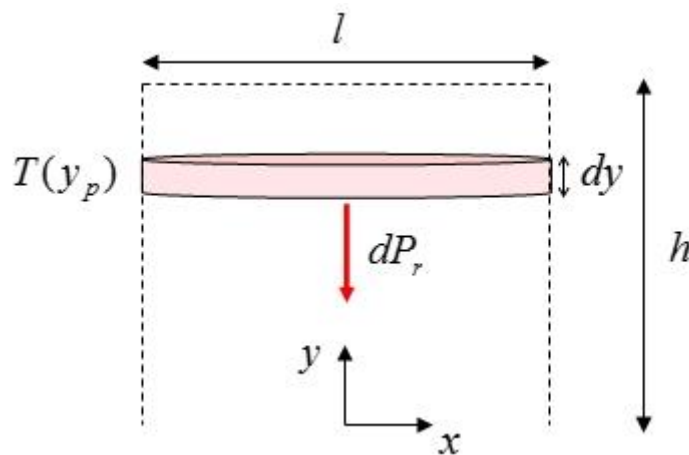


Figure 12 : Elément infinitésimal de récepteur

En utilisant la loi de Stefan-Boltzmann on peut écrire pour un élément de hauteur  $y_p$  :

$$dP_r = \varepsilon \times \sigma \times (1 - A) \times dS \times T(y_p)^4 = \varepsilon \times \sigma \times (1 - A) \times \pi \times l \times dy \times T(y_p)^4 \quad (4)$$

Il vient ensuite :

$$P_r = \int_0^h dP_r = \int_0^h \varepsilon \times \sigma \times (1 - A) \times \pi \times l \times T(y)^4 dy \quad (5)$$

Pour évaluer la puissance radiative perdue il nous faut connaître le profil de température le long de la paroi du récepteur.

Nous connaissons déjà les consignes de la boucle thermique du projet CSP4AFRICA : le fluide qui circule dans le récepteur est chauffé de 70°C à 250°C. Puisque ce fluide est directement en contact avec la paroi du récepteur sa température en est assez proche. Nous supposons que la variation de température est linéaire sur la paroi du récepteur (comme illustré sur la figure 13).

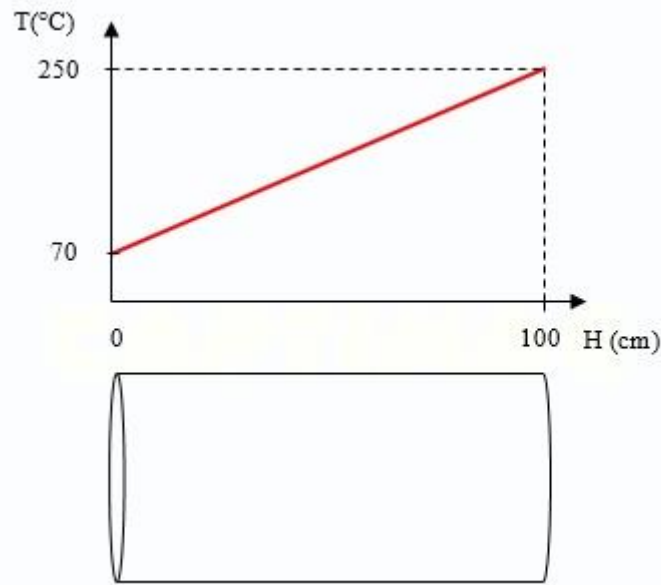


Figure 13 : Profil de température considéré sur la paroi verticale de la cavité réceptrice

De cette manière il est possible de déterminer la fonction  $T(y)$  et d'effectuer l'intégration ci-dessus de manière numérique.

L'absorptivité apparente de notre cavité est définie par rapport à la puissance radiative perdue ( $P_r$ ) et à la puissance incidente ( $P_i$ ) sur le récepteur par :

$$A = \frac{(P_i - P_r)}{P_i} \quad (6)$$

Par ailleurs, la puissance radiative perdue par mètre linéaire au niveau de l'élément infinitésimal s'exprime par :

$$dP_r = \frac{n_s}{n_t} \times \varepsilon \times \sigma \times dy \times T(y)^4 \quad (7)$$

La puissance radiative perdue par la cavité et par mètre linéaire est l'intégrale de la puissance

perdue par l'élément infinitésimal :  $P_r = \int_0^h dP_r$  (8)

## 2. Modélisation 3D

Pour l'étude des différentes modélisations qui suivent, nous utiliserons les coordonnées sphériques qui nous permettront d'obtenir la distribution angulaire souhaitée [16].

### a. Etude d'une cavité parallélépipède

Nous considérons une cavité parallélépipédique de hauteur (H), de longueur (L), de largeur (l) et de diamètre d'ouverture (D).

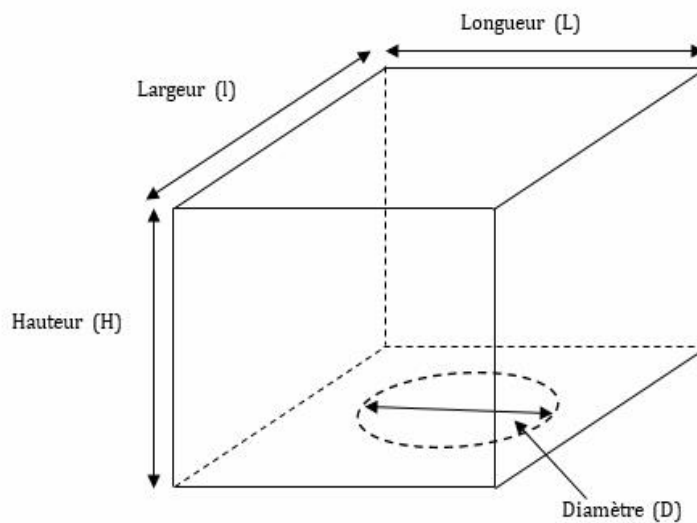


Figure 14 : Aperçu graphique de la cavité à étudier

Dans un souci d'uniformité au niveau de la répartition du flux incident sur les parois latérales, la longueur et la largeur sont maintenues identiques.

La démarche suivie lors de la modélisation en 2D est la même que celle de la modélisation 3D. Cependant, de nouvelles mises en équations sont nécessaires pour notre étude. L'existence de l'axe de symétrie de la cavité passant par le centre de l'ouverture entraîne une similitude entre les 4 faces latérales, ainsi donc notre étude se focalisera sur la façade du fond et une des façades latérales.

Pour ce faire, nous supposons que le rayonnement est isotrope, la température des parois uniforme et nous nous plaçons en coordonnées sphériques.

Les exemples qui sont illustrés dans la suite porte sur un parallélépipède de Hauteur (1m) et de largeur équivalente à la longueur (0,7m).

### Façade du fond

Un repère local ( $0', x', y', z'$ ) orienté conformément au repère fixe ( $0, x, y, z$ ) est placé sur la façade du fond de la cavité.

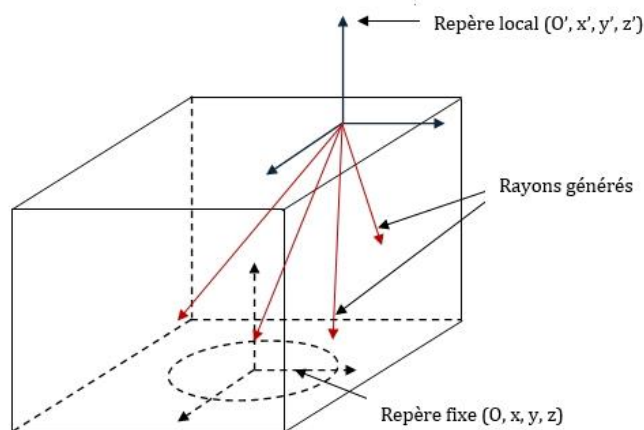


Figure 15 : Aperçu graphique de la génération de rayons à partir d'un point de la façade du fond de la cavité

Un rayon est généré à partir de  $O'$  et on considère un point  $M$  appartenant à ce rayon.

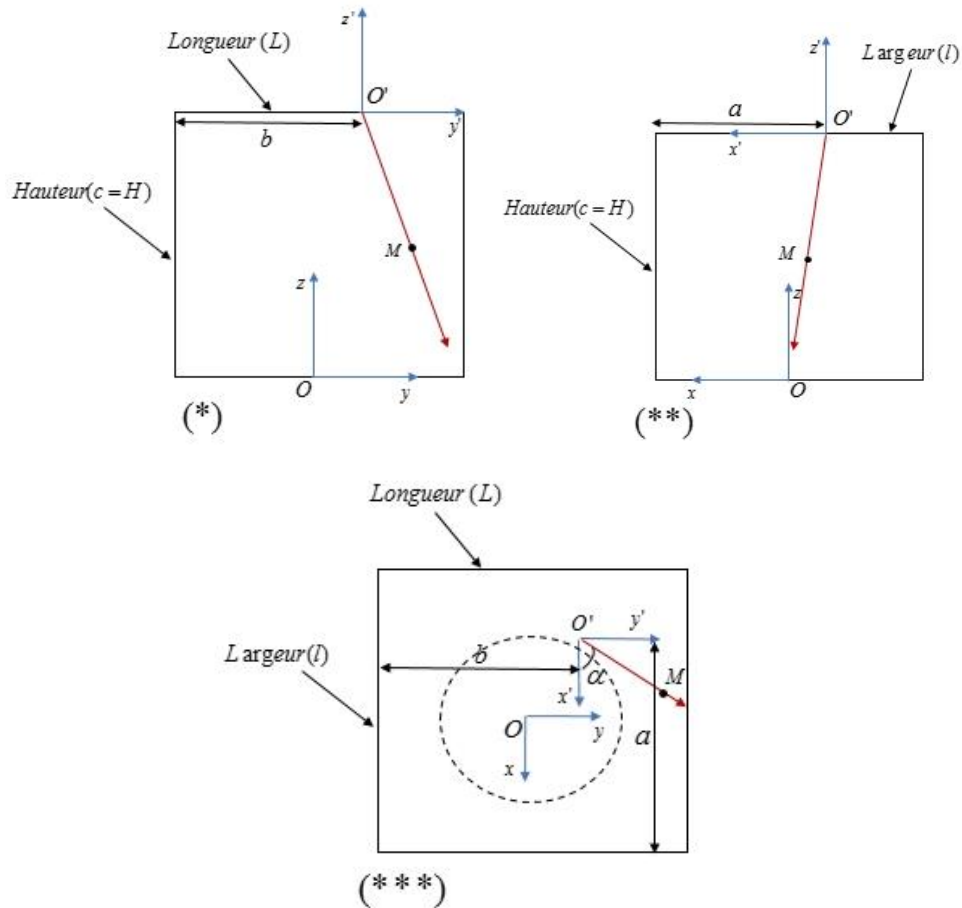


Figure 16 : Observation d'un rayon généré sur la façade interne suivant les plans yz (\*), zx (\*\*) et xy (\*\*\*)

Le rayon généré ci-dessus (Figure 16) est paramétré suivant les coordonnées sphériques. On obtient ainsi pour la mise en équation dans le repère  $(O, x', y', z')$  :

$$\begin{cases} X' = r \sin \theta \cos \varphi \\ Y' = r \sin \theta \sin \varphi \\ Z' = r \cos \theta \end{cases} \quad (9) \quad \text{Avec} \quad \begin{cases} r = d(O, M) \\ \theta \in \left[ \frac{\pi}{2}; \pi \right] \\ \varphi \in [0; 2\pi] \end{cases}$$

Soit  $(a, b, c)$  les coordonnées du repère local dans le repère fixe  $(O, x, y, z)$ , on obtient pour la mise en équation dans  $(O, x, y, z)$  :

$$\begin{cases} X = r \sin \theta \cos \varphi + \frac{l}{2} - a \\ Y = r \sin \theta \sin \varphi - \frac{L}{2} + b \\ Z = r \cos \theta + c \end{cases} \quad (10) \quad \text{Avec} \quad \begin{cases} l = \text{largeur} \\ L = \text{Longueur} \end{cases}$$



On fait varier les deux angles  $\alpha$  et  $\theta$  par pas réguliers. Les critères pour vérifier la sortie d'un rayon par l'ouverture sont :

$$\begin{cases} Z = 0 \Leftrightarrow r \cos \theta + h = 0 \Rightarrow r = \frac{-h}{\cos \theta} \\ \sqrt{X^2 + Y^2} \leq \frac{D}{2} \end{cases}$$

On peut caractériser tout rayon sortant par l'ouverture de la cavité par la condition :

$$\sqrt{X^2 + Y^2} \leq \frac{D}{2}, \begin{cases} X = -H \tan \theta \cos \varphi + \frac{l}{2} - a \\ Y = -H \tan \theta \sin \varphi - \frac{L}{2} + b \end{cases} \quad (11)$$

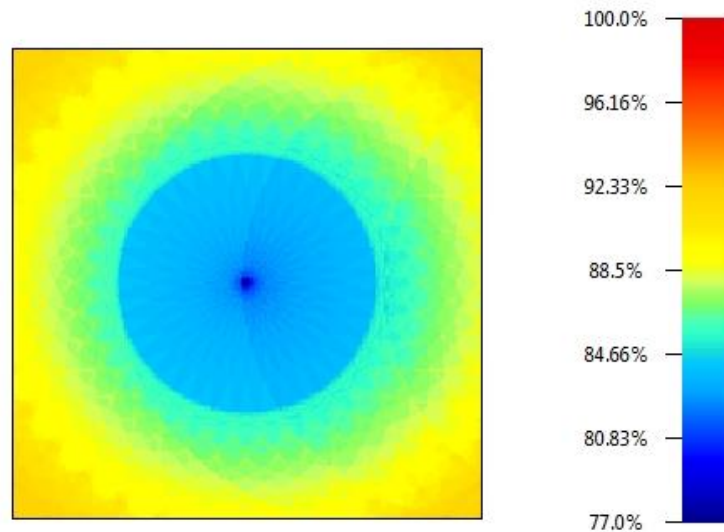


Figure 17 : Aperçu graphique de l'évolution de l'efficacité sur la façade de fond d'une cavité parallélépipédique

L'observation de ce graphique montre que plus on s'éloigne du centre géométrique de la façade, et plus l'efficacité augmente. Cela se traduit par la diminution de l'angle de vue de l'ouverture lorsqu'on s'éloigne du centre et il est facilement remarquable que les points de plus grande efficacité sont situés au niveau des quatre angles droits.

Par ailleurs, la distribution de l'efficacité est uniforme suivant toutes les directions du plan, ce qui montre la validité des résultats.

### Façade latérale

Le même repère local est cette fois ci-placé sur une façade latérale comme l'indique la figure ci-dessous.

