

IIA project – Simulations of Mode Coupling

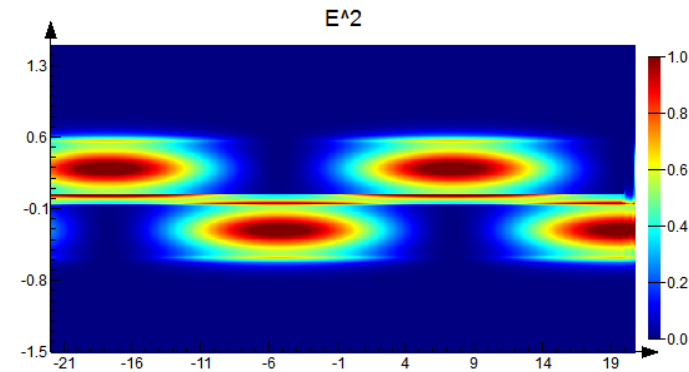
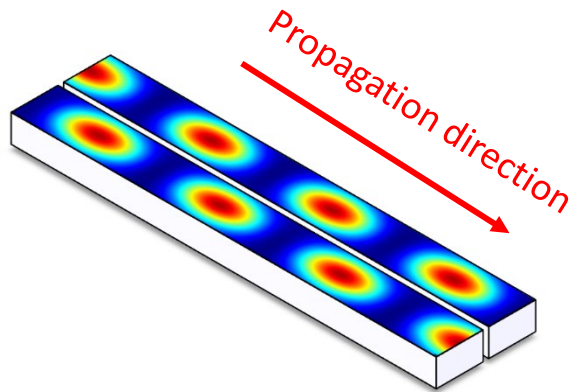
Prof. Qixiang Cheng, Dr. Chunhui Yao

Basics of Mode Coupling

When two waveguides are brought close to each other, a phenomenon called **mode coupling** occurs, which means that optical energy periodically transfers between the two waveguides.

This happens because:

- The **evanescent fields** of the two waveguides extend into the surrounding space, and when placed close enough, **they overlap and interact**.
- This interaction enables **periodic energy transfer** between the two waveguides, where energy oscillates back and forth between the waveguides (like beating)



Basics of Mode Coupling

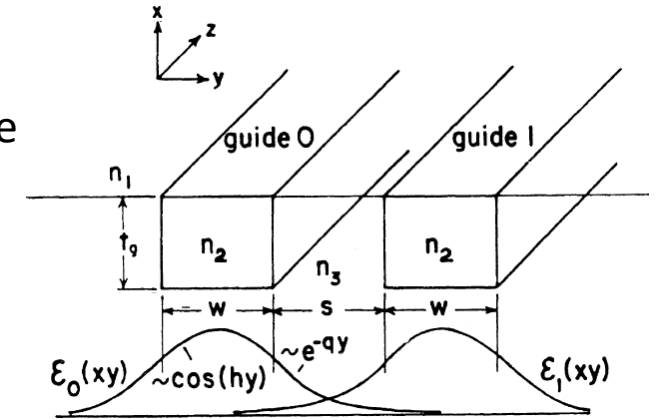
Coupled-Mode Theory of Synchronous Coupling

Coupling between two identical lossless waveguides ($\beta = \beta_0 = \beta_1$), and the field of the propagating mode is denoted, as:

$$\bar{E}(x, y, z) = A(z) \bar{E}(x, y)$$

where $A(z)$ is the complex amplitude of the field, such that the mode power equals

$$P(z) = |A(z)|^2 = A(z)A^*(z)$$



The mode coupling can be described by the general **coupled mode equations** for the amplitudes of the two modes:

$$\begin{cases} \frac{dA_0(z)}{dz} = -i\beta A_0(z) - i\kappa A_1(z) \\ \frac{dA_1(z)}{dz} = -i\beta A_1(z) - i\kappa A_0(z) \end{cases} \quad (1.11)$$

where β_0 and β_1 are the mode propagation constants, and κ is the **coupling coefficient** between modes, which quantifies the strength of interaction between the two waveguides: the larger κ , the faster the energy transfer between them.



