

POLITECNICO DI TORINO

Collegio di Ingegneria Chimica e dei Materiali

**Corso di Laurea Magistrale
in Ingegneria Chimica e dei Processi Sostenibili**

Tesi di Laurea Magistrale

Ottimizzazione di un sistema multistadio per la separazione di CO₂ con membrana da miscele binarie N₂-CO₂ e CH₄-CO₂



Relatore

prof. Santarelli Massimo

Correlatore

dott. Cannone Salvatore Francesco

Candidato

Pedio Giulia

Marzo 2023

Indice

1. Introduzione.....	1
1.1 Introduzione al problema delle emissioni.....	1
1.2 Il metodo CCS: Carbon Capture and Storage.....	5
1.2.1 Possibili utilizzi della CO ₂	5
1.2.2 Fonti di emissione.....	6
1.2.3 Maturità delle tecnologie CCS.....	7
1.2.4 Tecnologie di post-conversione.....	9
1.3 Obiettivi e metodi.....	12
2. Separazione di CO₂ con membrana: stato dell'arte.....	15
2.1 Meccanismi di trasporto.....	16
2.1.1 Trasporto in membrane porose.....	17
2.1.2 Trasporto in membrane non porose.....	18
2.1.2.1 Solubilità.....	19
2.1.2.2 Diffusività.....	21
2.1.2.3 Permeabilità.....	22
2.1.2.4 Selettività.....	24
2.1.2.5 Effetto della temperatura.....	24
2.1.2.6 Effetto Joule-Thomson.....	25
2.2 Materiali.....	26
2.2.1 Membrane inorganiche.....	26
2.2.1.1 Membrane a base di carbonio.....	27
2.2.1.2 Membrane a base di silice.....	27
2.2.1.3 Membrane a base di zeolite.....	27
2.2.1.4 Membrane a base di allumina.....	28
2.2.2 Membrane polimeriche.....	28
2.2.2.1 Polimeri vetrosi.....	28
2.2.2.2 Polimeri gommosi.....	29
2.2.3 Biopolimeri: un nuovo approccio.....	30
3. Modellazione del modulo a membrana.....	33
3.1 Design del modulo a membrana.....	34
3.1.1 Modulo plate-and-frame.....	34
3.1.2 Modulo spiral-wound.....	35
3.1.3 Modulo tubolare.....	36
3.1.4 Modulo capillare.....	36
3.1.5 Modulo a fibra cava.....	37
3.2 Modelli di flusso.....	38
3.2.1 Modello di completa miscelazione.....	39

3.2.2	Modello cross-flow.....	42
3.2.3	Modello controcorrente.....	46
3.2.3.1	Metodo di Runge-Kutta a 4 stadi.....	50
3.3	Validazione dei modelli e analisi parametrica.....	51
4.	Sistemi di separazione a membrana con due stadi.....	59
4.1	Schemi con due stadi di separazione.....	60
4.1.1	Due stadi in serie.....	60
4.1.2	Due stadi in cascata.....	61
4.1.3	Due stadi in serie con ricircolo.....	63
4.1.4	Due stadi in cascata con ricircolo.....	65
4.2	Obiettivi di progettazione.....	67
5.	Ottimizzazione a due obiettivi del sistema multistadio.....	69
5.1	NSGA-II (Elitist Non-Dominated Sorting Genetic Algorithm).....	70
5.1.1	Principi di base di un algoritmo genetico.....	70
5.1.2	Approcci per la ricerca del fronte di Pareto.....	72
5.1.3	Smistamento non dominato.....	74
5.1.4	Procedura complessiva dell'NSGA-II.....	74
5.1.5	Elitarismo controllato nell'NSGA-II.....	76
5.2	Algoritmi di ottimizzazione a un solo obiettivo.....	78
5.2.1	PSO (Particle Swarm Optimization).....	78
5.2.2	Algoritmo di ottimizzazione dei surrogati.....	79
5.2.3	Algoritmi di ricerca diretta.....	81
5.3	Applicazione degli algoritmi al problema di separazione.....	83
5.3.1	Ottimizzazione con focus sul permeato.....	84
5.3.2	Ottimizzazione con focus sul retentato.....	89
5.3.3	Scelta di un punto sulla curva di Pareto.....	92
6.	Casi studio.....	95
6.1	Separazione della CO ₂ da un gas di scarico.....	95
6.2	Separazione della CO ₂ da un biogas.....	100
7.	Conclusioni	107
	Lista dei simboli.....	109
	Riferimenti bibliografici.....	113
	Appendice A	
	Appendice B	
	Appendice C	
	Appendice D	
	Appendice E	
	Appendice F	
	Appendice G	

1. Introduzione

Nell'ultimo decennio, le emissioni globali di CO₂ e in generale dei gas serra (GHG) hanno raggiunto quantitativi allarmanti, che dovrebbero promuovere un rinnovato interesse nello sviluppo di nuove strategie che limitino le ripercussioni sul nostro pianeta. Tra le principali cause dei cambiamenti climatici e del riscaldamento globale, le emissioni dei gas serra in atmosfera aumentano inesorabilmente, seppur con un tasso di crescita annuo ridotto (2,1% tra il 2000 e il 2009, 1,3% tra il 2010 e il 2019 [1]). Nel decennio 2010-2019 le emissioni nette globali di GHG di origine antropica hanno raggiunto livelli più alti rispetto a qualsiasi altro periodo storico, con un massimo di 59 GtCO_{2,eq} nel 2019 [1]. Gli impatti dei cambiamenti climatici si riflettono su innumerevoli aspetti della vita umana, dalle infrastrutture al peggioramento delle condizioni di salute, con effetti negativi anche sui servizi ecosistemici, sulla sicurezza idrica e alimentare e sulla biodiversità. L'obiettivo di mantenere a 1,5°C l'aumento della temperatura media globale diventa sempre più difficile da raggiungere e richiederebbe interventi rapidi e mirati a livello internazionale.

1.1 Introduzione al problema delle emissioni

Quando si parla di emissioni globali antropiche di gas serra ci si riferisce a diversi gruppi di gas: la CO₂ prodotta dalla combustione di combustibili fossili e da processi industriali (CO₂-FFI); la CO₂ proveniente invece dall'uso del suolo, dai cambiamenti nell'uso del suolo e dalla silvicoltura (CO₂-LULUCF); il metano (CH₄); il protossido di azoto (N₂O) e i gas fluorurati (HFC, PFC, SF₆, NF₃). Le loro emissioni vengono espresse in termini di GtCO_{2,eq} e convertite in base al loro potenziale di riscaldamento globale riferito a un intervallo di tempo di 100 anni (GWP-100). Ognuno dei gruppi citati ha raggiunto il quantitativo di emissioni medie globali più alto di sempre. Se le emissioni nette di CO₂-LULUCF hanno mostrato nel corso degli anni variazioni trascurabili, le emissioni di CO₂-FFI, al contrario, risultano le più critiche (FIGURA 1.1). Nell'ultimo decennio, le emissioni globali di CO₂ da fonti fossili sono state certamente favorite dal prodotto interno lordo (PIL) pro capite e dalla crescita demografica.

Con la pandemia COVID-19 e la conseguente attuazione di politiche di blocco si è verificata, in concomitanza con una profonda contrazione economica, la più importante riduzione di emissioni globali di CO₂ dai tempi della Seconda Guerra Mondiale. Si stima un calo del 5,8% delle emissioni di CO₂ rispetto al 2019, e cioè una diminuzione totale di 2,2 GtCO₂ (FIGURA 1.2). Ovviamente è stata la diretta conseguenza di una riduzione della domanda di energia in quasi tutti i settori, tranne che negli edifici residenziali. Il settore dei trasporti ne è stato particolarmente colpito, si pensi che le emissioni generate dall'aviazione internazionale sono diminuite del 45%. L'allentamento delle politiche di blocco ha però fatto rinnalzare rapidamente il livello delle emissioni. Ancor peggio, per una ripresa economica sono oggi spesso favorite le industrie dei combustibili fossili: nel 2021 la domanda di tutti i combustibili fossili è cresciuta di almeno il 5%.

Si è stimato che le emissioni future di CO₂ prodotte dalle infrastrutture attualmente esistenti a combustibili fossili superano già le rimanenti emissioni nette cumulative di CO₂ programmate nei piani finalizzati a limitare a 1,5°C il riscaldamento globale (senza *overshooting* o con un *overshooting* limitato). Per allineare le future emissioni di CO₂ con questi percorsi che prevedono di limitare il riscaldamento a 1,5°C o massimo 2°C sarà necessario un cambiamento sostanziale del sistema energetico nei prossimi 30 anni. L'utilizzo dei combustibili fossili andrà necessariamente limitato, vanno incentivate le fonti energetiche a basso o nullo contenuto di

carbonio, l'utilizzo dell'elettricità e di vettori energetici alternativi. La crescita media annua delle emissioni di gas serra nel settore energetico è già diminuita dal 2,3% per il decennio 2000-2009 all'1,0% per il 2010-2019, grazie ad un miglioramento dell'efficienza energetica e a una riduzione dell'intensità di carbonio del rifornimento energetico, principalmente grazie al passaggio dal carbone al gas naturale. Dal 2010, infatti, le centrali elettriche a carbone sono state ritirate in Europa e negli Stati Uniti, uno scarso numero di impianti è stato aggiunto in Cina, e un gran numero di impianti che erano stati pianificati a livello globale sono stati demoliti o convertiti alla co-combustione con biomassa. Ma non è abbastanza. La quota di mercato combinata delle tecnologie di generazione eolica e fotovoltaica è ancora inferiore al 10%, e la crescita della domanda di elettricità sicuramente non aiuta al raggiungimento dell'obiettivo del 100% di energia elettrica a basse emissioni di carbonio per il 2050.

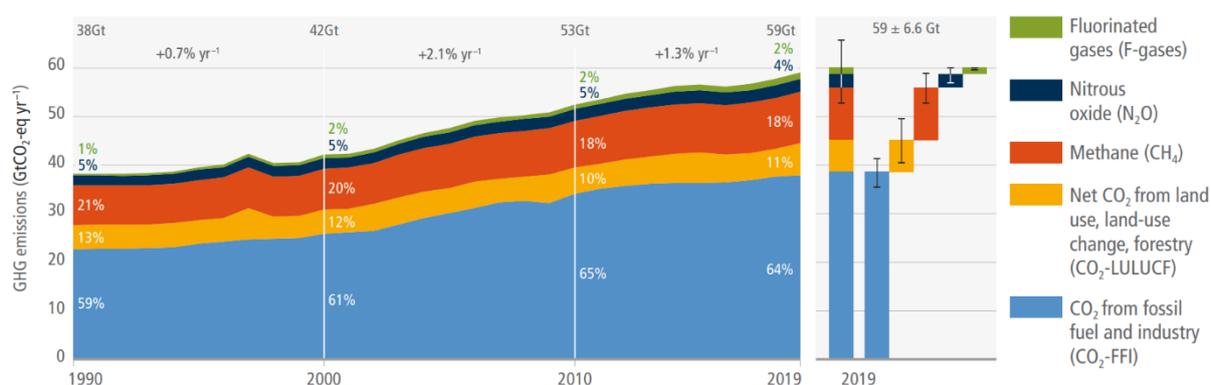


Figura 1.1 Emissioni nette di gas serra di natura antropogenica negli anni 1990-2019 [1]

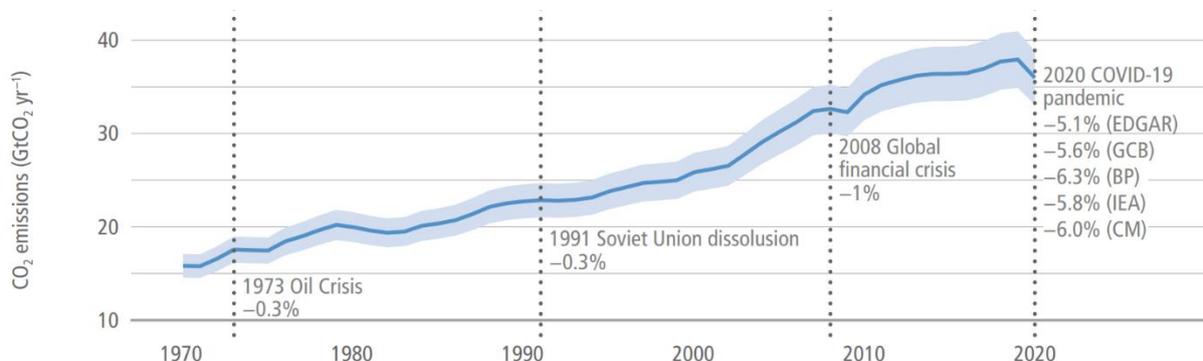


Figura 1.2 Emissioni globali di CO₂-FFI e l'impatto di eventi economici e geopolitici nella storia [1]

Volgendo lo sguardo alle emissioni globali di GHG prodotte per settore, ci si rende conto che in realtà elettricità e riscaldamento sono causa di solo il 23% delle emissioni, mentre i settori dell'industria, dei trasporti e degli edifici ne costituiscono ben il 44% (FIGURA 1.3). Se poi si considerano oltre alle emissioni dirette anche quelle indirette, quindi riallocando le emissioni da elettricità e riscaldamento al settore finale di utilizzo, queste tre categorie raggiungono il 66% di emissioni globali di GHG. Queste ultime sono fortemente influenzate dalla crescita della domanda per prodotti manifatturieri e materie di base e dalla tendenza globale all'aumento della superficie pro capite, delle distanze percorse, del peso e dimensione dei veicoli, dell'utilizzo dei servizi energetici degli edifici. La decarbonizzazione del settore industriale è possibile se si favoriscono tecnologie che utilizzano elettricità e idrogeno per l'energia e le materie prime, tecnologie per la cattura e l'utilizzo dell'anidride carbonica e una filosofia di riutilizzo degli scarti. Ma se a questo non si accompagna una riduzione della domanda di

materie di base e un rapido scale-up delle innovazioni a basse emissioni di carbonio, le emissioni dell'industria non rallenteranno. Una misura più facilmente perseguibile e certamente conveniente è l'elettrificazione del trasporto di massa, ovviamente caricando i veicoli con elettricità a bassa emissione di anidride carbonica. Quello degli *e-vehicles* è il campo in più rapida crescita dell'industria automobilistica, essendo riuscito a raggiungere una quota di mercato a due cifre in molti Paesi entro il 2020. Nei Paesi più sviluppati, inoltre, una significativa transizione dall'utilizzo del carbone e della biomassa negli edifici in favore di vettori energetici più moderni e tecnologie di conversione più efficienti ha avuto degli effetti positivi nella riduzione delle emissioni. Le emissioni prodotte da altri tipi di attività a fini energetici, come la raffinazione del petrolio, l'estrazione del carbone, fughe di petrolio e gas, e altro si limitano al 10% [1].

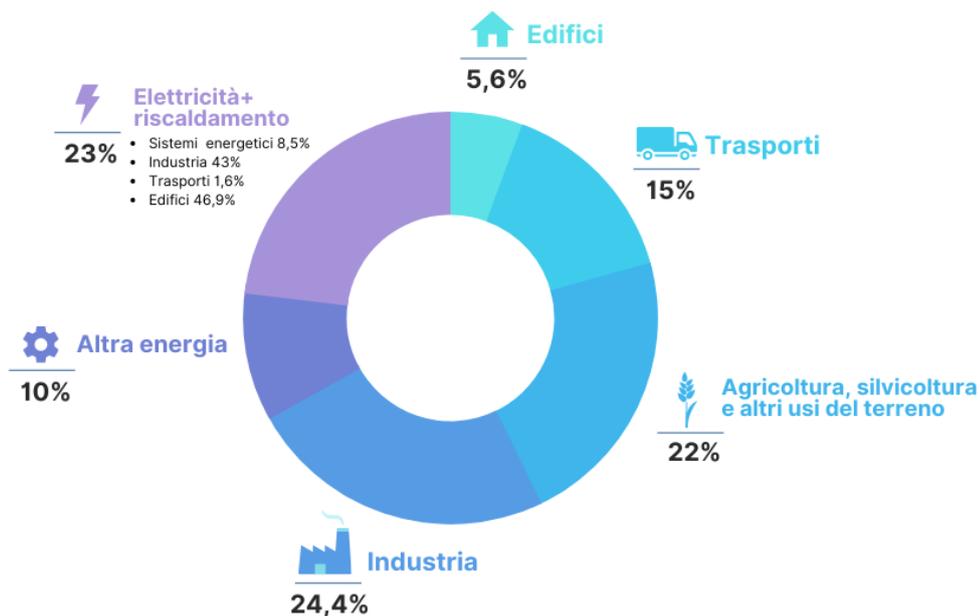


Figura 1.3 Emissioni totali antropogeniche dirette di GHG nell'anno 2019

Il *World Energy Outlook 2022* dell'*International Energy Agency* (IEA) ha fornito tre scenari con lo scopo di mostrare l'attuale quadro di riferimento sul futuro dell'energia. Ogni scenario si fonda sulle diverse possibili risposte alla crisi energetica globale in atto, e quindi si differenziano principalmente per le ipotesi sulle politiche governative. Per ognuno di questi scenari è stata stimata l'evoluzione delle future emissioni di CO₂ fino al 2050 e il conseguente aumento della temperatura media globale nel 2100. È immediato osservare in FIGURA 1.4 i progressi compiuti dall'Accordo di Parigi del 2015, prima del quale si prevedeva che le emissioni sarebbero aumentate fino a più di 50 GtCO₂ nel 2050, con un incremento della temperatura media di 3,5°C entro il 2100. Sono poi indicate tre diverse traiettorie, a seconda dello scenario:

- Lo scenario delle politiche dichiarate (STEPS, *Stated Policies Scenario*), che non tiene conto di ciò che i governi dichiarano di voler realizzare, ma unicamente di ciò che stanno attualmente attuando per raggiungere gli obiettivi prefissati; si tratta quindi di un'analisi pragmatica dell'attuale panorama politico e di cosa accadrebbe in assenza di iniziative politiche specifiche; lo STEPS prevede un plateau delle emissioni a metà del decennio 2020-2029, a 37 GtCO₂, per poi subire una lenta riduzione a 32 GtCO₂ nel 2050, con

un incremento della temperatura media di 2,5°C; ovviamente non è la giusta risposta al problema del riscaldamento globale e non ne eviterebbe i gravi impatti sul pianeta;

- Lo scenario degli impegni annunciati (APS, *Announced Pledges Scenario*) considera che tutti gli impegni climatici assunti dai governi verranno integralmente e puntualmente rispettati; tiene in considerazione anche quei Paesi che, pur non avendo annunciato alcun impegno a lungo termine, beneficerebbero della riduzione dei costi di una serie di tecnologie ad energia pulita; in questa previsione, le emissioni di CO₂ raggiungeranno un picco a metà degli anni 2020 per poi abbassarsi a 12 Gt nel 2050; l'aumento della temperatura media globale arriverebbe a 1,7°C nel 2100, dimostrazione del fatto che le intenzioni politiche procedono nel verso giusto; tuttavia, come sostenuto dall'IPCC, non basterebbe ancora ad evitare effetti negativi per le società di tutto il mondo;
- Lo scenario di zero emissioni nette entro il 2050 (NZE, *Net Zero Emissions*) è uno scenario normativo che definisce un percorso da seguire per limitare l'aumento della temperatura media a 1,5°C, facendo affidamento sulle sole riduzioni delle emissioni nel settore energetico; questo scenario è in accordo anche con alcuni degli Obiettivi di Sviluppo Sostenibile dell'Agenda 2030, come l'accessibilità universale all'energia e il miglioramento della qualità dell'aria; seguendo questo piano le emissioni di CO₂ diminuiscono a 23 Gt nel 2030 e si azzerano nel 2050, cioè la quantità di CO₂ che l'attività umana immette in atmosfera eguaglia la quantità di CO₂ che rimuove dall'atmosfera.

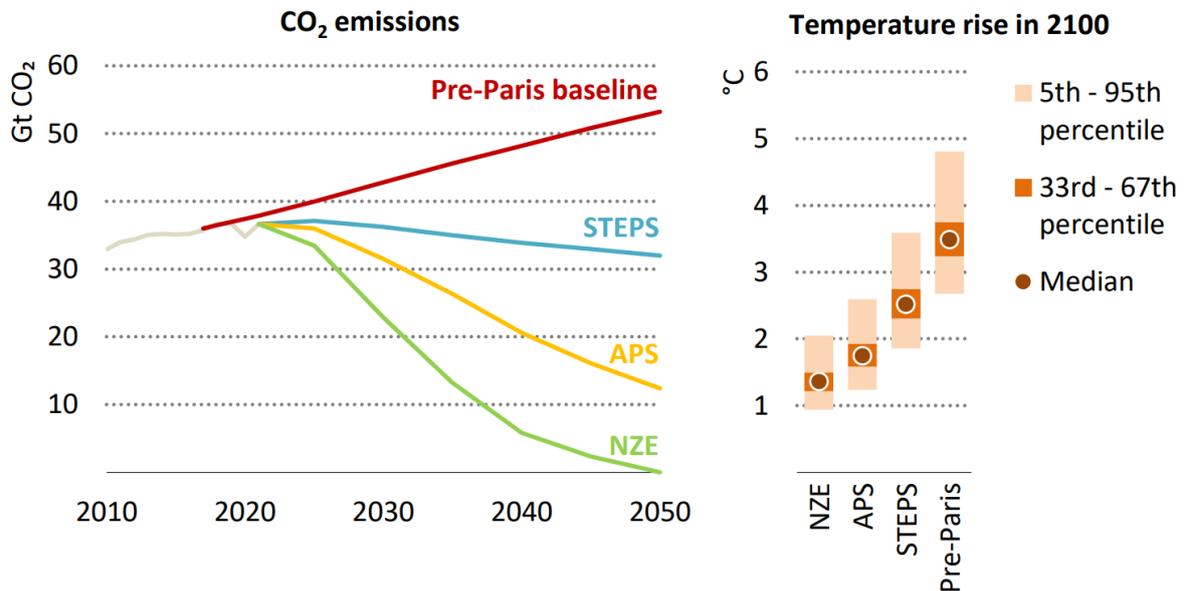


Figura 1.4 Emissioni di CO₂ legate all'energia e ai processi negli anni 2010-2050; aumento di temperatura nel 2100 a seconda dello scenario [2]

Il piano a zero emissioni prevede diverse soluzioni: l'utilizzo di sistemi elettrici che non producono CO₂ o che comunque la rimuovono dall'atmosfera; l'elettrificazione di trasporti leggeri, il riscaldamento degli ambienti e la cottura dei cibi; un notevole decremento nell'utilizzo di combustibili fossili, a favore di vettori energetici alternativi come l'idrogeno, la bioenergia e l'ammoniaca; un miglioramento dell'efficienza energetica; l'implementazione di un sistema energetico integrato tra regioni e tra i componenti del sistema stesso; una rapida

accelerazione nell'implementazione di soluzioni per la cattura, l'utilizzo e lo stoccaggio dell'anidride carbonica (CCUS), che permetterebbe una spiccata riduzione delle emissioni nei settori dell'industria, dell'energia e della trasformazione dei combustibili.

1.2 Il metodo CCS: *Carbon Capture and Storage*

Affinché possano essere perseguiti gli obiettivi del piano a zero emissioni nette entro il 2050, risulta essenziale contare e investire sulle tecnologie che permettono la cattura delle emissioni in eccesso di CO₂ in quegli impianti in cui sono inevitabili, come nelle centrali elettriche e nell'industria pesante. Questa tecnica di cattura e successivo immagazzinamento della CO₂ in impianti di stoccaggio sotterraneo prende il nome di CCS (*Carbon Capture and Storage*). La CCS prevede tre fasi: la cattura della CO₂ direttamente dalla fonte delle emissioni grazie ad opportune tecnologie; la fase di trasporto in cui la CO₂ viene compressa e trasformata in un fluido, per essere trasportata mediante oleodotti, navi o altri veicoli; infine la fase di stoccaggio, in cui la CO₂ viene immagazzinata in profondità nel sottosuolo, tipicamente in giacimenti esauriti di gas e petrolio, giacimenti carboniferi e formazioni saline profonde. La capacità di stoccaggio geologico della CO₂ si stima essere di 1000 GtCO₂, quindi superiore a quella necessaria per limitare il riscaldamento globale a 1,5°C entro il 2100 [3], ciononostante la fase di stoccaggio costituisce comunque un argomento di discussione. La spinta verso lo sviluppo della CCS è giustificata dal fatto che nonostante si stiano facendo molti sforzi per aiutare la decarbonizzazione del settore energetico tramite le rinnovabili, sarà difficile rinunciare totalmente e immediatamente ai combustibili fossili, e in più le industrie pesanti come quella del cemento, dell'alluminio, del ferro e l'acciaio, della carta e della cellulosa continueranno inevitabilmente ad emettere CO₂. I promotori di queste tecnologie affermano che si potrà catturare anche l'85-90% della CO₂ emessa da questi impianti. Tuttavia, le tecnologie CCS sono ancora per la maggior parte in fase prototipale, quindi non ancora disponibili per l'utilizzo commerciale su ampia scala, nonostante alcune realtà le stiano già testando o addirittura utilizzando. Ci sono ancora alcune sfide tecnologiche, economiche, ecologico-ambientali e istituzionali da risolvere. Gli impianti CCS commerciali, operativi o in costruzione, erano solo 31 in tutto il mondo nel 2021, con una capacità di cattura della CO₂ di 40 milioni di tonnellate all'anno. Per soddisfare lo scenario NZE per il 2050, la capacità di cattura della CO₂ dovrà raggiungere 1,7 miliardi di tonnellate annue entro il 2030, il che richiederebbe la costruzione di altri 70-100 impianti all'anno [3].

1.2.1 Possibili utilizzi della CO₂

Più recente è l'idea di poter accoppiare la cattura della CO₂ con dei processi che permettono di trasformarla in prodotti di valore, come prodotti chimici o combustibili. Questa tecnica ha preso il nome di CCU (*Carbon Capture and Utilization*) e ha il vantaggio, rispetto alla CCS, di essere un'attività redditizia, in quanto permetterebbe la vendita dei prodotti ottenuti da una materia di scarto. Da un punto di vista economico, quindi, la CCU sembrerebbe essere l'opzione migliore, tuttavia va sempre fatta una valutazione complessiva sugli impatti ambientali e l'efficacia dei costi per garantire un bilancio economico e ambientale positivo. Molto dipende dal tipo di utilizzo a cui si sceglie di finalizzare la CO₂ catturata [4]:

- Utilizzo diretto della CO₂: nell'industria alimentare e delle bevande ha già numerosi utilizzi, è un agente gassante, fa da solvente per l'estrazione di aromi, inertizza gli imballaggi, fa da conservante ed è parte del processo di decaffeinizzazione. Anche nell'industria farmaceutica può essere utilizzato come intermedio di reazione per la sintesi di farmaci. In queste applicazioni la CO₂ deve essere fornita ad elevata purezza;

- Recupero avanzato del greggio (EOR, *Enhanced Oil Recovery*): è una tecnica già utilizzata che permette, grazie all'iniezione di gas o altri agenti, di aumentare la pressione nel giacimento petrolifero e permettere l'estrazione del greggio che altrimenti resterebbe irrecuperabile. Iniettata in condizioni supercritiche, si miscela bene col petrolio diminuendone la viscosità. Ponendo attenzione alle condizioni di iniezione si potrebbe far in modo di intrappolare la CO₂ nel sottosuolo, ottenendo un risultato simile allo stoccaggio geologico. Tuttavia molti vedono la contraddittorietà di questo utilizzo;
- Conversione della CO₂ in *chemicals* e combustibili: può essere utilizzata come precursore nelle reazioni di carbosilazione e la sintesi di prodotti organici come carbonati e polimeri; può essere sottoposta a riduzione per l'ottenimento di molecole quali metanolo, metano, syngas, acido formico e urea; può costituire un reagente nei processi del tipo Fischer-Tropsch per la produzione di combustibili. Si tratterebbe in ogni caso di uno "stoccaggio" temporaneo della CO₂, in quanto sarà inevitabilmente riemessa con l'utilizzo di questi prodotti;
- Carbonatazione minerale: si fa reagire con un ossido metallico (come magnesio o calcio) per la produzione di carbonati, prodotti stabili in grado di immagazzinare la CO₂ per lunghi periodi. Tuttavia i costi e l'energia necessari a questo tipo di applicazione sono ancora troppo elevati e ne penalizzano la diffusione su larga scala;
- Sintesi di biocarburanti da microalghe: la CO₂ potrebbe essere utilizzata per la coltivazione di microalghe al fine di produrre biocarburanti. Le microalghe sono in grado di fissare l'anidride carbonica direttamente dai flussi di scarto, come i gas di scarico, e allo stesso tempo utilizzare l'azoto come nutriente. Anche questo tipo di applicazione necessita di ulteriori studi.

1.2.2 Fonti di emissione

Quasi la metà delle emissioni di CO₂ di origine antropica proviene da fonti distribuite, come i trasporti e il riscaldamento residenziale e commerciale, mentre il resto viene emesso da fonti puntuali massicce, come le centrali termoelettriche, le industrie dell'acciaio e del cemento, i processi di fermentazione, le raffinerie, gli impianti di combustione. Le emissioni da fonti di tipo stazionario costituiscono il principale obiettivo per la cattura dell'anidride carbonica. L'idoneità e la selezione di una tecnologia CCS dipende fortemente dalla fonte e quindi dalle caratteristiche del flusso, come portata, composizione chimica, temperatura e pressione. È intuibile che il costo e la richiesta energetica per la cattura aumentano al diminuire della pressione parziale di CO₂ nel flusso da trattare. Perciò una buona fonte di cattura sono i flussi prodotti da molti processi industriali, altamente concentrati in CO₂. La sfida per il settore industriale sarà piuttosto mantenere la competitività internazionale pur integrando al processo le tecnologie CCS, che ne aumentano i costi complessivi. In genere, i flussi di scarico degli impianti di produzione di energia elettrica sono caratterizzati da pressioni parziali di CO₂ molto inferiori rispetto a quelle dei flussi di scarico industriali. Questo rende la separazione della CO₂ relativamente più difficile e costosa. Il motivo risiede nel fatto che spesso la combustione dei combustibili fossili si fa avvenire in presenza di aria, per cui il gas di scarico è principalmente costituito da azoto. Nei gas di scarico delle centrali a carbone la concentrazione di CO₂ è tra il 10-15%, per le centrali a gas naturale questo valore si abbassa al 3-5% e si aggiunge il vapore acqueo come componente significativo [5]. In TABELLA 1.1 sono indicate concentrazioni e pressioni parziali della CO₂ per i gas di scarico di alcuni impianti industriali e centrali termoelettriche.

Le stesse tecnologie CCS possono essere applicate ai flussi di scarico degli impianti a biomassa, accoppiando in questo modo la produzione di energia da materie prime biogeniche con la cattura della CO₂, che in questo modo non viene riemessa in atmosfera. Si parla in questo caso di BECCS (o Bio-CCS) e comporterà generalmente emissioni nette negative. Tuttavia, l'entità di tali emissioni negative dipenderà dall'intero ciclo vita dell'impianto, compreso il cambiamento di destinazione d'uso del suolo, il pretrattamento e il trasporto della biomassa, il processo di conversione e il processo CCS. La quantità di emissioni di CO₂ da trattare può essere variabile, perché la CCS può essere integrata a una vasta gamma di tecnologie, come la combustione dedicata di biomassa o co-combustione per la produzione di energia, la conversione della biomassa in combustibili liquidi e gassosi, le cartiere, gli impianti di etanolo, impianti di gassificazione della biomassa e le raffinerie di biogas [6, 7]. Quest'ultima è proprio una delle tecnologie più promettenti su ampia scala. Il biogas è essenzialmente una miscela di metano e anidride carbonica, solitamente con un rapporto 3:2. Si forma dalla digestione anaerobica o dalla co-digestione anaerobica di composti organici, come letame, rifiuti alimentari, liquami e residui di colture. Questi composti vengono degradati grazie a varie interazioni metaboliche tra diversi microrganismi. In generale, si ritiene che agli impianti di BECCS possano essere associati costi più elevati rispetto agli impianti CCS solo fossili, poiché sono essenzialmente più piccoli [8]. Questo problema potrebbe essere superato espandendo l'uso dell'energia da biomassa.

Tabella 1.1 Concentrazioni e pressioni parziali tipiche di CO₂ nei gas di scarico di diversi impianti di produzione industriale e di generazione di energia [5]

<i>Tipo di impianto o industria</i>	<i>Conc. media di CO₂ [%v]</i>	<i>P_{CO2} [MPa]</i>
Centrale elettrica a carbone	12-14	0.012-0.014
Centrale elettrica con turbina a gas naturale	3-4	0.003-0.004
Centrale elettrica a olio combustibile	11-13	0.011-0.013
Ciclo combinato a gassificazione integrata	12-14	0.012-0.014
Produzione di cemento	14-33	0.014-0.033
Produzione di acciaio	20	0.040-0.060
Produzione di ammoniaca	18	0.5
Produzione di ossido di etilene	8	0.2
Produzione di idrogeno (<i>steam reforming</i> dal metano)	15-20	0.3-0.5
Produzione di metanolo	10	0.27

1.2.3 Maturità delle tecnologie CCS

Per ogni fase della CCS esistono una serie di tecnologie in via di sviluppo. In genere, prima che una tecnologia venga diffusa su scala commerciale, deve essere verificata su scala di laboratorio, poi su scala pilota e ancora su scala dimostrativa. In FIGURA 1.5 sono riportate alcune delle tecnologie più note per le fasi di cattura, trasporto, stoccaggio e utilizzo della CO₂, a ciascuna delle quali è associato il proprio livello di maturità tecnologica (TRL, *Technology Readiness Level*). Il TRL è una metodologia attraverso cui si identifica il livello di maturità di una tecnologia, utilizzando una scala di valori da 1 a 9, dove 1 rappresenta la base ("osservazione dei principi di base della tecnologia") e 9 ne è l'apice ("sistema dimostrato in ambiente operativo reale"). Molte delle tecnologie sono ferme a un TRL di 3 ("proof of concept sperimentale"), 6 ("tecnologia dimostrata in ambiente rilevante (o industriale)") e 7 ("dimostrazione del prototipo in ambiente operativo"). L'avanzamento di una tecnologia oltre

il livello 3 richiede ulteriori finanziamenti per la ricerca, mentre la progressione oltre il TRL 5 e 7 necessita di importanti investimenti finanziari e/o di interesse commerciale. La tecnica di cattura della CO₂ per assorbimento chimico, ad esempio usando soluzioni acquose di ammine, è stata utilizzata per decenni per la deacidificazione del gas naturale, per cui ha TRL 9. Altre tecnologie per la cattura sono riuscite a raggiungere un TRL pari a 7 e potrebbero ottenere lo stato commerciale nel prossimo futuro. Tra queste l'ossicombustione delle centrali a carbone, l'adsorbimento e le membrane polimeriche. La membrana *Polaris* è già disponibile in commercio ed è utilizzata per la separazione dell'anidride carbonica dal *syngas*. Le tecnologie di trasporto mediante navi o condutture sono già utilizzate commercialmente ed hanno TRL 9. Sono 46500 i km di condutture (*on-shore* e *off-shore*) disponibili in tutto il mondo per il trasporto di CO₂, molte delle quali sono associate alle operazioni di EOR negli Stati Uniti. Si intuisce quindi che anche l'EOR ha TRL 9, così come le formazioni saline per lo stoccaggio della CO₂. Al contrario, lo stoccaggio oceanico e lo stoccaggio minerario sono ancora nelle prime fasi di sviluppo, e hanno TRL rispettivamente 2 e 3. Alcune tecnologie di utilizzo della CO₂, come l'utilizzo diretto nell'industria alimentare e per la produzione di *chemicals*, sono già mature. Altre sono ferme a un TRL pari a 6. In Giappone esiste un impianto che utilizza la CO₂ separata dallo scarico di un inceneritore per la coltivazione di alghe.

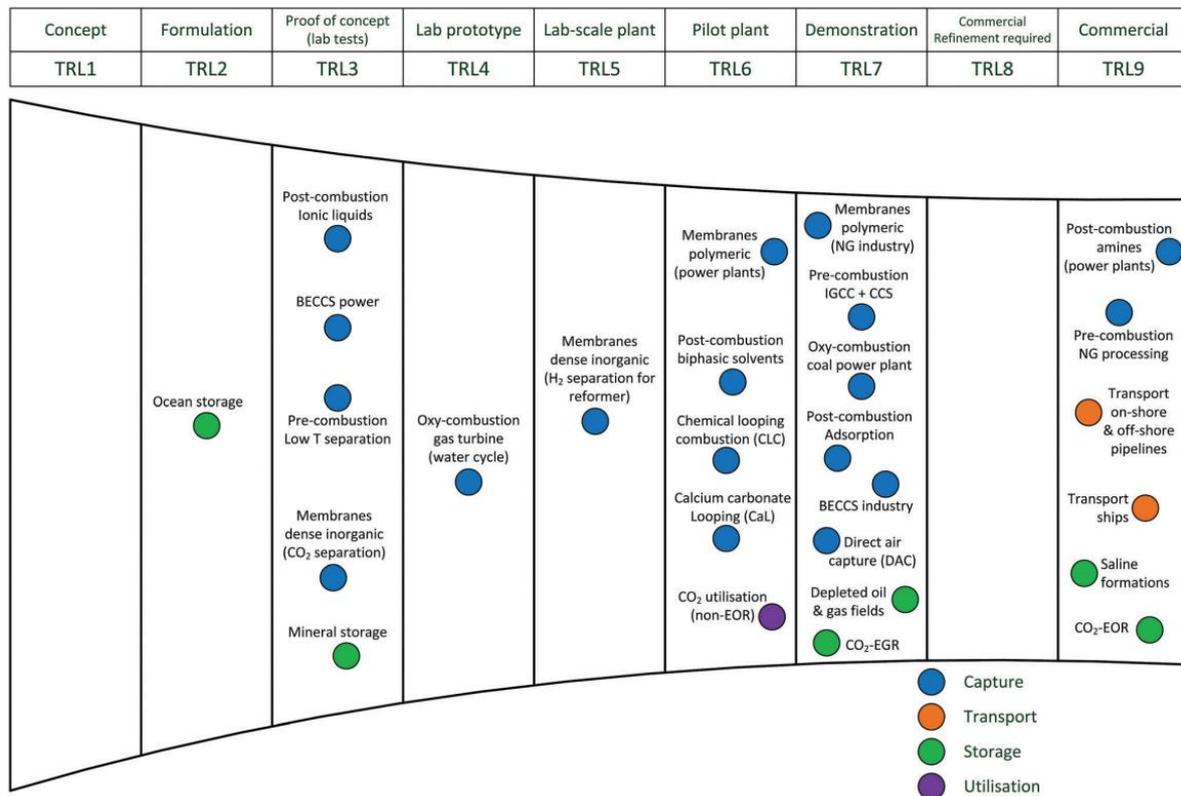


Figura 1.5 Attuali progressi nello sviluppo delle tecnologie di cattura, trasporto, stoccaggio e utilizzo della CO₂ in termini di livello di maturità tecnologica (TRL). L'elenco delle tecnologie non intende essere esaustivo. BECCS = *Bio-energy CCS*; IGCC = *Integrated Gasification Combined Cycle*; EGR = *Enhanced Gas Recovery*; EOR = *Enhanced Oil Recovery* [7]

In generale, esistono quattro categorie di cattura della CO₂ [4, 9]:

- La cattura pre-conversione: si tratta sostanzialmente della gassificazione del combustibile, che sia carbone, biomassa, o gas naturale, con formazione di una miscela contenente principalmente CO e H₂, che viene inviata in un reattore catalitico (*shift*

converter) insieme a vapore acqueo per la reazione di *water gas shift* ottenendo ulteriore H₂ e CO₂; la miscela viene raffreddata e la CO₂ separata tramite un'unità di separazione dei gas acidi con *Selexol*; l'idrogeno viene utilizzato come combustibile in un impianto a ciclo combinato con turbina a gas (IGCC); il vantaggio sta nell'aver convertito un combustibile carbonioso in un combustibile privo di carbonio, e nel trattare un flusso con alta concentrazione di CO₂; il costo capitale totale dell'impianto è però molto elevato.

- L'ossicombustione: prevede la combustione in presenza di ossigeno invece che di aria, ottenendo dei gas di scarico con una concentrazione di CO₂ anche superiore all'80%; per mantenere la temperatura al valore di una tipica combustione ad aria, i gas di scarico vengono in parte ricircolati nella camera di combustione; per la produzione di O₂ e la cattura della CO₂ sono sufficienti semplici metodi di separazione fisica, evitando in questo modo l'utilizzo di reagenti e/o solventi e limitando quindi l'impatto ambientale e i costi operativi del processo; lo svantaggio è che la quantità di ossigeno necessaria per l'ossicombustione è costosa sia in termini energetici che in termini di costo capitale.
- La cattura diretta dell'aria dall'atmosfera (DAC, *Direct Air Capture*), attualmente una misura energeticamente ed economicamente costosa.
- La cattura post-conversione: la CO₂ viene separata alla fonte dell'emissione, successivamente alla conversione della risorsa carboniosa in CO₂.

La FIGURA 1.6 fornisce una visione complessiva di quanto detto finora e anticipa quelle che sono le tecnologie CCS, che verranno brevemente descritte nella prossima sezione.

1.2.4 Tecnologie di post-conversione

Lo scopo di questa sezione sarà di fornire sommariamente una descrizione delle principali tecnologie di post-conversione per la cattura della CO₂ [7, 9].

- Assorbimento fisico. Prevede uno step di assorbimento in cui la CO₂ si solubilizza in un opportuno solvente e uno step di stripping per la rigenerazione del solvente e l'ottenimento di una corrente di CO₂ pura. Non avviene alcuna reazione chimica, ma l'interazione tra CO₂ e solvente è regolata dalla legge di Henry, per cui è favorita ad alte pressioni parziali e basse temperature. Per la rigenerazione del solvente è sufficiente operare con un aumento della temperatura o una riduzione della pressione, per cui la richiesta energetica non è eccessiva. I solventi sono già da tempo disponibili su scala commerciale, come il *Selexol* (dimetiletere del polietilenglicole), metanolo a bassa temperatura (processo *Rectisol*) e il carbonato di propilene (processo *Fluor*).
- Assorbimento chimico. Comporta l'utilizzo di solventi basici in grado di interagire con la CO₂ tramite reazione acido-base; anche in questo caso la colonna di assorbimento è seguita da una colonna di rigenerazione del solvente, che avviene ad alte temperature. La tecnica più ampiamente utilizzata è sicuramente l'assorbimento con monoetanolamina (MEA), che ha basso costo e buone proprietà basiche, ma purtroppo anche una tendenza alla degradazione ed è fortemente impattante dal punto di vista ambientale. Si stanno dunque studiando delle alternative, come soluzioni acquose di ammoniaca e miscele di ammine. C'è un crescente interesse per i solventi poveri di acqua, come liquidi ionici e miscele di ammine organiche non acquose, che tuttavia tendono ad essere più costosi e per il momento sono limitati solo su scala di laboratorio.
- Adsorbimento. Si basa sulle proprietà di una sostanza di passare dalla fase gassosa alla superficie di un materiale solido (adsorbente) creando un legame di tipo fisico

(fisisorbimento) o chimico (chemisorimento). Sono stati testati sia materiali adsorbenti classici (carboni attivi, zeoliti, silice, allumina) che nuovi come idrotalciti, strutture metalliche organiche, polimeri, supporti di ammine. La rigenerazione può essere ottenuta tramite una riduzione della pressione (PSA, *Pressure Swing Adsorption*), o un aumento della temperatura (TSA, *Temperature Swing Adsorption*), o mediante l'utilizzo di corrente elettrica (*Electrical Swing Adsorption*) o ancora tramite lavaggio.

- Separazione criogenica. Prevede la separazione della miscela gassosa tramite condensazione frazionata e distillazione a bassa temperatura. Il processo si svolge a temperatura molto bassa, tale da liquefare e separare la CO₂, ottenendo quindi un prodotto vantaggioso per il trasporto ed eliminando i problemi di corrosione, dell'uso di acqua e di sostanze nocive. È una tecnologia commerciale che si utilizza comunemente per liquefare e purificare la CO₂ da fonti con purezza elevata (maggiore del 90%).
- Tecnologia a membrana. Si tratta di un concetto di cattura relativamente recente e che negli ultimi tre decenni è stato utilizzato a livello industriale per la separazione dell'aria o in applicazioni per l'arricchimento di ossigeno, finalizzate ai processi di combustione. Le membrane sono delle barriere semipermeabili, di materiale organico o inorganico, che permettono il passaggio selettivo solo di alcuni componenti della miscela alimentata. Generalmente hanno una buona affinità con la CO₂, che si concentrerà nella corrente permeata. Oltre che nei processi di post-conversione, potrebbero essere utilizzate anche nei sistemi di pre-conversione (per la separazione della CO₂ dall'H₂, ad esempio) e per la separazione dell'aria nei sistemi di ossicombustione, con possibilità di implementare un'intensificazione di processo.

I vantaggi e gli svantaggi di queste tecnologie sono riassunti in TABELLA 1.2, insieme alle possibilità di miglioramento che si stanno studiando e sviluppando.

Tabella 1.2 Tabella riassuntiva sui vantaggi e svantaggi delle attuali tecnologie di post-conversione per la cattura nella CO₂, principali sfide future e i relativi benefici che potrebbero apportare; HR=*heat ratio* per la rigenerazione [kJ per kg di CO₂ alimentata]; CR=*capture ratio* [kg di solvente per kg di CO₂ alimentata] [7,9,10]

<i>Tecnologia</i>	<i>Vantaggi</i>	<i>Svantaggi</i>	<i>Sfide future</i>	<i>Possibili benefici</i>
Assorbimento fisico	<ul style="list-style-type: none"> • Tecnologia matura • Alta efficienza di cattura • HR di rigenerazione basso • Simultanea disidratazione 	<ul style="list-style-type: none"> • Bassa selettività • Perdite di idrocarburi • Più economico ad alte pressioni • <i>Range</i> di temperatura limitato 	<ul style="list-style-type: none"> • Nuovi solventi (es. solventi fluorurati, liquidi ionici, solventi ibridi) • Sistema ibrido assorbimento-distillazione criogenica/membrana 	<ul style="list-style-type: none"> • Più stabili ad alte temperature; migliore solubilità della CO₂ • Minori costi ed energia

Tabella 1.2 (Continuazione)

<i>Tecnologia</i>	<i>Vantaggi</i>	<i>Svantaggi</i>	<i>Sfide future</i>	<i>Possibili benefici</i>
Assorbimento chimico	<ul style="list-style-type: none"> • Tecnologia matura • Alta efficienza di separazione • Alta selettività • Basse perdite di idrocarburi 	<ul style="list-style-type: none"> • Alto CR • Alto HR di rigenerazione • Corrosione • Emissioni • Degradazione del solvente 	<ul style="list-style-type: none"> • Nuovi solventi (es. miscele di solventi, solventi ibridi, solventi multifase, ioni liquidi) • Sistema ibrido assorbimento-distillazione criogenica/membrana 	<ul style="list-style-type: none"> • Maggiore stabilità termochimica; minore CR e HR; minori T di stripping • Minori costi ed energia
Adsorbimento	<ul style="list-style-type: none"> • Possibilità di <i>retrofitting</i> • Ampia <i>range</i> di temperatura e pressione 	<ul style="list-style-type: none"> • Adsorbimento competitivo di altri componenti • Bassa tolleranza dell'umidità • Difficoltà nella definizione di un processo ciclico 	<ul style="list-style-type: none"> • Nuovi adsorbenti (es. scarti di attività industriali e agricole, strutture di metalli organici) • Adsorbenti con funzionalità amminiche 	<ul style="list-style-type: none"> • Alta area superficiale; selettività e facilità di rigenerazione, con ridotta richiesta energetica • Energia per la rigenerazione ridotta
Distillazione criogenica	<ul style="list-style-type: none"> • Tecnologia matura • Alta selettività • Basse perdite di idrocarburi • Vantaggi nel trasporto (CO₂ liquida) 	<ul style="list-style-type: none"> • Alta energia di refrigerazione • Evitare il congelamento della CO₂ 	<ul style="list-style-type: none"> • Sistema ibrido distillazione criogenica-assorbimento (fisico o chimico) 	<ul style="list-style-type: none"> • Minori costi ed energia
Membrane	<ul style="list-style-type: none"> • Già dimostrata per il trattamento del gas naturale • Nessuna rigenerazione • Nessuna sostanza chimica • Ingombro ridotto • Basso impatto ambientale 	<ul style="list-style-type: none"> • Possibile intasamento da impurità • Bassa tolleranza di umidità • Perdite di idrocarburi • Compromesso permeabilità-selettività • Decadimento delle prestazioni nel tempo • Basso TRL per le applicazioni di post-conversione 	<ul style="list-style-type: none"> • Nuovi materiali • Schemi multistadio • Utilizzo di <i>sweep gas</i> • Supporto con solvente • Sistema ibrido membrana-assorbimento (fisico o chimico) 	<ul style="list-style-type: none"> • Alta permeabilità e selettività per la CO₂; superiore stabilità termochimica • Maggiore efficienza • Maggiore efficienza; minore area • Migliore selettività • Minori costi ed energia

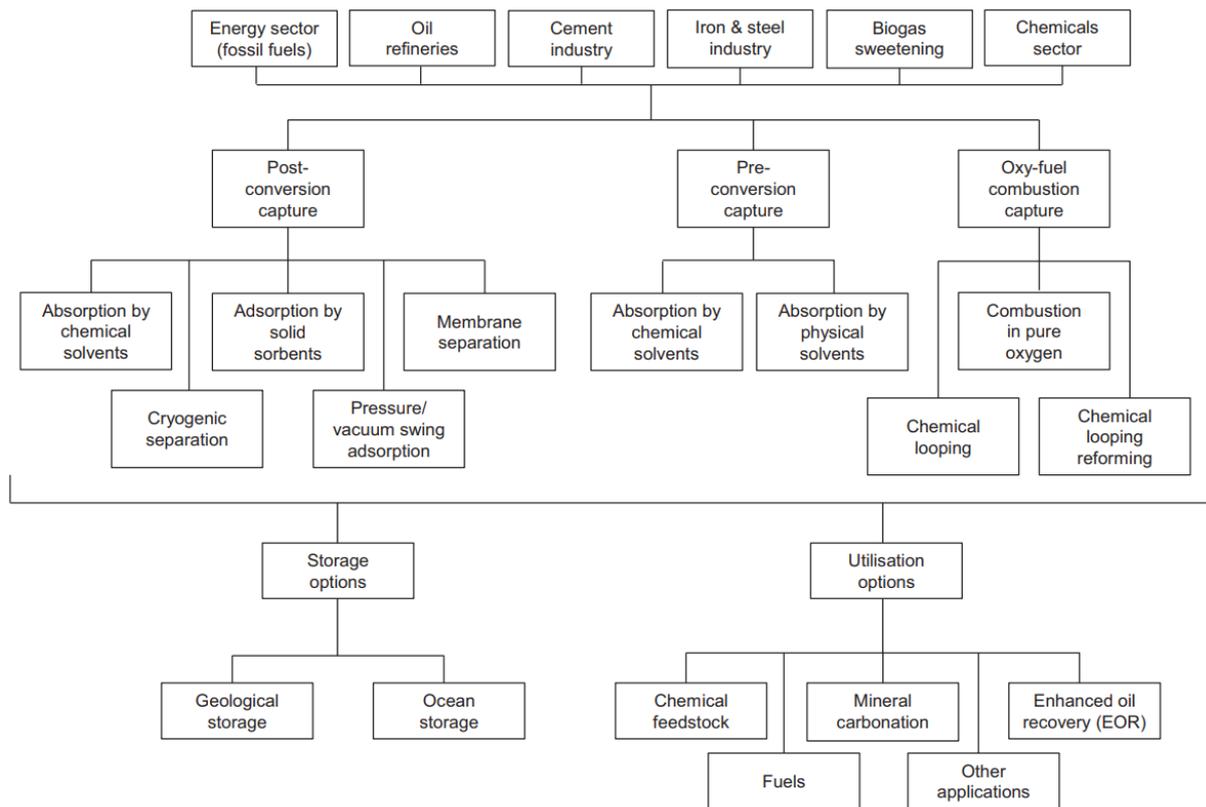


Figura 1.6 Overview sulle fonti di emissione e le diverse opzioni per la cattura, lo stoccaggio e l'utilizzo della CO₂ [4]

1.3 Obiettivi e metodi

La presente tesi sarà focalizzata sulla separazione di CO₂ mediante tecnologia a membrana, considerando in particolar modo miscele binarie N₂-CO₂ e CH₄-CO₂. Le prime possono essere rappresentative dei gas di scarico degli impianti a combustione, nonostante venga trascurata la presenza di altre impurità; le seconde sono caratteristiche del gas naturale o del biogas prodotto da un impianto di digestione anaerobica.

La separazione con tecnologia a membrana offre chiaramente dei vantaggi rispetto alle classiche tecnologie di assorbimento e adsorbimento. Innanzitutto non vi è necessità di rigenerazione, con conseguente minore richiesta energetica, che sarà dettata unicamente dalla necessità di avere un gradiente di pressione mediante l'utilizzo di compressori o pompe a vuoto. Le membrane non producono flussi di scarto, sono sistemi modulari semplici, poco ingombranti e facilmente scalabili. Questi motivi spiegano l'attenzione che si sta ponendo su questo tipo di tecnologia, con lo sviluppo di membrane sempre più performanti, con una maggiore permeabilità e selettività verso la CO₂.

Il principale limite nell'utilizzo della tecnologia a membrana sussiste nella difficoltà a garantire un'elevata efficienza di cattura della CO₂, prestazione indesiderata soprattutto in presenza di metano, il cui scarso recupero costituirebbe una perdita energetica ed economica. Per superare questo limite, allo studio di membrane più permeabili e selettive è possibile affiancare un'implementazione del processo attraverso l'utilizzo di più stadi e/o del riciclaggio.

L'obiettivo della presente tesi sarà proprio quello di valutare le diverse possibilità nella progettazione di un sistema di separazione a membrana multistadio. In particolare, i prossimi capitoli si occuperanno di:

- Offrire uno sguardo d'insieme su ciò che, ad oggi, è noto sulla tecnologia a membrana. Si descriveranno quali possono essere i meccanismi di separazione che permettono la realizzazione del processo e quali sono le condizioni affinché si verifichino. Si farà poi una classificazione dei materiali utilizzati come membrane e una descrizione delle loro caratteristiche.
- Nel terzo capitolo ci si soffermerà su cosa è un modulo a membrana, come può essere progettato e in che modo può verificarsi la separazione al suo interno. Si analizzeranno alcune tipologie di flusso (ideale) e le corrispondenti equazioni che descrivono la separazione e permettono di calcolarne le prestazioni in termini di purezza e recupero. Verrà spiegato come i tre modelli di flusso analizzati sono stati implementati in MATLAB® per calcolare la separazione all'interno di un modulo a membrana sotto determinate condizioni. Si dimostrerà infine come un solo stadio di separazione non basta a soddisfare le richieste di prodotto.
- Si illustreranno nel quarto capitolo quali sono gli schemi multistadio considerati. In particolare, verranno esaminati solo schemi con due stadi di separazione, in diverse condizioni di pressione, con o senza ricircolo. Si esporranno quali possono essere i vantaggi per ciascuna tipologia e gli obiettivi che ci si propone di soddisfare per una progettazione ottimizzata del sistema.
- Si spiegherà la teoria alla base di un'ottimizzazione multiobiettivo per la progettazione ottimale di un processo. Nel quinto capitolo si esamineranno alcuni tra gli algoritmi di ottimizzazione presenti in funzioni MATLAB® e come si è deciso di utilizzarli nel trovare le condizioni operative migliori per gli schemi descritti nel capitolo 4.
- Infine, nel sesto capitolo, il codice di ottimizzazione verrà utilizzato in due casi studio: la separazione della CO₂ da un gas di scarico di una centrale termoelettrica, e l'upgrading del biogas a biometano. Si presenteranno i risultati dell'ottimizzazione e si delinea lo schema e le condizioni ottimali per il sistema di separazione nel caso esaminato.

2. Separazione di CO₂ con membrana: stato dell'arte

Realizzare una qualsiasi separazione necessita innanzitutto che venga fornita dell'energia. Il valore minimo di questa energia deve essere almeno pari o superiore all'entalpia libera di miscelazione del sistema. In pratica, dovrà essere di molto superiore. Esiste una grande varietà di tecnologie in grado di realizzare una separazione e ognuna di esse richiede una quantità di energia diversa. I due criteri generali che si applicano per la scelta di una tecnologia sono la fattibilità tecnica ed economica.

L'utilizzo delle membrane nei processi di separazione è una tecnica relativamente recente e ad oggi sono utilizzate in un'ampia gamma di applicazioni. Tra i processi a membrana di prima generazione troviamo l'osmosi inversa, la microfiltrazione, l'ultrafiltrazione, la nanofiltrazione, l'elettrodialisi, l'elettrolisi con membrana; la separazione di gas fa invece parte di quei processi di seconda generazione, insieme alla pervaporazione, la distillazione con membrana e i contattori a membrana. Ciò che tutti questi processi hanno in comune è l'utilizzo della membrana, questa barriera semipermeabile o interfaccia tra due fasi, che ha la capacità di trasportare un componente più facilmente di qualsiasi altro nella miscela alimentata. Dalla portata di alimentazione si otterranno due flussi, un flusso concentrato (il retentato) e un flusso che ha attraversato la membrana (il permeato) (FIGURA 2.1).

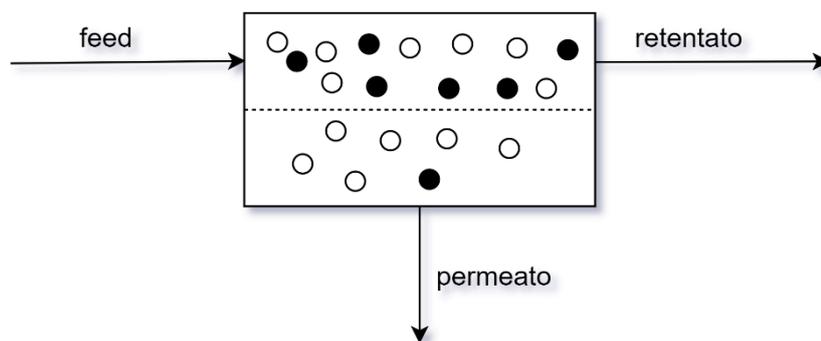


Figura 2.1 Rappresentazione schematica di un sistema bifasico separato da una membrana

Quando si parla di membrane ci si può riferire a un materiale di diversa natura e morfologia, le cui caratteristiche influenzano il trasporto di materia. Ci sono diversi modi per classificarle [11]. Una prima differenziazione può essere fatta tra membrane di origine naturale e membrane sintetiche, che si distinguono sia per struttura che per funzionalità. A loro volta le membrane sintetiche possono essere suddivise in membrane inorganiche (metalliche o ceramiche) e membrane organiche (liquide o polimeriche). Una seconda classificazione può essere fatta sulla base della morfologia e della struttura, che determina il meccanismo di separazione e quindi anche il contesto in cui la membrana può essere applicata. Considerando le sole membrane sintetiche solide, di nostro interesse, si può distinguere tra membrane simmetriche (porose o non porose) e membrane asimmetriche. Le prime hanno generalmente uno spessore compreso tra 10 e 200 μm , da cui dipende la resistenza al trasporto di materia: minore è lo spessore maggiore sarà la velocità di permeazione. Le membrane asimmetriche sono invece costituite da un sottile strato tra 0,1-0,5 μm di materiale polimerico denso (quindi non poroso) depositato su di un supporto poroso di spessore tra i 50 e i 150 μm . Questo tipo di membrane hanno costituito un buon punto di svolta per lo sviluppo in applicazioni industriali. In tal modo si combina la buona selettività dello strato denso con l'alta velocità di permeazione del substrato, la cui

resistenza al trasporto di massa può essere considerata trascurabile. Un ulteriore sviluppo si è avuto nella sintesi di membrane composite, costituite quindi da due materiali polimerici differenti. In questo caso si tratta di una membrana asimmetrica rivestita da un sottile strato polimerico denso.

Il trasporto attraverso la membrana è il risultato di una forza spingente che viene applicata sull'alimentazione. In molti casi la relazione tra il flusso attraverso la membrana e la forza spingente può essere descritta da un'equazione fenomenologica lineare (2.1), in cui κ è il coefficiente di proporzionalità o coefficiente fenomenologico, J è il flusso e la forza spingente viene espressa come gradiente della proprietà ψ (sia essa temperatura, pressione o concentrazione) lungo la coordinata X , perpendicolare alla superficie della membrana.

$$J = -\kappa \frac{d\psi}{dX} \quad (2.1)$$

In questi termini la membrana è vista come una *black-box* e il problema viene considerato da un punto di vista macroscopico. Non sappiamo nulla su come la natura fisica e chimica della membrana influisca sul trasporto, ma il coefficiente A è una misura della resistenza alla diffusione esercitata dalla membrana quando una forza esterna viene applicata.

Nel caso specifico della separazione di componenti gassosi, la forza spingente è data dalla differenza di concentrazione tra un lato e l'altro della membrana, spesso espressa in termini di pressione parziale. In questo tipo di applicazione possono essere utilizzati due tipi di membrane, caratterizzate da meccanismi di trasporto completamente diversi: membrane dense, in cui il trasporto avviene per il meccanismo di soluzione-diffusione, o membrane porose, in cui si verifica il flusso di Knudsen (FIGURA 2.2).

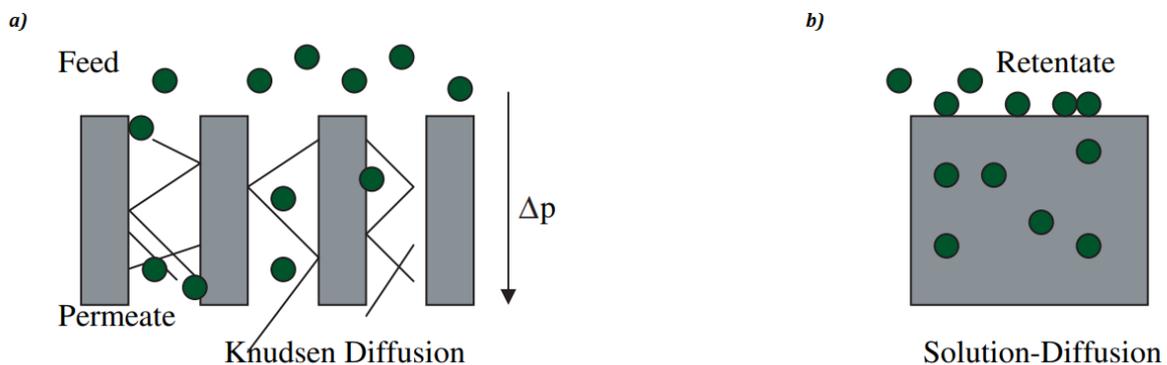


Figura 2.2 Rappresentazione schematiche di due possibili meccanismi per la separazione di gas: diffusione di Knudsen (a) e meccanismo di soluzione-diffusione (b) [9]

2.1 Meccanismi di trasporto

In caso di membrane porose il trasporto avviene attraverso i pori della membrana, e parametri come la porosità, la dimensione dei pori e la distribuzione delle dimensioni dei pori sono da tenere in considerazione per lo sviluppo del modello. In questo tipo di membrane la selettività è determinata dalla differenza di dimensione tra i pori della membrana e le particelle del gas.

Il principio su cui si basa il trasporto nelle membrane dense, invece, è completamente diverso. Le molecole di gas devono innanzitutto dissolversi sulla superficie della membrana, per poi essere trasportate da un lato all'altro della membrana per diffusione, grazie all'applicazione di una forza esterna (pressione). La selettività dipende, quindi, dall'affinità tra membrana e permeato, da differenze nella solubilità e/o diffusività dei componenti.

2.1.1 Trasporto in membrane porose

Le membrane asimmetriche o composite sono caratterizzate da un cambiamento della morfologia lungo il loro spessore. Nel passaggio del gas da uno strato polimerico denso a uno strato poroso possono distinguersi diversi meccanismi di trasporto (FIGURA 2.3):

- Meccanismo di soluzione-diffusione nello strato denso (non poroso)
- Flusso di Knudsen nei pori stretti
- Flusso viscoso nei pori larghi
- Diffusione superficiale lungo le pareti dei pori

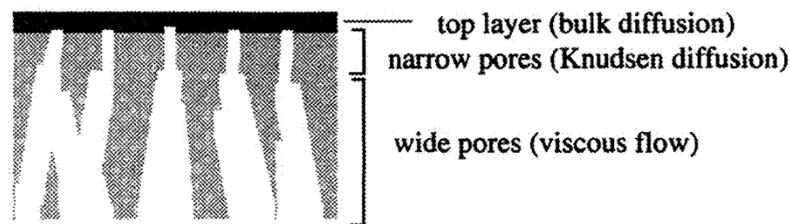


Figura 2.3 Trasporto in una membrana asimmetrica come risultato di diversi meccanismi: diffusione nello strato superficiale non poroso; diffusione di Knudsen nei pori stretti; flusso viscoso nei pori larghi [11]

È principalmente il trasporto nello strato denso a determinare la velocità di permeazione, tuttavia anche il substrato può mostrare una certa resistenza al trasporto. Infatti è noto che la porosità superficiale è generalmente molto bassa, inferiore all'1%. Questo può portare ad avere uno spessore effettivo maggiore di quello del solo strato denso, e conseguentemente una maggiore resistenza al trasporto. Se una molecola penetra nel punto A, come mostrato in FIGURA 2.4, lo spessore effettivo diventa:

$$l_{eff} = \epsilon l_0 + (1 - \epsilon) \frac{l_A + l_0}{2} \quad (2.2)$$

quindi dipende dalla porosità superficiale del substrato. Per il calcolo della resistenza al trasporto sarà allora necessario determinare la distribuzione delle dimensioni dei pori.