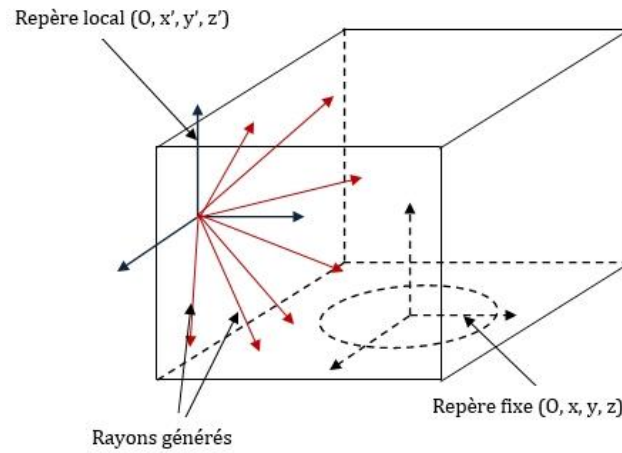


Figure 18 : Aperçu graphique de la génération de rayons à partir d'un point de la façade latérale de la cavité

Un rayon est généré à partir d'un point  $O'$  de la paroi latérale. Un point  $M$  est localisé sur ce rayon. La démarche est la même que celle suivie pour la façade du fond étudiée ci-dessus.

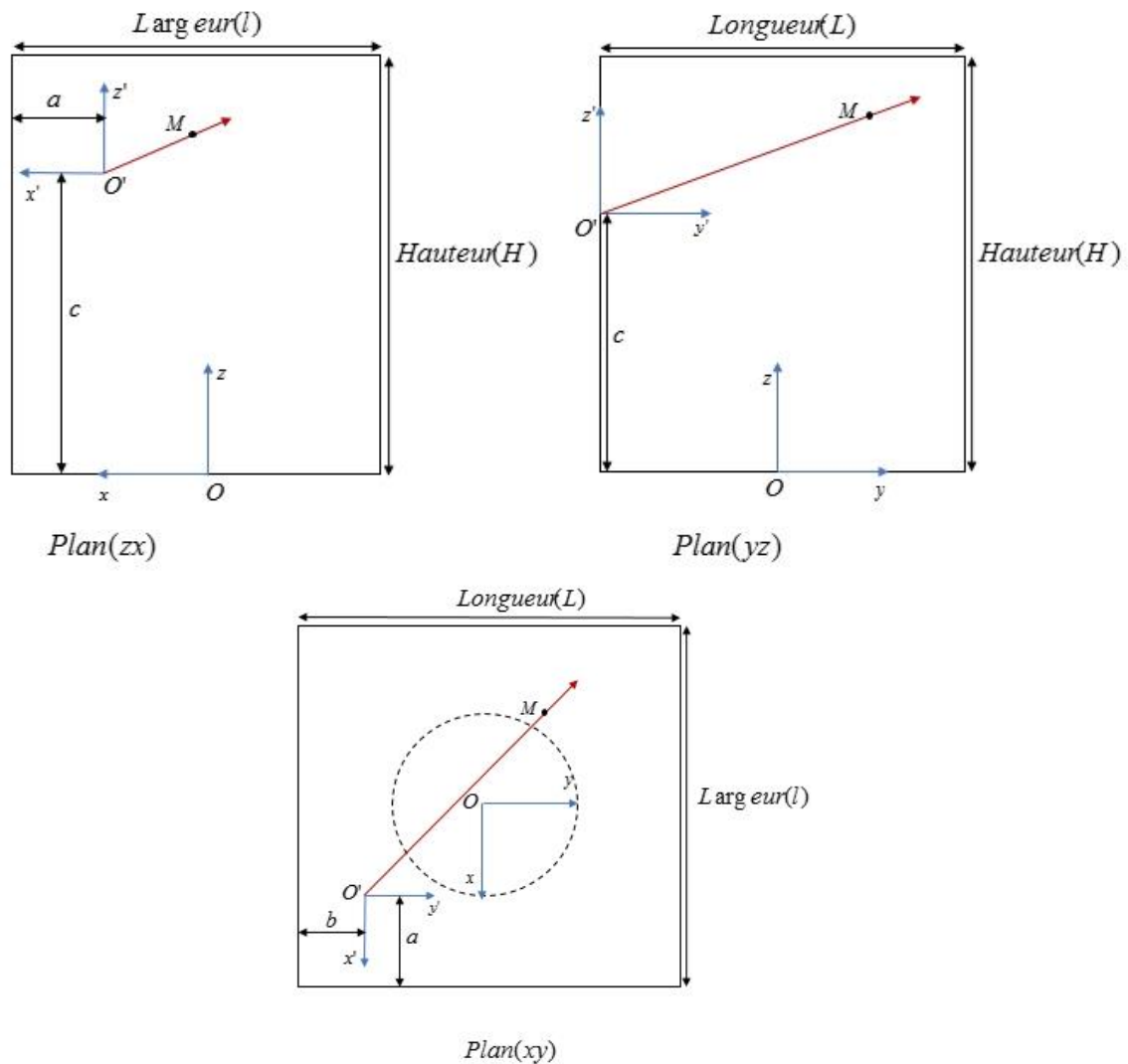


Figure 19 : Observation d'un rayon généré sur la façade latérale suivant les plans (zx), (yz) et (xy)

Les coordonnées du vecteur O'M conservent leurs expressions (identiques à celles définies pour la façade du fond) dans le repère local. Soit  $(a, b, c)$  les coordonnées de ce repère local dans le repère fixe  $(0, x, y, z)$ , on obtient pour la mise en équation dans  $(0, x, y, z)$  :

$$\left\{ \begin{array}{l} X = r \sin \theta \cos \varphi + \frac{l}{2} - a \\ Y = r \sin \theta \sin \varphi - \frac{L}{2} \\ Z = r \cos \theta + c \end{array} \right\} \quad (12) \quad \text{Avec} \quad \left\{ \begin{array}{l} l = \text{largeur} \\ L = \text{Longueur} \\ \theta \in [0; \pi] \\ \varphi \in [0; \pi] \end{array} \right.$$

Les critères pour vérifier la sortie du rayon par l'ouverture :

$$\left\{ \begin{array}{l} Z = 0 \Leftrightarrow r \cos \theta + c = 0 \Rightarrow r = \frac{-c}{\cos \theta} \\ \sqrt{X^2 + Y^2} \leq \frac{D}{2} \end{array} \right. \quad \text{permettent d'aboutir à la condition de caractérisation du}$$

rayon sortant :

$$\sqrt{X^2 + Y^2} \leq \frac{D}{2}, \quad \left\{ \begin{array}{l} X = -c \tan \theta \cos \varphi + \frac{l}{2} - a \\ Y = -c \tan \theta \sin \varphi - \frac{L}{2} \end{array} \right. \quad (13)$$

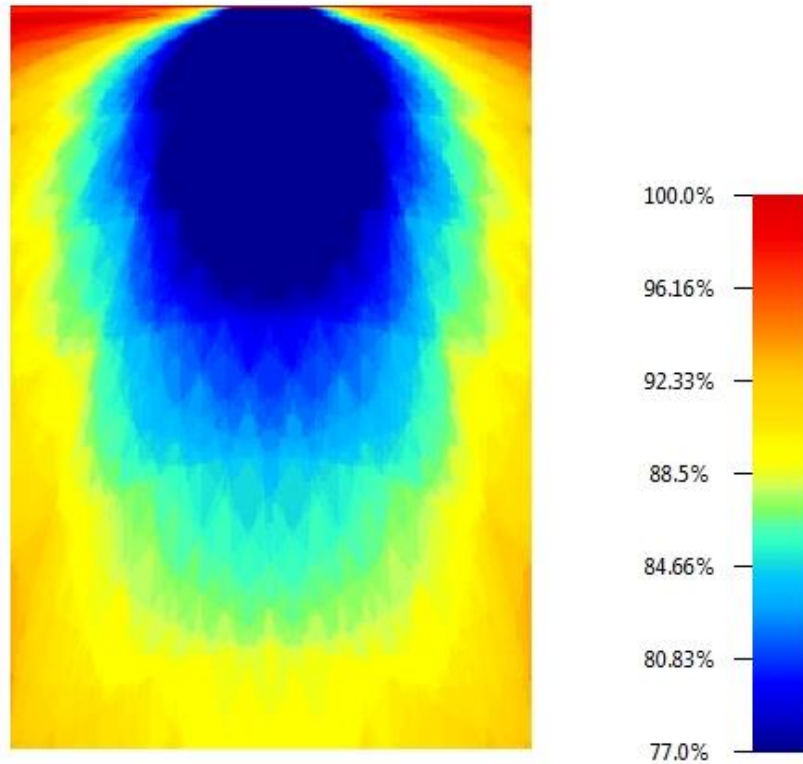


Figure 20 : Aperçu graphique de l'évolution de l'efficacité sur une façade latérale de la cavité parallélépipédique

Compte du masque entourant le diamètre d'ouverture, les points de la base de cette façade ont un angle de vue sur assez réduit l'ouverture: leur efficacité est donc assez élevée. Toutefois, quand on monte vers la partie supérieure de la cavité, cet angle de vue s'agrandit et donc l'efficacité diminue. L'autre remarque est que les points situés directement autour et sur l'axe de symétrie (suivant la hauteur) de cette façade ont un angle de vue sur l'ouverture assez grand et c'est ce qui explique que les plus faibles efficacités sont concentrées autour de cet axe et diminuent avec l'augmentation de la hauteur. Par contre, quand on s'éloigne de cet axe de symétrie et qu'on se rapproche des angles droits, l'efficacité augmente excessivement. La localisation de ces points fait que la grande majorité des rayons qu'ils génèrent atteignent plutôt les parois que l'ouverture.

#### b. Etude d'une cavité cylindrique

Considérons le repère fixe (O, x, y, z) situé au centre de la base inférieure ; l'axe (Oz) représente l'axe de symétrie du cylindre en question ce qui réduit les éléments à étudier au nombre de deux : la surface latérale et la base supérieure. Grâce à la méthode dite du Lancer de rayons, on procédera de

la même manière que dans le cas du parallélépipède tout en refaisant la mise en équations mais en conservant les différents repères.

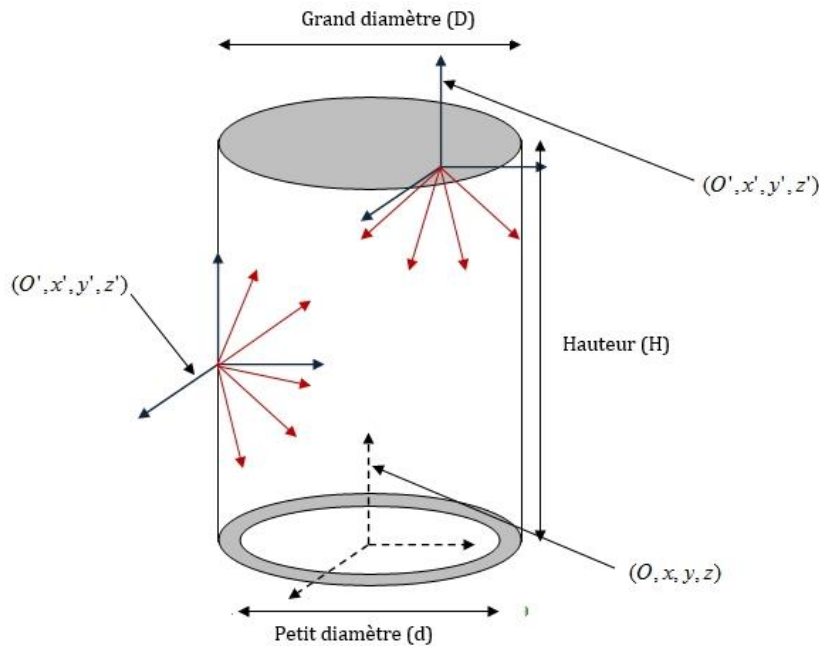


Figure 21 : Aperçu de la cavité cylindrique avec les différents repères d'études

Les études du comportement de cette cavité cylindrique vont porter sur la surface latérale et la base supérieure.

Les exemples qui sont illustrés dans la suite porte sur un cylindre de Hauteur (1m) et de Diamètre d'ouverture égale au Diamètre du cylindre (0,7m).

### Surface latérale

Au vu de l'axe de symétrie du cylindre, l'étude de la paroi se fait sur une unique génératrice géométrique. La mise en équation sur la paroi latérale se fait de la même façon que celle de la façade latérale de la cavité cubique. La différence réside les coordonnées  $(a, b, c) = (0, 0, c)$  du repère local  $(O, x', y', z')$  par rapport au repère fixe.

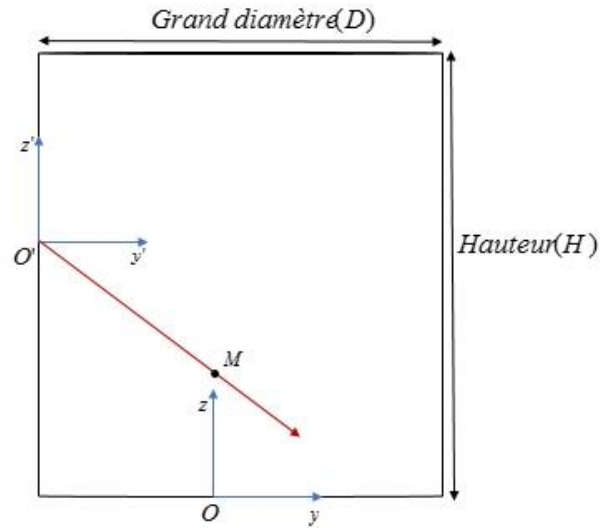


Figure 22 : Observation de la cavité cylindrique suivant le plan (yz)

Nous obtenons ainsi :

$$\begin{cases} X = r \sin \theta \cos \varphi \\ Y = r \sin \theta \sin \varphi - \frac{D}{2} \\ Z = r \cos \theta + c \end{cases} \quad (14) \quad \text{Avec} \quad \begin{cases} \theta \in [0; \pi] \\ \varphi \in [0; \pi] \end{cases}$$

Les critères pour déterminer un rayon sortant par l'ouverture sont les mêmes que précédemment, on en déduit la condition de caractérisation d'un rayon sortant par :

$$\sqrt{X^2 + Y^2} \leq \frac{d}{2}, \quad \begin{cases} X = -c \tan \theta \cos \varphi \\ Y = -c \tan \theta \sin \varphi - \frac{D}{2} \end{cases} \quad (15)$$

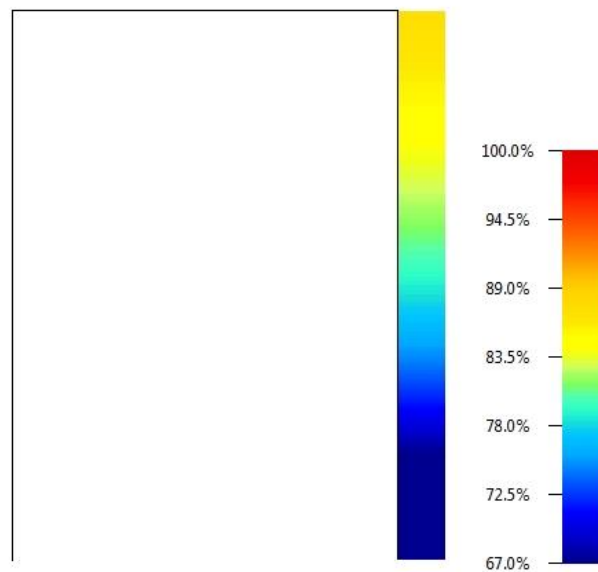


Figure 23 : Aperçu graphique de l'évolution de l'efficacité sur la paroi verticale

La distribution de l'efficacité sur les parois de cette cavité montre que l'efficacité augmente avec la hauteur, ce qui s'explique simplement par le fait que l'angle d'observation de l'ouverture diminue avec la hauteur. La distribution de l'efficacité est la même pour toutes les génératrices géométriques raison pour laquelle une seule a été considérée pour cette étude.