Basics of Mode Coupling

Assume that the light is all within Waveguide 0 at the position $z=0$, we have:

$$A_0(0) = 1 \quad \text{and} \quad A_1(0) = 0$$

Thus, the solutions of the coupled mode equations are:

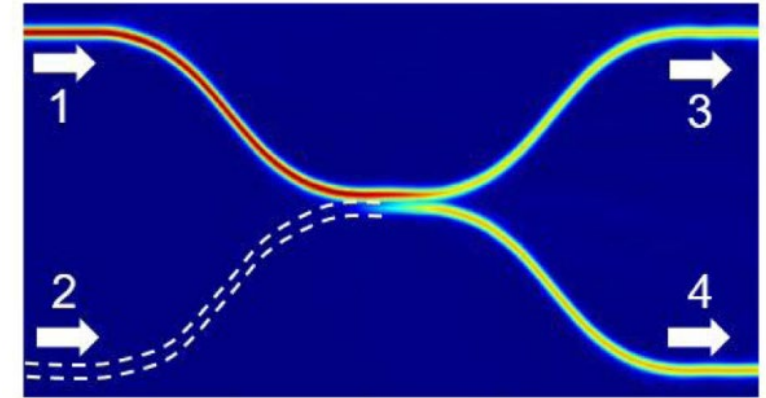
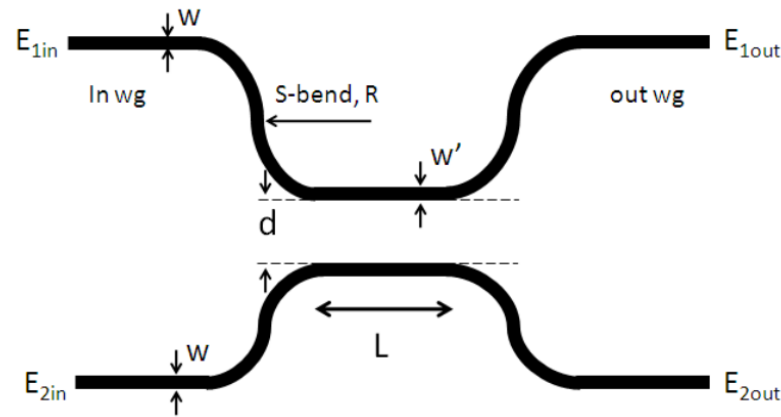
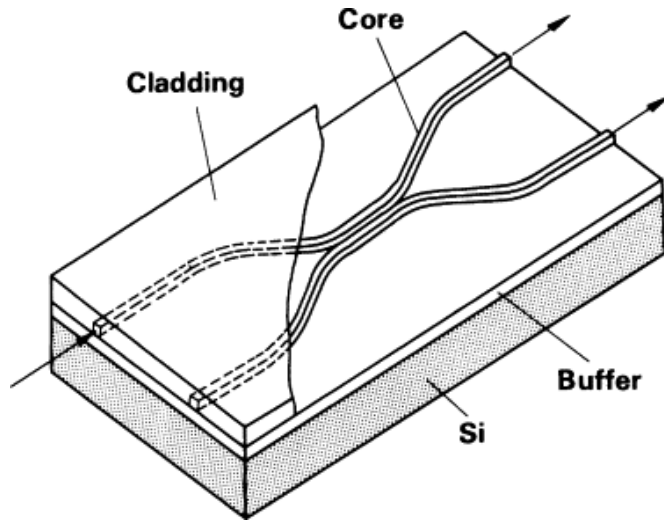
$$\left\{ \begin{array}{l} \frac{dA_0(z)}{dz} = -i\beta A_0(z) - i\kappa A_1(z) \\ \frac{dA_1(z)}{dz} = -i\beta A_1(z) - i\kappa A_0(z) \end{array} \right. \xrightarrow{\text{red arrow}} \left\{ \begin{array}{l} A_0(z) = \cos(\kappa z) e^{-i\beta z} \\ A_1(z) = -i \sin(\kappa z) e^{-i\beta z} \end{array} \right. \xrightarrow{\text{red arrow}} \left\{ \begin{array}{l} P_0(z) = A_0(z)A_0^*(z) = \cos^2(\kappa z) \\ P_1(z) = A_1(z)A_1^*(z) = \sin^2(\kappa z) \end{array} \right. \quad \text{power flow in waveguides}$$

It can be seen that the **coupling length** L necessary for complete transfer of power from one guide to the other is given by:

$$L = \frac{\pi}{2\kappa} + \frac{m\pi}{\kappa} \quad \text{where } m = 0, 1, 2, 3 \dots$$



Directional Coupler



By adjusting the waveguide width, coupling region length, gap, and other geometrical parameters, directional couplers can be engineered to achieve various power-splitting ratios.