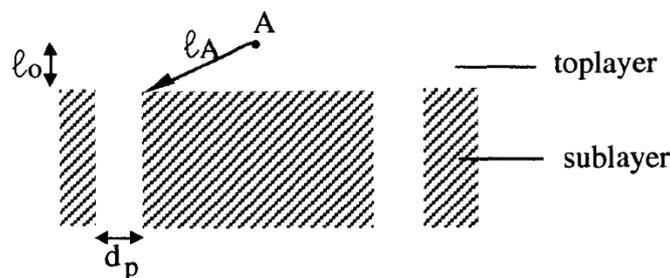


Figura 2.4 Rappresentazione schematica dei percorsi di diffusione in una membrana composta [11]

Il verificarsi della diffusione di Knudsen o della diffusione viscosa dipende essenzialmente dalla dimensione dei pori e dal cammino libero delle molecole. Per pori con raggio maggiore di 10 [µm] prevalgono gli urti tra le molecole, che si comportano come in assenza di membrana e non avviene alcuna separazione. Per raggi minori, invece, le molecole collidono

frequentemente con le pareti del poro e si instaura la diffusione di Knudsen. In altri termini, quando il cammino libero delle molecole è inferiore alle dimensioni del poro si ha flusso viscoso, al contrario si avrà diffusione di Knudsen. Il cammino libero delle molecole dipende sia dalla temperatura che dalla pressione operativa, in particolare è proporzionale alla temperatura a pressione costante, ed aumenta al diminuire della pressione. Il flusso di Knudsen può essere espresso con l'EQUAZIONE 2.3:

$$J_k = \frac{\pi n r^2 D_k \Delta p}{R T \tau l} \quad (2.3)$$

con il coefficiente di diffusione di Knudsen si calcola come nell'EQUAZIONE 2.4:

$$D_k = 0.66 r \sqrt{\frac{8 R T}{\pi M_w}} \quad (2.4)$$

Per una certa membrana a una data pressione e temperatura, la separazione di due gas nel regime di flusso di Knudsen dipende dalla radice quadrata del rapporto tra i rispettivi pesi molecolari. Si può intuire che in genere si ottengono dei fattori di separazione molto bassi e il processo non risulta economicamente conveniente. L'utilizzo di membrane porose nella separazione di gas ha senso solo all'interno di membrane asimmetriche o composite.

2.1.2 Trasporto in membrane non porose

Se le molecole da separare hanno dimensioni comparabili, allora una membrana porosa non offre alcuna utilità. È necessario utilizzare membrane non porose, il cui trasporto al loro interno può essere descritto con un meccanismo di soluzione-diffusione. In queste membrane le porosità sono presenti a livello molecolare affinché sia reso possibile il passaggio del gas. Per i sistemi non interagenti come quelli che si prenderanno in esame, acquista importanza nelle proprietà di trasporto il concetto di volume libero. Il volume libero può essere semplicemente inteso come il volume della membrana disponibile, non occupato dalle macromolecole. Dipende da parametri fisici della membrana come la densità o il volume specifico e specialmente nel caso di membrane polimeriche varia a seconda della temperatura di utilizzo. Può essere definito come il volume generato dall'espansione termica delle macromolecole che a 0 [K] sono chiuse su loro stesse (2.5).

$$V_f = V_T - V_0 \quad (2.5)$$

V_T è il volume misurato alla temperatura T e V_0 il volume occupato dalle macromolecole a 0 [K]. Spesso si utilizza il volume libero frazionario v_f , dato dal rapporto tra il volume libero e il volume osservato alla temperatura T (2.6).

$$v_f = \frac{V_f}{V_T} \quad (2.6)$$

La teoria del volume libero spiega le diverse proprietà di permeazione che differenziano le membrane polimeriche elastomeriche da quelle vetrose, come sarà possibile osservare nei prossimi paragrafi. Allo stato vetroso, quindi quando la temperatura operativa è inferiore alla temperatura di transizione vetrosa T_g caratteristica del polimero, le macromolecole hanno mobilità estremamente limitata e l'energia termica non è sufficiente a permettere la rotazione dei segmenti della catena attorno alla catena principale. In questa condizione il volume libero frazionario è praticamente costante (FIGURA 2.5). Nello stato gommoso, al di sopra della temperatura di transizione vetrosa, le molecole acquistano maggiore libertà di movimento e il volume libero frazionario aumenta linearmente con la temperatura secondo l'EQUAZIONE 2.7.

$$v_f = v_{f,T_g} + \Delta\lambda(T - T_g) \quad (2.7)$$

dove $\Delta\lambda$ è la differenza del coefficiente di espansione termica al di sopra e al di sotto della temperatura di transizione vetrosa.

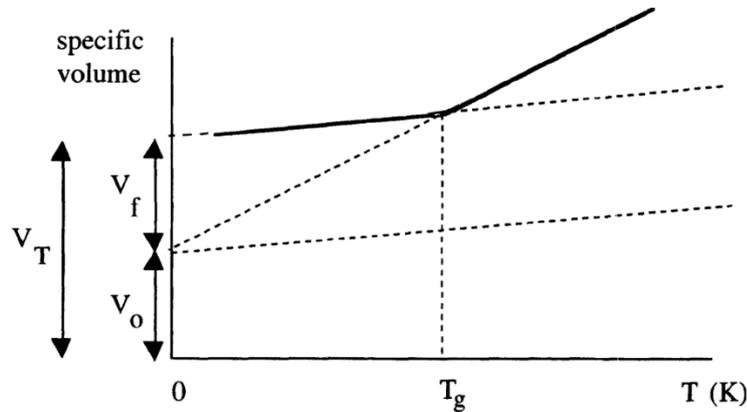


Figura 2.5 Volume specifico di un polimero amorfo in funzione della temperatura [11]

Secondo il meccanismo di soluzione-diffusione, il gas alimentato nella zona di *upstream* della membrana, a pressione maggiore p_h , dissolve sulla superficie della membrana lato *feed* e diffonde lungo il suo spessore grazie ad un gradiente di concentrazione, per poi desorbirsi dalla superficie della membrana del lato *downstream*, a bassa pressione p_l (FIGURA 2.6). Entrano quindi in gioco due importanti parametri: la solubilità e la diffusività.

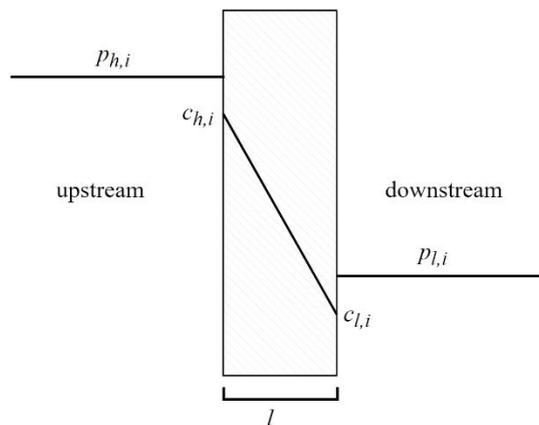


Figura 2.6 Rappresentazione schematica del trasporto in una membrana non porosa

2.1.2.1 Solubilità

La solubilità è un parametro termodinamico che esprime la quantità di gas assorbito sulla membrana in condizioni di equilibrio. In generale la solubilità dei gas nei polimeri è molto bassa. Nel caso più semplice di un sistema ideale, in cui la solubilità è indipendente dalla concentrazione, la solubilità può essere descritta dalla legge di Henry e l'isoterma di assorbimento è lineare (FIGURA 2.7A). In altri termini, la concentrazione di permeante che si assorbe sul polimero è proporzionale alla pressione applicata, e il coefficiente di

proporzionalità, la costante di Henry, corrisponde alla solubilità (2.8). Questo è il tipico caso dei polimeri elastomerici.

$$c = k_d \cdot p = s \cdot p \quad (2.8)$$

L'isoterma di assorbimento ha invece un andamento che si discosta dalla linearità nel caso di polimeri vetrosi. In questo caso l'assorbimento viene descritto dalla teoria *dual sorption*, secondo cui è dato dalla combinazione di due meccanismi: la legge di Henry e l'assorbimento di tipo Langmuir (FIGURA 2.7). La concentrazione di gas assorbito sul polimero è data dall'EQUAZIONE 2.9.

$$c = k_d p + \frac{c'_h b p}{1 + b p} \quad (2.9)$$

in cui b è la costante di affinità di foro [Pa^{-1}] e c'_h è la costante di saturazione [$\text{m}_{\text{STP}}^3/\text{m}^3$].

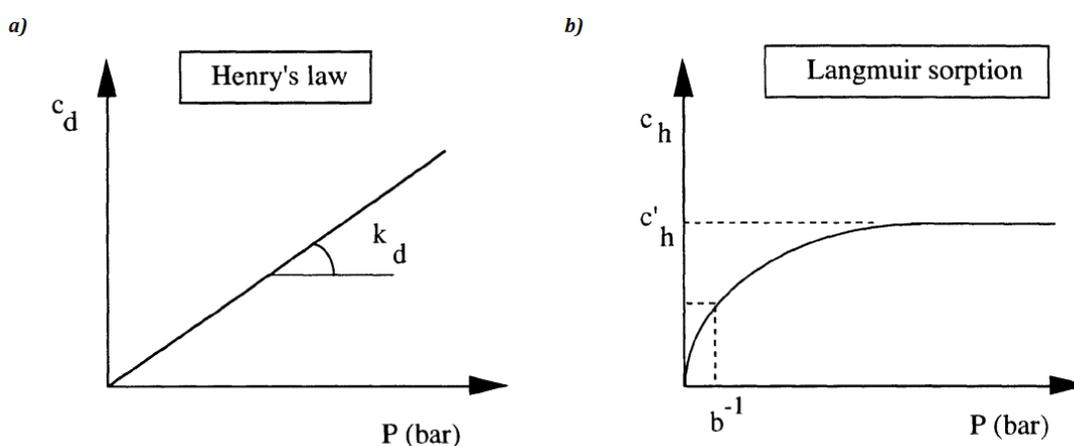


Figura 2.7 I due contributi della teoria *dual sorption*: la legge di Henry (a) e l'assorbimento di Langmuir (b) [11]

L'unità di misura del coefficiente di solubilità è [$\text{m}_{\text{STP}}^3/(\text{m}^3\text{Pa})$]. È determinato principalmente dalla facilità di condensazione del gas: con l'aumentare del diametro molecolare aumenta la facilità di condensazione e quindi anche la solubilità. Un indicatore della facilità a condensare è la temperatura critica del gas. In TABELLA 2.1 si confrontano i coefficienti di solubilità di diversi gas, accompagnati dalla rispettiva temperatura critica, in riferimento alla gomma naturale. In TABELLA 2.2 si riportano invece a titolo di esempio i coefficienti di solubilità della CO_2 per diversi polimeri elastomerici e un paio di polimeri vetrosi. La solubilità di un gas è maggiore per i polimeri con cui ha una migliore affinità.

Tabella 2.1 Temperatura critica T_c e coefficiente di solubilità s di vari gas in gomma naturale [11]

gas	T_c [K]	s [$\text{cm}_{\text{STP}}^3/(\text{cm}^3 \cdot \text{cmHg})$]
H_2	33.3	0.0005
N_2	126.1	0.0010
O_2	154.4	0.0015
CH_4	190.7	0.0035
CO_2	304.2	0.0120

Tabella 2.2 Solubilità della CO₂ riferita a polimeri elastomerici (25°C) [12] e polimeri vetrosi (PSF e PSF-M a 35°C) [13]

<i>Membrana</i>	$s_{CO_2} [cm_{STP}^3/(cm^3 \cdot atm)]$
Gomma naturale	0.90
Buna S	0.87
Perbunan	1.28
Neoprene G	0.83
Oppanol B 200	0.69
PSF	2.1
PSF-M	1.1

2.1.2.2 Diffusività

La diffusività è un parametro cinetico che dà una misura della velocità con cui un certo componente riesce ad attraversare la membrana. Può essere vista come un trasporto molecolare statistico conseguenza del moto casuale delle molecole. Macroscopicamente, si verifica un flusso a causa di una differenza di concentrazione. Il trasporto di gas attraverso una membrana può essere semplicemente descritto dalla prima legge di Fick (2.10). Il flusso di gas che attraversa la superficie della membrana è proporzionale al gradiente di concentrazione lungo la direzione del flusso, perpendicolare alla superficie della membrana stessa. Si considera che il gradiente di concentrazione sia non nullo solo lungo la direzione x e che quindi la diffusione è monodimensionale. Il coefficiente di proporzionalità è il coefficiente di diffusività ed ha come unità di misura [m²/s].

$$J = -D \frac{dc}{dx} \quad (2.10)$$

La variazione nel tempo della concentrazione di gas all'interno della membrana è chiaramente legata al flusso che la attraversa lungo il suo spessore (2.11).

$$\frac{\partial c}{\partial t} = -\frac{\partial J}{\partial x} = -\frac{\partial}{\partial x} \left(-D \frac{\partial c}{\partial x} \right) \quad (2.11)$$

Nella separazione di gas, in cui l'interazione gas-polimero è molto bassa, la diffusività si può considerare indipendente dalla concentrazione ed è possibile portarla fuori dal termine di derivazione, ottenendo quella che è comunemente conosciuta come la seconda legge di Fick (2.12).

$$\frac{\partial c}{\partial t} = D \frac{\partial^2 c}{\partial x^2} \quad (2.12)$$

Il coefficiente di diffusività dipende sia dalla dimensione delle molecole diffondenti che dalla natura del mezzo attraverso cui si ha la diffusione. In generale, maggiore è il diametro molecolare minore è il coefficiente di diffusività. Infatti, secondo la teoria del volume libero, una molecola può diffondere all'interno della membrana solo se c'è abbastanza volume libero. Un incremento delle dimensioni della molecola penetrante deve comportare anche un aumento del volume libero. La velocità di diffusione di un certo penetrante dipende allora dalla probabilità della molecola di trovare uno spazio libero che abbia dimensioni superiori alle proprie. Il coefficiente di diffusività termodinamico può essere messo in relazione al volume libero frazionario secondo l'EQUAZIONE 2.13.

$$D_T = R T A_f \exp \left(-\frac{B}{v_f} \right) \quad (2.13)$$

A_f dipende dalla forma e la dimensione della molecola penetrante e B esprime il minimo volume libero locale necessario al penetrante. Per i sistemi ideali il coefficiente di diffusività termodinamico e la diffusività osservata coincidono.

In TABELLA 2.3 si possono osservare i coefficienti di diffusività di diversi gas in riferimento a una membrana in PEEK-WC, polimero vetroso. In TABELLA 2.4 sono invece riportati i coefficienti di diffusività della CO₂ riferiti ad alcune membrane elastomeriche e vetrose.

Tabella 2.3 Coefficienti di diffusività di 6 gas in riferimento al polimero vetroso PEEK-WC, prove effettuate a 25°C [14]

<i>Gas</i>	<i>Diffusività [10⁻¹² m² s⁻¹]</i>
N ₂	0.45
O ₂	2.02
CO ₂	0.58
CH ₄	0.14
H ₂	135
He	529

Tabella 2.4 Diffusività della CO₂ riferita a polimeri elastomerici (25°C) [12] e polimeri vetrosi (PSF e PSF-M a 35°C) [13]

<i>Membrana</i>	<i>D_{CO2} [10⁻⁷ cm²/s]</i>
Gomma naturale	6.7
Buna S	6.8
Perbunan	1.0
Neoprene G	1.3
Oppanol B 200	0.32
PSF	0.2
PSF-M	0.27

2.1.2.3 Permeabilità

La permeabilità è un parametro funzione di solubilità e diffusività e viene spesso utilizzato come indicatore delle prestazioni di una membrana. Per un sistema ideale in condizioni stazionarie, la 2.10 può essere integrata ottenendo l'EQUAZIONE 2.14.

$$J = \frac{D}{l} (c_{h,i} - c_{l,i}) \quad (2.14)$$

La solubilità può essere descritta dalla legge di Henry e l'EQUAZIONE 2.14 può essere riscritta in funzione delle pressioni parziali del componente gassoso i tra il lato ad alta pressione (*upstream*) e il lato a bassa pressione (*downstream*) della membrana (2.15).

$$J = \frac{sD}{l} (p_{h,i} - p_{l,i}) \quad (2.15)$$

Il coefficiente di permeabilità è definito come il prodotto tra il coefficiente di solubilità e il coefficiente di diffusività (2.16) ed è generalmente fornito in [Barrer] (1 Barrer = 10⁻¹⁰ cm_{STP}³·cm·cm⁻²·s⁻¹·cmHg⁻¹ = 0.76·10⁻¹⁷ m_{STP}³·m·m⁻²·s⁻¹·Pa⁻¹). Il flusso di un gas attraverso una

membrana è direttamente proporzionale alla differenza delle relative pressioni parziali e inversamente proporzionale allo spessore della membrana (2.17).

$$P = s D \quad (2.16)$$

$$J = \frac{P}{l} (p_{h,i} - p_{l,i}) \quad (2.17)$$

Questa espressione è valida solo per i sistemi ideali, negli altri casi la permeabilità non si può considerare costante e dipenderà dalla pressione applicata.

Nel caso della permeabilità, non è automatico affermare che molecole più piccole permeano più facilmente, in quanto dipende sia dalla solubilità che dalla diffusività. Molecole molto piccole come elio e idrogeno hanno in generale elevata permeabilità perché prevale l'alta diffusività. Una molecola un po' più grande come l'anidride carbonica ha comunque un'alta permeabilità grazie alla sua buona solubilità. Si possono osservare i valori del coefficiente di permeabilità di diversi gas riferiti ancora alla PEEK-WC in TABELLA 2.5. Inoltre, la permeabilità in un elastomero è in generale maggiore rispetto ad un polimero vetroso, grazie alla migliore mobilità delle sue catene. Tuttavia esistono delle eccezioni. Si riportano i coefficienti di permeabilità della CO₂ riferiti ad alcuni polimeri elastomerici e vetrosi in TABELLA 2.6.

Tabella 2.5 Coefficienti di permeabilità di 6 gas in riferimento al polimero vetroso PEEK-WC, prove effettuate a 25°C [14]

<i>Gas</i>	<i>Permeabilità [Barrer]</i>
N ₂	0.24
O ₂	1.24
CO ₂	6.04
CH ₄	0.25
H ₂	13.4
He	12.5

Tabella 2.6 Temperatura di transizione vetrosa (T_g) e permeabilità della CO₂ (P_{CO2}) per diversi polimeri elastomerici e vetrosi [11]

<i>Polimero</i>	<i>T_g [°C]</i>	<i>P_{CO2} [Barrer]</i>
Gomma siliconica	-110	3200
Gomma naturale	-70	130
Polistirene	100	11
Nylon 6	47	0.16
Lexan	147	10
Polisolfone	188	4.4
Acetato di cellulosa	198	6.0
Ultem	217	1.5

2.1.2.4 Selettività

Il parametro che insieme alla permeabilità definisce l'efficienza di una membrana è la selettività, una misura di quanto una membrana lascia permeare un componente rispetto ad un altro. Si definisce selettività ideale di due componenti il rapporto tra i rispettivi coefficienti di permeabilità (2.18). Si fa in modo che la selettività abbia valore maggiore di 1. La selettività reale solitamente non coincide con quella ideale, a causa di alcuni fattori, come la plasticizzazione del polimero che sfavorisce la selettività in favore della permeabilità, o il rapporto di pressione tra i due lati della membrana. La selettività diminuisce al diminuire della forza spingente.

$$\alpha_{ij,ideal} = \frac{P_i}{P_j} \quad (2.18)$$

Avere una migliore permeabilità comporta spesso una selettività più bassa e per avere una buona efficienza di separazione è necessario trovare il giusto compromesso tra i due parametri. Al contrario di quanto detto per il coefficiente di permeabilità, la selettività tra due polimeri è generalmente maggiore nei polimeri vetrosi rispetto agli elastomeri. La TABELLA 2.7 presenta la selettività CO₂-CH₄ di alcuni polimeri elastomerici e vetrosi.

Tabella 2.7 Temperatura di transizione vetrosa (T_g) e selettività della CO₂ con CH₄ (α_{CO₂/CH₄) per diversi polimeri elastomerici e vetrosi [11]}

<i>Polimero</i>	<i>T_g [°C]</i>	<i>α_{CO₂/CH₄}</i>
Gomma siliconica	-110	3.4
Gomma naturale	-70	4.6
Polistirene	100	8.5
Nylon 6	47	11.2
Lexan	147	26.7
Polisolfone	188	30.0
Acetato di cellulosa	198	31.0
Ultem	217	45.0

2.1.2.5 Effetto della temperatura

Oltre alla pressione, anche la temperatura influenza la velocità di trasporto. La relazione tra il coefficiente di permeabilità e la temperatura può essere espressa con un'equazione di tipo Arrhenius (2.19). Lo stesso vale per il coefficiente di solubilità e per la diffusività, da cui la permeabilità è strettamente dipendente (2.20, 2.21, 2.22).

$$P = P_0 \cdot \exp(-E_p/RT) \quad (2.19)$$

$$s = s_0 \cdot \exp(-\Delta H_s/RT) \quad (2.20)$$

$$D = D_0 \cdot \exp(-E_d/RT) \quad (2.21)$$

$$P = D_0 s_0 \exp\left(-\frac{\Delta H_s + E_d}{RT}\right) = P_0 \exp\left(-\frac{E_p}{RT}\right) \quad (2.22)$$

P₀, s₀ e D₀ sono i fattori preesponenziali, indipendenti dalla temperatura. ΔH_s è l'entalpia di soluzione, data dalla somma dell'entalpia di miscelazione e dell'entalpia di condensazione. Per le molecole gassose non interagenti di nostro interesse, l'entalpia di soluzione è positiva

(processo endotermico) e assume valori molto bassi, a significare che la solubilità crescerà leggermente con la temperatura. E_d è l'energia di attivazione per la diffusione ed è il principale contributo all'energia di attivazione di permeazione E_p . Non è detto che l'energia di attivazione sia esplicitamente correlata alla facilità di permeazione. Per le molecole considerate la dipendenza della permeabilità con la temperatura è strettamente legata alla relazione della diffusività con la temperatura.

2.1.2.6 Effetto di Joule-Thomson

Durante il processo di permeazione il gas è soggetto ad espansione lungo la membrana. Per i gas reali, quest'espansione adiabatica porta ad una variazione di temperatura, più o meno rilevante a seconda del tipo di gas e della pressione applicata. Come conseguenza si verifica una variazione delle proprietà di permeazione. Se ad esempio la temperatura diminuisse si otterrebbe una riduzione del flusso e un aumento della selettività. Questo fenomeno prende il nome di effetto di Joule-Thomson. La variazione di energia interna è data dall'EQUAZIONE 2.23.

$$\Delta U = U_l - U_h = -p_l V_l + p_h V_h \quad (2.23)$$

In altri termini, il processo è isoentalpico (2.24).

$$H_h = U_h + p_h V_h = U_l + p_l V_l = H_l \quad (2.24)$$

La variazione di temperatura può quindi essere espressa come $(\partial T / \partial p)_H$, nota come coefficiente di Joule-Thomson μ_{JT} [11]. Si parte dal differenziale dell'entalpia funzione di temperatura e pressione, che si pone pari a 0 (2.25), trovando l'espressione (2.26).

$$dH = \left(\frac{\partial H}{\partial p}\right)_T dp + \left(\frac{\partial H}{\partial T}\right)_p dT \quad (2.25)$$

$$\left(\frac{\partial T}{\partial p}\right)_H = -\left(\frac{\partial H}{\partial p}\right)_T \left(\frac{\partial T}{\partial H}\right)_p \quad (2.26)$$

Di cui si sa che

$$\left(\frac{\partial H}{\partial T}\right)_p = C_p \quad (2.27)$$

Per un processo reversibile, la variazione di entalpia si può scrivere secondo la (2.28) per ricavare la (2.29):

$$dH = V dp + T dS \quad (2.28)$$

$$\left(\frac{\partial H}{\partial p}\right)_T = V + T \left(\frac{\partial S}{\partial p}\right)_T \quad (2.29)$$

Per la relazione di Maxwell è valida la (2.30) e si può ottenere l'espressione che regola il coefficiente di Joule-Thomson (2.31).

$$-\left(\frac{\partial S}{\partial p}\right)_T = \left(\frac{\partial V}{\partial T}\right)_p \quad (2.30)$$

$$\left(\frac{\partial T}{\partial p}\right)_H = \mu_{JT} = -\frac{1}{C_p} \left[V - T \left(\frac{\partial V}{\partial T}\right)_p \right] \quad (2.31)$$

Quindi l'aumento o decremento della temperatura è determinato dalla grandezza relativa dei due contributi. In TABELLA 2.8 sono forniti a titolo di esempio i valori del coefficiente di Joule-Thomson per alcuni gas. Se ne deduce che bisogna porre particolare attenzione al verificarsi di

questo effetto quando si ha a che fare con la rimozione della CO₂ ad alte pressioni, per evitare che si abbia un peggioramento delle prestazioni di separazione.

Tabella 2.8 Coefficienti di Joule-Thomson per alcuni gas a 1 [bar] e 298 [K] [11]

<i>gas</i>	μ_{JT} [K/bar]
He	-0.06
CO	0.01
H ₂	0.03
O ₂	0.30
N ₂	0.25
CH ₄	0.70
CO ₂	1.11

2.2 Materiali

Nel corso degli anni sono state sintetizzate e testate membrane di diversa natura per la cattura della CO₂, dai materiali inorganici (carbonio, zeoliti, ceramica, metalli) a quelli polimerici, per arrivare a una combinazione dei due nelle cosiddette membrane a matrice mista (MMM, *Mixed Matrix Membranes*). La scelta del materiale è fondamentale per ottenere delle buone *performance* di separazione, in quanto l'affinità del gas con la membrana determina parametri importanti come la permeabilità e la selettività. Si esploreranno di seguito i principali materiali tuttora esistenti finalizzati alla separazione dell'anidride carbonica, con annesse le relative caratteristiche.

2.2.1 Membrane inorganiche

Le membrane inorganiche possono essere dense o porose, ma solo queste ultime sono di interesse per la separazione della CO₂. È possibile depositare il materiale inorganico microporoso su di un supporto, a sua volta poroso, in grado di migliorarne la resistenza meccanica senza influire negativamente sulle proprietà di trasporto. Il supporto può essere scelto in materiale ceramico, come silice, allumina, zirconia, titania, oppure metallico, ad esempio l'acciaio inossidabile. Il vantaggio delle membrane inorganiche risiede nella loro resistenza alle alte temperature e alle impurità, utile nel trattamento dei gas di scarico. Tuttavia sono particolarmente soggette alla presenza di umidità (adsorbimento competitivo) e in genere non si utilizzano a temperature superiori ai 200°C per non avere selettività troppo basse. I fattori determinanti le prestazioni delle membrane inorganiche porose sono la dimensione dei pori (rispetto alla molecola penetrante), la porosità, la tortuosità, il calore di adsorbimento (che indica l'affinità con la specie penetrante), l'area superficiale specifica e il suo spessore. Pur essendo le più economiche non offrono delle efficienze particolarmente elevate, per cui si sono sviluppate nel tempo diverse tecniche per migliorarne le caratteristiche. Strumenti e tecniche di lavorazione avanzate permettono una micro e nano-strutturazione per regolare la dimensione dei pori e la loro distribuzione. La superficie può anche essere sottoposta a funzionalizzazione, spesso con gruppi amminici, che abbiano una migliore affinità della CO₂ e ne migliorino la capacità di adsorbimento [5].

2.2.1.1 Membrane a base di carbonio

I materiali adsorbenti a base di carbonio sono stati ampiamente utilizzati industrialmente per diverse applicazioni, vista la loro ampia disponibilità, economicità, stabilità termica e natura idrofobica. Questo tipo di membrane vengono generalmente ottenute tramite pirolisi di composti organici o polimerici a temperature tra i 500 e i 1000°C. A seconda del precursore o delle condizioni di pirolisi possono essere ottenute delle microstrutture (nanotubi a parete singola o multipla, fibre cave, setacci molecolari) con una distribuzione della dimensione dei pori interni tra meno di 1 [nm] e diversi nanometri. Queste microstrutture possono essere supportate da dei materiali porosi inerti, polimerici o ceramici come zirconia, titania, allumina. La selettività nei confronti della CO₂ non è altissima (per il carbonio attivo il calore di adsorbimento è di -30 kJ/mol) e peggiore all'aumentare della temperatura operativa [5].

2.2.1.2 Membrane a base di silice

Le membrane porose di silice amorfa possono essere sintetizzate tramite il metodo idrotermale o sol-gel, che permette di ottenere una certa controllabilità e omogeneità nella dimensione e distribuzione dei pori. La possibilità di ottenere degli strati di membrana di basso spessore, tra i 50 e i 200 nm, permette di avere delle buone velocità di permeazione. Tuttavia le membrane a base di silice non costituiscono la scelta migliore in termini di affinità con la CO₂, e vengono spesso utilizzate solo come supporto per membrane più attive e selettive. Per fare da fase attiva, le membrane porose di silice possono essere sottoposte a una funzionalizzazione con ammine, che ne migliora la capacità di adsorbimento e di separazione della CO₂. Questo tipo di membrane riescono a raggiungere anche selettività fino a 800 per miscele CO₂-N₂, tuttavia sono particolarmente soggette all'assorbimento competitivo con il vapore acqueo e non costituiscono mai la prima scelta per questo tipo di applicazione [5, 9].

2.2.1.3 Membrane a base di zeolite

Le membrane zeolitiche sono delle strutture inorganiche cristalline caratterizzate da una distribuzione uniforme e ordinata dei pori. Sono sintetizzate con processo idrotermale su supporti porosi in acciaio inossidabile o allumina. Sono state testate membrane a pori piccoli, medi e grandi, principalmente per la separazione della CO₂ nel gas naturale. CO₂ e CH₄ hanno dei diametri cinetici molto simili, rispettivamente di 0.33 e 0.38 [nm], più piccoli dei diametri delle zeoliti a pori medi e grandi, per cui la loro separazione non si basa su una discriminazione dimensionale ma piuttosto sull'adsorbimento competitivo delle due specie. Per la maggior parte delle zeoliti, i calori di adsorbimento seguono l'ordine CO₂ > N₂ > CH₄, secondo l'intensità della forza di interazione elettrostatica tra la molecola gassosa e la zeolite. Si comprende quindi che la separazione della CO₂ dall'azoto è più complicata, e per migliorarne la selettività anche in questo caso si può ricorrere a una funzionalizzazione amminica o a una modifica chimica e strutturale della zeolite mediante cationi idonei. Le selettività più alte riscontrate, in riferimento a una miscela CO₂N₂ 50:50 a temperatura ambiente, sono di 70 e 200 per la faujasiti e le zeoliti di tipo T, rispettivamente. Con le zeoliti di tipo T si è anche dimostrata una selettività pari a 400 per una miscela CO₂-CH₄ a 35°C, con una differenza di pressione tra i due lati della membrana di 0.1 MPa e il lato permeato in sottovuoto. In ogni caso l'adsorbimento preferenziale della CO₂ è favorito alle basse temperature, dopodiché aumentando la temperatura la selettività della CO₂ nei confronti di N₂ e CH₄ diminuisce, così come l'influenza dell'umidità nell'adsorbimento competitivo [5, 9].

2.2.1.4 Membrane a base di allumina

L'allumina è un materiale con una buona resistenza termica, chimica e idrotermica, quindi particolarmente ideale per le applicazioni ad elevata temperatura. Se scelta come fase attiva, vengono sintetizzate membrane di allumina mesoporose, con le dimensioni dei pori dell'ordine dei nanometri, in modo che la separazione avvenga secondo meccanismo di Knudsen. Tuttavia l'affinità dell'allumina con la CO₂ è molto bassa e spesso si preferisce usarla come supporto per migliorare la resistenza meccanica, termica e chimica della membrana [5].

2.2.2 Membrane polimeriche

Le membrane polimeriche sintetizzate come membrane dense, in cui quindi il trasporto avviene con il meccanismo di soluzione-diffusione. Possono essere suddivise in due grandi categorie: le membrane polimeriche gommose, che operano ad una temperatura al di sopra della loro temperatura di transizione vetrosa, e le membrane polimeriche vetrose che al contrario vengono utilizzate a una temperatura al di sotto della loro temperatura di transizione vetrosa. Nelle membrane gommose, le macromolecole hanno una maggiore libertà di movimento, sono in grado di riarrangiarsi in tempi ragionevoli, raggiungendo l'equilibrio termodinamico. Per queste membrane vale la legge di Henry per la solubilità e in generale offrono una buona permeabilità e una scarsa selettività. Le membrane polimeriche vetrose hanno caratteristiche opposte: la mobilità delle macromolecole è fortemente limitata e non riescono mai a raggiungere l'equilibrio termodinamico. Le catene sono arranziate in una maniera imperfetta, formando dei microvuoti all'interno della matrice polimerica che portano ad avere l'adsorbimento di Langmuir, con un conseguente aumento della solubilità. I polimeri vetrosi offrono in generale un'elevata selettività ma una bassa permeabilità. Lo scopo delle ricerche è quello sintetizzare delle membrane polimeriche più performanti e selettive nei confronti della CO₂, agendo sia sulla composizione per migliorarne la solubilità, che sulla disposizione delle catene all'interno della membrana, per favorire la diffusione. Ci sono alcuni aspetti critici da considerare: il gas da trattare deve essere privato delle impurità e dell'umidità (il vapore acqueo permea più facilmente della CO₂) e la temperatura operativa in genere non può superare i 100°C, vista la scarsa stabilità termica dei polimeri.

2.2.2.1 Polimeri vetrosi

I polimeri vetrosi sono particolarmente apprezzati grazie alla loro resistenza meccanica, basso costo e buona riproducibilità, nonostante siano soggetti a corrosione e a una scarsa resistenza termica. Un'altra criticità è il fenomeno di rigonfiamento (o *swelling*) del polimero, dovuto alla sua plasticizzazione durante la lenta diffusione delle molecole di CO₂ attraverso la matrice. Si avrà un basso grado di rigonfiamento se le interazioni polimero-polimero sono molto forti, e ciò dipende dal grado di cristallinità e dalla presenza di *crosslinking* o legami idrogeno; se invece prevalgono le interazioni tra il polimero e la CO₂ il grado di rigonfiamento sarà alto, con un peggioramento delle prestazioni. Tra i polimeri vetrosi più utilizzati si trovano il polisolfone (PSF), il polietersolfone (PES), la poliimmide (PI), la polieterimmide (PEI), ma sono tutti soggetti al fenomeno dell'invecchiamento che ne peggiora le prestazioni. Tra questi il PSF ha una buona resistenza chimica e meccanica ed in più è molto economico, viene spesso scelto per la separazione di miscele CO₂-CH₄ grazie all'interazione della CO₂ col gruppo sulfonile. Molto nota è la Matrimid®5218, un PI semicristallino con una temperatura di transizione vetrosa tra 305 e 315°C. La Matrimid®5218 ha una buona resistenza termica, meccanica e chimica, e il suo basso volume libero la rende ideale per la separazione di miscele CO₂-CH₄, con una buona selettività anche se una bassa permeabilità della CO₂. Le sue *performance* possono esser

migliorate attraverso la combinazione del polimero con altri materiali, come le strutture organiche metalliche (MOF). La Matrimid può a sua volta essere utilizzata per migliorare le proprietà di un altro tipo di polimeri, limitandone l'invecchiamento. Si tratta dei polimeri a microporosità intrinseca (PIMs), la cui matrice è una rete di piccoli pori interconnessi di circa 2 [nm] di diametro, caratterizzata da un elevato volume libero che permette di avere un'alta permeabilità della CO₂. Ancora molto competitivo rispetto ai tradizionali PI è il polietere etere chetone solfonato (PEEK-WC). Le proprietà di queste membrane polimeriche vetrose per la separazione di miscele CO₂-CH₄ sono presentate in TABELLA 2.9.

Tabella 2.9 Proprietà nella separazione di CO₂ per alcuni polimeri vetrosi [15]

<i>Polimeri</i>	<i>P_{CO2} [Barrer]</i>	<i>α_{CO2/CH4}</i>
6FDA-durene	~ 458	~ 16
6FDA-TMPDA	~ 556	~ 23
Matrimid®5218	~ 10	~ 36
PIM-1	~ 2300	~ 18
PIM-7	~ 1100	~ 26
PES/PHE	~ 1	~ 46
Polidimetilsilossano	~ 3800	~ 3

2.2.2.2 Polimeri gommosi

La flessibilità delle catene polimeriche favorisce l'affinità del polimero con la CO₂. Tra le membrane polimeriche gommose troviamo polimeri a base di poli(ossido di etilene) (PEO), la polivinilammina (PVAm), il poli(dimetil-silossano) (PDMS) e il poli(ammide-6-b-etilene ossido) (Pebax). Questi ultimi due particolarmente noti per la separazione della CO₂. Il Pebax®1657 in particolare è un copolimero che combina le proprietà elastiche dei blocchi di polietere con i blocchi di poliammide che aumentano l'affinità per la CO₂. La funzionalizzazione amminica della Pebax®1657 ha inoltre permesso di raggiungere una permeabilità della CO₂ di 100 [Barrer], con una selettività rispettivamente di 50 e 17 in riferimento all'azoto e al metano. Sono state testate poi diverse possibilità per l'ottenimento di membrane ibride o MMMs, anche con l'inserimento di strutture organiche metalliche. Le proprietà per la separazione della CO₂ di alcune membrane polimeriche gommose sono raccolte in TABELLA 2.10.

Tabella 2.10 Proprietà per la separazione di CO₂ di alcune membrane polimeriche gommose [15]

<i>Polimero</i>	<i>Miscela</i>	<i>P_{CO2} [Barrer]</i>	<i>α</i>
Gomma naturale	CO ₂ -CH ₄	~ 134	~ 5
Polietilene	CO ₂ -CH ₄	~ 13	~ 4
PDMS	CO ₂ -N ₂	~ 2000	~ 34
Pebax®1657	CO ₂ -CH ₄	~ 100	~ 20
Pebax®1657	CO ₂ -N ₂	~ 100	~ 78
Pebax	CO ₂ -CH ₄	~ 98	~ 16

2.2.3 Biopolimeri: un nuovo approccio

L'utilizzo di biopolimeri come sostituti ai polimeri di origine fossile nelle applicazioni con membrana è stato esplorato negli ultimi due decenni. È un'opzione che potrebbe risultare più sostenibile dal punto di vista ambientale, grazie alla loro biodegradabilità, biocompatibilità e compostabilità. I biopolimeri possono essere classificati in sei gruppi, come mostrato in FIGURA 2.8, sulla base della loro origine.

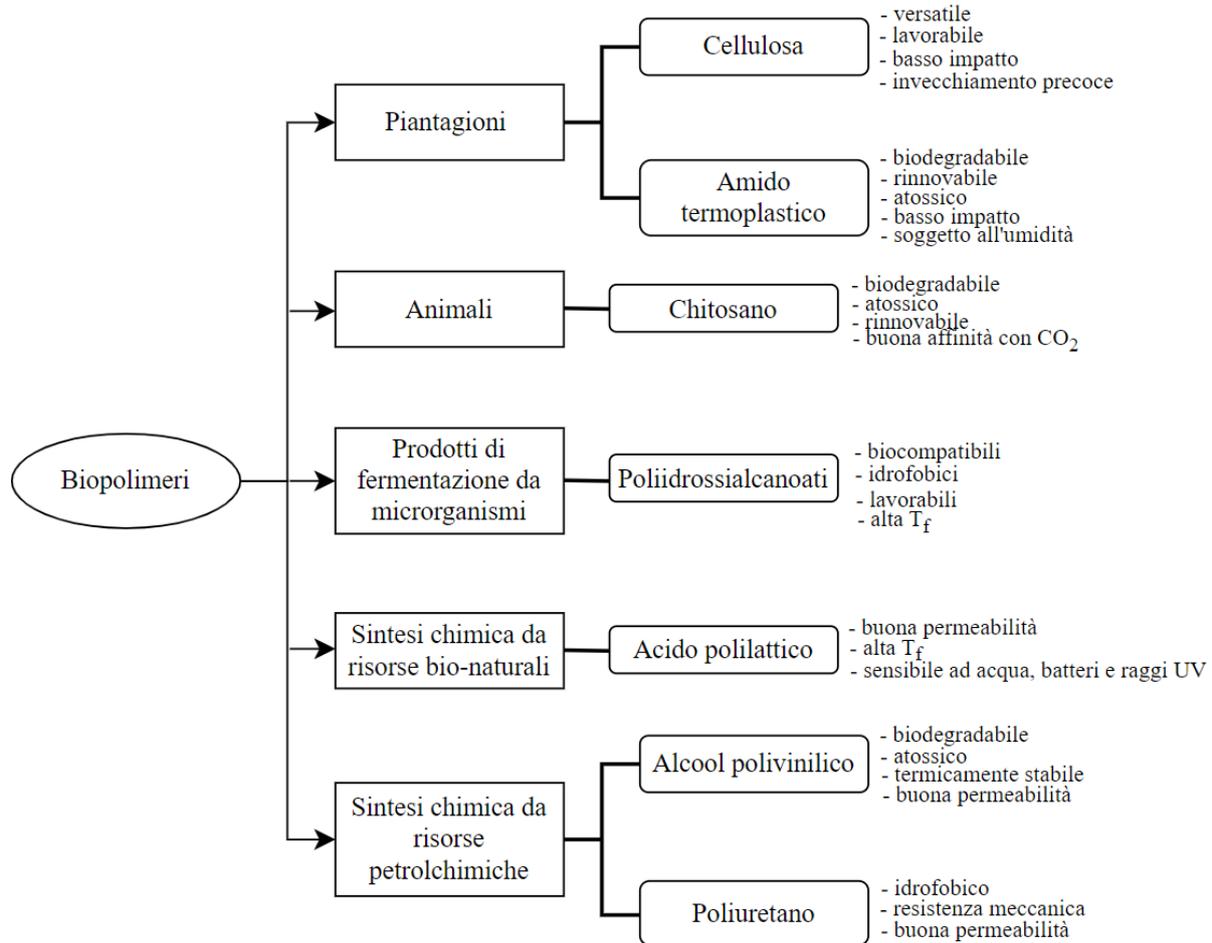


Figura 2.8 Classificazione dei biopolimeri per la separazione della CO₂ sulla base della loro origine

Il primo tra questi materiali ad essere esplorato è stata la cellulosa (CA). Dei biopolimeri possono essere ottenuti dall'esterificazione della cellulosa in polvere, proveniente da piante come la canna da zucchero, il cotone e il legno. Si tratta di biopolimeri vetrosi particolarmente soggetti al fenomeno della plastificazione. Godono di un basso impatto ambientale, versatilità e una buona lavorabilità. Spesso i biopolimeri derivanti da cellulosa vengono utilizzati nelle MMMs insieme a materiali inorganici come zeoliti, silice e nanotecnologie al carbonio (CNF), oppure sono additivati con materiali polimerici come PEG e SBR per migliorarne la flessibilità o la stabilità all'invecchiamento. Un altro biopolimero ottenibile dalle piante, e in particolare da quelle amidacee (come riso, mais, grano, patate), è l'amido termoplastico (TPS). Racchiude i vantaggi di rinnovabilità, biodegradabilità, atossicità e basso impatto ambientale grazie alla bassa energia richiesta per la loro produzione. Le membrane risultano trasparenti e omogenee e la loro bassa tollerabilità dell'umidità può essere migliorata con l'aggiunta di altri polimeri (come il chitosano) o l'incorporazione di nanoparticelle. Un biopolimero molto promettente per

la separazione di CO₂ è il chitosano (CS), un derivato della deacetilazione della chitina, un polisaccaride estratto dal guscio dei crostacei. Il chitosano offre numerosi vantaggi, tra cui quelli di interesse ambientale come la rinnovabilità, la biodegradabilità e l'atossicità, ma anche un buon assorbimento e permeazione della CO₂ grazie ai numerosi gruppi amminici e idrossilici che gli appartengono. Un'innovativa combinazione del carbossimetilchitosano con la poli(amidoammina) PAMAM ha anche permesso di ottenere un biopolimero con una buona stabilità termica. Uno dei biopolimeri più interessanti per la preparazione di membrane è il polimero dell'acido lattico (PLA), che può essere ottenuto o per policondensazione diretta dell'acido idrossilico oppure per polimerizzazione con apertura dell'anello del lattide. Il PLA ha una temperatura di transizione vetrosa tra i 50 e i 65°C, a seconda del grado di cristallinità, e una temperatura di fusione abbastanza alta, tra i 170 e i 180°C. Il problema principale è che in presenza di acqua, batteri e raggi UV il PLA è soggetto a degradazione per idrolisi, per cui sono state sviluppate diverse strategie di miscelazione per migliorarne la sua stabilità ottica e meccanica, insieme ad aumentarne la permeabilità. Un'altra classe di biopolimeri è quella derivante dalla fermentazione microbica di risorse rinnovabili di carbonio come glucosio, maltosio, sciroppo di mais, glicerolo di scarto. Si tratta di omo e co-poliesteri lineari caratterizzati da proprietà interessanti come la biocompatibilità, l'idrofobicità, l'elevata lavorabilità e la purezza ottica. Hanno in genere una temperatura di transizione vetrosa tra i 2 e i 5°C e proprietà paragonabili a quelle dei polimeri tradizionali come PE, PS e PP. L'alcool polivinilico (PVA) ha dimostrato di essere un buon materiale per la preparazione di membrane selettive per la CO₂. Si tratta di un polimero sintetico biodegradabile, atossico, con una buona stabilità termica e poco flessibile a causa dei gruppi idrossilici che generano legami a idrogeno. La sua intrinseca idrofilia lo rende altamente permeabile e selettivo per la CO₂, e anche compatibile con le ammine, che possono migliorarne le prestazioni. Un materiale che sta emergendo per la separazione della CO₂ è il poliuretano (PU), che ha permeabilità e selettività paragonabili a quelle delle membrane attualmente in commercio, come la PEBAX®. Caratteristiche importanti sono l'idrofobicità, la buona resistenza meccanica, la bassa rigidità. Può derivare da fonti rinnovabili come gli oli vegetali, principalmente olio di ricino e olio di soia. L'introduzione di nanoparticelle può aumentare il volume libero della membrana e migliorarne le proprietà di permeazione e la stabilità meccanica e termica. Per avere un'idea delle prestazioni dei biopolimeri rispetto ai polimeri descritti nelle sezioni precedenti, in particolare per la separazione della CO₂, alcuni di essi sono riportati in TABELLA 2.11 con le corrispondenti permeabilità e selettività riscontrate in alcuni test.

Tabella 2.11 Performance di alcuni biopolimeri per la separazione di CO₂; P_{CO_2} : Permeanza= $P_{CO_2}l$ [15]

Biopolimero	Miscela	P_{CO_2} [Barrer]	P_{CO_2} [GPU]	α	Condizioni
ZIF8@CNF	CO ₂ -CH ₄	550	-	36.2	$p=30$ [bar]; $T=25^\circ\text{C}$.
CA	CO ₂ -N ₂	13.8	-	39.5	$p=1$ [bar]; $T=25^\circ\text{C}$.
PES/AS	CO ₂ -CH ₄	-	7.57	58	$p=27$ [bar]; $T=30^\circ\text{C}$.
Chitosan/PVA	CO ₂ -N ₂	-	3000	19	45% di acqua; PVA: 5 [mg/ml].
PAA/CS	CO ₂ -N ₂	-	39	260	$p=2$ [bar]; $T=90^\circ\text{C}$.
PLA Easy Fil™	CO ₂ -CH ₄	65	-	250	$T_g=60^\circ\text{C}$.
PHB	CO ₂ -N ₂	9.1	-	2.2-3.4	$T=277-296$ [K].
Miscela PVAm/PVA	CO ₂ -N ₂	-	215	174	Stabilità in 400 ore di test.
PVA/CNC	CO ₂ -CH ₄	-	107	43	1% CNC; $pH=6-10$.
PU/NiO	CO ₂ -N ₂	337-348	-	46-38	NiO=0-10%; $p=1$ [bar]; $T=30^\circ\text{C}$.
PU	CO ₂ -CH ₄	332	-	13	$T=30^\circ\text{C}$.

3. Modellazione del modulo a membrana

La parte centrale di un impianto a membrana è il modulo, cioè la più piccola unità (si parla di unità di separazione) nella quale viene impacchettata l'area della membrana. Il gas da trattare entra nel modulo con una certa portata, composizione, temperatura e pressione, e si separa in due flussi, che prendono il nome di permeato (il gas che è riuscito ad attraversare la membrana) e retentato. In questo lavoro si considereranno solo miscele gassose binarie e membrane affini alla CO₂. La CO₂, componente più permeabile rispetto all'azoto e al metano, si concentrerà nel permeato. In generale, per raggiungere gli obiettivi di separazione desiderati su larga scala, è necessaria un'area della membrana molto grande. Esistono diverse possibilità per posizionarla e organizzarla in un modulo, ma si basano tutte su due tipi di configurazioni della membrana: piatta o tubolare. Tra i moduli a membrana piatta troviamo i *plate-and-frame* e le *spiral-wound*. Si hanno poi i moduli tubolari, capillari e a fibra cava, che si differenziano principalmente per le dimensioni dei tubi impiegati. La scelta del tipo di configurazione dipende più che altro da considerazioni di tipo economico, tenendo conto di alcuni parametri come la possibilità di pulizia e di manutenzione, la compattezza, il tipo di separazione e la possibilità di sostituire la membrana.

Un certo numero di moduli che vengono collegati tra loro, in serie o in parallelo, prende il nome di stadio. Anche la disposizione dei moduli all'interno di uno stadio è una scelta ingegneristica. Il design più semplice per il sistema è il *dead-end*, in cui tutta l'alimentazione viene forzata a passare attraverso la membrana, aumentando la concentrazione lato *feed* dei componenti rigettati, con un peggioramento delle prestazioni e quindi della qualità del permeato (FIGURA 3.1A). In realtà, per la separazione di gas si opta sempre per un design del tipo *cross-flow*, in cui l'alimentazione fluisce parallelamente alla superficie della membrana e si divide in due correnti, permettendo alle molecole non permeate di lasciare il modulo (FIGURA 3.1B). Le condizioni di flusso che possono verificarsi in questo tipo di operazione verranno descritte in questo capitolo.

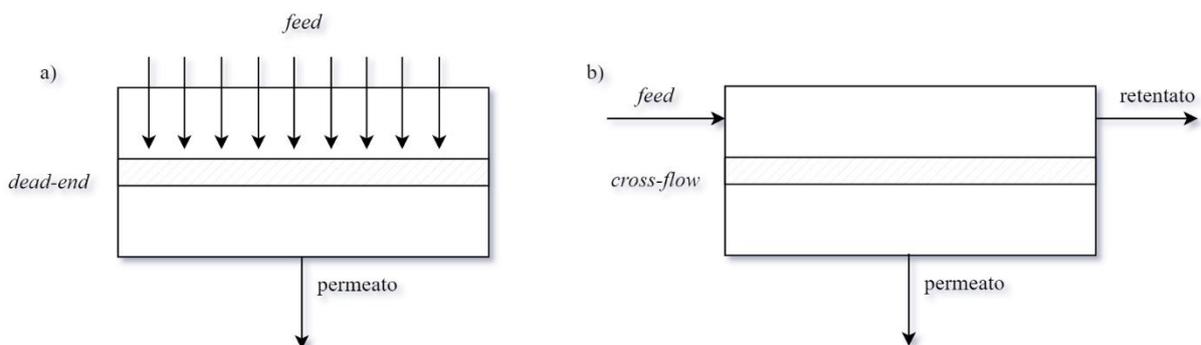


Figura 3.1 Rappresentazione schematica di due modalità operative di base dei moduli a membrana: *dead-end* (a) e *cross-flow* (b)

Nel presente capitolo verranno descritte le principali configurazioni per un modulo a membrana. Sarà in seguito fornita la descrizione dei possibili modelli di flusso all'interno del modulo o del singolo stadio e di come questi sono stati implementati in MATLAB®. Infine si tenterà di convalidare i suddetti modelli con i risultati sperimentali riscontrati in letteratura.

3.1 Design del modulo a membrana

La scelta del design migliore per il modulo a membrana si basa su considerazioni di tipo economico, ma non solo. Molto è determinato dal tipo di separazione che si intende effettuare. Ogni configurazione ha delle caratteristiche che devono essere prese in considerazione. I moduli tubolari, ad esempio, sono i più costosi ma particolarmente adatti per le applicazioni con alta tendenza al *fouling* (accumulo di materia sulla superficie della membrana) grazie alla loro controllabilità e facilità di pulizia. Vale l'opposto per i moduli a fibra cava, in cui è essenziale un pretrattamento del flusso di alimentazione. In FIGURA 3.2 si confrontano le caratteristiche dei diversi moduli [11].

	tubolare	plate-and-frame	spiral-wound	capillare	fibra cava
densità di impaccamento	→				
investimenti	←				
tendenza al fouling	→				
facilità di pulizia	←				
sostituzione della membrana	√/X	√	X	X	X

Figura 3.2 Confronto qualitativo delle diverse configurazioni dei moduli a membrana

3.1.1 Modulo *plate-and-frame*

La configurazione *plate-and-frame* è quella che più si avvicina alla membrana piatta da laboratorio. Le membrane sono organizzate in set da due, in modo che i lati di alimentazione siano adiacenti, separati da un adeguato distanziatore (*spacer*) che fornisce resistenza al modulo (FIGURA 3.3A). Per un sistema *cross-flow* il flusso viene alimentato in modo tale che scorra tangenzialmente alla membrana, come mostrato in FIGURA 3.3B. Il permeato attraversa la membrana e si raccoglie all'interno della piastra di supporto, mentre il retentato si porta all'estremità finale della piastra. Sono stati introdotti dei dischi di arresto per permettere una distribuzione uniforme del flusso e ridurre la canalizzazione (la tendenza del flusso a tracciare un percorso fisso).

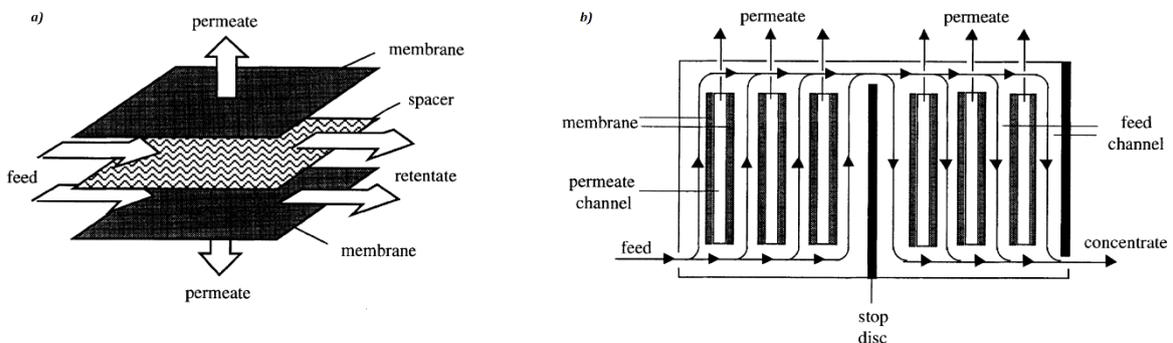


Figura 3.3 Rappresentazione schematica di un modulo *plate-and-frame* (a) e del suo flusso (b) [11]

Il numero di set è definito dall'area della membrana necessaria ad effettuare la separazione desiderata. La densità di impaccamento, definita come il rapporto tra la superficie della membrana e il volume del modulo, è generalmente intorno a 100-400 [m²/m³], molto bassa rispetto ad altri design. Il modulo *plate-and-frame* è vantaggioso soprattutto per la sua facilità di fabbricazione e quindi economicità.

3.1.2 Modulo *spiral-wound*

Un'implementazione del modulo *plate-and-frame* è il modulo *spiral-wound*, in cui un sistema di piastre e telai viene avvolto attorno a un tubo perforato centrale per la raccolta del permeato (FIGURA 3.4). Questa configurazione permette di mantenere la facilità di fabbricazione tipica delle membrane piatte, ma allo stesso tempo aumentare la densità di impaccamento e ridurre le perdite di carico. Il modulo è composto quindi da quattro fogli: il distanziatore per il *feed*, una membrana, un distanziatore per il permeato e una seconda membrana. Prima di essere arrotolati, i fogli hanno dimensioni di circa 1-1.5 [m] per 2-2.5 [m]. Il modulo finale ha un diametro tra i 100 e i 300 [mm] e una lunghezza in direzione assiale di 1-1.5 [m]. Per quanto riguarda la densità di impaccamento siamo tra i 300 e i 1000 [m²/m³] a seconda dell'altezza del cilindro, e quindi dello spessore dei distanziatori per il *feed* e il permeato. Il distanziatore lato *feed*, che si trova tra la superficie superiore di due membrane, ha uno spessore di circa 1 [mm] e ha l'ulteriore funzione di promuovere la turbolenza, influenzando sulla caduta di pressione e il trasferimento di massa. Un'immagine più chiara della sua disposizione e struttura è fornita in FIGURA 3.5A. Il distanziatore poroso posto dal lato di permeato tra due membrane ha invece uno spessore di circa 0.2 [mm] ed è incollato ad esse lungo i tre lati in modo da formare un involucro. In figura FIGURA 3.5B si può facilmente comprendere come fluiscono le correnti all'interno del modulo. L'alimentazione fluisce parallelamente al tubo centrale, percorrendo il modulo nella sua direzione assiale, mentre il permeato scorre radialmente, raccogliendosi nel tubo centrale. Spesso si utilizza un certo numero di separatori a spirale posti in serie all'interno di un recipiente in pressione, e connessi tra loro attraverso il tubo centrale di raccolta.

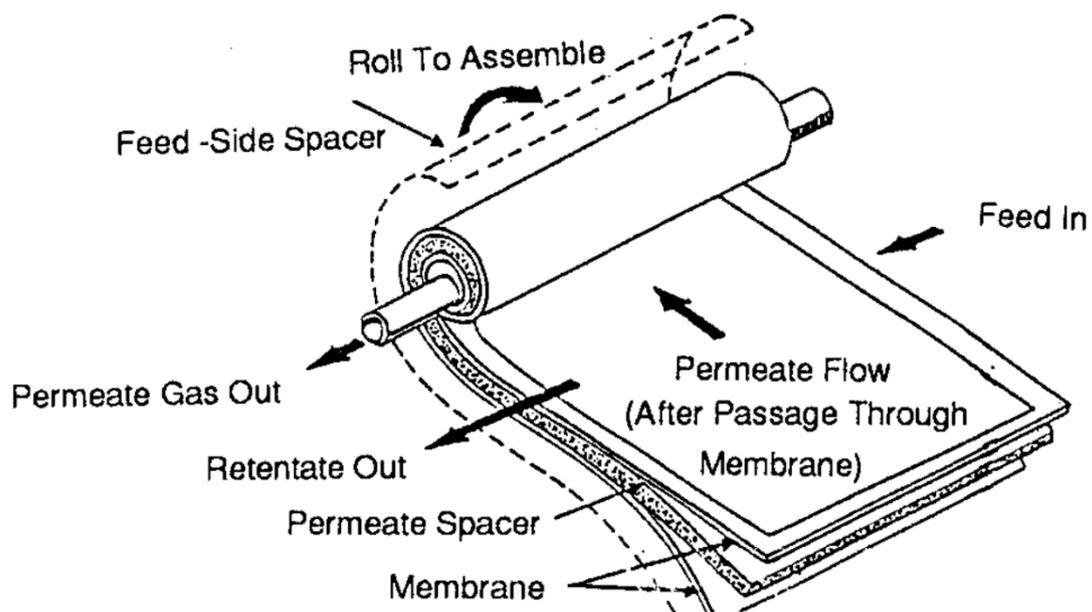


Figura 3.4 Rappresentazione schematica del modulo *spiral-wound* [16]

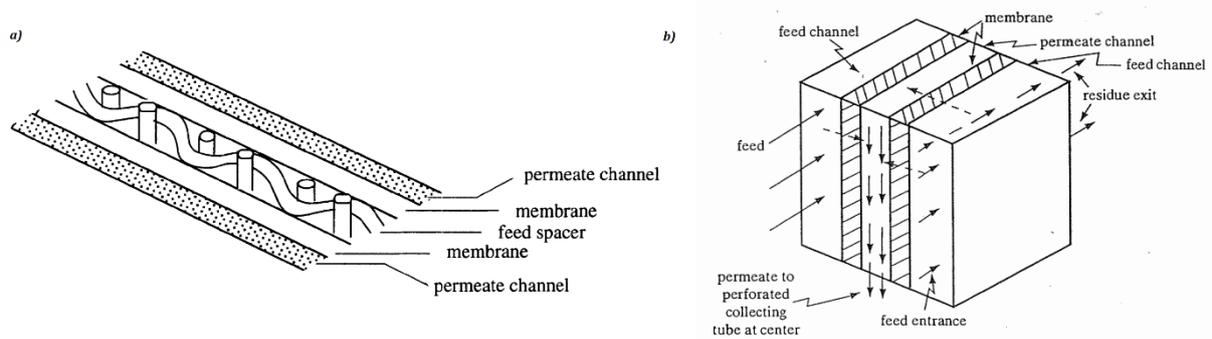


Figura 3.5 Sezione trasversale di un modulo *spiral-wound* (a) [11] e rappresentazione del percorso del flusso locale (b) [17]

3.1.3 Modulo tubolare

Le membrane in un modulo tubolare sono supportate da tubi porosi di diametro in genere superiore ai 10 [mm]. Il materiale dei tubi può essere in acciaio inossidabile, plastica o ceramica e il loro numero all'interno del modulo può variare da 4 a 18, o anche più. La densità di impaccamento di questo tipo di modulo non è altissima, anche meno di 300 [m²/m³], ma ha il vantaggio di essere economico e di permettere una facile pulizia. L'alimentazione passa sempre dal centro dei tubi e ne esce il retentato dall'altra estremità del modulo, mentre il permeato attraversa le pareti porose dei tubi raccogliendosi nell'*housing* del modulo. Una rappresentazione schematica è fornita in FIGURA 3.6. Per questo tipo di modulo vengono spesso utilizzate membrane ceramiche, come nel caso del monolite ceramico, un particolare tipo di modulo tubolare. In questa struttura il monolite portante è in materiale ceramico poroso, e i *toplayer* che ricoprono la parete interna dei tubi vengono scelti in γ -allumina o zirconia.

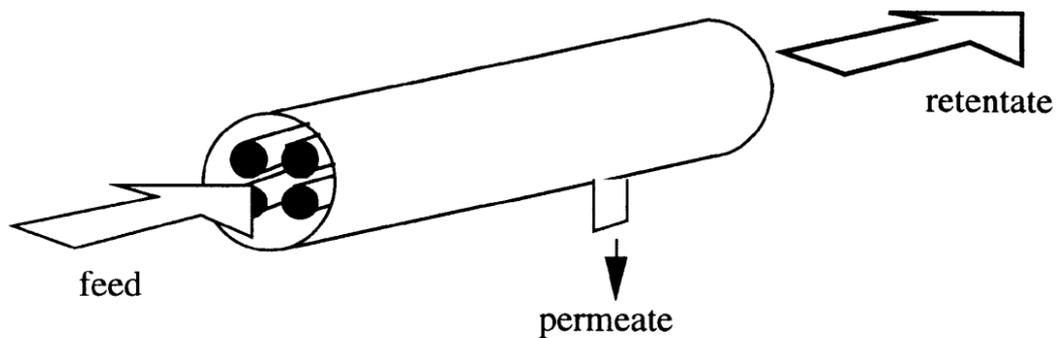


Figura 3.6 Rappresentazione schematica del modulo tubolare [11]

3.1.4 Modulo capillare

Il modulo capillare si differenzia dal modulo tubolare principalmente per le dimensioni della membrana tubolare, che avrà un diametro tra 0.5 e 10 [mm]. Il modulo capillare è autoportante e le estremità libere delle fibre sono rivestite con gomma siliconica, poliuretano o resine epossidiche. La densità di impaccamento è di 600-1200 [m²/m³], quindi una via di mezzo tra i moduli tubolari e quelli a fibra cava. Si può scegliere tra due configurazioni di flusso, come illustrato in FIGURA 3.7. La configurazione *inside-out* prevede che il *feed* venga alimentato

all'interno dei capillari e che il permeato si raccolga nel mantello. In alternativa si può scegliere di far passare l'alimentazione all'interno del mantello, facendo permeare la corrente di permeato all'interno dei capillari (*outside-in*). A seconda di quale delle due modalità viene scelta, si fabbricheranno delle membrane asimmetriche con il *layer* denso rivolto verso l'interno o l'esterno del capillare. Quale tra le due vada preferita dipende da fattori come il tipo di membrana disponibile, la pressione operativa e la caduta di pressione.

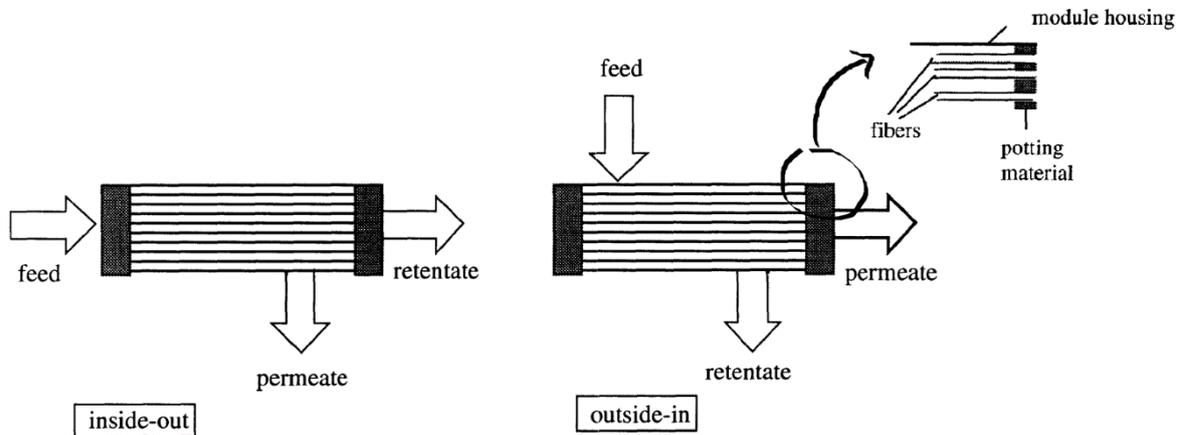


Figura 3.7 Rappresentazione schematica del modulo capillare, a sinistra con l'alimentazione lato tubo (*inside-out*), a destra con l'alimentazione lato mantello (*outside-in*) [11]

3.1.5 Modulo a fibra cava

Nel modulo a fibra cava le membrane sono organizzate in tubicini (fibre cave) di dimensioni molto piccole, con un diametro interno tra 100 e 500 $[\mu\text{m}]$ e un diametro esterno tra 200 e 1000 $[\mu\text{m}]$. Il modulo ha le sembianze di uno scambiatore di calore a fascio tubiero, con migliaia di fibre, di lunghezza tra i 3 e i 5 $[\text{m}]$, legate insieme ad ogni estremità e circondate da un *housing* (mantello) di metallo di 0.1-0.2 $[\text{m}]$ di diametro (FIGURA 3.8A). È il modulo con la più alta densità di impaccamento, può anche raggiungere valori di 30000 $[\text{m}^2/\text{m}^3]$. Può essere utilizzato per applicazioni in cui il flusso di alimentazione è relativamente pulito, per evitare fenomeni quali incrostazione e canalizzazione. Come per il modulo capillare, si può scegliere tra due modalità di flusso. Per la separazione di gas è preferibile la scelta del sistema *outside-in* in modo da evitare elevate cadute di pressione e da avere a disposizione un'area della membrana maggiore. Il flusso di alimentazione, quindi, entra da un lato del mantello e fuoriesce come retentato dall'estremità opposta. Il gas in grado di permeare la membrana entra all'interno delle fibre e fluisce in contro-corrente rispetto al flusso nel mantello, fino a raccogliersi in una camera alla fine del modulo in cui terminano le estremità aperte delle fibre, per poi fuoriuscire. Quanto descritto è illustrato in FIGURA 3.8B.