### Etude de la base supérieure

Considérant l'axe de symétrie passant par le centre de la base, l'étude va se restreindre sur un unique rayon. Le repère local est de coordonnées  $(a, b, c) = (0, b, h)$ .

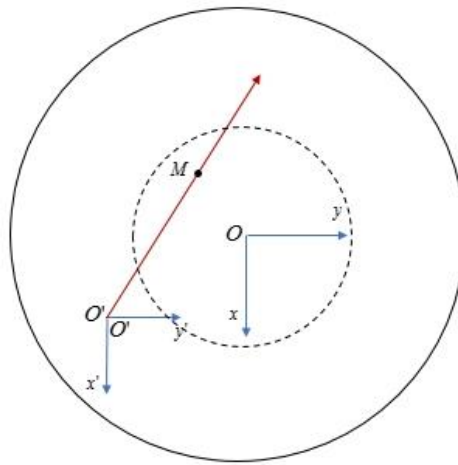


Figure 24 : Observation de la base supérieure de la cavité cylindrique suivant le plan  $(xy)$

Nous obtenons comme mise en équations :

$$\begin{cases} X = r \sin \theta \cos \varphi \\ Y = r \sin \theta \sin \varphi - b \\ Z = r \cos \theta + h \end{cases} \quad (16) \quad \text{Avec} \quad \begin{cases} \theta \in \left[ \frac{\pi}{2}; \pi \right] \\ \varphi \in [0; 2\pi] \end{cases}$$

La condition pour caractériser un rayon sortant par l'ouverture est donc :

$$\sqrt{X^2 + Y^2} \leq \frac{d}{2}, \quad \begin{cases} X = -c \tan \theta \cos \varphi \\ Y = -c \tan \theta \sin \varphi - b \end{cases} \quad (17)$$

La modélisation faite ci-dessus sur Python a pour objectif de comparer l'efficacité globale des cavités parallélépipédique et cylindrique afin d'opter pour un choix de géométrie pour notre récepteur solaire. Ne disposant du facteur de formes pour le parallélépipède, nous avons utilisé le 'Lancer de rayons' qui se prête bien aux simulations optiques. Au cours de ce travail, l'équipe

CSP4AFRICA a opté pour la géométrie cylindrique pour sa facilité de mise en œuvre et les pertes de charges réduites lors de la circulation du fluide caloporteur à travers le système tubulaire. Dans le même temps, le facteur de forme entre un élément de paroi et l'ouverture du cylindre a été obtenu. Nous avons donc procédé à une nouvelle modélisation que nous nommons ci le 'Modèle simplifié'.

### 3. Modèle simplifié

Ce modèle est basé sur les travaux de thèse de Stefania TESCARI qui ont porté sur l'« Optimisation géométrique dérivée de l'approche constructale pour réacteurs thermochimiques sous rayonnement solaire concentré ». Le revêtement qui sera utilisé pour la surface absorbante de notre récepteur possède une émissivité supérieure à 0,9. Dans ce modèle, nous étudierons le récepteur comme étant un corps parce que cette hypothèse reste acceptable pour une émissivité supérieure à 0,6 en considérant l'objectif du modèle simplifié [17].

Le facteur de forme entre un élément de paroi  $i$  et l'ouverture (o) d'un cylindre est donnée par la

$$\text{relation } F_{i \rightarrow o} = F_{o \rightarrow i} = \frac{Z}{2} \left\{ \frac{X}{\left[ X^2 - 4 \times R^2 \right]^{\frac{1}{2}}} - 1 \right\} \quad (18) \quad \text{avec } Z = \frac{z}{r_1}; R = \frac{r_2}{r_1}; X = 1 + Z^2 + R^2. \quad \text{Les}$$

différentes grandeurs sont indiquées sur la figure 27.

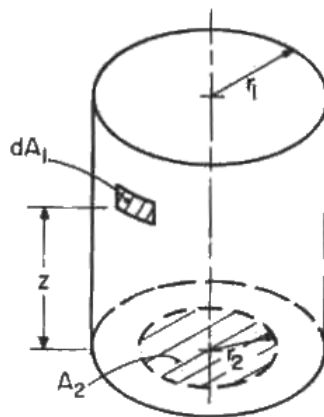


Figure 25 : Vue globale d'un cylindre avec les différentes grandeurs entrant dans le calcul du Facteur de forme

Le présent modèle prend en compte les transferts radiatifs ainsi que les échanges convectifs avec le fluide caloporteur. Les nombres adimensionnels requis pour le calcul du coefficient de convection sont :



$$Re = v_r \times d \times \frac{r_o}{\mu} \quad (19) \quad \text{avec} \quad \begin{cases} v_r : \text{Vitesse réelle de l'huile} \\ d : \text{Diamètre des tubes} \\ r_o : \text{Densité de l'huile} \\ \mu : \text{Viscosité de l'huile} \end{cases}$$

$$P_r = \mu \times Cp \times \frac{1000}{\lambda} \quad (20) \quad \text{avec} \quad \begin{cases} Cp : \text{Capacité calorifique de l'huile} \\ \lambda : \text{Conductivité de l'huile} \end{cases}$$

$$Nu = \begin{cases} 0,023 \times Re^{0,8} \times Pr^{0,33} & \text{si } Re > 2100 \\ 3,66 & \text{sinon} \end{cases} \quad (21)$$

Nous obtenons par suite, le coefficient d'échanges convectifs :

$$h = Nu \times \frac{\lambda}{d} \quad (22)$$

Les pertes de charges sont obtenues la

$$\text{relation } \Delta P = \text{blasius} \times l \times v_r^2 \times \frac{r_o}{2d} \quad (23) \quad \text{avec } l : \text{longueur totale de la tuyauterie}$$

Pour une puissance (Q) de 100 kWth envoyée dans le cylindre, la densité de flux moyen  $d\phi_m$  passant par l'ouverture de diamètre Do est obtenue par la relation :

$$d\phi_m = \frac{4 \times Q}{\pi \times Do^2} \quad (24)$$

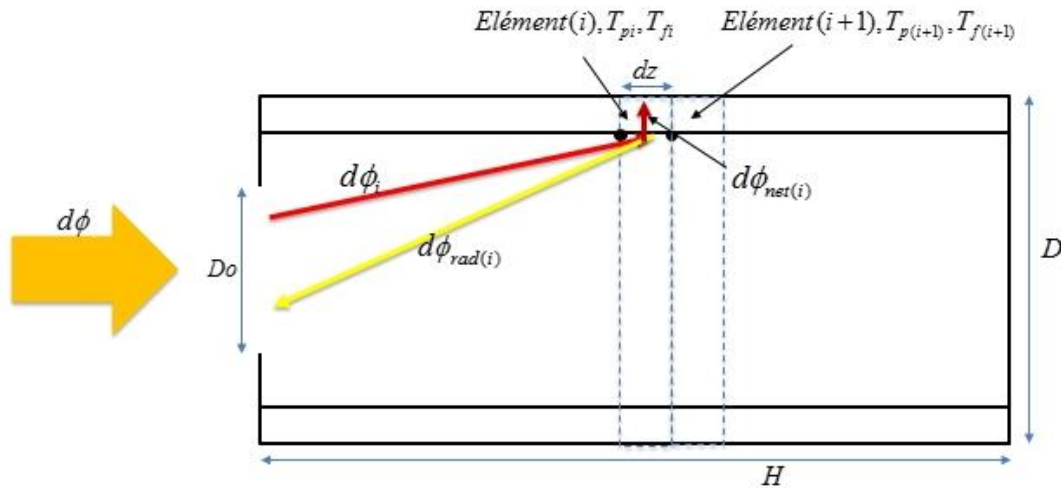


Figure 26 : Représentation graphique des échanges radiatifs et convectifs sur un élément de paroi du récepteur

La densité de flux entrant dans la cavité est réfléchiée par la paroi d'une part vers l'intérieur et de l'autre vers l'extérieur. Le modèle qui est décrit ci-dessous va permettre de déterminer la température du fluide à chaque point de la paroi verticale sur toute la hauteur du récepteur.

Lors du contact de la densité de flux avec l'élément de paroi, la quantité reçue par l'élément (i) est calculée par

$$d\phi_i = d\phi_m \times F_{o \rightarrow i} . \quad (25)$$

Lors de la réflexion sur l'élément (i), la densité de flux sortant par l'ouverture correspond aux pertes radiatives  $d\phi_{rad(i)}$  et s'obtient par :

$$d\phi_{rad(i)} = \sigma \times T^4 \times F_{i \rightarrow o} . \quad (26)$$

La densité de flux net  $\phi_{net(i)}$  reçue par l'élément (i) équivaut donc à la densité de flux qui est réellement absorbée par la paroi en ce point et est obtenue par :

$$d\phi_{net(i)} = d\phi_i - d\phi_{rad(i)} \quad (27)$$

Cette différence est ensuite fournie au fluide caloporteur qui va monter en température. Les consignes données pour la boucle thermique correspondent une température de 70°C à l'entrée du récepteur et 250°C à sa sortie. Pour ce modèle, 70°C sera donc considéré comme la température d'entrée du fluide et l'objectif sera d'atteindre les 250°C en sortie du récepteur. En posant le bilan de puissance, on obtient  $m_r C_p (T_{f(i+1)} - T_{f(i)}) = d\phi_{net(i)} \times \pi \times dz \times D$  (28).

Ainsi donc, connaissant la température du fluide au niveau de l'élément de paroi (i), on en déduit sa

température au niveau de l'élément de paroi (i+1) par  $T_{f(i+1)} = T_{f(i)} + \left( \frac{\pi \times D \times dz \times d\phi_{net(i)}}{m_r \times C_p} \right)$  (29).

Par intégration numérique sur toute la hauteur du récepteur, on connaît le profil de température du fluide le long de la paroi latérale du récepteur.

La différence de température  $\Delta T$  entre le fluide et la paroi en (i) est ensuite calculée grâce à la densité de flux net et au coefficient moyen de convection par le bilan de densité de flux :

$$h(T_{p(i)} - T_{f(i)}) = d\phi_{net(i)} = d\phi_{conv} . \quad (30)$$

Nous obtenons par suite la température de paroi de l'élément (i) et par intégration numérique, le profil de température de la paroi sur la hauteur du récepteur. Ces deux profils de température permettent de caractériser le comportement global de notre récepteur. A eux, s'ajoutent certains paramètres à savoir le diamètre des tubes, le diamètre du cylindre et la vitesse du fluide caloporteur, que nous avons fait varier pour obtenir les dimensions de notre récepteur. Il faut aussi rappeler que l'un des objectifs de la modélisation est d'absorber le maximum de flux sur la paroi latérale.

Nous obtenons après variation des différents paramètres, les graphes suivants :

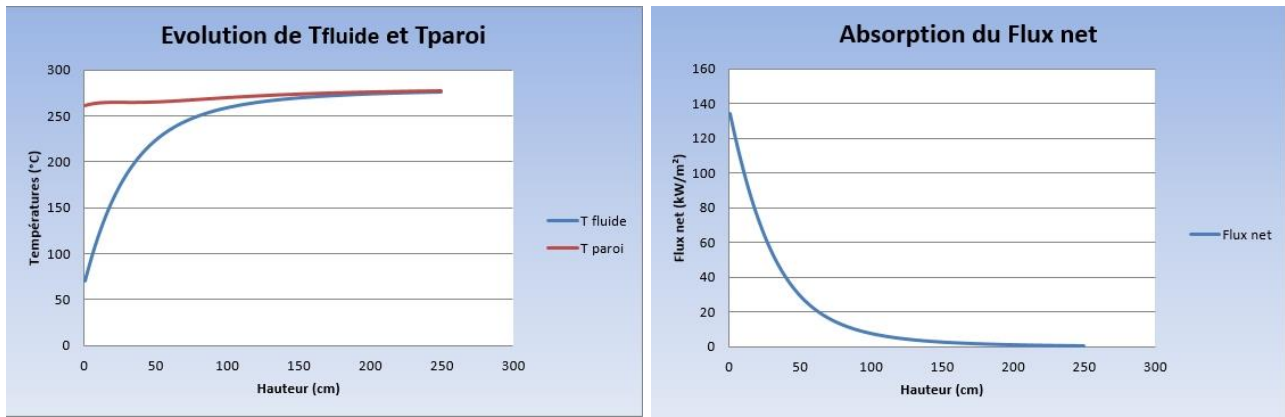


Figure 27 : Courbes d'évolution des températures de fluide et de paroi (Gauche) et Absorption du flux en fonction de la hauteur du récepteur

Il faut remarquer sur le graphe d'évolutions des températures (fluide et paroi) que l'absence de masque permet à la température de la paroi de s'élever dès la base où se trouve l'ouverture. Par suite, la température de paroi reste quasi-constante sur toute la hauteur. Cette élévation de la température de paroi permet à celle du fluide de grimper rapidement. On atteint les 250°C requis par la boucle thermodynamique bien avant le mètre de hauteur. Quant au flux arrivant sur la paroi, elle a été rapidement absorbée toujours grâce à l'absence de masque sur l'ouverture. A l'approche du mètre de récepteur, on a moins de 10% du flux entrant.

Le dimensionnement a abouti une à une hauteur de 1m, un diamètre d'ouverture équivalent au diamètre du système tubulaire de 0,7m et un diamètre de tube de 2,5 cm.