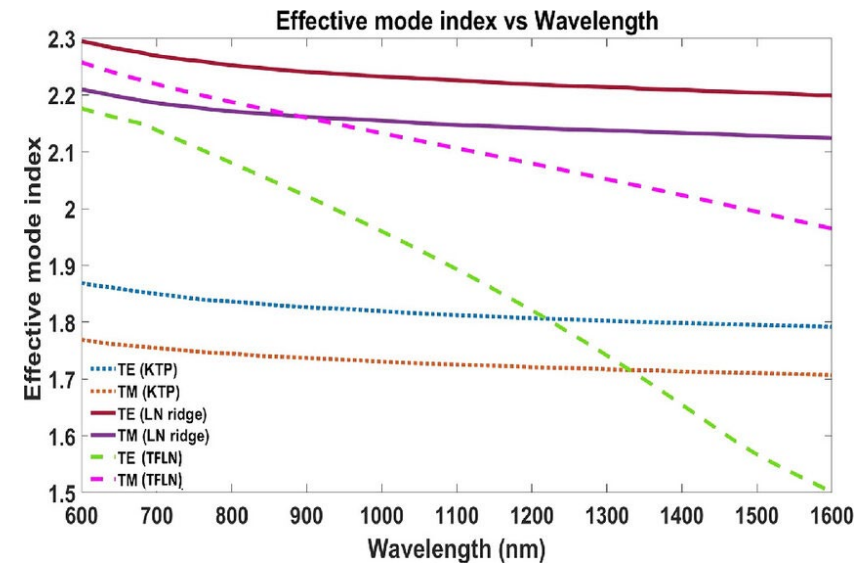
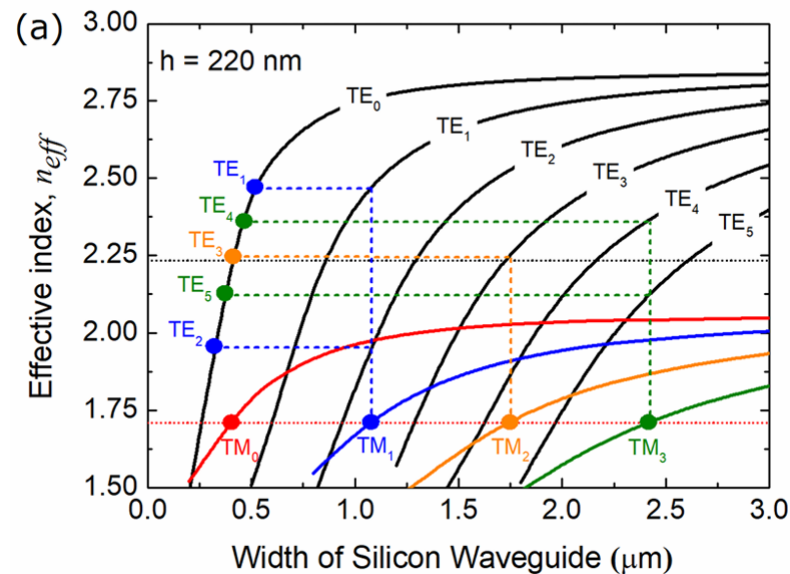
Coupling between asymmetrical waveguides

Mode Effective Index

$$\begin{cases} \frac{dA_0(z)}{dz} = -i\beta A_0(z) - i\kappa A_1(z) \\ \frac{dA_1(z)}{dz} = -i\beta A_1(z) - i\kappa A_0(z) \end{cases}$$

where β is the propagation constant, $\beta = kn_{\text{eff}} = \omega n_{\text{eff}}/c$