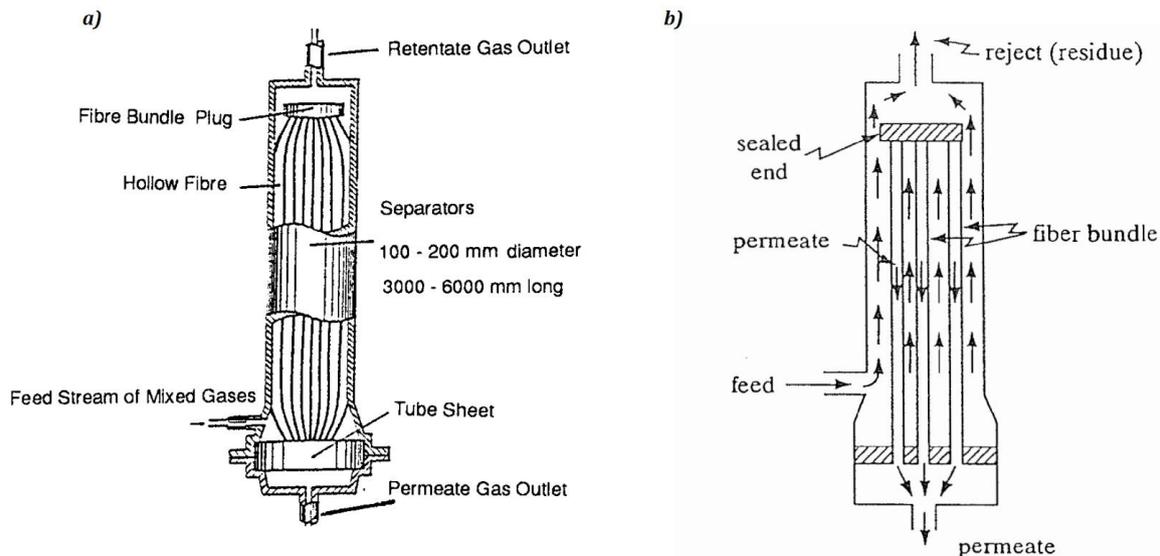


Figura 3.8 Rappresentazione di un modulo a fibra cava (a) [16] e rappresentazione schematica delle correnti all'interno del modulo (b) [17]

3.2 Modelli di flusso

Si è detto che in un modulo o più in generale in uno stadio a membrana l'alimentazione, con una certa portata e composizione iniziale, viene fatta fluire tangenzialmente alla superficie della membrana stessa. Poiché il retentato uscente dallo stadio avrà caratteristiche differenti da quelle alimentate, è implicito che portata e composizione variano all'interno del modulo e in particolare sono funzioni della lunghezza del modulo. Per derivare un modello teorico che descriva come variano questi parametri e quali sono le prestazioni del modulo (o stadio), scelta la membrana e le condizioni operative, è necessario innanzitutto individuare il tipo di flusso che si verifica ai due lati della membrana. Si considerano principalmente due condizioni: perfetta miscelazione, quindi la concentrazione del gas (lato *feed* o lato permeato) è uniforme e pari alle condizioni di uscita, oppure *plug-flow*, avendo un gradiente di concentrazione lungo la direzione parallela alla superficie della membrana. In FIGURA 3.9 sono rappresentati schematicamente quattro modelli di flusso ideali, per ognuno dei quali andrà derivato un differente modello teorico. Il modello più semplice è quello di completa miscelazione, in cui sia la camera di alimentazione che la camera del permeato sono considerati in condizioni di perfetta miscelazione. Il modello *cross-flow* ideale prevede che il flusso alimentato sia in *plug-flow*, e che il permeato fluisca perpendicolarmente alla superficie della membrana senza miscelarsi. Se invece si considera sia la corrente dal lato *feed* che la corrente di permeato in *plug-flow*, si parla di modello in equicorrente o in controcorrente, a seconda del verso relativo dei due flussi. Nella derivazione dei modelli teorici si considereranno alcune semplificazioni:

- Condizioni isotermitiche;
- Stato stazionario;
- Cadute di pressione trascurabili, quindi la pressione a ogni lato della membrana si mantiene costante;
- Permeabilità e selettività costanti, indipendenti da pressione e concentrazione.

Per la progettazione di un sistema di separazione con membrana vanno innanzitutto definiti alcuni parametri di processo. Nel caso di miscele binarie come quelle che verranno considerate, ci si può riferire ai due componenti come componente A e componente B. Generalmente, con

A ci si riferisce al componente più permeante, quindi in questo caso la CO_2 . Ogni corrente viene caratterizzata da un valore di portata Q e dalla frazione molare di un componente (di solito ci si riferisce al componente A). Con x ci si riferisce alla frazione molare nel lato della membrana ad alta pressione, mentre si utilizza y per indicare la frazione molare dal lato del permeato. Con i pedici f , r e p si fa riferimento rispettivamente alla corrente di alimentazione, al retentato e al permeato. Un parametro di processo spesso considerato è il recupero del componente A nel permeato, definito come il flusso di A che riesce ad attraversare la membrana, rispetto a quello alimentato allo stadio (3.1).

$$R_p = \frac{y_{p,A} Q_p}{x_{f,A} Q_f} \quad (3.1)$$

In caso di separazione di miscele $\text{CO}_2\text{-CH}_4$, in cui il principale prodotto desiderato è il metano, concentrato nel retentato, risulta più comodo e utile riferirsi al recupero di B nel retentato, come definito nell'EQUAZIONE 3.2.

$$R_r = \frac{x_{r,B} Q_r}{x_{f,B} Q_f} \quad (3.2)$$

Insieme al recupero, un altro importante parametro di processo è la purezza del permeato o del retentato, che coincide con la frazione molare di A nel permeato nel primo caso, e con la frazione molare di B nel retentato nel secondo caso. Come si potrà osservare in seguito, esiste un *trade-off* tra recupero e purezza del prodotto di interesse, e non potranno essere soddisfatti entrambi i parametri usando un unico stadio.

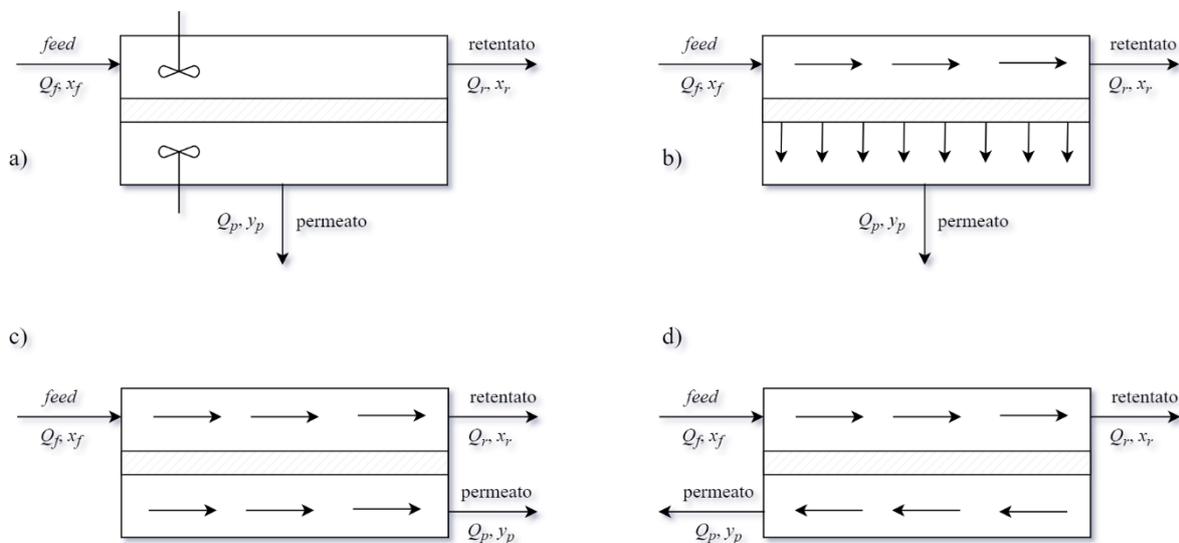


Figura 3.9 Rappresentazione schematica dei modelli ideali di flusso nella separazione di gas con membrana: a) completa miscelazione; b) *cross-flow*; c) equicorrente; d) controcorrente

3.2.1 Modello di completa miscelazione

Il modello di completa miscelazione fornisce una buona previsione delle caratteristiche del permeato quando il recupero del sistema è molto basso, e cioè la velocità della corrente di permeato è solo una piccola frazione di quella in ingresso, per cui la variazione di concentrazione ai due lati della membrana è minima. La FIGURA 3.10 fornisce una rappresentazione del modello. Si distingueranno adesso due casi. Il primo vede il permeato come il prodotto di interesse. Nel secondo caso si è interessati al componente meno permeabile, concentrato nel retentato.

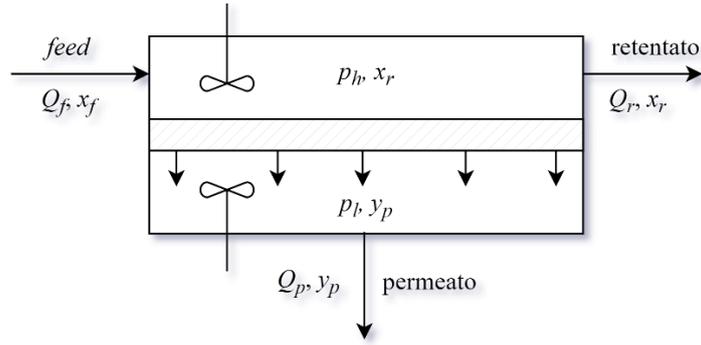


Figura 3.10 Rappresentazione delle correnti nel modello di completa miscelazione

Caso 1: Vengono fornite le pressioni ai due lati della membrana p_h e p_l , le permeabilità dei due componenti (P_A e P_B), le condizioni di alimentazione (Q_f e x_f), lo spessore della membrana l e il recupero di A nel permeato R_p . Le frazioni molari indicate si riferiscono al componente A. Si può scrivere il bilancio complessivo di materia (3.3) e quello riferito al componente A (3.4).

$$Q_f = Q_r + Q_p \quad (3.3)$$

$$Q_f x_f = Q_r x_r + Q_p y_p \quad (3.4)$$

Considerando la definizione del recupero (3.1) e il bilancio di materia parziale si può esprimere anche la portata di A nel retentato come funzione del recupero (3.5), ottenendo le due equazioni 3.6 e 3.7 per le portate di permeato e retentato.

$$Q_r x_r = Q_f x_f (1 - R_p) \quad (3.5)$$

$$Q_p = \frac{R_p Q_f x_f}{y_p} \quad (3.6)$$

$$Q_r = \frac{Q_f x_f (1 - R_p)}{x_r} \quad (3.7)$$

Sostituendo la (3.6) e la (3.7) nella (3.3), si ottiene l'espressione (3.8) e quindi una relazione tra la composizione del retentato x_r e la composizione del permeato y_p (3.9).

$$Q_f = \frac{Q_f x_f (1 - R_p)}{x_r} + \frac{Q_f x_f R_p}{y_p} \quad (3.8)$$

$$x_r = y_p \frac{x_f - R_p x_f}{y_p - R_p x_f} \quad (3.9)$$

Considerando che il trasporto all'interno della membrana avviene per il meccanismo di soluzione-diffusione e che il calcolo viene effettuato allo stato stazionario, la velocità di permeazione del componente A attraverso la membrana può essere descritta con l'equazione 3.10. Allo stesso modo può essere espressa la velocità di permeazione del componente B (3.11).

$$J_A = \frac{Q_A}{A_m} = \frac{Q_p y_p}{A_m} = \frac{P_A}{l} (p_h x_r - p_l y_p) \quad (3.10)$$

$$J_B = \frac{Q_B}{A_m} = \frac{Q_p (1 - y_p)}{A_m} = \frac{P_B}{l} (p_h (1 - x_r) - p_l (1 - y_p)) \quad (3.11)$$

La velocità di permeazione è uguale in ogni punto della membrana, essendo costanti le pressioni parziali dei due componenti ai due lati della membrana. Definendo con β il rapporto tra bassa e alta pressione (3.12) e dividendo alla (3.10) la (3.11) si ottiene l'equazione 3.13. Mettendo a sistema la (3.13) con la (3.9) si ottiene un sistema di due equazioni in due incognite, e cioè le

composizioni di permeato e retentato, che permette di calcolare la frazione molare di A nel permeato.

$$\beta = \frac{p_l}{p_h} \quad (3.12)$$

$$\left\{ \begin{array}{l} \frac{y_p}{1-y_p} = \frac{P_A \cdot (x_r - \beta y_p)}{P_B \cdot [(1-x_r) - \beta(1-y_p)]} \\ x_r = y_p \frac{x_f - R_p x_f}{y_p - R_p x_f} \end{array} \right. \quad (3.13)$$

$$(3.9)$$

Svolgendo il sistema, si ottiene un'equazione di secondo grado la cui soluzione è riportata di seguito:

$$y_p = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (3.14)$$

Con

$$a = \beta(P_B - P_A) \quad (3.15)$$

$$b = P_B[1 - \beta + x_f(R_p - 1 - \beta R_p)] + P_A[\beta + x_f(1 - R_p + \beta R_p)] \quad (3.16)$$

$$c = P_B R_p x_f (\beta - 1) - P_A x_f (1 - R_p + \beta R_p) \quad (3.17)$$

Trovato y_p , è immediato il calcolo di x_r , Q_r e Q_p dalle (3.9), (3.7) e (3.6), rispettivamente. Si può infine determinare l'area della membrana che permette di effettuare tale separazione con l'equazione (3.18), ottenuta a partire dalla (3.10).

$$A_m = \frac{Q_p y_p}{(P_A/l)(p_h x_r - p_l y_p)} \quad (3.18)$$

Caso 2: Se piuttosto che il recupero di A nel permeato si vuole fornire al problema il recupero di B nel retentato, si è scelto di riferire le frazioni molari al componente B e di risolvere come quanto spiegato sopra. Le equazioni di bilancio totale (3.3) e parziale (3.4) rimangono invariate. La definizione del retentato è la (3.2), da cui derivano le nuove espressioni per la portata di permeato (3.19) e di retentato (3.20).

$$Q_p = \frac{(1-R_r)Q_f x_f}{y_p} \quad (3.19)$$

$$Q_r = \frac{R_r Q_f x_f}{x_r} \quad (3.20)$$

Si può quindi scrivere la nuova relazione tra x_r e y_p (3.21) e risolvere il nuovo sistema:

$$\left\{ \begin{array}{l} x_r = \frac{y_p R_r x_f}{y_p - (1-R_r)x_f} \\ \frac{y_p}{1-y_p} = \frac{P_B \cdot (x_r - \beta y_p)}{P_A \cdot [(1-x_r) - \beta(1-y_p)]} \end{array} \right. \quad (3.21)$$

$$(3.22)$$

La soluzione del sistema è la (3.14) con:

$$a = \beta(P_A - P_B) \quad (3.23)$$

$$b = P_A[1 - \beta + x_f(\beta R_r - R_r - \beta)] + P_B[\beta + x_f(R_r + \beta - \beta R_r)] \quad (3.24)$$

$$c = P_A x_f (R_r + \beta - \beta R_r - 1) - P_B x_f (R_r + \beta - \beta R_r) \quad (3.25)$$

x_r , Q_r e Q_p si calcoleranno con le equazioni (3.21), (3.20) e (3.19), rispettivamente, mentre per il calcolo dell'area della membrana si utilizzerà l'espressione (3.26).

$$A_m = \frac{Q_p y_p}{(P_B/l)(p_h x_r - p_l y_p)} \quad (3.26)$$

Le funzioni implementate in MATLAB® per risolvere il modulo a membrana in condizioni di completa miscelazione, per ognuno dei casi, sono riportate nell'APPENDICE A.

3.2.2 Modello *cross-flow*

Il modello *cross-flow* fornisce una buona descrizione del sistema quando il flusso alimentato ha una velocità longitudinale tale da poter essere considerato in *plug-flow* nella camera di alimentazione, fluendo parallelamente alla superficie della membrana, e la camera del permeato è portata quasi in sottovuoto, tanto da poter considerare il flusso di permeato perpendicolare alla membrana. La soluzione del modello *cross-flow* è stata derivata da Weller e Steiner [18]. Si considera che in nessuna delle due camere si verifica il *mixing*, e quindi la composizione del permeato varierà in ogni punto della membrana, a seconda delle velocità di permeazione relative dei due componenti in quella posizione. Il modello *cross-flow* può essere una buona approssimazione del flusso in un modulo *spiral-wound* che utilizza una membrana asimmetrica sottoposta a un flusso elevato. Una rappresentazione di questo modello di flusso è fornita in FIGURA 3.11. Anche qui verranno descritti due casi, a seconda del prodotto a cui si è interessati.

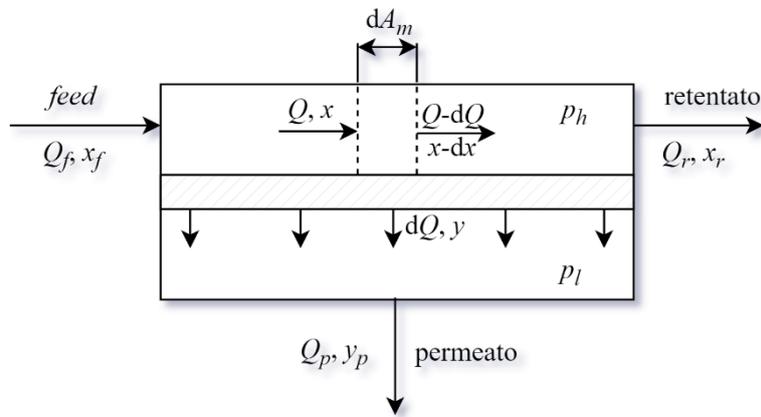


Figura 3.11 Rappresentazione del modello *cross-flow*

Caso 1: Vengono fornite le pressioni ai due lati della membrana p_h e p_l , le permeabilità dei due componenti (P_A e P_B), le condizioni di alimentazione (Q_f e x_f), lo spessore della membrana l e il recupero di A nel permeato R_p . le frazioni molari indicate saranno riferite al componente A. Si consideri un'area differenziale della membrana dA_m , la velocità di permeazione in ogni punto della membrana è data dalla (3.27) per il componente A e dalla (3.28) per il componente B, per il meccanismo di soluzione-diffusione.

$$-dQ_A = -y dQ = \frac{P_A}{l} (p_h x - p_l y) dA_m \quad (3.27)$$

$$-dQ_B = -(1 - y) dQ = \frac{P_B}{l} [p_h (1 - x) - p_l (1 - y)] dA_m \quad (3.28)$$

Scrivendo le permeabilità in termini di selettività ideale e facendo il cambio di variabile espresso dalla (3.29) e la (3.30), il rapporto tra la (3.27) e la (3.28) si può scrivere come l'equazione 3.31.

$$j = \frac{dQ_A}{dQ_B} = \frac{y}{1-y} \quad (3.29)$$

$$i = \frac{Q_A}{Q_B} = \frac{x}{1-x} \quad (3.30)$$

$$j = \alpha \frac{\left(p_{h_{i+1}}^i - p_{l_{j+1}}^j \right)}{\left(p_{h_{i+1}}^1 - p_{l_{j+1}}^1 \right)} \quad (3.31)$$

Risolvendo questa quadratica e sostituendo al rapporto di pressioni β (3.12) si ottiene l'espressione 3.32.

$$j = f(i) = Di - F + [D^2i^2 + 2Ei + F^2]^{1/2} \quad (3.32)$$

Con

$$D = \frac{1}{2}[(1 - \alpha)\beta + \alpha] \quad (3.33)$$

$$F = -\frac{1}{2}[(1 - \alpha)\beta - 1] \quad (3.34)$$

$$E = \frac{\alpha}{2} - DF \quad (3.35)$$

Notando che

$$j = \frac{dQ_A}{dQ_B} = f(i) = f\left(\frac{Q_A}{Q_B}\right) \quad (3.36)$$

si può scegliere di utilizzare come variabili Q_B ed i e scrivere dQ_B/Q_B in funzione di i . Questo è facilmente attuabile ricordando che

$$di = d\left(\frac{Q_A}{Q_B}\right) = \frac{Q_B dQ_A - Q_A dQ_B}{Q_B^2} \quad (3.37)$$

e potendo quindi scrivere la relazione 3.38.

$$\frac{dQ_B}{Q_B} = \frac{di}{f(i)-i} = \frac{di}{(D-1)i - F + [D^2i^2 + 2Ei + F^2]^{0.5}} \quad (3.38)$$

Il membro a destra della (3.38) può essere razionalizzato utilizzando un cambio di variabile, come definito nella (3.39):

$$i = \frac{0.5(F^2 - u^2)}{Du - E} \quad (3.39)$$

e ottenere dall'integrazione della (3.38):

$$\ln \frac{Q_B}{Q_{f,B}} = - \int_{u_f}^u \frac{Du^2 - 2Eu + DF^2}{(Du - E)(u - \alpha + F)(u - F)} dt \quad (3.40)$$

o ancora

$$\ln \frac{Q_B}{Q_{f,B}} = G \cdot \ln \frac{u_f - E/D}{u - E/D} + S \cdot \ln \frac{u_f - \alpha + F}{u - \alpha + F} + T \cdot \ln \frac{u_f - F}{u - F} \quad (3.41)$$

con

$$G = \frac{1}{2D-1} \quad (3.42)$$

$$S = \frac{\alpha(D-1)+F}{(2D-1)(\alpha/2-F)} \quad (3.43)$$

$$T = \frac{1}{1-D-E/F} \quad (3.44)$$

$$u = -Di + (D^2i^2 + 2Ei + F^2)^{0.5} \quad (3.45)$$

e u_f calcolato alle condizioni iniziali di alimentazione:

$$u_f = u(i_f) = u(x_f) \quad (3.46)$$

Si può definire un recupero parziale R_p^* in riferimento alla portata di A che permea in corrispondenza di x , come descritto dall'espressione (3.47).

$$R_p^* = 1 - \frac{Qx}{Q_f x_f} \quad (3.47)$$

Quando $x = x_r$ allora $R_p^* = R_p$. Il primo membro della (3.41) può allora essere riscritto con la (3.48) ed ottenere l'equazione non lineare (3.49).

$$\frac{Q_B}{Q_{f,B}} = \frac{(1-x)Q}{(1-x_f)Q_f} = \frac{(1-x)}{(1-x_f)} \cdot \frac{x_f(1-R_p^*)}{x} = \frac{x_f(1-R_p^*)}{i(1-x_f)} \quad (3.48)$$

$$\frac{x_f(1-R_p^*)}{i(1-x_f)} = \left(\frac{u_f-E/D}{u-E/D}\right)^G \left(\frac{u_f-\alpha+F}{u-\alpha+F}\right)^S \left(\frac{u_f-F}{u-F}\right)^T \quad (3.49)$$

Manualmente, l'equazione 3.49 può essere risolta attraverso un metodo del tipo trial&error, quindi partendo da un primo valore di tentativo $x_r \Rightarrow i_r = i(x_r)$ e risolvendo la (3.49) per ottenere un valore di R_p^* da confrontare con R_p assegnato. Il calcolo si ripete con nuovi tentativi di x_r fino a quando non si verifica che $R_p^* = R_p$. In MATLAB® la soluzione del problema si semplifica in quanto l'equazione 3.49 può essere facilmente risolta sfruttando la funzione fzero, ponendo $R_p^* = R_p$ e rimanendo con i_r come unica incognita. La funzione fzero calcola un'approssimazione della radice dell'equazione $f(i_r) = 0$ con una certa tolleranza tol e a partire da un'approssimazione iniziale $i_{r,0}$. Una volta trovato i_r , gli altri parametri si calcolano di conseguenza secondo le equazioni già indicate. Le equazioni di bilancio (3.3) e (3.4) e la definizione (3.1) rimangono valide. Si ha allora che:

$$x_r = \frac{i_r}{i_r+1} \quad (3.50)$$

$$Q_r = \frac{(1-R_p)Q_f x_f}{x_r} \quad (3.51)$$

$$Q_p = Q_f - Q_r \quad (3.52)$$

$$y_p = \frac{R_p Q_f x_f}{Q_p} \quad (3.53)$$

Per il calcolo dell'area di permeazione si utilizzano le espressioni (3.38) e (3.48) all'interno della (3.28), scrivendo opportunamente il tutto in funzione di i e ottenendo l'espressione per il calcolo dell'area (3.55).

$$dQ_B = \frac{Q_B}{f(i)-i} di = \frac{x_f(1-R_p^*)Q_{f,B}}{(1-x_f)} \frac{di}{i[f(i)-i]} = -\frac{P_B}{l} \left[p_h \frac{1}{i+1} - p_l \frac{1}{f(i)+1} \right] dA_m \quad (3.54)$$

$$A_m = \frac{lQ_f}{p_h P_B} \int_{i_r}^{i_f} \frac{x_f(1-R_p^*)}{i[f(i)-i] \cdot \left[\frac{1}{1+i} - \beta \frac{1}{f(i)+1} \right]} di = \frac{lQ_f}{p_h P_B} \int_{i_r}^{i_f} \mathcal{F}(i) di \quad (3.55)$$

in cui per $f(i)$ si utilizza la (3.32) e per R_p^* la (3.49). Il valore dell'integrale per il calcolo dell'area della membrana può essere determinato in MATLAB® grazie alla funzione `integral`. `integral` permette di integrare numericamente la funzione specificata $\mathcal{F}(i)$ tra gli estremi di integrazione i_r e i_f mediante una formula di quadratura di tipo adattativo.

Caso 2: Anche questa volta si vuole riformulare il modello *cross-flow* per il caso in cui al problema si fornisce il recupero di B nel retentato, invece che il recupero di A nel permeato. Si sceglie quindi di riferire le frazioni molari al componente B. Il procedimento è lo stesso spiegato per il caso 1, ma questa volta le espressioni riferite al componente A saranno quelle scritte per il componente B, e viceversa. Ne risulta che nelle espressioni in cui compare α si sostituisce α^{-1} e che per la definizione del recupero (3.2) la (3.48) diventa:

$$\frac{Q_A}{Q_{f,A}} = \frac{(1-x)Q}{(1-x_f)Q_f} = \frac{(1-x)}{x} \frac{x_f}{(1-x_f)} \frac{Q \cdot x}{Q_f x_f} = \frac{x_f R_r^*}{i(1-x_f)} \quad (3.56)$$

E l'equazione non lineare da risolvere, che ha i come incognita, si esprime secondo la (3.57).

$$\frac{x_f R_r^*}{i(1-x_f)} = \left(\frac{u_f - E/D}{u - E/D} \right)^G \left(\frac{u_f - \alpha^{-1} + F}{u - \alpha^{-1} + F} \right)^S \left(\frac{u_f - F}{u - F} \right)^T \quad (3.57)$$

con

$$D = \frac{1}{2} [(1 - \alpha^{-1})\beta + \alpha^{-1}] \quad (3.58)$$

$$F = -\frac{1}{2} [(1 - \alpha^{-1})\beta - 1] \quad (3.59)$$

$$E = \frac{1}{2\alpha} - DF \quad (3.60)$$

$$S = \frac{\alpha^{-1}(D-1)+F}{(2D-1)(1/2\alpha-F)} \quad (3.61)$$

E u , G , T date da (3.45), (3.42) e (3.44), rispettivamente. Una volta trovato i_r , gli altri parametri di interesse sono dati dalla (3.50), la (3.52) e da quanto segue:

$$Q_r = \frac{R_r Q_f x_f}{x_r} \quad (3.62)$$

$$y_p = \frac{(1-R_r)Q_f x_f}{Q_p} \quad (3.63)$$

Seguendo lo stesso ragionamento già spiegato, per il calcolo dell'area della membrana si utilizzerà l'espressione (3.65).

$$dQ_A = \frac{Q_A}{f(i)-i} di = \frac{x_f R_r^* Q_{f,A}}{(1-x_f) i[f(i)-i]} \frac{di}{i} = -\frac{P_A}{l} \left[p_h \frac{1}{i+1} - p_l \frac{1}{f(i)+1} \right] dA_m \quad (3.64)$$

$$A_m = \frac{l Q_f}{p_h P_A} \int_{i_r}^{i_f} \frac{x_f R_r^*}{i[f(i)-i] \left[\frac{1}{1+i} - \beta \frac{1}{f(i)+1} \right]} di \quad (3.65)$$

L'implementazione in MATLAB® del modello *cross-flow* per i due casi è riportata nell'APPENDICE B.

3.2.3 Modello controcorrente

Nel modello controcorrente entrambi i flussi ai due lati della membrana sono considerati in *plug-flow*, e scorrono in versi opposti. La derivazione del modello teorico è stata affrontata, tra gli altri, da Walawender e Stern [19]. Una rappresentazione dei flussi in un modello di membrana in controcorrente è fornita dalla FIGURA 3.12.

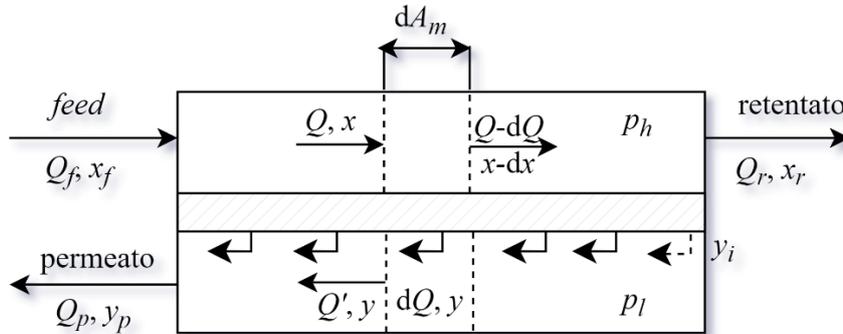


Figura 3.12 Rappresentazione schematica del modello in controcorrente

Si descriveranno i due casi che sono stati implementati in MATLAB® per la soluzione di un modulo in controcorrente, quando l'obiettivo della separazione è il permeato e quando invece ad interessarci è il retentato.

Caso 1: Al sistema vengono fornite le pressioni operative ai due lati della membrana (p_h e p_l), le caratteristiche della corrente alimentata (frazione molare di A x_f e portata Q_f) lo spessore della membrana l e le permeabilità dei due gas (P_A e P_B), il recupero di A nel permeato R_p (3.1). Le frazioni molari saranno riferite al componente A. Per risolvere il sistema, si consideri innanzitutto un elemento di volume differenziale, come indicato in FIGURA 3.12. Si scrive il bilancio di materia totale (3.66) e quello per il componente A (3.67) tra il volumetto differenziale e il retentato.

$$\begin{cases} Q = Q_r + Q' & (3.66) \\ Qx = Q_r x_r + Q'y & (3.67) \end{cases}$$

Dalla differenziazione della (3.67) si ottiene la (3.68).

$$d(Qx) = 0 + d(Q'y) \Rightarrow d(Qx) = d(Q'y) \quad (3.68)$$

Si può poi fare un bilancio del componente A nel volumetto, tra i lati ad alta e a bassa pressione della membrana (3.69), da cui si ottiene l'espressione 3.70 (trascurando il termine $dQdx$).

$$Qx = (Q - dQ)(x - dx) + dQ \cdot y \quad (3.69)$$

$$ydQ = xdQ + Qdx = d(Qx) \quad (3.70)$$

Tenendo presente il meccanismo di soluzione-diffusione per la permeazione del gas attraverso la membrana, il flusso di A attraverso l'elementino di area dA_m può essere scritto come nella (3.71), a cui si combinano la (3.70) e la (3.68).

$$-ydQ = -d(Qx) = -d(Q'y) = \frac{P_A}{l} (p_h x - p_l y) dA_m \quad (3.71)$$

Allo stesso modo si può scrivere per il componente B la relazione (3.72).

$$-(1 - y)dQ = -d[(1 - x)Q] = -d[(1 - y)Q'] = \frac{P_B}{l} [p_h(1 - x) - p_l(1 - y)] dA_m \quad (3.72)$$

Mettendo a sistema le prime due equazioni 3.66 e 3.67 si può ottenere una relazione in cui Q' viene eliminato (3.73), e a cui si può moltiplicare dx (3.74).

$$Qx = Q_r x_r + (Q - Q_r)y \Rightarrow Q(x - y) = Q_r(x_r - y) \quad (3.73)$$

$$Q_r dx = \frac{x-y}{y-x_r} (-Q dx) \quad (3.74)$$

Si può dimostrare la validità dell'espressione 3.75:

$$-Q dx = -(1-x)d(Qx) + xd[Q(1-x)] \quad (3.75)$$

La (3.75) può essere riscritta sfruttando le relazioni (3.71) e (3.72):

$$-Q dx = \frac{P_A}{l} (p_h x - p_l y)(1-x) dA_m - \frac{P_B}{l} [p_h(1-x) - p_l(1-y)] x dA_m \quad (3.76)$$

E sostituendo la (3.76) nella relazione (3.74):

$$Q_r dx = \frac{x-y}{y-x_r} \left\{ \frac{P_A}{l} (p_h x - p_l y)(1-x) - \frac{P_B}{l} [p_h(1-x) - p_l(1-y)] x \right\} dA_m \quad (3.77)$$

Dall'espressione (3.77), moltiplicando entrambi i membri per x_r e spostando i termini noti a sinistra, scrivendo le permeabilità e le pressioni in termini di α (2.18) e β (3.12), si ottiene finalmente una relazione (3.78) che descrive la variazione della composizione x lungo la superficie della membrana dal lato ad alta pressione.

$$\frac{x_r Q_r l}{p_l P_B} \frac{dx}{dA_m} = x_r \frac{x-y}{y-x_r} \{ \alpha(\beta^{-1}x - y)(1-x) - [\beta^{-1}(1-x) - (1-y)]x \} \quad (3.78)$$

Per semplicità, si racchiudono i termini noti al primo membro della (3.78) in un'unica costante K (3.79), il cui prodotto $x_r Q_r$ può essere espresso in termini del recupero (3.1) e sfruttando l'equazione di bilancio (3.4).

$$K = \frac{x_r Q_r l}{p_l P_B} = \frac{(1-R_p) Q_f x_f l}{p_l P_B} \quad (3.79)$$

Seguendo la stessa procedura, si può ottenere un'espressione simile per la composizione del permeato. Si parte dal sistema di bilancio (3.66, 3.67) per ottenere un'equazione in cui non compare Q e si moltiplica dy ambo i membri (3.81).

$$(Q_r + Q')x = Q_r x_r + Q' y \Rightarrow Q_r(x - x_r) = Q'(y - x) \quad (3.80)$$

$$Q_r dy = \frac{x-y}{x-x_r} (-Q' dy) \quad (3.81)$$

Anche in questo caso si può dimostrare valida l'espressione (3.82)

$$Q' dy = (1-y)d(Q'y) - yd[Q'(1-y)] \quad (3.82)$$

per sostituirla all'interno della (3.81) insieme alle espressioni delle velocità di permeazione (3.71) e (3.72):

$$Q_r dy = \frac{x-y}{x-x_r} \left\{ \frac{P_A}{l} (p_h x - p_l y)(1-y) - \frac{P_B}{l} [p_h(1-x) - p_l(1-y)] y \right\} dA_m \quad (3.83)$$

Ne deriva l'equazione differenziale che descrive la variazione della composizione del permeato lungo la membrana (3.84), che messa a sistema con la (3.78) permette di risolvere il problema e ottenere le composizioni di permeato e retentato.

$$K \frac{dy}{dA_m} = x_r \frac{x-y}{x-x_r} \{ \alpha(\beta^{-1}x - y)(1-y) - [\beta^{-1}(1-x) - (1-y)]y \} \quad (3.84)$$

Il sistema di equazioni differenziali può essere risolto numericamente, scegliendo la x come variabile indipendente. L'idea è quella di implementare una procedura del tipo *trial&error*, in

cui si assume un valore di primo tentativo di x_r^* e si risolve il sistema, ottenendo un valore per y_p . Dato che le equazioni di bilancio scritte per il sistema in perfetta miscelazione rimangono valide, con y_p si può calcolare il corrispondente valore di x_r dalla relazione (3.9). Se x_r è in disaccordo col valore assunto x_r^* , la procedura si ripete con il nuovo valore di x_r , fino a quando la tolleranza desiderata tra i due termini non viene soddisfatta. Si considera più conveniente, quindi, riscrivere le equazioni del sistema come segue: la prima (3.85) sarà il rapporto tra la (3.84) e la (3.78); la seconda equazione (3.86), utilizzata per il calcolo dell'area della membrana, si ottiene invertendo la (3.78).

$$\frac{dy}{dx} = \frac{y-x_r \{ \alpha(\beta^{-1}x-y)(1-y) - [\beta^{-1}(1-x) - (1-y)]y \}}{x-x_r \{ \alpha(\beta^{-1}x-y)(1-x) - [\beta^{-1}(1-x) - (1-y)]x \}} \quad (3.85)$$

$$\frac{dA_m}{dx} = \frac{K}{x_r} \frac{(y-x_r)/(x-y)}{\alpha(\beta^{-1}x-y)(1-x) - [\beta^{-1}(1-x) - (1-y)]x} \quad (3.86)$$

Il problema da risolvere si riconduce quindi a un sistema di equazioni differenziali a condizioni iniziali, e cioè a partire da y e A_m valutate in $x=x_r$. Il valore dell'area non è noto, per cui si setta arbitrariamente $A_m=0$ per $x=x_r$, ottenendo dalla soluzione numerica un valore negativo dell'area, a cui andrà cambiato il segno. Il valore di y nella sezione di *outlet* del retentato verrà indicato come y_i e può essere calcolato dalla (3.71) e la (3.72), il cui rapporto fornisce l'equazione (3.87), come già visto nel modello in perfetta miscelazione.

$$\frac{y_i}{1-y_i} = \frac{\alpha(x_r - \beta y_i)}{(1-x_r) - \beta(1-y_i)} \quad (3.87)$$

La (3.87) può essere espressa in forma quadratica per y_i ed essere risolta (3.88), ottenendo la composizione del permeato y_i in corrispondenza di $x=x_r$.

$$y_i = \frac{(\alpha-1)(\beta^{-1}x_r+1) + \beta^{-1} - \{ [(\alpha-1)(\beta^{-1}x_r+1) + \beta^{-1}]^2 - 4\alpha\beta^{-1}x_r(\alpha-1) \}^{0.5}}{2(\alpha-1)} \quad (3.88)$$

Per la soluzione del sistema, quindi, le equazioni verranno inizialmente valutate in $x=x_r$. Ci si rende immediatamente conto che l'equazione 3.85 all'*outlet* del retentato assume una forma indeterminata. Si può allora scrivere che:

$$\left(\frac{dy}{dx} \right)_{x_r} = \left(\frac{dy}{dA_m} \right)_{A_m=0} / \left(\frac{dx}{dA_m} \right)_{A_m=0} \quad (3.89)$$

e valutare separatamente la (3.84) e la (3.78) alla sezione di *outlet* del retentato. Per valutare $(dy/dA_m)_{A_m=0}$ può essere applicata la regola di de L'Hopital per $A_m \rightarrow 0$, ottenendo:

$$\left(\frac{dy}{dA_m} \right)_{A_m=0} = \frac{(x_r - y_i)\beta^{-1}[\alpha - y_i(\alpha-1)]}{\frac{K}{x_r} \frac{(x_r - y_i)[(\alpha-1)(2y_i - \beta^{-1}x_r - 1) - \beta^{-1}]}{(dx/dA_m)_{A_m=0}}} \quad (3.90)$$

$(dx/dA_m)_{A_m=0}$ si può facilmente ottenere dalla (3.78) sostituendo y_i ad y e x_r ad x :

$$\left(\frac{dx}{dA_m} \right)_{A_m=0} = \frac{1}{K} \left[\frac{\alpha(\beta^{-1}x_r - y_i)(x_r - y_i)x_r}{y_i} \right] \quad (3.91)$$

Per la risoluzione numerica del sistema (3.85, 3.86) si è scelto di operare col seguente cambio di variabile:

$$z = x - x_r \quad (3.92)$$

ottenendo il sistema di equazioni differenziali scritto nella forma (3.93, 3.94).

$$\left\{ \begin{aligned} \frac{dy}{dz} &= \frac{y-x_r}{z} \left\{ \frac{(1-y)\alpha(\beta^{-1}z+\beta^{-1}x_r-y)-y[\beta^{-1}(1-z-x_r)-(1-y)]}{(1-z-x_r)\alpha(\beta^{-1}z+\beta^{-1}x_r-y)-(z+x_r)[\beta^{-1}(1-z-x_r)-(1-y)]} \right\} & (3.93) \\ \frac{dA_m}{dz} &= \frac{K}{x_r} \frac{(y-x_r)/(z+x_r-y)}{(1-z-x_r)\alpha(\beta^{-1}z+\beta^{-1}x_r-y)-(z+x_r)[\beta^{-1}(1-z-x_r)-(1-y)]} & (3.94) \end{aligned} \right.$$

Per il calcolo delle espressioni all'*outlet* di alta pressione continua a valere la (3.89):

$$\left(\frac{dy}{dz} \right)_{z=0} = \left(\frac{dy}{dx} \right)_{x=x_r} = \left(\frac{dy}{dA_m} \right)_{A_m=0} / \left(\frac{dx}{dA_m} \right)_{A_m=0} \quad (3.95)$$

La procedura di calcolo implementata in MATLAB® è la seguente:

1. Si definiscono le costanti α (2.18), β (3.12), K (3.79);
2. Si sceglie un valore di primo tentativo per x_r , ad esempio $x_r^* = 0.5 \cdot x_f$, il numero massimo di iterazioni e la tolleranza per la procedura *trial&error*;
3. Si calcolano y_i (3.88) e $(dy/dz)_{z=0}$ con la (3.95), (3.90) e (3.91);
4. Si risolve il sistema di equazioni differenziali a condizioni iniziali con il metodo numerico di Runge-Kutta a 4 stadi, che è stato definito in un'apposita funzione; la risoluzione del sistema fornisce un valore di y_p^* e un valore di A_m^* ;
5. Si calcola il nuovo valore di x_r da y_p^* utilizzando la relazione (3.9);
6. Si confrontano x_r^* e x_r e se la tolleranza sul valore assoluto non è soddisfatta si imposta il nuovo $x_r^* = x_r$ e si ripete la procedura dal punto 3;
7. Se la tolleranza è soddisfatta, si calcolano anche le portate di permeato e retentato secondo la (3.96) e la (3.97), rispettivamente, e si cambia il segno di A_m .

$$Q_p = \frac{R_p Q_f x_f}{y_p} \quad (3.96)$$

$$Q_r = Q_f - Q_p \quad (3.97)$$

Caso 2: Nel caso in cui il prodotto di interesse è il retentato, si è pensato di fornire al problema oltre ai soliti dati di pressione (p_h, p_l), permeabilità (P_A, P_B), alimentazione (Q_f, x_f riferito ad A), e spessore della membrana (l), la frazione molare di B nel retentato, altrimenti detta purezza del retentato $x_{r,B} = 1 - x_{r,A}$. La purezza, insieme al recupero, costituisce uno dei parametri di processo che si vogliono fissare nel sistema per garantire le prestazioni di separazione desiderate. Le frazioni molari saranno di seguito riferite al componente A. Tutte le equazioni già scritte per il caso 1 (3.66-3.97) rimangono valide, con la differenza che stavolta R_p non è noto, e quindi non lo è neanche la costante K . In realtà il calcolo descritto per il caso 1 si semplifica, in quanto x_r è già noto e la procedura *trial&error* non è più necessaria. Sarà sufficiente risolvere numericamente le equazioni differenziali (3.93) e (3.94). La nuova procedura è descritta di seguito:

1. Si definiscono le costanti α (2.18), β (3.12);
2. Si calcolano y_i (3.88) e $(dy/dz)_{z=0}$ con la (3.95), (3.90) e (3.91);
3. Si risolve la (3.93) con il metodo di Runge-Kutta a 4 stadi, ottenendo la concentrazione di A nel permeato y_p ;

4. Il recupero di A nel permeato R_p è in relazione con x_r e y_p dalla (3.9), può allora essere calcolato grazie all'espressione (3.98) per permettere di determinare le portate di permeato e retentato con le relazioni (3.96) e (3.97), rispettivamente.

$$R_p = \frac{y_p(x_r - x_f)}{x_f(x_r - y_p)} \quad (3.98)$$

5. Può essere utile calcolare il recupero di B nel retentato, di maggior interesse nel caso esaminato, attraverso la (3.99) per definizione:

$$R_r = \frac{Q_r(1 - x_r)}{Q_f(1 - x_f)} \quad (3.99)$$

6. Si calcola K (3.79) e si risolve il sistema di equazioni differenziali (3.93, 3.94) con il metodo di Runge-Kutta a 4 stadi per determinare la superficie della membrana che permette la separazione sopra definita. Il segno dell'area andrà cambiato affinché risulti positiva.

3.2.3.1 Metodo di Runge-Kutta a 4 stadi

I metodi di Runge-Kutta sono una classe importante di metodi numerici a un passo per la risoluzione di un problema di Cauchy. Per problema di Cauchy si intende un problema matematico in cui devono essere determinate le funzioni (con derivate prime continue) soluzione di un sistema di equazioni differenziali e passanti per i punti definiti dalle condizioni iniziali indicate. Un problema di Cauchy si definisce scritto in forma canonica quando si trova nella seguente forma:

$$\begin{cases} \mathbf{r}'(z) = \mathbf{f}(z, \mathbf{r}(z)) & z > a \\ \mathbf{r}(a) = \mathbf{r}_0 \end{cases} \quad (3.100)$$

Dove \mathbf{r} è il vettore contenente le funzioni da determinare (nel nostro caso $y(z)$ e $A_m(z)$), con \mathbf{r}' si indicano le loro derivate ed \mathbf{r}_0 è il vettore colonna con le condizioni iniziali (quindi y_i e 0 in $z=0$). I metodi numerici considerano una certa suddivisione dell'intervallo di integrazione, con passo h costante o variabile:

$$a \equiv z_0 < z_1 < \dots < z_N \quad z_{n+1} = z_n + h$$

e forniscono i valori \mathbf{r}_n , che sono approssimazioni della soluzione $\mathbf{r}(z)$ nei punti della partizione definiti z_n . Nei metodi numerici a un passo, come quelli di Runge-Kutta, l'approssimazione della soluzione \mathbf{r}_{n+1} dipende unicamente dall'approssimazione \mathbf{r}_n calcolata al passo precedente:

$$\mathbf{r}_{n+1} = \mathbf{r}_n + h\phi(z_n, \mathbf{r}_n; h, f) \quad n = 0, 1, 2, \dots, N - 1 \quad (3.102)$$

L'espressione (3.102) definisce un metodo a un passo esplicito. La funzione ϕ dipende dal metodo scelto. Per i metodi di Runge-Kutta, $\phi(z_n, \mathbf{r}_n; h, f)$ è una media pesata di valori di $\mathbf{r}'(z)$ in un intorno destro di z_n . Il generico metodo di Runge-Kutta a s stadi, infatti, può essere scritto con la seguente forma:

$$\begin{cases} \mathbf{r}_{n+1} = \mathbf{r}_n + h \sum_{i=1}^s a_i \mathbf{K}_i & (3.103) \\ \mathbf{K}_1 = \mathbf{f}(z_n, \mathbf{r}_n) & (3.104) \\ \mathbf{K}_i = \mathbf{f}(z_n + b_i h, \mathbf{r}_n + h \sum_{j=1}^{i-1} c_{ij} \mathbf{K}_j) & i = 2, \dots, s \end{cases} \quad (3.105)$$

I tre coefficienti a_i , b_i e c_{ij} vengono scelti in modo tale che il metodo risulti convergente e abbia un ordine di convergenza elevato. In particolare, il metodo di Runge-Kutta a 4 stadi utilizzato in questo lavoro è il seguente:

$$\left\{ \begin{array}{l} \mathbf{r}_{n+1} = \mathbf{r}_n + \frac{h}{6}(\mathbf{K}_1 + 2\mathbf{K}_2 + 2\mathbf{K}_3 + \mathbf{K}_4) \end{array} \right. \quad (3.106)$$

$$\mathbf{K}_1 = \mathbf{f}(z_n, \mathbf{r}_n) \quad (3.107)$$

$$\mathbf{K}_2 = \mathbf{f}\left(z_n + \frac{h}{2}, \mathbf{r}_n + \frac{h}{2}\mathbf{K}_1\right) \quad (3.108)$$

$$\mathbf{K}_3 = \mathbf{f}\left(z_n + \frac{h}{2}, \mathbf{r}_n + \frac{h}{2}\mathbf{K}_2\right) \quad (3.109)$$

$$\mathbf{K}_4 = \mathbf{f}(z_n + h, \mathbf{r}_n + h\mathbf{K}_3) \quad (3.110)$$

Va ricordato che poiché nel problema specifico del modello in controcorrente la funzione $r_i'(z)$ assume una forma indeterminata in $z=0$, alla funzione che implementa il metodo di Runge-Kutta andrà fornito oltre al vettore $\mathbf{r}'(z)=\mathbf{f}(z,\mathbf{r}(z))$ anche il vettore $\mathbf{r}'(z=0)=\mathbf{f}_0(z,\mathbf{r}(z))$, il cui primo termine è la costante data dalle (3.95, 3.90, 3.91). In realtà, la funzione f_0 non verrà utilizzata solo per il calcolo di \mathbf{r}_0 in $z=0$, ma anche per i termini \mathbf{r}_n corrispondenti a valori di z prossimi allo 0. In caso contrario, infatti, si è osservato che errori di tipo numerico nei primi k passi si propagano nei passi successivi e non permettono la convergenza del metodo.

In APPENDICE C si riportano le funzioni scritte in MATLAB® per la descrizione del modello in controcorrente, in ciascuno dei due casi, e la funzione implementata per la risoluzione di un sistema di equazioni differenziali con il metodo di Runge-Kutta a 4 stadi.

3.3 Validazione dei modelli e analisi parametrica

L'obiettivo del presente paragrafo sarà quello di convalidare i tre modelli appena descritti utilizzando i dati sperimentali disponibili in letteratura. Tranchino et al. [20] hanno assemblato un modulo a membrana da laboratorio per testarlo sotto diverse condizioni operative e regimi di flusso. Nello specifico, è stata utilizzata una membrana composita organizzata in un modulo a fibra cava, per un totale di 100 fibre e 300 [cm²] di superficie attiva, all'interno di un cilindro in acciaio inox di 1 [cm] di diametro (interno) e 15 [cm] di lunghezza utile. Il gas testato è una miscela CO₂-CH₄ e i risultati sono stati registrati allo stato stazionario. Le permeabilità dei due gas sono state determinate testandoli individualmente, in diverse condizioni di temperatura e di pressione. Mantenendo pressione, temperatura e composizione di alimentazione costanti, ed essendo costante la superficie della membrana, variare la portata di alimentazione permette di testare il modulo per diversi valori di recupero della CO₂. La composizione del permeato sarà funzione esclusivamente del recupero, il quale caratterizzerà il regime di flusso all'interno del modulo. Si è verificato che per valori di recupero molto bassi la composizione ai due lati della membrana si può considerare uniforme, e il modulo può essere descritto con il modello in perfetta miscelazione. Sono state eseguite alcune prove a diverse differenze di pressione transmembrana (1,3,6 [atm]) e a due temperature (25 e 65°C) in condizioni di recupero molto basso (≈ 0), in modo da poter verificare la validità del modello in perfetta miscelazione. In TABELLA 3.1 sono riportate le condizioni operative e i dati inseriti nel modello per il confronto con le prove sperimentali. In TABELLA 3.2 sono invece presentati gli errori relativi percentuali tra il modello e i risultati sperimentali. In FIGURA 3.13 si può infine osservare, in condizioni di basso recupero, la dipendenza della composizione del permeato con la pressione, per ciascuna delle due temperature esaminate. Il grafico è stato realizzato prendendo i valori medi delle permeabilità e del recupero presentati in TABELLA 3.1. Secondo il modello teorico, l'aumento della pressione transmembrana comporta un aumento sia della portata di permeato che del suo contenuto di CO₂. La curva teorica è in accordo con i dati sperimentali solo per una differenza di pressione inferiore a 3 [atm], dopodiché la composizione di permeato sperimentale si

mantiene circa costante. Questo discostamento è dovuto a effetti di non idealità, non considerati nel modello, come ad esempio eventuali cambiamenti strutturali delle fibre sottoposte ad alta pressione, che influenzano selettività e velocità di permeazione. A una temperatura operativa più alta ciò che si verifica è che aumenta la velocità di permeazione ma diminuisce la frazione molare di CO₂ nel permeato, e ciò è concorde con i valori di permeabilità e selettività in TABELLA 3.1. La curva riferita a una temperatura di 65°C si adatta meglio ai valori sperimentali, avendo un coefficiente di determinazione $R^2=0.7950$, rispetto all' $R^2=0.5548$ a 25°C.

Tabella 3.1 Dati utilizzati per la convalida del modello in perfetta miscelazione [20]

	$T=25\text{ °C}$	$T=65\text{ °C}$
p_h [ata]	2; 4; 7	2; 4; 7
p_l [ata]	1	1
Pe_{CO_2} [cm _{STP} ³ min ⁻¹ cm ⁻² atm ⁻¹]	0.043; 0.043; 0.044	0.081; 0.082; 0.086
Pe_{CH_4} [cm _{STP} ³ min ⁻¹ cm ⁻² atm ⁻¹]	0.0118; 0.0120; 0.0127	0.0269; 0.0283; 0.0307
Q_f ** [cm _{STP} ³ min ⁻¹]	1333; 4200; 8633	2733; 8267; 17000
x_{f,CO_2}	0.6	0.6
R_p **	0.0064; 0.0067; 0.0067	0.0061; 0.0064; 0.0064

**Parametri calcolati da altri dati presenti nel riferimento

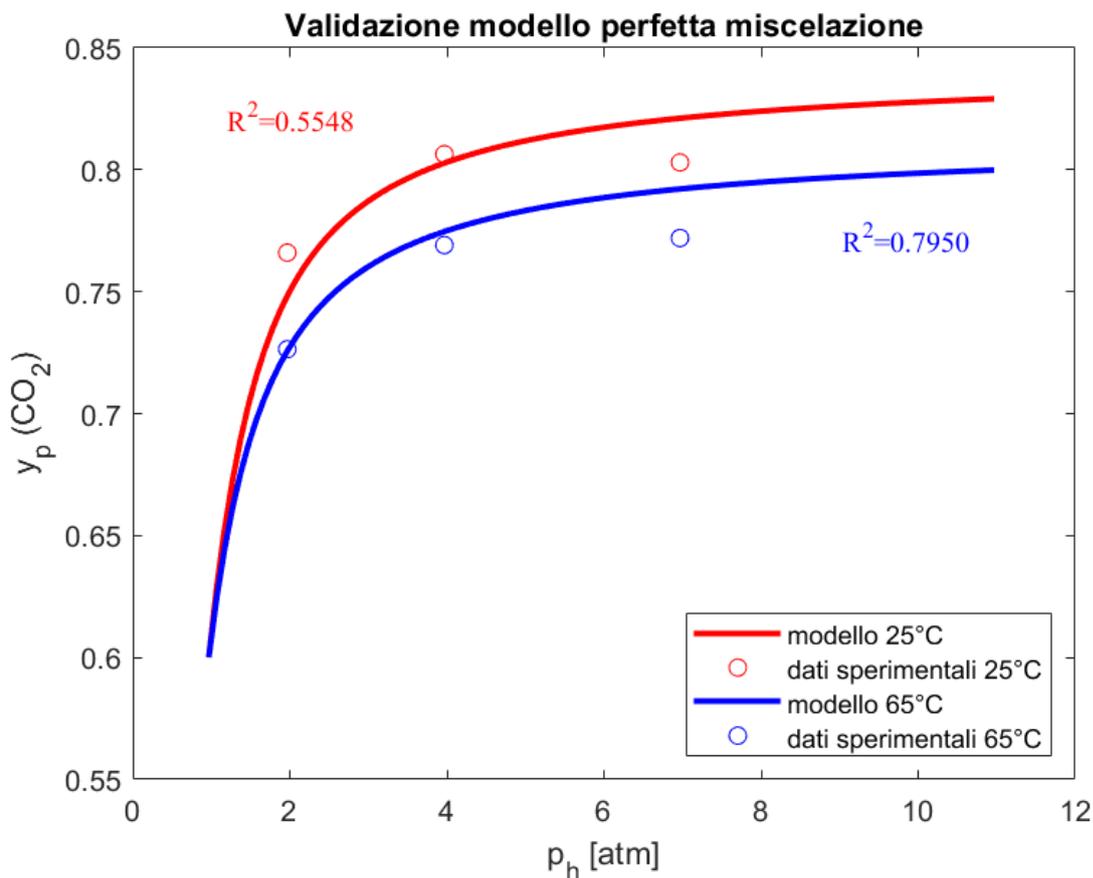


Figura 3.13 Confronto tra le curve teoriche e i risultati sperimentali [20] della frazione molare di CO₂ nel permeato (y_p) in diverse condizioni di pressione (lato feed/retentato); $p_l=1$ [ata]; $R_p \approx 0$; alimentazione 60% CO₂ e 40% CH₄; permeabilità e selettività diverse per le due temperature (TABELLA 3.1)

Tabella 3.2 Errori relativi percentuali della frazione molare di CO₂ nel permeato e dell'area della membrana. Valutati dal confronto tra i risultati sperimentali [20] e i valori calcolati dal modello in perfetta miscelazione

Δp [atm]	$T=25^{\circ}\text{C}$			$T=65^{\circ}\text{C}$		
	1	3	6	1	3	6
ε_r per y_p [%]	1.99	0.36	1.79	0.48	0.71	1.94
ε_r per A_m [%]	12.32	17.90	22.26	13.54	20.71	25.51

Lo stesso apparato sperimentale è stato poi fatto operare sia in equicorrente che in controcorrente per valutare la composizione del permeato in diversi regimi di flusso, e quindi per diversi valori di recupero. Questi dati sono stati qui utilizzati per convalidare il modello in controcorrente sviluppato per il “caso 1”, e cioè quando si è interessati al permeato e si fornisce al sistema il recupero R_p come variabile indipendente. La FIGURA 3.14 rappresenta la dipendenza della purezza del permeato y_p dal recupero R_p di CO₂ per il modello teorico in controcorrente, insieme a quanto è stato trovato sperimentalmente. I parametri utilizzati per l’ottenimento del grafico sono descritti in TABELLA 3.3. Si può affermare che nelle condizioni descritte il modello teorico in controcorrente descrive abbastanza bene il sistema, avendo un coefficiente di determinazione pari a 0.9642.

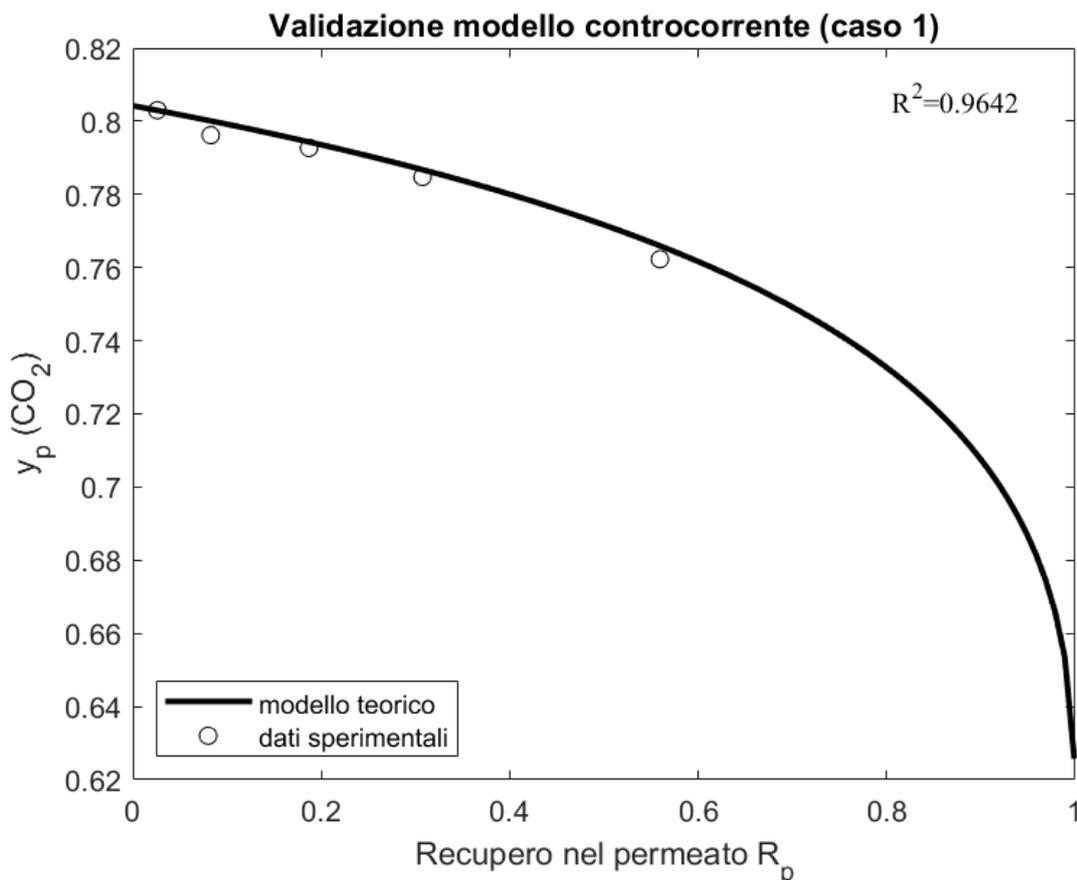


Figura 3.14 Confronto tra i risultati sperimentali [20] e quanto calcolato dal modello teorico in controcorrente (caso 1) per la dipendenza della composizione del permeato dal recupero; $\Delta p=3$ [atm]; $T=28^{\circ}\text{C}$; alimentazione 60% CO₂ e 40% CH₄; $\alpha=3.58$.

La FIGURA 3.14 dimostra anche il *trade-off* tra purezza e recupero in un sistema di separazione a membrana: in questo caso, per avere un recupero molto alto, la CO₂ nella corrente di permeato non può superare il 70%.

Tabella 3.3 Parametri utilizzati per la convalida del modello in controcorrente (caso 1) [20]

T [°C]	28
p_h [ata]	4
p_l [ata]	1
x_{f,CO_2}	0.6
Pe_{CO_2} [cm _{STP} ³ min ⁻¹ cm ⁻² atm ⁻¹]	0.043
Pe_{CH_4} [cm _{STP} ³ min ⁻¹ cm ⁻² atm ⁻¹]	0.0120

Nel lavoro di Feng et al. [21] sono stati riportati i risultati dei test sperimentali sia per la corrente di permeato che per la corrente di retentato, il che permette di convalidare il modello in controcorrente per il secondo caso, quello in cui al sistema viene fornita la purezza del retentato. L'apparato sperimentale è costituito da una membrana asimmetrica anche in questo caso organizzata nel modulo in forma di fibra cava. È stato utilizzato un fascio di 368 fibre cave, con lunghezza utile di 25 [cm] e un diametro esterno nominale di 160 [μm], per una superficie attiva della membrana di 462 [cm²]. La miscela testata è O₂-N₂ e le permeabilità dei singoli gas sono state determinate attraverso test individuali. La miscela è stata testata in diversi regimi di flusso, variando la portata alimentata. La FIGURA 3.15 mostra la relazione tra la purezza del retentato e il rispettivo recupero di N₂, calcolata dal modello teorico in controcorrente (caso 2) e confrontata con i risultati sperimentali. I dati utilizzati sono riportati in TABELLA 3.4. In queste condizioni il modello adatta molto bene i dati sperimentali, con un coefficiente di determinazione pari a 0.9807. Con altri dati presenti nel riferimento, per le stesse prove rappresentate in FIGURA 3.15, è stato possibile ricavare i valori di portata alimentata e quindi calcolare l'area della membrana restituita dal modello teorico. Gli errori relativi nel calcolo del recupero R_r e dell'area A_m con il modello teorico, rispetto ai valori sperimentali, sono raccolti in TABELLA 3.5. Anche in questo caso è evidente come per ottenere una buona purezza del retentato, il recupero e la produttività di N₂ saranno minimi.