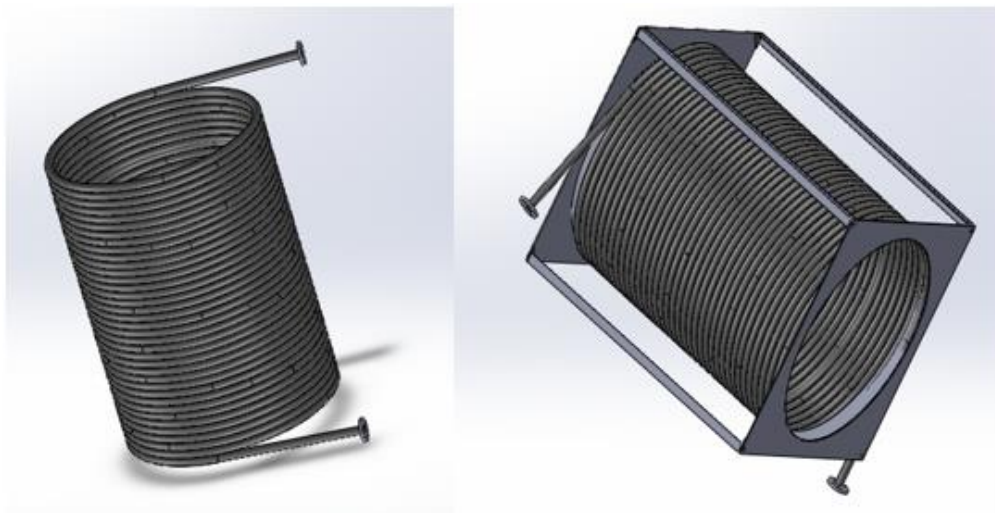


Figure 28 : Aperçu de l'absorbeur (gauche) et du récepteur (droite)

## V. Fabrication

Un dossier d'appel d'offres (DAO) a été préparé puis lancé après la phase de conception du récepteur solaire. Le dépouillement des offres a retenu la société AROM-H pour la fabrication du récepteur. Ce travail a commencé tardivement suite à de longues procédures administratives.

De la ferraille a d'abord été aplatie puis enroulée pour former un tambour cylindrique de diamètre 0,7 m autour duquel seront enroulés les tubes pour obtenir le corps de chauffe.



*Figure 29 : Aplatissement de la ferraille (Gauche) et constitution du tambour (Droite)*

Lors de la constitution proprement dite du corps de chauffe, un premier tube en acier a été enroulé autour du tambour ; un second tube a été soudé à son extrémité pour permettre un enroulement successif. Cependant, la soudure n'a cessé de céder. La solution trouvée consistait à enrouler l'ensemble des tubes pour passer ensuite à la soudure. Les tubes en acier ont donc été enroulés un à un autour de ce tambour, ensuite soudés les uns aux autres pour constituer le corps de chauffe.



*Figure 30 : Soudure des différents tubes (Gauche) pour constituer le corps de chauffe (Droite)*

La fabrication a donc pu avancer grâce à cette solution mais à ce jour, le récepteur encore moins le corps de chauffe n'a pu être totalement fabriqué.

Nous avons parallèlement à cette phase de fabrication conduit certains tests expérimentaux qui nous ont permis d'élaborer un protocole pour la caractérisation du récepteur.

De la tôle en Aluminium est utilisée pour former une cavité cylindrique (diamètre 23 cm, base supérieure fermée et base inférieure entièrement ouverte). Ne disposant pas de résistances de grande puissance qui fonctionnent à l'air ambiant, nous les avons substituées par des thermoplongeurs de puissance 1,5 kW trouvés sur le marché.



Figure 31 : Présentation de la cavité en tôle et du thermoplongeur

L'objectif de cette petite expérience était d'évaluer le comportement à long terme du thermoplongeur lorsqu'il fonctionne dans l'air ambiant et le profil thermique sur la paroi latérale de la cavité cylindrique.

Trente-cinq (35) minutes après la mise en fonctionnement du thermoplongeur, nous avons atteint une température de 150°C sur la paroi latérale à la hauteur où se trouve la résistance. Lorsqu'on tend vers les deux bases de la cavité, cette température diminue considérablement toutefois les points situés vers la base inférieure restent plus chauds que ceux situés vers le fond. Le même test a été conduit avec deux thermoplongeurs et une température de 186°C est atteinte après les trente-cinq minutes. Cependant, après 1h30 min de fonctionnement, la couleur rouge vive du thermoplongeur en fonctionnement tendait vers le noir simultanément avec la diminution des températures des différents points de la paroi latérale ; 30 min plus tard, le thermoplongeur était hors d'usage : nous avons donc conclu que la température de fonctionnement du thermoplongeur est bien trop élevée pour sa surface d'échange thermique. L'idée est donc d'augmenter cette surface d'échange afin de diminuer la température de fonctionnement de ce dernier. Pour ce faire, nous



avons enroulé autour de cette dernière une plaque cylindrique en aluminium de hauteur 20 cm et de diamètre 6 cm. La plaque en aluminium a commencé à fondre après 1h de fonctionnement : nous avons donc décidé de contrôler la puissance électrique fournie au thermoplongeur afin de limiter sa température de fonctionnement et par suite prolonger sa durée de vie.

Nous avons ensuite, élaboré un protocole expérimental comprenant deux expériences pour caractériser notre récepteur : la première consiste à vérifier l'isolation avec la laine de verre et la seconde à évaluer le profil thermique sur la paroi du récepteur en fonction de la puissance reçue.

## VI. Protocole expérimental

Pour la caractérisation du récepteur, le rayonnement solaire concentré est requis. Cependant, ne disposant d'un champ solaire, nous avons trouvé une solution alternative qui nous permet de créer un environnement radiatif à l'intérieur du récepteur : nous allons donc utiliser des résistances chauffantes que nous allons alimenter par du courant électrique. Dans l'incapacité de trouver des résistances de grande puissance qui fonctionnent dans l'air ambiant sans dommages, nous avons ainsi opté pour l'utilisation des thermoplongeurs. Un autre problème se pose cependant : ces thermoplongeurs sont conçus pour chauffer les liquides donc se dégradent lors de leur fonctionnement à l'air ambiant sous l'effet de l'élévation de la température. La solution trouvée est de limiter la tension que le thermoplongeur reçoit du secteur par l'utilisation d'un générateur de tension variable pour contrôler la puissance thermique qu'elle fournit en aval. L'utilisation du générateur de tension variable de tension permet donc de contrôler la tension de chaque thermocouple et par suite limiter la puissance  $P = \frac{U^2}{R}$  (31) qu'elle reçoit à 1 kW pour éviter d'atteindre la puissance maximale (1,5 kW), ce qui permettrait une diminution assez nette de la température de fonctionnement et ainsi éviter la dégradation des thermocouples.

Le schéma de principe de cette expérimentation tient compte de 6 thermocouples. Nous rappelons toutefois que ce nombre peut augmenter compte tenu des caractéristiques de notre générateur de tension.

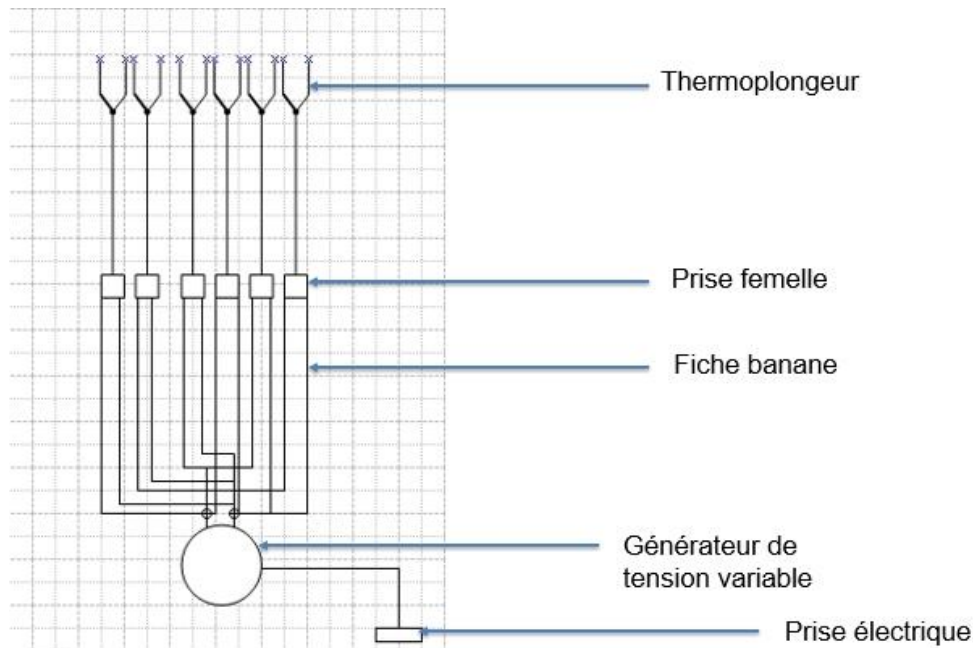


Figure 32 : Schéma de principe pour la création d'un environnement radiatif à l'intérieur du récepteur

### 1. Etude de l'isolation

Le but de cette étude est de s'assurer que l'isolation a été bien faite.

L'absorbeur sera d'abord isolé avec de la laine de verre. Une plaque métallique sera fixée sur l'ouverture afin de faire du récepteur une cavité fermée et compacte. La partie supérieure du récepteur sera percée afin d'offrir une ouverture (raisonnable) pour l'insertion des thermoplongeurs. Les sondes des thermocouples seront disposées sur une même génératrice géométrique (Figure 33), la première moitié à l'intérieur de la cavité et la seconde moitié à l'extérieur sur la laine de verre de telle sorte à ce que les sondes soient deux à deux en regard sur un même rayon. Elles seront ensuite connectées au banc d'acquisition.

Une fois dans la cavité et le dispositif expérimental en place, les thermoplongeurs seront alimentés pour fournir une puissance de chauffage assez élevée. Face au faible nombre de thermocouples dont nous disposons, il est nécessaire de refaire l'expérience en disposant les thermocouples sur d'autres génératrices géométriques et recommencer l'expérimentation. Les données attendues sont les profils de température au niveau des différentes génératrices géométriques sur lesquelles les sondes sont disposées.

L'analyse de ces différentes courbes permettra de déceler les ponts thermiques au cas où il y en aurait et refaire l'isolation.



Pour chaque génératrice géométrique considérée, l'expérience va se dérouler sur deux heures. Les différentes étapes sont les suivantes :

- Enlever la plaque creuse de la base inférieure du récepteur puis la remplacer par une plaque compacte afin de fermer la partie ouverte.
- Isoler tout le récepteur avec la laine de verre
- Percer de façon centrée la plaque de la base inférieure afin de créer une ouverture pour faire entrer les thermoplongeurs.
- Poser le récepteur en verticale de sorte que la base bercée soit orientée vers le haut
- Disposer la moitié des sondes à l'intérieur du récepteur sur une même génératrice géométrique de la paroi latérale.
- Disposer d'autres sondes à l'extérieur du récepteur sur la laine de verre de sorte que les sondes se trouvent 2 à 2 sur la même trajectoire radiale.
- Brancher les différentes sondes au Banc d'acquisition comme indiqué dans la notice.
- Mettre en place le dispositif expérimental du schéma de principe et insérer les thermoplongeurs dans la cavité à travers l'ouverture de la plaque.
- Mettre le chronomètre en marche simultanément avec l'alimentation des thermoplongeurs et la mise en fonctionnement du Banc d'acquisition. Le chauffage de la cavité se fera avec une puissance non variable donc la plus élevée (5kW).
- Relevez les températures indiquées par les différentes sondes dans le tableau ci-dessous pour une durée de 120 minutes pour un pas de temps de 5 minutes.
- Recommencer l'expérience en considérant d'autres génératrices géométriques.

Tableau de relevés des mesures thermiques

	T1 (°C)		T2 (°C)		T3 (°C)		T4 (°C)	
Temps (min)	Intérieur	Extérieur	Intérieur	Extérieur	Intérieur	Extérieur	Intérieur	Extérieur
5								
10								
15								
20								
25								

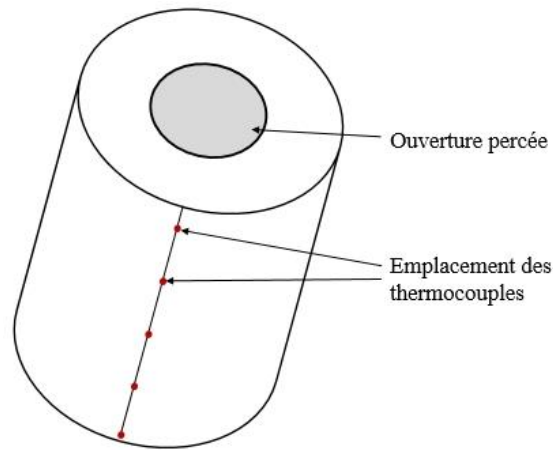


Figure 33 : Aperçu de la disposition des thermocouples sur l'extérieur du récepteur

## 2. Caractérisation du récepteur

Le but de cette expérimentation est de caractériser le prototype de récepteur. Il s'agit donc d' :

- Evaluer les pertes radiatives
- Evaluer les pertes convectives et ensuite déterminer le coefficient de pertes convectives  $h$

Pour ce faire, nous considérerons notre récepteur isolé avec la laine de verre. Les sondes des thermocouples seront disposées sur une génératrice unique à l'intérieur. Le schéma de principe sera utilisé pour engendrer le chauffage dans la cavité. Les thermoplongeurs seront envoyés à l'intérieur de la cavité avec l'aide d'un support pour les maintenir si possible dans l'axe de symétrie.

Le début de l'expérience va coïncider avec l'entrée de l'eau dans le système tubulaire du récepteur, l'alimentation des résistances en courant électrique et la mise en marche du chronomètre. Notons que la pression d'eau envoyée dans le système tubulaire, la puissance de chauffage et l'angle d'inclinaison du récepteur seront variables.

Il s'agit ici pour chaque pression d'eau de :

- Déterminer le temps de séjour de l'eau dans le système tubulaire
- Déterminer le débit correspondant
- Relever les valeurs des températures d'entrée et de sortie de l'eau du système tubulaire

Et pour chaque puissance de chauffage :

- Relever les valeurs des températures des différentes sondes du début de l'expérience jusqu'à l'équilibre thermique
- Déterminer le temps d'attente pour l'équilibre thermique de l'absorbeur.