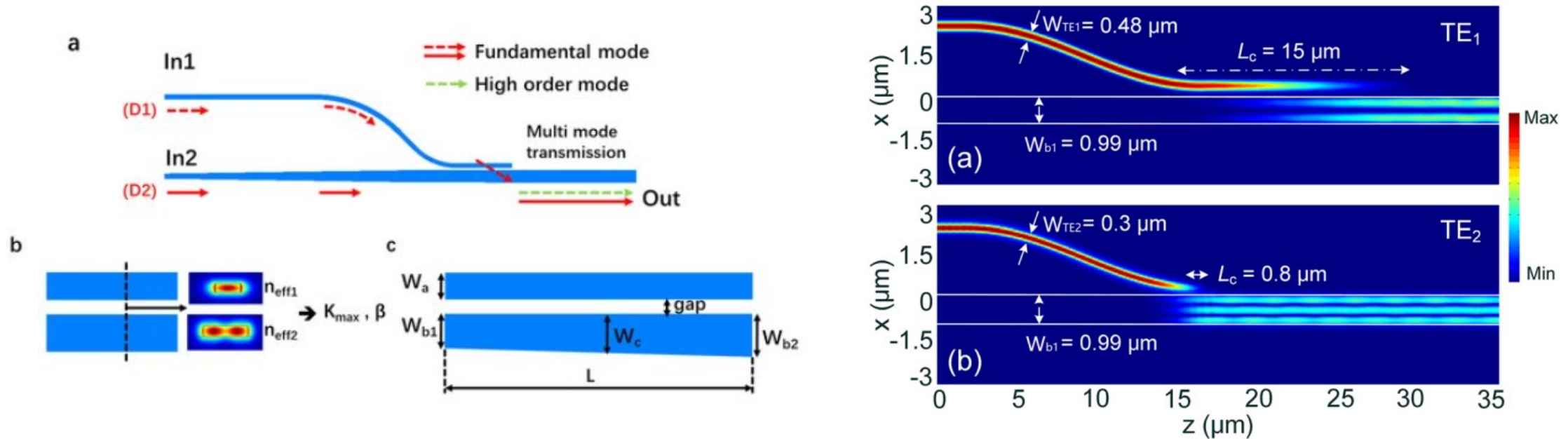
Effective refractive index experienced by the guided mode



Varies over Mode Order, Waveguide geometry, Wavelength, etc...



Coupling between asymmetrical waveguides



By tuning the waveguide width, the effective indices of different mode orders can be matched, enabling the structure to function as a **mode order converter**, facilitating transitions between guided modes (e.g., $TE_0 \rightarrow TE_1$).



For remote control

<https://help.eng.cam.ac.uk/network/remote-access/pc-access-from-windows/>

<https://www.chiark.greenend.org.uk/~sgtatham/putty/>

Download Putty



The screenshot shows the University of Cambridge Engineering Department website. The header includes the university logo and navigation links. The main content area is titled "Computing help and support" and "Office computer access from Windows". A sidebar on the left lists various support topics. The main content area provides instructions for configuring remote access, including a note about security and a link to the IT Helpdesk.

UNIVERSITY OF CAMBRIDGE

Study at Cambridge | About the University | Research at Cambridge

Quick Links | Search

Department of Engineering / Computing help and support / Network / Remote access / Office computer access from Windows

DEPARTMENT OF ENGINEERING

Computing help and support

Home | For staff and postgraduates | For undergraduates | Rules and Policies | Contact us

Office computer access from Windows

Computing help and support

Network

Remote access

Office computer access from Mac

Office computer access from Windows

Teaching System Access from Linux

Teaching System Access from MacOS

Teaching System Access from Windows

Troubleshooting

Technical information and advanced

Configuring your Engineering machine to allow Remote Desktop access

Note: These instructions should be carried out on your Engineering machine.

The following links provide answers to frequently asked questions about Remote Desktop connectivity on Windows computers. Each has a section on enabling the Remote Desktop feature:

- Windows 8.1
- Windows 10

For increased security you may want to change the default port that Remote Desktop connections uses. If you would like to do that you can follow the instructions [here](#)

Setting up a Secure Shell connection

Note: These instructions should be carried out on the machine you will be using to connect to your Engineering machine.