Tabella 3.4 Dati utilizzati per la convalida del modello in controcorrente (caso 2) [21]

T [°C]	23
p_h [kPa]	690
p_l [kPa]	100
x_{f,O_2}	0.205
Pe_{O_2} [GPU]	9.30
Pe_{N_2} [GPU]	1.80

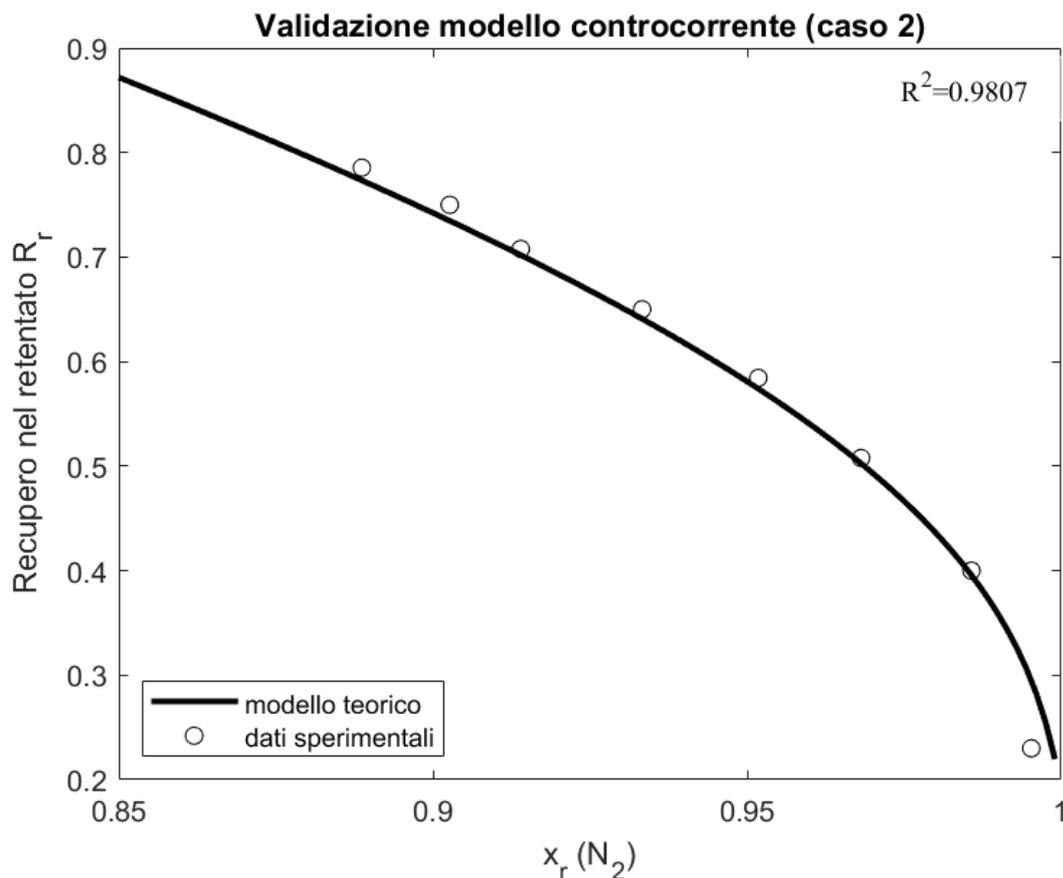


Figura 3.15 Relazione tra il recupero di N_2 nel retentato e la sua purezza, calcolata dal modello teorico in controcorrente (caso 2) e confrontata con i valori sperimentali [21]; $T=23^\circ\text{C}$; $\Delta p=590$ [kPa]; alimentazione 20.5% O_2 e 79.5% N_2 ; $\alpha=5.17$.

Tabella 3.5 Errori relativi nel calcolo del recupero di N_2 nel retentato e della superficie della membrana utilizzando il modello in controcorrente (caso 2), rispetto ai punti sperimentali [21] rappresentati in FIGURA 3.15, letta da sinistra verso destra.

Dato sperimentale	1	2	3	4	5	6	7	8
ε_r per R_r [%]	1.55	2.04	0.84	1.40	1.78	1.04	1.08	28.26
ε_r per A_m [%]	5.00	5.63	11.17	5.02	7.00	7.67	9.95	18.65

Quando il modulo è sottoposto a un alto flusso di permeazione, la miscelazione locale del permeato è trascurabile e il sistema può essere descritto più appropriatamente con un modello del tipo *cross-flow*, indipendentemente dalla direzione del flusso al di fuori dello strato poroso della membrana (ci si riferisce qui a una membrana asimmetrica). Per la convalida del modello *cross-flow* è stato utilizzato il lavoro di Sidhoum et al. [22], i quali fanno uso di una membrana asimmetrica in acetato di cellulosa in forma di fibre cave, per la separazione di una miscela CO_2-N_2 . La FIGURA 3.16 rappresenta il confronto tra i risultati sperimentali e il modello teorico in *cross-flow*, diagrammando la purezza del permeato in funzione del recupero di CO_2 . I dati utilizzati per la convalida del modello sono in TABELLA 3.6. Anche in questo caso il modello si può ritenere adeguato, con un coefficiente di determinazione di 0.9139.

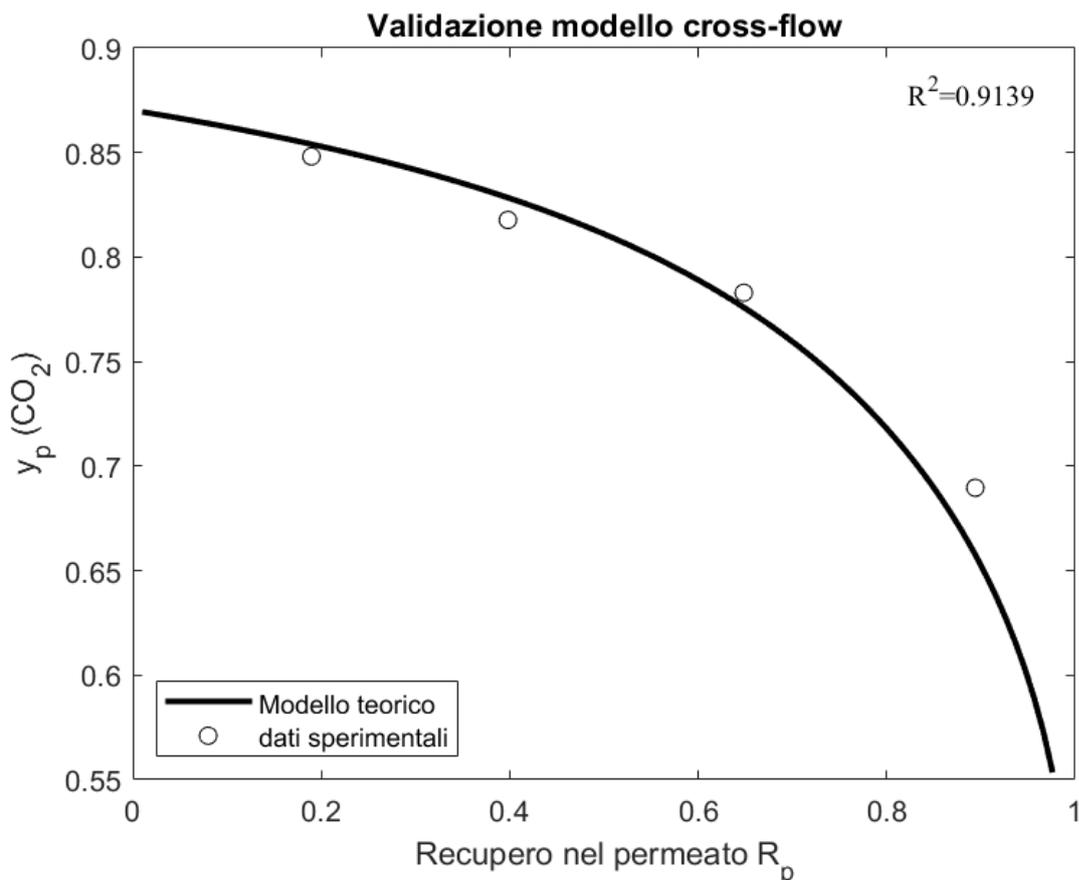


Figura 3.16 Purezza del permeato in funzione del recupero di CO_2 , confronto tra i risultati ottenuti dal modello teorico in *cross-flow* e i dati sperimentali [22]; $T=25^\circ\text{C}$; $\Delta p=303$ [kPa]; alimentazione 40% CO_2 60% N_2 ; $\alpha=20.85$.

Tabella 3.6 Dati utilizzati per la convalida del modello in *cross-flow* [22]

T [$^\circ\text{C}$]	25
p_h [kPa]	404
p_l [kPa]	101
x_{f, CO_2}	0.40
Pe_{CO_2} [GPU]	63.6
Pe_{N_2} [GPU]	3.05

I modelli teorici esposti si basano tutti su un comportamento ideale del sistema. Eventuali discostamenti da quanto viene osservato sperimentalmente sono conseguenza del verificarsi di eventi di non idealità, come:

- La permeazione competitiva dei componenti nella miscela, per cui permeabilità e selettività differiscono da quelle misurate nelle prove sui componenti puri;
- L'incremento dei difetti nella membrana, sottoposta ad alte pressioni e temperature, con variazioni delle caratteristiche di permeazione;
- L'effetto di Joule-Thomson;

- Cadute di pressione lungo il modulo, con conseguente decremento della forza spingente;
- Regimi di flusso non ideali, come zone stagnanti o ostruzioni.

I tre modelli possono essere confrontati per analizzare quali sono le prestazioni di un modulo a membrana, al variare di alcuni parametri di processo (FIGURA 3.17). Per l'analisi parametrica si sceglie una miscela CO₂-N₂ e le caratteristiche della membrana presentate in TABELLA 3.6. Si può considerare un'alimentazione al 13% di CO₂, tipica per un gas di scarico. Ciò che è immediato osservare è che tra i tre modelli il più performante è sempre quello in controcorrente, poiché la condizione di flusso permette di sfruttare nella maniera più efficiente la forza spingente lungo la membrana. Il modello *cross-flow* fornisce dei risultati molto simili a quelli ottenuti dal modello in controcorrente, soprattutto quando le condizioni del sistema permettono di avere una forza spingente maggiore. I risultati calcolati dai due modelli si discostano all'aumentare del recupero e quindi della portata di permeato. A parità delle condizioni di alimentazione e delle caratteristiche della membrana, un incremento della differenza di pressione transmembrana giova sicuramente alle prestazioni del processo, permettendo di ottenere un permeato di maggior purezza grazie a un miglioramento della forza spingente (FIGURA 3.17A). Per lo stesso motivo, l'area della membrana necessaria alla separazione sarà minore (FIGURA 3.17B), il che costituirà un vantaggio di tipo economico. Un'alternativa che permette di migliorare la purezza del permeato, mantenendo costante il recupero, è sicuramente quella di scegliere una membrana più permselectiva, e che quindi favorisca in maniera ancora più preferenziale il passaggio di CO₂ rispetto a quello di N₂ (FIGURA 3.17C). In linea generale, però, la superficie richiesta della membrana aumenta con l'aumentare della selettività (FIGURA 3.17D), in quanto la maggiore concentrazione di CO₂ nel permeato comporta una forza motrice di permeazione minore. La *performance* del sistema di separazione con membrana è anche strettamente dipendente dalle condizioni di alimentazione: una corrente il cui componente più permeabile è già presente con una buona concentrazione, rende la separazione più facile e meno dispendiosa, con una migliore purezza della corrente di permeato (FIGURA 3.17E) e un minore utilizzo di membrana (FIGURA 3.17F). Quando la frazione di CO₂ nella corrente alimentata è molto bassa, come nel tipico caso del trattamento dei gas di scarico di una centrale termoelettrica, la sua forza spingente non sarà sufficiente ad avere una velocità di permeazione competitiva rispetto agli altri componenti della miscela. In ogni caso, è facile osservare come per nessuno dei tre modelli, e in nessuna delle condizioni esposte, sia stato possibile raggiungere gli standard di separazione richiesti dalle normative o comunque adatti ad uno scopo commerciale. Un solo stadio di separazione con membrana risulta inadatto e non competitivo con altri tipi di tecnologie. Si rende allora necessaria la progettazione di un sistema multistadio, con o senza ricircoli.

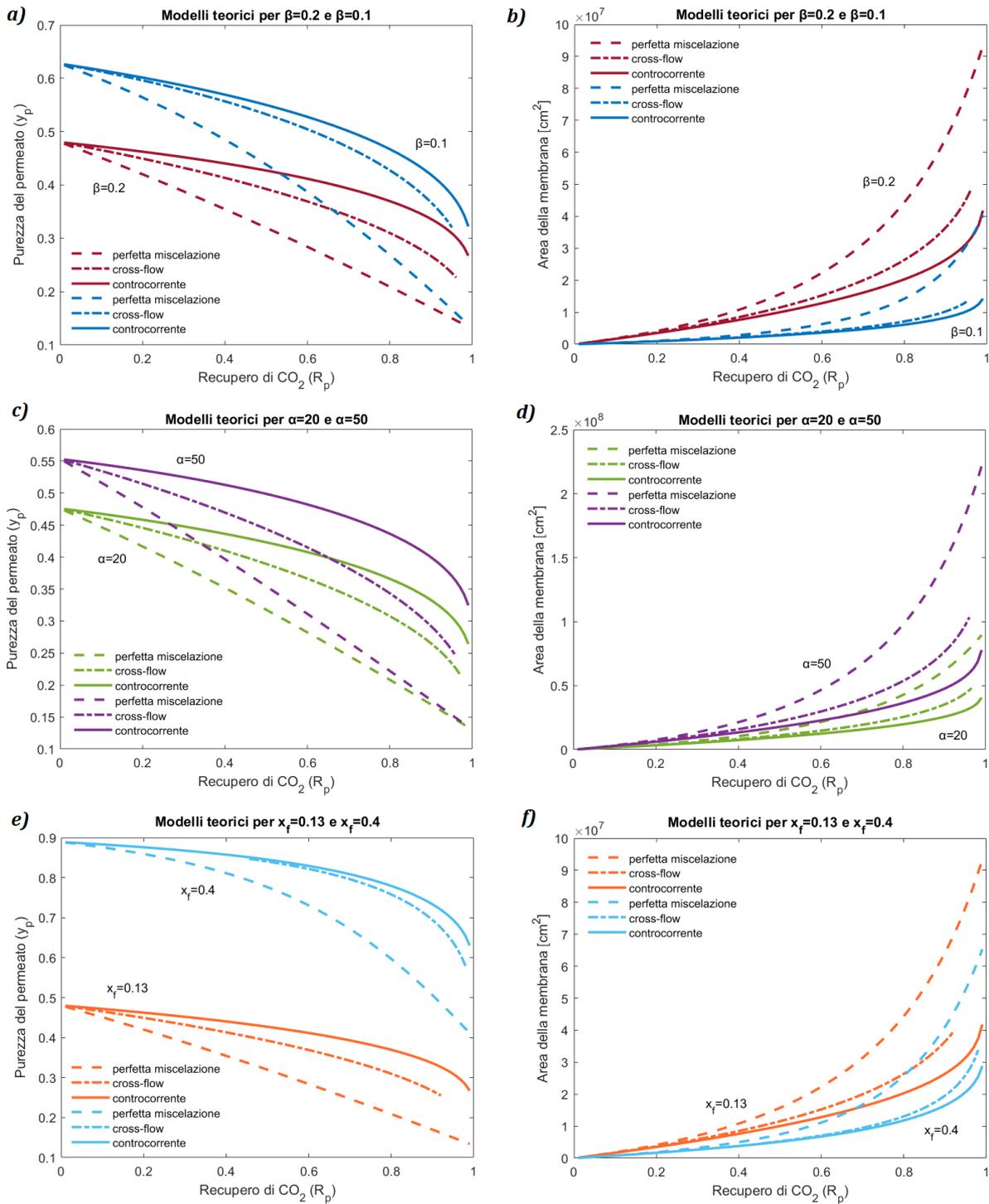


Figura 3.17 Confronto dei tre modelli teorici (perfetta miscelazione, *cross-flow*, controcorrente) in diverse condizioni di processo. Per ciascun caso, valutazione della frazione molare di CO_2 nel permeato e dell'area totale della membrana al variare del recupero di CO_2 . **a,b)** $Pe_{\text{CO}_2}=63.6$ [GPU], $\alpha=20.85$, $Q_f=10^5$ [$\text{cm}_{\text{STP}}^3/\text{s}$], $x_{f,\text{CO}_2}=0.13$, $p_f=1$ [ata], confronto tra due valori di alta pressione (5 e 10 [ata]); **c,d)** $Pe_{\text{CO}_2}=63.6$ [GPU], $Q_f=10^5$ [$\text{cm}_{\text{STP}}^3/\text{s}$], $x_{f,\text{CO}_2}=0.13$, $p_f=1$ [ata], $p_h=5$ [ata], confronto tra due valori di selettività (20 e 30); **e,f)** $Pe_{\text{CO}_2}=63.6$ [GPU], $\alpha=20.85$, $Q_f=10^5$ [$\text{cm}_{\text{STP}}^3/\text{s}$], $p_f=1$ [ata], $p_h=5$ [ata], confronto tra due composizioni di alimentazione (0.13 e 0.4).

4. Sistemi di separazione a membrana con due stadi

Nella progettazione di un processo di separazione, due importanti fattori da considerare sono il recupero e la purezza del prodotto. In un processo di separazione con membrana, il prodotto di interesse può essere il flusso permeato, e allora si parla di “problema di recupero”, oppure il flusso di retentato, nel caso del cosiddetto “problema di purificazione”. I due problemi dovranno essere formulati in maniera differente, ma in ogni caso il fine della progettazione è il calcolo dell’area di membrana necessaria a realizzare la separazione desiderata. Come la membrana sarà organizzata all’interno dello stadio dipenderà dalla geometria scelta (come spiegato nel capitolo 3) e dalle caratteristiche dei moduli disponibili in commercio. Tuttavia, nel capitolo precedente si è dimostrato come, per una certa superficie attiva di membrana, a una migliore purezza è associato un minor recupero del prodotto, e viceversa. Se il prodotto di interesse è il permeato, aumentare la superficie della membrana ne migliora il recupero ma a discapito della sua purezza. Il contrario accade per il retentato, infatti un aumento della superficie della membrana ne aumenta la purezza ma peggiora il recupero.

Nella pratica industriale, un solo stadio di separazione con membrana si rivela inadeguato. Si ricorre allora alla progettazione di schemi che prevedono il collegamento tra più stadi di separazione. In alcuni casi si prevede anche il riciclo di alcune correnti tra gli stadi, per migliorare le caratteristiche del prodotto. In questo lavoro si considereranno solo schemi con due stadi di separazione, e in particolare nelle seguenti quattro disposizioni [23]:

1. Due stadi di separazione in serie
2. Due stadi di separazione in cascata
3. Due stadi di separazione in serie, con ricircolo del permeato del secondo stadio
4. Due stadi di separazione in cascata, con ricircolo del retentato del secondo stadio

Questi quattro schemi verranno descritti in seguito più nel dettaglio.

Si è visto come alla base della separazione con membrana vi è la necessità di fornire una differenza di pressione tra i due lati della membrana. Questo può essere garantito:

- a) comprimendo il gas da alimentare allo stadio, mentre la camera di permeato sarà a pressione atmosferica; lo schema prevede allora un compressore a monte dello stadio di separazione;
- b) mantenendo il vuoto sul lato del permeato; si prevede allora una pompa a vuoto che agisce sulla corrente di permeato;
- c) inviando uno *sweep gas* dal lato del permeato, in modo da ridurre la concentrazione e quindi la pressione parziale del componente che si vuole separare.

Le prime due opzioni sono le più praticate e quelle che verranno qui considerate.

Per ciascuno dei quattro schemi elencati, allora, dovranno essere considerate quattro configurazioni differenti, a seconda di come la differenza di pressione viene fornita. Quando l’alimentazione viene compressa ad alta pressione, si può pensare ad un recupero energetico accoppiando un espansore sulla corrente di retentato, riportandola a pressione atmosferica. Negli schemi saranno introdotti anche degli scambiatori di calore laddove necessari, in modo che il gas venga alimentato allo stadio sempre alla temperatura operativa adeguata.

4.1 Schemi con due stadi di separazione

Si descriveranno e illustreranno i quattro schemi di separazione a due stadi considerati.

4.1.1 Due stadi in serie

Un sistema di separazione a membrana con due stadi viene definito in serie quando il retentato prodotto dal primo stadio costituisce l'alimentazione del secondo stadio. In FIGURA 4.1 lo schema in serie viene illustrato nelle sue quattro varianti, a seconda dell'utilizzo e posizione di compressori e/o pompe a vuoto. Dal sistema si ottengono due correnti di permeato, che una volta miscelate possiedono le stesse caratteristiche del permeato ottenuto da un solo stadio nelle stesse condizioni. Poiché le due correnti di permeato avranno composizioni differenti, una configurazione di questo tipo può essere vantaggiosa nel momento in cui le due correnti possono essere in qualche modo utilizzate separatamente. Nella separazione di una miscela CO₂-CH₄, ad esempio, il permeato con una maggiore concentrazione di metano potrebbe essere utilizzato come combustibile a basso contenuto energetico per le caldaie. Un secondo vantaggio può essere di tipo energetico, scegliendo di operare in condizioni meno spinte nel primo stadio, in cui una buona parte dell'impurità viene rimossa, per poi nel secondo stadio sottoporre a compressione un volume di gas ridotto. Le equazioni di bilancio che descrivono il sistema rimangono la (3.3) e la (3.4). Si possono scrivere le equazioni di bilancio totale e parziale per ciascuno stadio:

$$Q_f = Q_{r1} + Q_{p1} \quad (4.1)$$

$$Q_f x_f = Q_{r1} x_{r1} + Q_{p1} y_{p1} \quad (4.2)$$

per il primo stadio e

$$Q_{r1} = Q_r + Q_{p2} \quad (4.3)$$

$$Q_{r1} x_{r1} = Q_r x_r + Q_{p2} y_{p2} \quad (4.4)$$

per il secondo stadio, ricordando che la portata e composizione del permeato totale si ottengono come:

$$Q_p = Q_{p1} + Q_{p2} \quad (4.5)$$

$$Q_p y_p = Q_{p1} y_{p1} + Q_{p2} y_{p2} \quad (4.6)$$

Ciascuno stadio sarà caratterizzato da un valore di purezza e di recupero (per il permeato e per il retentato), ma ciò che è noto del sistema è solo il recupero o la purezza del prodotto finale, in quanto variabile di processo fissata.

$$R_{p1} = \frac{Q_{p1} y_{p1}}{Q_f x_f} \quad (4.7)$$

$$R_{p2} = \frac{Q_{p2} y_{p2}}{Q_{r1} x_{r1}} = \frac{Q_f x_f R_p - Q_{p1} y_{p1}}{Q_{r1} x_{r1}} \quad (4.8)$$

$$R_{r1} = \frac{Q_{r1} x_{r1}}{Q_f x_f} \quad (4.9)$$

$$R_{r2} = \frac{Q_r x_r}{Q_{r1} x_{r1}} = \frac{Q_f x_f R_r}{Q_{r1} x_{r1}} \quad (4.10)$$

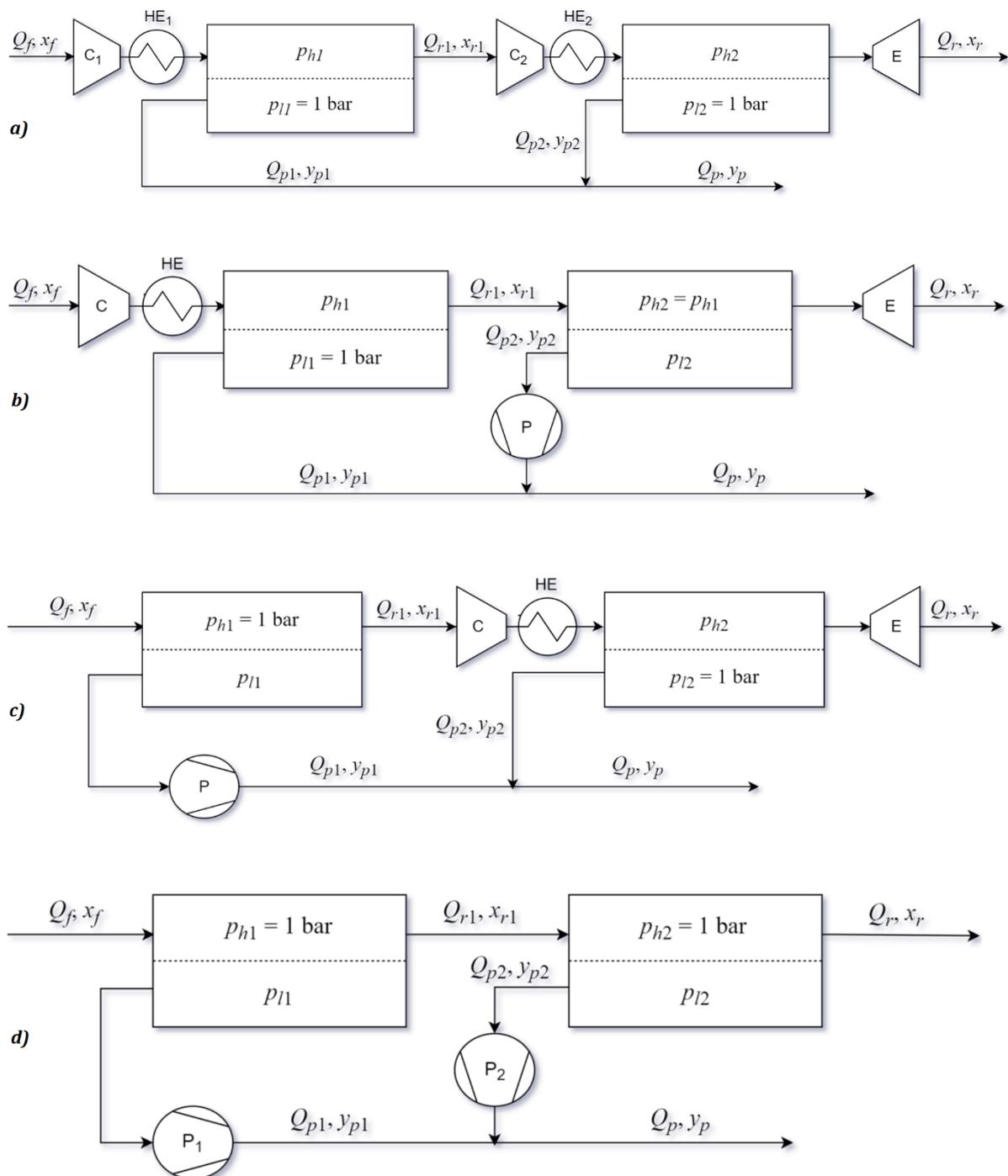


Figura 4.1 Schemi considerati per un sistema di separazione a membrana con due stadi in serie

4.1.2 Due stadi in cascata

Uno schema con due stadi di separazione con membrana si definisce “in cascata” quando al secondo stadio viene alimentato il permeato del primo stadio. Questo sistema permette di migliorare la purezza del permeato e di aumentare il recupero del retentato. I quattro schemi di separazione in cascata, che si differenziano per l’utilizzo di compressori e/o pompe a vuoto, sono rappresentati in FIGURA 4.2. Il bilancio totale (3.3) e quello parziale (3.4) al sistema rimangono validi. Si possono scrivere il bilancio totale e il bilancio a un componente per ciascuno dei due stadi.

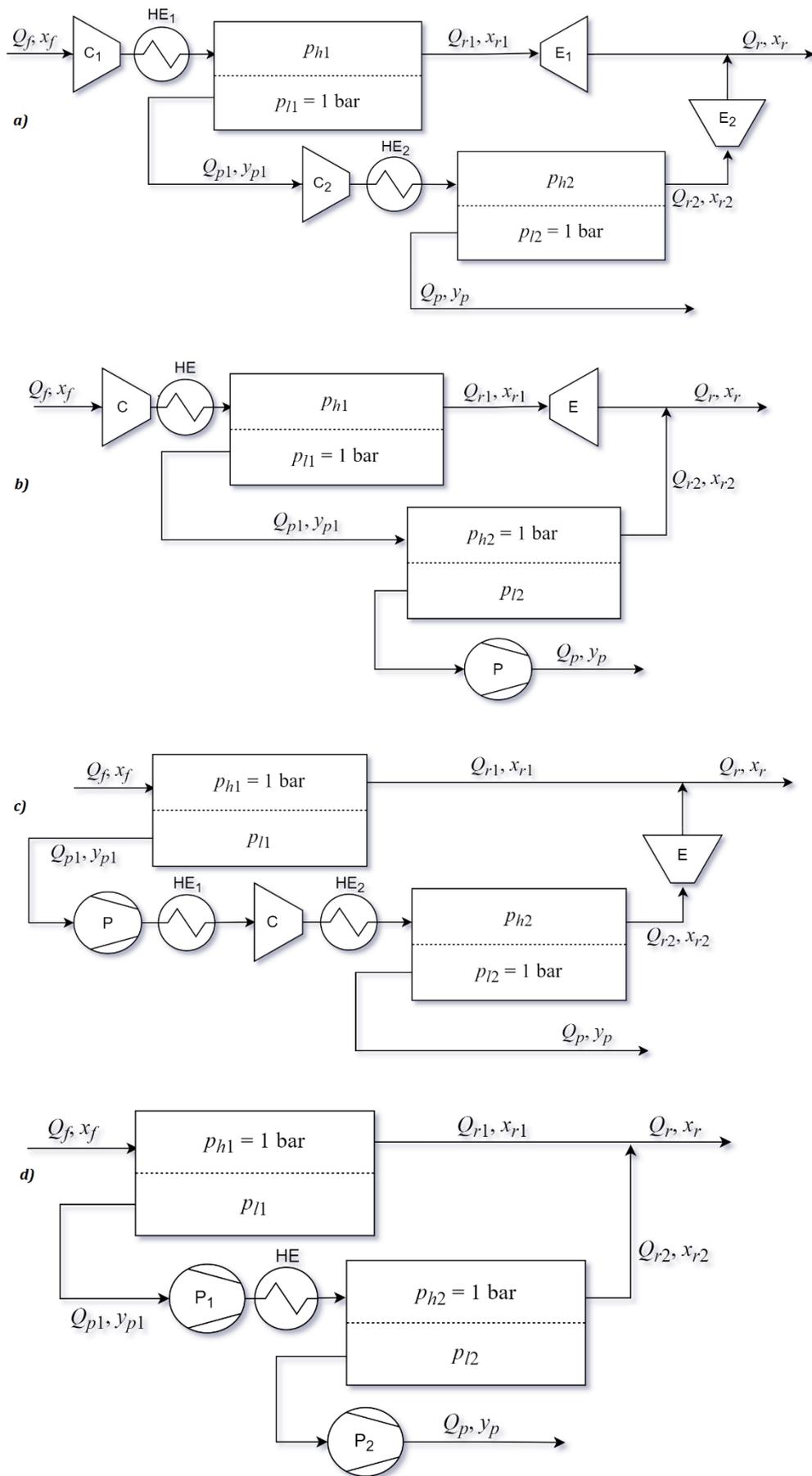


Figura 4.2 Schemi considerati per un sistema di separazione a membrana con due stadi in cascata

$$Q_f = Q_{r1} + Q_{p1} \quad (4.11)$$

$$Q_f x_f = Q_{r1} x_{r1} + Q_{p1} y_{p1} \quad (4.12)$$

per il primo stadio e

$$Q_{p1} = Q_{r2} + Q_p \quad (4.13)$$

$$Q_{p1} y_{p1} = Q_{r2} x_{r2} + Q_p y_p \quad (4.14)$$

per il secondo stadio. Il retentato prodotto dal sistema è ottenuto dalla miscelazione dei retentati di ciascuno stadio, per cui la sua portata e composizione sono date da:

$$Q_r = Q_{r1} + Q_{r2} \quad (4.15)$$

$$Q_r x_r = Q_{r1} x_{r1} + Q_{r2} x_{r2} \quad (4.16)$$

Ciascuno stadio è caratterizzato da un valore di purezza e di recupero, in riferimento al permeato e al retentato, ma del sistema è noto solo il recupero o la purezza del prodotto finale. Il recupero di ciascuno stadio può essere scritto come segue:

$$R_{p1} = \frac{Q_{p1} y_{p1}}{Q_f x_f} \quad (4.17)$$

$$R_{p2} = \frac{Q_p y_p}{Q_{p1} y_{p1}} = \frac{R_p}{R_{p1}} \quad (4.18)$$

se riferito al permeato e

$$R_{r1} = \frac{Q_{r1} x_{r1}}{Q_f x_f} \quad (4.19)$$

$$R_{r2} = \frac{Q_{r2} x_{r2}}{Q_{p1} y_{p1}} = \frac{Q_f x_f R_r - Q_{r1} x_{r1}}{Q_{p1} y_{p1}} \quad (4.20)$$

in riferimento al retentato.

4.1.3 Due stadi in serie con ricircolo

La maggior parte dei processi multistadio con membrane prevede un riciclo per migliorare la separazione e il recupero dei prodotti. Al sistema con due stadi di separazione in serie si può pensare di riciclare al primo stadio il permeato del secondo stadio, come illustrato in FIGURA 4.3. Ovviamente la portata ricircolata andrà portata alla pressione di alimentazione del primo stadio. Un sistema di questo tipo è utile se il prodotto è il gas permeato (dal primo stadio) e si desidera una purezza relativamente elevata: il ricircolo aiuta a migliorarne il recupero. Come sempre, valgono le equazioni di bilancio (3.3) e (3.4). L'alimentazione al primo stadio in questo caso non è nota, in quanto dipende dalle prestazioni dei due stadi di separazione. Si possono scrivere i bilanci di materia, totali e parziali, per ciascuno stadio, come segue:

$$Q_{f1} = Q_f + Q_{p2} = Q_{r1} + Q_p \quad (4.21)$$

$$Q_{f1} x_{f1} = Q_f x_f + Q_{p2} y_{p2} = Q_{r1} x_{r1} + Q_p y_p \quad (4.22)$$

per il primo stadio e

$$Q_{r1} = Q_r + Q_{p2} \quad (4.23)$$

$$Q_{r1} x_{r1} = Q_r x_r + Q_{p2} y_{p2} \quad (4.24)$$

per il secondo stadio.

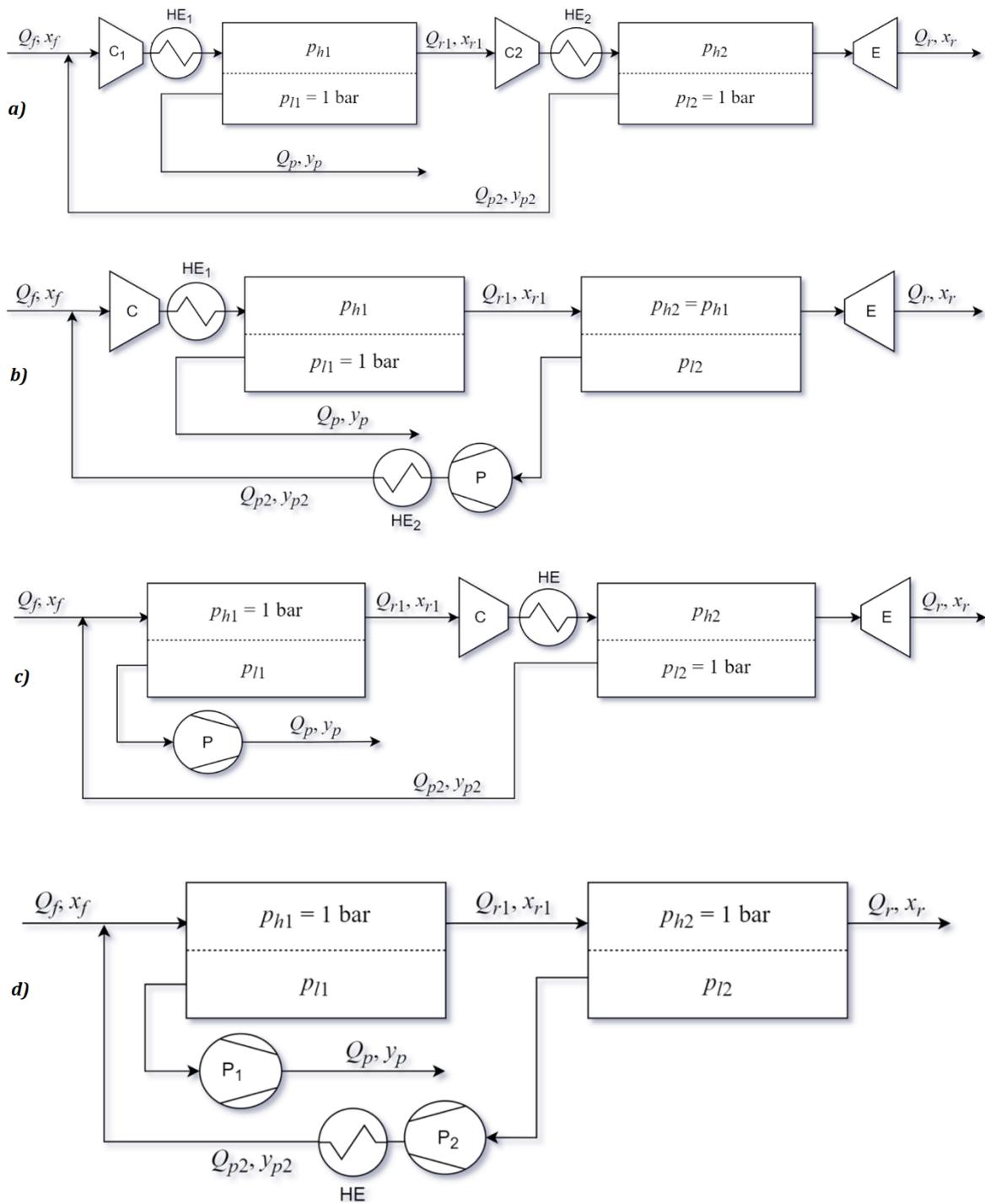


Figura 4.3 Schemi considerati per un sistema di separazione a membrana con due stadi in serie e ricircolo del permeato del secondo stadio

Si definiscono il recupero riferito al permeato e riferito al retentato per ciascuno stadio:

$$R_{p1} = \frac{Q_p y_p}{Q_{f1} x_{f1}} = \frac{R_p Q_f x_f}{Q_f x_f + Q_{p2} y_{p2}} \quad (4.25)$$

$$R_{p2} = \frac{Q_{p2} y_{p2}}{Q_{r1} x_{r1}} \quad (4.26)$$

$$R_{r1} = \frac{Q_{r1}x_{r1}}{Q_{f1}x_{f1}} \quad (4.27)$$

$$R_{r2} = \frac{Q_r x_r}{Q_{r1} x_{r1}} = \frac{R_r Q_f x_f}{Q_{r1} x_{r1}} \quad (4.28)$$

4.1.4 Due stadi in cascata con ricircolo

Lo schema con due stadi di separazione in cascata può essere migliorato, in termini di *performance*, prevedendo il riciclo al primo stadio del retentato del secondo stadio. In FIGURA 4.4 sono rappresentate le quattro varianti per un sistema di questo tipo, a seconda dell'apparecchiatura che viene utilizzata per fornire il gradiente di pressione. Tale sistema migliora drasticamente il recupero del residuo, ma è vantaggioso anche se il prodotto è il permeato, in quanto ne può incrementare la purezza. Anche in questo caso l'introduzione del ricircolo rende ignota la portata e composizione dell'alimentazione al primo stadio. Oltre ai bilanci su tutto il sistema (3.3) e (3.4), si scrivono i bilanci totali e parziali al singolo stadio:

$$Q_{f1} = Q_f + Q_{r2} = Q_r + Q_{p1} \quad (4.29)$$

$$Q_{f1}x_{f1} = Q_f x_f + Q_{r2}x_{r2} = Q_r x_r + Q_{p1}y_{p1} \quad (4.30)$$

per il primo stadio e

$$Q_{p1} = Q_{r2} + Q_p \quad (4.31)$$

$$Q_{p1}y_{p1} = Q_{r2}x_{r2} + Q_p y_p \quad (4.32)$$

per il secondo stadio.

Del sistema viene data come specifica la purezza o il recupero del prodotto finale, mentre il recupero per ciascuno stadio può essere definito come segue:

$$R_{p1} = \frac{Q_{p1}y_{p1}}{Q_{f1}x_{f1}} \quad (4.33)$$

$$R_{p2} = \frac{Q_p y_p}{Q_{p1}y_{p1}} = \frac{R_p Q_f x_f}{Q_{p1}y_{p1}} \quad (4.34)$$

riferito al permeato e

$$R_{r1} = \frac{Q_r x_r}{Q_{f1}x_{f1}} = \frac{R_r Q_f x_f}{Q_f x_f + Q_{r2}x_{r2}} \quad (4.35)$$

$$R_{r2} = \frac{Q_{r2}x_{r2}}{Q_{p1}y_{p1}} \quad (4.36)$$

riferito al retentato.

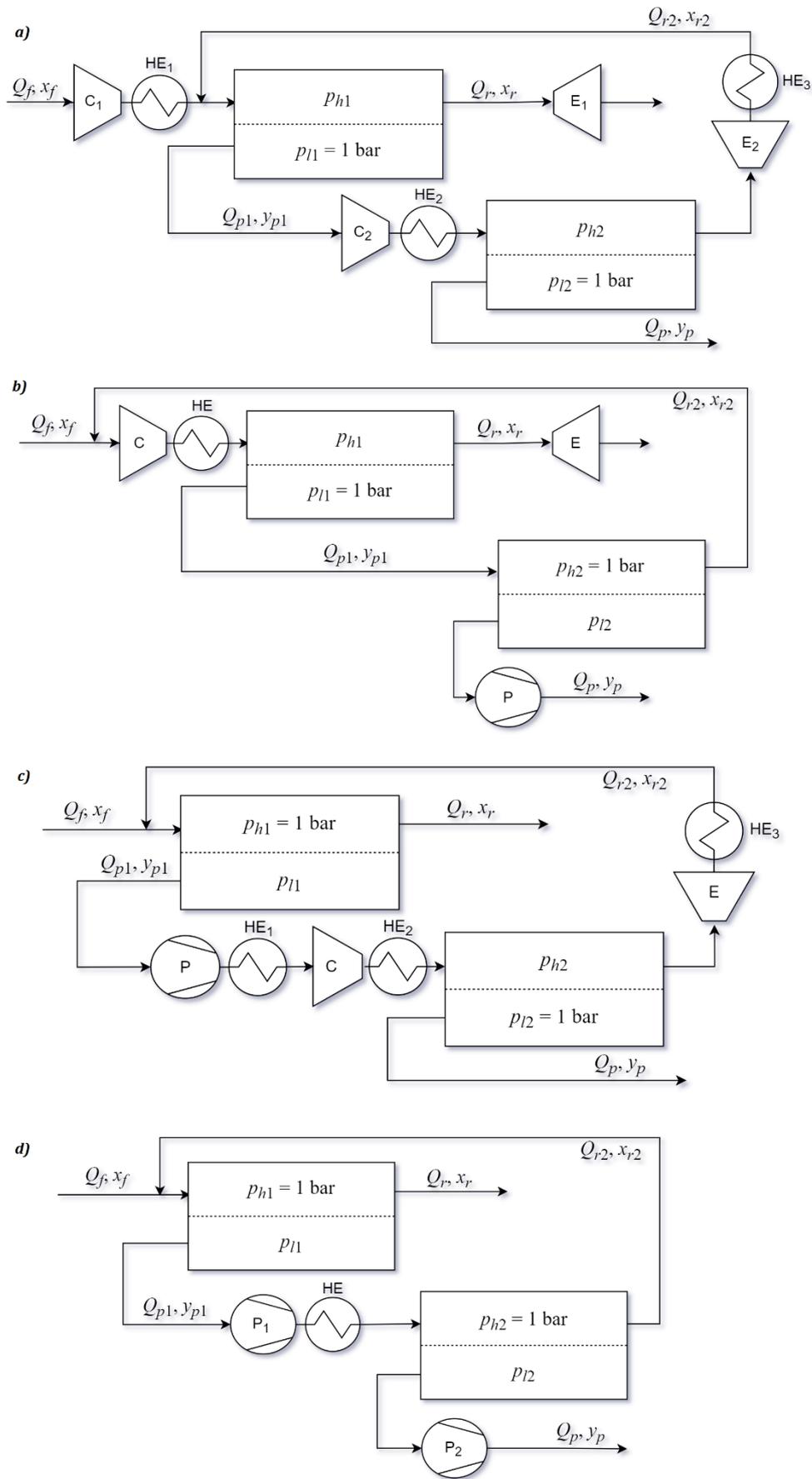


Figura 4.4 Schemi considerati per un sistema di separazione a membrana con due stadi in cascata e ricircolo del retentato del secondo stadio

4.2 Obiettivi di progettazione

Un sistema di separazione a membrana monostadio è sicuramente l'opzione con il più basso investimento di capitale. L'aggiunta di altri stadi implica un aumento dei costi sia per le apparecchiature che per le utenze, tuttavia può spesso risultare più conveniente grazie alle migliori caratteristiche di prodotto. Una valutazione economica deve tener conto di tre elementi:

- a) il costo dell'impianto a membrana, e cioè gli elementi della membrana stessa e gli alloggiamenti in pressione;
- b) i costi capitali e operativi del sistema di compressione;
- c) le eventuali perdite di prodotto, ad esempio il metano non recuperato nel permeato.

Se si fissano variabili di processo come la purezza o il recupero del prodotto, le condizioni operative di pressione e la separazione realizzata da ciascuno stadio possono essere regolate in maniera da bilanciare i costi e quindi ottimizzare il processo in termini economici. Si osservi che una pressione di alimentazione più elevata, o una pressione di permeato più bassa, migliorano la purezza del prodotto e riducono la superficie di membrana richiesta. Allo stesso tempo, però, aumentano il costo del sistema di compressione, sia in termini di investimento che di costi operativi. La soluzione al problema non è semplice. Esistono degli algoritmi che permettono di valutare e individuare tra le diverse condizioni operative quelle che realizzano al meglio gli obiettivi proposti per il sistema. Nel nostro caso, ciò che ci interesserà sarà andare a minimizzare sia la superficie di membrana da dover installare che i consumi energetici, in termini di energia elettrica spesa per pompe e compressori e di combustibile perduto. L'estensione della membrana può essere considerata un indicatore dei costi capitali dell'impianto, mentre i consumi energetici rappresenteranno i costi operativi. Si vuole quindi realizzare un'ottimizzazione a due obiettivi. Si vedrà che non esiste un'unica soluzione che vada a minimizzare allo stesso tempo sia l'area che i consumi, perché le condizioni che richiedono una superficie di membrana minore sono energeticamente più dispendiose, e viceversa. Si otterrà piuttosto una curva di punti, ognuno dei quali rappresenta una condizione di minimo del sistema.

Nel calcolo dei consumi del sistema di compressione si considererà la potenza richiesta da compressori (4.37) e/o pompe a vuoto (4.38), ma anche la potenza guadagnata da eventuali espansioni del retentato ad alta pressione (4.39) [24, 25].

$$E_c = \frac{Q_c}{V_m \eta_c} \frac{\gamma R T_{in}}{\gamma - 1} \left[v^{(\gamma-1)/\gamma} - 1 \right] \quad (4.37)$$

$$E_{pv} = \frac{Q_{pv}}{V_m \eta_{pv}} \frac{\gamma R T_{in}}{\gamma - 1} \left[v^{(\gamma-1)/\gamma} - 1 \right] \quad (4.38)$$

$$E_{ex} = \eta_{ex} \frac{Q_{ex}}{V_m} \frac{\gamma R T_{in}}{\gamma - 1} \left[1 - v^{(\gamma-1)/\gamma} \right] \quad (4.39)$$

T_{in} è la temperatura in ingresso all'apparecchiatura, $V_m=22.414$ [L/mol] il volume molare di un gas in condizioni standard, v il rapporto di compressione (4.40), γ il fattore di espansione adiabatica, che dipende dalla composizione della miscela e viene calcolato con la (4.41). Per il calore specifico delle specie in gioco si utilizza un valore medio calcolato dalle espressioni (4.42) per la CO₂, (4.43) per l'N₂, (4.44) per il CH₄ [26].

$$v = \frac{p_{out}}{p_{in}} \quad (4.40)$$

$$\gamma = \frac{c_{p,mix}}{c_{v,mix}} = \frac{c_{p,A} \cdot x_A + c_{p,B} \cdot (1-x_A)}{(c_{p,A}-R)x_A + (c_{p,B}-R)(1-x_A)} \quad (4.41)$$

$$\frac{c_{p,CO_2}}{R} = 5.457 + 1.045 \cdot 10^{-3}T - 1.157 \cdot 10^5 T^{-2} \quad (4.42)$$

$$\frac{c_{p,N_2}}{R} = 3.28 + 0.593 \cdot 10^{-3}T + 0.04 \cdot 10^5 T^{-2} \quad (4.43)$$

$$\frac{c_{p,CH_4}}{R} = 1.702 + 9.081 \cdot 10^{-3}T - 2.164 \cdot 10^{-6}T^2 \quad (4.44)$$

Le efficienze η di compressori, pompe a vuoto ed espansori si assumono dell'85% [24, 25].

Se la miscela da trattare è CH_4-CO_2 e quindi il prodotto desiderato è il metano, si include nel consumo energetico la perdita di metano nel permeato in termini di potenza chimica perduta (4.45).

$$E_{ch,CH_4} = LHV \cdot \rho \cdot Q_p y_{p,CH_4} \quad (4.45)$$

$LHV=50$ [MJ/kg] è il potere calorifico inferiore del metano [27], $\rho=0.7168$ [kg/m³] la densità del metano in condizioni standard [28]. Il consumo energetico totale è dato dalla (4.46).

$$E_{tot} = E_c + E_{pv} - E_{ex} + E_{ch} \quad (4.46)$$

La potenza utilizzata dagli scambiatori di calore si calcola invece dall'espressione (4.47).

$$E_{he} = \frac{Q_{he} c_{p,mix} \Delta T}{V_m \eta_{he}} = \frac{Q_{he} c_{p,mix} T}{V_m \eta_{he}} \left[\nu^{(\gamma-1)/\gamma} - 1 \right] \quad (4.47)$$

Si assume l'efficienza del processo di riscaldamento/raffreddamento η_{he} dell'85% [25].

I risultati del problema di ottimizzazione si confronteranno in realtà in termini di area specifica a_m e consumo specifico e_{tot} , e cioè l'area totale della membrana e il consumo energetico totale vengono rapportati alla quantità di prodotto ottenuta dal processo. Se il prodotto è il metano nel retentato, si utilizzeranno le espressioni (4.48) e (4.49), al contrario se si è interessati alla CO_2 separata gli obiettivi del problema saranno dati dalle espressioni (4.50) e (4.51).

$$a_m = \frac{A_{m,1} + A_{m,2}}{Q_r x_{r,CH_4}} \frac{V_m}{M_{w,CH_4}} \quad (4.48)$$

$$e_{tot} = \frac{E_{tot}}{Q_r x_{r,CH_4}} \frac{V_m}{M_{w,CH_4}} \quad (4.49)$$

$$a_m = \frac{A_{m,1} + A_{m,2}}{Q_p y_{p,CO_2}} \frac{V_m}{M_{w,CO_2}} \quad (4.50)$$

$$e_{tot} = \frac{E_{tot}}{Q_p y_{p,CO_2}} \frac{V_m}{M_{w,CO_2}} \quad (4.51)$$

con $M_{w,CH_4}=16.04$ [g/mol] e $M_{w,CO_2}=44.01$ [g/mol].

5. Ottimizzazione a due obiettivi del sistema multistadio

Se da un lato i sistemi di separazione multistadio permettono di migliorare notevolmente le prestazioni del processo, dall'altro la progettazione ottimale del sistema, sia in termini di configurazione che di condizioni operative, richiede la soluzione di un problema non banale. Diversi studi in letteratura trattano per lo più l'analisi parametrica o l'ottimizzazione che hanno come singolo obiettivo i costi o il consumo energetico. Tuttavia bisognerebbe innanzitutto tener conto di due criteri, che hanno implicazioni dirette su costi capitali e operativi: l'ingombro del sistema (e quindi la superficie della membrana) e i consumi energetici, come anticipato nel capitolo 4. Diventa allora importante ai fini della progettazione e gestione del sistema eseguire un'ottimizzazione multiobiettivo che mostri il trade-off tra questi due elementi. Esistono diversi algoritmi di ottimizzazione che sono stati testati in letteratura per alcune configurazioni multistadio, volendo individuare le condizioni operative che permettono di raggiungere le specifiche nella maniera più ottimale [25, 29, 30, 31].

Un problema di ottimizzazione multiobiettivo si differenzia per molti motivi da quello a un solo obiettivo, a partire dal tipo di soluzione che viene generata. Innanzitutto, un problema di ottimizzazione multiobiettivo (MOOP) può essere matematicamente espresso come segue:

$$\left. \begin{array}{lll} \text{minimizza/massimizza} & f_m(\mathbf{x}) & m = 1, 2, \dots, M; \\ \text{soggetta a} & g_j(\mathbf{x}) \geq 0 & j = 1, 2, \dots, J; \\ & h_k(\mathbf{x}) = 0 & k = 1, 2, \dots, K; \\ & x_i^{(L)} \leq x_i \leq x_i^{(U)} & i = 1, 2, \dots, n. \end{array} \right\} \quad (5.1)$$

tenendo conto dei vincoli che qualsiasi soluzione fattibile, compresa quella ottimale, deve rispettare. \mathbf{x} è il vettore delle n variabili decisionali, ciascuna delle quali è vincolata ad assumere un valore compreso tra un limite inferiore $x_i^{(L)}$ e un limite superiore $x_i^{(U)}$. Questi vincoli definiscono lo spazio delle variabili decisionali \mathcal{D} . $g_j(\mathbf{x})$ e $h_k(\mathbf{x})$ prendono il nome di funzioni di vincolo, e al problema vengono associati j vincoli di disuguaglianza e k vincoli di uguaglianza. Una soluzione \mathbf{x} che non rispetta tutti i vincoli del problema si definisce soluzione infattibile. Non è detto che l'intero spazio delle variabili decisionali \mathcal{D} sia fattibile, per cui si definisce la regione di fattibilità \mathcal{S} come l'insieme di tutte le soluzioni fattibili, quindi che rispettano tutti i vincoli del problema. Si hanno poi M funzioni obiettivo $f_m(\mathbf{x})$. Una prima differenza con l'ottimizzazione a un solo obiettivo si ha nell'esistenza di un ulteriore spazio multidimensionale, quello delle funzioni obiettivo, denominato spazio degli obiettivi \mathcal{Z} . Ad ogni punto nello spazio \mathcal{S} corrisponde un punto nello spazio \mathcal{Z} . La soluzione al problema non è un solo valore di minimo. In genere, le funzioni obiettivo sono in conflitto tra loro e quindi lo scopo di un'ottimizzazione multiobiettivo diventa quello di trovare le soluzioni Pareto-ottimali. Una soluzione \mathbf{x}^* nello spazio di fattibilità \mathcal{S} si definisce ottimale di Pareto se non esiste un altro \mathbf{x} che vada a ridurre almeno una funzione obiettivo senza aumentarne un'altra. Gli ottimali di Pareto sono soluzioni non dominate. Senza ulteriori informazioni, tutte le soluzioni ottimali di Pareto sono considerate ugualmente buone. Si può notare che l'insieme ottimale di Pareto \mathcal{Z}_P si trova sempre al confine dello spazio degli obiettivi \mathcal{Z} e prende il nome di fronte di Pareto. Se le funzioni obiettivo sono solo due, gli estremi del fronte di Pareto coincidono con il minimo di ciascuna funzione obiettivo, assumendo che i minimi siano unici.

La maggior parte degli algoritmi classici utilizza una procedura deterministica punto-per-punto per arrivare alla soluzione ottimale. Partono da una prima soluzione ipotizzata, di tentativo, e sulla base di considerazioni locali l'algoritmo suggerisce la direzione in cui procedere la ricerca,

trovando una soluzione intermedia. La procedura si ripete per un certo numero di volte. Questi algoritmi si differenziano per la regola di transizione utilizzata, che definisce la direzione di ricerca a ogni soluzione intermedia. Si possono classificare in due gruppi:

- i metodi diretti, in cui la ricerca è definita solo dalle funzioni obiettivo e dai vincoli; sono solitamente lenti e richiedono molte valutazioni della funzione prima di raggiungere la convergenza;
- i metodi basati sul gradiente che utilizzano anche le derivate di primo e/o secondo grado delle funzioni obiettivo e/o dei vincoli; rispetto ai metodi diretti convergono più rapidamente, ma non sono efficienti nei problemi non differenziabili o discontinui.

Col tempo sono stati introdotti degli algoritmi di ricerca non classici e stocastici. Tra questi vi sono gli algoritmi evolutivi (EA) che raggiungono la soluzione ottimale secondo una procedura che imita i principi evolutivi della natura. Ad ogni iterazione viene utilizzata una popolazione di soluzioni invece che una sola soluzione, e il risultato del problema sarà anch'esso una popolazione di soluzioni [32].

In questo lavoro si è deciso di utilizzare la funzione già integrata nel *Global Optimization Toolbox* di MATLAB® `gamultiobj`, che implementa un algoritmo genetico controllato ed elitario (una variante dell'NSGA-II) per un'ottimizzazione multiobiettivo.

5.1 NSGA-II (*Elitist Non-Dominated Sorting Genetic Algorithm*)

Gli algoritmi genetici sono tra quei metodi basati sugli algoritmi evolutivi che portano a una generazione diretta del fronte di Pareto, andando ad ottimizzare simultaneamente i singoli obiettivi. Essendo degli algoritmi evolutivi, il metodo di risoluzione si basa sui principi di evoluzione biologica. A partire da una popolazione iniziale, gli individui migliori vengono selezionati e una nuova generazione di individui si forma grazie alle operazioni di mutazione e *crossover*, fino al raggiungimento della popolazione finale a cui corrispondono le soluzioni ottimali. Gli algoritmi genetici hanno il vantaggio di valutare soluzioni multiple in un'unica iterazione e offrono una buona flessibilità, anche per quei problemi in cui non sono disponibili troppe informazioni a priori.

5.1.1 Principi di base di un algoritmo genetico

In un algoritmo genetico ciascuna possibile soluzione (individuo) viene rappresentata tramite una stringa, in modo da avere una rappresentazione pseudo-cromosomica. La maniera più semplice per rappresentare il cromosoma è quella di utilizzare una stringa di bit. Ciascun cromosoma contiene le n variabili decisionali, rappresentate dai geni. All'inizio della procedura deve essere generata una popolazione, costituita da N_p individui. N_p è la dimensione della popolazione. La popolazione iniziale può essere generata utilizzando un generatore di numeri casuali, ma si può anche scegliere di fornire una propria soluzione iniziale di tentativo, se nota. A questo punto viene associato a ciascun individuo un valore di *fitness*, che comunichi la bontà della soluzione e permetta di confrontarli tra loro. Maggiore è il valore di *fitness*, migliore è la soluzione. Una funzione di *fitness* molto comune prevede di sottrarre al valore massimo ottenuto dalla popolazione corrente della funzione obiettivo, il corrispondente valore della funzione obiettivo dell' i -esimo individuo.

L'obiettivo di un algoritmo genetico diventa quello di andare a creare una nuova popolazione il cui valore medio di *fitness* sia maggiore rispetto a quello della popolazione precedente. Questo *step* prende il nome di riproduzione o selezione, e permette di eliminare le soluzioni più

deboli e utilizzare quelle migliori per formare la generazione successiva. Gli individui selezionati sono definiti genitori e la nuova popolazione viene creata utilizzando due metodi: il *crossover* e la mutazione. Con il *crossover* un nuovo individuo viene generato grazie alla combinazione di due genitori. Ad esempio, i due cromosomi possono essere scissi in corrispondenza di una certa posizione casuale, per poi andare a scambiare le parti di stringa separate. La percentuale di individui che viene sottoposta a *crossover* e i punti in cui l'operazione deve avvenire fanno parte della natura euristica dell'algoritmo genetico, spesso vengono utilizzate selezioni casuali. Lo stesso vale per l'operazione di mutazione, che rispetto al *crossover* coinvolge un solo cromosoma che subisce uno scambio di elementi all'interno della propria stringa. La mutazione fa sì che materiale genetico prezioso non venga perso con l'operazione di *crossover*. Non è detto che tutti gli individui della popolazione vadano sottoposti a uno dei due processi. Un numero di *crossover* molto elevato potrebbe portare ad allontanarsi eccessivamente dai valori della funzione di fitness desiderati, peggiorando le prestazioni dell'algoritmo. La mutazione, invece, permette comunque di rimanere nelle vicinanze della popolazione corrente.

La riproduzione continua, come appena descritto e indicato in FIGURA 5.1, fino a quando non vengono soddisfatti i criteri di arresto. Come criterio di arresto si può avere o che il miglioramento del valore di fitness è inferiore a una certa tolleranza ε_0 per le ultime n riproduzioni consecutive, oppure che il numero di iterazioni ha superato un valore massimo specificato [33].

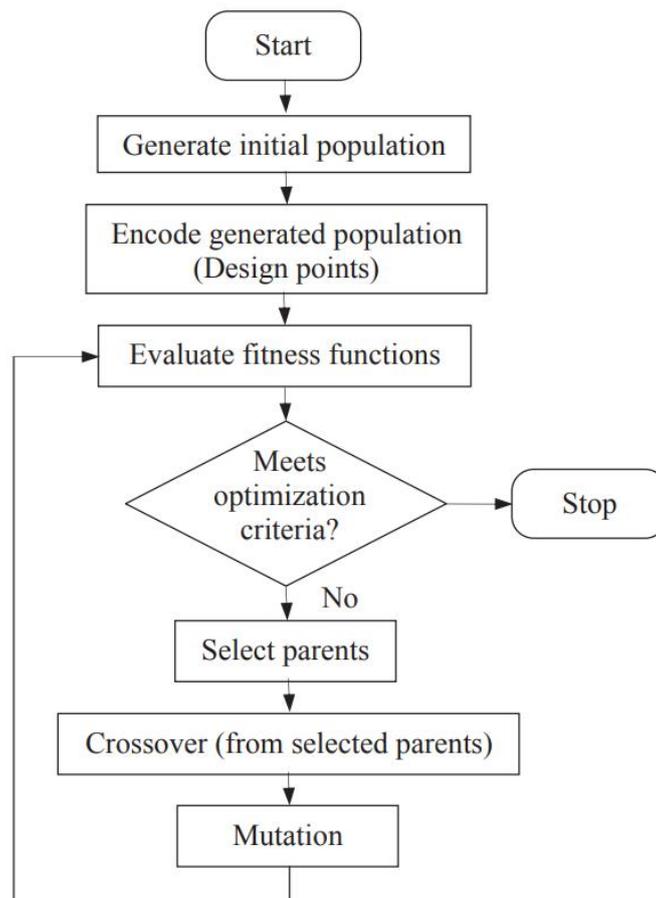


Figura 5.1 Procedura risolutiva tipica di un algoritmo genetico [33]

Un algoritmo genetico richiede un grande numero di valutazioni di funzione ad ogni iterazione, per cui in problemi particolarmente complessi potrebbe richiedere un tempo di calcolo significativo. La sfida di un algoritmo genetico in un'ottimizzazione multiobiettivo è trovare il modo migliore per direzionare la soluzione verso gli ottimali di Pareto, senza perdere la diversità della popolazione convergendo prematuramente. L'NSGA-II implementato in MATLAB® fa parte di quel gruppo di algoritmi genetici che seguono un approccio basato su Pareto.

5.1.2 Approcci per la ricerca del fronte di Pareto

Oltre alle tecniche di mutazione e *crossover*, andranno introdotti tre nuovi elementi che permettono di raggiungere gli ottimali di Pareto: il *ranking*, il meccanismo di *niching* e la strategia elitaria [34].

L'idea alla base dell'algoritmo è quella di individuare nella popolazione l'insieme di individui non dominati. Assegnare a ciascun individuo un rango, e quindi un valore di *fitness*, permette di ordinare le soluzioni in base alla loro idoneità e selezionare le soluzioni migliori. Il ranking, quindi, determina la probabilità di riproduzione di un individuo. L'NSGA utilizza una tecnica di classificazione a strati (FIGURA 5.2), in cui a tutti gli individui non dominati viene assegnato lo stesso rango, il più basso possibile. In generale, la tecnica prevede che gli individui di rango k siano dominati solo dagli individui di rango $k-1$ o inferiore. Più basso è il rango, maggiore è la probabilità che gli individui vengano selezionati. Alle soluzioni non fattibili viene assegnato un rango peggiore di qualsiasi altro individuo fattibile.

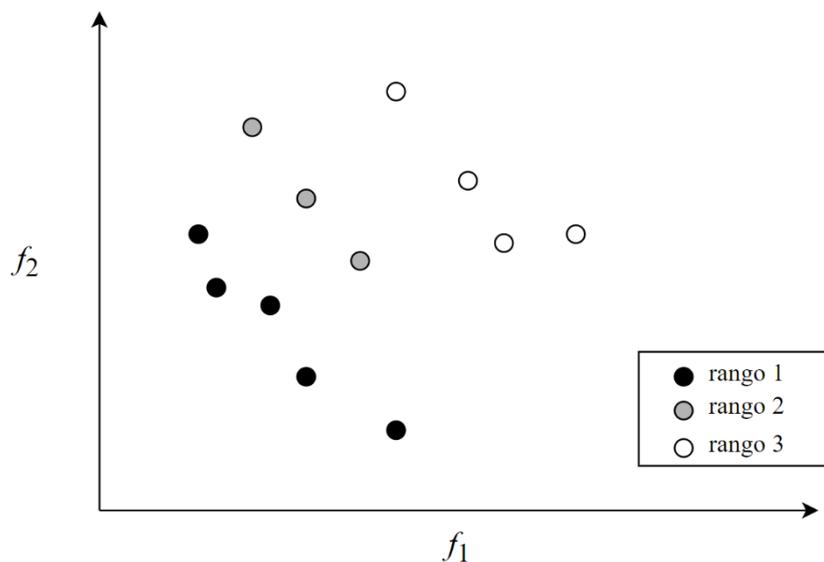


Figura 5.2 Esempio di *ranking* per un algoritmo genetico di smistamento non dominato (NSGA)

Gli algoritmi genetici hanno la tendenza a convergere intorno a una regione limitata di ottimali di Pareto, fenomeno noto come deriva genetica. Si definisce una nicchia un gruppo di punti vicini tra loro. Per limitare lo sviluppo di una singola nicchia e spingere invece la ricerca dell'algoritmo verso un maggior numero di nicchie si sono sviluppate tecniche apposite. I metodi che limitano la convergenza dell'algoritmo verso una regione limitata del fronte di Pareto prendono il nome di meccanismi o tecniche di *niching*. Una strategia comune è quella di sfavorire il valore di *fitness* dei punti appartenenti a una nicchia, andando a dividerlo per una

costante proporzionale al numero dei punti che si trovano al di sotto di una certa distanza specificata. In questo modo la probabilità che un punto di nicchia venga selezionato per la nuova generazione diminuisce. L'NSGA-II utilizza un metodo di *niching* differente, che coinvolge due elementi: una stima della densità e un operatore per il confronto dell'affollamento. Viene dapprima stimata la densità di individui che circondano una particolare soluzione. Questa stima si ottiene calcolando la distanza media tra due punti sui due lati della soluzione considerata, lungo ciascuno dei due obiettivi del problema. La quantità ottenuta prende il nome di distanza di affollamento. Nell'esempio rappresentato in FIGURA 5.3, la distanza di affollamento per la soluzione i -esima è il valore medio del lato del rettangolo tratteggiato, che ha come vertici le due soluzioni più vicine. Per il calcolo della distanza di affollamento è necessario innanzitutto ordinare gli individui in base al corrispondente valore della funzione obiettivo, in ordine crescente. Agli estremi di ciascuna funzione obiettivo viene assegnata una distanza di affollamento infinita. Dopodiché, per ciascun valore intermedio si calcola la differenza assoluta, normalizzata, tra due valori adiacenti della funzione. Si segue lo stesso metodo per ciascuna funzione obiettivo. La distanza di affollamento complessiva per un individuo viene infine calcolata come la somma delle distanze corrispondenti a ciascuna funzione obiettivo. A questo punto ogni individuo della popolazione è caratterizzato da un valore di rango e una distanza di affollamento. Durante il processo di selezione vengono preferite le soluzioni con in rango più basso. L'operatore per il confronto dell'affollamento fa sì che tra due soluzioni con uguale rango, venga preferita quella che si trova in una regione meno affollata.