Le récepteur étant supposé isolé et prêt à l'emploi, pour chaque inclinaison par rapport à la verticale :

- Placer les sondes sur une génératrice géométrique à l'intérieur de la paroi latérale du récepteur
- Placer une sonde à l'entrée puis une autre à la sortie du système de circulation du fluide pour mesurer les températures de l'eau à l'entrée et à la sortie du récepteur.
- Connectez les différentes sondes au Banc d'acquisition comme indiqué dans la notice d'utilisation
- Mettre en place le schéma de principe et insérer les thermocouples dans l'axe de symétrie de la cavité.
- Démarrer le chronomètre simultanément avec l'entrée de l'eau dans le système de circulation, l'alimentation des thermoplongeurs et la mise en marche du Banc d'acquisition.
- Pour chaque pression d'eau au niveau du robinet :
  - Déterminer le temps de séjour de l'eau dans le système de circulation
  - Déterminer le débit correspondant
  - Relever les températures des sondes à l'entrée et à la sortie du système
  - Relever les températures des sondes sur la paroi latérale par pas de 5 minutes jusqu'à stagnation
  - Déterminer le temps de stagnation de la température sur la paroi latérale
  - Déterminer la température maximale après stagnation
- Les différentes analyses à mener vont porter sur :
  - La température de sortie de l'eau en fonction de la puissance de chauffage et du débit de l'eau dans le système
  - La variation du temps de stagnation (pour une puissance de chauffage identique) en fonction du débit d'eau dans le système.
  - L'impact de l'inclinaison sur le débit
- Déterminer les grandeurs relatives à la caractérisation du récepteur
  - Une corrélation entre la puissance de chauffage et la température de sortie de l'eau du récepteur avec le débit comme paramètre
  - Une corrélation entre la température de stagnation et le débit de circulation du fluide
  - Une corrélation entre le débit et l'inclinaison du récepteur

## Tableau de relevés de mesure

			Temps (min)	T1 (°C)	T2 (°C)	T3 (°C)	T4 (°C)	T5 (°C)	Temps de stagnation
<b>Inclinaison 1</b>	<b>Débit 1</b>	<b>Puissance 1</b>	5						
			10						
			15						
			20						
			25						
		<b>Puissance 2</b>	5						
			10						
			15						
			20						
			25						
		<b>Puissance 3</b>	5						
			10						
			15						
			20						
			25						
	<b>Débit 2</b>	<b>Puissance 1</b>	5						
			10						
			15						
			20						
			25						
		<b>Puissance 2</b>	5						
			10						
			15						
			20						
			25						
		<b>Puissance</b>	5						
			10						
			15						
			20						
			25						

Pour la détermination du coefficient moyen de convection, il est nécessaire connaître la quantité de flux reçue par l'absorbeur c'est-à-dire le flux net. Pour cela, il est nécessaire d'évaluer la quantité de flux perdue par l'ouverture du récepteur. Au vu des moyens dont nous disposons, la détermination du flux net se fera sur la base de certaines hypothèses et considérations.

### Détermination du Flux net

Nous avons par définition :

$$\phi_n = \phi_g - \phi_p \quad (32) \quad \text{Avec} \quad \begin{cases} \phi_g = \text{Flux g n r  par les thermoplongeurs} \\ \phi_p = \text{Flux directement rayonn  vers l'ouverture par les thermoplongeurs} \\ \phi_n = \text{Flux net re u par la paroi} \end{cases}$$

Par consid ration g om trique, la puissance perdue est la quantit  de puissance thermique g n r e par la r sistance qui sort par l'ouverture du r cepteur lors du rayonnement des thermoplongeurs. Elle peut donc  tre assimil e   l'angle solide sous lequel les thermoplongeurs voient l'ouverture du r cepteur.

Deux hypoth ses sont alors pos es :

- Le bloc des thermoplongeurs se trouve sur l'axe de sym trie
- Le bloc des thermoplongeurs sera assimil    un  l ment ponctuel qui se trouve en son centre g om trique ainsi l'angle solide   partir de ce point sera fonction de la distance du centre de l'ouverture (l)   l' l ment ponctuel et du rayon (r) de cette ouverture.

L'ouverture et l' l ment ponctuel consid r  forment donc un c ne : on exprime l'angle solide par  $\Omega = 2\pi * (1 - \cos\alpha)$  avec  $\alpha$  : la moiti  de l'angle de r volution du c ne  nonc  ci-dessus.

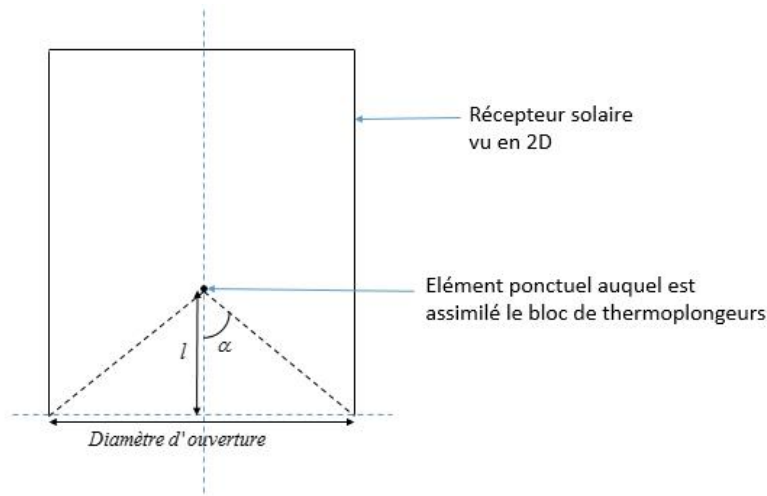


Figure 34 : Localisation des diff rents  l ments sur la vue en 2D du r cepteur

$$\text{On obtient alors } \phi_p = \phi_g \times \frac{\Omega}{4\pi} = \frac{1}{2} \times (1 - \cos\alpha) \times \phi_g \quad (33)$$

En revenant   la d termination du coefficient de convection, on peut alors poser :

$$\phi_{conv} = \phi_n - \phi_{rad} \quad (34) \quad \text{avec} \quad \begin{cases} \phi_{conv} = hS\Delta T \\ \phi_{rad} = \text{Pertes radiatives} \end{cases}$$

On en d duira ensuite le coefficient moyen (h) des pertes convectives.

## VII. Conclusion

Dans ce mémoire qui a porté sur la conception et le montage d'un récepteur solaire pour centrale à tour, il était question de mettre en place un modèle de dimensionnement d'un récepteur et superviser la fabrication de ce dernier.

Après une étude bibliographique détaillée sur les récepteurs solaires pour centrales à tour, nous avons opté pour un récepteur de type cavité pour ses faibles pertes thermiques et son absorptivité excellente. Au vu de l'objectif du projet CSP4AFRICA et dans le souci de réduire les coûts de mise en place de la centrale, nous avons choisi un absorbeur surfacique de forme cylindrique pour sa simplicité de mise en œuvre au niveau local. Nous avons ensuite mis en place un « modèle simplifié » qui prend en compte les transferts radiatifs et les échanges convectifs avec le fluide caloporteur ; ce modèle nous a permis de faire notre dimensionnement et obtenir un récepteur solaire de hauteur  $H=1\text{m}$ , un diamètre d'ouverture équivalent au diamètre du système  $D=0,7\text{m}$ .

Lors de la phase de fabrication, nous avons suivi de près les travaux de mise en œuvre du corps de chauffe. Certains problèmes techniques ont surgi et des solutions ont pu être apportées notamment au niveau de l'enroulement des tubes pour la constitution de l'absorbeur.

Nous avons parallèlement à l'étape de fabrication mis en place un protocole expérimental qui comprend deux expériences distinctes. La première permet de vérifier la réussite de l'isolation du récepteur et la seconde de faire sa caractérisation. L'absence de résistances fonctionnant sans dommages à l'air ambiant nous a poussés à les substituer par des thermoplongeurs et élaborer ainsi un protocole adapté.

Cependant, la fabrication de notre récepteur aura pris plusieurs semaines. Ce dernier n'a donc pas pu être livré avant la fin de notre stage.

Nous n'avons donc pas pu le caractériser et encore moins mettre à l'épreuve le protocole élaboré.

- [1] **Energy, US Department of.** *www.eere.energy.gov.* [En ligne] [Citation : 04 Juin 2012.]
- [2] **ADEME.** *Feuille de route Solaire thermodynamique.* s.l. : AXEL Client.com, 2011. 6915.
- [3] **AZOUMAH, Yao K.** Le Solaire thermique. Ouagadougou : s.n
- [4] **IEPF.** *L'énergie solaire thermique à concentration.* s.l. : PRISME, 2008. 4.
- [5] **FERRIERE, Alain et VRINAT, Mathieu.** Centrales à tour. Perpignan : s.n., 2010.
- [6] **QUOILIN, Sylvain.** Les Centrales Solaires à Concentration. Liège : s.n., 2007.
- [7] **Energy, US Department of.** *www.eere.energy.gov.* [En ligne] [Citation : 04 Juin 2012.]
- [8] *www.ecosources.com.* [En ligne] [Citation : 07 Juin 2012.]
- [9] **Syndicat des Energies renouvelables.** Les différents types de centrales solaires thermodynamiques. [En ligne] Juin 2011.
- [10] **FLAMANT, Gilles.** Solaire à concentration, du présent au futur. 2012.
- [11] **FERRIERE, Alain.** Les récepteurs solaires surfaciques: état de l'art et applications. 2012.
- [12] **STINE, William B.** <http://www.powerfromthesun.net/Book/chapter10/chapter10.html>. [En ligne] [Citation : 28 Juin 2012.]
- [13] **CALIOT, Cyril.** Les récepteurs volumiques pour centrales solaires à concentration: état de l'art et applications. 2012.
- [14] **Prakash, M., Kedare, S.B. et Nayak, J.K.** Numerical study of natural convection loss from open cavities. *International Journal of Thermal Sciences.* Elsevier Masson SAS, 2011, Vol. 51.
- [15] [http://fr.wikipedia.org/wiki/Lancer\\_de\\_rayon](http://fr.wikipedia.org/wiki/Lancer_de_rayon)
- [16] [http://fr.wikipedia.org/wiki/Radiosit%C3%A9#Distribution\\_angulaire](http://fr.wikipedia.org/wiki/Radiosit%C3%A9#Distribution_angulaire)
- [17] Tescari, S. Thèse de doctorat en Sciences pour l'Ingénieur, Université de Perpignan, 2010.

## VIII. Annexes

### Explosion de rayons sur la paroi horizontale (Modèle 2D)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def formerecepteur():
    "Contour du recepteur"
    coul='black'
    can.create_line(350-larg/2,350-haut/2,350+larg/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350-haut/2,350+larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350+haut/2,350+diam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-diam/2,350+haut/2,350-larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-larg/2,350+haut/2,350-larg/2,350-haut/2,width=1,fill=coul)

def rayonreflechi(event):
    global effx, larg, haut, diam, Eff
    can.delete(ALL)
    larg=4*eval(largeur.get())
    haut=4*eval(hauteur.get())
    diam=4*eval(diametre.get())
    formerecepteur()
    lstx=[]
    lsty=[]
    Trsx=0
    Trsy=0
    for x in range(350-larg/2,350+larg/2,300):
        angle=pi/180
        tot=0
        while angle<pi:
            decide(x,angle)
            can.create_line(x,350-haut/2,x+haut/tan(angle),350+haut/2,width=1,fill=col)
            angle=angle+pi/180
```





```
if col=='green':  
    tot=tot+1  
    effx=(180-tot)*100/180.  
    lstx.append(effx)  
    Trsx+=tot  
print Trsx  
print lstx
```

```
def decide(a,b):  
    global col  
    if (a+haut/tan(b)>350-diam/2) and (a+haut/tan(b)<350+diam/2):  
        col='green'  
    else:  
        col='red'
```

```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',  
      '#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',  
      '#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',  
      '#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',  
      '#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',  
      '#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',  
      '#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',  
      '#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecdd','#27f4d0','#2afacbb',  
      '#2fffc7','#39ffc0','#3fffbdb','#49ffb9','#4fffb0','#5affa9','#62ff90',  
      '#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',  
      '#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',  
      '#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa000',  
      '#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',  
      '#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',  
      '#ffd400','#ffd200','#ffcd00','#ffcb00','#ffc800','#ffc200','#ffbc00',  
      '#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',  
      '#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',  
      '#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',
```



```
'#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',  
'#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',  
'#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',  
'#e40000','#e20000','#e00000']
```

```
def couleur():
```

```
    # Fonction qui permet au programme de choisir la couleur en fonction
```

```
    # de la valeur de la mesure et de l'échelle choisie
```

```
    global c
```

```
    if effx<=deb:
```

```
        c=coul[0]
```

```
    elif effx>=fi:
```

```
        c=coul[149]
```

```
    else:
```

```
        x1=deb
```

```
        i=1
```

```
        while i<149:
```

```
            if effx>x1 and effx<=x1+fi/149.:
```

```
                c=coul[i]
```

```
                i+=1
```

```
                x1+=(fi-deb)/149.
```

```
fen=Tk()
```

```
fen.title("Lancer de rayons Model 2D")
```

```
largeur,hauteur,diametre,temperature,debut,final,emissivite=StringVar(),StringVar(),StringVar(),Str  
ingVar(),StringVar(),StringVar(),StringVar()
```

```
txt1=Label(fen,text='Largeur:').grid(row=0,sticky=E)
```

```
txt2=Label(fen,text='Hauteur:').grid(row=1,sticky=E)
```

```
txt3=Label(fen,text='Diametre:').grid(row=2,sticky=E)
```

```
entr1=Entry(fen,textvariable=largeur)
```

```
entr2=Entry(fen,textvariable=hauteur)
```

```
entr3=Entry(fen,textvariable=diametre)
```



```
entr1.grid(row=0,column=1)
entr2.grid(row=1,column=1)
entr3.grid(row=2,column=1)
can=Canvas(fen,bg='white',height=700,width=700)
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Affichage',command=formerecepteur)
b1.grid(row=3,column=1)
b1.bind('<ButtonPress-1>',rayonreflechi)
fen.mainloop()
```

## Explosion de rayons sur la paroi latérale (Modèle 2D)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def formerecepteur():
    "Contour du recepteur"
    coul='black'
    can.create_line(350-larg/2,350-haut/2,350+larg/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350-haut/2,350+larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350+haut/2,350+diam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-diam/2,350+haut/2,350-larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-larg/2,350+haut/2,350-larg/2,350-haut/2,width=1,fill=coul)

def rayonreflechi(event):
    global effy,larg,haut,diam,Eff
    can.delete(ALL)
    larg=4*eval(largeur.get())
    haut=4*eval(hauteur.get())
    diam=4*eval(diametre.get())
    formerecepteur()
    lstx=[]
    lsty=[]
    Trsx=0
    Trsy=0
    for y in range(0,haut,350):
        angle=0
        tot=0
        while angle<pi/2:
            decide2(y,angle)
            can.create_line(350+larg/2,350+haut/2-y,350+larg/2-
y*tan(angle),350+haut/2,width=1,fill=col)
            angle=angle+pi/1800
```



```
if col=='green':
    tot=tot+1
effy=(1800-tot)*100/1800.
lsty.append(effy)
Trsy+=tot
print Trsy
print lsty
```

```
def decide2(c,d):
```

```
    global col
    if (350+larg/2-c*tan(d)>350-diam/2) and (350+larg/2-c*tan(d)<350+diam/2):
        col='green'
    else:
        col='red'
```

```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',
      '#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',
      '#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',
      '#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',
      '#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',
      '#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',
      '#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',
      '#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecdd','#27f4d0','#2afacbf',
      '#2fffc7','#39ffc0','#3fffbf','#49ffb9','#4fffb0','#5affa9','#62ff90',
      '#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',
      '#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',
      '#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa000',
      '#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',
      '#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',
      '#ffd400','#ffd200','#ffcd00','#ffcb00','#ffc800','#ffc200','#ffbc00',
      '#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',
      '#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',
      '#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',
```



```
'#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',  
'#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',  
'#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',  
'#e40000','#e20000','#e00000']
```

```
def couleur2():
```

```
    # Fonction qui permet au programme de choisir la couleur en fonction  
    # de la valeur de la mesure et de l'échelle choisie
```

```
    global c
```

```
    if effy<=deb:
```

```
        c=coul[0]
```

```
    elif effy>=fi:
```

```
        c=coul[149]
```

```
    else:
```

```
        y1=deb
```

```
        i=1
```

```
        while i<149:
```

```
            if effy>y1 and effy<=y1+fi/149.:
```

```
                c=coul[i]
```

```
                i+=1
```

```
                y1+=(fi-deb)/149.
```

```
fen=Tk()
```

```
fen.title("Lancer de rayons Model 2D")
```

```
largeur,hauteur,diametre,temperature,debut,final,emissivite=StringVar(),StringVar(),StringVar(),Str  
ingVar(),StringVar(),StringVar(),StringVar()
```

```
txt1=Label(fen,text='Largeur:').grid(row=0,sticky=E)
```

```
txt2=Label(fen,text='Hauteur:').grid(row=1,sticky=E)
```

```
txt3=Label(fen,text='Diametre:').grid(row=2,sticky=E)
```

```
entr1=Entry(fen,textvariable=largeur)
```

```
entr2=Entry(fen,textvariable=hauteur)
```

```
entr3=Entry(fen,textvariable=diametre)
```

```
entr1.grid(row=0,column=1)
```



```
entr2.grid(row=1,column=1)
entr3.grid(row=2,column=1)
can=Canvas(fen,bg='white',height=700,width=700)
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Affichage',command=formerecepteur)
b1.grid(row=3,column=1)
b1.bind('<KeyPress-1>',rayonreflechi)
fen.mainloop()
```