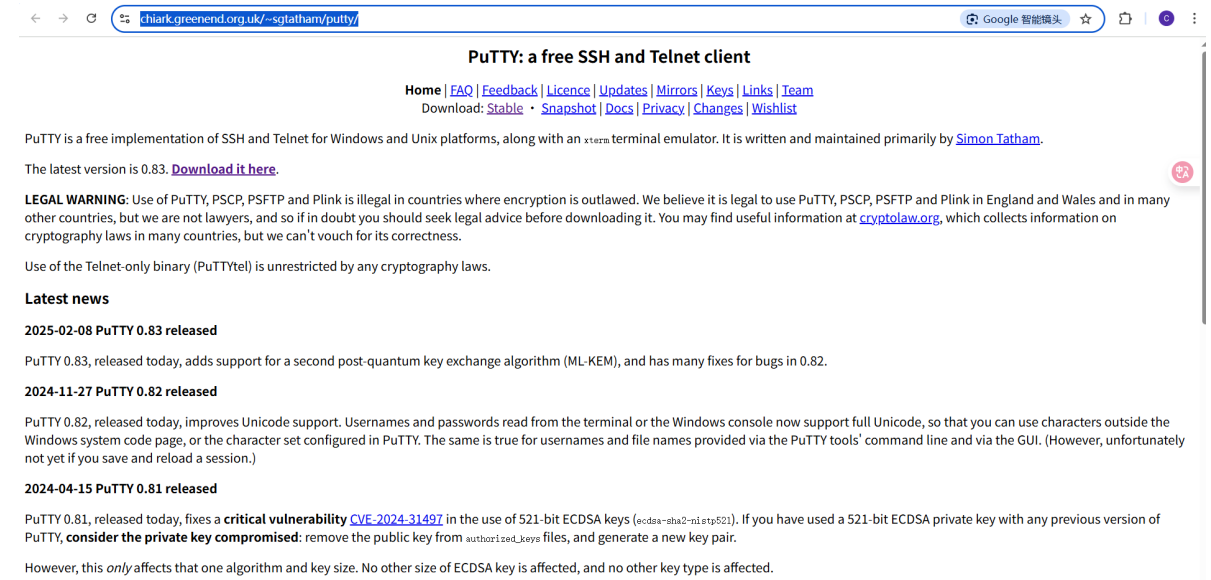
The following procedure requires a Secure Shell (SSH) client. The latest versions of Windows 10 include an OpenSSH client, which can be installed via "Manage optional features" from the Start Menu / Settings.

Log in with Raven (ICS staff)

IT Helpdesk

E-mail: helpdesk@eng.cam.ac.uk

Tel: +441223 (3)32686



The screenshot shows the PuTTY website. The header includes the title "PuTTY: a free SSH and Telnet client" and a navigation menu. The main content area provides information about the latest version (0.83) and a list of recent releases (2025-02-08, 2024-11-27, 2024-04-15). It also includes a legal warning and a note about the private key compromised.

PuTTY: a free SSH and Telnet client

Home | [FAQ](#) | [Feedback](#) | [Licence](#) | [Updates](#) | [Mirrors](#) | [Keys](#) | [Links](#) | [Team](#)

Download: [Stable](#) • [Snapshot](#) | [Docs](#) | [Privacy](#) | [Changes](#) | [Wishlist](#)

PuTTY is a free implementation of SSH and Telnet for Windows and Unix platforms, along with an `xterm` terminal emulator. It is written and maintained primarily by [Simon Tatham](#).

The latest version is 0.83. [Download it here](#).

LEGAL WARNING: Use of PuTTY, PSCP, PSFTP and Plink is illegal in countries where encryption is outlawed. We believe it is legal to use PuTTY, PSCP, PSFTP and Plink in England and Wales and in many other countries, but we are not lawyers, and so if in doubt you should seek legal advice before downloading it. You may find useful information at [cryptolaw.org](#), which collects information on cryptography laws in many countries, but we can't vouch for its correctness.

Use of the Telnet-only binary (PuTTYtel) is unrestricted by any cryptography laws.

Latest news

2025-02-08 PuTTY 0.83 released

PuTTY 0.83, released today, adds support for a second post-quantum key exchange algorithm (ML-KEM), and has many fixes for bugs in 0.82.

2024-11-27 PuTTY 0.82 released

PuTTY 0.82, released today, improves Unicode support. Usernames and passwords read from the terminal or the Windows console now support full Unicode, so that you can use characters outside the Windows system code page, or the character set configured in PuTTY. The same is true for usernames and file names provided via the PuTTY tools' command line and via the GUI. (However, unfortunately not yet if you save and reload a session.)

2024-04-15 PuTTY 0.81 released

PuTTY 0.81, released today, fixes a **critical vulnerability** [CVE-2024-31497](#) in the use of 521-bit ECDSA keys (`ecdsa-sha2-nistp521`). If you have used a 521-bit ECDSA private key with any previous version of PuTTY, **consider the private key compromised**: remove the public key from `authorized_keys` files, and generate a new key pair.

However, this *only* affects that one algorithm and key size. No other size of ECDSA key is affected, and no other key type is affected.