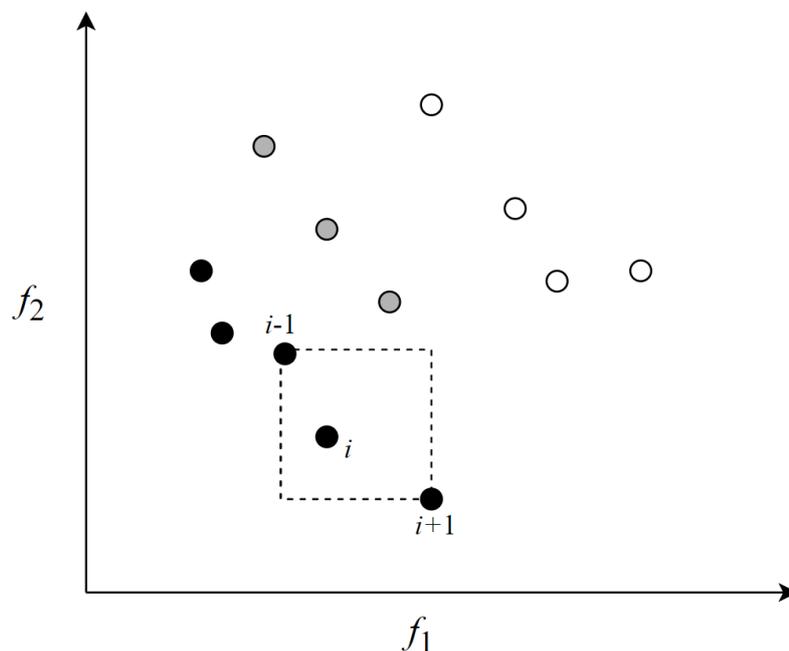


Figura 5.3 Calcolo della distanza di affollamento per un l'individuo i -esimo

La strategia elitaria, d'altra parte, garantisce che le soluzioni ottimali di Pareto non vadano perdute. Funziona indipendentemente dallo schema di *ranking*. Viene definito un insieme di individui, definito dalla popolazione attuale, e un insieme provvisorio di individui non dominati, approssimazioni temporanee degli ottimali di Pareto. Ad ogni iterazione, gli individui non dominati della popolazione corrente vengono aggiunti all'insieme provvisorio, mentre gli individui dell'insieme provvisorio diventati dominati vengono scartati. Ad ogni step di riproduzione, successivamente alle operazioni di mutazione e *crossover*, un certo numero,

specificato dall'utente, di individui appartenenti all'insieme provvisorio entra a far parte della popolazione attuale. Questi individui prendono il nome di punti d'élite.

5.1.3 Smistamento non dominato

Si è compreso come la determinazione degli individui non dominati in ciascuna popolazione sia uno *step* essenziale per direzionare il calcolo verso gli ottimali di Pareto. La procedura di confronto tra gli individui è anche quella che richiede un grande sforzo da parte dell'algoritmo. Si consideri infatti che ogni soluzione richiede $q(N - 1)$ confronti, e il primo ciclo di ordinamento prevede già $\sum_{i=1}^{N-1} q(N - i) = \frac{1}{2}qN(N - 1)$ confronti (in cui q è il numero di funzioni obiettivo e N la dimensione della popolazione). Di solito si sceglie di raccogliere i risultati di questa operazione all'interno di una matrice $N \times N$. Ogni termine della matrice rappresenta il risultato del confronto tra due individui della popolazione. Per capirne la struttura, si considerino due soluzioni della popolazione corrente, x e y . Se entrambe sono fattibili, e x domina y , allora il corrispondente elemento (x,y) della matrice sarà uguale a 1, viceversa verrà assegnato il valore -1. Allo stesso modo, se x è una soluzione fattibile e y è una soluzione infattibile, nella posizione (x,y) la matrice avrà valore 1, e viceversa avrà valore -1. Se sia x che y sono infattibili, si confronta il grado di violazione dei vincoli e i valori 1 o -1 verranno assegnati come sopra. Per formare le classi di individui, si fa uso di due criteri: il conteggio del dominio n_p e l'insieme D_p di individui dominati dalla soluzione. Il conteggio del dominio n_p si ottiene sommando il numero di -1 nella colonna p -esima della matrice. Consiste quindi nel contare quanti individui della popolazione dominano la soluzione p -esima. L'insieme D_p è formato dagli individui a cui si è associato il valore -1 nella riga p -esima della matrice. In sostanza, dell'insieme fanno parte gli individui della popolazione dominati dalla soluzione p -esima. Il primo fronte ND_1 degli individui non dominati è costituito dalle soluzioni che hanno n_p nullo. Dopodiché, per ciascun individuo appartenente all'insieme ND_1 , si prendono i corrispondenti individui dominati (appartenenti al suo insieme D_p) e si riduce di 1 il loro conteggio n_p . Le nuove soluzioni che hanno raggiunto n_p nullo (o in alternativa l'intero più basso) andranno a costituire il secondo fronte non-dominato ND_2 . La stessa procedura viene ripetuta fino a quando tutti i fronti non vengono identificati [34]. Come già spiegato, a ciascun fronte verrà assegnato un rango che ne definirà la bontà delle soluzioni.

5.1.4 Procedura complessiva dell'NSGA-II

Come un generico algoritmo genetico, anche per l'NSGA-II si crea una prima popolazione iniziale casuale P_0 di dimensioni N . Per ogni individuo della popolazione viene valutato il valore delle funzioni obiettivo e verificato il rispetto dei vincoli. Questo permetterà di eseguire lo smistamento non dominato e il calcolo della distanza di affollamento. A questo punto si procede con la creazione di una nuova popolazione Q_0 di dimensioni N , utilizzando le tecniche di selezione binaria a torneo (che utilizza oltre al rango anche l'operatore per il confronto dell'affollamento), *crossover* e mutazione. Essendo un algoritmo elitario, dopo la prima iterazione la procedura cambia leggermente. Per tenere in considerazione le migliori soluzioni non dominate trovate in precedenza, nell' i -esima iterazione viene formata una popolazione combinata $R_i = P_i \cup Q_i$, di dimensioni $2N$. gli individui vengono ordinati secondo lo smistamento non dominato e si calcola la distanza di affollamento. A questo punto si vuole formare la popolazione P_{i+1} con i membri migliori. Le migliori soluzioni sono quelle appartenenti all'insieme non dominato ND_1 . Se quest'insieme è costituito da un numero di individui inferiore a N , vengono selezionati tutti i membri dell'insieme ND_1 e i restanti membri vengono scelti dai fronti successivi, nell'ordine ND_2 , ND_3 e così via. Per completare la

popolazione P_{i+1} con N individui esatti, le soluzioni dell'ultimo fronte ND_i che viene utilizzato saranno scelte con il criterio dell'affollamento. A questo punto la popolazione genitoriale P_{i+1} viene utilizzata per la generazione della nuova popolazione Q_{i+1} con i metodi di selezione binaria a torneo, *crossover* e mutazione. La procedura descritta è schematizzata in FIGURA 5.4.

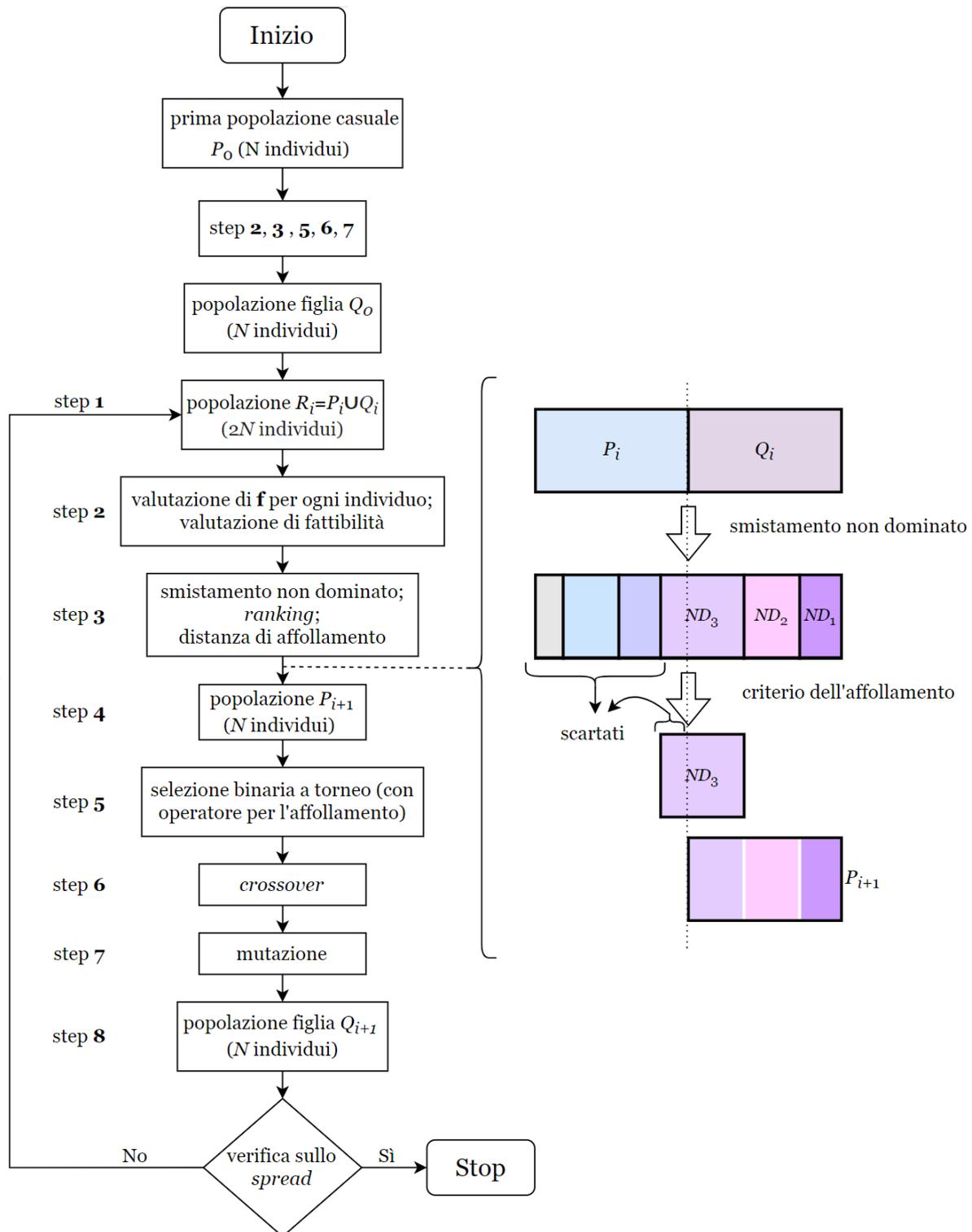


Figura 5.4 Schema della procedura risolutiva dell'NSGA-II

Come condizione di arresto viene introdotto il concetto di *spread*. Lo *spread* è una misura del movimento dell'insieme di Pareto e viene calcolato secondo la (5.2).

$$spread = \frac{\xi - \sigma}{\xi - F_1 d} \quad (5.2)$$

Con F_1 si indica il numero di individui nel primo fronte (approssimazione del fronte di Pareto), e d è la distanza media tra questi punti. σ è la deviazione standard delle loro distanze di affollamento (per i punti che hanno distanza finita). L'algoritmo poi valuta, per ogni funzione obiettivo, la norma della differenza tra il minimo del fronte di Pareto per la popolazione corrente e lo stesso minimo per l'iterazione precedente. La somma di queste norme per ogni funzione obiettivo fornisce ξ . Lo *spread* è piccolo quando ξ e σ sono piccoli, e cioè quando tra un'iterazione e l'altra gli estremi del fronte di Pareto non variano granché quando i punti sul fronte di Pareto sono distribuiti in maniera uniforme. L'algoritmo si arresta quando la media geometrica della variazione relativa dello *spread* nelle ultime n iterazioni (definite dall'utente) soddisfa la tolleranza ε_0 , e inoltre lo *spread* finale è minore dello *spread* medio nelle ultime n iterazioni [35].

Il meccanismo elitario dell'NSGA-II evita che delle soluzioni ottimali vengano rimosse. In più, il criterio dell'affollamento favorisce la diversità tra le soluzioni non dominanti, limitando l'avanzamento dell'algoritmo verso una regione limitata del fronte di Pareto. Da un certo punto di vista, però, l'operatore di confronto dell'affollamento potrebbe sfavorire la convergenza dell'algoritmo. Nelle ultime iterazioni, quando il primo fronte di soluzioni non dominate ha più di N membri, l'utilizzo della distanza di affollamento come criterio di selezione nel formare la popolazione P_{i+1} potrebbe preferire delle soluzioni non dominate ma non Pareto-ottimali a delle soluzioni compatte che invece erano degli ottimali di Pareto. L'algoritmo ricadrebbe in un ciclo di generazione di soluzioni Pareto-ottimali e soluzioni non Pareto-ottimali prima di raggiungere effettivamente un insieme ben distribuito di ottimali di Pareto [32].

5.1.5 Elitarismo controllato nell'NSGA-II

L'algoritmo implementato in MATLAB® con la funzione `gamultiobj` è in realtà una variante con elitarismo controllato dell'NSGA-II. La differenza sta nel metodo di selezione degli N individui per la popolazione P_{i+1} a partire dalla popolazione combinata $R_i = P_i \cup Q_i$. L'intenzione è quella di limitare il numero di individui selezionati dai fronti inferiori e mantenere una distribuzione predefinita di individui da tutti i fronti. Viene introdotto un parametro $r < 1$, definito dall'utente, che prende il nome di tasso di riduzione. La procedura è di tipo adattativo e segue una certa distribuzione, come quella geometrica qui definita. Si consideri che dalla popolazione R_i (di dimensione $2N$) sono stati individuati K fronti. Allora il numero massimo di individui che può essere selezionato dal fronte i -esimo è definito dalla (5.3).

$$N_i = N \frac{1-r}{1-r^K} r^{i-1} \quad (5.3)$$

Questo vuol dire che dal primo fronte si selezionerà il numero più alto di individui, che decrescerà poi esponenzialmente per i fronti successivi. Ovviamente è possibile che gli individui costituenti il fronte i -esimo non siano abbastanza da soddisfare il valore calcolato dalla (5.3). In questo caso, i posti rimanenti vengono aggiunti al numero massimo di individui ammessi dal fronte successivo, e si continua in questa maniera fino a quando tutti gli N posti della popolazione P_{i+1} non vengono riempiti. Nel caso in cui raggiunto l'ultimo fronte il numero di individui sia ancora inferiore a N , si ritorna al primo fronte. Questo avviene in particolare quando r è grande. Questa procedura con elitarismo controllato permette di ottenere una popolazione più diversificata, e questo può aiutare il raggiungimento progressivo del fronte di

Pareto ottimale nei problemi particolarmente complessi. Il valore ottimale di r dipende dal problema considerato [32]. La FIGURA 5.5 rappresenta la differenza tra un NSGA-II classico e l'NSGA-II con elitarismo controllato.

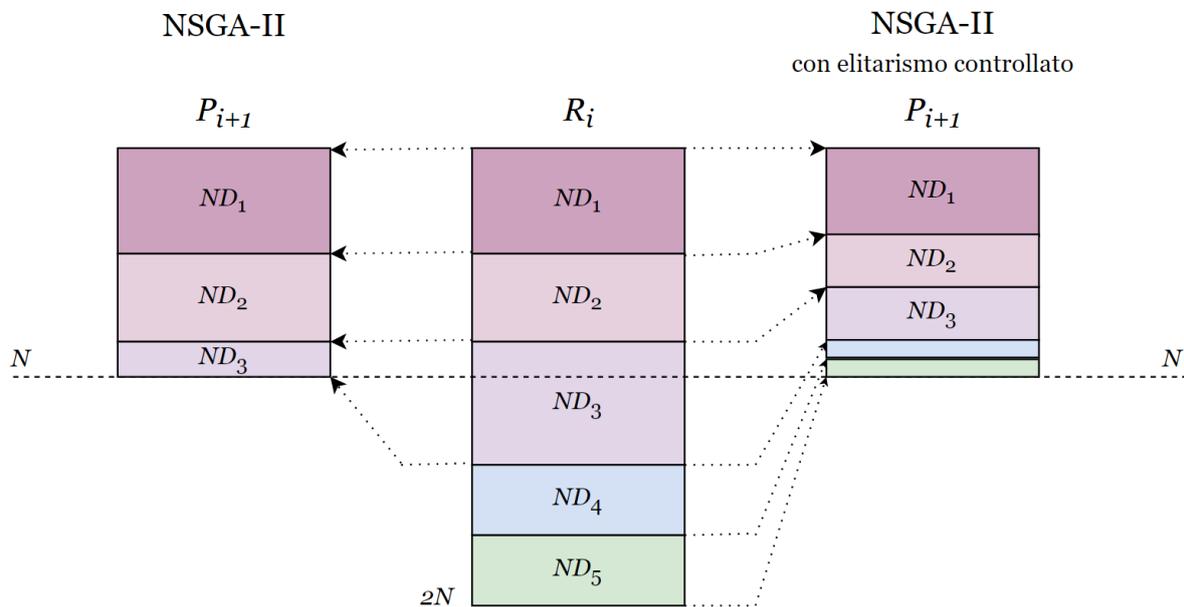


Figura 5.5 Rappresentazione della differenza tra la procedura elitaria dell'NSGA-II e la sua variante con procedura elitaria controllata

Nonostante tutti gli elementi utilizzati dall'NSGA-II per il raggiungimento del fronte di Pareto, in problemi particolarmente complessi la selezione casuale di una popolazione iniziale può portare a concentrare le soluzioni in una regione limitata del fronte di Pareto, non permettendone una completa esplorazione. In più, si è accennato a come la procedura di smistamento non ordinato sia particolarmente impegnativa e come il criterio dell'affollamento può portare l'algoritmo ad un ciclo di iterazioni in cui soluzioni non ottimali vengono preferite a soluzioni ottimali. Questo fa sì che il tempo di calcolo dell'algoritmo diventi notevole pur non ottenendo un risultato ottimale. Si è dimostrato come è possibile migliorare la velocità e l'accuratezza dell'algoritmo se per la popolazione iniziale l'utente fornisce già gli estremi del fronte [36]. Considerato il meccanismo elitario dell'algoritmo, queste soluzioni rimarranno con buona probabilità nell'insieme d'élite per tutto il tempo della ricerca. In questo modo il fronte di Pareto ottenuto dovrebbe essere sufficientemente esteso, riducendo al tempo stesso il tempo di calcolo. Si rende allora necessario andare ad effettuare innanzitutto un'ottimizzazione mono-obiettivo per ciascuna funzione obiettivo del sistema, per poi fornire la soluzione all'NSGA-II. In questo lavoro, si è fatto uso di diversi algoritmi presenti in MATLAB® in grado di eseguire un'ottimizzazione a obiettivo singolo. Si è preferito, laddove possibile, non utilizzare la funzione *ga* (che implementa l'algoritmo genetico mono-obiettivo) in quanto poco performante in questo tipo di problema.

5.2 Algoritmi di ottimizzazione a un solo obiettivo

5.2.1 PSO (*Particle Swarm Optimization*)

La funzione `particleswarm` nel *Global Optimization Toolbox* di MATLAB® si basa sull'algoritmo descritto da Kennedy ed Eberhart [37], con le modifiche apportate successivamente da Mezura-Montes e Coello Coello [38] e da Pedersen [39]. Nell'ottimizzazione *particle swarm* (PSO) le possibili soluzioni prendono il nome di particelle, e l'algoritmo funziona facendo muovere casualmente queste particelle nello spazio di ricerca, modificandone gradualmente la direzione sulla base dei miglioramenti scoperti da ciascuna delle altre particelle. La *performance* di un algoritmo PSO è strettamente dipendente dai parametri utilizzati nel metodo.

L'inizializzazione del metodo, quindi, consiste nel distribuire le particelle casualmente e uniformemente all'interno dello spazio di ricerca, entro i limiti stabiliti (se forniti al problema). La posizione della particella i sarà definita dal vettore $x(i)$, un vettore riga con un numero di elementi pari al numero di variabili decisionali del problema. Allo stesso modo, vengono assegnate casualmente e uniformemente le velocità iniziali v di ciascuna particella, all'interno dell'intervallo $[-r, r]$. r è un vettore, anch'esso con un numero di elementi pari a quello delle variabili decisionali, e la sua componente k -esima viene scelta come il minimo tra l'ampiezza dell'intervallo in cui la variabile è limitata, e l'ampiezza dello sciame iniziale (nella componente k -esima). Per ciascuna particella i viene valutata la funzione obiettivo e la posizione viene registrata nel vettore $p(i)$. Nelle iterazioni successive, in questo vettore sarà registrata la posizione in cui alla particella i corrispondeva il miglior valore della funzione obiettivo. b è invece, tra tutte le particelle, il miglior valore della funzione obiettivo trovato, a cui corrisponde la posizione d . Vengono poi inizializzati:

- la dimensione N del vicinato (massimo 2);
- l'inerzia W , come l'estremo superiore dell'intervallo di inerzia; di *default* è $[0.1, 1.1]$;
- il contatore di stallo $c=0$.

Verranno utilizzate due variabili per una regolazione ponderata della velocità: y_1 per la posizione migliore di ogni particella, y_2 per la posizione migliore del vicinato. Entrambi sono dei parametri che possono essere forniti al sistema, e di *default* sono impostati a 1.49.

Il procedimento seguito dall'algoritmo per aggiornare la posizione delle particelle e rivalutare la funzione obiettivo è descritto di seguito, per la generica iterazione n e la generica particella i :

1. Si sceglie un sottoinsieme casuale S di N particelle, oltre alla particella i .
2. Dell'insieme S , si valuta qual è il miglior valore della funzione obiettivo e si registra la posizione della particella corrispondente in g .
3. Si aggiorna la velocità v , in ogni sua componente k , secondo l'equazione (5.4).

$$v_{n,k} = W \cdot v_{n-1,k} + y_1 \cdot u_{1,k} \cdot (p_k - x_k) + y_2 \cdot u_{2,k} \cdot (g_k - x_k) \quad (5.4)$$

In cui u_1 e u_2 sono due vettori della stessa lunghezza di v , e le cui componenti hanno valori compresi tra 0 e 1 scelti in maniera casuale e uniforme. Ciò che si fa è quindi una somma pesata che considera la velocità dell'iterazione precedente, la differenza tra la posizione attuale e la migliore avuta dalla particella, la differenza tra la posizione attuale della particella e quella attualmente migliore nel vicinato.

4. Si aggiorna allora la posizione della particella (5.5).

$$\mathbf{x}_n = \mathbf{x}_{n-1} + \mathbf{v}_n \quad (5.5)$$

Se una qualsiasi componente di \mathbf{x} supera i limiti imposti, la si pone uguale a quel limite; se per la stessa componente di velocità supera a sua volta il limite, la si pone uguale a zero.

5. Si valuta la funzione obiettivo corrispondente alla nuova posizione \mathbf{x} e se necessario si aggiorna \mathbf{p} .

Una volta valutata la funzione obiettivo per le nuove posizioni di ogni particella, si considera l'intero sciame:

6. Si trova il più piccolo valore della funzione obiettivo ottenuto nell' n -esima iterazione e se necessario si aggiornano \mathbf{b} e \mathbf{d} .
7. Se \mathbf{b} è stato aggiornato si eseguono le seguenti azioni:
 - Si reimposta il contatore di stallo c come il massimo tra 0 e $c-1$.
 - Si pone N pari al valore minimo settato.
 - Si aggiorna l'inerzia a $W = 2W$ se c è più piccolo di 2, oppure $W = W/2$ se c è maggiore di 5, assicurandosi che resti all'interno dei limiti.

Al contrario:

- Si aumenterà c di 1
- Si porrà N come il minimo tra la dimensione dello sciame ed N sommato al suo minimo possibile (parametro impostato).

Come criterio di arresto, oltre al numero massimo di iterazioni raggiunto, si valuta la variazione relativa di \mathbf{g} nelle ultime n iterazioni e l'algoritmo si ferma se questa è minore della tolleranza impostata ϵ_0 [40].

Il PSO si dimostra un algoritmo fortemente dipendente dai processi stocastici. Questa caratteristica permette di esplorare approfonditamente gli spazi tra le regioni che si sono rivelate relativamente buone, mentre la modifica delle velocità piuttosto che la loro sostituzione permette di testare zone inesplorate nel dominio di ricerca.

Il *Particle Swarm* si è rivelato un buon algoritmo per il problema studiato, raggiungendo il minimo della funzione obiettivo in tempi accettabili. Tuttavia la funzione `particleswarm` non supporta le funzioni di vincolo di uguaglianza e disuguaglianza. Per risolvere alcune delle configurazioni presentate nel capitolo 4, si è reso necessario optare per altri due algoritmi.

5.2.2 Algoritmo di ottimizzazione dei surrogati

La funzione `surrogateopt` di MATLAB® è un risolutore ideale per le funzioni obiettivo che richiedono lunghi tempi di calcolo. Supporta la ricerca del minimo globale di una funzione in più dimensioni, con variabili limitate e funzioni di vincolo, sia lineari che non lineari. L'algoritmo fa uso dei surrogati, cioè di funzioni che approssimano le funzioni obiettivo, con il vantaggio di raggiungere il minimo del surrogato in breve tempo. Quest'ultimo verrà preso come approssimazione del minimo della funzione obiettivo. L'idea è quella di trovare il minimo globale della funzione in pochi tentativi, sfruttando l'esplorazione e la velocità dell'algoritmo.

L'algoritmo si alterna tra due fasi [41, 42]:

- La costruzione della funzione surrogata.

Viene generata una piccola quantità di punti campione (`MinSurrogatePoints`) per cui viene valutato il valore della funzione obiettivo. Attraverso questi punti di prova la funzione obiettivo $f(x)$ viene approssimata con la funzione surrogata $s(x)$ utilizzando un interpolatore cubico di funzioni a base radiale (RBF) con coda lineare. Il punto migliore prende il nome di punto incidente.

L'utilizzo di una RBF è vantaggioso per diversi motivi: la formula per la valutazione della RBF è la stessa a prescindere dalla dimensione del problema o dal numero di punti considerati; il tempo richiesto per la sua valutazione è piccolo; anche l'aggiunta di un punto alla surrogata preesistente richiede poco tempo; nei punti valutati il valore della surrogata e della funzione obiettivo coincide.

- La ricerca del minimo.

Si valuta un gran numero di punti (10^2 - 10^3 punti di prova) pseudocasuali nelle vicinanze del punto incidente e la corrispondente funzione di merito (5.6), escludendo i punti che si trovano a una distanza `MinSampleDistance` (default 10^{-6}) di un punto già valutato. Il punto con il più basso valore della funzione di merito prende il nome di punto adattativo e viene usato per aggiornare la funzione surrogata.

$$\mu_{\omega}(x) = \omega \frac{s(x) - s_{min}}{s_{max} - s_{min}} + (1 - \omega) \frac{d_{max} - d(x)}{d_{max} - d_{min}} \quad (5.6)$$

s_{min} ed s_{max} sono rispettivamente il valore minimo e massimo della funzione surrogata per i punti campione. d_{min} e d_{max} sono rispettivamente le distanze minima e massima tra tutte le possibili coppie di punti campione. Come $d(x)$ si considera la distanza più piccola del punto x con un altro punto. ω è un coefficiente pesato, positivo e minore di 1. Per valori di ω piccoli, si sposta l'attenzione verso punti più lontani da quelli valutati, andando ad esplorare nuove regioni del dominio di ricerca. Per grandi valori di ω viene data importanza alla funzione surrogata, favorendo la ricerca verso la sua minimizzazione. Allora ω viene fatta variare ciclicamente tra quattro valori: 0.3, 0.5, 0.8 e 0.95, a ciascuno dei quali corrisponde un metodo per il campionamento casuale dei punti.

Si valuta la funzione obiettivo per il punto adattativo e se è sufficientemente inferiore rispetto al punto incidente il solutore considera che la ricerca ha avuto successo. Il punto adattativo diventa il nuovo punto incidente. In caso di fallimento, il punto incidente non cambia. La dispersione dei punti da valutare viene aggiornata verso l'alto o verso il basso sulla base dei successi e dei fallimenti verificatisi. La procedura continua fino a quando tutti i punti di prova non si trovano all'interno della `MinSampleDistance`. A questo punto si rimuovono dalla funzione surrogata tutti i punti adattativi trovati e si torna al primo *step* della procedura valutando `MinSurrogatePoints` punti campione casuali.

La procedura appena descritta è illustrata in FIGURA 5.6. Non c'è un vero e proprio criterio di arresto che fermi il risolutore quando ha raggiunto il minimo globale. Piuttosto si definisce un numero di valutazioni massime o un certo lasso di tempo, oltre i quali viene presa come soluzione la migliore trovata in questo budget computazionale [43]. Non sempre la funzione `surrogateopt` riesce a trovare il minimo globale della funzione obiettivo entro il numero di valutazioni imposte. La soluzione trovata dall'algoritmo potrebbe essere solo avvicinarsi a quella reale. Tuttavia il tempo di calcolo è talmente esiguo che `surrogateopt` può essere tranquillamente utilizzato per fornire l'intorno di ricerca a un altro algoritmo di ottimizzazione.

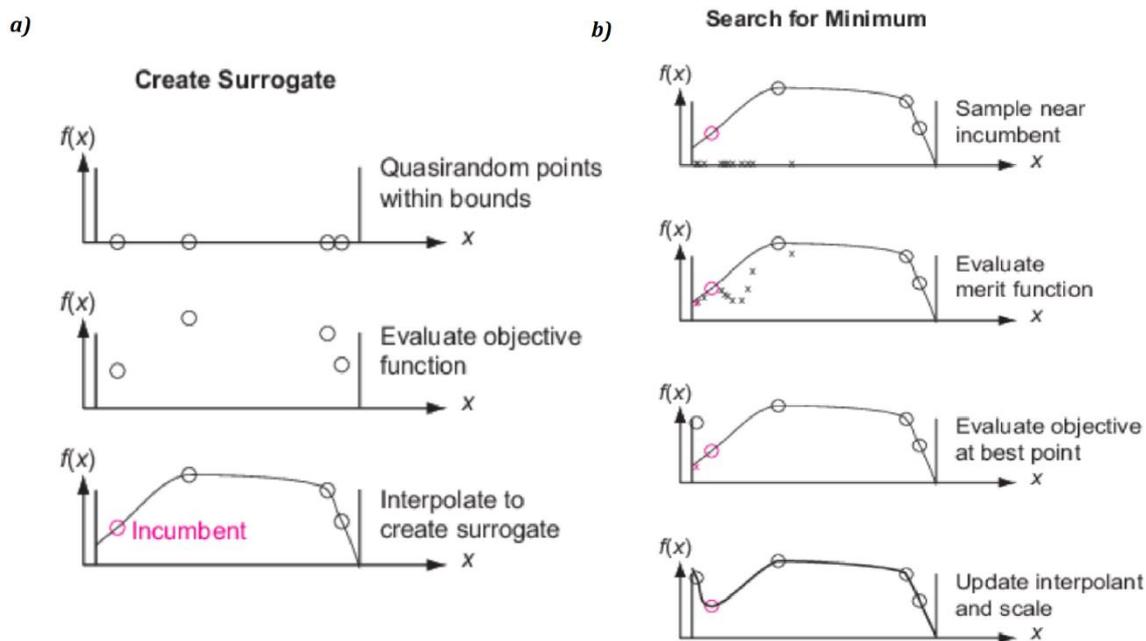


Figura 5.6 Rappresentazione delle due fasi dell'ottimizzazione dei surrogati: a) creazione della surrogata; b) ricerca del minimo [42]

5.2.3 Algoritmi di ricerca diretta

Gli algoritmi di ricerca diretta costituiscono una buona scelta per i casi in cui la derivata della funzione non è nota o la funzione non è continua. Si basano sulla valutazione di un insieme di punti, che prende il nome di maglia, intorno al punto corrente fornito dall'iterazione precedente, con lo scopo di trovare un punto in cui la funzione obiettivo ha un valore minore rispetto a quello del punto corrente. Nel *Global Optimization Toolbox* di MATLAB® sono implementati tre algoritmi di ricerca diretta: il GPS (*Generalized Pattern Search*), il GSS (*Generating Set Search*) e il MADS (*Mesh Adaptive Search*). La funzione che permette di realizzare questo tipo di ottimizzazione è `patternsearch`.

Con il termine *pattern* si intende l'insieme dei vettori $\{v_i\}$ che viene utilizzato dall'algoritmo per determinare i punti da valutare ad ogni iterazione. La dimensione del *pattern* è definita dal numero di variabili decisionali N della funzione obiettivo e dall'insieme di basi positive utilizzato. Per quanto riguarda quest'ultimo, si usano comunemente la base massima, e quindi si avranno $2N$ vettori, oppure la base minima, con $N+1$ vettori. Il GPS utilizza dei vettori a direzione fissata. Per esempio, se si considera un problema con 3 variabili indipendenti, per una base positiva $2N$ si avrà:

$$v_1 = [1 \ 0 \ 0]; \ v_2 = [0 \ 1 \ 0]; \ v_3 = [0 \ 0 \ 1]; \ v_4 = [-1 \ 0 \ 0]; \ v_5 = [0 \ -1 \ 0]; \ v_6 = [0 \ 0 \ -1]$$

e per una base positiva $N+1$:

$$v_1 = [1 \ 0 \ 0]; \ v_2 = [0 \ 1 \ 0]; \ v_3 = [0 \ 0 \ 1]; \ v_4 = [-1 \ -1 \ -1]$$

La funzione `patternsearch` utilizza di *default* l'algoritmo GPS con base positiva $2N$. L'algoritmo GSS utilizza lo stesso *pattern* del GPS tranne nei casi in cui il problema ha dei vincoli lineari e il punto corrente si trova vicino al limite del vincolo. In questi casi viene definito un dominio di lavoro che utilizza la distanza dai vincoli. Questa variazione fa sì che il

GSS sia più efficiente del GPS in caso di vincoli lineari. Nel MADS i vettori del *pattern* vengono scelti in modo casuale, con la stessa logica del GPS: con una base di $2N$ vettori verranno scelti casualmente N vettori e gli N negativi corrispondenti; con $N+1$ vettori saranno scelti N vettori più uno che è il negativo della somma degli altri.

L'insieme dei punti che vengono valutati ad ogni iterazione prende il nome di *mesh*. Per creare la *mesh*, viene innanzitutto generato l'insieme di vettori $\{d_i\}$ come il prodotto tra ogni vettore $\{v_i\}$ del *pattern* e uno scalare, Δ^m , la dimensione della *mesh*. Poi si somma ciascun vettore d_i al punto corrente, cioè il punto in cui si è trovato il miglior valore della funzione obiettivo. Ogni vettore d_i prende il nome di direzione del punto della *mesh* generato. I punti della *mesh* che non soddisfano i limiti e le funzioni di vincolo del problema vengono modificati in modo che risultino fattibili.

Lo *step* in cui l'algoritmo valuta la funzione obiettivo per ogni punto della *mesh* è noto come *polling*. Secondo l'impostazione di *default* della funzione (`CompletePoll=off`) il *polling* si interrompe non appena viene trovato un punto il cui valore della funzione obiettivo è inferiore rispetto a quello del punto corrente. Allora il *polling* ha avuto successo e il nuovo punto trovato diventa il punto corrente nell'iterazione successiva. Se l'opzione `CompletePoll` è settata a `on`, l'algoritmo valuta la funzione obiettivo per ciascun punto della *mesh*, dopodiché considera il valore più basso e lo confronta con quello del punto corrente. Nel caso in cui il *polling* non trovasse un valore della funzione obiettivo migliore, l'azione verrà considerata un fallimento e il punto corrente rimarrà invariato per l'iterazione successiva.

Ad ogni iterazione la dimensione della maglia viene modificata. L'opzione di *default* prevede che Δ^m venga moltiplicata per 2 in caso di successo del *polling*, e divisa per 2 dopo un fallimento [44].

L'algoritmo si arresta quando si verifica una delle seguenti condizioni [45]:

- La dimensione della *mesh* è minore della tolleranza impostata (*default* 10^{-6});
- Viene superato il numero massimo di iterazioni (*default* 100 per il numero delle variabili decisionali);
- Viene raggiunto il valore massimo impostato per il numero di valutazioni della funzione obiettivo (*default* 2000 per il numero delle variabili decisionali);
- Viene superato il tempo massimo di calcolo (di *default* è impostato come infinito);
- Al termine di un *polling* concluso con successo, la distanza tra i punti trovati nelle ultime due iterazioni e Δ^m sono entrambi minori di una `StepTolerance` impostata (*default* 10^{-6});
- Al termine di un *polling* concluso con successo, la variazione della funzione obiettivo nelle ultime due iterazioni è minore della tolleranza imposta (*default* 10^{-6}) e la dimensione della *mesh* è minore di `StepTolerance`.

Come spiegato, la procedura seguita dall'algoritmo prevede l'esistenza di un punto corrente, da cui poter determinare la nuova *mesh*. La funzione `patternsearch` richiede quindi che tra gli *input* venga fornito anche un punto iniziale x_0 che possa essere utilizzato come punto corrente nella prima iterazione. Nel codice di ottimizzazione sviluppato, il punto x_0 fornito alla funzione è proprio la soluzione ottenuta dall'algoritmo `surrogateopt`. In questo modo la ricerca del minimo verrà di gran lunga facilitata. Si è verificato che le due funzioni `surrogateopt+patternsearch` sono più efficienti del `ga` per lo stesso problema, fornendo una soluzione miglior in un minor tempo.

5.3 Applicazione degli algoritmi al problema di separazione

Nel capitolo 4 sono stati presentati gli schemi di separazione a due stadi che si intende confrontare, in modo da individuare quali garantiscono le migliori prestazioni con il minor impatto economico. Si considerano quattro tipologie di schemi di processo (in serie, in cascata, in serie con ricircolo, in cascata con ricircolo) ognuno dei quali verrà analizzato in quattro varianti, a seconda della modalità con cui si sceglie di fornire la differenza di pressione transmembrana. Si ha quindi un totale di 16 configurazioni, per ognuna delle quali verrà effettuata un'ottimizzazione che permetta di individuare le migliori condizioni di processo. Le varianti di ogni schema di processo verranno considerate nel seguente ordine:

- 1) Entrambi gli stadi vengono fatti operare ad alta pressione, per cui è prevista la compressione dell'alimentazione di ciascuno stadio;
- 2) L'alimentazione al primo stadio è sottoposta a compressione, in modo da operare ad alta pressione, mentre si opererà in condizioni di vuoto nel lato di permeato del secondo stadio;
- 3) La differenza di pressione al primo stadio è fornita con una pompa a vuoto sul lato del permeato, mentre l'alimentazione al secondo stadio è soggetta a compressione;
- 4) Si opera in condizioni di bassa pressione in entrambi gli stadi, grazie a delle pompe a vuoto sui flussi di permeato.

Per ogni schema di processo si è sviluppata una funzione, che ne risolve le equazioni nelle sue quattro varianti. Si è scelto di utilizzare il modello ideale in controcorrente per la descrizione del singolo stadio di separazione, in quanto si è dimostrato il più performante in confronto al modello di perfetta miscelazione e al modello *cross-flow*.

In ogni stadio, la pressione dal lato della membrana determinata dal grado di compressione o di vuoto applicato diventa una variabile decisionale del sistema di ottimizzazione, mentre la pressione dal lato opposto della membrana viene mantenuta a pressione atmosferica, a meno di eccezioni in cui rimane applicato lo stato di compressione dello stadio precedente. In generale, il flusso da trattare arriva all'impianto di separazione a pressione atmosferica.

L'ottimizzazione del processo ha il duplice obiettivo di minimizzare l'area totale della membrana (somma dei due stadi) necessaria a realizzare la separazione, e minimizzare i consumi energetici dell'impianto, considerando il sistema di compressione e l'eventuale perdita di prodotto con valore energetico. L'ottimizzazione multiobiettivo verrà svolta con la funzione `gamultiobj` di MATLAB®, che usa la seguente sintassi:

```
[optval,objval] = gamultiobj(FitnessFunction,nvars,A,b,Aeq,beq,lb,ub,options)
```

Gli *output* della funzione sono gli ottimali di Pareto (`optval`) le cui colonne si riferiscono a ciascuna variabile decisionale, e i valori ottimizzati delle funzioni obiettivo (`objval`), in cui ogni colonna è riferita a un obiettivo. Gli *input* essenziali sono le funzioni obiettivo (`FitnessFunction`), scritte in una funzione a sé e dipendenti dal vettore di variabili decisionali `optvar`, e il numero di variabili decisionali `nvars`. In tutti i casi esaminati si hanno tre variabili decisionali, come sarà possibile osservare in seguito. Le funzioni scritte per ciascuno schema di processo (nominate `duestadiserie`, `duesticascata`, `duestadiseriercy`, `duesticascatarcy`) sono quelle utilizzate come `FitnessFunction=@(optvar)`. Per ogni funzione, l'indice `type` (da 1 a 4) definisce quale variante dello schema deve essere risolta. Gli altri *input* della funzione di ottimizzazione definiscono i vincoli del problema, come segue:

$A \cdot \text{optvar} \leq b$

$A_{\text{eq}} \cdot \text{optvar} = b$

$lb \leq \text{optvar} \leq ub$

In *options* si chiede di trovare il fronte di Pareto sostituendo ai parametri di *default* quelli inseriti dall'utente. In particolare, si indicherà una dimensione della popolazione (*PopulationSize*) di 500 individui, un valore intermedio considerato il tempo di calcolo e l'accuratezza delle soluzioni. Come già anticipato, all'algoritmo si forniranno anche gli estremi della curva di Pareto calcolati con un'ottimizzazione di ciascun obiettivo singolarmente. Nelle opzioni si indicherà come *InitialPopulationMatrix* la matrice di dimensioni [numero di obiettivi x numero di variabili], costituita dalle soluzioni (nello spazio delle variabili decisionali) dell'ottimizzazione di ciascuna funzione obiettivo. Come algoritmo di ottimizzazione a un obiettivo, verrà utilizzato il *particle swarm* nei casi in cui gli unici vincoli del problema sono i limiti inferiore e superiore delle variabili decisionali:

```
[sol,minobj]=particleswarm(objfun,nvars,lb,ub,options)
```

sol è la soluzione dell'ottimizzazione, *minobj* il minimo trovato della funzione obiettivo, e *objfun* la funzione obiettivo dipendente da *optvar*. In *options* si può ad esempio scegliere di fornire la *SwarmSize*, quindi il numero di particelle che costituiscono lo sciame, che di *default* è 30; o anche la *FunctionTolerance*, che definisce il criterio d'arresto dell'algoritmo. Per i casi in cui ai limiti delle variabili decisionali va aggiunta anche una funzione di vincolo di disuguaglianza, che consisterà in una relazione tra le pressioni operative dei due stadi, si accoppierà l'algoritmo dei surrogati al *pattern search*. La funzione *surrogateopt* si occuperà di avvicinarsi alla soluzione ottimale e fornire il punto iniziale alla funzione *patternsearch*:

```
[sol,minobj]=surrogateopt(objfun,lb,ub,[],A,b,[],[],options)
```

```
[sol,minobj]=patternsearch(objfun,sol0,A,b,[],[],lb,ub,[],options)
```

La soluzione del *surrogateopt* può essere migliorata andando a modificare alcuni parametri di *default*: *MaxFunctionEvaluations*, come criterio d'arresto; *MinSurrogatePoints*, quindi il numero iniziale di punti utilizzato per creare la funzione surrogata; *MinSampleDistance*, la distanza minima tra i punti generati affinché essi vengano valutati. Per l'algoritmo *pattern search*, si può scegliere di cambiare il *PollMethod* da *GPSPositiveBasis2N* a *GSSPositiveBasis2N*, e diminuire i criteri d'arresto *FunctionTolerance*, *MeshTolerance* e *StepTolerance*.

Come già fatto nello sviluppo dei modelli di flusso in un modulo a membrana, anche la procedura di ottimizzazione verrà sviluppata per due casi differenti: quello in cui il prodotto del processo è la corrente di permeato, e quello in cui il prodotto di interesse è il retentato. Tra i due casi a variare sono principalmente i dati forniti al problema, per cui andranno seguite procedure diverse, soprattutto nella risoluzione di ciascuno schema di processo. Nei prossimi paragrafi si descriveranno le funzioni sviluppate, le variabili decisionali del problema e i rispettivi vincoli.

5.3.1 Ottimizzazione con focus sul permeato

Lo stadio di separazione viene modellato con il modello di flusso in controcorrente in cui si forniscono le specifiche sul permeato. La funzione è la *countercurrentmodel* sviluppata per il caso 1 (APPENDICE C). Gli *input* necessari per risolvere ciascuno stadio e gli *output* forniti dalla funzione sono quelli descritti nel capitolo 3.

I parametri del problema noti sono: pressione, temperatura, portata e composizione della miscela da trattare; selettività, permeabilità e spessore della membrana; la temperatura operativa tollerata dalla membrana; infine il recupero totale del componente di interesse nel permeato. Si vuole fissare anche una specifica sulla purezza del permeato, per cui si aggiunge

tra i dati del problema la purezza minima $y_{p,min}$ che il prodotto finale deve possedere. Per ogni configurazione, si effettua una prima ottimizzazione che ha come solo obiettivo quello di massimizzare la purezza del permeato. Se la purezza massima ottenibile è comunque minore della purezza minima tollerata, non vi è motivo di procedere con un'ottimizzazione multiobiettivo, la configurazione corrente viene scartata e si passa direttamente all'analisi della successiva.

Le variabili decisionali di ciascuna configurazione sono tre, di cui una pressione operativa per ciascuno stadio (del lato di alta o bassa pressione, a seconda se viene utilizzato un compressore o una pompa a vuoto), e il recupero di uno dei due stadi, che ne definisce la separazione. La pressione sul lato opposto della membrana è nota perché mantenuta in condizioni atmosferiche, così come la pressione di mandata delle pompe. Il secondo recupero può essere calcolato da alcune considerazioni sui bilanci di materia, come già spiegato nel capitolo 4. Ogni variabile decisionale è vincolata all'interno di limiti definiti. Si è scelta una pressione massima di compressione $p_{max}=10$ [bar], e una pressione minima di vuoto $p_{min}=0.2$ [bar] sulla base dei limiti pratici delle pompe a vuoto su larga scala industriale [25].

Le quattro funzioni che risolvono ciascuno schema di processo sono riportate in APPENDICE D.

- Due stadi in serie (SS)

Le variabili decisionali con i rispettivi vincoli sono specificate in TABELLA 5.1, per ciascuna variante dello schema. Nella configurazione SS-1 si impone l'ulteriore vincolo che l'alta pressione del secondo stadio sia maggiore dell'alta pressione del primo stadio. Le apparecchiature per la compressione/espansione del gas, la rispettiva alimentazione e rapporto di compressione, sono schematizzati in TABELLA 5.2.

Tabella 5.1 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in serie

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SS-1	p_{h1}^* [bar]	1.5	10	R_{p1}	0	R_p	p_{h2}^* [bar]	1.5	10
SS-2	p_{h1} [bar]	1.5	10	R_{p1}	0	R_p	p_{l2} [bar]	0.2	0.9
SS-3	p_{l1} [bar]	0.2	0.5	R_{p1}	0	R_p	p_{h2} [bar]	1.5	10
SS-4	p_{l1} [bar]	0.2	0.5	R_{p1}	0	R_p	p_{l2} [bar]	0.2	0.5

*funzione di vincolo: $p_{h1} - p_{h2} \leq 0$

Il retentato uscente dal primo stadio diventa l'alimentazione del secondo stadio. Il recupero del secondo stadio si calcola con l'equazione (4.8), facendo uso delle condizioni di alimentazione di primo e secondo stadio e del recupero totale specificato. La portata del permeato totale ottenuto dai due stadi e la sua composizione vengono calcolati con le equazioni (4.5) e (4.6).

Tabella 5.2 Tabella riassuntiva del sistema di compressione in ciascuna variante dello schema di separazione in serie; è specificato il tipo di apparecchiatura (C: compressore; P: pompa a vuoto; E: turbina), la corrente trattata e il rapporto di compressione.

	1			2			3		
	tipo	Q	v	tipo	Q	v	tipo	Q	v
SS-1	C	Q_f	$\frac{p_{h1}}{p_f}$	C	Q_{r1}	$\frac{p_{h2}}{p_{h1}}$	E	Q_r	$\frac{p_{atm}}{p_{h2}}$
SS-2	C	Q_f	$\frac{p_{h1}}{p_f}$	P	Q_{p2}	$\frac{p_{atm}}{p_{l2}}$	E	Q_r	$\frac{p_{atm}}{p_{h1}}$
SS-3	P	Q_{p1}	$\frac{p_{atm}}{p_{l1}}$	C	Q_{r1}	$\frac{p_{h2}}{p_f}$	E	Q_r	$\frac{p_{atm}}{p_{h2}}$
SS-4	P	Q_{p1}	$\frac{p_{atm}}{p_{l1}}$	P	Q_{p2}	$\frac{p_{atm}}{p_{l2}}$	-	-	-

- Due stadi in cascata (SC)

In TABELLA 5.3 sono specificate le variabili decisionali del problema di ottimizzazione, per ciascuna variante dello schema con due stadi in cascata. Gli unici vincoli del problema saranno i limiti inferiore e superiore delle variabili decisionali. In TABELLA 5.4 si schematizzano, invece, le apparecchiature utilizzate per la compressione e/o espansione del gas, con annessa la portata trattata e il rapporto di compressione.

Tabella 5.3 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in cascata

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SC-1	p_{h1} [bar]	1.5	10	R_{p1}	R_p	1	p_{h2} [bar]	1.5	10
SC-2	p_{h1} [bar]	1.5	10	R_{p1}	R_p	1	p_{l2} [bar]	0.2	0.5
SC-3	p_{l1} [bar]	0.2	0.5	R_{p1}	R_p	1	p_{h2} [bar]	1.5	10
SC-4	p_{l1} [bar]	0.2	0.5	R_{p1}	R_p	1	p_{l2} [bar]	0.2	0.5

Come per lo schema dei due stadi in serie, anche in questo caso il sistema può essere risolto molto facilmente. Al secondo stadio viene alimentato il permeato ottenuto con il primo stadio di separazione. Il secondo stadio ha un recupero calcolato dalla relazione (4.18), equivale quindi al rapporto tra il recupero totale e il recupero del secondo stadio. Tale relazione è stata ottenuta esaminando le equazioni di bilancio, e spiega anche il motivo per cui sono stati imposti i limiti $R_p < R_{p1} < 1$. Si ricorda, infatti, che il recupero ha sempre un valore compreso da 0 e 1, per definizione. Il permeato finale si ottiene dal secondo stadio. Il retentato di primo e secondo stadio vengono miscelati, ottenendo un retentato con portata e composizione definiti dalle espressioni (4.15) e (4.16).

Tabella 5.4 Tabella riassuntiva del sistema di compressione in ciascuna variante dello schema di separazione in cascata; è specificato il tipo di apparecchiatura (C: compressore; P: pompa a vuoto; E: turbina), la corrente trattata e il rapporto di compressione.

	1			2			3			4		
	tipo	Q	v									
SC-1	C	Q_f	$\frac{p_{h1}}{p_f}$	C	Q_{p1}	$\frac{p_{h2}}{p_{atm}}$	E	Q_{r1}	$\frac{p_{atm}}{p_{h1}}$	E	Q_{r2}	$\frac{p_{atm}}{p_{h2}}$
SC-2	C	Q_f	$\frac{p_{h1}}{p_f}$	P	Q_p	$\frac{p_{atm}}{p_{l2}}$	E	Q_{r1}	$\frac{p_{atm}}{p_{h1}}$	-	-	-
SC-3	P	Q_{p1}	$\frac{p_{atm}}{p_{l1}}$	C	Q_{p1}	$\frac{p_{h2}}{p_{atm}}$	E	Q_{r2}	$\frac{p_{atm}}{p_{h2}}$	-	-	-
SC-4	P	Q_{p1}	$\frac{p_{atm}}{p_{l1}}$	P	Q_p	$\frac{p_{atm}}{p_{l2}}$	-	-	-	-	-	-

- Due stadi in serie con ricircolo (SSR)

Nello schema in cui due stadi di separazione sono posti in serie e il permeato del secondo stadio viene ricircolato al primo stadio, si sceglie come variabile decisionale del problema, oltre alle pressioni operative, il recupero del secondo stadio (riferito al permeato). Infatti, l'introduzione del ricircolo nel sistema complica il problema, non possedendo informazioni né sulla portata del primo stadio né su quella del secondo stadio. Impostare il recupero del secondo stadio come variabile decisionale permette di conoscere la portata del componente A ricircolata al primo stadio (5.8) e di calcolare il recupero nel primo stadio dall'espressione (4.25).

$$Q_{r1}x_{r1} = \frac{Q_r x_r}{(1-R_{p2})} = \frac{Q_f x_f (1-R_p)}{(1-R_{p2})} \quad (5.7)$$

è la portata di A alimentata al secondo stadio.

$$Q_{p2}y_{p2} = Q_{r1}x_{r1}R_{p2} \quad (5.8)$$

A questo punto si può ipotizzare l'alimentazione al primo stadio come la somma di Q_f e $Q_{p2}y_{p2}$, solo come portata di primo tentativo, e calcolarne la composizione come:

$$x_{f1} = \frac{Q_f x_f + Q_{p2} y_{p2}}{Q_{f1}} \quad (5.9)$$

si procede con la risoluzione del primo e poi del secondo stadio, trovando una nuova portata Q_{p2} e composizione y_{p2} della corrente da ricircolare. Si ricalcolano le condizioni di alimentazione al primo stadio, seguendo una procedura *trial&error* fino a quando l'errore relativo tra la portata Q_{f1} corrente e quella calcolata al tentativo precedente non è inferiore alla tolleranza impostata (10^{-7}). Il prodotto finale è il permeato prodotto dal primo stadio. In TABELLA 5.5 si specificano i limiti entro cui le variabili decisionali sono vincolate. Per la configurazione SSR-1 si impone l'ulteriore vincolo che l'alta pressione del secondo stadio sia maggiore dell'alta pressione al primo stadio. In TABELLA 5.6 si riportano le apparecchiature del sistema di compressione, con le relative portate e rapporti di compressione. Il ricircolo si miscela al *feed* a pressione atmosferica, assicurandosi che si trovi alla temperatura operativa opportuna.

Tabella 5.5 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in serie e il permeato del secondo stadio viene ricircolato

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SSR-1	p_{h1}^* [bar]	1.5	10	p_{h2}^* [bar]	1.5	10	R_{p2}	0	1
SSR-2	p_{h1} [bar]	1.5	10	p_{l2} [bar]	0.2	0.9	R_{p2}	0	1
SSR-3	p_{l1} [bar]	0.2	0.5	p_{h2} [bar]	1.5	10	R_{p2}	0	1
SSR-4	p_{l1} [bar]	0.2	0.5	p_{l2} [bar]	0.2	0.5	R_{p2}	0	1

*funzione di vincolo: $p_{h1} - p_{h2} \leq 0$

Tabella 5.6 Tabella riassuntiva del sistema di compressione in ciascuna variante dello schema di separazione in serie (con ricircolo); è specificato il tipo di apparecchiatura (C: compressore; P: pompa a vuoto; E: turbina), la corrente trattata e il rapporto di compressione.

	1			2			3		
	tipo	Q	v	tipo	Q	v	tipo	Q	v
SSR-1	C	Q_{f1}	$\frac{p_{h1}}{p_{atm}}$	C	Q_{r1}	$\frac{p_{h2}}{p_{h1}}$	E	Q_r	$\frac{p_{atm}}{p_{h2}}$
SSR-2	C	Q_{f1}	$\frac{p_{h1}}{p_{atm}}$	P	Q_{p2}	$\frac{p_{atm}}{p_{l2}}$	E	Q_r	$\frac{p_{atm}}{p_{h1}}$
SSR-3	P	Q_p	$\frac{p_{atm}}{p_{l1}}$	C	Q_{r1}	$\frac{p_{h2}}{p_{atm}}$	E	Q_r	$\frac{p_{atm}}{p_{h2}}$
SSR-4	P	Q_p	$\frac{p_{atm}}{p_{l1}}$	P	Q_{p2}	$\frac{p_{atm}}{p_{l2}}$	-	-	-

▪ Due stadi in cascata con ricircolo (SCR)

Anche in questo caso l'introduzione di un ricircolo nel sistema richiede un calcolo di tipo iterativo per individuare le caratteristiche dell'alimentazione al primo stadio. In questo caso le variabili decisionali tornano ad essere le pressioni operative, una per ogni stadio, e il recupero del primo stadio (TABELLA 5.7). Da quest'ultimo è possibile calcolare la portata di A ricircolata (5.10), da cui si trova la portata di A alimentata al secondo stadio (5.11) e il suo recupero (4.34).

$$Q_{r2}x_{r2} = \frac{Q_f x_f (R_{p1} - R_p)}{1 - R_{p1}} \quad (5.10)$$

$$Q_{p1}y_{p1} = (Q_f x_f + Q_{r2}x_{r2})R_{p1} \quad (5.11)$$

Si ipotizzano le condizioni di alimentazione al primo stadio (5.12, 5.13), di primo tentativo. Ciò permette di risolvere i due stadi e trovare portata Q_{r2} e composizione x_{r2} della corrente da ricircolare. Si ricalcolano le caratteristiche del flusso da alimentare al primo stadio e la procedura si ripete fino a quando non viene soddisfatta la tolleranza sull'errore relativo della portata di alimentazione nelle ultime due iterazioni.

$$Q_{f1, \text{primo tentativo}} = Q_f + 2Q_{r2}x_{r2} \quad (5.12)$$

$$x_{f1} = \frac{Q_f x_f + Q_{r2}x_{r2}}{Q_{f1}} \quad (5.13)$$

Il sistema di compressione utilizzato in ciascuna variante dello schema è riassunto in TABELLA 5.8, con le rispettive portate alimentate e i rapporti di compressione. Il ricircolo viene miscelato all'alimentazione a pressione atmosferica e alla temperatura operativa del modulo a membrana. Un'eccezione si ha nello schema SCR-1, in cui si impone come funzione di vincolo che l'alta pressione del secondo stadio sia maggiore dell'alta pressione del primo stadio: questo permette di poter espandere il retentato del secondo stadio fino alla temperatura operativa del primo stadio, evitando che venga ulteriormente compresso.

Tabella 5.7 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in cascata e il retentato del secondo stadio viene ricircolato

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SCR-1	p_{h1}^* [bar]	1.5	10	R_{p1}	R_p	1	p_{h2}^* [bar]	1.5	10
SCR-2	p_{h1} [bar]	1.5	10	R_{p1}	R_p	1	p_{l2} [bar]	0.2	0.5
SCR-3	p_{l1} [bar]	0.2	0.5	R_{p1}	R_p	1	p_{h2} [bar]	1.5	10
SCR-4	p_{l1} [bar]	0.2	0.5	R_{p1}	R_p	1	p_{l2} [bar]	0.2	0.5

*funzione di vincolo: $p_{h1} - p_{h2} \leq 0$

Tabella 5.8 Tabella riassuntiva del sistema di compressione in ciascuna variante dello schema di separazione in cascata (con ricircolo); è specificato il tipo di apparecchiatura (C: compressore; P: pompa a vuoto; E: turbina), la corrente trattata e il rapporto di compressione.

	1			2			3			4		
	tipo	Q	v	tipo	Q	v	tipo	Q	v	tipo	Q	v
SCR-1	C	Q_f	$\frac{p_{h1}}{p_f}$	C	Q_{p1}	$\frac{p_{h2}}{p_{atm}}$	E	Q_{r1}	$\frac{p_{atm}}{p_{h1}}$	E	Q_{r2}	$\frac{p_{h1}}{p_{h2}}$
SCR-2	C	Q_{f1}	$\frac{p_{h1}}{p_{atm}}$	P	Q_p	$\frac{p_{atm}}{p_{l2}}$	E	Q_{r1}	$\frac{p_{atm}}{p_{h1}}$	-	-	-
SCR-3	P	Q_{p1}	$\frac{p_{atm}}{p_{l1}}$	C	Q_{p1}	$\frac{p_{h2}}{p_{atm}}$	E	Q_{r2}	$\frac{p_f}{p_{h2}}$	-	-	-
SCR-4	P	Q_{p1}	$\frac{p_{atm}}{p_{l1}}$	P	Q_p	$\frac{p_{atm}}{p_{l2}}$	-	-	-	-	-	-

5.3.2 Ottimizzazione con focus sul retentato

A volte, il prodotto di maggior interesse nel processo è il componente concentrato nel retentato. In questi casi le specifiche di progetto si riferiscono alla corrente di retentato, appunto, per cui nello svolgere l'ottimizzazione si è fatto uso della funzione `countercurrentmodel.m` sviluppata per il caso 2 (APPENDICE C). La funzione permette di calcolare le prestazioni di un solo stadio di separazione e la rispettiva superficie di membrana da installare, fornendo gli *input* descritti nel capitolo 3.

I parametri noti sono: le caratteristiche del *feed* (portata, composizione, pressione e temperatura); le proprietà della membrana (selettività, permeabilità, spessore); la temperatura operativa; la composizione del retentato finale.

Tra le variabili decisionali vi è la composizione del retentato prodotto da uno dei due stadi, oltre alla pressione in ciascuno stadio definita dal grado di compressione. Si sceglie di impostare come pressione massima $p_{h,max}=10$ [bar], e come limite inferiore di vuoto raggiungibile $p_{l,min}=0.2$ [bar] (causa limiti pratici delle pompe a vuoto su larga scala industriale [25]). La pressione di mandata delle pompe a vuoto si fissa a pressione atmosferica.

Le funzioni che risolvono le equazioni per ciascuno schema di processo sono riportate in APPENDICE E.

- Due stadi in serie (SS)

La composizione del retentato del secondo stadio è nota in quanto specifica di progetto. Ponendo la composizione del retentato del primo stadio come una delle tre variabili decisionali, ogni parametro necessario al calcolo è noto. I limiti entro cui sono vincolate le variabili decisionali sono raccolti in TABELLA 5.9. Nella variante dello schema SS-1 si imposta la pressione operativa del secondo stadio maggiore di quella nel primo stadio. Si calcola infine il recupero di B nel retentato secondo la definizione (3.2). I parametri per il calcolo dei consumi del sistema di compressione rimangono quelli descritti in TABELLA 5.2.

Tabella 5.9 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in serie

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SS-1	p_{h1} * [bar]	1.5	10	x_{r1}	x_r	x_f	p_{h2} * [bar]	1.5	10
SS-2	p_{h1} [bar]	1.5	10	x_{r1}	x_r	x_f	p_{l2} [bar]	0.2	0.9
SS-3	p_{l1} [bar]	0.2	0.5	x_{r1}	x_r	x_f	p_{h2} [bar]	1.5	10
SS-4	p_{l1} [bar]	0.2	0.5	x_{r1}	x_r	x_f	p_{l2} [bar]	0.2	0.5

*funzione di vincolo $p_{h1} - p_{h2} \leq 0$

- Due stadi in cascata (SC)

In questo schema il retentato finale è dato dalla miscelazione dei retentati dei due stadi, di cui non si conosce né la composizione né la portata. Il primo stadio può essere facilmente risolto scegliendo di porre la composizione del retentato come una delle tre variabili decisionali. La composizione del retentato del secondo stadio è legata agli altri parametri del processo secondo l'equazione (5.14).

$$x_{r2} = \frac{Q_r x_r - Q_{r1} x_{r1}}{Q_{r2}} = \frac{(Q_r - Q_{r1}) x_r - Q_{r1} x_{r1}}{Q_{r2}} \quad (5.14)$$

Si utilizza un procedimento di tipo iterativo in cui si ipotizza la portata Q_{r2} , si calcola x_{r2} e si risolvono le equazioni per il secondo stadio, da cui si ottiene un nuovo valore di portata da confrontare con quello dell'iterazione precedente. La procedura termina quando viene soddisfatta la tolleranza sull'errore relativo della portata. In TABELLA 5.10

si riassumono i vincoli delle variabili decisionali. Il sistema di compressione è lo stesso presentato in TABELLA 5.4.

Tabella 5.10 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in cascata

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SC-1	p_{h1} [bar]	1.5	10	x_{r1}	0	x_r	p_{h2} [bar]	1.5	10
SC-2	p_{h1} [bar]	1.5	10	x_{r1}	0	x_r	p_{l2} [bar]	0.2	0.5
SC-3	p_{l1} [bar]	0.2	0.5	x_{r1}	0	x_r	p_{h2} [bar]	1.5	10
SC-4	p_{l1} [bar]	0.2	0.5	x_{r1}	0	x_r	p_{l2} [bar]	0.2	0.5

▪ Due stadi in serie con ricircolo (SSR)

La composizione del retentato del secondo stadio è nota. Quella del primo stadio diventa una variabile decisionale del problema. Per risolvere le incognite del sistema dovute al ricircolo, si ipotizza una prima portata Q_{r1} che viene alimentata al secondo stadio, in modo da risolverlo con il modello in controcorrente, ottenere le caratteristiche del permeato da ricircolare e calcolare la corrispondente alimentazione al primo stadio (4.21, 5.9). Risolvendo a sua volta le equazioni per il primo stadio, si trova una portata di nuovo tentativo da alimentare al secondo stadio e la procedura viene ripetuta fino al raggiungimento della tolleranza su quest'ultimo parametro Q_{r1} . Il sistema di compressione è lo stesso schematizzato in TABELLA 5.6, mentre in TABELLA 5.11 si riassumono le variabili decisionali e i loro vincoli. La pressione operativa del secondo stadio nella variante SSR-1 è sempre maggiore di quella al primo stadio.