## Représentation de l'efficacité sur la paroi horizontale (Modèle 2D)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def formerecepteur():
    "Contour du recepteur"
    coul='black'
    can.create_line(350-larg/2,350-haut/2,350+larg/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350-haut/2,350+larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350+haut/2,350+diam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-diam/2,350+haut/2,350-larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-larg/2,350+haut/2,350-larg/2,350-haut/2,width=1,fill=coul)

def rayonreflechi(event):
    global effx,deb,fi,larg,haut,diam,Pertes,Eff
    can.delete(ALL)
    larg=4*eval(largeur.get())
    haut=4*eval(hauteur.get())
    diam=4*eval(diametre.get())
    deb=eval(debut.get())
    fi=eval(final.get())
    formerecepteur()
    lstx=[]
    Trsx=0
    coloriage()
    for x in range(350-larg/2,350++larg/2,1):
        angle=pi/180
        tot=0
        while angle<pi:
            decide(x,angle)
            #can.create_line(x,350-haut/2,x+haut/tan(angle),350+haut/2,width=1,fill=col)
            angle=angle+pi/180
```





```
    if col=='green':
        tot=tot+1
    effx=(180-tot)*100/180.
    lstx.append(effx)
    Trsx+=tot
#print Trsx
#print lstx
for j in range(0,larg,1):
    effx=lstx[j]
    couleur()
    can.create_line(350-larg/2+j,350-haut/2,350-larg/2+j,350-haut/2-35,fill=c)
```

```
def decide(a,b):
    global col
    if (a+haut/tan(b)>350-diam/2) and (a+haut/tan(b)<350+diam/2):
        col='green'
    else:
        col='red'
```

```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',
      '#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',
      '#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',
      '#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',
      '#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',
      '#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',
      '#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',
      '#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecd4','#27f4d0','#2afacb',
      '#2fffc7','#39ffc0','#3fffbf','#49ffb9','#4fffb0','#5affa9','#62ff90',
      '#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',
      '#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',
      '#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa00',
      '#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',
```



```
'#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',  
'#ffd400','#ffd200','#ffcd00','#ffc000','#ffc800','#ffc200','#ffbc00',  
'#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',  
'#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',  
'#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',  
'#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',  
'#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',  
'#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',  
'#e40000','#e20000','#e00000']
```

```
def coloriage():
```

```
    global eff,deb,fi,ecar,ecarligne
```

```
    ecar=(fi-deb)/6.0
```

```
    ecarligne=int((353+larg/2-(353+larg/2-300))/6.0*100)/100.0
```

```
    for i in range(150):
```

```
        can.create_rectangle(350+larg/2+120,350+haut/2-2*i,380+larg/2+120,353+haut/2-  
2*i,width=0,fill=coul[i])
```

```
        for i in range(0,7,1):
```

```
            can.create_text(350+larg/2+80,350+haut/2+2-  
i*ecarligne,text=str(int((deb+i*ecar)*100)/100.0)+"%")
```

```
            can.create_line(350+larg/2+110,350+haut/2+2-i*ecarligne,350+larg/2+120,350+haut/2+2-  
i*ecarligne,width=1)
```

```
def couleur():
```

```
    # Fonction qui permet au programme de choisir la couleur en fonction
```

```
    # de la valeur de la mesure et de l'échelle choisie
```

```
    global c
```

```
    if effx<=deb:
```

```
        c=coul[0]
```

```
    elif effx>=fi:
```

```
        c=coul[149]
```

```
    else:
```



```
x1=deb
i=1
while i<149:
    if effx>x1 and effx<=x1+fi/149.:
        c=coul[i]
        i+=1
    x1+=(fi-deb)/149.
```

```
fen=Tk()
fen.title("Lancer de rayons Model 2D")
largeur,hauteur,diametre,temperature,debut,final,emissivite=StringVar(),StringVar(),StringVar(),Str
ingVar(),StringVar(),StringVar(),StringVar()
txt1=Label(fen,text='Largeur:').grid(row=0,sticky=E)
txt2=Label(fen,text='Hauteur:').grid(row=1,sticky=E)
txt3=Label(fen,text='Diametre:').grid(row=2,sticky=E)
txt4=Label(fen,text='Debut:').grid(row=3,sticky=E)
txt5=Label(fen,text='Final:').grid(row=4,sticky=E)
entr1=Entry(fen,textvariable=largeur)
entr2=Entry(fen,textvariable=hauteur)
entr3=Entry(fen,textvariable=diametre)
entr4=Entry(fen,textvariable=debut)
entr5=Entry(fen,textvariable=final)
entr1.grid(row=0,column=1)
entr2.grid(row=1,column=1)
entr3.grid(row=2,column=1)
entr4.grid(row=3,column=1)
entr5.grid(row=4,column=1)
can=Canvas(fen,bg='white',height=700,width=700)
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Affichage',command=formerecepteur)
b1.grid(row=5,column=1)
b1.bind('<ButtonPress-1>',rayonreflechi)
```



fen.mainloop()

## Représentation de l'efficacité sur la paroi latérale (Modèle 2D)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def formerecepteur():
    "Contour du recepteur"
    coul='black'
    can.create_line(350-larg/2,350-haut/2,350+larg/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350-haut/2,350+larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350+haut/2,350+diam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-diam/2,350+haut/2,350-larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-larg/2,350+haut/2,350-larg/2,350-haut/2,width=1,fill=coul)

def rayonreflechi(event):
    global effy,deb,fi,larg,haut,diam,Pertes,Eff
    can.delete(ALL)
    larg=4*eval(largeur.get())
    haut=4*eval(hauteur.get())
    diam=4*eval(diametre.get())
    deb=eval(debut.get())
    fi=eval(final.get())
    formerecepteur()
    lsty=[]
    Trsy=0
    coloriage()
    for y in range(0,haut,1):
        angle=0
        tot=0
        while angle<pi/2:
            decide2(y,angle)
            #can.create_line(350+larg/2,350+haut/2-y,350+larg/2-
y*tan(angle),350+haut/2,width=1,fill=col)
```



```
angle=angle+pi/180
if col=='green':
    tot=tot+1
effy=(180-tot)*100/180.
lsty.append(effy)
Trsy+=tot
#print Trsy
#print lsty
for j in range(0,haut,1):
    effy=lsty[j]
    couleur2()
    can.create_line(350+larg/2,350+haut/2-j,350+larg/2+35,350+haut/2-j,fill=c)
```

```
def decide2(c,d):
    global col
    if (350+larg/2-c*tan(d)>350-diam/2) and (350+larg/2-c*tan(d)<350+diam/2):
        col='green'
    else:
        col='red'
```

```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',
      '#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',
      '#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',
      '#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',
      '#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',
      '#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',
      '#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',
      '#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecd4','#27f4d0','#2afacb',
      '#2fffc7','#39ffc0','#3fffbf','#49ffb9','#4fffb0','#5affa9','#62ff90',
      '#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',
      '#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',
      '#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa00',
      '#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',
```



```
'#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',  
'#ffd400','#ffd200','#ffcd00','#ffcb00','#ffc800','#ffc200','#ffbc00',  
'#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',  
'#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',  
'#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',  
'#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',  
'#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',  
'#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',  
'#e40000','#e20000','#e00000']
```

```
def coloriage():
```

```
    global eff,deb,fi,ecar,ecarligne
```

```
    ecar=(fi-deb)/6.0
```

```
    ecarligne=int((353+larg/2-(353+larg/2-300))/6.0*100)/100.0
```

```
    for i in range(150):
```

```
        can.create_rectangle(350+larg/2+120,350+haut/2-2*i,380+larg/2+120,353+haut/2-  
2*i,width=0,fill=coul[i])
```

```
    for i in range(0,7,1):
```

```
        can.create_text(350+larg/2+80,350+haut/2+2-  
i*ecarligne,text=str(int((deb+i*ecar)*100)/100.0)+"% ")
```

```
        can.create_line(350+larg/2+110,350+haut/2+2-i*ecarligne,350+larg/2+120,350+haut/2+2-  
i*ecarligne,width=1)
```

```
def couleur2():
```

```
    # Fonction qui permet au programme de choisir la couleur en fonction
```

```
    # de la valeur de la mesure et de l'échelle choisie
```

```
    global c
```

```
    if effy<=deb:
```

```
        c=coul[0]
```

```
    elif effy>=fi:
```

```
        c=coul[149]
```

```
    else:
```

```
        y1=deb
```



```
i=1
while i<149:
    if effy>y1 and effy<=y1+fi/149.:
        c=coul[i]
        i+=1
        y1+=(fi-deb)/149.
```

```
fen=Tk()
fen.title("Lancer de rayons Model 2D")
largeur,hauteur,diametre,temperature,debut,final,emissivite=StringVar(),StringVar(),StringVar(),StringVar(),StringVar(),StringVar(),StringVar()
txt1=Label(fen,text='Largeur:').grid(row=0,sticky=E)
txt2=Label(fen,text='Hauteur:').grid(row=1,sticky=E)
txt3=Label(fen,text='Diametre:').grid(row=2,sticky=E)
txt4=Label(fen,text='Debut:').grid(row=3,sticky=E)
txt5=Label(fen,text='Final:').grid(row=4,sticky=E)
entr1=Entry(fen,textvariable=largeur)
entr2=Entry(fen,textvariable=hauteur)
entr3=Entry(fen,textvariable=diametre)
entr4=Entry(fen,textvariable=debut)
entr5=Entry(fen,textvariable=final)
entr1.grid(row=0,column=1)
entr2.grid(row=1,column=1)
entr3.grid(row=2,column=1)
entr4.grid(row=3,column=1)
entr5.grid(row=4,column=1)
can=Canvas(fen,bg='white',height=700,width=700)
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Affichage',command=formerecepteur)
b1.grid(row=5,column=1)
b1.bind('<ButtonPress-1>',rayonreflechi)
fen.mainloop()
```



## Explosion de rayons (Modèle 2D)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def formerecepteur():
    "Contour du recepteur"
    coul='black'
    can.create_line(350-larg/2,350-haut/2,350+larg/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350-haut/2,350+larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350+larg/2,350+haut/2,350+diam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-diam/2,350+haut/2,350-larg/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-larg/2,350+haut/2,350-larg/2,350-haut/2,width=1,fill=coul)

def rayonreflechi(event):
    global effx,effy,deb,fi,larg,haut,diam,Pertes,Eff,emiss
    can.delete(ALL)
    larg=4*eval(largeur.get())
    haut=4*eval(hauteur.get())
    diam=4*eval(diametre.get())
    deb=eval(debut.get())
    fi=eval(final.get())
    emiss=eval(emissivite.get())
    formerecepteur()
    lstx=[]
    lsty=[]
    Trsx=0
    Trsy=0
    coloriage()
    for x in range(350-larg/2,350++larg/2,1):
        angle=pi/1800
        tot=0
        while angle<pi:
```



```
decide(x,angle)
#can.create_line(x,350-haut/2,x+haut/tan(angle),350+haut/2,width=1,fill=col)
angle=angle+pi/1800
if col=='green':
    tot=tot+1
effx=(1800-tot)*100/1800.
lstx.append(effx)
Trsx+=tot
#print Trsx
#print lstx
for j in range(0,larg,1):
    effx=lstx[j]
    couleur()
    can.create_line(350-larg/2+j,350-haut/2,350-larg/2+j,350-haut/2-35,fill=c)
for y in range(0,haut,1):
    angle=0
    tot=0
    while angle<pi/2:
        decide2(y,angle)
        #can.create_line(350+larg/2,350+haut/2-y,350+larg/2-
y*tan(angle),350+haut/2,width=1,fill=col)
        angle=angle+pi/1800
        if col=='green':
            tot=tot+1
        effy=(1800-tot)*100/1800.
        lsty.append(effy)
        Trsy+=tot
#print Trsy
T=(larg-2)*1798.0+2*(haut-2)*1798.0
Ts=Trsx+2*Trsy
Pertes=(Ts/T)
print T,Ts,Pertes
print "Sur un total de " +str(T)+" rayons, " +str(Ts)+ " sont sortis."
```



```
#print "Nous avons une perte de " +str()+ " % pour cette configuration"
#print lsty
for j in range(0,haut,1):
    effy=lsty[j]
    couleur2()
    can.create_line(350+larg/2,350+haut/2-j,350+larg/2+35,350+haut/2-j,fill=c)
```

```
def decide(a,b):
    global col
    if (a+haut/tan(b)>350-diam/2) and (a+haut/tan(b)<350+diam/2):
        col='green'
    else:
        col='red'
```

```
def decide2(c,d):
    global col
    if (100+larg-c*tan(d)>350-diam/2) and (100+larg-c*tan(d)<350+diam/2):
        col='green'
    else:
        col='red'
```

```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',
      '#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',
      '#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',
      '#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',
      '#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',
      '#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',
      '#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',
      '#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecd4','#27f4d0','#2afacb',
      '#2fffc7','#39ffc0','#3fffbf','#49ffb9','#4fffb0','#5affa9','#62ff90',
      '#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',
      '#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',
      '#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa000',
```



```
'#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',  
'#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',  
'#ffd400','#ffd200','#ffcd00','#ffcb00','#ffc800','#ffc200','#ffbc00',  
'#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',  
'#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',  
'#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',  
'#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',  
'#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',  
'#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',  
'#e40000','#e20000','#e00000']
```