UNIVERSITY OF
CAMBRIDGE

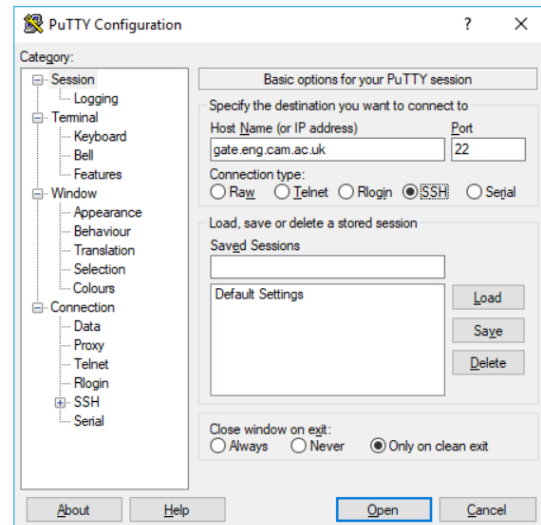
For remote control

<https://help.eng.cam.ac.uk/network/remote-access/pc-access-from-windows/>

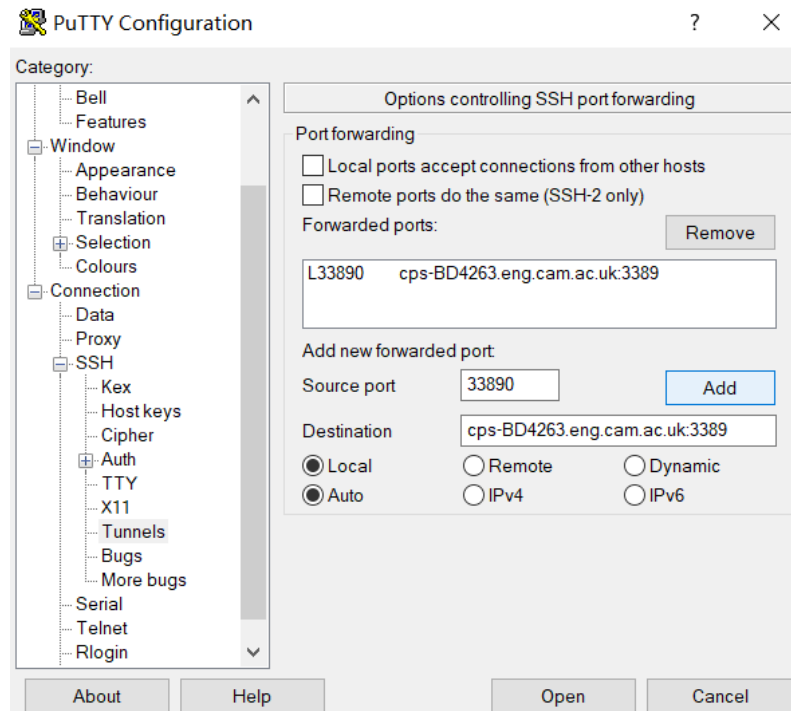
```
ssh -N -f -L 33890:mypc.eng.cam.ac.uk:3389 crsid@gate.eng.cam.ac.uk
```

The following example uses the PuTTY SSH client for Windows, which can be [freely downloaded](#).

Enter the following settings:



The next step is to create a port forward tunnel entry so that once the SSH connection is established any traffic sent to the local network port 33890 is redirected to the



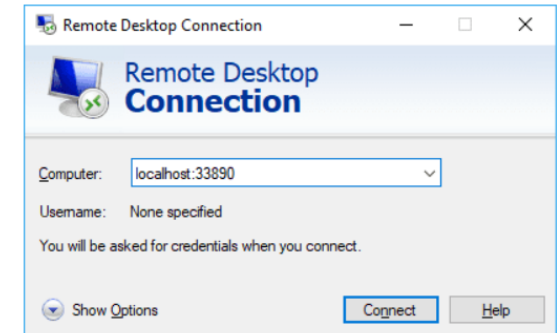
the screen.

Once you have established the SSH connection you can minimise the PuTTY window.

Note: You must establish this connection first before opening Remote Desktop

Connecting with the Remote Desktop Client

Open the Remote Desktop client on your PC and enter the following:



Click **Connect** and enter the username and password for your PC in the Department to gain access.

Troubleshooting

For remote control

Current Workstations address:

- cps-BD3977
- cps-BD4263
- cps-BD4682

Five/Six more to be ready....

lum-teach-lmserv.eng.cam.ac.uk