Tabella 5.11 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in serie e il permeato del secondo stadio viene ricircolato

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SSR-1	p_{h1}^* [bar]	1.5	10	x_{r1}	x_r	0.9	p_{h2}^* [bar]	1.5	10
SSR-2	p_{h1} [bar]	1.5	10	x_{r1}	x_r	0.9	p_{l2} [bar]	0.2	0.9
SSR-3	p_{l1} [bar]	0.2	0.5	x_{r1}	x_r	0.9	p_{h2} [bar]	1.5	10
SSR-4	p_{l1} [bar]	0.2	0.5	x_{r1}	x_r	0.9	p_{l2} [bar]	0.2	0.5

*funzione di vincolo $p_{h1} - p_{h2} \leq 0$

▪ Due stadi in cascata con ricircolo (SCR)

È nota la composizione del retentato del primo stadio, mentre quella del secondo stadio, ricircolata, diventa una variabile decisionale del problema. Si ipotizza una portata di primo tentativo da ricircolare, in modo da calcolare le caratteristiche dell'alimentazione

al primo stadio (4.29, 5.13) e risolverne le equazioni. Il permeato ottenuto viene usato come alimentazione al secondo stadio, trovando una nuova portata Q_{r2} di ricircolo, su cui si fa una verifica della tolleranza prima di procedere con il metodo iterativo. I limiti entro cui sono vincolate le variabili decisionali sono descritti in TABELLA 5.12. per il sistema di compressione continua a valere la TABELLA 5.8.

Tabella 5.12 Variabili decisionali e i rispettivi limiti inferiore e superiore per uno schema di processo in cui i due stadi sono in cascata e il retentato del secondo stadio viene ricircolato

	Variabile 1			Variabile 2			Variabile 3		
	var	lb	up	var	lb	up	var	lb	up
SCR-1	p_{h1}^* [bar]	1.5	10	p_{h2}^* [bar]	1.5	10	x_{r2}	0	0.99
SCR-2	p_{h1} [bar]	1.5	10	p_{l2} [bar]	0.2	0.5	x_{r2}	0	0.99
SCR-3	p_{l1} [bar]	0.2	0.5	p_{h2} [bar]	1.5	10	x_{r2}	0	0.99
SCR-4	p_{l1} [bar]	0.2	0.5	p_{l2} [bar]	0.2	0.5	x_{r2}	0	0.99

*funzione di vincolo: $p_{h1} - p_{h2} \leq 0$

5.3.3 Scelta di un punto sulla curva di Pareto

Per ciascuna configurazione ottimizzata si ottiene una curva di Pareto. Un primo confronto tra i fronti di Pareto dà un'indicazione di quali configurazioni andrebbero scartate e quali invece possono costituire una migliore scelta di progetto. Si ricorda che ogni punto appartenente al fronte di Pareto può essere considerato ottimale, sta poi al progettista scegliere la soluzione più appropriata sulla base di considerazioni più approfondite.

La rappresentazione di una generica curva di Pareto è fornita in FIGURA 5.7. Si considerino le due generiche funzioni obiettivo f_1 ed f_2 . Il punto A sulla curva rappresenta le condizioni di progetto che minimizzano la funzione obiettivo f_2 . Il punto B, invece, corrisponde alle condizioni che ottimizzano l'obiettivo f_1 . Si comprende, allora, che nel punto C entrambi gli obiettivi si trovano al loro minimo, e ciò corrisponderebbe alle condizioni ottimali, o meglio alla soluzione ideale. Tuttavia, il punto C non appartiene alla curva di Pareto e prende il nome di punto di utopia. L'idea è quella di optare per una soluzione di compromesso, e cioè la soluzione ottimale di Pareto che minimizza la distanza tra il punto di utopia e la curva di Pareto [46, 47]. Nel scegliere la soluzione più vicina a quella ideale si può utilizzare la distanza euclidea che, tenendo presente la definizione del problema di ottimizzazione (5.1), può essere calcolata secondo la (5.15).

$$d_e(\mathbf{x}) = |\mathbf{f}(\mathbf{x}) - \mathbf{f}^u| = \left\{ \sum_1^M [f_m(\mathbf{x}) - f_m^u]^2 \right\}^{0.5} \quad (5.15)$$

Con \mathbf{f}^u si indica il punto di utopia. Si sceglie allora la soluzione \mathbf{x} che minimizza d_e .

Per poter utilizzare il criterio di minimizzazione della distanza euclidea è necessario che le funzioni obiettivo possedano la stessa unità di misura. In caso contrario, andranno rese adimensionate. Possono essere utilizzati diversi criteri di trasformazione delle funzioni. L'approccio più sicuro prevede la normalizzazione delle funzioni obiettivo, facendo in modo che i loro valori siano compresi tra 0 e 1 (5.16).

$$f_m^t = \frac{f_m(\mathbf{x}) - f_m^u}{f_m^{max} - f_m^u} \quad (5.16)$$

Quindi con f_m^t si indica la funzione obiettivo trasformata, di cui f_m^u è il suo valore minimo, mentre f_m^{max} è il massimo valore della funzione obiettivo m valutato per una soluzione x_i che minimizza un'altra funzione obiettivo.

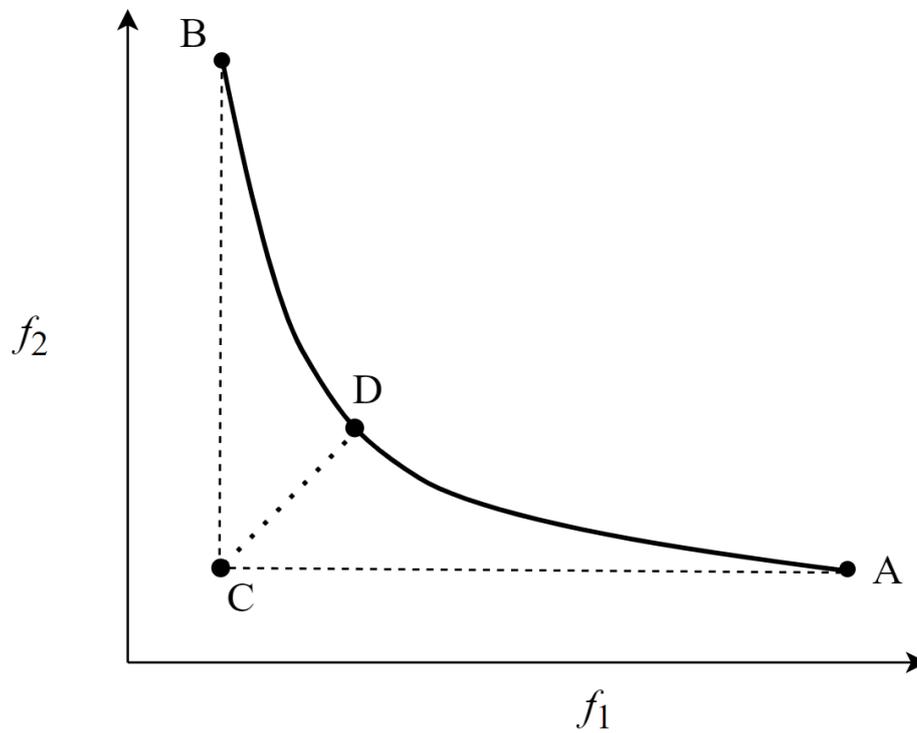


Figura 5.7 Rappresentazione di una curva di Pareto e del punto di utopia

6. Casi studio

6.1 Separazione della CO₂ da un gas di scarico

Si considera una miscela binaria N₂-CO₂ rappresentativa di un gas di scarico di una centrale termoelettrica. I dati (TABELLA 6.1) sono riferiti a una centrale termoelettrica a carbone polverizzato da 650 [MW], e a una membrana polimerica selettiva per la CO₂ [25]. L'impatto sulla membrana delle impurità (NO_x, SO_x, Hg) in piccole concentrazioni non viene considerato, trattando la miscela come binaria al 15% molare di CO₂ e 85% di N₂. Si vuole determinare lo schema di processo ottimale e le migliori condizioni che permettono di ottenere un permeato secondo gli standard per il trasporto della CO₂ nelle condutture [8]. Si specificano quindi un recupero della CO₂ del 90% e una sua purezza nel permeato di almeno il 96%.

Tabella 6.1 Dati utilizzati nel caso studio per la separazione di una miscela N₂-CO₂

Parametro	Valore	Unità di misura
Portata di <i>feed</i> (Q_f)	$2.1 \cdot 10^6$	mSTP ³ h ⁻¹
Pressione di <i>feed</i> (p_f)	1.01	bar
Temperatura (T)	40	°C
Frazione molare CO ₂ (x_f)	0.15	-
Recupero CO ₂ (R_p)	0.90	-
Purezza minima CO ₂ ($y_{p,min}$)	0.96	-
Permeanza (P_{eCO_2})	1000	GPU
Selettività CO ₂ /N ₂ (α)	50	-

Si verificano innanzitutto le prestazioni massime raggiungibili con un solo stadio di separazione nelle condizioni indicate in TABELLA 6.1, con il massimo gradiente di pressione entro i vincoli specificati (10 [bar] per l'alta pressione e 0.2 [bar] per la bassa pressione). Quindi si considera uno schema in cui la differenza di pressione è fornita da un compressore, e uno schema in cui lo stadio opera in condizioni di vuoto. Nel primo, con un recupero del 90% si può raggiungere massimo una purezza del 57.3%, che nel secondo caso diminuisce al 44.8% (TABELLA 6.2).

Tabella 6.2 Prestazioni di un solo stadio di separazione in due diverse condizioni di pressione; area specifica necessaria alla separazione e consumo specifico del sistema di compressione

Singolo stadio	β	y_p [%]	a_m [m ² /(kgCO ₂ s ⁻¹)]	e [MW/(kgCO ₂ s ⁻¹)]
Alta pressione	0.101	57.33	$2.886 \cdot 10^3$	1.137
Bassa pressione	0.198	44.80	$5.2487 \cdot 10^4$	0.305

Il codice sviluppato per l'ottimizzazione è in APPENDICE F. Sono state eseguite 5 simulazioni, di cui si riporteranno i risultati migliori.

La procedura di ottimizzazione finalizzata a trovare la massima purezza del permeato ha escluso tutte le configurazioni senza ricircolo. Come è possibile osservare in TABELLA 6.3, gli schemi con due stadi in serie e due stadi in cascata, senza ricircolo, non permettono di raggiungere la purezza minima di CO₂ richiesta.

Tabella 6.3 Risultati dell'ottimizzazione della purezza per gli schemi in serie e in cascata, senza ricircolo; si riportano la massima purezza di CO₂ nel permeato ottenibile e i corrispondenti valori ottimizzati delle variabili decisionali

schema	$y_{p,max}$	variabile 1	variabile 2	variabile 3
SS-1	0.5733	$p_{h1}=10$ [bar]	$R_{p1}=0.001$	$p_{h2}=10$ [bar]
SS-2	0.7345	$p_{h1}=10$ [bar]	$R_{p1}=0.001$	$p_{l2}=0.2$ [bar]
SS-3	0.5732	$p_{l1}=0.2$ [bar]	$R_{p1}=0.001$	$p_{h2}=10$ [bar]
SS-4	0.4479	$p_{l1}=0.2$ [bar]	$R_{p1}=0.001$	$p_{l2}=0.2$ [bar]
SC-1	0.9142	$p_{h1}=10$ [bar]	$R_{p1}=0.961$	$p_{h2}=10$ [bar]
SC-2	0.8676	$p_{h1}=10$ [bar]	$R_{p1}=0.957$	$p_{l2}=0.2$ [bar]
SC-3	0.8588	$p_{l1}=0.2$ [bar]	$R_{p1}=0.967$	$p_{h2}=10$ [bar]
SC-4	0.7871	$p_{l1}=0.2$ [bar]	$R_{p1}=0.962$	$p_{l2}=0.2$ [bar]

Rimangono le otto configurazioni che includono una corrente di ricircolo. La FIGURA 6.1 rappresenta i fronti di Pareto (generati dall'ottimizzazione multiobiettivo) per le quattro varianti (descritte nel quarto capitolo) dello schema in cui i due stadi di separazione sono posti in serie e il permeato del secondo stadio viene ricircolato al primo stadio. Gli estremi delle curve di Pareto, con i rispettivi valori delle variabili decisionali, sono raccolti in TABELLA 6.4.

Tabella 6.4 Estremi del fronte di Pareto per ciascuna variante dello schema con due stadi di separazione in serie (e il ricircolo del permeato); si riportano le coordinate del punto e la corrispondente soluzione ottimale

schema	a_m [m ² /(kgCO ₂ s ⁻¹)]	e [MW/(kgCO ₂ s ⁻¹)]	p_{h1} o p_{l1} [bar]	p_{h2} o p_{l2} [bar]	R_{p2}
SSR-1	$4.793 \cdot 10^3$	2.129	10	10	0.974
	$4.2203 \cdot 10^4$	1.642	3.64	3.64	0.987
SSR-2	$2.441 \cdot 10^3$	2.508	10	0.2	0.971
	$4.3951 \cdot 10^4$	1.657	3.26	0.853	0.988
SSR-3	$7.390 \cdot 10^3$	2.777	0.2	10	0.986
	$1.0547 \cdot 10^6$	1.079	0.2	1.5	0.993
SSR-4	$9.6349 \cdot 10^4$	1.248	0.2	0.2	0.982
	$4.9628 \cdot 10^5$	1.049	0.2	0.5	0.989

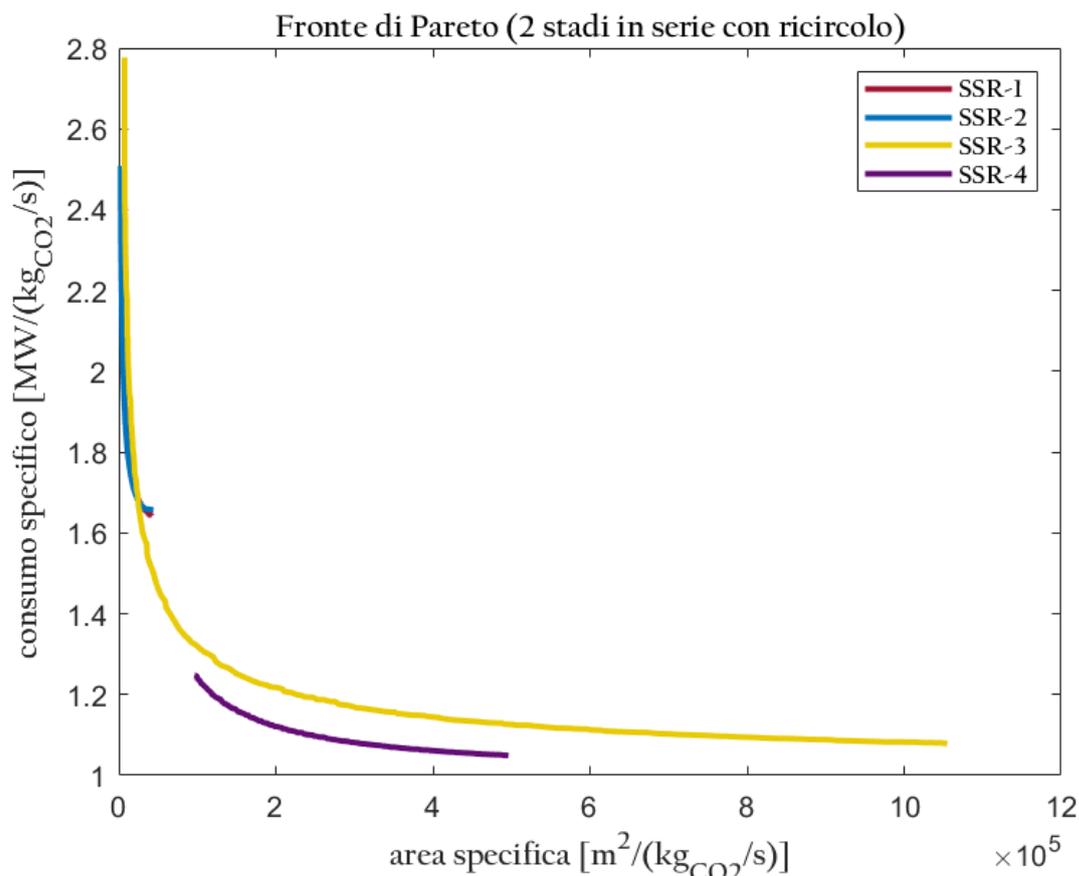


Figura 6.1 Fronti di Pareto per le quattro varianti dello schema con due stadi di separazione in serie e il ricircolo del permeato; ottimizzazione effettuata con la funzione `gamultiobj` di MATLAB®; dati da TABELLA 6.1.

Dai fronti di Pareto si può innanzitutto osservare che lavorare in condizioni di vuoto in entrambi gli stadi permette di raggiungere le specifiche del prodotto con un consumo energetico minore rispetto alle altre tre varianti, anche se l'area della membrana da installare sarà maggiore. Lo schema SSR-3 è quello con la maggiore variabilità dei parametri di progetto. Le varianti SSR-1 e SSR-2 permettono di utilizzare una superficie di membrana minore ma a costo di un maggior consumo energetico.

Gli stessi risultati si riportano per le quattro varianti dello schema con due stadi di separazione in cascata e il ricircolo del permeato del secondo stadio (FIGURA 6.2, TABELLA 6.5). Per la configurazione SCR-4 la procedura di ottimizzazione ha trovato un solo punto sulla curva di Pareto che soddisfi le specifiche del prodotto. In questo caso, è lo schema SCR-2 a fornire una grande variabilità dei parametri di progetto. Lo schema SCR-3 permette di ridurre il consumo energetico, mentre lo schema SCR-1 consente un risparmio in termini di membrana utilizzata.

Per facilitare il confronto tra gli schemi, si è trovata la soluzione di compromesso per ciascuna curva di Pareto, previa normalizzazione delle funzioni obiettivo come indicato nel quinto capitolo. Si confrontano in TABELLA 6.6 le soluzioni ottimali per ciascuna configurazione in grado di produrre un permeato al 96% di purezza (minimo). Si deduce immediatamente che adottare uno schema con due stadi in cascata (più ricircolo) permette di ridurre l'impatto del sistema di separazione, sia in termini di ingombro che di consumi.

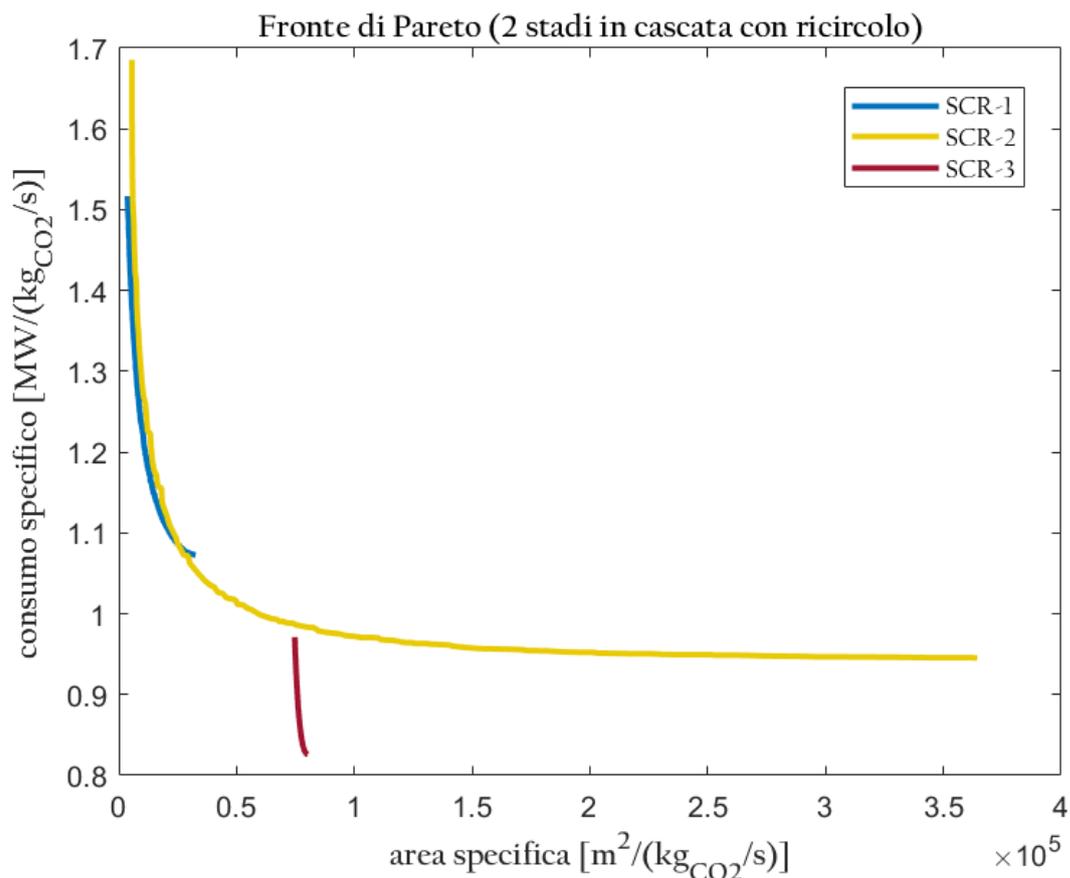


Figura 6.2 Fronti di Pareto per le quattro varianti dello schema con due stadi di separazione in cascata e il ricircolo del permeato; ottimizzazione effettuata con la funzione gamultiobj di MATLAB®; dati da TABELLA 6.1.

Tabella 6.5 Estremi del fronte di Pareto per ciascuna variante dello schema con due stadi di separazione in cascata (e il ricircolo del retentato); si riportano le coordinate del punto e la corrispondente soluzione ottimale

schema	a_m [m ² /(kg _{CO2} s ⁻¹)]	e [MW/(kg _{CO2} s ⁻¹)]	p_{h1} o p_{l1} [bar]	R_{p1}	p_{h2} o p_{l2} [bar]
SCR-1	$3.725 \cdot 10^3$	1.517	10	0.916	10
	$3.2623 \cdot 10^4$	1.072	3.82	0.963	4.01
SCR-2	$5.606 \cdot 10^3$	1.685	10	0.954	0.2
	$3.643 \cdot 10^5$	0.945	1.83	0.985	0.2
SCR-3	$7.4695 \cdot 10^4$	0.971	0.2	0.938	10
	$8.0176 \cdot 10^4$	0.826	0.2	0.956	3.58
SCR-4	$7.9571 \cdot 10^4$	0.654	0.2	0.948	0.2

Tabella 6.6 Area specifica e consumo specifico negli otto schemi con ricircolo (in serie e in cascata) corrispondenti alle soluzioni di compromesso, ottenute a partire dalle curve di Pareto; si indica anche la distanza dal punto di utopia, che normalizzando le funzioni corrisponde al punto (0,0)

Schema	a_m [m ² /(kgCO ₂ s ⁻¹)]	e [MW/(kgCO ₂ s ⁻¹)]
SSR-1	1.4438·10 ⁴	1.769
SSR-2	1.0641·10 ⁴	1.813
SSR-3	1.2420·10 ⁵	1.280
SSR-4	2.1594·10 ⁵	1.111
SCR-1	1.2236·10 ⁴	1.181
SCR-2	4.2136·10 ⁴	1.026
SCR-3	7.6576·10 ⁴	0.870
SCR-4	7.9571·10 ⁴	0.654

In tabella sono evidenziate in grassetto le cinque configurazioni i cui punti di compromesso risultano non dominati. Tra questi, la migliore configurazione viene scelta con la stessa procedura seguita per trovare la soluzione di compromesso: si fa una nuova normalizzazione e si prende il punto con la minima distanza da quello utopico. La configurazione ottimale, allora, risulta essere la SCR-1, come rappresentata in FIGURA 6.3, nelle condizioni schematizzate in TABELLA 6.7.

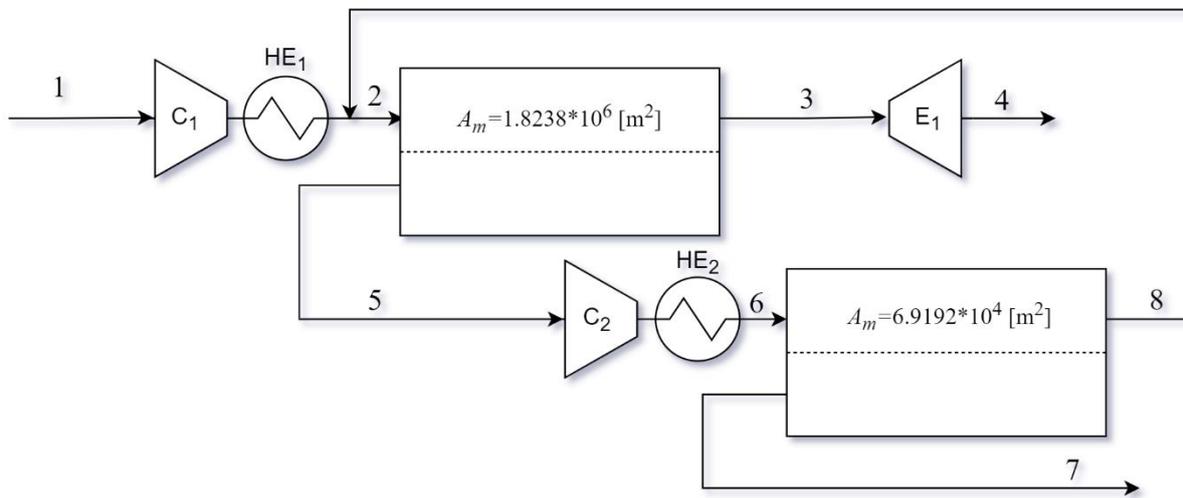


Figura 6.3 Schema ottimale di processo. Due stadi di separazione in cascata con ricircolo del retentato del secondo stadio. Entrambi gli stadi operano ad alta pressione. Le caratteristiche delle correnti e le potenze di compressori, turbine e scambiatori sono riassunti in TABELLA 6.7.

Tabella 6.7 Parametri del processo ottimizzato
(configurazione SCR-1 per la separazione della CO₂ nelle condizioni di TABELLA 6.1)

corrente	Q [cm _{STP} ³ /s]	x_{CO_2}	p [bar]
1	$5.83 \cdot 10^8$	0.15	1.01
2	$7.67 \cdot 10^8$	0.195	5.68
3	$5.01 \cdot 10^8$	0.018	5.68
4	$5.01 \cdot 10^8$	0.018	1.01
5	$2.65 \cdot 10^8$	0.531	1.01
6	$2.65 \cdot 10^8$	0.531	5.68
7	$8.20 \cdot 10^7$	0.960	1.01
8	$1.83 \cdot 10^8$	0.340	5.68
apparecchiatura	E [MW]	T_{in} [°C]	T_{out} [°C]
C₁	174	40	219
HE₁	174	219	40
E₁	-67.9	40	-
C₂	76.5	40	188
HE₂	76.5	188	40

6.2 Separazione della CO₂ da un biogas

Si vuole progettare uno schema di separazione con membrana per l'*upgrading* del biogas a biometano. Il biogas è prodotto da un processo di digestione anaerobica e sottoposto a pretrattamenti per la rimozione delle impurità, prima che venga messo a contatto con la membrana. L'alimentazione viene quindi trattata come una miscela binaria CH₄-CO₂. Le caratteristiche di alimentazione, le condizioni operative, le proprietà della membrana e le specifiche del prodotto sono raccolte in TABELLA 6.8 [48]. Viene utilizzata una membrana polimerica (Ultem 1000) selettiva per la CO₂. I parametri di qualità del biometano da rispettare, affinché possa essere convogliato in rete, sono quelli previsti dalla Regola Tecnica sulle caratteristiche chimico-fisiche del gas combustibile e sulla presenza di altri componenti (D.M. 18/05/2018). Non si definisce un recupero di metano, ma la sua perdita nel permeato viene conteggiata tra i consumi energetici in termini di potenza chimica perduta, per cui sarà minimizzata nella procedura di ottimizzazione.

Si può innanzitutto osservare come utilizzando un solo stadio di separazione, sia esso ad alta pressione o in condizioni di vuoto, la perdita di metano nel permeato costituisce un costo troppo alto (TABELLA 6.9). La potenza perduta con il metano permeato è di un'ordina di grandezza superiore rispetto alle potenze di compressione, costituendo una percentuale sostanziale dei consumi. I prossimi schemi dovranno necessariamente aumentare il recupero del metano nel retentato.

Tabella 6.8 Dati utilizzati nel caso studio per la separazione di una miscela CH₄-CO₂

Parametro	Valore	Unità di misura
Portata di <i>feed</i> (Q_f)	$1 \cdot 10^3$	$\text{m}_{\text{STP}}^3 \text{ h}^{-1}$
Pressione di <i>feed</i> (p_f)	1	bar
Temperatura (T)	30	°C
Frazione molare CO ₂ (x_f)	0.4	-
Purezza CH ₄ ($x_{r,CH4}$)	0.975	-
Permeanza (P_{eCO2})	86.30	GPU
Selettività CO ₂ /CH ₄ (α)	33.2	-

Tabella 6.9 Prestazioni di un solo stadio di separazione, in due condizioni di pressione. Recupero di metano, area specifica della membrana e consumo energetico specifico, di cui si riporta la percentuale di potenza chimica perduta

Singolo stadio	β	R_r [%]	a_m [$\text{m}^2/(\text{kg}_{\text{CH}_4} \text{ s}^{-1})$]	e [$\text{MW}/(\text{kg}_{\text{CH}_4} \text{ s}^{-1})$]	e_{ch}/e_{tot} [%]
Alta pressione	0.1	82.35	$1.8941 \cdot 10^4$	11.55	92.88
Bassa pressione	0.2	73.42	$3.4815 \cdot 10^5$	18.57	97.67

Il codice di ottimizzazione utilizzato è riportato in appendice G. Sono state eseguite cinque simulazioni, di cui si riporteranno i risultati migliori.

I due stadi in serie restituiscono ancora un consumo troppo alto, lo si può osservare dalle soluzioni di compromesso in TABELLA 6.10.

Tabella 6.10 Condizioni di minimo per le varianti dello schema con due stadi in serie. Area specifica della membrana e consumo energetico specifico totale.

Schema	a_m [$\text{m}^2/(\text{kg}_{\text{CH}_4} \text{ s}^{-1})$]	e [$\text{MW}/(\text{kg}_{\text{CH}_4} \text{ s}^{-1})$]
SS-1	$1.8945 \cdot 10^4$	11.52
SS-2	$1.0659 \cdot 10^4$	7.24
SS-3	$1.9075 \cdot 10^4$	11.56
SS-4	$3.4817 \cdot 10^5$	18.57

Per le quattro varianti dello schema con due stadi di separazione in cascata, è evidente già osservando i fronti di Pareto (FIGURA 6.4) come tra tutti lo schema SC-1 offra parametri di progetto migliori. Gli altri tre possono dunque essere scartati.

In FIGURA 6.5 sono rappresentati i fronti di Pareto per le quattro varianti dello schema con due stadi di separazione in serie, e l'aggiunta del ricircolo del permeato del secondo stadio. La configurazione SSR-2 permetterebbe di utilizzare meno membrana, mentre SSR-3 raggiunge valori minori in termini di consumo. Le quattro varianti verranno confrontate attraverso le rispettive soluzioni di compromesso.

Si riportano infine i fronti di Pareto per le quattro varianti dello schema con due stadi di separazione in cascata, e l'aggiunta del ricircolo del retentato del secondo stadio (FIGURA 6.6). dal grafico si può già escludere la variante SCR-3, le cui soluzioni sono tutte dominate da quelle delle altre configurazioni. La variante SCR-4 raggiunge i più bassi valori di consumo energetico, ma con un utilizzo maggiore di superficie della membrana. Uno schema più compatto si potrebbe ottenere con la configurazione SCR-1. Anche in questo caso si calcoleranno le soluzioni di compromesso in modo da poter confrontare meglio i quattro sistemi.

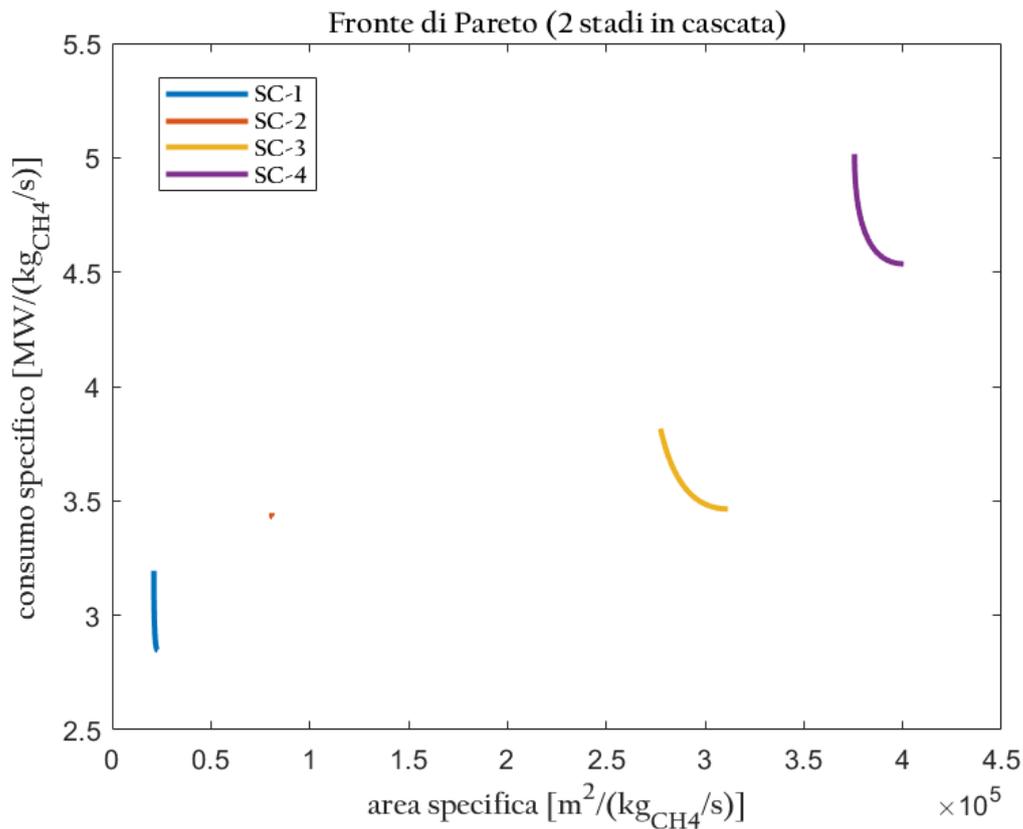


Figura 6.4 Fronti di Pareto per le quattro varianti dello schema con due stadi di separazione in cascata. Dati da TABELLA 6.8

Per completezza di informazioni, si riportano in TABELLA 6.11 i valori degli estremi delle curve di Pareto che si vogliono confrontare, con i rispettivi valori delle variabili decisionali. Area e consumi in TABELLA 6.11 sono le coordinate del punto di utopia di ciascuna funzione obiettivo utilizzate nel calcolo delle soluzioni di compromesso, raccolte in TABELLA 6.12. qui sono state evidenziate in grassetto le soluzioni che risultano essere non dominate in questo nuovo insieme di punti, da cui si è calcolata la condizione di minimo generale. La configurazione ottimale risulta essere, anche in questo caso, la SCR-1, rappresentata in FIGURA 6.7, nelle condizioni di processo riassunte in TABELLA 6.13.

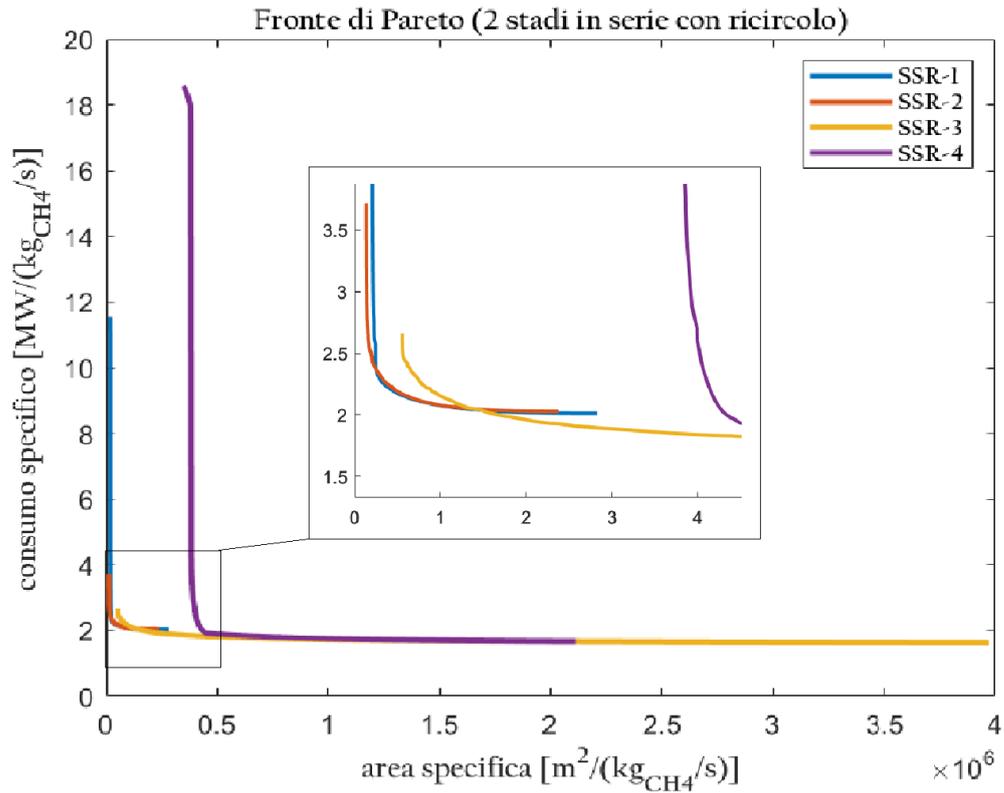


Figura 6.5 Fronti di Pareto per le quattro varianti dello schema con due stadi di separazione in serie (e il ricircolo del permeato del secondo stadio). Dati da TABELLA 6.8

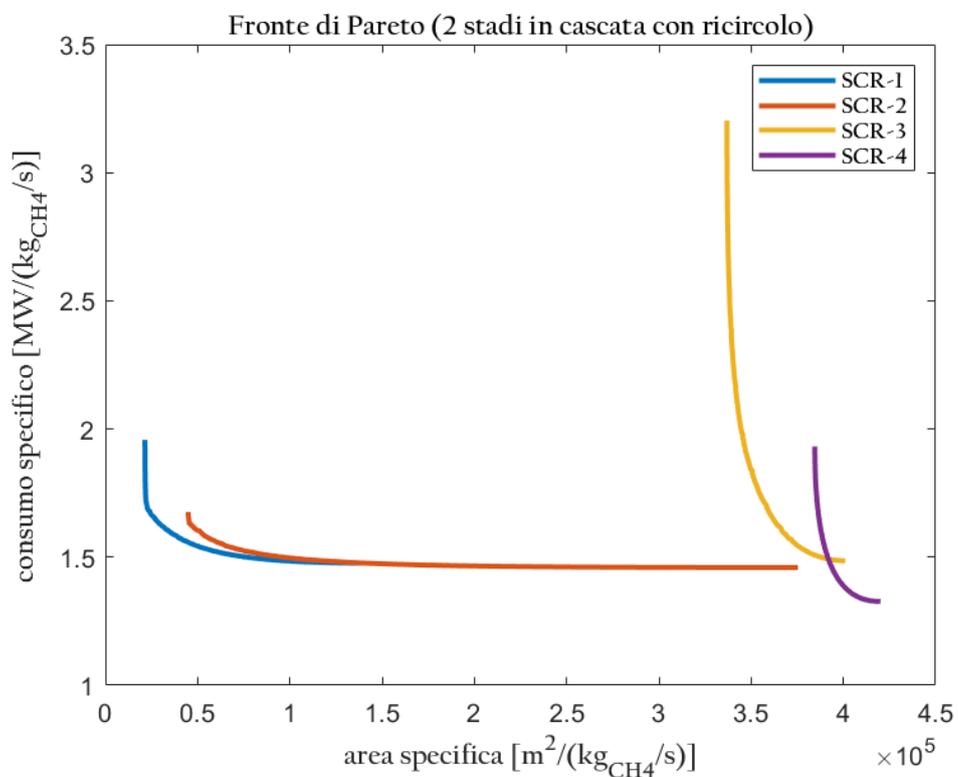


Figura 6.6 Fronti di Pareto per le quattro varianti dello schema con due stadi di separazione in cascata (e il ricircolo del retentato del secondo stadio). Dati da TABELLA 6.8

Tabella 6.11 Estremi delle curve di Pareto che vogliono essere confrontate, con i corrispondenti valori delle variabili decisionali

schema	a_m [m ² /(kg _{CH4} /s)]	e [MW/(kg _{CH4} /s)]	variabile 1	variabile 2	variabile 3
SC-1	2.1138·10 ⁴	3.195	$p_{h1}=10$ [bar]	$x_{r1}=0.025$	$p_{h2}=10$ [bar]
	2.3033·10 ⁴	2.847	$p_{h1}=10$ [bar]	$x_{r1}=0.014$	$p_{h2}=10$ [bar]
SSR-1	1.8947·10 ⁴	11.55	$p_{h1}=10$ [bar]	$x_{r1}=0.025$	$p_{h2}=10$ [bar]
	2.8264·10 ⁵	2.011	$p_{h1}=3.11$ [bar]	$x_{r1}=0.677$	$p_{h2}=3.11$ [bar]
SSR-2	1.3313·10 ⁴	3.717	$p_{h1}=10$ [bar]	$x_{r1}=0.219$	$p_{l2}=0.2$ [bar]
	2.3733·10 ⁵	2.028	$p_{h1}=3.14$ [bar]	$x_{r1}=0.671$	$p_{l2}=0.90$ [bar]
SSR-3	5.5245·10 ⁴	2.663	$p_{l1}=0.2$ [bar]	$x_{r1}=0.683$	$p_{h2}=10$ [bar]
	3.9683·10 ⁶	1.643	$p_{l1}=0.2$ [bar]	$x_{r1}=0.686$	$p_{h2}=1.5$ [bar]
SSR-4	3.4834·10 ⁵	18.56	$p_{l1}=0.2$ [bar]	$x_{r1}=0.025$	$p_{l2}=0.2$ [bar]
	2.1104·10 ⁶	1.678	$p_{l1}=0.2$ [bar]	$x_{r1}=0.667$	$p_{l2}=0.5$ [bar]
SCR-1	2.1467·10 ⁴	1.957	$p_{h1}=10$ [bar]	$p_{h2}=10$ [bar]	$x_{r2}=0.305$
	1.3986·10 ⁵	1.477	$p_{h1}=3.98$ [bar]	$p_{h2}=3.98$ [bar]	$x_{r2}=0.718$
SCR-2	4.4918·10 ⁴	1.676	$p_{h1}=10$ [bar]	$p_{l2}=0.2$ [bar]	$x_{r2}=0.763$
	3.7534·10 ⁵	1.459	$p_{h1}=2.75$ [bar]	$p_{l2}=0.20$ [bar]	$x_{r2}=0.712$
SCR-4	3.8453·10 ⁵	1.932	$p_{l1}=0.2$ [bar]	$p_{l2}=0.2$ [bar]	$x_{r2}=0.343$
	4.2013·10 ⁵	1.327	$p_{l1}=0.2$ [bar]	$p_{l2}=0.22$ [bar]	$x_{r2}=0.704$

Tabella 6.12 Punti ottimali di Pareto corrispondenti alle soluzioni di compromesso, per le configurazioni che si vogliono confrontare

Schema	a_m [m ² /(kg _{CH4} s ⁻¹)]	e [MW/(kg _{CH4} s ⁻¹)]
SC-1	2.1611·10 ⁴	2.930
SSR-1	2.6766·10 ⁴	2.317
SSR-2	3.8387·10 ⁴	2.247
SSR-3	5.5665·10 ⁵	1.797
SSR-4	4.1895·10 ⁵	2.134
SCR-1	4.1714·10 ⁴	1.568
SCR-2	1.0308·10 ⁵	1.494
SCR-4	3.9302·10 ⁵	1.471

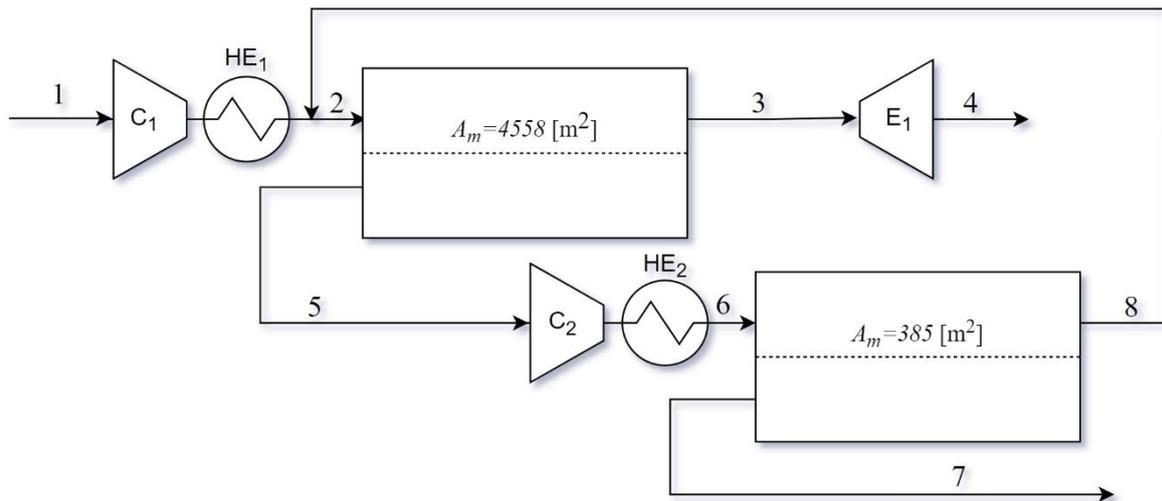


Figura 6.7 Configurazione ottimale per il caso studio descritto in TABELLA 6.8. Due stadi in cascata con ricircolo del retentato del secondo stadio. Entrambi gli stadi ad alta pressione. Le caratteristiche delle correnti e le potenze di compressori, turbine e scambiatori sono riassunti in TABELLA 6.13.

Tabella 6.13 Parametri del processo ottimizzato (configurazione SCR-1 per la separazione della CO₂ nelle condizioni di TABELLA 6.8)

corrente	Q [cm _{STP} ³ /s]	x_{CO_2}	p [bar]
1	$2.78 \cdot 10^5$	0.4	1
2	$4.17 \cdot 10^5$	0.488	7.06
3	$1.70 \cdot 10^5$	0.025	7.06
4	$1.70 \cdot 10^5$	0.025	1
5	$2.47 \cdot 10^5$	0.806	1
6	$2.47 \cdot 10^5$	0.806	7.07
7	$1.08 \cdot 10^5$	0.988	1
8	$1.39 \cdot 10^5$	0.665	7.07
apparecchiatura	E [MW]	T_{in} [°C]	T_{out} [°C]
C ₁	0.087	30	165
HE ₁	0.670	165	30
E ₁	-0.027	30	-
C ₂	0.077	30	166
HE ₂	0.610	166	30
Perdita CH ₄	0.048 [MW]	-	-
Recupero CH ₄	99.20 [%]	-	-

7. Conclusioni

Il presente lavoro di tesi ha voluto sviluppare un codice per l'ottimizzazione di un sistema di separazione a due stadi con membrana, quando ad essere alimentata è una miscela gassosa binaria. Si sono scelti due parametri di progetto da voler minimizzare: la superficie della membrana da installare, che è un'indicazione dell'ingombro dell'impianto e può essere rappresentativa dei costi capitali del processo; i consumi energetici in termini di potenza utilizzata dal sistema di compressione e di potenza chimica perduta con il prodotto non recuperato, indicativi dei costi operativi. Si è dimostrata la necessità di adottare un sistema a più stadi, nonostante questo richieda un maggior investimento per la sezione di separazione. Il principale limite della tecnologia a membrana è infatti quello di possedere una bassa efficienza di cattura della CO₂ rispetto alle altre tecnologie, come l'assorbimento e la separazione criogenica. Nei due casi studio esaminati si dimostra che con l'aggiunta di un altro stadio è possibile raggiungere le specifiche di prodotto richieste dalla normativa. In particolare, l'aggiunta di una corrente di ricircolo nel sistema incrementa notevolmente le caratteristiche del prodotto, sia in termini di recupero che di purezza.

Non è sufficiente aumentare l'efficienza di cattura della CO₂ per far sì che la tecnologia a membrana risulti competitiva rispetto a quelle più largamente utilizzate. Il processo deve anche risultare conveniente da un punto di vista economico. Questo è il motivo che rende necessaria un'ottimizzazione del sistema, in modo da trovare le condizioni operative e i parametri di processo in grado di minimizzare l'impatto della sezione di separazione sull'impianto.

Sono stati esaminati quattro schemi con due stadi di separazione, in innumerevoli condizioni diverse. L'ottimizzazione a due obiettivi è stata svolta con la funzione `gamultiobj` di MATLAB®, che implementa una variante dell'NSGA-II, appartenente alla famiglia degli algoritmi genetici. L'ottimizzazione multiobiettivo restituisce una curva di Pareto per ciascuno schema, rappresentante tutte le possibili condizioni ottimali di progetto. Ogni curva è caratterizzata da una soluzione ideale, in cui sia l'area della membrana che i consumi sono nella loro condizione di minimo, che però è irraggiungibile. Allora per ogni curva è stato scelto il punto che più si avvicinava alla soluzione ideale. Tra tutti, la migliore condizione di minimo viene scelta e ne determina il tipo di schema e le condizioni di progetto ottimali.

Sono stati esaminati due casi studio, rappresentanti le miscele binarie N₂-CO₂ e CH₄-CO₂: la separazione dell'anidride carbonica da un gas di scarico di una centrale termoelettrica a carbone, e l'*upgrading* a biometano del biogas prodotto da un impianto di digestione anaerobica. Per entrambi i casi, la procedura di ottimizzazione e di scelta del punto ottimale ha portato come schema migliore quello in cui i due stadi di separazione sono disposti in cascata, il retentato del secondo stadio viene ricircolato al primo, ed entrambi gli stadi sono in condizioni di alta pressione. L'obiettivo dell'ottimizzazione è stato raggiunto. Per i casi considerati, il sistema necessita di un'area di membrana dell'ordine di 10⁴ [m²/(kg_{prodotto}/s)]. Ulteriori considerazioni dovranno portare alla scelta della tipologia di modulo in cui quest'area andrà organizzata. Per quanto riguarda i consumi, si è raggiunta una richiesta di meno di 1.5 [MW/(kg_{prodotto}/s)]. Questo si può considerare un ottimo risultato, se si tiene conto dei range di richiesta energetica per una separazione criogenica o un processo di cattura con assorbimento: 6-10 [MW/(kg_{prodotto}/s)] e 4-6 [MW/(kg_{prodotto}/s)], rispettivamente [49].

Ovviamente, a un'ottimizzazione di questo tipo può essere integrata in un secondo momento un'analisi economica più approfondita, in modo da tener conto di tutti i costi capitali dell'impianto (compresa l'unità di separazione, compressori, pompe a vuoto, turbine, scambiatori), i costi operativi delle utenze, i costi di manutenzione e così via.

L'analisi effettuata è da considerarsi un primo step per una fase di progettazione. Per la modellazione dello stadio a membrana sono state fatte alcune ipotesi di idealità, tra cui l'invariabilità di permeabilità e selettività con la pressione, la resistenza meccanica della membrana in tutti i *range* di pressione esaminati, l'isotermicità della separazione e soprattutto non sono state considerate le perdite di carico. Queste possono non essere trascurabili in moduli come quello a fibra cava. Una volta scelto il modulo, potranno essere effettuate eventuali modifiche al modello che tengano conto della non idealità del sistema.

Lista dei simboli

A_m	superficie della membrana	$[m^2]$
a_m	superficie specifica della membrana	$[m^2/(kg/s)]$
b	affinità di foro	$[Pa^{-1}]$
C_p	capacità termica a pressione costante	$[J/K]$
c	concentrazione	$[m_{STP}^3/m^3]$
c'_h	costante di saturazione	$[m_{STP}^3/m^3]$
c_p	calore specifico a pressione costante	$[J/(K \cdot mol)]$
c_v	calore specifico a volume costante	$[J/(K \cdot mol)]$
D	coefficiente di diffusività	$[m^2/s]$
D_0	coefficiente pre-esponenziale nella relazione $D(T)$	$[m^2/s]$
D_k	coefficiente di Knudsen	$[m^2/s]$
D_T	coefficiente di diffusività termodinamico	$[m^2/s]$
E	potenza spesa/guadagnata	$[W]$
E_d	energia di attivazione per la diffusione	$[J/mol]$
E_p	energia di attivazione per la permeazione	$[J/mol]$
e	consumo specifico	$[W/(kg/s)]$
H	entalpia	$[J]$
H_s	entalpia di soluzione	$[J/mol]$
J	flusso di gas attraverso la membrana	$[m_{STP}^3/(m^2s)]$
J_k	flusso di Knudsen	$[mol/s]$
k_d	costante di Henry	$[m_{STP}^3/(m^3Pa)]$
LHV	potere calorifico inferiore	$[J/kg]$
l	spessore della membrana	$[m]$
l_0	spessore del <i>toplayer</i> della membrana asimmetrica	$[m]$
l_A	distanza tra il punto di dissoluzione e l'ingresso del poro	$[m]$
l_{eff}	spessore effettivo della membrana	$[m]$
M_w	peso molecolare	$[kg/mol]$
n	numero di pori	$[-]$
P	coefficiente di permeabilità	$[m_{STP}^3m/(m^2sPa)]$
P_0	coefficiente pre-esponenziale nella relazione $P(T)$	$[m_{STP}^3m/(m^2sPa)]$
Pe	permeanza	$[m_{STP}^3/(m^2sPa)]$
p	pressione	$[Pa]$

p_h	pressione di <i>upstream</i>	[Pa]
$p_{h,i}$	pressione parziale di <i>upstream</i> del componente i	[Pa]
p_l	pressione di <i>downstream</i>	[Pa]
$p_{l,i}$	pressione parziale di <i>downstream</i> del componente i	[Pa]
Q	portata volumica	[m _{STP} ³ /s]
R	costante dei gas ideali	[J/(K mol)]
R^2	coefficiente di determinazione	[-]
R_p	recupero di A nel permeato	[-]
r	raggio dei pori	[m]
S	entropia	[J]
s	coefficiente di solubilità	[m _{STP} ³ /(m ³ Pa)]
s_0	coefficiente pre-esponenziale nella relazione $s(T)$	[m _{STP} ³ /(m ³ Pa)]
T	temperatura	[K]
T_c	temperatura critica	[K]
T_g	temperatura di transizione vetrosa	[K]
t	tempo	[s]
U	energia interna	[J]
V	volume di gas	[m ³]
V_0	volume occupato dalle molecole della membrana a 0 [K]	[m ³]
V_f	volume libero della membrana densa	[m ³]
V_m	volume molare	[m _{STP} ³ /mol]
V_T	volume occupato dalle molecole della membrana alla temperatura T	[m ³]
v_f	volume libero frazionario	[-]
X	coordinata spaziale perpendicolare alla superficie della membrana	[m]
x	frazione molare lato <i>feed</i>	[-]
y	frazione molare lato permeato	[-]

Simboli greci

α	selettività	[-]
β	rapporto tra bassa e alta pressione ai lati della membrana	[-]
γ	fattore di espansione adiabatica	[-]
ε	porosità superficiale della membrana	[-]
ε_r	errore relativo	[-]
η	efficienza	[-]

λ	coefficiente di espansione termica	[K ⁻¹]
μ_{JT}	coefficiente di Joule-Thomson	[K/Pa]
ν	rapporto di compressione	[-]
ρ	densità	[kg/m _{STP} ³]
τ	tortuosità	[-]

Pedici e apici

1	riferito al primo stadio
2	riferito al secondo stadio
A	riferito al componente A, più permeante
B	riferito al componente B
c	del compressore
ch	chimica
ex	dell'espansore
f	riferito alla corrente di alimentazione
h	lato della membrana ad alta pressione (<i>upstream</i>)
he	dello scambiatore di calore
i	componente i-esimo
in	condizioni di ingresso
j	componente j meno permeabile
k	meccanismo di Knudsen
l	lato della membrana a bassa pressione (<i>downstream</i>)
mix	riferito alla miscela
out	condizioni di uscita
p	riferito al permeato
pv	della pompa a vuoto
r	riferito al retentato
t	totale

Riferimenti bibliografici

- [1] M. Pathak, R. Slade, P.R. Shukla, J. Skea, R. Pichs-Madruga, D. Ürge-Vorsatz, 2022: Technical Summary. In: *Climate Change 2022: Mitigation of Climate Change. Contribution of Working Group III to the Sixth Assessment Report of the Intergovernmental Panel on Climate Change* [P.R. Shukla, J. Skea, R. Slade, A. Al Khouradajie, R. van Diemen, D. McCollum, M. Pathak, S. Some, P. Vyas, R. Fradera, M. Belkacemi, A. Hasija, G. Lisboa, S. Luz, J. Malley, (eds.)]. Cambridge University Press, Cambridge, UK and New York, NY, USA. [DOI: 10.1017/9781009157926.002.]
- [2] IEA (2022), *World Energy Outlook 2022*. OECD Publishing, Paris. [DOI: 10.1787/3a469970-en.]
- [3] Global CCS Institute, 2021. *The Global Status of CCS: 2021*. Australia.
- [4] Cuéllar-Franca R.M. & Azapagic A., 2015. Carbon capture, storage and utilisation technologies: A critical analysis and comparison of their life cycle environmental impacts. *Journal of CO2 Utilization* **9**, 82-102. [DOI: 10.1016/j.jcou.2014.12.001.]
- [5] Gür T.M., 2022. Carbon Dioxide Emissions, Capture, Storage and Utilization: Review of Materials, Processes and Technologies. *Progress in Energy and Combustion Science* **89**, 100965. [DOI: 10.1016/j.pecs.2021.100965.]
- [6] Kemper J., 2015. Biomass and carbon dioxide capture and storage: A review. *International Journal of Greenhouse Gas Control* **40**, 401-430. [DOI: 10.1016/j.ijggc.2015.06.012.]
- [7] Bui M. et al., 2018. Carbon capture and storage (CCS): the way forward. *Energy Environ. Sci.* **11**, 1062. [DOI: 10.1039/c7ee02342a.]
- [8] Bambokela J. E., Muzenda E. & Belaid M., 2018. Prospective Synergy of Biogas Upgrading Technologies with Carbon Capture and Sequestration (CCS) Techniques. In *Proceedings of the International Conference on Industrial Engineering and Operations Management*. Vol. 2018, No. JUL, pp. 3236-3249.
- [9] Olajire A.A., 2010. CO2 capture and separation technologies for end-of-pipe applications—A review. *Energy* **35**(6), 2610-2628. [DOI: 10.1016/j.energy.2010.02.030.]
- [10] Araújo O. D. Q. F. & de Medeiros J. L., 2017. Carbon capture and storage technologies: present scenario and drivers of innovation. *Current Opinion in Chemical Engineering* **17**, 22-34. [DOI: 10.1016/j.coche.2017.05.004.]
- [11] Mulder M., 1996. *Basic Principles of Membrane Technology*. 2nd Edition, Kluwer Academic Publishers, Dordrecht. [DOI: 10.1007/978-94-009-1766-8.]
- [12] Van Amerongen G. J., 1946. The permeability of different rubbers to gases and its relation to diffusivity and solubility. *Journal of Applied Physics* **17**(11), 972-985. [DOI: 10.1063/1.1707667.]
- [13] Matteucci S., Yampolskii Y., Freeman B. D. & Pinnau, I., 2006. Transport of gases and vapors in glassy and rubbery polymers. *Materials science of membranes for gas and vapor separation*, 1-47.
- [14] Monteleone M., Mobili R., Milanese C., Esposito E., Fuoco A., La Cognata S., Amendola V. & Jansen J.C., 2021. PEEK–WC-Based Mixed Matrix Membranes Containing

- Polyimine Cages for Gas Separation. *Molecules* **26**, 5557. [DOI: 10.3390/molecules26185557.]
- [15] Russo F., Galiano F., Iulianelli A., Basile A. & Figoli A., 2021. Biopolymers for sustainable membranes in CO₂ separation: A review. *Fuel Processing Technology* **213**, 106643. [DOI: 10.1016/j.fuproc.2020.106643.]
- [16] Hinchliffe A. B. & Porter K. E., 1997. Gas separation using membranes. 1. Optimization of the separation process using new cost parameters. *Industrial & engineering chemistry research* **36**(3), 821-829. [DOI: 10.1021/ie9603272.]
- [17] Geankoplis C. J., 1993. Membrane Separation Processes, in “*Transport Processes and Unit Operations*”, Chap. 13. 3rd Edition, Prentice-Hall international, Inc, Englewood Cliffs, New Jersey, pp. 754-782. [ISBN 0-13-045253-X]
- [18] Weller S. & Steiner W. A., 1950. Separation of Gases by Fractional Permeation through Membranes. *Journal of Applied Physics* **21**, 279-283. [DOI: 10.1063/1.1699653.]
- [19] Walawender W. P. & Stern S. A., 1972. Analysis of Membrane Separation Parameters. II. Counter-current and Cocurrent Flow in a Single Permeation Stage. *Separation Science* **7**(5), 553-584. [DOI: 10.1080/00372367208056054.]
- [20] Tranchino L., Santarossa R., Carta F., Fabiani C. & Bimbi L., 1989. Gas Separation in a Membrane Unit: Experimental Results and Theoretical Predictions. *Separation Science and Technology* **24**(14), 1207-1226. [DOI: 10.1080/01496398908049898.]
- [21] Feng X., Ivory J. & Rajan V. S., 1999. Air separation by integrally asymmetric hollow-fiber membranes. *AIChE journal* **45**(10), 2142-2152. [DOI: 10.1002/aic.690451013.]
- [22] Sidhoum M., Sengupta A. & Sirkar K. K., 1988. Asymmetric cellulose acetate hollow fibers: studies in gas permeation. *AIChE journal* **34**(3), 417-425. [DOI: 10.1002/aic.690340309.]
- [23] Spillman R., 1995. Economics of gas separation membrane processes, in “*Membrane Science and Technology*”, Chap13, Vol. 2, Elsevier, pp. 589-667. [DOI: 10.1016/S0927-5193(06)80015-X.]
- [24] Yang D., Wang Z., Wang J. & Wang S., 2009. Potential of two-stage membrane system with recycle stream for CO₂ capture from postcombustion gas. *Energy & fuels* **23**(10), 4755-4762. [DOI: 10.1021/ef801109p.]
- [25] Yuan M., Narakornpijit K., Haghpanah R. & Wilcox J., 2014. Consideration of a nitrogen-selective membrane for postcombustion carbon capture through process modeling and optimization. *Journal of membrane science* **465**, 177-184. [DOI: 10.1016/j.memsci.2014.04.026.]
- [26] Koretsky M.D, 2004. *Engineering and Chemical Thermodynamics*. Wiley internat. ed. Hoboken NJ: Wiley, pp 641-642. [ISBN 978-0-470-25961-0]
- [27] Engineering ToolBox, 2003. *Fuels - Higher and Lower Calorific Values*. [online] Available at: https://www.engineeringtoolbox.com/fuels-higher-calorific-values-d_169.html [Accessed 22 Feb. 2023].
- [28] Engineering ToolBox, 2008. Methane - Thermophysical Properties. [online] Available at: https://www.engineeringtoolbox.com/methane-d_1420.html [Accessed 22 Feb. 2022].
- [29] Gabrielli P., Gazzani M. & Mazzotti M., 2017. On the optimal design of membrane-based gas separation processes. *Journal of Membrane Science* **526**, 118-130. [DOI: 10.1016/j.memsci.2016.11.022.]

- [30] Chang H. & Hou W. C., 2006). Optimization of membrane gas separation systems using genetic algorithm. *Chemical Engineering Science* 61(16), 5355-5368. [DOI: 10.1016/j.ces.2006.04.020.]
- [31] Ramírez-Santos Á. A., Bozorg M., Addis B., Piccialli V., Castel C. & Favre E., 2018. Optimization of multistage membrane gas separation processes. Example of application to CO₂ capture from blast furnace gas. *Journal of Membrane Science* 566, 346-366. [DOI: 10.1016/j.memsci.2018.08.024.]
- [32] Deb K., 2001. *Multi-Objective Optimization using Evolutionary Algorithms*. John Wiley & Sons, Ltd, Chichester, England. [ISBN 0-471-87339-X]
- [33] Chang K.H., 2015. Chapter 3 - Design Optimization, in “*Design Theory and Methods Using CAD/CAE*”, Kuang-Hua Chang, Academic Press, pp. 103-210. [DOI: 10.1016/B978-0-12-398512-5.00003-7.]
- [34] Chang K.H., 2015. Chapter 5 - Multiobjective Optimization and Advanced Topics, in “*Design Theory and Methods Using CAD/CAE*”, Kuang-Hua Chang, Academic Press, pp. 325-406. [DOI: 10.1016/B978-0-12-398512-5.00005-0.]
- [35] MathWorks, 2022. *gamultiobj* Algorithm. [online] Available at: https://it.mathworks.com/help/gads/gamultiobj-algorithm.html#mw_5520b9a9-5617-4b41-9abf-9a2aae8ab779 [Accessed 25 Feb. 2023].
- [36] Guariso G. & Sangiorgio M., 2020. Improving the performance of multiobjective genetic algorithms: An elitism-based approach. *Information* 11(12), 587. [DOI: 10.3390/info11120587.]
- [37] Kennedy J. & R. Eberhart, 1995. Particle Swarm Optimization. *Proceedings of the IEEE International Conference on Neural Networks*. Perth, Australia, pp. 1942–1945. [DOI: 10.1109/ICNN.1995.488968.]
- [38] Mezura-Montes E. & C. A. Coello Coello, 2011. Constraint-handling in nature-inspired numerical optimization: Past, present and future, in “*Swarm and Evolutionary Computation*”, Volume 1, Issue 4, pp. 173–194. [DOI: 10.1016/j.swevo.2011.10.001.]
- [39] Pedersen M. E., 2010. Good Parameters for Particle Swarm Optimization. Luxembourg: Hvass Laboratories.
- [40] MathWorks, 2022. Particle Swarm Optimization Algorithm. [online] Available at: <https://it.mathworks.com/help/gads/particle-swarm-optimization-algorithm.html> [Accessed 25 Feb. 2023].
- [41] Panarese P., 2022. Approximation Techniques with MATLAB®. *Dolomites Research Notes on Approximation* 15(2). [DOI: 10.14658/pupj-drna-2022-2-10.]
- [42] MathWorks, 2022. Surrogate Optimization Algorithm. [online] Available at: https://it.mathworks.com/help/gads/surrogate-optimization-algorithm.html#mw_fd4feaed-717f-4e95-96f3-53890110908d [Accessed 26 Feb. 2023].
- [43] MathWorks, 2022. What is Surrogate Optimization? [online] Available at: <https://it.mathworks.com/help/gads/what-is-surrogate-optimization.html> [Accessed 26 Feb. 2023].
- [44] MathWorks, 2022. Pattern Search Terminology [online] Available at: <https://it.mathworks.com/help/gads/pattern-search-terminology.html> [Accessed 26 Feb. 2023].