```
def coloriage():
```

```
    global eff,deb,fi,ecar,ecarligne
```

```
    ecar=(fi-deb)/6.0
```

```
    ecarligne=int((353+larg/2-(353+larg/2-300))/6.0*100)/100.0
```

```
    for i in range(150):
```

```
        can.create_rectangle(350+larg/2+120,350+haut/2-2*i,380+larg/2+120,353+haut/2-  
2*i,width=0,fill=coul[i])
```

```
        for i in range(0,7,1):
```

```
            can.create_text(350+larg/2+80,350+haut/2+2-  
i*ecarligne,text=str(int((deb+i*ecar)*100)/100.0)+"%")
```

```
            can.create_line(350+larg/2+110,350+haut/2+2-i*ecarligne,350+larg/2+120,350+haut/2+2-  
i*ecarligne,width=1)
```

```
def couleur():
```

```
    # Fonction qui permet au programme de choisir la couleur en fonction
```

```
    # de la valeur de la mesure et de l'échelle choisie
```

```
    global c
```

```
    if effx<=deb:
```

```
        c=coul[0]
```

```
    elif effx>=fi:
```

```
        c=coul[149]
```

```
    else:
```

```

x1=deb
i=1
while i<149:
    if effx>x1 and effx<=x1+fi/149.:
        c=coul[i]
        i+=1
    x1+=(fi-deb)/149.

```

```

def couleur2():
    # Fonction qui permet au programme de choisir la couleur en fonction
    # de la valeur de la mesure et de l'échelle choisie
    global c
    if effy<=deb:
        c=coul[0]
    elif effy>=fi:
        c=coul[149]
    else:
        y1=deb
        i=1
        while i<149:
            if effy>y1 and effy<=y1+fi/149.:
                c=coul[i]
                i+=1
            y1+=(fi-deb)/149.

```

```

def calculpuissance(event):
    global temp,emiss
    temp=eval(temperature.get())
    emiss=eval(emissivite.get())
    formulepuissance()
    p=diam*emiss*sigma*temp**4/100000.0
    chaine.configure(text="Pertes radiatives du RS: " +str(int(Pperdue*100)/100.)+ " kW")
    chaine2.configure(text="Pertes radiatives en reception directe: " +str(int(p*100)/100.)+ " kW")

```



```
sigma=5.67*10**-8
```

```
def formulepuissance():
```

```
    global Eff,larg,haut,Pperdue
```

```
    Pparois=(emiss*sigma*temp**4)*(2*haut+larg)/100000.0
```

```
    Pperdue=Pertes*Pparois
```

```
    Eff=(100-Pperdue)*100/100.0
```

```
def graphe(event):
```

```
    global deb,fi,temp,Eff,emiss
```

```
    can.delete(ALL)
```

```
    deb=eval(debut.get())
```

```
    fi=eval(final.get())
```

```
    lsttemp=[]
```

```
    lsteff=[]
```

```
    temp=deb
```

```
    while temp<fi:
```

```
        formulepuissance()
```

```
        lsttemp.append(temp)
```

```
        lsteff.append(Eff)
```

```
        temp=temp+1
```

```
    print lsttemp
```

```
    print lsteff
```

```
    i=0
```

```
    while i<fi-deb-1:
```

```
        can.create_line(100,600,100+fi-deb,600,width=1)
```

```
        can.create_line(100,600,100,100,width=1)
```

```
        can.create_line(100+i,600-5*lsteff[i],100+i+1,600-5*lsteff[i+1],width=1,fill='red')
```

```
        i=i+1
```

```
fen=Tk()
```

```

fen.title("Lancer de rayons Model 2D")
largeur,hauteur,diametre,temperature,debut,final,emissivite=StringVar(),StringVar(),StringVar(),Str
ingVar(),StringVar(),StringVar(),StringVar()
txt1=Label(fen,text='Largeur:').grid(row=0,sticky=E)
txt2=Label(fen,text='Hauteur:').grid(row=1,sticky=E)
txt3=Label(fen,text='Diametre:').grid(row=2,sticky=E)
txt4=Label(fen,text='Debut:').grid(row=3,sticky=E)
txt5=Label(fen,text='Final:').grid(row=4,sticky=E)
txt6=Label(fen,text='Emissivite:').grid(row=5,sticky=E)
txt7=Label(fen,text='Temperature:').grid(row=7,sticky=E)
entr1=Entry(fen,textvariable=largeur)
entr2=Entry(fen,textvariable=hauteur)
entr3=Entry(fen,textvariable=diametre)
entr4=Entry(fen,textvariable=debut)
entr5=Entry(fen,textvariable=final)
entr6=Entry(fen,textvariable=emissivite)
entr7=Entry(fen,textvariable=temperature)
entr1.grid(row=0,column=1)
entr2.grid(row=1,column=1)
entr3.grid(row=2,column=1)
entr4.grid(row=3,column=1)
entr5.grid(row=4,column=1)
entr6.grid(row=5,column=1)
entr7.grid(row=7,column=1)
can=Canvas(fen,bg='white',height=1000,width=1000)
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Affichage',command=formerecepteur)
b1.grid(row=6,column=1)
b1.bind('<ButtonPress-1>',rayonreflechi)
b2=Button(fen,text='Efficacite',command=formulepuissance)
b2.grid(row=8,column=1)
b2.bind('<ButtonPress-1>',calculpuissance)
b3=Button(fen,text='Courbe',command=formulepuissance)

```



```
b3.grid(row=11,column=1)
b3.bind('<ButtonPress-1>',graphe)
chaine=Label(fen)
chaine.grid(row=9,column=1)
chaine2=Label(fen)
chaine2.grid(row=10,column=1)
fen.mainloop()
```



## Explosion de rayons sur le Parallélépipède (Modèle 3D)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def formefond():
    "Contour du recepteur"
    coul='black'
    can.create_line(350-larg/2,350-lon/2,350+larg/2,350-lon/2,width=1,fill=coul)
    can.create_line(350+larg/2,350-lon/2,350+larg/2,350+lon/2,width=1,fill=coul)
    can.create_line(350+larg/2,350+lon/2,350-larg/2,350+lon/2,width=1,fill=coul)
    can.create_line(350-larg/2,350+lon/2,350-larg/2,350-lon/2,width=1,fill=coul)

def modelfond(event):
    global efffond,teta,psi,diam,a,b,d,larg,lon,haut,deb,fi,pertes,totfond,eff
    can.delete(ALL)
    larg=4*eval(largeur.get())
    lon=4*eval(longueur.get())
    haut=4*eval(hauteur.get())
    diam=4*eval(diametre.get())
    deb=eval(debut.get())
    fi=eval(final.get())
    formefond()
    listea=[]
    listeb=[]
    listefffond=[]
    totfond=0
    coloriagefond()
    for i in range(larg):
        listefffond.append([])
    for a in range(0,larg,1):
        for b in range(0,lon,1):
            vert=0
```



```
rouge=0
teta=91*pi/180
while teta<179*pi/180:
    psi=0
    while psi<2*pi:
        xinter=-haut*sin(teta)*cos(psi)/cos(teta)+larg/2-a
        yinter=-haut*sin(teta)*sin(psi)/cos(teta)-lon/2+b
        inter=sqrt(xinter**2+yinter**2)
        if inter<diam/2:
            vert+=1
        else:
            rouge+=1
        efffond=100-vert/324.0*100
        psi=psi+pi/18
    teta=teta+pi/18
totfond+=vert
listea.append(a)
listeb.append(b)
listefffond[a].append(efffond)
print totfond
for a in range(0,larg,1):
    for b in range(0,lon,1):
        efffond=listefffond[a][b]
        couleurfond()
        can.create_rectangle(349.5-larg/2+a,349.5-lon/2+b,350.5-larg/2+a,350.5-
lon/2+b,fill=c,outline=c)

def formelaterale():
    "Contour du recepteur"
    coul='black'
    can.create_line(350-lon/2,350-haut/2,350+lon/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+lon/2,350-haut/2,350+lon/2,350+haut/2,width=1,fill=coul)
```



```
can.create_line(350+lon/2,350+haut/2,350-lon/2,350+haut/2,width=1,fill=coul)
can.create_line(350-lon/2,350+haut/2,350-lon/2,350-haut/2,width=1,fill=coul)
```

```
def modellaterale(event):
    global efflaterale,teta,psi,diam,a,d,larg,lon,haut,deb,fi,totlaterale,eff
    can.delete(ALL)
    larg=4*eval(largeur.get())
    lon=4*eval(longueur.get())
    haut=4*eval(hauteur.get())
    diam=4*eval(diametre.get())
    deb=eval(debut.get())
    fi=eval(final.get())
    formelaterale()
    listea=[]
    listed=[]
    listefflaterale=[]
    totlaterale=0
    colorielaterale()
    for i in range(larg):
        listefflaterale.append([])
    for a in range(0,larg,1):
        for d in range(0,haut,1):
            vert=0
            rouge=0
            teta=91*pi/180
            while teta<179*pi/180:
                psi=0
                while psi<pi:
                    xinter=-d*tan(teta)*cos(psi)+larg/2-a
                    yinter=-d*tan(teta)*sin(psi)-lon/2
                    inter=sqrt(xinter**2+yinter**2)
                    if inter<diam/2:
                        vert+=1
```

```

else:
    rouge+=1
    efflaterale=100-vert/324.0*100
    psi=psi+pi/18
    teta=teta+pi/18
    totlaterale+=vert
    listea.append(a)
    listed.append(d)
    listefflaterale[a].append(efflaterale)
print totlaterale
for a in range(0,larg,1):
    for d in range(0,haut,1):
        efflaterale=listefflaterale[a][d]
        couleurlaterale()
        can.create_rectangle(349.5-lon/2+a,349.5-haut/2+d,350.5-lon/2+a,350.5-
haut/2+d,fill=c,outline=c)

def couleurfond():
    global c
    if efffond<=deb:
        c=coul[0]
    elif efffond>=fi:
        c=coul[149]
    else:
        x1=deb
        i=1
        while i<149:
            if efffond>x1 and efffond<=x1+fi/149.:
                c=coul[i]
                i+=1
                x1+=(fi-deb)/149.

def couleurlaterale():

```



global c

if efflaterale<=deb:

    c=coul[0]

elif efflaterale>=fi:

    c=coul[149]

else:

    x1=deb

    i=1

    while i<149:

        if efflaterale>x1 and efflaterale<=x1+fi/149.:

            c=coul[i]

            i+=1

        x1+=(fi-deb)/149.

def coloriageilaterale():

    global eff,deb,fi,ecar,ecarligne

    ecar=(fi-deb)/6.0

    ecarligne=int((353+haut/2-(353+haut/2-300))/6.0\*100)/100.0

    for i in range(150):

        can.create\_rectangle(350+lon/2+120,350+haut/2-2\*i,380+lon/2+120,353+haut/2-2\*i,width=0,fill=coul[i])

        for i in range(0,7,1):

            can.create\_text(350+lon/2+80,350+haut/2+2-i\*ecarligne,text=str(int((deb+i\*ecar)\*100)/100.0)+"%")

            can.create\_line(350+lon/2+110,350+haut/2+2-i\*ecarligne,350+lon/2+120,350+haut/2+2-i\*ecarligne,width=1)

def coloriagefond():

    global eff,deb,fi,ecar,ecarligne

    ecar=(fi-deb)/6.0

    ecarligne=int((353+larg/2-(353+larg/2-300))/6.0\*100)/100.0

    for i in range(150):



```
can.create_rectangle(350+lon/2+120,350+larg/2-2*i,380+lon/2+120,353+larg/2-
2*i,width=0,fill=coul[i])
for i in range(0,7,1):
    can.create_text(350+lon/2+80,350+larg/2+2-
i*ecarligne,text=str(int(((deb+i*ecar)*100)/100.0)+"%")
    can.create_line(350+lon/2+110,350+larg/2+2-i*ecarligne,350+lon/2+120,350+larg/2+2-
i*ecarligne,width=1)
```

```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',
'#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',
'#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',
'#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',
'#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',
'#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',
'#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',
'#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecd4','#27f4d0','#2afacb',
'#2fffc7','#39ffc0','#3fffbf','#49ffb9','#4fffb0','#5affa9','#62ff90',
'#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',
'#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',
'#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa00',
'#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',
'#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',
'#ffd400','#ffd200','#ffcd00','#ffcb00','#ffc800','#ffc200','#ffbc00',
'#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',
'#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',
'#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',
'#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',
'#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',
'#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',
'#e40000','#e20000','#e00000']
```



```
def calculpuissance(event):  
    global temp,emiss,pperdue  
    temp=273.15+eval(temperature.get())  
    emiss=eval(emissivite.get())  
    formulepuissance()  
    p=pi*(diam**2/4)*emiss*sigma*temp**4/10000000.0  
    chaine.configure(text="Pertes radiatives du RS: " +str(pperdue)+ " kW")  
    chaine2.configure(text="Pertes radiatives en reception directe: " +str(int(p*100)/100.)+ " kW")  
    can.create_text(250,100,text="L'efficacite globale du RS est egal a " +str(eff))
```

```
sigma=5.67*10**-8
```

```
def formulepuissance():  
    global eff,larg,lon,haut,pertes,totfond,totlaterale,pperdue  
    pertes=int((((totfond+4*totlaterale)/((324*lon*larg+324*4*larg*haut)*1.0))*100)/100.0  
    pparois=(emiss*sigma*temp**4)*(lon*larg+4*lon*haut)/10000000.0  
    pperdue=pertes*pparois  
    eff=(100-pperdue)*100/100.0  
    print pertes  
    print pperdue  
    print eff
```

```
fen=Tk()  
fen.title("Modelisation Cube 3D")  
largeur,longueur,hauteur,diametre,debut,final,emissivite,temperature=StringVar(),StringVar(),String  
Var(),StringVar(),StringVar(),StringVar(),StringVar(),StringVar()  
txt1=Label(fen,text='Largeur:').grid(row=0,sticky=E)  
txt2=Label(fen,text='Longueur:').grid(row=1,sticky=E)  
txt3=Label(fen,text='Hauteur:').grid(row=2,sticky=E)  
txt4=Label(fen,text='Diametre:').grid(row=3,sticky=E)  
txt5=Label(fen,text='Debut:').grid(row=4,sticky=E)  
txt6=Label(fen,text='Final:').grid(row=5,sticky=E)  
txt7=Label(fen,text='Emissivite:').grid(row=8,sticky=E)
```



```
txt8=Label(fen,text='Temperature:').grid(row=9,sticky=E)
entr1=Entry(fen,textvariable=largeur)
entr2=Entry(fen,textvariable=longueur)
entr3=Entry(fen,textvariable=hauteur)
entr4=Entry(fen,textvariable=diametre)
entr5=Entry(fen,textvariable=debut)
entr6=Entry(fen,textvariable=final)
entr7=Entry(fen,textvariable=emissivite)
entr8=Entry(fen,textvariable=temperature)
entr1.grid(row=0,column=1)
entr2.grid(row=1,column=1)
entr3.grid(row=2,column=1)
entr4.grid(row=3,column=1)
entr5.grid(row=4,column=1)
entr6.grid(row=5,column=1)
entr7.grid(row=8,column=1)
entr8.grid(row=9,column=1)
can=Canvas(fen,bg='white',height=700,width=700)
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Interne',command=formefond)
b1.grid(row=6,column=1)
b1.bind('<ButtonPress-1>',modelfond)
b2=Button(fen,text='Laterale',command=formelaterale)
b2.grid(row=7,column=1)
b2.bind('<ButtonPress-1>',modellaterale)
b3=Button(fen,text='Efficacite',command=formulepuissance)
b3.grid(row=10,column=1)
b3.bind('<ButtonPress-1>',calculpuissance)
chaine=Label(fen)
chaine.grid(row=11,column=1)
chaine2=Label(fen)
chaine2.grid(row=12,column=1)
fen.mainloop()
```



## Face latérale du cylindre (Modèle 3D – Facteur de forme)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def contour():
    "Surface laterale du recepteur"
    coul='black'
    can.create_line(350-gdiam/2,350-haut/2,350+gdiam/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+gdiam/2,350-haut/2,350+gdiam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350+gdiam/2,350+haut/2,350+pdiam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-pdiam/2,350+haut/2,350-gdiam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-gdiam/2,350+haut/2,350-gdiam/2,350-haut/2,width=1,fill=coul)

def model(event):
    global haut,gdiam,pdiam,deb,fi,z,eff,x,y,r,ff,pertes,sommeff
    can.delete(ALL)
    haut=eval(hauteur.get())
    gdiam=eval(granddiametre.get())
    pdiam=eval(petitdiametre.get())
    deb=eval(debut.get())
    fi=eval(final.get())
    contour()
    coloriage()
    sommeff=0
    for z in range(1,haut,1):
        y=2*z/(gdiam*1.0)
        r=gdiam/(gdiam*1.0)
        x=1+y**2+r**2
        ff=(y/2.0)*(x/((x**2-4*(r**2))**(1/2.0))-1)
        sommeff+=ff
        eff=100-100*ff
        couleur()
```

```
can.create_line(350+gdiam/2,350+haut/2-z,385+gdiam/2,350+haut/2-z,fill=c)
```

```
lost=int((sommeff/(haut*1.0))*10000)/100.0
```

```
pertes=sommeff/(haut*1.0)
```

```
#print sommeff
```

```
print str(lost)+" %"
```

```
def couleur():
```

```
    global c
```

```
    if eff<=deb:
```

```
        c=coul[0]
```

```
    elif eff>=fi:
```

```
        c=coul[149]
```

```
    else:
```

```
        x1=deb
```

```
        i=1
```

```
        while i<149:
```

```
            if eff>x1 and eff<=x1+fi/149.:
```

```
                c=coul[i]
```

```
                i+=1
```

```
            x1+=(fi-deb)/149.
```

```
def coloriage():
```

```
    global eff,deb,fi,ecar,ecarligne
```

```
    ecar=(fi-deb)/6.0
```

```
    ecarligne=int((353+gdiam/2-(353+gdiam/2-450))/6.0*100)/100.0
```

```
    for i in range(150):
```

```
        can.create_rectangle(350+gdiam/2+100,350+haut/2-3*i,380+gdiam/2+100,353+haut/2-3*i,width=0,fill=coul[i])
```

```
    for i in range(0,7,1):
```

```
        can.create_text(350+gdiam/2+80,350+haut/2+3-i*ecarligne,text=int((deb+i*ecar)*100)/100.0)
```

```
        can.create_line(350+gdiam/2+90,350+haut/2+3-i*ecarligne,350+gdiam/2+100,350+haut/2+3-i*ecarligne,width=1)
```

```
def calculpuissance(event):
    global temp,emiss,pertes,pperdue
    temp=273.15+eval(temperature.get())
    emiss=eval(emissivite.get())
    formulepuissance()
    chaine.configure(text="Pertes radiatives du RS: " +str(int(pperdue*100)/100.)+ " kW")
```

```
sigma=5.67*10**-8
```

```
def formulepuissance():
    global Eff,haut,gdiam,pperdue,pertes
    pparois=(emiss*sigma*temp**4)*(pi*gdiam*haut)/10000000.0
    pperdue=pertes*pparois
    Eff=(100-pperdue)*100/100.0
```

```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',
      '#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',
      '#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',
      '#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',
      '#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',
      '#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',
      '#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',
      '#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecdd','#27f4d0','#2afacb',
      '#2fffc7','#39ffc0','#3fffbf','#49ffb9','#4fffb0','#5affa9','#62ff90',
      '#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',
      '#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',
      '#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa000',
      '#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',
      '#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',
      '#ffd400','#ffd200','#ffcd00','#ffcb00','#ffc800','#ffc200','#ffbc00',
      '#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',
```



```
'#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',  
'#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',  
'#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',  
'#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',  
'#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',  
'#e40000','#e20000','#e00000']
```

```
fen=Tk()  
fen.title("Modelisation Cylindre 3D")  
hauteur,granddiametre,petitdiametre,debut,final,emissivite,temperature=StringVar(),StringVar(),StringVar(),StringVar(),StringVar(),StringVar(),StringVar()  
txt1=Label(fen,text='Hauteur:').grid(row=0,sticky=E)  
txt2=Label(fen,text='Diametre sup:').grid(row=1,sticky=E)  
txt3=Label(fen,text='Diametre inf:').grid(row=2,sticky=E)  
txt4=Label(fen,text='Debut:').grid(row=3,sticky=E)  
txt5=Label(fen,text='Final:').grid(row=4,sticky=E)  
txt6=Label(fen,text='Emissivite:').grid(row=6,sticky=E)  
txt7=Label(fen,text='Temperature:').grid(row=7,sticky=E)  
entr1=Entry(fen,textvariable=hauteur)  
entr2=Entry(fen,textvariable=granddiametre)  
entr3=Entry(fen,textvariable=petitdiametre)  
entr4=Entry(fen,textvariable=debut)  
entr5=Entry(fen,textvariable=final)  
entr6=Entry(fen,textvariable=emissivite)  
entr7=Entry(fen,textvariable=temperature)  
entr1.grid(row=0,column=1)  
entr2.grid(row=1,column=1)  
entr3.grid(row=2,column=1)  
entr4.grid(row=3,column=1)  
entr5.grid(row=4,column=1)  
entr6.grid(row=6,column=1)  
entr7.grid(row=7,column=1)  
can=Canvas(fen,bg='white',height=700,width=700)
```



```
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Affichage',command=contour)
b1.grid(row=5,column=1)
b1.bind('<KeyPress-1>',model)
b2=Button(fen,text='Pertes',command=formulepuissance)
b2.grid(row=8,column=1)
b2.bind('<KeyPress-1>',calculpuissance)
chaine=Label(fen)
chaine.grid(row=9,column=1)
fen.mainloop()
```

## Face latérale du cylindre (Modèle 3D – Explosion de rayons)

```
#-*-coding:Utf-8-*-
from Tkinter import *
from math import *
def contour():
    "Surface laterale du recepteur"
    coul='black'
    can.create_line(350-gdiam/2,350-haut/2,350+gdiam/2,350-haut/2,width=1,fill=coul)
    can.create_line(350+gdiam/2,350-haut/2,350+gdiam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350+gdiam/2,350+haut/2,350+pdiam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-pdiam/2,350+haut/2,350-gdiam/2,350+haut/2,width=1,fill=coul)
    can.create_line(350-gdiam/2,350+haut/2,350-gdiam/2,350-haut/2,width=1,fill=coul)

def model(event):
    global haut,gdiam,pdiam,deb,fi,z,eff,x,y,r,ff,pertes,tot,lost
    can.delete(ALL)
    haut=4*eval(hauteur.get())
    gdiam=4*eval(grandiametre.get())
    pdiam=4*eval(petitdiametre.get())
    deb=eval(debut.get())
    fi=eval(final.get())
    contour()
    listeff=[]
    coloriage()
    lost=0
    tot=0
    for z in range(0,haut,1):
        vert=0
        rouge=0
        teta=91*pi/180
        while teta<179*pi/180:
            psi=0
```

```

while psi<pi:
    xinter=-z*tan(teta)*cos(psi)
    yinter=-z*tan(teta)*sin(psi)-gdiam/2
    inter=sqrt(xinter**2+yinter**2)
    if inter<pdiam/2:
        vert+=1
    else:
        rouge+=1
    #eff=100-vert/324.0*100
    psi=psi+pi/180
    teta=teta+pi/180
    eff=100-vert/32400.0*100
    tot+=vert
    lost=int((tot/(32400.0*haut))*10000)/100.0
    listeff.append(eff)
#print tot
print str(lost)+" %"
for z in range(1,haut,1):
    eff=listeff[z]
    couleur()
    can.create_line(350+gdiam/2,350+haut/2-z,385+gdiam/2,350+haut/2-z,fill=c)

def couleur():
    global c
    if eff<=deb:
        c=coul[0]
    elif eff>=fi:
        c=coul[149]
    else:
        x1=deb
        i=1
        while i<149:
            if eff>x1 and eff<=x1+fi/149.:

```

```
c=coul[i]
i+=1
x1+=(fi-deb)/149.
```

```
def coloriage():
```

```
    global eff,deb,fi,ecar,ecarligne
    ecar=(fi-deb)/6.0
    ecarligne=int((353+gdiam/2-(353+gdiam/2-300))/6.0*100)/100.0
    for i in range(150):
        can.create_rectangle(350+gdiam/2+120,350+haut/2-2*i,380+gdiam/2+120,353+haut/2-
2*i,width=0,fill=coul[i])
        for i in range(0,7,1):
            can.create_text(350+gdiam/2+80,350+haut/2+2-
i*ecarligne,text=str(int(((deb+i*ecar)*100)/100.0)+"%")
            can.create_line(350+gdiam/2+110,350+haut/2+2-
i*ecarligne,350+gdiam/2+120,350+haut/2+2-i*ecarligne,width=1)
```

```
def calculpuissance(event):
```

```
    global temp,emiss,perdes,pperdue,tot
    temp=273.15+eval(temperature.get())
    emiss=eval(emissivite.get())
    formulepuissance()
    chaine.configure(text="Pertes radiatives du RS: " +str(int(pperdue*100)/100.)+ " kW")
```

```
sigma=5.67*10**-8
```

```
def formulepuissance():
```

```
    global Eff,haut,gdiam,pdiam,pperdue,perdes,tot
    perdes=int((tot/((32400*haut)*1.0))*100)/100.0
    pparois=(emiss*sigma*temp**4)*(pi*gdiam*haut)/10000000.0
    pperdue=perdes*pparois
    Eff=(100-pperdue)*100/100.0
```



```
coul=['#000090','#000095','#000099','#0000a4','#0000a9','#0000b2','#0000b6',
      '#0000bf','#0000c5','#0000ce','#0000d4','#0000d9','#0000df','#0000e4',
      '#0000e9','#0000ef','#0000f4','#0000f9','#0000ff','#0005ff','#0010ff',
      '#0018ff','#0020ff','#0028ff','#0030ff','#0038ff','#0040ff','#0048ff',
      '#0050ff','#0058ff','#0060ff','#0068ff','#0070ff','#0078ff','#0080ff',
      '#0088ff','#0090ff','#0098ff','#00a0ff','#00a8ff','#00acff','#00afff',
      '#00b5ff','#00b9ff','#00bcff','#00bfff','#00c4ff','#05c9fb','#09ccf6',
      '#0fd2f0','#14d9dd','#18deda','#1be8d7','#1fecd4','#27f4d0','#2afacb',
      '#2fffc7','#39ffc0','#3fffbf','#49ffb9','#4fffb0','#5affa9','#62ff90',
      '#6dff80','#75ff70','#80ff60','#8dff60','#99ff60','#a0ff60','#b0ff60',
      '#c0ff60','#d0ff60','#d8ff50','#dfff40','#e4ff30','#ecff20','#f2ff10',
      '#faff10','#ffff00','#ffff00','#ffff00','#ffff00','#ffff00','#ffa00',
      '#fff600','#fff400','#ffed00','#ffe800','#ffe600','#ffe400','#ffe200',
      '#ffe100','#ffdf00','#ffdd00','#ffdb00','#ffd900','#ffd700','#ffd600',
      '#ffd400','#ffd200','#ffcd00','#ffcb00','#ffc800','#ffc200','#ffbc00',
      '#ffb600','#ffb000','#ffaa00','#ffa500','#ff9e00','#ff9800','#ff9300',
      '#ff8d00','#ff8800','#ff8300','#ff7c00','#ff7800','#ff7000','#ff6b00',
      '#ff6500','#ff6000','#ff5c00','#ff5600','#ff5000','#ff4c00','#ff4600',
      '#ff4000','#ff3c00','#ff3700','#ff3100','#ff2c00','#ff2700','#ff2000',
      '#fb1b00','#f71800','#f71000','#f70b00','#f70800','#f50000','#f30000',
      '#f10000','#f00000','#ee0000','#ec0000','#ea0000','#e80000','#e60000',
      '#e40000','#e20000','#e00000']
```

```
fen=Tk()
```

```
fen.title("Modelisation Cylindre 3D")
```

```
hauteur,granddiametre,petitdiametre,debut,final,emissivite,temperature=StringVar(),StringVar(),StringVar(),StringVar(),StringVar(),StringVar(),StringVar()
```

```
txt1=Label(fen,text='Hauteur:').grid(row=0,sticky=E)
```

```
txt2=Label(fen,text='Diametre sup:').grid(row=1,sticky=E)
```

```
txt3=Label(fen,text='Diametre inf:').grid(row=2,sticky=E)
```

```
txt4=Label(fen,text='Debut:').grid(row=3,sticky=E)
```



```
txt5=Label(fen,text='Final:').grid(row=4,sticky=E)
txt6=Label(fen,text='Emissivite:').grid(row=6,sticky=E)
txt7=Label(fen,text='Temperature:').grid(row=7,sticky=E)
entr1=Entry(fen,textvariable=hauteur)
entr2=Entry(fen,textvariable=granddiametre)
entr3=Entry(fen,textvariable=petitdiametre)
entr4=Entry(fen,textvariable=debut)
entr5=Entry(fen,textvariable=final)
entr6=Entry(fen,textvariable=emissivite)
entr7=Entry(fen,textvariable=temperature)
entr1.grid(row=0,column=1)
entr2.grid(row=1,column=1)
entr3.grid(row=2,column=1)
entr4.grid(row=3,column=1)
entr5.grid(row=4,column=1)
entr6.grid(row=6,column=1)
entr7.grid(row=7,column=1)
can=Canvas(fen,bg='white',height=700,width=700)
can.grid(row=0,column=2,rowspan=25,padx=10,pady=10)
b1=Button(fen,text='Affichage',command=contour)
b1.grid(row=5,column=1)
b1.bind('<ButtonPress-1>',model)
b2=Button(fen,text='Pertes',command=formulepuissance)
b2.grid(row=8,column=1)
b2.bind('<ButtonPress-1>',calculpuissance)
chaine=Label(fen)
chaine.grid(row=9,column=1)
fen.mainloop()
```