- [45] MathWorks, 2022. How Pattern Search Polling Works [online] Available at: <https://it.mathworks.com/help/gads/how-pattern-search-polling-works.html> [Accessed 26 Feb. 2023].
- [46] Dincer I. & Rosen M.A., 2021. Chapter 24 - Exergy and multiobjective optimization, in “*Exergy*”, 3rd Edition, Ibrahim Dincer, Marc A. Rosen, Elsevier, pp. 641-665. [DOI: 10.1016/B978-0-12-824372-5.00024-5.]
- [47] Marler R.T. & Arora J.S., 2004. Survey of multi-objective optimization methods for engineering. *Structural and multidisciplinary optimization* **26**, 369-395. [DOI: 10.1007/s00158-003-0368-6.]
- [48] Gilassi S., Taghavi S. M., Rodrigue D. & Kaliaguine S., 2019. Optimizing membrane module for biogas separation. *International Journal of Greenhouse Gas Control* **83**, 195-207. [DOI: 10.1016/j.ijggc.2019.02.010.]
- [49] Brunetti A., Scura F., Barbieri G. & Drioli E., 2010. Membrane technologies for CO₂ separation. *Journal of Membrane Science* **359**(1-2), 115-125. [DOI: 10.1016/j.memsci.2009.11.040.]

Appendice A

A.1 Modello di completa miscelazione, caso 1, implementato in MATLAB®

```
function [y_p,Q_p,x_r,Q_r,A] =  
perfectmixmodel(p_high,p_low,permA,permB,Q_f,x_f,R,l)  
%Modello di perfetta miscelazione per uno stadio di separazione; miscela binaria.  
%x_f è riferito al componente A, più permeante. R è riferito a A nel permeato.  
beta=p_low/p_high;  
a=beta*(permB-permA);  
b=permB*(1-x_f+R*x_f-beta-beta*R*x_f)+permA*(beta+x_f-R*x_f+beta*R*x_f);  
c=permB*(-R*x_f+beta*R*x_f)-permA*(x_f-R*x_f+beta*R*x_f);  
y_p=(-b+sqrt(b^2-4*a*c))/(2*a); %frazione molare di A nel permeato  
x_r=y_p*(x_f-R*x_f)/(y_p-R*x_f); %frazione molare di A nel retentato  
Q_p=R*x_f*Q_f/y_p; %portata di permeato  
Q_r=Q_f-Q_p; %portata di retentato  
A=R*x_f*Q_f*l/(permA*(p_high*x_r-p_low*y_p)); %superficie della membrana  
end
```

A.2 Modello di completa miscelazione, caso 2, implementato in MATLAB®

```
function [y_p,Q_p,x_r,Q_r,A] =  
perfectmixmodel(p_high,p_low,permA,permB,Q_f,x_f,R,l)  
%Modello di perfetta miscelazione per uno stadio di separazione; miscela binaria.  
%x_f è riferito a B, il componente meno permeante. R è riferito a B nel retentato.  
beta=p_low/p_high;  
a=beta*(permA-permB);  
b=permA*(1-R*x_f-beta-beta*x_f+beta*R*x_f)+permB*(beta+R*x_f+beta*x_f-  
beta*R*x_f);  
c=permA*x_f*(R+beta-1-beta*R)-permB*x_f*(R+beta-beta*R);  
y_p=(-b+sqrt(b^2-4*a*c))/(2*a); %frazione molare di B nel permeato  
x_r=y_p*R*x_f/(y_p-(1-R)*x_f); %frazione molare di B nel retentato  
Q_p=(1-R)*x_f*Q_f/y_p; %portata di permeato  
Q_r=Q_f-Q_p; %portata di retentato  
A=(1-R)*x_f*Q_f*l/(permB*(p_high*x_r-p_low*y_p)); %superficie della membrana  
end
```

Appendice B

B.1 Modello *cross-flow*, caso 1, implementato in MATLAB®

```
function [y_p,Q_p,x_r,Q_r,A] =
crossflowmodel(p_high,p_low,permA,permB,Q_f,x_f,R,l)
%Modello cross-flow per uno stadio di separazione; miscela binaria.
%x_f è riferito al componente A, più permeante; R è il recupero di A nel permeato.
alpha=permA/permB;
%scrittura dell'equazione da risolvere
D=0.5*((1-alpha)*p_low/p_high+alpha);
F=-0.5*((1-alpha)*p_low/p_high-1);
E=alpha/2-D*F;
G=1/(2*D-1);
S=(alpha*(D-1)+F)/((2*D-1)*(alpha/2-F));
T=1/(1-D-(E/F));
i=@(x) x/(1-x);
u=@(i) -D.*i+sqrt(D^2.*i.^2+2*E.*i+F^2);
i_f=i(x_f); %i calcolato alle condizioni id alimentazione
u_f=u(i_f); %u calcolato alle condizioni di alimentazione
eq=@(i) x_f*(1-R)/(i*(1-x_f))-((u_f-E/D)/(u(i)-E/D))^G*((u_f-alpha+F)/(u(i)-
alpha+F))^S*((u_f-F)/(u(i)-F))^T;
%valore iniziale per x
x0=0.1;
%risoluzione per x_r, y_p, Q_r, Q_p
[i_r,fval,exitflag]=fzero(eq,i(x0));
x_r=i_r/(i_r+1); %frazione di A nel retentato
Q_r=x_f*Q_f*(1-R)/x_r; %portata di retentato
Q_p=Q_f-Q_r; %portata di permeato
y_p=R*x_f*Q_f/Q_p; %frazione di A nel permeato
%calcolo dell'area totale
f_i=@(i) (D.*i-F)+sqrt(D^2.*i.^2+2*E.*i+F^2);
Rstar=@(i) 1-i.*(1-x_f)./x_f.*((u_f-E/D)/(u(i)-E/D)).^G.*((u_f-
alpha+F)/(u(i)-alpha+F)).^S.*((u_f-F)/(u(i)-F)).^T;

F_i=@(i) x_f.*(1-Rstar(i))./(i.*(f_i(i)-i).*(1./(1+i)-
p_low/p_high.*(1./(1+f_i(i)))));
int=integral(F_i,i_r,i_f); %calcolo dell'integrale
A=l*Q_f/(p_high*permB)*int; %area della membrana
end
```

B.2 Modello *cross-flow*, caso 2, implementato in MATLAB®

```
function [y_p,Q_p,x_r,Q_r,A] =
crossflowmodel(p_high,p_low,permA,permB,Q_f,x_f,R,l)
%Modello in cross-flow per uno stadio di separazione; miscela binaria.
%x_f è riferito al componente B, meno permeante; R è il recupero di B nel retentato.
alpha=permB/permA;
%scrittura dell'equazione da risolvere
D=0.5*((1-alpha)*p_low/p_high+alpha);
F=-0.5*((1-alpha)*p_low/p_high-1);
E=alpha/2-D*F;
G=1/(2*D-1);
S=(alpha*(D-1)+F)/((2*D-1)*(alpha/2-F));
T=1/(1-D-(E/F));
i=@(x) x/(1-x);
u=@(i) -D.*i+sqrt(D^2.*i.^2+2*E.*i+F^2);
i_f=i(x_f); %i alle condizioni di alimentazione
```

```

u_f=u(i_f); %u alle condizioni di alimentazione
eq=@(i) x_f*R/(i*(1-x_f))-((u_f-E/D)/(u(i)-E/D))^G*((u_f-alpha+F)/(u(i)-
    alpha+F))^S*((u_f-F)/(u(i)-F))^T;
%valore iniziale per x
x0=0.95;
%risoluzione per x_0, y_p, Q_0, Q_p
[i_r,fval,exitflag]=fzero(eq,i(x0));
x_r=i_r/(i_r+1); %frazione molare di B nel retentato
Q_r=x_f*Q_f*R/x_r; %portata di retentato
Q_p=Q_f-Q_r; %portata di permeato
y_p=(1-R)*x_f*Q_f/Q_p; %frazione molare di B nel permeato
%calcolo dell'area totale
f_i=@(i) (D.*i-F)+sqrt(D^2.*i.^2+2*E.*i+F^2);
Rstar=@(i) i.*(1-x_f)./x_f.*((u_f-E/D)/(u(i)-E/D)).^G.*((u_f-alpha+F)/(u(i)-
    alpha+F)).^S.*((u_f-F)/(u(i)-F)).^T;
F_i=@(i) x_f.*Rstar(i)/(i.*(f_i(i)-i).*(1./(1+i)-
    p_low/p_high.*(1./(1+f_i(i)))));
int=integral(F_i,i_r,i_f); %calcolo dell'integrale
A=1*Q_f/(p_high*permA)*int; %area della membrana
end

```


C.2 Modello in controcorrente, caso 2, implementato in MATLAB®

```
function [y_p,Q_p,Q_r,R,A] =
countercurrentmodel(p_high,p_low,permA,permB,Q_f,x_f,x_r,l)
%Modello in controcorrente per uno stadio di separazione; miscela binaria.
%x_f e x_r sono riferiti al componente A, più permeante.
    alpha=permA/permB;
    beta=p_high/p_low;
    N=2000;
    y_i=((alpha-1)*(beta*x_r+1)+beta-sqrt(((alpha-1)*(beta*x_r+1)+beta)^2-
        4*alpha*beta*x_r*(alpha-1)))/(2*(alpha-1));
    F=@(z,r) (r-x_r)/z*(((1-r(1))*alpha*(beta*z+beta*x_r-r(1))-r(1)*(beta*(1-z-
        x_r)-(1-r(1))))/((1-z-x_r)*alpha*(beta*z+beta*x_r-r(1))-
        (z+x_r)*(beta*(1-z-x_r)-(1-r(1)))));
    KdxDA0=alpha*(beta*x_r-y_i)*(x_r-y_i)*x_r/y_i;
    KdyDA0=KdxDA0*(x_r-y_i)*beta*(alpha-y_i*(alpha-1))/(KdxDA0/x_r-(x_r-
        y_i)*((alpha-1)*(2*y_i-beta*x_r-1)-beta));
    F0=@(z,r) KdyDA0/KdxDA0;
    [z,y]=RK4_system(F,F0,y_i,0,x_f-x_r,N);
    y_p=y(end); %frazione molare di A nel permeato
    if isnan(y_p)
        disp('I dati di input con cui si è verificato l'errore sono:')
        fprintf('%6.6e\t%6.6e\t%6.6e\t%6.6e\t%6.6e\t%6.6e\t%6.6e\t%6.6e',
            p_high,p_low,permA,permB,Q_f,x_f,x_r,l)
    end
    R=y_p*(x_r-x_f)/(x_f*(x_r-y_p)); %recupero di A nel permeato
    Q_p=R*x_f*Q_f/y_p; %portata di permeato
    Q_r=Q_f-Q_p; %portata di retentato
    R=(1-x_r)*Q_r/((1-x_f)*Q_f); %recupero di B nel retentato
    K=x_r*Q_r*1/p_low/permB;
    F=@(z,r) [(r(1)-x_r)/z*(((1-r(1))*alpha*(beta*z+beta*x_r-r(1))-r(1)*(beta*(1-
        z-x_r)-(1-r(1))))/((1-z-x_r)*alpha*(beta*z+beta*x_r-r(1))-
        (z+x_r)*(beta*(1-z-x_r)-(1-r(1)))));
        K*((r(1)-x_r)/(z+x_r-r(1)))/(x_r*((1-z-x_r)*alpha*(beta*z+beta*x_r-
        r(1))-(z+x_r)*(beta*(1-z-x_r)-(1-r(1)))));];
    F0=@(z,r) [KdyDA0/KdxDA0;
        K*((r(1)-x_r)/(z+x_r-r(1)))/(x_r*((1-z-x_r)*alpha*(beta*z+beta*x_r-
        r(1))-(z+x_r)*(beta*(1-z-x_r)-(1-r(1)))));];
    r0=[y_i; 0];
    [z,r]=RK4_system(F,F0,r0,0,x_f-x_r,N);
    A=-r(end,2); %area della membrana
end
```

C.3 Funzione implementata in MATLAB® per la risoluzione di un sistema di equazioni differenziali con il metodo di Runge-Kutta a 4 stadi

```
function [z,r]=RK4_system(f,f0,r0,z0,zN,N)
%Metodo di Runge-Kutta a 4 stadi per la risoluzione di un sistema di equazioni
differenziali con condizioni iniziali.

%f contiene le equazioni differenziali da risolvere dr/dz;
%f0 contiene le espressioni delle equazioni differenziali valutate in z=z0 (e
r=r0);
%r0 è il vettore contenente le condizioni iniziali;
%z0 e zN sono gli estremi dell'intervallo di integrazione;
%N è il numero di sottointervalli in cui verrà ripartito l'intervallo di
integrazione.
```

```

m=length(r0); %si definisce il numero di equazioni differenziali da risolvere
h=(zN-z0)/N; %passo della partizione
z=[z0:h:zN]'; %definizione della partizione
r=zeros(m,N+1); %inizializzazione della matrice soluzione r
r(:,1)=r0; %condizioni iniziali
%implementazione del metodo
for n=1:N
    if n<=20 %utilizzo f0 anche per z=z0 per non incorrere in errori di
        calcolo numerico
            k1=f0(z(n),r(:,n));
            k2=f0(z(n)+h/2,r(:,n)+h/2*k1);
            k3=f0(z(n)+h/2,r(:,n)+h/2*k2);
            k4=f0(z(n)+h,r(:,n)+h*k3);
            r(:,n+1)=r(:,n)+h/6*(k1+2*k2+2*k3+k4);
        else
            k1=f(z(n),r(:,n));
            k2=f(z(n)+h/2,r(:,n)+h/2*k1);
            k3=f(z(n)+h/2,r(:,n)+h/2*k2);
            k4=f(z(n)+h,r(:,n)+h*k3);
            r(:,n+1)=r(:,n)+h/6*(k1+2*k2+2*k3+k4);
        end
    end
    r=r';
end

```

Appendice D

D.1 Funzione implementata per l'ottimizzazione dello schema con due stadi in serie, quando il prodotto è il flusso permeato e la specifica di progetto è il recupero totale del componente A. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```
function [objfun] = duestadiserie(type,optvar,dati,I,E_u,E_g,Tout,Pt)
%si pone la pressione del lato della membrana noto a 1.01 [bar], come la
%pressione di feed.
p_f=1.01*75.0062; %[cmHg]
if type==1
    [y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),
                                                dati(3),dati(4),dati(5),optvar(2),dati(7));
    R2=(dati(4)*dati(5)*dati(6)-y_p1*Q_p1)/(x_r1*Q_r1);
    [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(3),p_f,dati(2),
                                                dati(3),Q_r1,x_r1,R2,dati(7));

    Q1=dati(4);
    y1=dati(5);
    Q2=Q_r1;
    y2=x_r1;
    Q3=Q_r;
    y3=x_r;
    v1=optvar(1)/dati(1);
    v2=optvar(3)/optvar(1);
    v3=p_f/optvar(3);
elseif type==2
    [y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),
                                                dati(3),dati(4),dati(5),optvar(2),dati(7));
    R2=(dati(4)*dati(5)*dati(6)-y_p1*Q_p1)/(x_r1*Q_r1);
    [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(1),optvar(3),dati(2),
                                                dati(3),Q_r1,x_r1,R2,dati(7));

    Q1=dati(4);
    y1=dati(5);
    Q2=Q_p2;
    y2=y_p2;
    Q3=Q_r;
    y3=x_r;
    v1=optvar(1)/dati(1);
    v2=p_f/optvar(3);
    v3=p_f/optvar(1);
elseif type==3
    [y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(dati(1),optvar(1),dati(2),
                                                dati(3),dati(4),dati(5),optvar(2),dati(7));
    R2=(dati(4)*dati(5)*dati(6)-y_p1*Q_p1)/(x_r1*Q_r1);
    [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(3),p_f,dati(2),
                                                dati(3),Q_r1,x_r1,R2,dati(7));

    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_r1;
    y2=x_r1;
    Q3=Q_r;
    y3=x_r;
    v1=p_f/optvar(1);
    v2=optvar(3)/dati(1);
```

```

v3=p_f/optvar(3);
elseif type==4
[y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(dati(1),optvar(1),dati(2),
dati(3),dati(4),dati(5),optvar(2),dati(7));
R2=(dati(4)*dati(5)*dati(6)-y_p1*Q_p1)/(x_r1*Q_r1);
[y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(dati(1),optvar(3),dati(2),
dati(3),Q_r1,x_r1,R2,dati(7));

Q1=Q_p1;
y1=y_p1;
Q2=Q_p2;
y2=y_p2;
Q3=0;
y3=0;
v1=p_f/optvar(1);
v2=p_f/optvar(2);
v3=1;
end
y_p=(y_p1*Q_p1+y_p2*Q_p2)/(Q_p1+Q_p2);
Q_p=Q_p1+Q_p2; %portata finale di permeato
if I==1
objfun=1-y_p;
elseif y_p<dati(8)
objfun=inf; %serve a escludere tutte le condizioni in cui la purezza minima
non è soddisfatta
elseif I==2
objfun=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4);%area specifica totale
[m^2/(kgCO2/s)]
elseif I==3
E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
E3=E_g(Q3,y3,v3); %energia prodotta dall'espansore, se presente
objfun=(E1+E2-E3)/(Q_p*y_p*0.04401/22414); %consumo energetico specifico
[MW/(kgCO2/s)]
elseif I==4
objfun(1)=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4); %[m^2/(kgCO2/s)]
E1=E_u(Q1,y1,v1);
E2=E_u(Q2,y2,v2);
E3=E_g(Q3,y3,v3);
objfun(2)=(E1+E2-E3)/(Q_p*y_p*0.04401/22414); %[MW/(kgCO2/s)]
elseif I==5
fprintf('%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t\r\n','stadio','y_p','Q_p
[cmSTP^3/s]','x_r','Q_r [cmSTP^3/s]','A [cm^2]')
fprintf('%16s\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t%16.4e\t\r\n','1',
y_p1,Q_p1,x_r1,Q_r1,A1,'2',y_p2,Q_p2,x_r,Q_r,A2)
elseif I==6 && type==1
objfun=[Tout(y1,v1)-273.15, Tout(y2,v2)-273.15;
Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414)
Pt(Q2,y2,v2)/(Q_p*y_p*0.04401/22414)];
elseif I==6 && type==2
objfun=[Tout(y1,v1)-273.15; Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414)];
elseif I==6 && type==3
objfun=[Tout(y2,v2)-273.15; Pt(Q2,y2,v2)/(Q_p*y_p*0.04401/22414)];
elseif I==6 && type==4
objfun={'-','-'};
end
end

```

D.2 Funzione implementata per l'ottimizzazione dello schema con due stadi in cascata, quando il prodotto è il flusso permeato e la specifica di progetto è il recupero totale del componente A. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```
function [objfun] = duetadicascata(type,optvar,dati,I,E_u,E_g,Tout,Pt)
%Si pone la pressione del lato della membrana noto a 1.01 [bar], come la pressione
di feed.
p_f=1.01*75.0062; %[cmHg]
if type==1
    [y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),
                                                dati(3),dati(4),dati(5),optvar(2),dati(7));
    R2=dati(6)/optvar(2);
    [y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(optvar(3),p_f,dati(2),
                                                dati(3),Q_p1,y_p1,R2,dati(7));
    Q1=dati(4);
    y1=dati(5);
    Q2=Q_p1;
    y2=y_p1;
    Q3=Q_r1;
    y3=x_r1;
    Q4=Q_r2;
    y4=x_r2;
    v1=optvar(1)/dati(1);
    v2=optvar(3)/p_f;
    v3=p_f/optvar(1);
    v4=p_f/optvar(3);
elseif type==2
    [y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),
                                                dati(3),dati(4),dati(5),optvar(2),dati(7));
    R2=dati(6)/optvar(2);
    [y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(p_f,optvar(3),dati(2),
                                                dati(3),Q_p1,y_p1,R2,dati(7));
    Q1=dati(4);
    y1=dati(5);
    Q2=Q_p;
    y2=y_p;
    Q3=Q_r1;
    y3=x_r1;
    Q4=0;
    y4=0;
    v1=optvar(1)/dati(1);
    v2=p_f/optvar(3);
    v3=p_f/optvar(1);
    v4=1;
elseif type==3
    [y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(dati(1),optvar(1),dati(2),
                                                dati(3),dati(4),dati(5),optvar(2),dati(7));
    R2=dati(6)/optvar(2);
    [y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(optvar(3),p_f,dati(2),
                                                dati(3),Q_p1,y_p1,R2,dati(7));
    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_p1;
    y2=y_p1;
    Q3=Q_r2;
```

```

y3=x_r2;
Q4=0;
y4=0;
v1=p_f/optvar(1);
v2=optvar(3)/p_f;
v3=p_f/optvar(3);
v4=1;
elseif type==4
[y_p1,Q_p1,x_r1,Q_r1,A1]=countercurrentmodel(dati(1),optvar(1),dati(2),
                                                dati(3),dati(4),dati(5),optvar(2),dati(7));
R2=dati(6)/optvar(2);
[y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(p_f,optvar(3),dati(2),
                                                dati(3),Q_p1,y_p1,R2,dati(7));

Q1=Q_p1;
y1=y_p1;
Q2=Q_p;
y2=y_p;
Q3=0;
y3=0;
Q4=0;
y4=0;
v1=p_f/optvar(1);
v2=p_f/optvar(3);
v3=1;
v4=1;
end
if I==1
objfun=1-y_p;
elseif y_p<dati(8)
objfun=inf; %serve a escludere tutte le condizioni in cui la purezza minima
            non è soddisfatta
elseif I==2
objfun=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4); %area specifica totale
                                                [m^2/(kgCO2/s)]
elseif I==3
E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
E3=E_g(Q3,y3,v3); % - energia prodotta dal primo espansore, se presente
E4=E_g(Q4,y4,v4); % - energia prodotta dal secondo espansore, se presente
objfun=(E1+E2-E3-E4)/(Q_p*y_p*0.04401/22414); %consumo energetico specifico
                                                [MW/(kgCO2/s)]
elseif I==4
objfun(1)=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4);
E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
E3=E_g(Q3,y3,v3); % - energia prodotta dal primo espansore, se presente
E4=E_g(Q4,y4,v4); % - energia prodotta dal secondo espansore, se presente
objfun(2)=(E1+E2-E3-E4)/(Q_p*y_p*0.04401/22414); %consumo energetico specifico
                                                [MW/(kgCO2/s)]
elseif I==5
fprintf('%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t\r\n','stadio','y_p','Q_p
        [cmSTP^3/s]','x_r','Q_r [cmSTP^3/s]','A [cm^2]')
fprintf('%16s\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t%16.4e\t\r\n','1',
        y_p1,Q_p1,x_r1,Q_r1,A1,'2',y_p,Q_p,x_r2,Q_r2,A2)
elseif I==6 && type==1 || I==6 && type==3
objfun=[Tout(y1,v1)-273.15, Tout(y2,v2)-273.15;
        Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414)
        Pt(Q2,y2,v2)/(Q_p*y_p*0.04401/22414)];
elseif I==6 && type==2 || I==6 && type==4

```

```

    objfun=[Tout(y1,v1)-273.15; Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414)];
end
end

```

D.3 Funzione implementata per l'ottimizzazione dello schema con due stadi in serie e il ricircolo del permeato del secondo stadio. Il prodotto è il flusso permeato e la specifica di progetto è il recupero totale del componente A. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```

function [objfun] = duestadiseriercy(type,optvar,dati,I,E_u,E_g,Tout,Pt)
NMAX=100; %numero massimo di iterazioni
tol=1e-7; %tolleranza desiderata
Qr1xr1=dati(4)*dati(5)*(1-dati(6))/(1-optvar(3)); %portata di A nel retentato
                                                uscente dal primo stadio
Qp2yp2=Qr1xr1*optvar(3); %portata di A ricircolata
R1=dati(6)*dati(5)*dati(4)/(dati(5)*dati(4)+Qp2yp2); %recupero del primo stadio
Q_f1_trial=dati(4)+Qp2yp2; %portata alimentata al primo stadio, valore di primo
                            tentativo
x_f1=(dati(4)*dati(5)+Qp2yp2)/Q_f1_trial; %frazione di A alimentata al primo
                                            stadio, valore di primo tentativo
%si pone la pressione del lato della membrana noto a 1.01 [bar], come la pressione
di feed.
p_f=1.01*75.0062; %[cmHg]
if type==1
    [y_p0,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),dati(3),
                                                Q_f1_trial,x_f1,R1,dati(7));
    [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(2),p_f,dati(2),
                                                dati(3),Q_r1,x_r1,optvar(3),dati(7));
    for n=1:NMAX
        Q_f1=dati(4)+Q_p2; %portata alimentata al primo stadio, nuovo tentativo
        x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1; %frazione di A alimentata al primo
                                                stadio, nuovo tentativo
        [y_p,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),
                                                dati(3),Q_f1,x_f1,R1,dati(7));
        [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(2),p_f,dati(2),
                                                dati(3),Q_r1,x_r1,optvar(3),dati(7));
        if abs(y_p-y_p0)<1e-2 && y_p<dati(8) %verifica se la purezza minima è
            soddisfatta; se no, si evitano ulteriori calcoli dispendiosi
            break
        elseif abs((Q_f1-Q_f1_trial)/Q_f1_trial)<=tol %controlla se è soddisfatta
            la tolleranza sulla portata alimentata
            break
        end
        Q_f1_trial=Q_f1;
        y_p0=y_p;
    end
    Q1=Q_f1;
    y1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;
    Q2=Q_r1;
    y2=x_r1;
    Q3=Q_r;
    y3=x_r;
    v1=optvar(1)/p_f;
    v2=optvar(2)/optvar(1);

```

```

v3=p_f/optvar(2);
elseif type==2
[y_p0,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),dati(3),
                                             Q_f1_trial,x_f1,R1,dati(7));
[y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(1),optvar(2),dati(2),
                                             dati(3),Q_r1,x_r1,optvar(3),dati(7));
for n=1:NMAX
    Q_f1=dati(4)+Q_p2;
    x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;

    [y_p,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(optvar(1),p_f,dati(2),dati(3),
                                             Q_f1,x_f1,R1,dati(7));
    [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(1),optvar(2),dati(2),
                                             dati(3),Q_r1,x_r1,optvar(3),dati(7));
    if abs(y_p-y_p0)<1e-2 && y_p<dati(8)
        break
    elseif abs((Q_f1-Q_f1_trial)/Q_f1_trial)<=tol
        break
    end
    Q_f1_trial=Q_f1;
    y_p0=y_p;
end
Q1=Q_f1; %questo è vero se la pressione di feed è di 1 bar
y1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;
Q2=Q_p2;
y2=y_p2;
Q3=Q_r;
y3=x_r;
v1=optvar(1)/p_f;
v2=p_f/optvar(2);
v3=p_f/optvar(1);
elseif type==3
[y_p0,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                             Q_f1_trial,x_f1,R1,dati(7));
[y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(2),p_f,dati(2),dati(3),
                                             Q_r1,x_r1,optvar(3),dati(7));
for n=1:NMAX
    Q_f1=dati(4)+Q_p2;
    x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;
    [y_p,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                             Q_f1,x_f1,R1,dati(7));
    [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(optvar(2),p_f,dati(2),dati(3),
                                             Q_r1,x_r1,optvar(3),dati(7));
    if abs(y_p-y_p0)<1e-2 && y_p<dati(8)
        break
    elseif abs((Q_f1-Q_f1_trial)/Q_f1_trial)<=tol
        break
    end
    Q_f1_trial=Q_f1;
    y_p0=y_p;
end
Q1=Q_p;
y1=y_p;
Q2=Q_r1;
y2=x_r1;
Q3=Q_r;
y3=x_r;
v1=p_f/optvar(1);
v2=optvar(2)/p_f; %vero se il feed è inviato a 1 bar

```

```

v3=p_f/optvar(2);
elseif type==4
[y_p0,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                             Q_f1_trial,x_f1,R1,dati(7));
[y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(p_f,optvar(2),dati(2),dati(3),
                                             Q_r1,x_r1,optvar(3),dati(7));
for n=1:NMAX
    Q_f1=dati(4)+Q_p2;
    x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;
    [y_p,Q_p,x_r1,Q_r1,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                             Q_f1,x_f1,R1,dati(7));
    [y_p2,Q_p2,x_r,Q_r,A2]=countercurrentmodel(p_f,optvar(2),dati(2),dati(3),
                                             Q_r1,x_r1,optvar(3),dati(7));
    if abs(y_p-y_p0)<1e-2 && y_p<dati(8)
        break
    elseif abs(Q_f1-Q_f1_trial)/Q_f1_trial<=tol
        break
    end
    Q_f1_trial=Q_f1;
    y_p0=y_p;
end
Q1=Q_p;
y1=y_p;
Q2=Q_p2;
y2=y_p2;
Q3=0;
y3=0;
v1=p_f/optvar(1);
v2=p_f/optvar(2);
v3=1;
end
if I==1
    objfun=1-y_p;
elseif y_p<dati(8)
    objfun=1e300; %serve a escludere tutte le condizioni in cui la purezza minima
                 non è soddisfatta
elseif I==2
    objfun=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4); %area specifica totale
                                                    [m^2/(kgCO2/s)]
elseif I==3
    E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
    E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
    E3=E_g(Q3,y3,v3); %energia prodotta dall'espansore, se presente
    objfun=(E1+E2-E3)/(Q_p*y_p*0.04401/22414); %consumo energetico specifico
                                                    [MW/(kgCO2/s)]
elseif I==4
    objfun(1)=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4); %area specifica totale
                                                    [m^2/(kgCO2/s)]
    E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
    E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
    E3=E_g(Q3,y3,v3); %energia prodotta dall'espansore, se presente
    objfun(2)=(E1+E2-E3)/(Q_p*y_p*0.04401/22414); %consumo energetico specifico
                                                    [MW/(kgCO2/s)]
elseif I==5
    fprintf('%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t\r\n','stadio','y_p','Q_p
           [cmSTP^3/s]','x_r','Q_r [cmSTP^3/s]','A [cm^2]')
    fprintf('%16s\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t%16.4e\t\r\n','1',
           y_p,Q_p,x_r1,Q_r1,A1,'2',y_p2,Q_p2,x_r,Q_r,A2)
elseif I==6 && type==1 || I==6 && type==2

```

```

    objfun=[Tout(y1,v1)-273.15, Tout(y2,v2)-273.15;
Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414) Pt(Q2,y2,v2)/(Q_p*y_p*0.04401/22414)];
elseif I==6 && type==3 || I==6 && type==4
    objfun=[Tout(y2,v2)-273.15; Pt(Q2,y2,v2)/(Q_p*y_p*0.04401/22414)];
end
end
end

```

D.4 Funzione implementata per l'ottimizzazione dello schema con due stadi in cascata e il ricircolo del retentato del secondo stadio. Il prodotto è il flusso permeato e la specifica di progetto è il recupero totale del componente A. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```

function [objfun] = duestadicascatarcy(type,optvar,dati,I,E_u,E_g,Tout,Pt)
NMAX=100; %numero massimo di iterazioni
tol=1e-7; %tolleranza desiderata
Qr2xr2=dati(4)*dati(5)*(optvar(2)-dati(6))/(1-optvar(2)); %portata di A
                                                    ricircolata
Qp1yp1=(dati(4)*dati(5)+Qr2xr2)*optvar(2); %portata di A nel permeato del primo
                                                    stadio
R2=dati(4)*dati(5)*dati(6)/Qp1yp1; %recupero del secondo stadio
Q_f1_trial=dati(4)+Qr2xr2/0.5; %portata alimentata al primo stadio, valore di
                                                    primo tentativo
x_f1=(dati(4)*dati(5)+Qr2xr2)/Q_f1_trial; %frazione di A alimentata al primo
                                                    stadio, valore di primo tentativo
%si pone la pressione del lato della membrana noto a 1.01 [bar], come la pressione
di feed.
p_f=1.01*75.0062; %[cmHg]
if type==1
    [y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(optvar(1),p_f,dati(2),dati(3),
                                                    Q_f1_trial,x_f1,optvar(2),dati(7)); %primo stadio
    [y_p0,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(optvar(3),p_f,dati(2),dati(3),
                                                    Q_p1,y_p1,R2,dati(7)); %secondo stadio, alimentato
                                                    con il permeato del primo stadio
    for n=1:NMAX
        Q_f1=dati(4)+Q_r2; %portata alimentata al primo stadio, nuovo tentativo
        x_f1=(dati(4)*dati(5)+Q_r2*x_r2)/Q_f1; %frazione di A alimentata al primo
                                                    stadio, nuovo tentativo
        [y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(optvar(1),p_f,dati(2),dati(3),
                                                    Q_f1,x_f1,optvar(2),dati(7));
        [y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(optvar(3),p_f,dati(2),dati(3),
                                                    Q_p1,y_p1,R2,dati(7));
        if abs(y_p-y_p0)<1e-2 && y_p<dati(8) %verifica se la purezza minima è
                                                    soddisfatta; se no, si evitano ulteriori calcoli dispendiosi
            break
        elseif abs((Q_f1-Q_f1_trial)/Q_f1_trial)<=tol %controlla se è soddisfatta
                                                    la tolleranza
            break
        end
        Q_f1_trial=Q_f1;
        y_p0=y_p;
    end
    Q1=dati(4);
    y1=dati(5);

```

```

Q2=Q_p1;
y2=y_p1;
Q3=Q_r;
y3=x_r;
Q4=Q_r2;
y4=x_r2;
v1=optvar(1)/dati(1);
v2=optvar(3)/p_f;
v3=p_f/optvar(1);
v4=optvar(1)/optvar(3);
elseif type==2
[y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(optvar(1),p_f,dati(2),dati(3),
                                             Q_f1_trial,x_f1,optvar(2),dati(7));
[y_p0,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(p_f,optvar(3),dati(2),dati(3),
                                             Q_p1,y_p1,R2,dati(7));
for n=1:NMAX
Q_f1=dati(4)+Q_r2;
x_f1=(dati(4)*dati(5)+Q_r2*x_r2)/Q_f1;
[y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(optvar(1),p_f,dati(2),dati(3),
                                             Q_f1,x_f1,optvar(2),dati(7));
[y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(p_f,optvar(3),dati(2),dati(3),
                                             Q_p1,y_p1,R2,dati(7));
if abs(y_p-y_p0)<1e-2 && y_p<dati(8)
break
elseif abs((Q_f1-Q_f1_trial)/Q_f1_trial)<=tol
break
end
Q_f1_trial=Q_f1;
y_p0=y_p;
end
Q1=Q_f1;
y1=(dati(4)*dati(5)+Q_r2*x_r2)/Q_f1;
Q2=Q_p;
y2=y_p;
Q3=Q_r;
y3=x_r;
Q4=0;
y4=0;
v1=optvar(1)/p_f;
v2=p_f/optvar(3);
v3=p_f/optvar(1);
v4=1;
elseif type==3
[y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                             Q_f1_trial,x_f1,optvar(2),dati(7));
[y_p0,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(optvar(3),p_f,dati(2),dati(3),
                                             Q_p1,y_p1,R2,dati(7));
for n=1:NMAX
Q_f1=dati(4)+Q_r2;
x_f1=(dati(4)*dati(5)+Q_r2*x_r2)/Q_f1;
[y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                             Q_f1,x_f1,optvar(2),dati(7));
[y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(optvar(3),p_f,dati(2),dati(3),
                                             Q_p1,y_p1,R2,dati(7));
if abs(y_p-y_p0)<1e-2 && y_p<dati(8)
break
elseif abs(Q_f1-Q_f1_trial)/Q_f1_trial<=tol
break
end
end

```

```

        Q_f1_trial=Q_f1;
        y_p0=y_p;
    end
    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_p1;
    y2=y_p1;
    Q3=Q_r2;
    y3=x_r2;
    Q4=0;
    y4=0;
    v1=p_f/optvar(1);
    v2=optvar(3)/p_f;
    v3=p_f/optvar(3);
    v4=1;
elseif type==4
    [y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                                Q_f1_trial,x_f1,optvar(2),dati(7));
    [y_p0,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(p_f,optvar(3),dati(2),dati(3),
                                                Q_p1,y_p1,R2,dati(7));

    for n=1:NMAX
        Q_f1=dati(4)+Q_r2;
        x_f1=(dati(4)*dati(5)+Q_r2*x_r2)/Q_f1;
        [y_p1,Q_p1,x_r,Q_r,A1]=countercurrentmodel(p_f,optvar(1),dati(2),dati(3),
                                                Q_f1,x_f1,optvar(2),dati(7));
        [y_p,Q_p,x_r2,Q_r2,A2]=countercurrentmodel(p_f,optvar(3),dati(2),dati(3),
                                                Q_p1,y_p1,R2,dati(7));

        if abs(y_p-y_p0)<1e-2 && y_p<dati(8)
            break
        elseif abs((Q_f1-Q_f1_trial)/Q_f1_trial)<=tol
            break
        end
        Q_f1_trial=Q_f1;
        y_p0=y_p;
    end
    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_p;
    y2=y_p;
    Q3=0;
    y3=0;
    Q4=0;
    y4=0;
    v1=p_f/optvar(1);
    v2=p_f/optvar(3);
    v3=1;
    v4=1;
end
if I==1
    objfun=1-y_p;
elseif y_p<dati(8)
    objfun=1e300; %serve a escludere tutte le condizioni in cui la purezza minima
                non è soddisfatta
elseif I==2
    objfun=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4); %area specifica totale
                                                    [m^2/(kgCO2/s)]
elseif I==3
    E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
    E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto

```

```

E3=E_g(Q3,y3,v3); %energia prodotta dal primo espansore, se presente
E4=E_g(Q4,y4,v4); %energia prodotta dal secondo espansore, se presente
objfun=(E1+E2-E3-E4)/(Q_p*y_p*0.04401/22414); %consumo energetico specifico
[MW/(kgCO2/s)]
elseif I==4
objfun(1)=(A1+A2)/(y_p*Q_p*0.04401/22414)*10^(-4); %area specifica totale
[m^2/(kgCO2/s)]
E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
E3=E_g(Q3,y3,v3); %energia prodotta dal primo espansore, se presente
E4=E_g(Q4,y4,v4); %energia prodotta dal secondo espansore, se presente
objfun(2)=(E1+E2-E3-E4)/(Q_p*y_p*0.04401/22414); %consumo energetico specifico
[MW/(kgCO2/s)]
elseif I==5
fprintf('%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t\r\n','stadio','y_p','Q_p
[cmSTP^3/s'],'x_r','Q_r [cmSTP^3/s'],'A [cm^2]')
fprintf('%16s\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t%16.4e\t\r\n','1',
y_p1,Q_p1,x_r,Q_r,A1,'2',y_p,Q_p,x_r2,Q_r2,A2)
elseif I==6 && type==1
objfun=[Tout(y1,v1)-273.15, Tout(y2,v2)-273.15, Tout(y4,v4)-273.15;
Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414) Pt(Q2,y2,v2)/(Q_p*y_p*0.04401/22414)
Pt(Q4,y4,v4)/(Q_p*y_p*0.04401/22414)];
elseif I==6 && type==2 || I==6 && type==4
objfun=[Tout(y1,v1)-273.15; Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414)];
elseif I==6 && type==3
objfun=[Tout(y1,v1)-273.15, Tout(y2,v2)-273.15, Tout(y3,v3)-273.15;
Pt(Q1,y1,v1)/(Q_p*y_p*0.04401/22414) Pt(Q2,y2,v2)/(Q_p*y_p*0.04401/22414)
Pt(Q3,y3,v3)/(Q_p*y_p*0.04401/22414)];
end
end
end

```

Appendice E

E.1 Funzione implementata per l'ottimizzazione dello schema con due stadi in serie, quando il prodotto è la corrente di retentato e la specifica di progetto è la purezza del componente B. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```
function [objfun] = duestadiserie(type,optvar,dati,I,E_u,E_g,LHV,Tout,Pt)
if type==1
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),dati(3),
                                                dati(4),dati(5),optvar(2),dati(7));
    [y_p2,Q_p2,Q_r,R2,A2]=countercurrentmodel(optvar(3),75.0062,dati(2),dati(3),
                                                Q_r1,optvar(2),dati(6),dati(7));

    Q1=dati(4);
    y1=dati(5);
    Q2=Q_r1;
    y2=optvar(2);
    Q3=Q_r;
    y3=dati(6);
    v1=optvar(1)/dati(1);
    v2=optvar(3)/optvar(1);
    v3=75.0062/optvar(3);
elseif type==2
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),dati(3),
                                                dati(4),dati(5),optvar(2),dati(7));
    [y_p2,Q_p2,Q_r,R2,A2]=countercurrentmodel(optvar(1),optvar(3),dati(2),dati(3),
                                                Q_r1,optvar(2),dati(6),dati(7));

    Q1=dati(4);
    y1=dati(5);
    Q2=Q_p2;
    y2=y_p2;
    Q3=Q_r;
    y3=dati(6);
    v1=optvar(1)/dati(1);
    v2=75.0062/optvar(3);
    v3=75.0062/optvar(1);
elseif type==3
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(dati(1),optvar(1),dati(2),dati(3),
                                                dati(4),dati(5),optvar(2),dati(7));
    [y_p2,Q_p2,Q_r,R2,A2]=countercurrentmodel(optvar(3),75.0062,dati(2),dati(3),
                                                Q_r1,optvar(2),dati(6),dati(7));

    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_r1;
    y2=optvar(2);
    Q3=Q_r;
    y3=dati(6);
    v1=75.0062/optvar(1);
    v2=optvar(3)/dati(1);
    v3=75.0062/optvar(3);
elseif type==4
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(dati(1),optvar(1),dati(2),dati(3),
                                                dati(4),dati(5),optvar(2),dati(7));
```


E.2 Funzione implementata per l'ottimizzazione dello schema con due stadi in cascata, quando il prodotto è la corrente di retentato e la specifica di progetto è la purezza del componente B. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```
function [objfun] = duetadicascata(type,optvar,dati,I,E_u,E_g,LHV,Tout,Pt)
NMAX=100; %numero massimo di iterazioni per il calcolo del tipo trial&error,
        spiegato in seguito
tol=1e-10; %tolleranza desiderata per fermare la procedura trial%error
if type==1
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),dati(3),
        dati(4),dati(5),optvar(2),dati(7));
    Q_r2trial=Q_p1*0.99; %portata di retentato del secondo stadio di primo
        tentativo
    for n=1:NMAX
        x_r2=((Q_r1+Q_r2trial)*dati(6)-Q_r1*optvar(2))/Q_r2trial; %calcolo della
            frazione molare di A nel retentato del
            secondo stadio dai bilanci di materia

        if x_r2<0
            disp('Attenzione calcolato x_r2<0')
        end
        [y_p,Q_p,Q_r2,R2,A2]=countercurrentmodel(optvar(3),75.0062,dati(2),
            dati(3),Q_p1,y_p1,x_r2,dati(7));
        if abs((Q_r2-Q_r2trial)/Q_r2trial)<=tol
            break
        end
        Q_r2trial=Q_r2; %imposto la portata di retentato calcolata come nuovo
            valore di tentativo
    end
    Q_r=Q_r1+Q_r2; %portata di retentato totale
    Q1=dati(4);
    y1=dati(5);
    Q2=Q_p1;
    y2=y_p1;
    Q3=Q_r1;
    y3=optvar(2);
    Q4=Q_r2;
    y4=x_r2;
    v1=optvar(1)/dati(1);
    v2=optvar(3)/75.0062;
    v3=75.0062/optvar(1);
    v4=75.0062/optvar(3);

elseif type==2
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),dati(3),
        dati(4),dati(5),optvar(2),dati(7));
    Q_r2trial=Q_p1*0.99;
    for n=1:NMAX
        x_r2=((Q_r1+Q_r2trial)*dati(6)-Q_r1*optvar(2))/Q_r2trial; %calcolo della
            frazione molare di A nel retentato del
            secondo stadio dai bilanci di materia

        if x_r2<0
            disp('Attenzione calcolato x_r2<0')
        end
        [y_p,Q_p,Q_r2,R2,A2]=countercurrentmodel(75.0062,optvar(3),dati(2),
            dati(3),Q_p1,y_p1,x_r2,dati(7));
```

```

        if abs((Q_r2-Q_r2trial)/Q_r2trial)<=tol
            break
        end
        Q_r2trial=Q_r2; %imposto la portata di retentato calcolata come nuovo
                        valore di tentativo
    end
    Q_r=Q_r1+Q_r2; %portata di retentato totale
    Q1=dati(4);
    y1=dati(5);
    Q2=Q_p;
    y2=y_p;
    Q3=Q_r1;
    y3=optvar(2);
    Q4=0;
    y4=0;
    v1=optvar(1)/dati(1);
    v2=75.0062/optvar(3);
    v3=75.0062/optvar(1);
    v4=1;
elseif type==3
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(75.0062,optvar(1),dati(2),dati(3),
                                                dati(4),dati(5),optvar(2),dati(7));
    Q_r2trial=Q_p1*0.99;
    for n=1:NMAX
        x_r2=((Q_r1+Q_r2trial)*dati(6)-Q_r1*optvar(2))/Q_r2trial;
        if x_r2<0
            disp('Attenzione calcolato x_r2<0')
        end
        [y_p,Q_p,Q_r2,R2,A2]=countercurrentmodel(optvar(3),75.0062,dati(2),
                                                dati(3),Q_p1,y_p1,x_r2,dati(7));
        if abs((Q_r2-Q_r2trial)/Q_r2trial)<=tol
            break
        end
        Q_r2trial=Q_r2;
    end
    Q_r=Q_r1+Q_r2;
    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_p1;
    y2=y_p1;
    Q3=Q_r2;
    y3=x_r2;
    Q4=0;
    y4=0;
    v1=75.0062/optvar(1);
    v2=optvar(3)/75.0062;
    v3=75.0062/optvar(3);
    v4=1;
elseif type==4
    [y_p1,Q_p1,Q_r1,R1,A1]=countercurrentmodel(75.0062,optvar(1),dati(2),dati(3),
                                                dati(4),dati(5),optvar(2),dati(7));
    Q_r2trial=Q_p1*0.99;
    for n=1:NMAX
        x_r2=((Q_r1+Q_r2trial)*dati(6)-Q_r1*optvar(2))/Q_r2trial;
        if x_r2<0
            disp('Attenzione calcolato x_r2<0')
        end
        [y_p,Q_p,Q_r2,R2,A2]=countercurrentmodel(75.0062,optvar(3),dati(2),
                                                dati(3),Q_p1,y_p1,x_r2,dati(7));

```

```

        if abs((Q_r2-Q_r2trial)/Q_r2trial)<=tol
            break
        end
        Q_r2trial=Q_r2;
    end
    Q_r=Q_r1+Q_r2;
    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_p;
    y2=y_p;
    Q3=0;
    y3=0;
    Q4=0;
    y4=0;
    v1=75.0062/optvar(1);
    v2=75.0062/optvar(3);
    v3=1;
    v4=1;
end
if x_r2<=0
    disp('Attenzione errore')
end
if I==1
    objfun=(A1+A2)/((1-dati(6))*Q_r*0.01604/22414)*10^(-4); %area specifica totale
                                                [m^2/(kgCH4/s)]
elseif I==2
    E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
    E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
    E3=E_g(Q3,y3,v3); % - energia prodotta dal primo espansore, se presente
    E4=E_g(Q4,y4,v4); % - energia prodotta dal secondo espansore, se presente
    Ch=LHV*Q_p*(1-y_p); %energia chimica della specie B nel permeato
    objfun=(E1+E2-E3-E4+Ch)/(Q_r*(1-dati(6))*0.01604/22414); %consumo energetico
                                                specifico [MW/(kgCH4/s)]
elseif I==3
    objfun(1)=(A1+A2)/((1-dati(6))*Q_r*0.01604/22414)*10^(-4);
    E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore
    E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore
    E3=E_g(Q3,y3,v3); % - energia prodotta dal primo espansore, se presente
    E4=E_g(Q4,y4,v4); % - energia prodotta dal secondo espansore, se presente
    Ch=LHV*Q_p*(1-y_p); %energia chimica della specie B nel permeato
    objfun(2)=(E1+E2-E3-E4+Ch)/(Q_r*(1-dati(6))*0.01604/22414); %consumo
                                                energetico specifico [MW/(kgCH4/s)]
elseif I==4
    R=(Q_r*(1-dati(6)))/(dati(4)*(1-dati(5)));
    fprintf('%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t\r\n','stadio','y_p',
            'Q_p [cmSTP^3/s]','x_r','Q_r [cmSTP^3/s]','Rr','A [cm^2]')
    fprintf('%16s\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t\r\n','1',
            y_p1,Q_p1,x_r1,Q_r1,R1,A1,'2',y_p,Q_p,x_r2,Q_r2,R2,A2)
    disp(['Il recupero di CH4 nel retentato è del ',num2str(R*100),' %'])
elseif I==5 && type==1 || I==5 && type==3
    Tout=[Tout(y1,v1), Tout(y2,v2)];
    objfun=[Tout(1)-273.15, Tout(2)-273.15;
            Pt(Q1,y1,v1,Tout(1))/(Q_r*(1-dati(6))*0.01604/22414)
            Pt(Q2,y2,v2,Tout(2))/(Q_r*(1-dati(6))*0.01604/22414)];
elseif I==5 && type==2 || I==5 && type==4
    Tout=Tout(y1,v1);
    objfun=[Tout-273.15; Pt(Q1,y1,v1,Tout)/(Q_r*(1-dati(6))*0.01604/22414)];
end
end

```

E.3 Funzione implementata per l'ottimizzazione dello schema con due stadi in serie e ricircolo del permeato del secondo stadio, quando il prodotto è la corrente di retentato e la specifica di progetto è la purezza del componente B. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```
function [objfun] = duetadiseriercy(type,optvar,dati,I,E_u,E_g,LHV,Tout,Pt)
NMAX=100; %numero massimo di iterazioni
tol=1e-10; %tolleranza desiderata
Q_r1trial=dati(4)/2;
if type==1
    for n=1:NMAX
        [y_p2,Q_p2,Q_r,R2,A2]=countercurrentmodel(optvar(3),75.0062,dati(2),
            dati(3),Q_r1trial,optvar(2),dati(6),dati(7));
            %secondo stadio, alimentato con il retentato
            di tentativo del primo stadio
        Q_f1=dati(4)+Q_p2; %portata di tentativo in ingresso al primo stadio
        x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1; %frazione di A alimentata al primo
            stadio, di tentativo
        [y_p,Q_p,Q_r1,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),
            dati(3),Q_f1,x_f1,optvar(2),dati(7)); %primo stadio
        if abs((Q_r1-Q_r1trial)/Q_r1trial)<=tol %controlla se è soddisfatta la
            tolleranza
            break
        end
        Q_r1trial=Q_r1;
    end
    Q1=Q_f1; %questo è vero se la pressione di feed è di 1 bar
    y1=x_f1;
    Q2=Q_r1;
    y2=optvar(2);
    Q3=Q_r;
    y3=dati(6);
    v1=optvar(1)/75.0062; %la pressione di feed è di 1 bar
    v2=optvar(3)/optvar(1);
    v3=75.0062/optvar(3);
elseif type==2
    for n=1:NMAX
        [y_p2,Q_p2,Q_r,R2,A2]=countercurrentmodel(optvar(1),optvar(3),dati(2),
            dati(3),Q_r1trial,optvar(2),dati(6),dati(7));

        Q_f1=dati(4)+Q_p2;
        x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;
        [y_p,Q_p,Q_r1,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),
            dati(3),Q_f1,x_f1,optvar(2),dati(7));

        if abs((Q_r1-Q_r1trial)/Q_r1trial)<=tol
            break
        end
        Q_r1trial=Q_r1;
    end
    Q1=Q_f1; %la pressione di feed è di 1 bar
    y1=x_f1;
    Q2=Q_p2;
    y2=y_p2;
    Q3=Q_r;
    y3=dati(6);
    v1=optvar(1)/75.0062; %la pressione di feed è di 1 bar
```

```

v2=75.0062/optvar(3);
v3=75.0062/optvar(1);
elseif type==3
    for n=1:NMAX
        [y_p2,Q_p2,Q_r,R2,A2]=countercurrentmodel(optvar(3),75.0062,dati(2),
                                                    dati(3),Q_r1trial,optvar(2),dati(6),dati(7));

        Q_f1=dati(4)+Q_p2;
        x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;
        [y_p,Q_p,Q_r1,R1,A1]=countercurrentmodel(75.0062,optvar(1),dati(2),
                                                    dati(3),Q_f1,x_f1,optvar(2),dati(7));

        if abs((Q_r1-Q_r1trial)/Q_r1trial)<=tol
            break
        end
        Q_r1trial=Q_r1;
    end
    Q1=Q_p;
    y1=y_p;
    Q2=Q_r1;
    y2=optvar(2);
    Q3=Q_r;
    y3=dati(6);
    v1=75.0062/optvar(1);
    v2=optvar(3)/75.0062; %il feed è inviato a 1 bar
    v3=75.0062/optvar(3);
elseif type==4
    for n=1:NMAX
        [y_p2,Q_p2,Q_r,R2,A2]=countercurrentmodel(75.0062,optvar(3),dati(2),
                                                    dati(3),Q_r1trial,optvar(2),dati(6),dati(7));

        Q_f1=dati(4)+Q_p2;
        x_f1=(dati(4)*dati(5)+Q_p2*y_p2)/Q_f1;
        [y_p,Q_p,Q_r1,R1,A1]=countercurrentmodel(75.0062,optvar(1),dati(2),
                                                    dati(3),Q_f1,x_f1,optvar(2),dati(7));

        if abs((Q_r1-Q_r1trial)/Q_r1trial)<=tol
            break
        end
        Q_r1trial=Q_r1;
    end
    Q1=Q_p;
    y1=y_p;
    Q2=Q_p2;
    y2=y_p2;
    Q3=0;
    y3=0;
    v1=75.0062/optvar(1); %se il feed è inviato a 1 bar
    v2=75.0062/optvar(3); %vero se il feed è inviato a 1 bar
    v3=1;
end
if I==1
    objfun=(A1+A2)/((1-dati(6))*Q_r*0.01604/22414)*10^(-4); %area specifica totale
                                                    [m^2/(kgCH4/s)]
elseif I==2
    E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
    E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
    E3=E_g(Q3,y3,v3); %energia prodotta dall'espansore, se presente
    Ch=LHV*Q_p*(1-y_p); %energia chimica della specie B nel permeato
    objfun=(E1+E2-E3+Ch)/(Q_r*(1-dati(6))*0.01604/22414); %consumo energetico
                                                    specifico [MW/(kgCH4/s)]
elseif I==3
    objfun(1)=(A1+A2)/((1-dati(6))*Q_r*0.01604/22414)*10^(-4);

```

```

E1=E_u(Q1,y1,v1); %consumo energetico del primo compressore/pompa a vuoto
E2=E_u(Q2,y2,v2); %consumo energetico del secondo compressore/pompa a vuoto
E3=E_g(Q3,y3,v3); %energia prodotta dall'espansore, se presente
Ch=LHV*Q_p*(1-y_p); %energia chimica della specie B nel permeato
objfun(2)=(E1+E2-E3+Ch)/(Q_r*(1-dati(6))*0.01604/22414); %consumo energetico
                                                    specifico [MW/(kgCH4/s)]
elseif I==4
    R=(Q_r*(1-dati(6)))/(dati(4)*(1-dati(5)));
    fprintf('%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t%16s\t\r\n','stadio','y_p',
            'Q_p [cmSTP^3/s]','x_r','Q_r [cmSTP^3/s]','Rr','A [cm^2]')
    fprintf('%16s\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t%16.4f\t%16.4e\t\r\n','1',
            y_p,Q_p,optvar(2),Q_r1,R1,A1,'2',y_p2,Q_p2,dati(6),Q_r,R2,A2)
    disp(['Il recupero di CH4 nel retentato è del ',num2str(R*100),' %'])
elseif I==5 && type==1 || I==5 && type==2
    Tout=[Tout(y1,v1), Tout(y2,v2)];
    objfun=[Tout(1)-273.15, Tout(2)-273.15;
            Pt(Q1,y1,v1,Tout(1))/(Q_r*(1-dati(6))*0.01604/22414)
            Pt(Q2,y2,v2,Tout(2))/(Q_r*(1-dati(6))*0.01604/22414)];
elseif I==5 && type==3 || I==5 && type==4
    Tout=[Tout(y2,v2)];
    objfun=[Tout-273.15; Pt(Q2,y2,v2,Tout)/(Q_r*(1-dati(6))*0.01604/22414)];
end
end
end

```

E.4 Funzione implementata per l'ottimizzazione dello schema con due stadi in cascata e ricircolo del retentato del secondo stadio, quando il prodotto è la corrente di retentato e la specifica di progetto è la purezza del componente B. La stessa funzione viene utilizzata come *fitness function* nelle ottimizzazioni a singolo e a due obiettivi; per il calcolo delle correnti e della *performance* di ciascuno stadio; nel determinare la potenza termica utilizzata dagli scambiatori.

```

function [objfun] = duestadicascatarcy(type,optvar,dati,I,E_u,E_g,LHV,Tout,Pt)
NMAX=100; %numero massimo di iterazioni
tol=1e-10; %tolleranza desiderata
Q_r2trial=dati(4)*1.3; %portata ricircolata al primo stadio, valore di primo
                    tentativo
if type==1
    %inizio del calcolo del tipo trial&error
    for n=1:NMAX
        Q_f1=dati(4)+Q_r2trial; %portata alimentata al primo stadio, di tentativo
        x_f1=(dati(4)*dati(5)+Q_r2trial*optvar(3))/Q_f1; %frazione di A alimentata
                    al primo stadio, di tentativo
        [y_p1,Q_p1,Q_r,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),
            dati(3),Q_f1,x_f1,dati(6),dati(7));
        [y_p,Q_p,Q_r2,R2,A2]=countercurrentmodel(optvar(2),75.0062,dati(2),
            dati(3),Q_p1,y_p1,optvar(3),dati(7));
        if abs((Q_r2-Q_r2trial)/Q_r2trial)<=tol %controlla se è soddisfatta la
            tolleranza
            break
        end
        Q_r2trial=Q_r2;
    end
end

```

```

Q1=dati(4);
y1=dati(5);
Q2=Q_p1;
y2=y_p1;
Q3=Q_r;
y3=dati(6);
Q4=Q_r2;
y4=optvar(3);
v1=optvar(1)/dati(1); %questo è vero se la pressione di feed è di 1 bar
v2=optvar(2)/75.0062;
v3=75.0062/optvar(1);
v4=optvar(1)/optvar(2);
elseif type==2
    for n=1:NMAX
        Q_f1=dati(4)+Q_r2trial;
        x_f1=(dati(4)*dati(5)+Q_r2trial*optvar(3))/Q_f1;
        [y_p1,Q_p1,Q_r,R1,A1]=countercurrentmodel(optvar(1),75.0062,dati(2),
                                                    dati(3),Q_f1,x_f1,dati(6),dati(7));
        [y_p,Q_p,Q_r2,R2,A2]=countercurrentmodel(75.0062,optvar(2),dati(2),
                                                    dati(3),Q_p1,y_p1,optvar(3),dati(7));
        if abs((Q_r2-Q_r2trial)/Q_r2trial)<=tol
            break
        end
        Q_r2trial=Q_r2;
    end
    Q1=Q_f1; %questo è vero se la pressione di feed è di 1 bar
    y1=x_f1;
    Q2=Q_p;
    y2=y_p;
    Q3=Q_r;
    y3=dati(6);
    Q4=0;
    y4=0;
    v1=optvar(1)/75.0062; %questo è vero se la pressione di feed è di 1 bar
    v2=75.0062/optvar(2);
    v3=75.0062/optvar(1);
    v4=1;
elseif type==3
    for n=1:NMAX
        Q_f1=dati(4)+Q_r2trial;
        x_f1=(dati(4)*dati(5)+Q_r2trial*optvar(3))/Q_f1;
        [y_p1,Q_p1,Q_r,R1,A1]=countercurrentmodel(75.0062,optvar(1),dati(2),
                                                    dati(3),Q_f1,x_f1,dati(6),dati(7));
        [y_p,Q_p,Q_r2,R2,A2]=countercurrentmodel(optvar(2),75.0062,dati(2),
                                                    dati(3),Q_p1,y_p1,optvar(3),dati(7));
        if abs((Q_r2-Q_r2trial)/Q_r2trial)<=tol
            break
        end
        Q_r2trial=Q_r2;
    end
    Q1=Q_p1;
    y1=y_p1;
    Q2=Q_p1;
    y2=y_p1;
    Q3=Q_r2;
    y3=optvar(3);
    Q4=0;
    y4=0;
    v1=75.0062/optvar(1);

```



```

    Tout=[Tout(y1,v1) Tout(y2,v2) Tout(y4,v4)];
    objfun=[Tout(1)-273.15, Tout(2)-273.15, Tout(3)-273.15;
            Pt(Q1,y1,v1,Tout(1))/(Q_r*(1-dati(6))*0.01604/22414)
            Pt(Q2,y2,v2,Tout(2))/(Q_r*(1-dati(6))*0.01604/22414)
            Pt(Q4,y4,v4,Tout(3))/(Q_r*(1-dati(6))*0.01604/22414)];
elseif I==5 && type==2 || I==5 && type==4
    Tout=Tout(y1,v1);
    objfun=[Tout-273.15; Pt(Q1,y1,v1,Tout)/(Q_r*(1-dati(6))*0.01604/22414)];
elseif I==5 && type==3
    Tout=[Tout(y1,v1) Tout(y2,v2) Tout(y3,v3)];
    objfun=[Tout(1)-273.15, Tout(2)-273.15, Tout(3)-273.15;
            Pt(Q1,y1,v1,Tout(1))/(Q_r*(1-dati(6))*0.01604/22414)
            Pt(Q2,y2,v2,Tout(2))/(Q_r*(1-dati(6))*0.01604/22414)
            Pt(Q3,y3,v3,Tout(3))/(Q_r*(1-dati(6))*0.01604/22414)];
end
end

```

Appendice F

Si riporta il codice utilizzato per l'ottimizzazione di un sistema di separazione alimentato con il gas di scarico di una centrale termoelettrica (miscela binaria N_2 - CO_2). Si utilizzano le funzioni `duestadiserie.m`, `duestadicascata.m`, `duestadiseriercy.m`, `duestadicascatarcy.m` (APPENDICE D) per risolvere le equazioni di bilancio e di trasporto di ciascuna configurazione analizzata. Il codice fornisce come *output* i grafici dei fronti di Pareto per le configurazioni che soddisfano le specifiche di progetto, e un file Excel in cui sono salvati tutti i risultati dell'ottimizzazione e le potenze termiche degli scambiatori calcolate.

```
clear all
close all
clc

tic
%creazione del file excel su cui verranno copiati i risultati
filename='simulazione.xlsx';

p_feed=1.01*75.0062; %pressione di arrivo del flue gas, convertita in [cmHg]
p_min=0.2*75.0062; %pressione minima raggiungibile dalla pompa a vuoto [cmHg]
p_max=10*75.0062; %pressione massima di compressione scelta [cmHg]
permCO2=1000e-6; %permeanza della CO2 [cmSTP^3/(cm^2*s*cmHg)]
permN2=permCO2/50; %permeanza dell'N2 (selettività=50) [cmSTP^3/(cm^2*s*cmHg)]
Q_f=2.1e6*10^6/3600; %portata alimentata [cmSTP^3 s^-1]
x_f=0.15; %frazione molare della CO2 nel feed
R=0.9; %recupero desiderato di CO2 nel permeato
Pmin=0.96; %purezza minima imposta, standard delle condutture
T=40+273.15; %temperatura operativa nel modulo a membrana [K]
l=1; %spessore della membrana, in questo caso non va utilizzato perchè si
    fornisce la permeanza

cp_CO2=@(T) 5.457+1.045e-3*T-1.157e5*T.^(-2); %calore specifico della CO2
    diviso per la costante dei gas
cp_N2=@(T) 3.28+0.593e-3*T+0.04e5*T.^(-2); %calore specifico dell'N2 diviso
    per la costante dei gas
Trange=linspace(298,580,1000); %range operativo di temperatura, a causa della
    compressione ed espansione del gas
cp=mean([cp_CO2(Trange)',cp_N2(Trange)']); %vettore che contiene i calori
    specifici medi (diviso la costante dei gas) di CO2 e N2
gamma=@(y) (y*cp(1)+(1-y)*cp(2))/(y*(cp(1)-1)+(1-y)*(cp(2)-1)); %fattore di
    espansione adiabatica per una miscela N2-CO2,
    con y frazione molare della CO2
Rconst=8.314; %costante dei gas ideali [J/molK]
eta=0.85; %efficienza di compressori, pompe a vuoto ed espansori
Vm=22414; %volume molare [cmSTP^3/mol]
E_u=@(Q,y,v) Q*gamma(y)*Rconst*T/(Vm*eta*(gamma(y)-1))*(v^((gamma(y)-
    1)/gamma(y))-1)*10^(-6); %potenza utilizzata da compressori e
    pompe a vuoto %[MW]
```

```

E_g=@(Q,y,v) eta*Q*gamma(y)*Rconst*T/(Vm*(gamma(y)-1))*(1-v^((gamma(y)-1)/gamma(y)))*10^(-6); %potenza guadagnata con l'espansione del
retentato [MW]
Tout=@(y,v) T*v^((gamma(y)-1)/gamma(y)); %temperatura in seguito a
compressione/espansione, in entrata allo scambiatore [K]
Pt=@(Q,y,v) Q/Vm*(cp(1)*y+cp(2)*(1-y))*Rconst*T*(v^((gamma(y)-1)/gamma(y))-1)/eta*10^(-6); %potenza termica scambiata nello scambiatore di
calore [MW]
dati=[p_feed permCO2 permN2 Q_f x_f R l Pmin];
t=table(dati(1),dati(2),dati(3),dati(4),dati(5),dati(6),dati(8),
'VariableNames',{'pressione feed [cmHg]','permCO2 x 10^(-6)
[GPU]','permN2 x 10^(-6) [GPU]','portata [cmSTP^3/s]','concentrazione
CO2','recupero CO2','purezza minima'},'RowNames',{'dati assegnati'})
writetable(t,filename,'WriteRowNames',true,'Sheet','uno stadio')

%CONFIGURAZIONE 1: singolo stadio.
%Tipo a
[y_p1a,Q_p1a,x_r1a,Q_r1a,A_1a]=countercurrentmodel(p_feed,p_min,permCO2,
permN2,Q_f,x_f,R,l);
if y_p1a<Pmin
disp(['Il singolo stadio a bassa pressione non permette di raggiungere una
purezza del ',num2str(Pmin*100),'%'])
end
a_1a=A_1a/(y_p1a*Q_p1a*0.04401/Vm)*10^(-4); %area specifica [m^2/(kg_CO2/s)]
e_1a=E_u(Q_p1a,y_p1a,p_feed/p_min)/(Q_p1a*y_p1a*0.04401/Vm); %consumo
elettrico specifico della pompa a vuoto [MW/(kg_CO2/s)]
%Tipo b
[y_p1b,Q_p1b,x_r1b,Q_r1b,A_1b]=countercurrentmodel(p_max,p_feed,permCO2,
permN2,Q_f,x_f,R,l);
if y_p1b<Pmin
disp(['Il singolo stadio ad alta pressione non permette di raggiungere una
purezza del ',num2str(Pmin*100),'%'])
end
a_1b=A_1b/(y_p1b*Q_p1b*0.04401/Vm)*10^(-4); %area specifica [m^2/(kg_CO2/s)]
e_1b=(E_u(Q_f,x_f,p_max/p_feed)-
E_g(Q_r1b,x_r1b,p_feed/p_max))/(Q_p1b*y_p1b*0.04401/Vm); %consumo
elettrico specifico [MW/(kg_CO2/s)]
Tout1b=Tout(x_f,p_max/p_feed); %temperatura in uscita dal compressore [K]
Pt1b=Pt(Q_f,x_f,p_max/p_feed)/(Q_p1b*y_p1b*0.04401/Vm); %potenza termica
specifico dello scambiatore [MW/(kg_CO2/s)]
t=table([p_feed;p_max],[p_min;75.7563],[y_p1a;y_p1b],[a_1a;a_1b],[e_1a;e_1b],
'VariableNames',{'p lato feed [cmHg]','p lato perm
[cmHg]','purezza','area specifica [m^2/(kgCO2/s)]','consumo specifico
[MW/(kgCO2/s)]'},'RowNames',{'singolo stadio bassa p','singolo stadio
alta p'},'DimensionNames',{'configurazioni','variabili'})
writetable(t,filename,'WriteRowNames',true,'Sheet','uno stadio','Range','A6')
c={'Tout compressione [°C]','Potenza termica [MW/(kgCO2/s)]';','';
Tout1b-273.15,Pt1b};
writecell(c,filename,'Sheet','uno stadio','Range','H6')

```

```

%CONFIGURAZIONE 2: 2 stadi in serie senza ricircolo.
lb2=[150 0.001 150; 150 0.001 p_min; p_min 0.001 150; p_min 0.001 p_min];
        %limiti inferiori delle variabili decisionali
ub2=[p_max R*0.999 p_max; p_max R*0.999 67.5056; 37.5031 R*0.999 p_max;
        37.5031 R*0.999 37.5031]; %limiti superiori delle variabili decisionali
%disposizioni generali per il file excel
writecell({'Tipo 1','2 compressori'},filename,'Sheet','2 stadi in serie')
writecell({'Tipo 2','compressore + pompa a vuoto'},filename,'Sheet','2 stadi
        in serie','Range','G1')
writecell({'Tipo 3','pompa a vuoto + compressore'},filename,'Sheet','2 stadi
        in serie','Range','M1')
writecell({'Tipo 4','2 pompe a vuoto'},filename,'Sheet','2 stadi in
        serie','Range','S1')
Position1=["A3","G3","M3","S3"];
Position2=["A5","G5","M5","S5"];
Position3=["A9","G9","M9","S9"];
Position4=["A11","G11","M11","S11"];
Position5=["B187","H187","N187","T187"];
Paretofront={'area specifica [m^2/(kgCO2/s)]','consumo specifico
        [MW/(kgCO2/s)]'};
Optimumvar=["p1 alta [cmHg]","recupero stadio1","p2 alta [cmHg]"; "p1 alta
        [cmHg]","recupero stadio1","p2 bassa [cmHg]";"p1 bassa
        [cmHg]","recupero stadio1","p2 alta [cmHg]";"p1 bassa
        [cmHg]","recupero stadio1","p2 bassa [cmHg]"];

figure
type=1; %tipologia 1
fun=@(optvar) duestadiserie(type,optvar,dati(1:7),1); %definizione della
        funzione di fitness
opts=optimoptions('ga','PopulationSize',90,'EliteCount',15); %opzione per
        l' algoritmo genetico
[var,obj]=ga(fun,3,[1 0 -1],0,[],[],lb2(1,:),ub2(1,:),[],opts);%ottimizzazione
        della purezza del permeato
c={'obiettivo','minimo',Optimumvar(1,1),Optimumvar(1,2),Optimumvar(1,3);
        'purezza',1-obj,var(1),var(2),var(3)};
writecell(c,filename,'Sheet','2 stadi in serie','Range',Position1(1))
if 1-obj<Pmin
    disp(['La purezza massima raggiungibile con 2 stadi in serie (tipologia 1)
        è del ',num2str((1-obj)*100),'%'])
    disp(['La purezza minima del ',num2str(Pmin*100),'% non è soddisfatta'])
else
    opts=optimoptions('ga','PopulationSize',90,'EliteCount',17); %opzioni per
        l' algoritmo genetico
    fun=@(optvar) duestadiserie(type,optvar,dati,2);
    [start1,amin]=ga(fun,3,[1 0 -1],0,[],[],lb2(1,:),ub2(1,:),[],opts);
        %minimizzazione dell'area
    fun=@(optvar) duestadiserie(type,optvar,dati,3,E_u,E_g);
    [start2,emin]=ga(fun,3,[1 0 -1],0,[],[],lb2(1,:),ub2(1,:),[],opts);
        %minimizzazione dei consumi
    c={'area specifica [m^2/(kg_CO2/s)]',amin,start1;'consumo specifico
        [MW/(kg_CO2/s)]',emin,start2};
    writecell(c,filename,'Sheet','2 stadi in serie','Range',Position2(1))

```

```

        %trascrizione in excel dell'ottimizzazione
        separata di ciascun obiettivo
FitnessFunction=@(optvar) duestadiserie(type,optvar,dati,4,E_u,E_g);
        %funzione di fitness per l'ottimizzazione multiobiettivo
opts=optimoptions('gamultiobj','PopulationSize',500,
    'InitialPopulationMatrix',[start1; start2]); %opzioni per il gamultiobj
[optval2,objval2]=gamultiobj(FitnessFunction,3,[1 0 -1],0,[],[],lb2(1,:),
    ub2(1,:),[],opts); %ottimizzazione a due obiettivi
[objval2(:,1),i]=sort(objval2(:,1));
objval2(:,2)=objval2(i,2);
optval2=optval2(i,:);
Heat_ex_1=duestadiserie(type,optval2(1,:),dati,6,E_u,E_g,Tout,Pt);
Heat_ex_2=duestadiserie(type,optval2(round(end/2),:),
    dati,6,E_u,E_g,Tout,Pt);
Heat_ex_3=duestadiserie(type,optval2(end,:),dati,6,E_u,E_g,Tout,Pt);
t=table(Paretofront,Optimumvar(1,:),'VariableNames',
    {'Fronte di Pareto','variabili ottimizzate'});
writetable(t,filename,'Sheet','2 stadi in serie','Range',Position3(1))
writematrix([objval2,optval2],filename,'Sheet',
    '2 stadi in serie','Range',Position4(1))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',
    {'scambiatore'},'RowNames',{'Tin [°C] amin','Pt [MW/(kgCO2/s)]
    amin','Tin [°C] p_medio','Pt [MW/(kgCO2/s)] p_medio','Tin [°C]
    emin','Pt [MW/(kgCO2/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet',
    '2 stadi in serie','Range',Position5(1))
plot(objval2(:,1),objval2(:,2),'linewidth',2,'DisplayName','tipo 1')
title('Pareto Front (2 stadi in serie)')
xlabel('area totale specifica [m^2/(kg_C_0_2/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_0_2/s)]')
legend show
hold on
end
for type=2:4
    fun=@(optvar) duestadiserie(type,optvar,dati(1:7),1);
    opts=optimoptions('particleswarm','SwarmSize',60,
        'FunctionTolerance',1e-7);
    [var,obj]=particleswarm(fun,3,lb2(type,:),ub2(type,:),opts);
    c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
        Optimumvar(type,3),'purezza',1-obj,var(1),var(2),var(3)};
    writecell(c,filename,'Sheet','2 stadi in serie','Range',Position1(type))
    if 1-obj<Pmin
        disp(['La purezza massima raggiungibile con 2 stadi in serie
            (tipologia ',num2str(type),') è del ',num2str((1-obj)*100),'%'])
        disp(['La purezza minima del ',num2str(Pmin*100),'% non è
            soddisfatta'])
    else
        fun=@(optvar) duestadiserie(type,optvar,dati,2);
        [start1,amin]=particleswarm(fun,3,lb2(type,:),ub2(type,:));
        fun=@(optvar) duestadiserie(type,optvar,dati,3,E_u,E_g);
        [start2,emin]=particleswarm(fun,3,lb2(type,:),ub2(type,:));
    end
end

```

```

c={'area specifica [m^2/(kg_CO2/s)]',amin,start1;
  'consumo specifico [MW/(kg_CO2/s)]',emin,start2};
writecell(c,filename,'Sheet','2 stadi in
  serie','Range',Position2(type))
FitnessFunction=@(optvar) duestadiserie(type,optvar,dati,4,E_u,E_g);
opts=optimoptions('gamultiobj','PopulationSize',500,
  'InitialPopulationMatrix',[start1; start2]);
[optval2,objval2]=gamultiobj(FitnessFunction,3,[],[],[],[],
  lb2(type,:),ub2(type,:),[],opts);
[objval2(:,1),i]=sort(objval2(:,1));
objval2(:,2)=objval2(i,2);
optval2=optval2(i,:);
t=table(Paretofront,Optimumvar(type,:),'VariableNames',{'Fronte di
  Pareto','variabili ottimizzate'});
writetable(t,filename,'Sheet','2 stadi in
  serie','Range',Position3(type))
writematrix([objval2,optval2],filename,'Sheet','2 stadi in
  serie','Range',Position4(type))
if type==2 || type==3 %per n=4 non vengono usati scambiatori
  Heat_ex_1=duestadiserie(type,optval2(1,:),dati,6,E_u,E_g,Tout,Pt);
  Heat_ex_2=duestadiserie(type,optval2(round(end/2),:),
    dati,6,E_u,E_g,Tout,Pt);
  Heat_ex_3=duestadiserie(type,optval2(end,:),
    dati,6,E_u,E_g,Tout,Pt);
  t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',
    {'scambiatore'},'RowNames',{'Tin [°C] amin','Pt
    [MW/(kgCO2/s)] amin','Tin [°C] p_medio','Pt
    [MW/(kgCO2/s)] p_medio','Tin [°C] emin','Pt
    [MW/(kgCO2/s)] emin'});
  writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
    serie','Range',Position5(type))
end
plot(objval2(:,1),objval2(:,2),'linewidth',2,'DisplayName',
  ['tipo ',num2str(type)])
title('Pareto Front (2 stadi in serie)')
xlabel('area totale specifica [m^2/(kg_C_O_2/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_O_2/s)]')
legend show
hold on
end
end
%CONFIGURAZIONE 3: 2 stadi in cascata senza ricircolo
lb3=[112.5093 R*1.0001 112.5093; 112.5093 R*1.0001 p_min; p_min R*1.0001
112.5093; p_min R*1.0001 p_min]; %limiti inferiori delle variabili decisionali
ub3=[p_max 0.9999 p_max; p_max 0.9999 37.5031; 37.5031 0.9999 p_max; 37.5031
0.9999 37.5031]; %limiti superiori delle variabili decisionali
writecell({'Tipo 1','2 compressori'},filename,'Sheet','2 stadi in cascata')
writecell({'Tipo 2','compressore + pompa a vuoto'},filename,'Sheet','2 stadi
in cascata','Range','G1')
writecell({'Tipo 3','pompa a vuoto + compressore'},filename,'Sheet','2 stadi
in cascata','Range','M1')

```

```

writecell({'Tipo 4','2 pompe a vuoto'},filename,'Sheet','2 stadi in
          cascata','Range','S1')
figure
for type=1:4
    %ottimizzazione della purezza
    fun=@(optvar) duestadicascata(type,optvar,dati(1:7),1);
    opts=optimoptions('particleswarm','SwarmSize',60,
                      'FunctionTolerance',1e-7);
    [var,obj]=particleswarm(fun,3,lb3(type,:),ub3(type,:),opts);
    c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
      Optimumvar(type,3);'purezza',1-obj,var(1),var(2),var(3)};
    writecell(c,filename,'Sheet','2 stadi in cascata','Range',Position1(type))
    if 1-obj<Pmin %verifica sulla purezza minima del permeato
        disp(['La purezza massima raggiungibile con 2 stadi in cascata
              (tipologia ',num2str(type),') è del ',num2str((1-obj)*100),'%'])
        disp(['La purezza minima del ',num2str(Pmin*100),'% non è
              soddisfatta'])
    else
        fun=@(optvar) duestadicascata(type,optvar,dati,2);
        [start1,amin]=particleswarm(fun,3,lb3(type,:),ub3(type,:));
        fun=@(optvar) duestadicascata(type,optvar,dati,3,E_u,E_g);
        [start2,emin]=particleswarm(fun,3,lb3(type,:),ub3(type,:));
        c={'area specifica',amin,start1;'consumo specifico',emin,start2};
        writecell(c,filename,'Sheet','2 stadi in
                  cascata','Range',Position2(type))
        FitnessFunction=@(optvar) duestadicascata(type,optvar,dati,4,E_u,E_g);
        opts=optimoptions('gamultiobj','PopulationSize',500,
                          'InitialPopulationMatrix',[start1; start2]);
        [optval3,objval3]=gamultiobj(FitnessFunction,3,[],[],[],[],
                                    lb3(type,:),ub3(type,:),[],opts);
        [objval3(:,1),i]=sort(objval3(:,1));
        objval3(:,2)=objval3(i,2);
        optval3=optval3(i,:);
        Heat_ex_1=duestadicascata(type,optval3(1,:),dati,6,E_u,E_g,Tout,Pt);
        Heat_ex_2=duestadicascata(type,optval3(round(end/2),:),dati,
                                   6,E_u,E_g,Tout,Pt);
        Heat_ex_3=duestadicascata(type,optval3(end,:),dati,6,E_u,E_g,Tout,Pt);
        t=table(Paretofront,Optimumvar(type,:),'VariableNames',
                {'Fronte di Pareto','variabili ottimizzate'});
        writetable(t,filename,'Sheet','2 stadi in
                  cascata','Range',Position3(type))
        writematrix([objval3,optval3],filename,'Sheet','2 stadi in
                  cascata','Range',Position4(type))
        t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',
                {'scambiatore'},'RowNames',{'Tin [°C] amin','Pt [MW/(kgCO2/s)]
                amin','Tin [°C] p_medio','Pt [MW/(kgCO2/s)] p_medio','Tin
                [°C] emin','Pt [MW/(kgCO2/s)] emin'});
        writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
                  cascata','Range',Position5(type))

```

```

plot(objval3(:,1),objval3(:,2), 'linewidth',2, 'DisplayName',
      ['tipo ', num2str(type)])
title('Pareto Front (2 stadi in cascata)')
xlabel('area totale specifica [m^2/(kg_C_0_2/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_0_2/s)]')
legend show
hold on
end
end
%CONFIGURAZIONE 4: 2 stadi in serie con ricircolo
lb4=[112.5093 112.5093 0.0001; 112.5093 p_min 0.0001; p_min 112.5093 0.0001;
p_min p_min 0.0001]; %limiti inferiori delle variabili decisonali
ub4=[p_max p_max 0.9999; p_max 67.5056 0.9999; 37.5031 p_max 0.9999; 37.5031
37.5031 0.9999]; %limiti superiori delle variabili decisonali
writecell({'Tipo 1','2 compressori'},filename,'Sheet','2 stadi in
serie+ricircolo')
writecell({'Tipo 2','compressore + pompa a vuoto'},filename,'Sheet','2 stadi
in serie+ricircolo','Range','G1')
writecell({'Tipo 3','pompa a vuoto + compressore'},filename,'Sheet','2 stadi
in serie+ricircolo','Range','M1')
writecell({'Tipo 4','2 pompe a vuoto'},filename,'Sheet','2 stadi in
serie+ricircolo','Range','S1')
Optimumvar=["p1 alta [cmHg]","p2 alta [cmHg]","recupero stadio2"; "p1 alta
[cmHg]","p2 bassa [cmHg]","recupero stadio2";"p1 bassa
[cmHg]","p2 alta [cmHg]","recupero stadio2";"p1 bassa [cmHg]","p2
bassa [cmHg]","recupero stadio2"];
figure
type=1; %si ottimizza la tipologia 1
fun=@(optvar) duestadiseriercy(type,optvar,dati,1);
[var,obj]=surrogateopt(fun,lb4(type,:),ub4(type,:),[],[1 -1 0],0);
c={'obiettivo','minimo',Optimumvar(1,1),Optimumvar(1,2),Optimumvar(1,3);'purezza',1-obj,var(1),var(2),var(3)};
writecell(c,filename,'Sheet','2 stadi in
serie+ricircolo','Range',Position1(type)) %trascrizione del risultato su excel
if 1-obj<Pmin %verifica sulla purezza minima
disp(['La purezza massima raggiungibile con 2 stadi in serie con ricircolo
(tipologia 1) è del ',num2str((1-obj)*100),'%'])
disp(['La purezza minima del ',num2str(Pmin*100),'% non è soddisfatta'])
else
fun=@(optvar) duestadiseriercy(type,optvar,dati,2);
opts=optimoptions('surrogateopt','MaxFunctionEvaluations',300,
'MinSurrogatePoints',50,'MinSampleDistance',1e-8);
start1=surrogateopt(fun,lb4(type,:),ub4(type,:),[],[1 -1 0],0,[],[],opts);
opts=optimoptions('patternsearch','FunctionTolerance',
1e-10,'MeshTolerance',1e-10,'StepTolerance',1e-10);
[start1,amin]=patternsearch(fun,start1,
[1 -1 0],0,[],[],lb4(type,:),ub4(type,:),[],opts);
fun=@(optvar) duestadiseriercy(type,optvar,dati,3,E_u,E_g);
opts=optimoptions('surrogateopt','MaxFunctionEvaluations',300,
'MinSurrogatePoints',50,'MinSampleDistance',1e-8);
start2=surrogateopt(fun,lb4(type,:),ub4(type,:),[],[1 -1 0],0,[],[],opts);

```

```

opts=optimoptions('patternsearch','PollMethod','GSSPositiveBasis2N',
    'FunctionTolerance',1e-10,'MeshTolerance',1e-10,'StepTolerance',1e-10);
[start2,emin]=patternsearch(fun,start2,
    [1 -1 0],0,[],[],lb4(type,:),ub4(type,:),[],opts);
c={'area specifica',amin,start1;'consumo specifico',emin,start2};
writecell(c,filename,'Sheet','2 stadi in
    serie+ricircolo','Range',Position2(type))
FitnessFunction=@(optvar) duestadiseriercy(type,optvar,dati,4,E_u,E_g);
opts=optimoptions('gamultiobj','PopulationSize',500,
    'InitialPopulationMatrix',[start1; start2]);
[optval4,objval4]=gamultiobj(FitnessFunction,3,
    [1 -1 0],0,[],[],lb4(type,:),ub4(type,:),[],opts);
[objval4(:,1),i]=sort(objval4(:,1));
objval4(:,2)=objval4(i,2);
optval4=optval4(i,:);
Heat_ex_1=duestadiseriercy(type,optval4(1,:),dati,6,E_u,E_g,Tout,Pt);
Heat_ex_2=duestadiseriercy(type,optval4(round(end/2),:),dati,
    6,E_u,E_g,Tout,Pt); %punto medio del fronte
Heat_ex_3=duestadiseriercy(type,optval4(end,:),dati,6,E_u,E_g,Tout,Pt);
t=table(Paretofront,Optimumvar(type,:),'VariableNames',
    {'Fronte di Pareto','variabili ottimizzate'});
writetable(t,filename,'Sheet','2 stadi in
    serie+ricircolo','Range',Position3(type))
writematrix([objval4,optval4],filename,'Sheet','2 stadi in
    serie+ricircolo','Range',Position4(type))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',
    {'scambiatore'},'RowNames',{'Tin [°C] amin','Pt [MW/(kgCO2/s)]
    amin','Tin [°C] p_medio','Pt [MW/(kgCO2/s)] p_medio','Tin [°C]
    emin','Pt [MW/(kgCO2/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
    serie+ricircolo','Range',Position5(type))
plot(objval4(:,1),objval4(:,2),'linewidth',2,'DisplayName','tipo 1')
title('Pareto Front (2 stadi in serie con ricircolo)')
xlabel('area totale specifica [m^2/(kg_C_0_2/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_0_2/s)]')
legend show
hold on
end
for type=2:4
fun=@(optvar) duestadiseriercy(type,optvar,dati,1);
[var,obj]=particleswarm(fun,3,lb4(type,:),ub4(type,:));
c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
    Optimumvar(type,3);'purezza',1-obj,var(1),var(2),var(3)};
writecell(c,filename,'Sheet','2 stadi in
    serie+ricircolo','Range',Position1(type))
if 1-obj<Pmin %verifica sulla purezza minima
    disp(['La purezza massima raggiungibile con 2 stadi in serie con
        ricircolo (tipologia ',num2str(type),') è del ',
        num2str((1-obj)*100),'%'])
    disp(['La purezza minima del ',num2str(Pmin*100),
        '% non è soddisfatta'])
end
end

```

```

else
    fun=@(optvar) duestadiseriercy(type,optvar,dati,2);
    opts=optimoptions('particleswarm','SwarmSize',80,
                     'FunctionTolerance',1e-8);
    [start1,amin]=particleswarm(fun,3,lb4(type,:),ub4(type,:),opts);
    fun=@(optvar) duestadiseriercy(type,optvar,dati,3,E_u,E_g);
    [start2,emin]=particleswarm(fun,3,lb4(type,:),ub4(type,:),opts);
    c={'area specifica',amin,start1;'consumo specifico',emin,start2};
    writecell(c,filename,'Sheet','2 stadi in
              serie+ricircolo','Range',Position2(type))
    FitnessFunction=@(optvar) duestadiseriercy(type,optvar,dati,
                                                4,E_u,E_g);
    opts=optimoptions('gamultiobj','PopulationSize',500,
                     'InitialPopulationMatrix',[start1; start2]);
    [optval4,objval4]=gamultiobj(FitnessFunction,3,[],[],[],[],
                                lb4(type,:),ub4(type,:),[],opts);
    [objval4(:,1),i]=sort(objval4(:,1));
    objval4(:,2)=objval4(i,2);
    optval4=optval4(i,:);
    Heat_ex_1=duestadiseriercy(type,optval4(1,:),dati,6,E_u,E_g,Tout,Pt);
    Heat_ex_2=duestadiseriercy(type,optval4(round(end/2),:),dati,6,
                                E_u,E_g,Tout,Pt); %punto medio del fronte
    Heat_ex_3=duestadiseriercy(type,optval4(end,:),dati,6,
                                E_u,E_g,Tout,Pt); %condizione di consumi minimi
    t=table(Paretofront,Optimumvar(type,:),'VariableNames',
            {'Fronte di Pareto','variabili ottimizzate'});
    writetable(t,filename,'Sheet','2 stadi in
              serie+ricircolo','Range',Position3(type))
    writematrix([objval4,optval4],filename,'Sheet','2 stadi in
              serie+ricircolo','Range',Position4(type))
    t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',
            {'scambiatore'},'RowNames',{'Tin [°C] amin','Pt [MW/(kgCO2/s)]
            amin','Tin [°C] p_medio','Pt [MW/(kgCO2/s)] p_medio','Tin [°C]
            emin','Pt [MW/(kgCO2/s)] emin'});
    writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
              serie+ricircolo','Range',Position5(type))
    plot(objval4(:,1),objval4(:,2),'linewidth',2,'DisplayName',
         ['tipo ',num2str(type)])
    title('Pareto Front (2 stadi in serie con ricircolo)')
    xlabel('area totale specifica [m^2/(kg_C_0_2/s)]')
    ylabel('consumo elettrico specifico [MW/(kg_C_0_2/s)]')
    legend show
    hold on
end
end
%CONFIGURAZIONE 5: 2 stadi in cascata con ricircolo.
lb5=[112.5093 R*1.0001 112.5093; 112.5093 R*1.0001 p_min; p_min R*1.0001
     112.5093; p_min R*1.0001 p_min]; %valori limite inferiore per le
                                     variabili da ottimizzare
ub5=[p_max 0.9999 p_max; p_max 0.9999 37.5031; 37.5031 0.9999 p_max; 37.5031
     0.9999 37.5031]; %valori limite superiore per le variabili da ottimizzare

```

```

writecell({'Tipo 1','2 compressori'},filename,'Sheet','2 stadi in
          cascata+ricircolo')
writecell({'Tipo 2','compressore + pompa a vuoto'},filename,'Sheet','2 stadi
          in cascata+ricircolo','Range','G1')
writecell({'Tipo 3','pompa a vuoto + compressore'},filename,'Sheet','2 stadi
          in cascata+ricircolo','Range','M1')
writecell({'Tipo 4','2 pompe a vuoto'},filename,'Sheet','2 stadi in
          cascata+ricircolo','Range','S1')
Optimumvar=["p1 alta [cmHg]","recupero stadio1","p2 alta [cmHg]"; "p1 alta
          [cmHg]","recupero stadio1","p2 bassa [cmHg]";"p1 bassa
          [cmHg]","recupero stadio1","p2 alta [cmHg]";"p1 bassa
          [cmHg]","recupero stadio1","p2 bassa [cmHg]"];

figure
type=1;
fun=@(optvar) duestadicascatarcy(type,optvar,dati,1);
[var,obj]=surrogateopt(fun,lb5(type,:),ub5(type,:),[],[1 0 -1],0);
c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
  Optimumvar(type,3);'purezza',1-obj,var(1),var(2),var(3)};
writecell(c,filename,'Sheet','2 stadi in
          cascata+ricircolo','Range',Position1(type))
if 1-obj<Pmin
  disp(['La purezza massima raggiungibile con 2 stadi in cascata con
        ricircolo (tipologia 1) è del ',num2str((1-obj)*100),'%'])
  disp(['La purezza minima del ',num2str(Pmin*100),'% non è soddisfatta'])
else
  fun=@(optvar) duestadicascatarcy(type,optvar,dati,2);
  start1=surrogateopt(fun,lb5(type,:),ub5(type,:),[],[1 0 -1],0);
  [start1,amin]=patternsearch(fun,start1,
    [1 0 -1],0,[],[],lb5(type,:),ub5(type,:));
  fun=@(optvar) duestadicascatarcy(type,optvar,dati,3,E_u,E_g);
  opts=optimoptions('surrogateopt','MaxFunctionEvaluations',200,
    'MinSurrogatePoints',50,'MinSampleDistance',1e-8);
  [start2,emin]=surrogateopt(fun,lb5(type,:),ub5(type,:),[],
    [1 0 -1],0,[],[],opts);
  c={'area specifica',amin,start1;'consumo specifico',emin,start2};
  writecell(c,filename,'Sheet','2 stadi in
          cascata+ricircolo','Range',Position2(type))
  FitnessFunction=@(optvar) duestadicascatarcy(type,optvar,dati,4,E_u,E_g);
  opts=optimoptions('gamultiobj','PopulationSize',500,
    'InitialPopulationMatrix',[start1; start2]);
  [optval5,objval5]=gamultiobj(FitnessFunction,3,
    [1 0 -1],0,[],[],lb5(type,:),ub5(type,:),[],opts);
  [objval5(:,1),i]=sort(objval5(:,1));
  objval5(:,2)=objval5(i,2);
  optval5=optval5(i,:);
  Heat_ex_1=duestadicascatarcy(type,optval5(1,:),dati,6,E_u,E_g,Tout,Pt);
  Heat_ex_2=duestadicascatarcy(type,optval5(round(end/2),:),dati,6,E_u,
    E_g,Tout,Pt); %punto medio del fronte
  Heat_ex_3=duestadicascatarcy(type,optval5(end,:),dati,6,E_u,E_g,Tout,Pt);
  t=table(Paretofront,Optimumvar(type,:),'VariableNames',
    {'Fronte di Pareto','variabili ottimizzate'});

```

```

writetable(t,filename,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position3(type))
writematrix([objval5,optval5],filename,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position4(type))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',
        {'scambiatore'},'RowNames',{'Tin [°C] amin','Pt [MW/(kgCO2/s)]
        amin','Tin [°C] p_medio','Pt [MW/(kgCO2/s)] p_medio','Tin [°C]
        emin','Pt [MW/(kgCO2/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position5(type))
plot(objval5(:,1),objval5(:,2),'linewidth',2,'DisplayName','tipo 1')
title('Pareto Front (2 stadi in cascata con ricircolo)')
xlabel('area totale specifica [m^2/(kg_C_O_2/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_O_2/s)]')
legend show
hold on
end
for type=2:4
    fun=@(optvar) duestadicascatarcy(type,optvar,dati,1);
    [var,obj]=particleswarm(fun,3,lb5(type,:),ub5(type,:));
    c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
        Optimumvar(type,3);'purezza',1-obj,var(1),var(2),var(3)};
    writecell(c,filename,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position1(type))
    if 1-obj<Pmin %verifica sulla purezza minima desiderata
        disp(['La purezza massima raggiungibile con 2 stadi in cascata con
                ricircolo (tipologia ',num2str(type),' ) è del ',num2str((1-
                obj)*100),'%'])
        disp(['La purezza minima del ',num2str(Pmin*100),'% non è
                soddisfatta'])
    else
        fun=@(optvar) duestadicascatarcy(type,optvar,dati,2);
        opts=[optimoptions('particleswarm');
            optimoptions('particleswarm','SwarmSize',60,'FunctionTolerance',1e-7);
            optimoptions('particleswarm','SwarmSize',70,
                'FunctionTolerance',1e-7)];
        [start1,amin]=particleswarm(fun,3,lb5(type,:),ub5(type,:));
        fun=@(optvar) duestadicascatarcy(type,optvar,dati,3,E_u,E_g);
        [start2,emin]=particleswarm(fun,3,lb5(type,:),ub5(type,:),
            opts(type-1));
        c={'area specifica',amin,start1;'consumo specifico',emin,start2};
        writecell(c,filename,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position2(type))
        FitnessFunction=@(optvar) duestadicascatarcy(type,optvar,dati,
            4,E_u,E_g);
        opts=optimoptions('gamultiobj','PopulationSize',500,
            'InitialPopulationMatrix',[start1; start2]);
        [optval5,objval5]=gamultiobj(FitnessFunction,3,[],[],[],[],
            lb5(type,:),ub5(type,:),[],opts);
        [objval5(:,1),i]=sort(objval5(:,1));
        objval5(:,2)=objval5(i,2);
    end
end

```

```

optval5=optval5(i,:);
Heat_ex_1=dustadicascatarcy(type,optval5(1,:),dati,6,E_u,
                             E_g,Tout,Pt); %coondizione con area minima
Heat_ex_2=dustadicascatarcy(type,optval5(round(end/2),:),dati,6,
                             E_u,E_g,Tout,Pt); %punto medio del fronte
Heat_ex_3=dustadicascatarcy(type,optval5(end,:),dati,6,
                             E_u,E_g,Tout,Pt); %condizione con consumi minimi
t=table(Paretofront,Optimumvar(type,:), 'VariableNames',
        {'Fronte di Pareto', 'variabili ottimizzate'});
writetable(t,filename,'Sheet','2 stadi in
            cascata+ricircolo','Range',Position3(type))
writematrix([objval5,optval5],filename,'Sheet','2 stadi in
            cascata+ricircolo','Range',Position4(type))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3], 'VariableNames',
        {'scambiatore'}, 'RowNames',{'Tin [°C] amin', 'Pt [MW/(kgCO2/s)]
        amin', 'Tin [°C] p_medio', 'Pt [MW/(kgCO2/s)] p_medio', 'Tin [°C]
        emin', 'Pt [MW/(kgCO2/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
            cascata+ricircolo','Range',Position5(type))
plot(objval5(:,1),objval5(:,2), 'linewidth',2, 'DisplayName',
      ['tipo ',num2str(type)])
title('Pareto Front (2 stadi in cascata con ricircolo)')
xlabel('area totale specifica [m^2/(kg_C_0_2/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_0_2/s)]')
legend show
hold on
end
end
toc

```

Appendice G

Si riporta il codice utilizzato per l'ottimizzazione di un sistema di separazione alimentato con il biogas proveniente da un impianto di digestione anaerobica (miscela binaria CH₄-CO₂). Si utilizzano le funzioni `duestadiserie.m`, `duesticascata.m`, `duestadiseriercy.m`, `duesticascatarcy.m` (APPENDICE E) per risolvere le equazioni di bilancio e di trasporto di ciascuna configurazione analizzata. Il codice fornisce come *output* i grafici dei fronti di Pareto, e un file Excel in cui sono salvati tutti i risultati dell'ottimizzazione e le potenze termiche degli scambiatori calcolate.

```
clear all
close all
clc

tic
%creazione del file excel su cui verranno copiati i risultati
filename='simulazione.xlsx';

p_feed=1*75.0062; %pressione del gas alimentato [cmHg]
p_min=0.2*75.0062; %limite inferiore del grado di vuoto raggiungibile [cmHg]
p_max=10*75.0062; %pressione massima raggiungibile lato feed [cmHg]
permCO2=86.3e-6; %permeanza della CO2 [cmSTP^3/(cm^2*s*cmHg)]
permCH4=2.6e-6; %permeanza del CH4 [cmSTP^3/(cm^2*s*cmHg)]
Vm=22414; %volume molare [cmSTP^3/mol]
Q_f=12.4*Vm; %portata alimentata [cmSTP^3 s^-1]
x_f=0.4; %frazione molare di CO2 nel feed
PCH4=0.975; %purezza del CH4 desiderata nel retentato
x_r=1-PCH4; %frazione residua di CO2 nel retentato
l=1; %spessore membrana [cm] in questo caso non utilizzato
T=30+273.15; %temperatura operativa [K]

cp_CO2=@(T) 5.457+1.045e-3*T-1.157e5*T.^(-2); %calore specifico della CO2
                                               %diviso per la costante dei gas
cp_CH4=@(T) 1.702+9.081e-3*T-2.164e-6*T.^2; %calore specifico del CH4 diviso
                                               %per la costante dei gas
Trange=linspace(298,600,1000); %range operativo di temperatura ipotizzato, a
                                %causa della compressione ed espansione del gas
cp=mean([cp_CO2(Trange)',cp_CH4(Trange)']);
gamma=@(y) (y*cp(1)+(1-y)*cp(2))/(y*(cp(1)-1)+(1-y)*(cp(2)-1));
Rconst=8.314; %costante dei gas ideali [J/molK]
eta=0.85; %efficienza di compressori, pompe a vuoto ed espansori
E_u=@(Q,y,v) Q*gamma(y)*Rconst*T/(Vm*eta*(gamma(y)-1))*(v^((gamma(y)-
1)/gamma(y))-1)*10^(-6); %potenza utilizzata da compressori e
                                %pompe a vuoto %[MW]
E_g=@(Q,y,v) eta*Q*gamma(y)*Rconst*T/(Vm*(gamma(y)-1))*(1-v^((gamma(y)-
1)/gamma(y)))*10^(-6); %potenza guadagnata con l'espansione del
                                %retentato [MW]
Tout=@(y,v) T*v^((gamma(y)-1)/gamma(y)); %temperatura in seguito a
                                %compressione/espansione, in entrata allo scambiatore [K]
```

```

Pt=@(Q,y,v,Tout) Q/Vm*(mean(cp_CO2(linspace(T,Tout)))*y+
    mean(cp_CH4(linspace(T,Tout)))*(1-y))*Rconst*T*(v^((gamma(y)-
    1)/gamma(y))-1)/eta*10^(-6); %potenza termica scambiata nello
    scambiatore di calore [MW]
LHV=50*0.7168*10^(-6); %potere calorifico inferiore del metano [MJ/cmSTP^3]

dati=[p_feed permCO2 permCH4 Q_f x_f x_r 1];
t=table(dati(1),dati(2),dati(3),dati(4),dati(5),dati(6),dati(7),
    'VariableNames',{'pressione feed [cmHg]', 'permCO2 x 10^(-6)
    [GPU]', 'permCH4 x 10^(-6) [GPU]', 'portata feed [cmSTP^3/s]', 'frazione
    CO2 feed', 'frazione CO2 retentato', 'spessore membrana
    [cm]'}, 'RowNames',{'dati assegnati'})
writetable(t,filename,'WriteRowNames',true,'Sheet','uno stadio')
%CONFIGURAZIONE 1: è il singolo stadio.
[y_p1a,Q_p1a,Q_r1a,R_1a,A_1a]=countercurrentmodel(p_feed,p_min,permCO2,
    permCH4,Q_f,x_f,x_r,1);
disp(['Il recupero di metano con un singolo stadio a bassa pressione risulta
    essere ',num2str(R_1a*100),'%'])
disp(['La potenza perduta con la mancata combustione del metano permeato è
    di ',num2str(LHV*Q_p1a*(1-y_p1a)), ' [MW]'])
a_1a=A_1a/(PCH4*Q_r1a*0.01604/Vm)*10^(-4); %area specifica [m^2/(kg_CH4/s)]
e_1a=(E_u(Q_p1a,y_p1a,75.0062/p_min)+LHV*Q_p1a*(1-
    y_p1a))/(Q_r1a*PCH4*0.01604/Vm); %consumo energetico specifico
    [MW/(kgCH4/s)]
[y_p1b,Q_p1b,Q_r1b,R_1b,A_1b]=countercurrentmodel(p_max,75.0062,permCO2,
    permCH4,Q_f,x_f,x_r,1);
disp(['Il recupero di metano con un singolo stadio ad alta pressione risulta
    essere ',num2str(R_1b*100),'%'])
disp(['La potenza perduta con la mancata combustione del metano permeato è
    di ',num2str(LHV*Q_p1b*(1-y_p1b)), ' [MW]'])
a_1b=A_1b/(PCH4*Q_r1b*0.01604/Vm)*10^(-4); %area specifica [m^2/(kg_CH4/s)]
e_1b=(E_u(Q_f,x_f,p_max/p_feed)-E_g(Q_r1b,x_r,75.0062/p_max)+LHV*Q_p1b*(1-
    y_p1b))/(Q_r1b*PCH4*0.01604/Vm); %consumo energetico specifico [MW/(kgCH4/s)]
Tout1b=Tout(x_f,p_max/p_feed); %temperatura in uscita del compressore [K]
Pt1b=Pt(Q_f,x_f,p_max/p_feed,Tout1b)/(Q_r1b*PCH4*0.01604/Vm); %potenza termica
t=table([p_feed;p_max],[p_min;75.0062],[R_1a;R_1b],[a_1a;a_1b],[e_1a;e_1b],
    'VariableNames',{'p lato feed [cmHg]', 'p lato perm
    [cmHg]', 'recupero', 'area specifica [m^2/(kgCH4/s)]', 'consumo specifico
    [MW/(kgCH4/s)]'}, 'RowNames',{'singolo stadio bassa p', 'singolo stadio
    alta p'}, 'DimensionNames',{'configurazioni', 'variabili'})
writetable(t,filename,'WriteRowNames',true,'Sheet','uno stadio','Range','A6')
c={'Tout compressione [°C]', 'Potenza termica [MW/(kgCH4/s)]';'', ''};
    Tout1b-273.15,Pt1b);
writecell(c,filename,'Sheet','uno stadio','Range','H6')
%CONFIGURAZIONE 2: 2 stadi in serie senza ricircolo.
lb2=[112.5093 x_r*1.001 112.5093; 112.5093 x_r*1.001 p_min; p_min x_r*1.001
    112.5093; p_min x_r*1.001 p_min]; %valori limite inferiore delle
    variabili decisionali
ub2=[p_max x_f*0.999 p_max; p_max x_f*0.999 67.5056; 37.5031 x_f*0.999
    p_max;37.5031 x_f*0.999 37.5031]; %valori limite superiore delle
    variabili decisionali

```

```

writecell({'Tipo 1','2 compressori'},filename,'Sheet','2 stadi in serie')
writecell({'Tipo 2','compressore + pompa a vuoto'},filename,'Sheet','2 stadi
in serie','Range','G1')
writecell({'Tipo 3','pompa a vuoto + compressore'},filename,'Sheet','2 stadi
in serie','Range','M1')
writecell({'Tipo 4','2 pompe a vuoto'},filename,'Sheet','2 stadi in
serie','Range','S1')

Position1=["A3","G3","M3","S3"];
Position2=["A4","G4","M4","S4"];
Position3=["A9","G9","M9","S9"];
Position4=["A11","G11","M11","S11"];
Position5=["B187","H187","N187","T187"];
Paretofront={'area specifica [m^2/(kgCH4/s)]','consumo specifico
[MW/(kgCH4/s)]'};

Optimumvar=["p1 alta [cmHg]","x_r1","p2 alta [cmHg]"; "p1 alta
[cmHg]","x_r1","p2 bassa [cmHg]"; "p1 bassa [cmHg]","x_r1","p2 alta
[cmHg]"; "p1 bassa [cmHg]","x_r1","p2 bassa [cmHg]"];

figure
opts=optimoptions('ga','PopulationSize',100);
fun=@(optvar) duestadiserie(1,optvar,dati,1);
[start1,amin]=ga(fun,3,[1 0 -1],0,[],[],lb2(1,:),ub2(1,:),[],opts);
fun=@(optvar) duestadiserie(1,optvar,dati,2,E_u,E_g,LHV);
[start2,emin]=ga(fun,3,[1 0 -1],0,[],[],lb2(1,:),ub2(1,:),[],opts);
c={'obiettivo','minimo',Optimumvar(1,1),Optimumvar(1,2),Optimumvar(1,3)};
writecell(c,filename,'Sheet','2 stadi in serie','Range',Position1(1))
c={'area specifica [m^2/(kg_CH4/s)]',amin,start1;'consumo specifico
[MW/(kg_CH4/s)]',emin,start2};
writecell(c,filename,'Sheet','2 stadi in serie','Range',Position2(1))
FitnessFunction=@(optvar) duestadiserie(1,optvar,dati,3,E_u,E_g,LHV);
opts=optimoptions('gamultiobj','PopulationSize',500,
'InitialPopulationMatrix',[start1; start2]);
[optval2,objval2]=gamultiobj(FitnessFunction,3,
[1 0 -1],0,[],[],lb2(1,:),ub2(1,:),[],opts);
[objval2(:,1),i]=sort(objval2(:,1));
objval2(:,2)=objval2(i,2);
optval2=optval2(i,:);
Heat_ex_1=duestadiserie(1,optval2(1,:),dati,5,E_u,E_g,LHV,Tout,Pt);
Heat_ex_2=duestadiserie(1,optval2(round(end/2),:),dati,5,E_u,E_g,LHV,Tout,Pt);
Heat_ex_3=duestadiserie(1,optval2(end,:),dati,5,E_u,E_g,LHV,Tout,Pt);
t=table(Paretofront,Optimumvar(1,:),'VariableNames',{'Fronte di
Pareto','variabili ottimizzate'});
writetable(t,filename,'Sheet','2 stadi in serie','Range',Position3(1))
writematrix([objval2,optval2],filename,'Sheet','2 stadi in
serie','Range',Position4(1))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',{'scambiatore'},
'RowNames',{'Tin [°C] amin','Pt [MW/(kgCH4/s)] amin','Tin [°C]
p_medio','Pt [MW/(kgCH4/s)] p_medio','Tin [°C] emin','Pt
[MW/(kgCH4/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
serie','Range',Position5(1))
plot(objval2(:,1),objval2(:,2),'linewidth',2,'DisplayName','tipo 1')

```

```

title('Pareto Front (2 stadi in serie)')
xlabel('area totale specifica [m^2/(kg_C_H_4/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_H_4/s)]')
legend show
hold on
for type=2:4
    c={'obiettivo', 'minimo', Optimumvar(type,1), Optimumvar(type,2),
                                           Optimumvar(type,3)};
    writecell(c,filename, 'Sheet', '2 stadi in serie', 'Range', Position1(type))
    fun=@(optvar) duestadiserie(type,optvar,dati,1);
    [start1,amin]=particleswarm(fun,3,lb2(type,:),ub2(type,:));
    fun=@(optvar) duestadiserie(type,optvar,dati,2,E_u,E_g,LHV);
    [start2,emin]=particleswarm(fun,3,lb2(type,:),ub2(type,:));
    c={'area specifica [m^2/(kg_CH4/s)]',amin,start1; 'consumo specifico
                                           [MW/(kg_CH4/s)]',emin,start2};
    writecell(c,filename, 'Sheet', '2 stadi in serie', 'Range', Position2(type))
    FitnessFunction=@(optvar) duestadiserie(type,optvar,dati,3,E_u,E_g,LHV);
    opts=optimoptions('gamultiobj', 'PopulationSize', 500,
                      'InitialPopulationMatrix', [start1; start2]);
    [optval2,objval2]=gamultiobj(FitnessFunction,3,[],[],[],[],lb2(type,:),
                                ub2(type,:),[],opts);

    [objval2(:,1),i]=sort(objval2(:,1));
    objval2(:,2)=objval2(i,2);
    optval2=optval2(i,:);
    t=table(Paretofront,Optimumvar(type,:), 'VariableNames',
            {'Fronte di Pareto', 'variabili ottimizzate'});
    writetable(t,filename, 'Sheet', '2 stadi in serie', 'Range', Position3(type))
    writematrix([objval2,optval2],filename, 'Sheet', '2 stadi in
                                                serie', 'Range', Position4(type))
    if type==2 || type==3 %per n=4 non vengono usati scambiatori
        Heat_ex_1=duestadiserie(type,optval2(1,:),dati,5,E_u,E_g,LHV,Tout,Pt);
        Heat_ex_2=duestadiserie(type,optval2(round(end/2),:),dati,5,E_u,E_g,
                                LHV,Tout,Pt);

        Heat_ex_3=duestadiserie(type,optval2(end,:),dati,5,E_u,E_g,
                                LHV,Tout,Pt);

        t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3], 'VariableNames',
                {'scambiatore'}, 'RowNames', {'Tin [°C] amin', 'Pt
                [MW/(kgCH4/s)] amin', 'Tin [°C] p_medio', 'Pt [MW/(kgCH4/s)]
                p_medio', 'Tin [°C] emin', 'Pt [MW/(kgCH4/s)] emin'});
        writetable(t,filename, 'WriteRowNames', true, 'Sheet', '2 stadi in
                                                serie', 'Range', Position5(type))
    end
    plot(objval2(:,1),objval2(:,2), 'linewidth', 2, 'DisplayName',
        ['tipo ', num2str(type)])
    title('Pareto Front (2 stadi in serie)')
    xlabel('area totale specifica [m^2/(kg_C_H_4/s)]')
    ylabel('consumo elettrico specifico [MW/(kg_C_H_4/s)]')
    legend show
    hold on
end

```

```

%CONFIGURAZIONE 3: 2 stadi in cascata senza ricircolo
lb3=[112.5093 0.001 112.5093; 112.5093 0.001 p_min; p_min 0.001 112.5093;
      p_min 0.001 p_min]; %valori limite inferiore delle variabili decisionali
ub3=[p_max x_r p_max; p_max x_r 37.5031; 37.5031 x_r p_max; 37.5031 x_r
      37.5031]; %valori limite superiore delle variabili decisionali
writecell({'Tipo 1','2 compressori'},filename,'Sheet','2 stadi in cascata')
writecell({'Tipo 2','compressore + pompa a vuoto'},filename,'Sheet','2 stadi
          in cascata','Range','G1')
writecell({'Tipo 3','pompa a vuoto + compressore'},filename,'Sheet','2 stadi
          in cascata','Range','M1')
writecell({'Tipo 4','2 pompe a vuoto'},filename,'Sheet','2 stadi in
          cascata','Range','S1')

figure
for type=1:4
    c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
      Optimumvar(type,3)};
    writecell(c,filename,'Sheet','2 stadi in cascata','Range',Position1(type))
    fun=@(optvar) duestadicascata(type,optvar,dati,1);
    [start1,amin]=particleswarm(fun,3,lb3(type,:),ub3(type,:));
    fun=@(optvar) duestadicascata(type,optvar,dati,2,E_u,E_g,LHV);
    [start2,emin]=particleswarm(fun,3,lb3(type,:),ub3(type,:));
    c={'area specifica',amin,start1;'consumo specifico',emin,start2};
    writecell(c,filename,'Sheet','2 stadi in cascata','Range',Position2(type))
    FitnessFunction=@(optvar) duestadicascata(type,optvar,dati,3,E_u,E_g,LHV);
    opts=optimoptions('gamultiobj','PopulationSize',500,
      'InitialPopulationMatrix',[start1; start2]);
    [optval3,objval3]=gamultiobj(FitnessFunction,3,[],[],[],[],
      lb3(type,:),ub3(type,:),[],opts);
    [objval3(:,1),i]=sort(objval3(:,1));
    objval3(:,2)=objval3(i,2);
    optval3=optval3(i,:);
    Heat_ex_1=duestadicascata(type,optval3(1,:),dati,5,E_u,E_g,LHV,Tout,Pt);
    Heat_ex_2=duestadicascata(type,optval3(round(end/2),:),dati,5,E_u,
      E_g,LHV,Tout,Pt);
    Heat_ex_3=duestadicascata(type,optval3(end,:),dati,5,E_u,E_g,LHV,Tout,Pt);
    t=table(Paretofront,Optimumvar(type,:), 'VariableNames',{'Fronte di
      Pareto','variabili ottimizzate'});
    writetable(t,filename,'Sheet','2 stadi in
      cascata','Range',Position3(type))
    writematrix([objval3,optval3],filename,'Sheet','2 stadi in
      cascata','Range',Position4(type))
    t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3], 'VariableNames',
      {'scambiatore'}, 'RowNames',{'Tin [°C] amin','Pt [MW/(kgCH4/s)]
      amin','Tin [°C] p_medio','Pt [MW/(kgCH4/s)] p_medio','Tin [°C]
      emin','Pt [MW/(kgCH4/s)] emin'});
    writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
      cascata','Range',Position5(type))
    plot(objval3(:,1),objval3(:,2),'linewidth',2,'DisplayName',
      ['tipo ',num2str(type)])
    title('Pareto Front (2 stadi in cascata)')
    xlabel('area totale specifica [m^2/(kg_C_H_4/s)]')

```

```

ylabel('consumo elettrico specifico [MW/(kg_C_H_4/s)]')
legend show
hold on
end
%CONFIGURAZIONE 4: 2 stadi in serie con ricircolo.
lb4=[112.5093 x_r*1.001 112.5093; 112.5093 x_r*1.001 p_min; p_min x_r*1.001
     112.5093; p_min x_r*1.001 p_min]; %valori di limite inferiore delle
                                     variabili decisionali
ub4=[p_max 0.9 p_max; p_max 0.9 67.5056; 37.5031 0.9 p_max; 37.5031 0.9
     37.5031]; %valori di limite superiore delle variabili decisionali
writecell({'Tipo 1','2 compressori'},filename,'Sheet','2 stadi in
          serie+ricircolo')
writecell({'Tipo 2','compressore + pompa a vuoto'},filename,'Sheet','2 stadi
          in serie+ricircolo','Range','G1')
writecell({'Tipo 3','pompa a vuoto + compressore'},filename,'Sheet','2 stadi
          in serie+ricircolo','Range','M1')
writecell({'Tipo 4','2 pompe a vuoto'},filename,'Sheet','2 stadi in
          serie+ricircolo','Range','S1')

figure
c={'obiettivo','minimo',Optimumvar(1,1),Optimumvar(1,2),Optimumvar(1,3)};
writecell(c,filename,'Sheet','2 stadi in
          serie+ricircolo','Range',Position1(1))
fun=@(optvar) duestadiseriercy(1,optvar,dati,1);
start1=surrogateopt(fun,lb4(1,:),ub4(1,:),[],[1 0 -1],0);
[start1,amin]=patternsearch(fun,start1,[1 0 -1],0,[],[],lb4(1,:),ub4(1,:));
fun=@(optvar) duestadiseriercy(1,optvar,dati,2,E_u,E_g,LHV);
opts=optimoptions('surrogateopt','MaxFunctionEvaluations',300,
                  'MinSurrogatePoints',50,'MinSampleDistance',1e-8);
start2=surrogateopt(fun,lb4(1,:),ub4(1,:),[],[1 0 -1],0,[],[],opts);
opts=optimoptions('patternsearch','FunctionTolerance',
                  1e-10,'MeshTolerance',1e-10,'StepTolerance',1e-10);
[start2,emin]=patternsearch(fun,start2,
                             [1 0 -1],0,[],[],lb4(1,:),ub4(1,:),[],opts);
c={'area specifica',amin,start1;'consumo specifico',emin,start2};
writecell(c,filename,'Sheet','2 stadi in
          serie+ricircolo','Range',Position2(1))
FitnessFunction=@(optvar) duestadiseriercy(1,optvar,dati,3,E_u,E_g,LHV);
opts=optimoptions('gamultiobj','PopulationSize',500,
                  'InitialPopulationMatrix',[start1; start2]);
[optval4,objval4]=gamultiobj(FitnessFunction,3,
                             [1 0 -1],0,[],[],lb4(1,:),ub4(1,:),[],opts);
[objval4(:,1),i]=sort(objval4(:,1));
objval4(:,2)=objval4(i,2);
optval4=optval4(i,:);
Heat_ex_1=duestadiseriercy(1,optval4(1,:),dati,5,E_u,E_g,LHV,Tout,Pt);
Heat_ex_2=duestadiseriercy(1,optval4(round(end/2),:),dati,5,E_u,E_g,LHV,
                             Tout,Pt);
Heat_ex_3=duestadiseriercy(1,optval4(end,:),dati,5,E_u,E_g,LHV,Tout,Pt);
t=table(Paretofront,Optimumvar(1,:), 'VariableNames',{'Fronte di
          Pareto','variabili ottimizzate'});

```

```

writetable(t,filename,'Sheet','2 stadi in
                serie+ricircolo','Range',Position3(1))
writematrix([objval4,optval4],filename,'Sheet','2 stadi in
                serie+ricircolo','Range',Position4(1))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',{'scambiatore'},
        'RowNames',{'Tin [°C] amin','Pt [MW/(kgCH4/s)] amin','Tin [°C]
        p_medio','Pt [MW/(kgCH4/s)] p_medio','Tin [°C] emin','Pt
        [MW/(kgCH4/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
                serie+ricircolo','Range',Position5(1))
plot(objval4(:,1),objval4(:,2),'linewidth',2,'DisplayName','tipo 1')
title('Pareto Front (2 stadi in serie con ricircolo)')
xlabel('area totale specifica [m^2/(kg_C_H_4/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_H_4/s)]')
legend show
hold on
for type=2:4
    c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
        Optimumvar(type,3)};
    writecell(c,filename,'Sheet','2 stadi in
                serie+ricircolo','Range',Position1(type))
    fun=@(optvar) duestadiseriercy(type,optvar,dati,1);
    opts=optimoptions('particleswarm','SwarmSize',60,
        'FunctionTolerance',1e-8);
    [start1,amin]=particleswarm(fun,3,lb4(type,:),ub4(type,:),opts);
    fun=@(optvar) duestadiseriercy(type,optvar,dati,2,E_u,E_g,LHV);
    [start2,emin]=particleswarm(fun,3,lb4(type,:),ub4(type,:),opts);
    c={'area specifica',amin,start1;'consumo specifico',emin,start2};
    writecell(c,filename,'Sheet','2 stadi in
                serie+ricircolo','Range',Position2(type))
    FitnessFunction=@(optvar) duestadiseriercy(type,optvar,dati,3,E_u,
        E_g,LHV);
    opts=optimoptions('gamultiobj','PopulationSize',500,
        'InitialPopulationMatrix',[start1; start2]);
    [optval4,objval4]=gamultiobj(FitnessFunction,3,[],[],[],[],
        lb4(type,:),ub4(type,:),[],opts);
    [objval4(:,1),i]=sort(objval4(:,1));
    objval4(:,2)=objval4(i,2);
    optval4=optval4(i,:);
    Heat_ex_1=duestadiseriercy(type,optval4(1,:),dati,5,E_u,E_g,LHV,Tout,Pt);
    Heat_ex_2=duestadiseriercy(type,optval4(round(end/2),:),dati,5,E_u,
        E_g,LHV,Tout,Pt);
    Heat_ex_3=duestadiseriercy(type,optval4(end,:),dati,5,E_u,E_g,
        LHV,Tout,Pt);
    t=table(Paretofront,Optimumvar(type,:), 'VariableNames',{'Fronte di
        Pareto','variabili ottimizzate'});
    writetable(t,filename,'Sheet','2 stadi in
                serie+ricircolo','Range',Position3(type))
    writematrix([objval4,optval4],filename,'Sheet','2 stadi in
                serie+ricircolo','Range',Position4(type))

```

```

t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3], 'VariableNames', {'scambiatore'},
        'RowNames', {'Tin [°C] amin', 'Pt [MW/(kgCH4/s)] amin', 'Tin [°C]
        p_medio', 'Pt [MW/(kgCH4/s)] p_medio', 'Tin [°C] emin', 'Pt
        [MW/(kgCH4/s)] emin'});
writetable(t,filename, 'WriteRowNames', true, 'Sheet', '2 stadi in
        serie+ricircolo', 'Range', Position5(type))
plot(objval4(:,1),objval4(:,2), 'linewidth', 2, 'DisplayName',
        ['tipo ', num2str(type)])
title('Pareto Front (2 stadi in serie con ricircolo)')
xlabel('area totale specifica [m^2/(kg_C_H_4/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_H_4/s)]')
legend show
hold on
end
%CONFIGURAZIONE 5: 2 stadi in cascata con ricircolo.
lb5=[112.5093 112.5093 0.001; 112.5093 p_min 0.001; p_min 112.5093 0.001;
        p_min p_min 0.001]; %valori di limite inferiore delle variabili
        decisionali
ub5=[p_max p_max 0.99; p_max 37.5031 0.99; 37.5031 p_max 0.99; 37.5031 37.5031
        0.99]; %valori di limite superiore delle variabili decisionali
writecell({'Tipo 1', '2 compressori'}, filename, 'Sheet', '2 stadi in
        cascata+ricircolo')
writecell({'Tipo 2', 'compressore + pompa a vuoto'}, filename, 'Sheet', '2 stadi
        in cascata+ricircolo', 'Range', 'G1')
writecell({'Tipo 3', 'pompa a vuoto + compressore'}, filename, 'Sheet', '2 stadi
        in cascata+ricircolo', 'Range', 'M1')
writecell({'Tipo 4', '2 pompe a vuoto'}, filename, 'Sheet', '2 stadi in
        cascata+ricircolo', 'Range', 'S1')
Optimumvar=["p1 alta [cmHg]", "p2 alta [cmHg]", "x_r2"; "p1 alta [cmHg]", "p2
        bassa [cmHg]", "x_r2"; "p1 bassa [cmHg]", "p2 alta [cmHg]", "x_r2"; "p1
        bassa [cmHg]", "p2 bassa [cmHg]", "x_r2"];
figure
c={'obiettivo', 'minimo', Optimumvar(1,1), Optimumvar(1,2), Optimumvar(1,3)};
writecell(c, filename, 'Sheet', '2 stadi in
        cascata+ricircolo', 'Range', Position1(1))
fun=@(optvar) duestadicascatarcy(1, optvar, dati, 1);
start1=surrogateopt(fun, lb5(1,:), ub5(1,:), [], [1 -1 0], 0);
[start1, amin]=patternsearch(fun, start1, [1 -1 0], 0, [], [], lb5(1,:), ub5(1,:));
fun=@(optvar) duestadicascatarcy(1, optvar, dati, 2, E_u, E_g, LHV);
opts=optimoptions('surrogateopt', 'MaxFunctionEvaluations', 200,
        'MinSurrogatePoints', 50, 'MinSampleDistance', 1e-8);
[start2, emin]=surrogateopt(fun, lb5(1,:), ub5(1,:), [], [1 -1 0], 0, [], [], opts);
c={'area specifica', amin, start1; 'consumo specifico', emin, start2};
writecell(c, filename, 'Sheet', '2 stadi in
        cascata+ricircolo', 'Range', Position2(1))
FitnessFunction=@(optvar) duestadicascatarcy(1, optvar, dati, 3, E_u, E_g, LHV);
opts=optimoptions('gamultiobj', 'PopulationSize', 500, 'InitialPopulationMatrix',
        [start1; start2]);
[optval5, objval5]=gamultiobj(FitnessFunction, 3,
        [1 -1 0], 0, [], [], lb5(1,:), ub5(1,:), [], opts);
[objval5(:,1), i]=sort(objval5(:,1));

```

```

objval5(:,2)=objval5(i,2);
optval5=optval5(i,:);
Heat_ex_1=duestadicascatarcy(1,optval5(1,:),dati,5,E_u,E_g,LHV,Tout,Pt);
Heat_ex_2=duestadicascatarcy(1,optval5(round(end/2),:),dati,5,E_u,E_g,
                                LHV,Tout,Pt);
Heat_ex_3=duestadicascatarcy(1,optval5(end,:),dati,5,E_u,E_g,LHV,Tout,Pt);
t=table(Paretofront,Optimumvar(1:3),'VariableNames',{'Fronte di
                                                    Pareto','variabili ottimizzate'});
writetable(t,filename,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position3(1))
writematrix([objval5,optval5],filename,'Sheet','2 stadi in
            cascata+ricircolo','Range',Position4(1))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3],'VariableNames',{'scambiatore'},
        'RowNames',{'Tin [°C] amin','Pt [MW/(kgCH4/s)] amin','Tin [°C]
                    p_medio','Pt [MW/(kgCH4/s)] p_medio','Tin [°C] emin','Pt
                    [MW/(kgCH4/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position5(1))
plot(objval5(:,1),objval5(:,2),'linewidth',2,'DisplayName','tipo 1')
title('Pareto Front (2 stadi in cascata con ricircolo)')
xlabel('area totale specifica [m^2/(kg_C_H_4/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_H_4/s)]')
legend show
hold on
for type=2:4
    c={'obiettivo','minimo',Optimumvar(type,1),Optimumvar(type,2),
        Optimumvar(type,3)};
    writecell(c,filename,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position1(type))
    fun=@(optvar) duestadicascatarcy(type,optvar,dati,1);
    opts=[optimoptions('particleswarm');
        optimoptions('particleswarm','SwarmSize',60,'FunctionTolerance',
            1e-7);
        optimoptions('particleswarm','SwarmSize',70,'FunctionTolerance',
            1e-7)];
    [start1,amin]=particleswarm(fun,3,lb5(type,:),ub5(type,:));
    fun=@(optvar) duestadicascatarcy(type,optvar,dati,2,E_u,E_g,LHV);
    [start2,emin]=particleswarm(fun,3,lb5(type,:),ub5(type,:),opts(type-1));
    c={'area specifica',amin,start1;'consumo specifico',emin,start2};
    writecell(c,filename,'Sheet','2 stadi in
                cascata+ricircolo','Range',Position2(type))
    FitnessFunction=@(optvar) duestadicascatarcy(type,optvar,dati,3,E_u,
        E_g,LHV);
    opts=optimoptions('gamultiobj','PopulationSize',500,
        'InitialPopulationMatrix',[start1; start2]);
    [optval5,objval5]=gamultiobj(FitnessFunction,3,[],[],[],[],
        lb5(type,:),ub5(type,:),[],opts);
    [objval5(:,1),i]=sort(objval5(:,1));
    objval5(:,2)=objval5(i,2);
    optval5=optval5(i,:);

```

```

Heat_ex_1=duetadicascatarcy(type,optval5(1,:),dati,5,E_u,
                                E_g,LHV,Tout,Pt);
Heat_ex_2=duetadicascatarcy(type,optval5(round(end/2),:),dati,5,
                                E_u,E_g,LHV,Tout,Pt);
Heat_ex_3=duetadicascatarcy(type,optval5(end,:),dati,5,E_u,E_g,
                                LHV,Tout,Pt);
t=table(Paretofront,Optimumvar(type,:), 'VariableNames',
        {'Fronte di Pareto','variabili ottimizzate'});
writetable(t,filename,'Sheet','2 stadi in
            cascata+ricircolo','Range',Position3(type))
writematrix([objval5,optval5],filename,'Sheet','2 stadi in
            cascata+ricircolo','Range',Position4(type))
t=table([Heat_ex_1;Heat_ex_2;Heat_ex_3], 'VariableNames',
        {'scambiatore'}, 'RowNames', {'Tin [°C] amin', 'Pt [MW/(kgCH4/s)]
        amin', 'Tin [°C] p_medio', 'Pt [MW/(kgCH4/s)] p_medio', 'Tin [°C]
        emin', 'Pt [MW/(kgCH4/s)] emin'});
writetable(t,filename,'WriteRowNames',true,'Sheet','2 stadi in
            cascata+ricircolo','Range',Position5(type))
plot(objval5(:,1),objval5(:,2), 'linewidth',2, 'DisplayName',
        ['tipo ',num2str(type)])
title('Pareto Front (2 stadi in cascata con ricircolo)')
xlabel('area totale specifica [m^2/(kg_C_H_4/s)]')
ylabel('consumo elettrico specifico [MW/(kg_C_H_4/s)]')
legend show
hold on
end
toc

```